POLSKA AKADEMIA NAUK
KOMITET ELEKTRONIKI I TELEKOMUNIKACJI

KWARTALNIK ELEKTRONIKI I TELEKOMUNIKACJI

ELECTRONICS AND TELECOMMUNICATIONS QUARTERLY

TOM 50 — ZESZYT 1

WARSZAWA 2004

ADRES REDAKCJI
00-665 Warszawa, ul. Nowowiejska 15/19 Politechnika, pok. 470
Instytut Telekomunikacji, Gmach im. prof. JANUSZA GROSZKOWSKIEGO

Dyżury Redakcji: poniedziałki i piątki, godz. 14–16
tel/fax (022) 660 77 37

Telefony domowe: Redaktora Naczelnego: 812 17 65
Zast. Red. Naczelnego: 826 83 41
Sekretarza Odpowiedzialnego: 633 92 52

Ark. wyd. 10,5 Ark. druk. 8,5	Podpisano do druku w lutym 2004 r.
Papier offset. kl. III 80 g. B-1	Druk ukończono w lutym 2004 r.

Skład, druk i oprawa: Warszawska Drukarnia Naukowa PAN
00-656 Warszawa, ul. Śniadeckich 8
Tel./fax: 628-87-77

Szanowni Autorzy

„Kwartalnik Elektroniki i Telekomunikacji" — Electronics and Telecommunications Quartery jest kontynuatorem tradycji powstałego 50 lat temu kwartalnika p.t. „Rozprawy Elektrotechniczne".

Kwartalnik jest czasopismem Komitetu Elektroniki i Telekomunikacji Polskij Akademii Nauk. Wydawany jest przez Warszawską Drukarnię Naukową PAN. Kwartalnik jest czasopismem naukowym, na którego łamach są publikowane artykuły i komunikaty prezentujące wyniki oryginalnych prac teoretycznych i doświadczalnych, a także przeglądowych. Związane są one z szeroko rozumianymi dziedzinami współczesnej elektroniki, telekomunikacji, mikroelektroniki, oproelektroniki, radiotechniki i elektroniki medycznej.

Autorami publikacji są wybitni naukowcy, znani specjaliści o wieloletnim doświadczeniu, a także młodzi badacze — głównie doktoranci.

Artykuły charakteryzują się oryginalnym ujęciem zagadnienia, interesującymi wynikami badań, krytyczną oceną teorii lub metod, omówieniem aktualnego stanu, lub postępu danej gałęzi techniki oraz omówieniem perspektyw rozwojowych. Sposób pisania matematycznej części artykułów zgodny jest z wytycznymi IEC (International Electronics Commision) oraz ISO (International Organization of Standarization).

Wszystkie publikowane w Kwartalniku artykuły są recenzowane przez znanych krajowych specjalistów, co zapewnia że publikacje te są uznawane jako autorski dorobek naukowy. Opublikowane w kwartalniku wyników prac naukowych zrealizowanych w ramach „GRANTów" Komitetu Badań Naukowych spełnia więc jeden z wymogów stawianych tym pracom.

Czasopismo dociera do wszystkich zajmujących się elektroniką i telekomunikacją krajowych ośrodków naukowych oraz technicznych, a także szeregu instytucji zagranicznych. Jest ponadto prenumerowane przez liczne grono specjalistów i biblioteki.

Każdy Autor otrzymuje bezpłatnie 20 egzemplarzy nadbitek swojego artykułu, co ułatwia przesłanie go do indywidualnych wybranych przez Autora osób i instytucji w kraju lub za granicą. Ułatwia to dodatkowo fakt, że w Kwartalniku są publikowane artykuły w języku angielskim.

Nadesłane do redakcji artykuły są publikowane w termini około pół roku, w przypadku sprawnj współpracy Autora z Redakcją. Wytyczne dla Autorów dotycząc formy publikacji są zamieszczone w zeszytach Kwartalnika, można je także otrzymać w siedzibie Redakcji.

Artykuły można dostarczać osobiście, lub pocztą pod adresem zamieszczanym na stronie redakcyjnej w każdym zeszycie.

Redakcja

SPIS TREŚCI

CONTENTS

KWARTALNIK ELEKTRONIKI I TELEKOMUNIKACJI, 2004, 50, z. 1, ss. 7–23

Algorytmy wyboru ścieżek LSP w sieciach MPLS

IRENEUSZ OLSZEWSKI

Wydział Telekomunikacji i Elektrotechniki
Akademia Techniczno-Rolnicza
Al. Prof. S. Kaliskiego 7
85-796 Bydgoszcz
e-mail: irek@mail.atr.bydgoszcz.pl

Otrzymano 2002.12.16
Autoryzowano 2003.03.27

W pracy przedstawiono algorytm wyboru ścieżek LSP w sieciach IP z protokołem MPLS. Pierwsza część zawiera wstęp, natomiast część druga zarys wieloprotokołowej komutacji etykietowanej z uwzględnieniem zalet istotnych z punktu widzenia inżynierii ruchu. W części trzeciej przytoczono matematyczne sformułowanie problemu optymalizacji oraz heurystyczny algorytm rozwiązujący ten problem. W tej części pracy zaproponowano również nowy algorytm, będący ulepszoną wersją algorytmu publikowanego w literaturze, a także wyniki symulacyjne, uzyskane po zastosowaniu prezentowanych algorytmów. W ostatniej części pracy przedstawiono wnioski końcowe.

Słowa kluczowe: algorytm, ścieżka LSP, protokół MPLS

1. WSTĘP

W pracy dokonano porównania algorytmów podanych w [1] i [4] oraz zaproponowano nowy algorytm wyboru ścieżek komutowanych etykietowo LSP (ang. Label Switched Path) w sieciach IP z protokołem MPLS (ang. *Multi-Protocol Label Switched*).

Rozważany problem dotyczy wyboru ścieżki LSP pomiędzy parą węzłów w sieci IP/MPLS, gdzie jedyną informacją, dostarczaną przez odpowiednie algorytmy [2], dla algorytmu wyboru ścieżki jest dostępne pasmo na poszczególnych łączach w sieci. W pracy przyjęto, podobnie jak w [4], że znany jest pewien zbiór par węzłów: ruter wejściowy — ruter wyjściowy, dla których prawdopodobieństwo odrzucenia ścieżki LSP jest mniejsze od prawdopodobieństwa odrzucenia ścieżki LSP dla pozostałych par węzłów w sieci. Dlatego też, nawet w ogólnym przypadku, kiedy każdy węzeł może

być ruterem wejściowym — wyjściowym, pewien podzbiór par węzłów będzie bardziej istotny od pozostałych par węzłów w sieci.

W pracy przyjęto, że w chwili pojawienia się zapotrzebowania na pasmo pomiędzy daną parą węzłów (ruterów brzegowych), generowane są żądania wyboru ścieżek LSP z gwarantowanym pasmem. Chwile pojawiania się żądań pomiędzy węzłami w poszczególnych parach węzłów są opisane procesem Poissona, zaś czas trwania ścieżek LSP jest wykładniczy, za wyjątkiem tych ścieżek LSP, które nie są rozłączane (ścieżki typu *long live*, Rozdział 3.3).

Algorytmy oparte na wyborze ścieżek LSP wzdłuż ścieżek najkrótszych, mierzonych liczbą łączy, są wystarczające do osiągnięcia połączenia, jednak nie zawsze pozwalają one w pełni wykorzystać zasoby sieci z punktu widzenia inżynierii ruchu. Głównym powodem jest fakt, że łącza na najkrótszych ścieżkach pomiędzy pewnymi parami węzłów: węzeł wejściowy — węzeł wyjściowy mogą być zatłoczone, podczas gdy łącza na alternatywnych ścieżkach pozostają wolne. Oznacza to, że zasoby sieci nie będą w sposób dostateczny wykorzystane, pomimo że istnieje pewien potencjał lepszego wykorzystania zasobów sieci w tej samej infrastrukturze sieci. Dlatego też, poszukiwanie algorytmów wyboru ścieżek LSP, które zwiększają wykorzystanie zasobów sieci jest w pełni uzasadnione.

Inżynieria ruchu jest związana zarówno z algorytmami wyboru ścieżek LSP pracującymi w trybie offline jak i online. W algorytmach offline przyjmuje się, że wszystkie ścieżki LSP, wraz z wielkościami pasma, są znane w trakcie realizacji tych algorytmów [9]. Celem działania tych algorytmów jest wybór ścieżek LSP w taki sposób, aby minimalizować kryterium nadrzędne, jakim jest na ogół koszt realizacji sieci [5] (analogicznie do problemu optymalizacji sieci z komutacją łączy). Jednakże, w praktyce jest prawdopodobne, że nowe żądania wyboru ścieżek LSP napłyną po realizacji grupy ścieżek LSP, lub też, wymagane zasoby istniejących ścieżek LSP mogą zmienić się w czasie. Można uniknąć problemu przyjęcia tych przyszłych ścieżek, jeśli istnieje możliwość przekierowania istniejących ścieżek LSP. Ponieważ algorytmy offline na ogół nie realizują realokacji zapotrzebowań będących już w realizacji, stąd też, gdy pewna grupa ścieżek LSP jest już zrealizowana i pojawiają się nowe żądania wyboru ścieżek LSP, kierowanie tych nowych ścieżek można realizować jedynie z użyciem algorytmów online.

Działanie algorytmów online zależy praktycznie od informacji stanu łączy osiąganych z różnych protokołów, np. OSPF [2] (ang. *Open Shortest Path First*). Podstawowa idea tych protokołów polega na tym, że każdy węzeł w sieci wysyła informację o stanie swoich łączy i skojarzonych z nimi atrybutów do wszystkich pozostałych węzłów. Ten mechanizm powoduje, że każdy węzeł w sieci zna topologię sieci. Topologia ta może być więc użyta przez poszczególne węzły do wypracowania własnych decyzji dotyczących następnego kroku dla danego węzła wyjściowego. Ścieżka LSP jest zestawiana z użyciem protokołu sygnalizacyjnego RSVP (ang. *Resource Reservation Protocol*). Ponieważ trudno uzyskać jest wielkość opóźnienia lub informację o zajętości buforów z protokołu sygnalizacyjnego [4], dlatego też, wybór ścieżek LSP nie może zależeć od

tych wielkości. Proponowany algorytm oraz algorytmy wyboru ścieżek prezentowane w pracach [1] i [4] bazują jedynie na stanie zajętości łączy.

2. PROTOKÓŁ MPLS

W tradycyjnych sieciach IP decyzja o tym gdzie należy przesłać otrzymany pakiet podejmowana jest niezależnie w każdym ruterze na podstawie analizy całego nagłówka i zgodnie ze stosowanym algorytmem wyboru drogi. Nagłówek zawiera na ogół znacznie więcej informacji, niż jest to niezbędne do wyznaczenia kolejnego odcinka drogi pakietu. Wyznaczenie takie składa się z dwóch faz: pierwsza z nich polega na podziale wszystkich możliwych pakietów na zbiór klas równoważności przekazywania FEC (ang. *Forwarding Equivalence Class*), natomiast druga, na wyznaczeniu następnego odcinka dla każdej z klas równoważności. Pakiety należące do tej samej klasy nie są rozróżnialne i zawsze kierowane są tą samą drogą. W każdym węźle sieci następuje odczytanie i analiza nagłówka pakietu, a następnie przypisanie go do jednej z klas równoważności [1], [3].

W przypadku stosowania wieloprotokołowej komutacji etykietowanej pakiet jest przydzielany do klasy równoważności tylko raz — w ruterze brzegowym, w chwili, gdy zostaje wprowadzony do sieci. Numer klasy koduje się w postaci krótkiej etykiety o stałej długości. Otrzymaną w ten sposób etykietę przesyła się razem z pakietem wzdłuż ścieżki LSP skojarzonej z daną klasa równoważności [1], [3]. W każdym węźle (ruterze) nie trzeba już analizować całego nagłówka, wystarczy odczytanie etykiety. Wieloprotokołowa komutacja etykietowana zwiększa szybkość przetwarzanych pakietów oraz pozwala na stosowanie wyrafinowanych procedur ich przekazywania. MPLS umożliwia zróżnicowane traktowanie pakietów wprowadzanych do sieci w różnych punktach sieci, tzn. pakietowi wprowadzonemu do sieci w danym węźle można przypisać inną etykietę, niż pakietowi wprowadzonemu do sieci w innym węźle. Technika MPLS pozwala na określenie całej ścieżki LSP w węźle początkowym (ruterze brzegowym), dzięki czemu możliwe jest omijanie przeciążonych węzłów w sieci, optymalizowanie rozpływu ruchu w sieci oraz zwiększenie wykorzystania zasobów sieci.

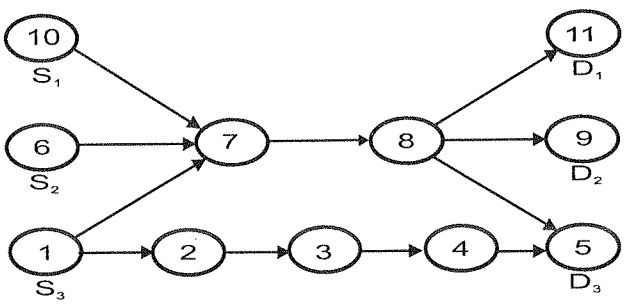
3. ALGORYTM WYBORU ŚCIEŻKI

W tej pracy, podobnie jak w [1] i [4], rozważamy jedynie wybór ścieżek LSP z gwarantowanym pasmem. Nie oznacza to jednak, że inne metryki ścieżek, takie jak opóźnienie oraz straty pakietów nie mogą być rozważane.

Najbardziej znanym algorytmem jest algorytm min-hop [4], który wybiera najkrótszą ścieżkę mierzoną liczbą łączy pomiędzy danym węzłem wejściowym oraz danym węzłem wyjściowym. Należy zaznaczyć, że podczas wyznaczania ścieżki LSP są brane pod uwagę tylko te łącza, które mają dostępne pasmo nie mniejsze niż pasmo żądanej ścieżki LSP.

Bardziej wyrafinowany algorytm, który wykorzystuje dostępne pasmo łączy łączących poszczególne węzły sieci, zaproponowano w [7]. Najkrótsza ścieżka pomiędzy daną parą węzłów jest wyszukiwana na podstawie wag łączy, które są wykładniczą funkcją natłoku na tych łączach. Jeśli dostatecznie krótka ścieżka nie istnieje, ścieżka LSP jest odrzucona.

Przedstawiony w zarysie algorytm min-hop bazuje na topologii sieci oraz dostępnej pojemności (ang. *residual capacity*) łączy, natomiast nie bierze pod uwagę rozmieszczenia węzłów wejściowych oraz wyjściowych. Jeżeli ścieżka LSP zostanie wybrana niezależnie od położenia tych węzłów względem położenia pozostałych, potencjalnych węzłów wejściowych oraz wyjściowych, może wystąpić interferencja. W celu wyjaśnienia tego zjawiska przytoczony zostanie następujący przykład [4], zilustrowany na rys. 1. Przyjmijmy, że istnieją trzy potencjalne pary węzeł wejściowy — węzeł wyjściowy. Na-



Rys. 1. Interferencja ścieżek w przypadku wyboru ścieżki 1-7-8-5 przez algorytm Min-hop

Fig. 1. Interference of the paths by Min-hop algorithm in the case of the choice of path 1-7-8-5

leży wybrać drogę pomiędzy parą węzłów (S_3 i D_3), przy założeniu, że każda krawędź posiada tylko jedną jednostkę przepustowości. Algorytm Min-hop wyznacza ścieżkę: 1-7-8-5. Zauważmy, że wybór tej ścieżki wyklucza możliwość wyboru jakiejkolwiek ścieżki pomiędzy parami węzłów (S_1 i D_1) oraz (S_2 i D_2). Natomiast w przypadku wyboru ścieżki 1-2-3-4-5, pomimo że dłuższej, istniałaby możliwość wyboru ścieżki pomiędzy parami węzłów (S_1 i D_1) lub (S_2 i D_2).

Z przedstawionego przykładu jasno wynika, że maksymalna wielkość pasma, jaką można zrealizować pomiędzy daną parą węzłów (S_1 i D_1), jak i innymi parami węzłów, jest ograniczona od góry poprzez wartość maksymalnego przepływu. Zatem, realizacja D jednostek pasma pomiędzy parą węzłów (S_1 i D_1) spowoduje spadek wartości maksymalnego przepływu o D jednostek pomiędzy tą parą węzłów oraz w przypadku interferencji, spadek wartości maksymalnego przepływu o wartość nie większą niż D jednostek pasma pomiędzy innymi parami węzłów. Dlatego też, wybór ścieżki LSP pomiędzy daną parą węzłów musi być zrealizowany w taki sposób, aby w jak najmniejszym stopniu interferować z potencjalnymi ścieżkami LSP (żądaniami wyboru ścieżek, które napłyną w przyszłości) pomiędzy innymi parami węzłów: węzeł wejściowy — węzeł wyjściowy. Bardziej formalnie, wybrana ścieżka LSP pomiędzy daną

parą węzłów np. (S_1 i D_1) powinna maksymalizować minimalną wartość z maksymalnych przepływów, określonych na dostępnych pojemnościach łączy sieci, pomiędzy pozostałymi parami węzłów.

3.1. MATEMATYCZNE SFORMUŁOWANIE PROBLEMU

Poniżej przytoczono ogólne sformułowanie problemu optymalizacji [4]. Niech $G(N, L, B)$ jest siecią, gdzie: N — zbiór węzłów (ruterów), L — zbiór łuków (łączy), B — wektor (o m -współrzędnych) wielkości pasma łączy; Ponad to: n — liczba węzłów, m — liczba łuków. P — jest wyróżnionym zbiorem par węzłów (węzeł wejściowy-węzeł wyjściowy). Wszystkie żądania wyboru ścieżek LSP pojawiają się pomiędzy parami węzłów ze zbioru P. M — macierz incydencji węzły — gałęzie (kolumna odpowiadająca łukowi (v, w) ma +1 w wierszu v oraz –1 w wierszu w), x^{sd} — zmienna (wektor o m współrzędnych) odpowiadająca parze $(s, d) \in P$, θ_{sd} — wartość maksymalnego przepływu pomiędzy (s, d) określona na dostępnych pojemnościach łączy w sieci, e^{sd} — wektor (o n -współrzędnych) przyjmuje +1 na pozycji d i –1 na pozycji s, R — wektor (o m - współrzędnych) dostępnych pojemności łączy.

Należy zrealizować zapotrzebowanie D jednostek pasma (LSP) pomiędzy węzłami (ruterami) a i b tak, aby maksymalizować najmniejszą wartość maksymalnego przepływu z dla zbioru par $P \setminus (a, b)$.

Problem ten (MAX-MIN-MAX) można sformułować jako problem zadania programowania liniowego całkowitoliczbowego:

Problem optymalizacji 1.

$$\max z$$

$$Mx^{sd} = \theta_{sd} e^{sd} \qquad \forall (s, d) \in P \setminus (a, b) \tag{1}$$

$$Mx^{ab} = -De^{ab} \tag{2}$$

$$x^{sd} + x^{ab} \leqslant R \qquad \forall (s, d) \in P \setminus (a, b) \tag{3}$$

$$z \leqslant \alpha_{sd} \theta_{sd} \qquad \forall (s, d) \in P \setminus (a, b) \tag{4}$$

$$x^{sd} \geqslant 0 \qquad \forall (s, d) \in P \tag{5}$$

$$x^{ab} \in \{0, d\}^m \tag{6}$$

Równanie 1 jest problemem maksymalnego przepływu dla każdej pary węzłów ze zbioru $P \setminus (a, b)$. Równanie 2 wymusza, że należy zrealizować D jednostek pasma pomiędzy parą węzłów (a, b). Równanie 3 zapewnia, że przepływy na łączach nie mogą przekroczyć dostępnych pojemnościach łączy w sieci. Równanie 4 ogranicza (maksymalizowaną) wartość funkcji celu poprzez wprowadzone wagi α_{sd}, które odzwierciedlają ważność relacji (s, d) w sieci i mogą być wprowadzane przez administratora sieci.

Wreszcie, równanie 6 zapewnia, że zapotrzebowanie pomiędzy parą węzłów (a, b) jest kierowane na pojedynczą ścieżkę.

Tylko wartość określająca funkcję celu jest najmniejszą wartością maksymalnego przepływu. Inne przepływy nie wpływają na wartość optymalnego rozwiązania. Przykładowo, jeśli dla czterech par: węzeł wejściowy — węzeł wyjściowy wektory rozwiązań odpowiednio wynoszą $(10, 5, 15, 20)$ oraz $(15, 5, 20, 30)$, wówczas rozwiązaniem optymalnym, maksymalizującym najmniejszą wartość maksymalnego przepływu w jednym i drugim przypadku jest wartość 5. Jednakże z praktycznego punktu widzenia drugie rozwiązanie jest korzystniejsze. Dlatego też, alternatywnym rozwiązaniem jest maksymalizacja ważonej sumy maksymalnych przepływów [4].

Problem optymalizacji 2.

$$\max \Sigma_{(s,d)\in P\setminus(a,b)} \alpha_{sd}\theta_{sd} \tag{7}$$

Przy graniczenia (1, 2, 3, 5 i 6)

gdzie: α_{sd} — jest wagą pary (s, d).

W pracy [4] wykazano, że powyższy problem (WSUM-MAX) jest problemem NP-trudnym (*NP – hard*).

3.2. OGÓLNA IDEA ALGORYTMU

Ponieważ przedstawiony problem optymalizacji (problem 2) jest problemem NP-trudnym, dlatego też, użycie algorytmów heurystycznych rozwiązujących ten problem jest w pełni uzasadnione. W pracy [4] zaproponowano następujący algorytm, pracujący w trybie online [1].

Definicja

Łuk nazywamy krytycznym dla danej pary węzłów (s, d), jeśli należy do minimalnego przekroju C_{sd} określonego dla tej pary węzłów. Minimalny przekrój jest obliczany z macierzy dostępnych pojemności łączy w sieci.

- Dane wejściowe: Graf $G(N, L)$, wektor R dostępnych pojemności łączy; węzły a i b, pomiędzy którymi należy zrealizować D jednostek pasma.
- Dane wyjściowe: Ścieżka pomiędzy węzłami a i b, pomiędzy którymi należy zrealizować D jednostek pasma.

Algorytm 1 (MIRA);

1. Wyznacz maksymalny przepływ dla wszystkich $(s, d) \in P \setminus (a, b)$;
2. Oblicz zbiór łączy krytycznych C_{sd} dla wszystkich $(s, d) \in P \setminus (a, b)$;
3. Oblicz wagi:
$$\omega(l) = \sum\nolimits_{(s,d):l\in C_{sd}} \alpha_{sd} \qquad \forall l \in L; \tag{8}$$
4. Usuń wszystkie łącza, które mają dostępne pasmo mniejsze niż D jednostek i uformuj sieć zredukowaną;
5. Wyznacz najkrótszą ścieżkę w zredukowanej sieci używając $\omega(l)$ jako wagi łącza l;

6. Skieruj D jednostek zapotrzebowania z węzła a do b wzdłuż najkrótszej ścieżki i dokonaj aktualizacji wektora R dostępnych pojemności w sieci. ☑

Komentarza w tym algorytmie wymaga krok 3 i 4. Wagi tych łączy (krok 3), które nie są krytyczne dla żadnej pary węzłów (s,d) przyjmują bardzo małą wartość. Podstawienie to zapewnia, że długość wybieranej ścieżki LSP w kroku 5, mierzona liczbą łączy, jest również minimalizowana. Z kolei, sieć zredukowana dana jest grafem $G(N,L')$, i powstaje poprzez usunięcie tych łuków z grafu $G(N,L)$, dla których dostępne pojemności, określone poprzez wektor R, są mniejsze od D jednostek pasma poszukiwanej ścieżki LSP.

Wartość α_{sd} może być tak dobrana, aby odzwierciedlać ważność pary (s,d) z punktu widzenia operatora sieci; W niniejszej pracy dokonano badania tego algorytmu dla trzech różnych wag:

1. Wagi $\alpha_{sd} = 1$ dla wszystkich par $(s,d) \in P$. W tym przypadku $\omega(l)$ reprezentuje liczbę par węzeł wejściowy — węzeł wyjściowy, dla których łącze l jest krytyczne;
2. Wagi α_{sd} odwrotnie proporcjonalne do wartości maksymalnego przepływu, tzn. $\alpha_{sd} = 1/\theta_{sd}$, gdzie θ_{sd} jest wartością maksymalnego przepływu pomiędzy parą (s,d);
3. Wagi $\alpha_{sd} = 1 - 1/\theta_{sd}$.

Uzyskane wyniki z użyciem wag określonych w punkcie 1 oznaczono jako mira 1, w punkcie 2 jako mira 2 oraz w punkcie 3 jako mira 3 (Rozdział 3.3). Wadą przytoczonego algorytmu jest brak zależności pomiędzy różnym łączami w trakcie określania wag $\omega(l)$.

Drugim algorytmem (Algorytm 2), prezentowanym poniżej jest algorytm proponowany przez autora. Najkrótsza ścieżka pomiędzy daną parą węzłów (a,b) jest wyszukiwana na podstawie wag krawędzi (łączy), które są odwrotnie proporcjonalne do dostępnych pojemności łączy w sieci zredukowanej. Algorytm ten, podobnie jak algorytm 1, został skonstruowany do pracy w trybie online.

Algorytm 2

1. Usuń wszystkie łącza, które mają dostępne pasmo mniejsze niż D jednostek i uformuj sieć zredukowaną;
2. Oblicz wagi:

$$\omega(l) = 1/R(l) \qquad \forall l \in L'; \tag{9}$$

3. Wyznacz najkrótszą ścieżkę w zredukowanej sieci używając $\omega(l)$ jako wagi łącza l;
4. Skieruj D jednostek zapotrzebowania z węzła a do b wzdłuż najkrótszej ścieżki i dokonaj aktualizacji wektora R dostępnych pojemności w sieci. ☑

Do wyznaczenia najkrótszej ścieżki w tym algorytmie (krok 3), podobnie jak w algorytmie 1 (krok 5), można użyć algorytmu Dijkstry lub algorytmu Forda-Bellmana [7]. W celu wyjaśnienia, czy proponowany algorytm bierze pod uwagę zjawisko interferencji (Rozdział 3), rozważony zostanie przykład na podstawie rys. 1. Przyjmijmy, podobnie jak poprzednio, że istnieją trzy potencjalne pary: węzeł wejściowy-węzeł wyjściowy. Należy wybrać ścieżkę, o jednostkowym paśmie, pomiędzy parą węzłów

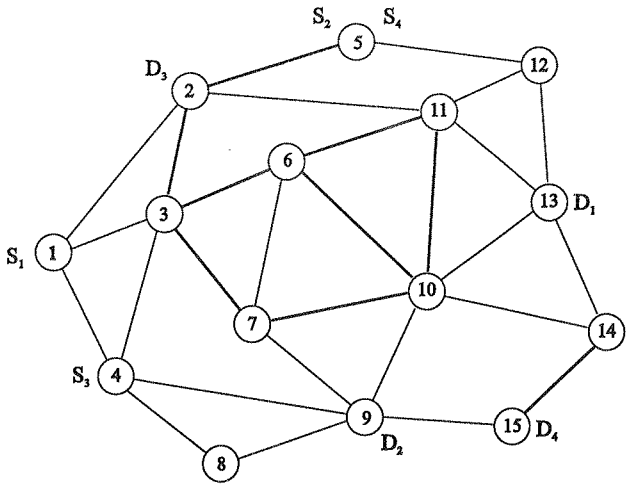
(S_3 i D_3), przy założeniu, że łuki: 1-7, 7-8 oraz 8-5 posiadają tylko jedną jednostkę przepustowości, natomiast łuki: 1-2, 2-3, 3-4 oraz 4-5 posiadają dwie jednostki przepustowości. Algorytm Min-hop wyznaczy ścieżkę: 1-7-8-5 wykluczając możliwość wyboru jakiejkolwiek ścieżki pomiędzy parami węzłów (S_1 i D_1) oraz (S_2 i D_2), natomiast algorytm 2 wybierze ścieżkę 1-2-3-4- 5, o łącznej wadze równej 2, umożliwiając wybór ścieżki pomiędzy parami węzłów: (S_1 i D_1) lub (S_2 i D_2). Dopiero wybór kolejnej ścieżki: 1-7-8-5, o łącznej wadze równej 3, pomiędzy parą węzłów (S_3 i D_3) przez algorytm 2 uniemożliwi wybór ścieżki pomiędzy parą węzłów (S_1 i D_1) lub (S_2 i D_2). Zatem, jak wynika z tego przykładu, algorytm 2 bierze pod uwagę (do pewnego stopnia) zjawisko interferencji.

Złożoność obliczeniowa algorytmów.

Funkcja złożoności obliczeniowej dla obydwóch algorytmów wyznaczona zostanie przy założeniu, że potencjalna liczba par węzeł wejściowy — węzeł wyjściowy (moc zbioru P) jest proporcjonalna do n^2. Złożoność obliczeniowa algorytmu 1 w kroku 1 wynosi $O(n^2(n^2\sqrt{m}))$, gdzie $O(n^2\sqrt{m})$ jest złożonością obliczeniową algorytmu maksymalnego przepływu [4]. Funkcja złożoności obliczeniowej w kroku 2 wynosi $O(m^2)$ [4], w kroku 3 $O(m)$, w kroku 4 $O(m)$, w kroku 5 $O(n^2)$ [8], i wreszcie w kroku 6 $O(n)$. Zatem, funkcja złożoności obliczeniowej całego algorytmu 1, dla wyznaczenia pojedynczej ścieżki LSP, będzie rzędu $O(n^4\sqrt{m})$. Z kolei, funkcja złożoności Algorytmu 2 jest również wielomianowa i wynosi: $O(m + m + n^2 + n) \approx O(n^2)$. Mniejsza złożoność obliczeniowa Algorytmu 2 w porównaniu ze złożonością obliczeniową Algorytmu 1 wynika stąd, że w Algorytmie 2 pominięto wyznaczanie maksymalnego przepływu oraz zbioru łączy krytycznych C_{sd} dla wszystkich par węzłów $(s, d) \in P \setminus (a, b)$ (Kroki 1 i 2 w Algorytmie 1).

3.3. UZYSKANE WYNIKI

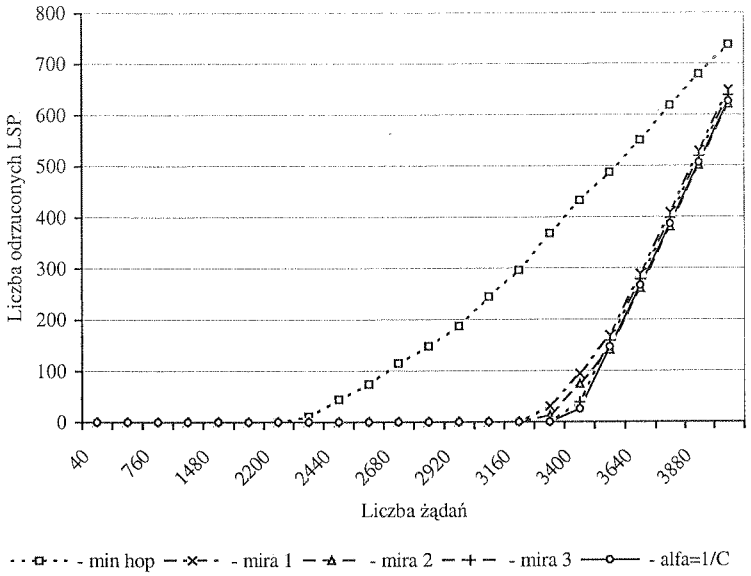
Weryfikacji algorytmów dokonano dla sieci zawierających 15 oraz 20 węzłów. Na rys. 2, pokazano strukturę topologiczną badanej sieci, zawierającą 15 węzłów (ruterów), połączonych łączami o przepustowości 1200 (linie cienkie) oraz 4800 (linie grube) jednostek [4]. Symulacji pracy sieci dokonano metodą Monte Carlo [6], gdzie ścieżka LSP jest przyjmowana lub odrzucana zgodnie ze stosowanym algorytmem (Algorytm 1, Algorytm 2 lub algorytm min-hop). W sieci tej wyróżniono cztery pary węzłów: (S_1 i D_1), (S_2 i D_2), (S_3 i D_3) oraz (S_4 i D_4), pomiędzy którymi generowane są żądania wyboru ścieżek LSP z pasmem symetrycznym (formalnie dwóch ścieżek LSP, skierowanych w przeciwnych kierunkach) z zakresu 1–4 jednostek przepustowości. Należy zaznaczyć, że można również uwzględniać żądanie wyboru ścieżki LSP z pasmem asymetrycznym. Badanie algorytmu 1 przeprowadzono dla trzech różnych współczynników: $\alpha_{sd} = 1$ (krzywa na wykresie oznaczona jako mira 1), $\alpha_{sd} = 1/\theta_{sd}$ (mira 2) oraz $\alpha_{sd} = 1 - 1/\theta_{sd}$(mira 3), gdzie θ_{sd} jest wartością maksymalnego przepływu pomiędzy parą węzłów (s, d). Dla celów porównawczych przedstawiono również wyniki uzyskane przy pomocy algorytmu min-hop (zajmo-



Rys. 2. Struktura topologiczna badanej sieci

Fig. 2. Topology structure of the investigated network

wana najkrótsza ścieżka mierzona liczbą krawędzi). Z kolei, wyniki uzyskane przy pomocy proponowanego algorytmu oznaczono jako alfa = 1/C. Na rys. 3 przedstawiono całkowitą liczbę odrzuconych ścieżek LSP pomiędzy wszystkimi wyróżnionymi parami węzłów w funkcji liczby żądań, przy założeniu, że zestawiane ścieżki nie są rozłączane (żądania *long live*). Z rysunku tego wynika, że liczba odrzuconych ścieżek LSP przez algorytm 1 jest prawie taka sama bez względu na zastosowane α_{sd} w tym algorytmie. Zatem, wynika stąd, że sugerowane wagi w [4] nie mają praktycznie wpływu na uzyskane wyniki przy pomocy tego algorytmu. Nieco mniejszą liczbę odrzuconych ścieżek LSP uzyskano przy pomocy proponowanego algorytmu 2 (alfa = 1/C). Z kolei, na rys. 4 pokazano całkowity dostępny przepływ w sieci pomiędzy wszystkimi wyróżnionymi parami węzłów w funkcji liczby żądań. Wyniki te również potwierdzają brak wpływu współczynników α_{sd} na pracę algorytmu 1. Z rysunku tego wynika, że zarówno Algorytm 1 (bez względu na stosowane współczynniki) jak i Algorytm 2 zachowują znacznie większy sumaryczny przepływ do 3280 żądań od algorytmu min-hop. Na rys. 5 i 6 przedstawiono odpowiednio liczbę odrzuconych ścieżek LSP oraz dostępny przepływ w relacji 1-13 (S_1 — D_1) w funkcji liczby żądań. Z rys. 5 wynika, że Algorytm 2 odrzuca nieco mniejszą liczbę ścieżek LSP, jednakże jest to okupione mniejszym dostępnym przepływem pomiędzy rozważaną parą węzłów. Ponieważ Algorytm 1 zachowuje się podobnie dla każdego z użytych współczynników, dlatego też, przyjęto w dalszej części prezentować ten algorytm ze współczynnikiem $\alpha_{sd} = 1/\theta_{sd}$. Na rys. 7 pokazano całkowitą liczbę odrzuconych ścieżek LSP dla wszystkich wyróżnionych par węzłów w pięciu różnych przebiegach symulacyjnych dla rozważanych algorytmów. W ba-

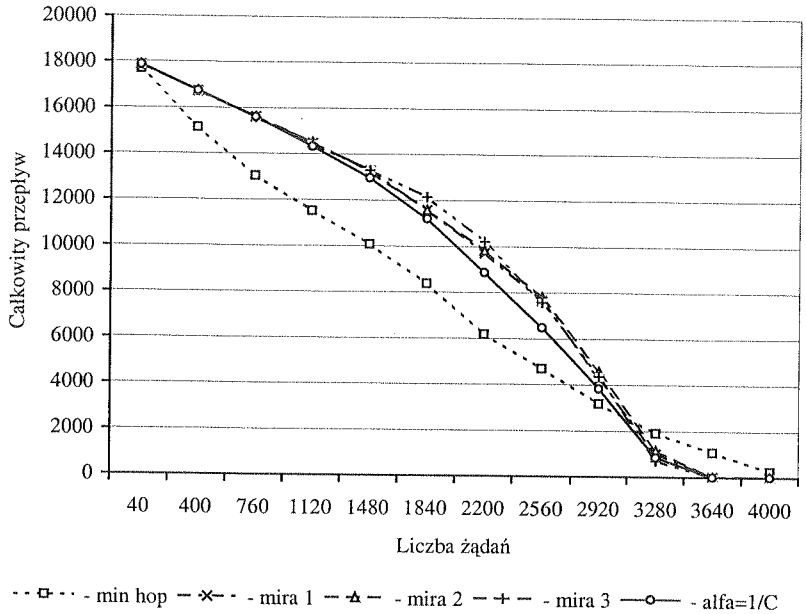


Rys. 3. Całkowita liczba odrzuconych ścieżek LSP

Fig. 3. Total number of rejected LSP paths

daniu tym przyjęto, że do obsługi przez sieć skierowanych zostaje 4000 żądań, bez rozłączania uprzednio zrealizowanych ścieżek LSP. Z rysunku tego wynika, że Algorytm 2 odrzuca najmniej żądań dla każdej symulacji. Z kolei, na rys. 8. przedstawiono czas działania każdego z badanych algorytmów, dla przyjętej liczby żądań, w pięciu przebiegach symulacyjnych (tych samych na podstawie, których sporządzono wykres na rys. 7). Z rysunku tego wynika, że Algorytm 2 jest znacznie szybszy od Algorytmu 1 (funkcja złożoności obliczeniowej Algorytmu 2 jest rzędu $O(n^2)$). Z kolei, na rys. 9. przedstawiono dynamiczne zachowanie się badanych algorytmów. Wyniki są rejestrowane po uprzednim zestawieniu 4000 ścieżek LSP. W trakcie rejestracji wyników przyjęto, że zestawione ścieżki LSP podlegają rozłączeniom (analogicznie jak w sieci z komutacją łączy). Badania przeprowadzono dla 10 niezależnych przebiegów symulacyjnych, generując po 8000 żądań dla każdego z nich. Z rysunku tego wynika, że liczby odrzuconych ścieżek LSP zarejestrowanych w poszczególnych przebiegach w przypadku użycia Algorytmów 1 i 2 są porównywalne i znacznie mniejsze od liczb odrzuconych ścieżek LSP uzyskanych z użyciem algorytmu min-hop. Na rys. 10. przedstawiono te same wyniki w odniesieniu do pary węzłów (1-13).

Analogiczne badania przeprowadzono dla sieci zawierających 20 węzłów. W przykładowej sieci, zawierającej 20 węzłów połączonych 41 łączami (w tym 32 o przepustowości 1200 jednostek oraz 9 łączami o przepustowości 4800 jednostek przepustowości)

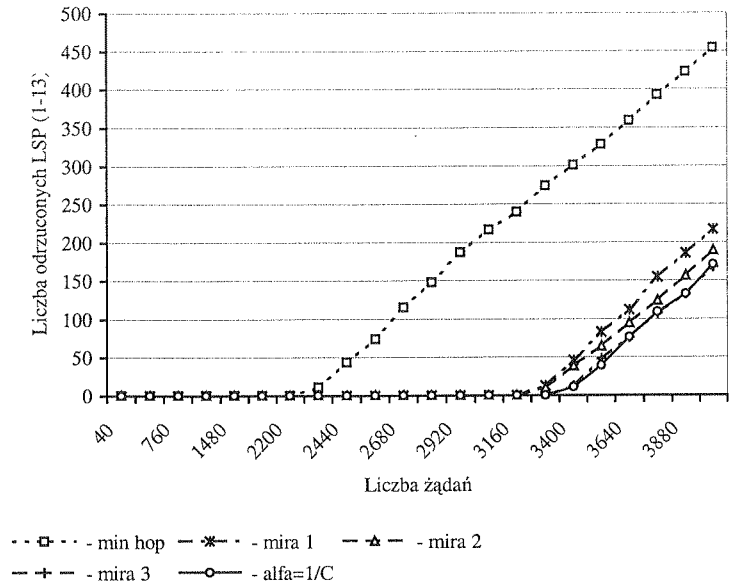


Rys. 4. Całkowity dostępny przepływ pomiędzy wszystkimi parami węzłów

Fig. 4. Total available flow between all pairs of nodes

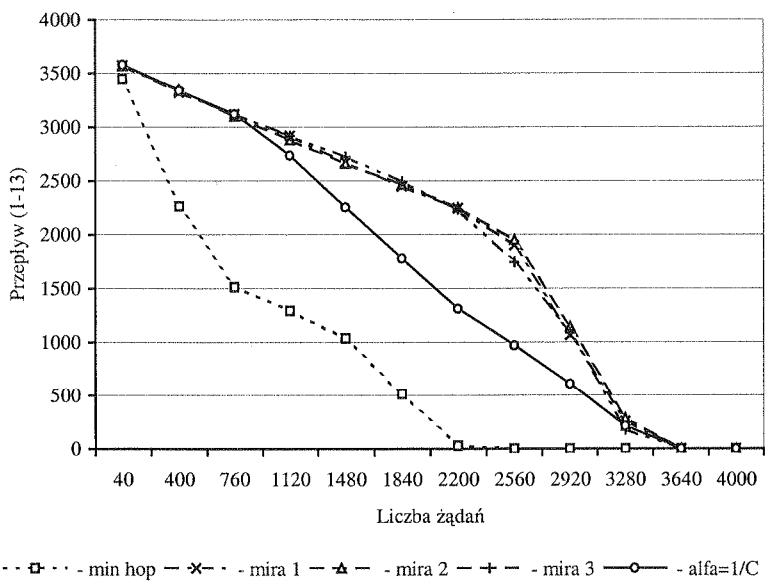
wyróżniono, podobnie jak dla sieci 15 węzłowych, cztery pary węzłów, pomiędzy którymi generowane są żądania wyboru ścieżek LSP z pasmem symetrycznym z zakresu 1-4 jednostek przepustowości. W badaniu tym przyjęto, że do obsługi przez sieć skierowanych zostaje 4000 żądań, bez rozłączania uprzednio zrealizowanych ścieżek. Na rys. 11 przedstawiono całkowitą liczbę odrzuconych ścieżek LSP pomiędzy wyróżnionymi parami węzłów w funkcji liczby żądań, natomiast na rys. 12 przedstawiono czas działania każdego z badanych algorytmów dla przyjętej liczby żądań. Wyniki z rys. 11 oraz rys. 12 przedstawiono dla pięciu niezależnych przebiegów symulacyjnych.

Uzyskane wyniki dowodzą, że Algorytm 2 odrzuca mniejszą liczbę ścieżek LSP od Algorytmu 1. Dla przytoczonych sieci, stosunki liczby ścieżek odrzuconych do liczby żądań, w warunkach statycznych (tylko żądania *long live*) oraz liczbie żądań równej 4000, w pięciu przebiegach symulacyjnych odpowiednio wynoszą: dla 15 węzłów oraz Algorytmu 1 (mira 2) od 15.8% do 17.2%, dla 15 węzłów oraz Algorytmu 2 (proponowany) od 14.8% do 16.1%, dla 20 węzłów oraz Algorytmu 1 od 6.2% do 7.6% i wreszcie dla 20 węzłów oraz Algorytmu 2 od 4.2% do 4.7%. Ponadto, należy podkreślić, że proponowany algorytm jest znacznie szybszy od Algorytmu 1. Duża szybkość działania Algorytmu 2 wynika z jego niewielkiej złożoności obliczeniowej, rzędu $O(n^2)$, co jest istotną zaletą tego algorytmu.



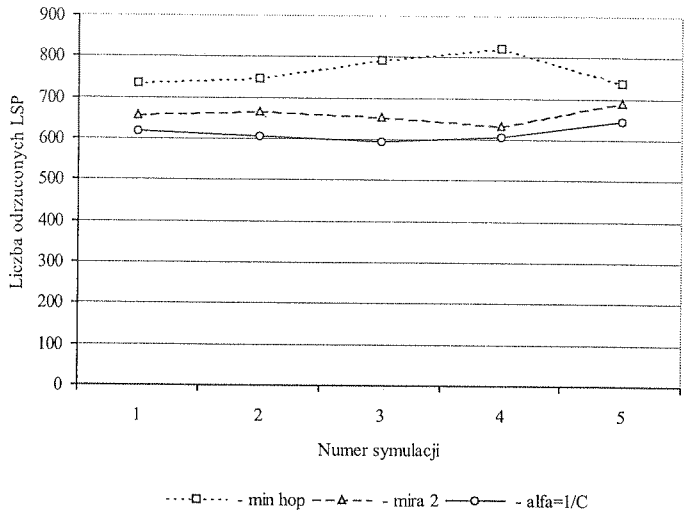
Rys. 5. Liczba odrzuconych LSP w relacji (1-13)

Fig. 5. Number of rejected LSP paths between (1-13)



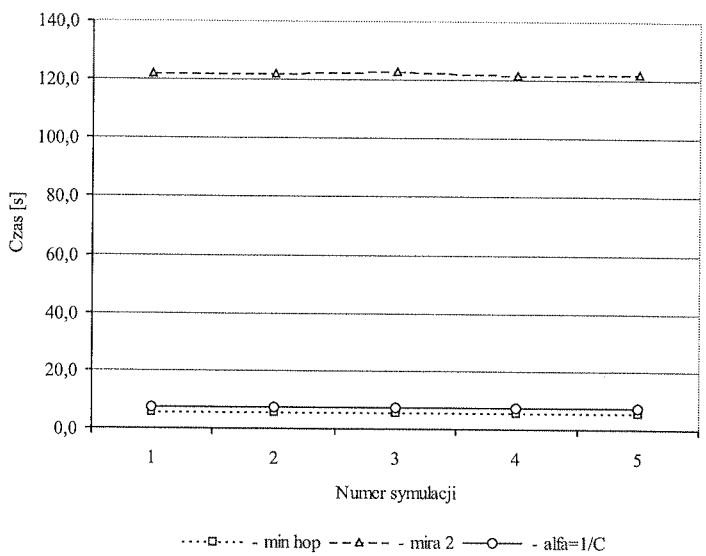
Rys. 6. Dostępny przepływ w relacji (1-13)

Fig. 6. Available flow between (1-13)



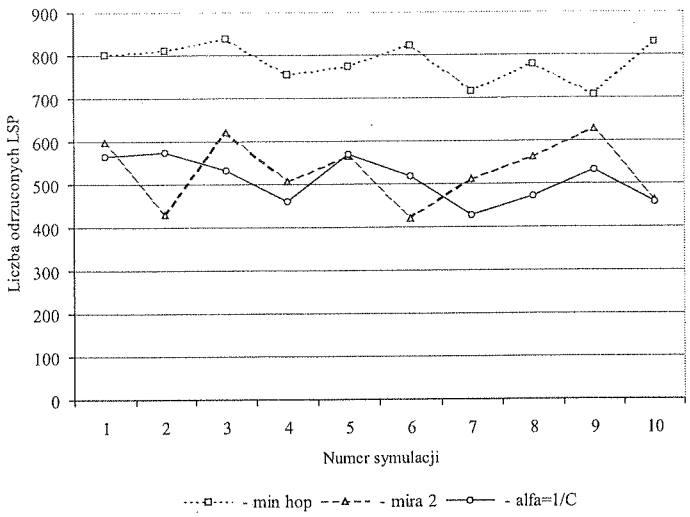
Rys. 7. Całkowita liczba odrzuconych LSP w pięciu symulacjach dla algorytmów: min hop, mira 2 oraz alfa=1/C

Fig. 7. Total number of rejected LSP paths for five experiments by the use of algorithms: min hop, mira 2 and alfa=1/C



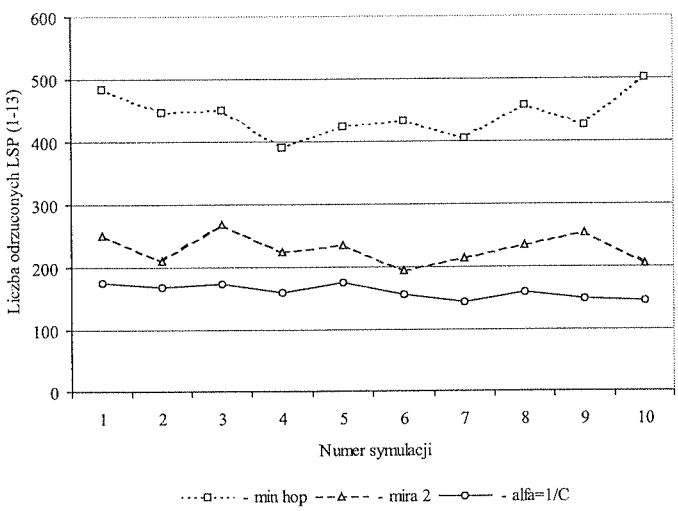
Rys. 8. Czas symulacji w pięciu przebiegach dla algorytmów: min hop, mira 2 oraz alfa=1/C

Fig. 8. Running time for five experiments by the use of algorithms: min hop, mira 2 and alfa=1/C



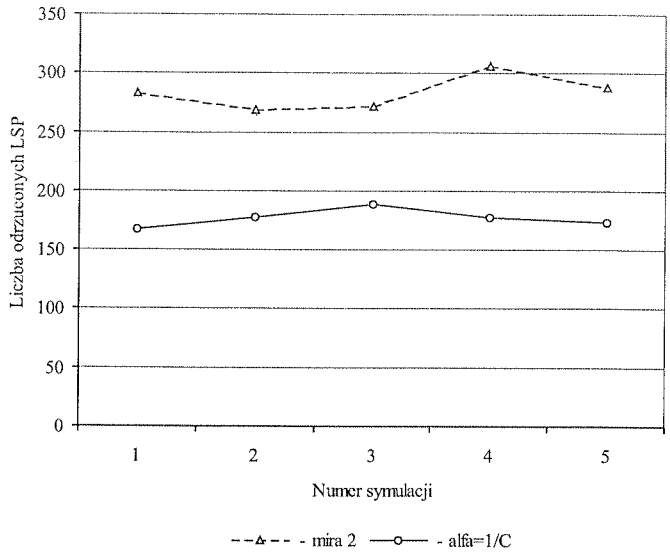
Rys. 9. Całkowita liczba odrzuconych LSP w dziesięciu przebiegach symulacji dla algorytmów: min-hop, mira 2 oraz alfa=1/C

Fig. 9. Total number of rejected LSP paths for ten experiments by the use of algorithms: min-hop, mira 2 and alfa=1/C



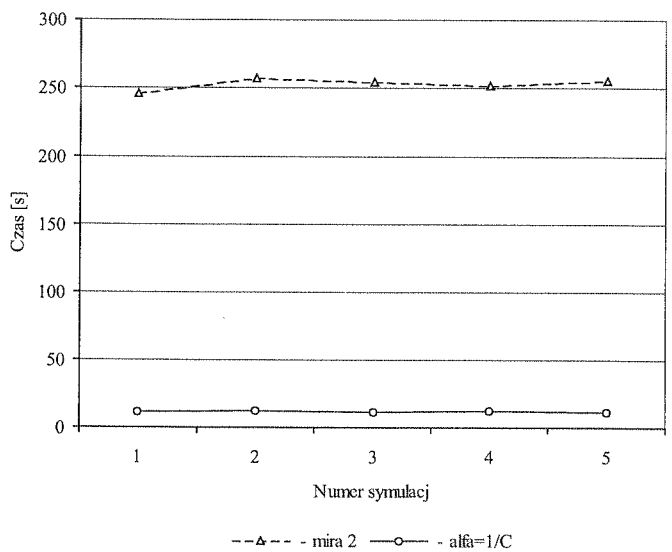
Rys. 10. Liczba odrzuconych LSP w relacji (1-13) w dziesięciu przebiegach symulacji dla algorytmów: min hop, mira 2 oraz alfa=1/C

Fig. 10. Number of rejected LSP paths between (1-13) for ten experiments by the use of algorithms: min hop, mira 2 and alfa=1/C



Rys. 11. Całkowita liczba odrzuconych LSP w pięciu symulacjach dla algorytmów: mira 2 oraz alfa=1/C

Fig. 11. Total number of rejected LSP paths for five experiments by the use of algorithms: mira 2 and alfa=1/C



Rys. 12. Czas symulacji w pięciu przebiegach dla algorytmów: mira 2 oraz alfa=1/C

Fig. 12. Running time for five experiments by the use of algorithms: mira 2 and alfa=1/C

4. WNIOSKI KOŃCOWE

W pracy zaproponowano algorytm wyboru ścieżek LSP w sieciach IP/MPLS oraz dokonano porównania tego algorytmu z algorytmem prezentowanym w literaturze. Rozważany problem w niniejszej pracy jest istotny nie tylko w sieciach z protokołem MPLS, lecz również w innych aplikacjach sieciowych żądających dynamicznego przydziału pasma, np. w sieciach optycznych. Uzyskane wyniki dowodzą, że liczba odrzuconych ścieżek LSP przez Algorytm 2 jest mniejsza od liczby odrzuconych ścieżek przez Algorytm 1, zarówno w warunkach statycznych (tylko ścieżki *long live*) jak i w warunkach dynamicznych (tj. przy założeniu, że zestawione ścieżki mogą być rozłączane). Niewątpliwą przewagą Algorytmu 2 nad Algorytmem 1 jest jego szybkość działania. Głównym czynnikiem decydującym o funkcji złożoności obliczeniowej Algorytmu 2 jest wywoływany każdorazowo algorytm Dijkstry. Większa szybkość działania tego algorytmu pozwala na jego implementację w sieci o znacznie większej liczbie węzłów niż w przypadku sieci, w której zastosowano Algorytm 1. Bezsprzeczną wadą badanych algorytmów jest fakt, że wagi łączy w obydwóch przypadkach (Algorytm 1 oraz Algorytm 2) są traktowane niezależnie od siebie. Dlatego też, dalsze badania powinny prowadzić do wprowadzenia zależności pomiędzy wagami poszczególnych łączy w sieci.

5. BIBLIOGRAFIA

1. P. Aukia, M. Kodialam, P. Koppol, T.V. Lakshman, H. Sarin, B. Suter: *RATES: a Server for MPLS Traffic Enginering.* IEEE Network, March/ April 2000.
2. R. Guerin, D. Williams, A. Orda: *QoS Routing Mechanisms and OSPF Extensions.* Proceedings of Globecom 1997.
3. A. Jajszczyk: *Dalekosiężne sieci telekomunikacyjne bez ATM i SDH.* Przegląd Telekomunikacyjny, Rocznik LXXII, nr 6/1999.
4. M. Kodialam, T. V. Lakshman: *Minimum Interference Routing with Applications to MPLS Traffic Engineering.* Proc. INFOCOM, Mar. 2000.
5. A. Mysłek: *Zastosowanie metod dokładnych i heurystycznych w topologicznym projektowaniu MPLS.* KST, Bydgoszcz 2001.
6. M. Pióro: *Podstawy projektowania cyfrowych sieci telekomunikacyjnych.* Poznań, Wydawnictwa EFP. 1995.
7. S. Plotkin: *Competitive Routing of Virtual Circuits in ATM Networks.* IEEE J. Selected Areas in Communications. 1995, vol 13, no 6, pp.1128-1136.
8. M. M. Sysło, N. Deo, J. S. Kowalik: *Algorytmy optymalizacji dyskretnej.* Warszawa PWN, 1995.
9. X. Xiao, A. Hannan, B. Bailey: *Traffic Engineering with MPLS in the Internet.* IEEE Network, March/April 2000.

I. OLSZEWSKI

THE ALGORITHMS OF CHOICE OF THE LSP PATHS IN THE MPLS NETWORKS

Summary

In this paper the algorithms of choice of the LSP paths in the Multi- Protocol Label Switched (MPLS) networks has been presented. In the first and second part the outline of this protocol has been pointed out. Especially the advantage of the traffic engineering has been taken. In the third part the mathematical optimization problem as well as the heuristic algorithm that solves this problem has been shown. In this part the new algorithm being the improved version of algorithm published in [4] has been proposed. Input data of the proposed algorithm are topological structure of the network and residual capacity of the links that connect the nodes of the network. For the pair of nodes the shortest path is chosen with respect to weights of links, which are function of residual capacity of these links. In the third part of the paper the simulation results has been shown. On the base of obtained results the analysis of algorithms in static and dynamic conditions were carried out. Obtained results prove, that the number of rejected LSP paths by the algorithm proposed in this paper are comparable with number of LSP paths rejected by the algorithm proposed in [4]. However the running time of proposed algorithm is much shorter and comparable with running time of the minimum hop algorithm. In the final part the conclusions has been drowed.

Keywords: algorithm, LSP path, protocol MPLS.

KWARTALNIK ELEKTRONIKI I TELEKOMUNIKACJI, 2004, 50, z. 1, ss. 25-32

Model kolejkowy dla analizy Sieci Inteligentnej

NATALIA KRYVINSKA[1], EUGENIUSZ JASZCZYSZYN[2]

[1]*Institute of Communication Networks, Vienna University of Technology*
Favoritenstrasse 9/388, A-1040, Vienna
e-mail:Natalia.Kryvinska@tuwien.ac.at;
[2]*Instytut Radioelektroniki, Politechnika Warszawska*
Nowowiejska 15/19, 00-665, Warszawa
e-mail: e.jaszczyszyn@ire.pw.edu.pl;

Otrzymano 2002.11.21
Autoryzowano 2003.05.03

Wraz z wdrażaniem usług Sieci Inteligentnych (angl. *Intelligent Network*) pojawia się potrzeba oszacowania wydajności systemu w przypadku wykorzystania platformy z rozproszoną inteligencją. Przedstawiony artykuł omawia model kolejkowy dla analizy połączeń telefonicznych (wywołań) Sieci Inteligentnej. Sieć Inteligentna została przedstawiona w postaci sieci kolejkowej przy założeniu, że łączna liczba abonentów (tj. SSP) jest stała, co powoduje, że mamy zamkniętą sieć kolejkową. Rozproszoną architekturę Sieci Inteligentnej przedstawiono w postaci modelu kolejkowego ze źródłem skończonym — M/M/1/K/K. W artykule oszacowano czas oczekiwania odpowiedzi, oraz przedstawiono wyniki analizy i odpowiednie zależności.

Słowa kluczowe: sieć inteligentna, zamknięta sieć kolejkowa, M/M/1/K/K model kolejkowy

1. WPROWADZENIE

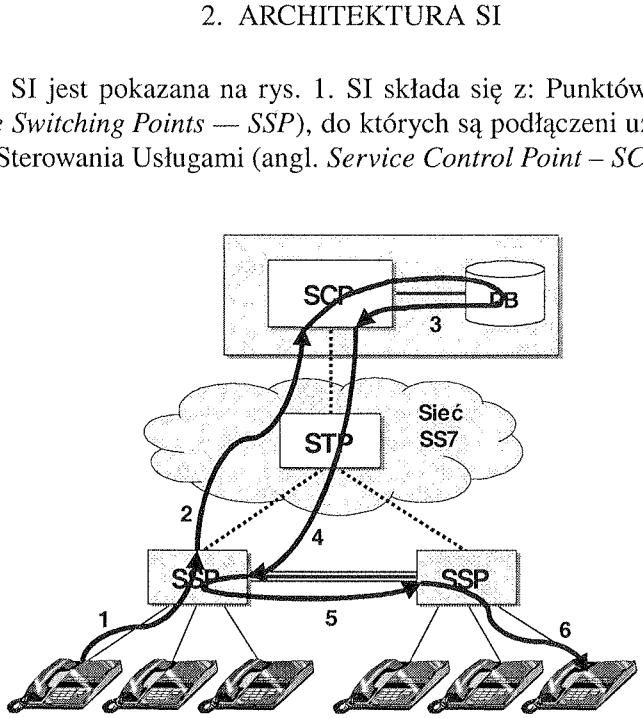
Z wdrażaniem usług Sieci Inteligentnych pojawia się potrzeba projektowania sieci z rozproszoną inteligencją, przystosowanych do jakichkolwiek zmian w żądaniach abonentów. W miarę tego jak przemysł telekomunikacyjny ewoluuje, użytkownicy coraz częściej oczekują natychmiastowego dostępu do usług. Dla prawidłowego działania systemu koniecznym jest aby: czas odpowiedź/reakcja systemu był minimalny; była zainstalowana wystarczająca pojemność w wypadku awarii, a także istniała możliwość kontroli dla zarządzania w sytuacjach wyjątkowych, które ciągle stanowią zagrożenie dla całości (integralności) sieci. Scenariusz usługi łączy żądanie usługi, fizyczną topologię sieci, przepływ komunikatów (wiadomości) sygnalizacyjnych, odwzorowanie

jednostek funkcjonalnych na składniki fizyczne, a także trasowanie (jako część procesu projektowania sieci dla upewnienia się, że wymagania wydajność/działania są zaspokojone) [1; 9; 10; 13; 14].

Z punktu widzenia wydajności systemu, najbardziej znaczącą rolę odnośnie Sieci Inteligentnej (SI) odgrywają: dystrybucja inteligentności sieciowej, a także tworzenie nowych usług (możliwe przy pomocy architektury rozproszonej). Podczas, gdy konwencjonalne wywołanie telefoniczne jest przetwarzane wewnątrz węzła komutacyjnego, w środowisku SI wywołanie wymaga przetwarzania zespołowego kilku elementów sieci, połączonych przez sieć sygnalizacyjną. Ta zasadnicza zmiana powoduje, że koniecznym jest optymalne projektowanie sieci dla upewnienia się, że Sieć Inteligentna zapewnienia użytkownikom usługi z wysoką wydajnością [2; 8; 11].

2. ARCHITEKTURA SI

Struktura SI jest pokazana na rys. 1. SI składa się z: Punktów Komutacji Usług (angl. *Service Switching Points — SSP*), do których są podłączeni użytkownicy końcowi; Punktów Sterowania Usługami (angl. *Service Control Point – SCP*), które mieszczą



Rys. 1. Architektura SI

Fig. 1. The IN architecture

Bazy Danych (angl. *DataBases — DB*) i wykonują rozszerzone sterowanie usługami, a także z Sieci Sygnalizacyjnej numer 7 (angl. *Signaling System No.7*), która przenosi komunikaty między SCP i SSP przez Punkty Transmisji Sygnałów (angl. *Signal Transfer Points — STP*).

Usługa SI typowo wywołuje oddziaływania wzajemne między SSP i niektórymi innymi węzłami SI dla: 1) kontroli sterowaniem wywołania i połączenia telefonicznego; 2) współpracy z użytkownikiem; 3) nadzoru zdarzeń w interfejsach sygnalizacyjnych. Na przykład, SSP może czasowo zawiesić obróbkę wywołania aby odesłać zapytanie do SCP (gdzie są wykonywane usługi bazowe, takie jak uwierzytelnienie, translacja numerów, wybór trasy, alternatywna opłata itd.). SSP może przekazywać komunikaty sygnalizacyjne do węzła IP (angl. *Intelligent Peripheral — IP*) w celu przeprowadzenia interakcji z użytkownikiem [2; 3; 12].

3. ZAMKNIĘTA SIEĆ KOLEJKOWA JAKO MODEL SIECI INTELIGENTNEJ

SI może być przedstawiona jako sieć kolejkowa, gdzie całkowita liczba użytkowników (czyli SSP) jest stała (żaden nowy użytkownik nie ma prawa dostępu do sieci i żaden obecny użytkownik nie ma prawa opuszczenia sieci). Sieci tego typu nazywają się zamknięte. Zamknięte sieci mogą być analizowane przy pomocy łańcuchów Markowa. A stan ustalony dystrybucji zajętości ma kształt iloczynu (angl. *product form*) z założenia podobnego do istniejącego w sieciach otwartych.

Rozpatrzmy sieć składającą się z K kolejek dla której:

- Nie ma zewnętrznych źródeł i odbiorników dla użytkowników;
- Jest $K < \infty$ użytkowników w sieci;
- Jest tylko jeden server dla każdej kolejki. Czas obsługi w i-tej kolejce jest niezależną wykładniczą zmienną losową μ_i. Czasy obsługi w różnych kolejkach są wzajemnie niezależne;
- Trasowanie w sieci jest losowe. Prawdopodobieństwo, że użytkownik odchodzi z kolejki i do kolejki j jest $r_{i,j}$.

Taka sieć nazywana jest *Zamkniętą Siecią Jacksona*. Stan w takiej sieci opisuje równanie:

$$\underline{n} = (n_1, n_2, \ldots, n_i, \ldots, n_K) \tag{1}$$

gdzie n_i — liczba użytkowników w kolejce „i". Prawdopodobieństwo stanu ustalonego, że system znajduje się w stanie $\underline{n}$ jest oznaczone jako $p(\underline{n})$.

Równanie równowagi globalnej dla zamkniętej sieci Jacksona może być ustalone według balansu między przepływem w sieci (całkowita przepustowość) w określonym stanie i przepływem w sieci poza pewnym stanem. Równanie równowagi globalnej w stanie $\underline{n}$ można przedstawić w postaci

$$\sum_{i=1}^{K} \mu_i p(\underline{n}) = \sum_{i=1}^{K} \sum_{j=1}^{K} \mu_i r_{i,j} p(\underline{n} - \underline{1}_j + \underline{1}_i). \tag{2}$$

W kategoriach równań dla strumienia ruchu możemy otrzymać rozwiązanie równania równowagi globalnej w postaci wielomianu:

$$p(\underline{n}) = p(\underline{0}) \prod_{i=1}^{K} \left(\frac{\theta_i}{\mu_i}\right)^{n_i}, \tag{3}$$

gdzie

$$p(\underline{0}) = \left\{ \sum_{\underline{n}} \left[\prod_{i=1}^{K} \left(\frac{\theta_i}{\mu_i}\right)^{n_i} \right] \right\}^{-1}. \tag{4}$$

Załóżmy, że θ_i jest średnią przepustowością kolejki j, którą można przedstawić w postaci

$$\theta_i = \sum_{i=1}^{K} r_{j,i}\theta_j, \qquad i = 1, 2, \ldots, K \tag{5}$$

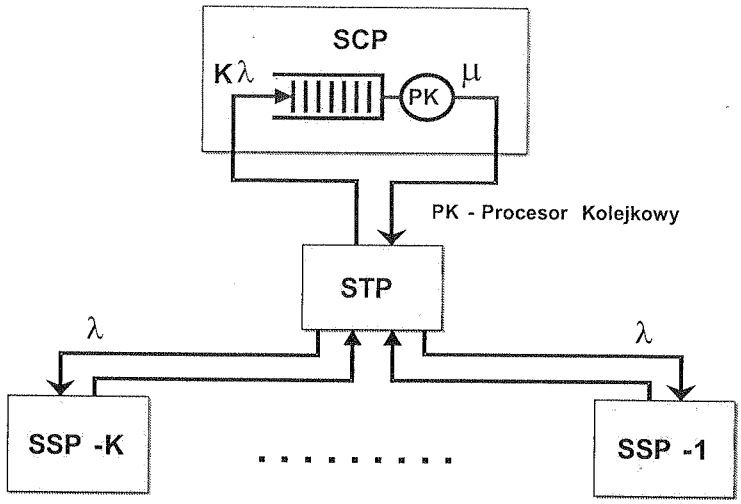
i które jest równaniem ruchu w sieci [4].

4. IMPLEMENTACJA SYSTEMU KOLEJKOWEGO M/M/1/K/K W SI

Tradycyjne usługi konwencjonalnej Sieci Telefonicznej (angl. *Public Switched Telephone Network — PSTN*) są realizowane przez scenariusz usługi w lokalnym węźle komutacyjnym. Wydajność tych usług jest ograniczona architekturą sieci i wydajnością składników usług wewnątrz węzła. SI ma architekturę rozproszoną, w której scenariusz usługi jest wykonywany zespołowo przez różne elementy sieci, które mogą być rozproszone geograficznie [2; 13; 14].

Na rys. 2 pokazano rozproszoną architekturę SI w postaci modelu z ograniczoną liczbą źródeł — czyli M/M/1/K/K. Taki model jest także znany jako *model naprawy maszyn*, lub model kolejek cyklicznych. W tym kontekście, w systemie krąży K zadań. System składa się z K SSP i Procesora Centralnego (angl. *Central Processor Unit — CPU*) z kolejką roboczą. Żądanie usługi SI wysyłane jest z SSP do SCP po pewnym czasie oczekiwania, który ma rozkład wykładniczy. Po przetwarzaniu w SCP odpowiedź wraca do SSP, który wchodzi w nową fazę oczekiwania. Wejściowe i wyjściowe komunikaty transakcji są interpretowane jako jedna złożona usługa. Także, czas oczekiwania i czas przetwarzania w SCP są brane pod uwagę jako średni czas działania [5].

Szybkość, z jaką żądania nadchodzą do SCP jest równa szybkości z którą żądania opuszczają system (system jest w stanie równowagi). Okres czasu, w którym szczególny SSP jest w stanie oczekiwania, jest stosunkiem średniego czasu oczekiwania $(1/\lambda)$ do średniego czasu podróży z powrotem $(T + 1/\lambda)$. Każdy z K SSP generuje w stanie oczekiwania żądania z szybkością λ na sekundę. Z innej strony okres czasu (kiedy SCP zajęty) jest równy $(1 - p_0)$, w ciągu którego szybkość wyjściowa jest równa μ. W ten sposób średnia szybkość wyjściowa zadań jest rów-



Rys. 2. Rozproszona architektura SI, jako model M/M/1/K/K

Fig. 2. The IN distributed architecture as a M/M/1/K/K model

na $[\mu(1 - p_0)]$. Przyrównując szybkości wejściową i wyjściową otrzymamy równanie w postaci:

$$K\lambda \frac{1/\lambda}{T + 1/\lambda} = \mu(1 - p_0). \tag{6}$$

Z (6) otrzymujemy równanie dla obliczenia średniego czasu w systemie T [6; 7]:

$$T = \frac{K/\mu}{1 - p_0} - \frac{1}{\lambda} \tag{7}$$

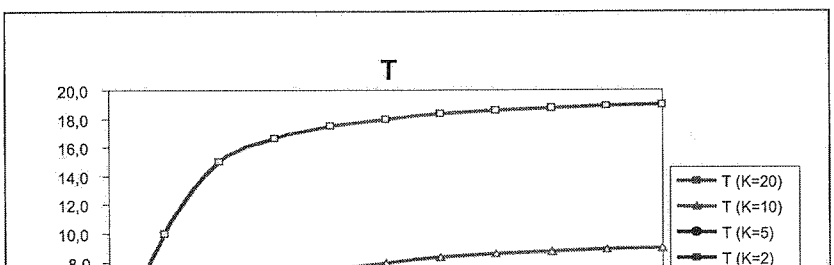
Wyniki numeryczne oczekiwanego czasu odpowiedzi, który jest sumą czasu oczekiwania w kolejce i czasu obsługiwania (średni czas w systemie — T), są podane w tabeli 1. Na rys. 3 jest pokazany czas T dla kilku wartości K, przy założeniu, że czas obsługi ma rozkład wykładniczy [5].

Tabela 1

Średni czas spędzenia zadania w systemie

The average time in system

λ	$T(K = 20)$	$T(K = 10)$	$T(K = 5)$	$T(K = 2)$	$T(K = 1)$
0.001	1.01933	1.00906	1.00401	1.00100	1.0
0.1	10.0375	2.73208	1.46630	1.09091	1.0
0.2	15.0000	5.18729	1.99171	1.16670	1.0
0.3	16.6670	6.68335	2.47526	1.23077	1.0
0.4	17.5000	7.50216	2.87479	1.28571	1.0
0.5	18.0000	8.00038	3.19048	1.33333	1.0
0.6	18.3333	8.33342	3.43741	1.37500	1.0
0.7	18.5714	8.57145	3.63177	1.41176	1.0
0.8	18.7500	8.75001	3.78677	1.44444	1.0
0.9	18.88889	8.88889	3.91225	1.47368	1.0
0.999	18.9990	8.99900	4.01445	1.49975	1.0



Rys. 3. Średni czas spędzenia zadania w systemie

Fig. 3. The average time in the system

5. PODSUMOWANIE

Analiza i modelowanie zagadnień wydajności systemu są istotnymi narzędziami dla projektowania i eksploatacji SI, i mogą być wykorzystywane na wszystkich etapach istnienia i działania usług SI. Proste, przybliżone modele są potrzebne na wczesnych etapach dla wyjawiania głównych problemów działania systemu. Problemy te mają wpływ na określenie architektury sieci jeszcze do początku eksploatacji sieci, kiedy

koszt naprawiania błędów stanie się zbyt duży. Narzędzia projektowe realizują szybkie stworzenie prototypu, pozwalając użytkownikom przejść przez trzy znaczące fazy: przewidywanie, projektowanie i porównanie. Opracowanie nowych metod dla szybkiej analizy, tworzenie nowych standardów SI, a także zbliżenie procesów analizy i tworzenia nowych usług są najbardziej interesującymi dla przyszłego szerokiego wdrożenia usług SI oraz i usług sieci konwergujących.

6. BIBLIOGRAFIA

1. R. Ackeley, A. Elvidge, T. Ingham, J. Shepherdson: *Network Intelligence — Performance by Design*. IEICE Transactions on Communication, vol. E80-B, No. 2, February 1997, pp. 219-229.
2. J. Yan D. MacDonald: *Teletraffic Performance in Intelligent Network Services*. Proceeding of the 14th International Teletraffic Congress, Antibes, Juan-les-Pins, France, June 1994, pp. 357-366.
3. E. Masuda, T. Mishima, N. Takaya, K. Nakai, M. Hirano: *A Large-Capacity Service Control Node Architecture Using Multicasting Access to Decentralized Databases in the Advanced Intelligent Network*. IEICE Transactions on Communication, vol. E84-B, No. 10, October 2001, pp. 2768-2780.
4. D. Bertsekas, R. Gallager: *Data Networks* — Second edition. Prentice-Hall, 1992.
5. M. H. van Hoorn: *Algorithms and Approximations for Queueing Systems*. PhD thesis, Vrije Universiteit te Amsterdam, Mathematical Center, Amsterdam, 1983.
6. L. Kleinrock: *Queueing Systems*. Volume II: *Computer Applications*. Wiley-Interscience, 1986.
7. L. Kleinrock, R. Gail: *Solution Manual for Queueing Systems*. Volume II: *Computer Applications*. Technology Transfer Institute, 1986.
8. ITU-T Q.1201 Recommendation Principles of Intelligent Network Architecture.
9. ITU-T Q.1202 Recommendation Intelligent Network — Service Plane Architecture.
10. ITU-T Q.1203 Recommendation Intelligent Network — Global Functional Plane.
11. ITU-T Q.1204 Recommendation Intelligent Network — Distributed Functional Plane.
12. ITU-T Q.1205 Recommendation Intelligent Network — Physical Plane.
13. ITU-T Q.1211-1219 Recommendations Intelligent Network Capability Set 1 — CS-1.
14. ITU-T Q.1220-1229 Recommendations Intelligent Network Capability Set 2 — CS-2.

N. KRYVINSKA, Y. YASHCHYSHYN

INTELLIGENT NETWORK ANALYSIS BY CLOSED NETWORK QUEUING MODELS

Summary

With increasing deployment of IN services, design and engineering of network intelligence platforms to accommodate the ever-changing and growing demands of customers, presents a rich market of opportunities and challenges. As telecommunications industry evolves, customers are increasingly coming expecting instantaneous access to service providers, together with transparency to network failures. System performance dictates that response times need to be minimized, sufficient redundant capacity to be installed in case of failure and controls embedded within the design to manage the exceptional situations that continually threaten network integrity. The service scenario mixes service demand, physical network

topology, signaling message flows, the mapping of functional entities to physical components, and routing as part of the network design process. For the system performance aspect, the most significant changes due to intelligent network are the distribution of network intelligence and the new services made possible by this distributed architecture. In IN environment, a call involves the cooperative processing of several network elements connected by a signaling network. This fundamental change possesses some new challenges to teletraffic experts to ensure that IN networks are designed to provide customers' services with good performance. This paper discusses a queuing system model for the performance analysis of IN call processing. The intelligent network is presented as a network of queues where the total number of customers (e.g., SSPs) is fixed, thus forming a closed queuing network. The IN distributed architecture is modeled as a finite source queuing model — M/M/1/K/K. The expected response time for that model is analyzed and computed. The numerical results and the corresponding curves are provided. And, related to open questions, future work is summarized.

Keywords: Intelligent Network, Closed Queuing Network, M/M/1/K/K model.

KWARTALNIK ELEKTRONIKI I TELEKOMUNIKACJI, 2004, 50, z. 1, ss. 33–48

Obszar zastosowania metody przewidywania nadchodzących danych do korekcji interferencji międzysymbolowej

JERZY KISILEWICZ, ARKADIUSZ GRZYBOWSKI

Katedra Systemów i Sieci Komputerowych
Wydział Elektroniki Politechniki Wrocławskiej
ul. Wybrzeże Wyspiańskiego 27, 50-370 Wrocław
e-mail: kisilewicz@zssk.pwr.wroc.pl

Otrzymano 2002.10.09
Autoryzowano 2003.05.27

W pracy opisano obszar zastosowania metody korekcji interferencji międzysymbolowej przy użyciu korektora z decyzyjnym sprzężeniem zwrotnym bez korektora wstępnego. Korektor wstępny zastąpiono przewidywaniem przyszłych danych. Decyzje podejmuje się tak, aby maksymalizować prawdopodobieństwo odbieranych danych. Przeprowadzono symulację transmisji przy użyciu opisanego korektora, tradycyjnego korektora z decyzyjnym sprzężeniem zwrotnym i z korektorem wstępnym. Porównano częstości występowania błędów w obecności białego szumu gaussowskiego dla dwóch klas kanałów. Symulowano kanały o dwóch i czterech symetrycznych próbkach interferencyjnych. Określono kanały, dla których opisany korektor zapewniał mniejszą stopę błędów.

Słowa kluczowe: transmisja danych, korektor transwersalny, decyzyjne sprzężenie zwrotne

1. WPROWADZENIE

Współczesne modemy wyposażone są w korektory interferencji międzysymbolowej, które dopasowują charakterystyki indywidualnych kanałów telekomunikacyjnych do potrzeb transmisji danych. Począwszy od szybkości transmisji 9,6 kb/s korektory takie są konieczne. Do najczęściej rozważanych korektorów należą: filtr transwersalny oraz korektor z decyzyjnym sprzężeniem zwrotnym, który wymaga wstępnego skorygowania kanału. Zwykle korektorami wstępnymi są filtry transwersalne. Problematyka odbioru danych w systemach zaopatrzonych w takie korektory jest przedmiotem roz-

ważań wielu autorów np. [1, 2, 5, 9, 13, 17, 18], zaś algorytmy obliczania korektorów transwersalnych opisano w [6, 10].

Istnieją kanały, które bardzo trudno korygują się za pomocą filtru transwersalnego. Przykłady takich kanałów podaje A. Dąbrowski w [8]. Dodatkową wadą korektorów w postaci filtru transwersalnego jest to, że stosunek mocy sygnału do mocy szumu (SNR) na wyjściu korektora jest mniejszy niż na jego wejściu [1], co ma szczególne znaczenie, gdy wartości SNR są niewielkie [1, 2, 18]. Ponadto korektor wstępny wprowadza do systemu transmisji danych dodatkowe opóźnienie.

Powyższe argumenty uzasadniają potrzebę poszukiwania metod użycia korektora z decyzyjnym sprzężeniem zwrotnym w układzie pozbawionym korektora wstępnego. W pracach [11, 12] opisano metodę, w której układ decyzyjny określa najbardziej prawdopodobne wartości odbieranych danych na podstawie znajomości nieskorygowanych próbek interferencyjnych oraz ocen odchyłek wartości próbek odebranego sygnału od ich możliwych nominalnych wartości. W pracy [11] opisano algorytm odbioru uwzględniający jedną nieskorygowaną wstępnie próbkę interferencji, natomiast w [12] uogólniono ten algorytm na większą liczbę nieskorygowanych próbek. W tych pracach przedstawiono wyniki eksperymentów, w których dla wybranych kanałów i różnych wartości SNR na wyjściu kanału porównano elementowe stopy błędów uzyskane w systemie z proponowanym korektorem oraz w systemach, jakimi są filtr transwersalny oraz korektor z decyzyjnym sprzężeniem zwrotnym i korektorem wstępnym. Wyniki te nie zawsze okazywały się najlepsze dla proponowanego algorytmu.

Celem tej pracy jest określenie, dla jakich kanałów i jakich wartości SNR zastosowanie algorytmu opisanego w [11] i [12] zmniejsza stopę błędów przy założeniu jednej i dwóch nieskorygowanych próbek interferencyjnych.

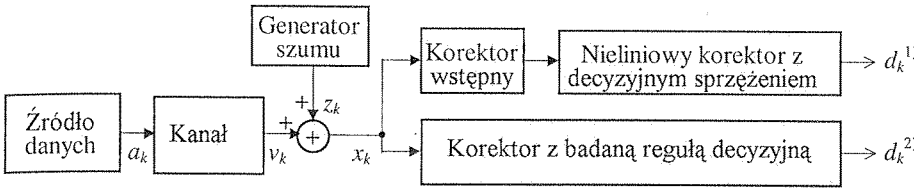
2. ORGANIZACJA EKSPERYMENTU

Aby określić obszary, w których badany algorytm [11, 12] znajdzie zastosowanie przeprowadzono eksperymenty obliczeniowe, w których symulowano pracę systemu złożonego ze źródła danych, kanału z zakłóceniami oraz dwóch porównywanych korektorów:

— korektora z decyzyjnym sprzężeniem zwrotnym i z liniowym korektorem wstępnym,
— korektora z decyzyjnym sprzężeniem zwrotnym i układem detekcji realizującym algorytmy opisane w [11 i 12].

System ten pokazano na rys. 1.

Za źródło sekwencji danych binarnych posłużył generator pseudoprzypadkowych szesnastobitowych liczb całkowitych. Poszczególne bity tych liczb wprowadzano na wejście kanału w postaci symboli o dwóch amplitudach równych ±1 (+1 dla bitu „1" oraz −1 dla bitu „0"). W eksperymencie symulowano biały szum gaussowski, choć ważniejszymi czynnikami powodującymi przekłamania są inne rodzaje zakłóceń, jak np. szum impulsowy i krótkie przerwy transmisji [8]. Wartości próbek białego szumu



Rys. 1. Schemat symulowanego systemu

Fig. 1. Block diagram of the simulated system

gaussowskiego o zadanej mocy losowane były za pomocą generatora liczb o rozkładzie normalnym. Założono przy tym wzajemną niezależność danych i zakłóceń.

Kanał symulowany był za pomocą liniowego filtru transwersalnego, który wprowadzał zadaną interferencję międzysymbolową. Do obliczonych wartości próbek v_k na wyjściu kanału dodawano wygenerowane wartości próbek szumu z_k. Próbki $x_k = v_k + z_k$ zaszumionego sygnału na wyjściu kanału podawano na wejście korektora z decyzyjnym sprzężeniem zwrotnym i korektorem wstępnym oraz na wejście korektora z decyzyjnym sprzężeniem zwrotnym i z badaną regułą decyzyjną. Dla różnych poziomów zakłóceń porównywano dane odbierane $d_k^{(i)}$ z danymi nadanymi a_k $(i = 1, 2)$. Dla każdego korektora wyznaczono liczbę błędnie odebranych bitów i odniesiono ją do liczby wszystkich odebranych bitów.

Eksperyment powtarzano zmieniając odstęp sygnału od szumu (SNR). We wszystkich eksperymentach generowano ciągi 65536 bitów, zatem każdy jeden przekłamany bit zwiększał obliczoną stopę błędów o $1,526 * 10^{-5}$. Ponieważ rozważane wartości elementowej stopy błędów mieściły się w przedziale od 10^{-1} do kilka razy 10^{-4}, przyjętą liczbę transmitowanych bitów można uznać za wystarczającą.

3. SYMULOWANE KANAŁY

Modelowany za pomocą filtru transwersalnego kanał wygodnie jest opisywać ciągiem próbek odpowiedzi na jednostkowy impuls wejściowy. W eksperymentach rozważano kanały opisywane ciągiem trzech oraz pięciu próbek: y_0, y_1, y_2 lub y_0, y_1, y_2, y_3, y_4. Założono, że kanały są dopasowane do sygnału, czyli założono, że próbka środkowa (y_1 dla kanału o trzech próbkach lub y_2 dla kanału o pięciu próbkach), zwana dalej próbką główną, jest dodatnia i największa co do wartości bezwzględnej, a pozostałe próbki, nazywane próbkami interferencyjnymi, są symetryczne względem próbki głównej.

Przyjęto też, że energia impulsu odpowiedzi kanału jest równa energii impulsu wejściowego. Oznacza to, że wartości próbek są unormowane tak, aby spełnić równanie

$$\sum_{i=0}^{g} y_i^2 = 1. \tag{1}$$

Do wyznaczania wartości próbek y_i dla założonych kanałów, opracowano dwa algorytmy.

Algorytm 1

Algorytm ten wyznacza wartości próbek y_0, y_1, y_2 dla kanału opisywanego ciągiem trzech próbek. Symetria próbek interferencyjnych oznacza, że $y_0 = y_2$, co pozwala zapisać (1) w postaci

$$2y_0^2 + y_1^2 = 1. \tag{2}$$

To zaś pozwala na określenie wartości próbki głównej

$$y_1 = \sqrt{1 - 2y_0^2}\,. \tag{3}$$

Próbka główna y_1 (o największej wartości bezwzględnej) spełnia nierówność $y_0^2 \leqslant y_1^2$, która uwzględniona w (2) daje przedział zmienności dla y_0

$$-\tfrac{1}{\sqrt{3}} \leqslant y_0 \leqslant \tfrac{1}{\sqrt{3}}. \tag{4}$$

W zakresie tego przedziału wartości y_0 zmieniano w trakcie eksperymentu ze stałym krokiem, natomiast wartości próbki głównej y_1 obliczano z (3).

Algorytm 2

W przypadku kanału opisanego przez pięć próbek odpowiedzi na jednostkowy impuls wejściowy, próbką główną jest y_2, natomiast próbki interferencyjne spełniają warunki symetrii: $y_0 = y_4$ oraz $y_1 = y_3$.

Oznaczmy przez α oraz β stosunki

$$\alpha = \frac{y_0}{y_2} \text{ oraz } \beta = \frac{y_1}{y_2}. \tag{5}$$

Uwzględniając (5) w (1) otrzymuje się

$$(2\alpha^2 + 2\beta^2 + 1)y_2^2 = 1. \tag{6}$$

Ostatecznie wartość próbki głównej obliczona z (6) wynosi

$$y_2 = \sqrt{\frac{1}{2\alpha^2 + 2\beta^2 + 1}}\,. \tag{7}$$

Kanały generowano zmieniając niezależnie wartości zmiennych α oraz β w przedziale od -1 do 1 z ustalonym krokiem, a następnie obliczając wartość próbki głównej z (7) i wartości próbek interferencyjnych z (5). Dla kanałów wygenerowanych za pomocą *Algorytmu 1* oraz *Algorytmu 2* zaprojektowano typowe korektory z decyzyjnymi sprzężeniami zwrotnymi oraz korektorami wstępnymi. Parametry korektorów dobierano według kryterium minimalizacji błędu średniokwadratowego.

4. WYNIKI I WNIOSKI

Dla kanałów opisanych trzema próbkami na rys. 2, 4, 6, 8 i 10 przedstawiono wykresy obliczonej elementowej stopy błędów Pe (stosunek liczby błędnie odebranych bitów do liczby wszystkich odebranych bitów) dla każdego korektora w funkcji y_0/y_1 oraz przy ustalonym (wyrażonym w decybelach) stosunku mocy sygnału do mocy szumu (SNR).

Na wszystkich wykresach poszczególne krzywe ilustrują wyniki dla systemów z następującymi korektorami:

Linia 1 — korektor z decyzyjnym sprzężeniem zwrotnym i z liniowym korektorem wstępnym,

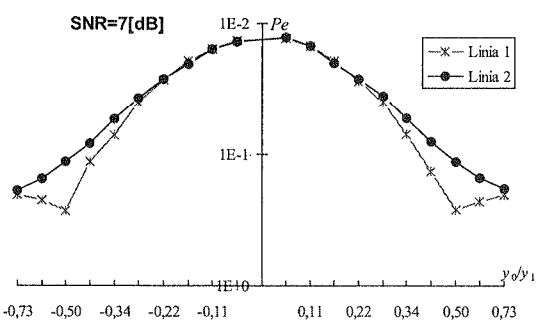
Linia 2 — korektor z decyzyjnym sprzężeniem zwrotnym i z badanym algorytmem decyzyjnym.

Na rysunkach od rys. 2 do rys. 11 dodatnie wartości stosunku y_0/y_1 oznaczają, że zarówno $y_0 > 0$ oraz $y_2 > 0$, natomiast ujemne wartości y_0/y_1 oznaczają ujemne wartości y_0 oraz y_2, ponieważ przyjęto, że główny prążek odpowiedzi kanału (w tym przypadku y_1) jest dodatni.

Na rys. numer 3, 5, 7, 9 i 11 zilustrowano te same wyniki pokazując stosunek elementowej stopy błędów $Pe(1)$ w systemie z decyzyjnym sprzężeniem zwrotnym i z liniowym korektorem wstępnym (Linia 1) do elementowej stopy błędów $Pe(2)$ w systemie z korektorem z decyzyjnym sprzężeniem zwrotnym i z badanym algorytmem decyzyjnym (Linia 2), jako K_{en}. Tak więc

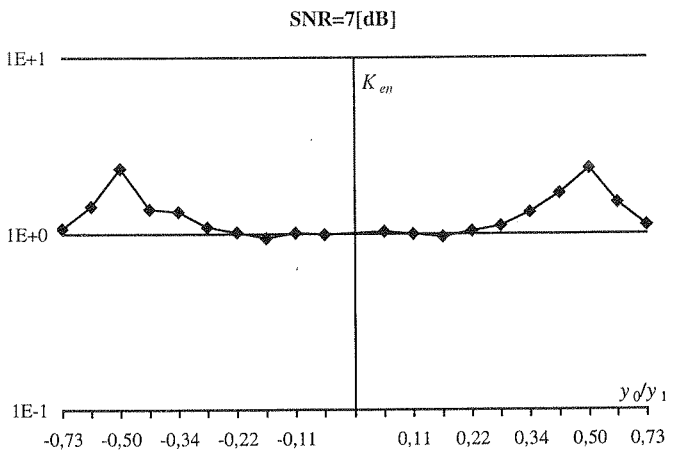
$$K_{en} = \frac{Pe(1)}{Pe(2)}. \tag{8}$$

Stosunek ten przedstawiono w funkcji stosunku y_0/y_1. Wartości $K_{en} > 1$ oznaczają, że system z badanym algorytmem przewidywania decyzji pracuje lepiej niż system z liniowym korektorem wstępnym.



Rys. 2. Zależność stopy błędów od stosunku y_0/y_1 dla SNR = 7 dB

Fig. 2. The bit error rate dependence on y_0/y_1 ratio for SNR = 7 dB



Rys. 3. Zależność K_{en} od y_0/y_1 dla SNR = 7 dB

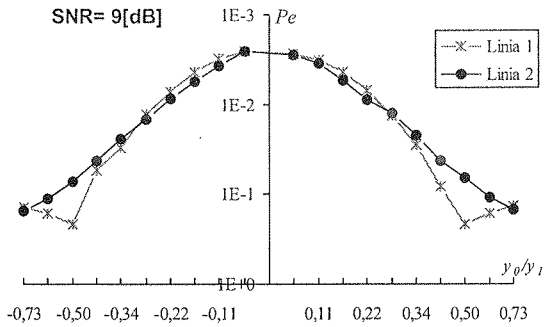
Fig. 3. The dependence of K_{en} on y_0/y_1 for SNR = 7 dB

Na rys. 2 i 3 prezentowane są uzyskane stopy błędów dla ustalonego poziomu sygnału nad szumem SNR = 7 dB (duże zakłócenia). Z przebiegu wykresów wynika, że system z przewidywaniem decyzji daje lepsze wyniki, bądź leżące na granicy kwalifikowania systemu jako lepszy lub gorszy. Gdy stosunek $y_0/y_1 \in (-0.72, -0.27)$ i $y_0/y_1 \in (0.27, 0.72)$ system z przewidywaniem decyzji wykazuje o ok. 10%, a nawet o 25% mniej błędów niż system z korektorem wstępnym. Poza określonym przedziałem system z przewidywaniem decyzji jest minimalnie lepszy lub otrzymane wyniki, są prawie identyczne jak dla korektorów tradycyjnych (wykresy przebiegają bardzo blisko siebie, lub się nakładają).

Dla mniejszych zakłóceń SNR = 9 dB (rys. 4 i 5), można zauważyć wzrost elementowej stopy błędów dla przypadków, gdy próbki y_i (i = 0, 2) przyjmują wartości bliskie ekstremalnym, oraz gdy stosunek y_0/y_1 jest mniejszy od 0.30. Obszar w którym system z przewidywaniem decyzji jest lepszy (wykazuje spadek liczby błędów nawet o ok. 30%) w porównaniu z korektorami tradycyjnymi, różni się nieznacznie z obszarem określonym dla przypadku gdy SNR = 7 dB.

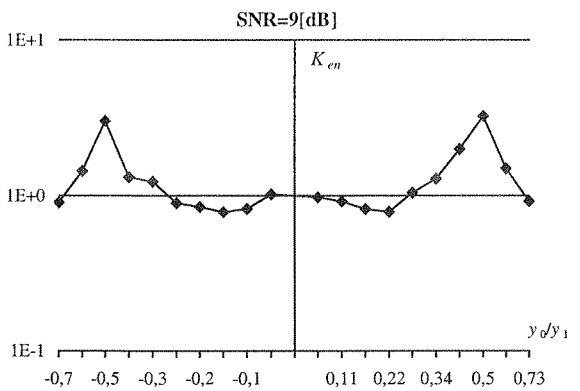
Wykresy na rys. 6 i 7 reprezentują wyniki dla SNR = 11 dB. Gdy wartości próbek odpowiedzi kanału są dodatnie i stosunek y_0/y_1 zawiera się w przedziale od 0.34 do 0.60, stosując system z przewidywaniem decyzji uzyskujemy mniej błędów o ok. 10% a nawet o 45% niż w przypadku systemu z korektorem wstępnym. Gdy $y_0 = y_2$ są ujemne, przedział wartości stosunku y_0/y_1, dla których korektor z przewidywaniem decyzji daje mniejszą stopę błędów, zawęża się do zakresu od −0.60 do −0.40.

W przypadku gdy SNR = 13 dB (rys. 8 i 9), najlepsze efekty z zastosowania systemu z przewidywaniem decyzji uzyskujemy dla stosunku y_0/y_1 mieszczącego się w przedziałach od 0.59 do 0.38 oraz od −0.45 do −0.59. Spadek liczby błędów jest rzędu 10% do 80% w porównaniu z korektorem tradycyjnym. Dla stosunku $y_0/y_1 \in (-0.16, 0.16)$



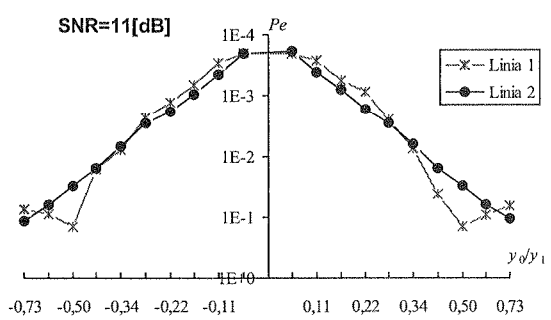
Rys. 4. Zależność stopy błędów od stosunku y_0/y_1 dla SNR = 9 dB

Fig. 4. The bit error rate dependence on y_0/y_1 ratio for SNR = 9 dB



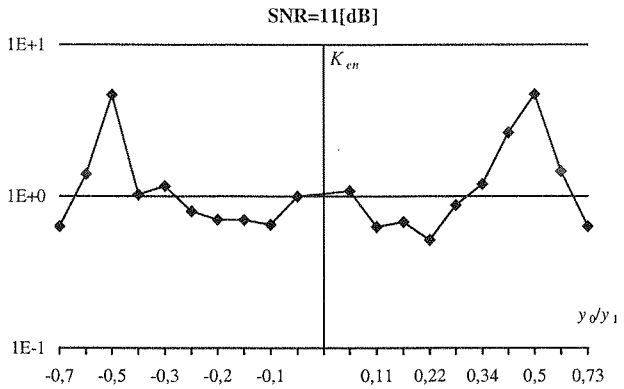
Rys. 5. Zależność K_{en} od y_0/y_1 dla SNR= 9 dB

Fig. 5. The dependence of K_{en} on y_0/y_1 for SNR = 9 dB



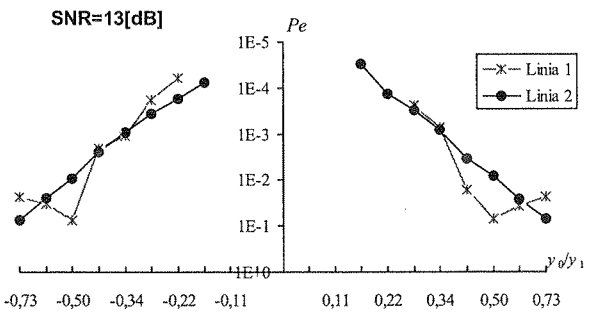
Rys. 6. Zależność stopy błędów od stosunku y_0/y_1 dla SNR = 11 dB

Fig. 6. The bit error rate dependence on y_0/y_1 ratio for SNR = 11 dB



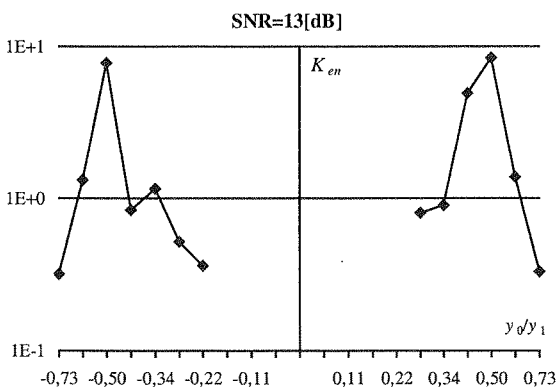
Rys. 7. Zależność K_{en} od y_0/y_1 dla SNR = 11 dB

Fig. 7. The dependence of K_{en} on y_0/y_1 for SNR = 11 dB



Rys. 8. Zależność stopy błędów od stosunku y_0/y_1 dla SNR = 13 dB

Fig. 8. The bit error rate dependence on y_0/y_1 ratio for SNR = 13 dB

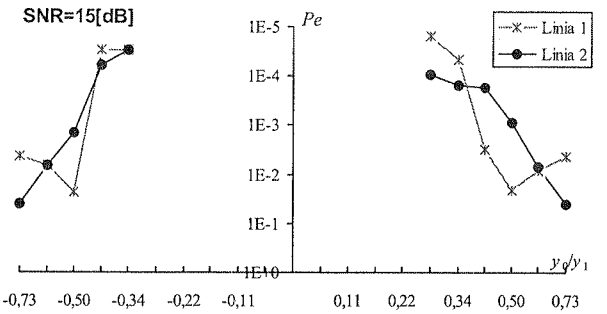


Rys. 9. Zależność K_{en} od y_0/y_1 dla SNR= 13 dB

Fig. 9. The dependence of K_{en} on y_0/y_1 for SNR= 13dB

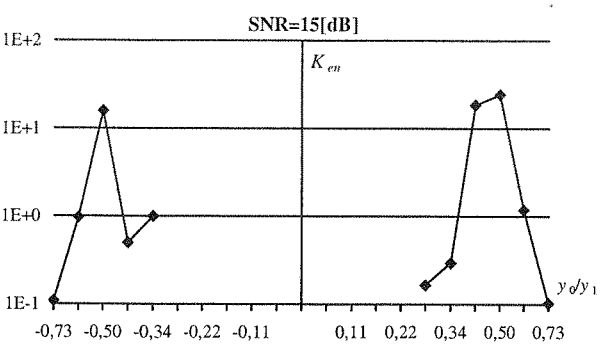
określenie obszaru zastosowania jest niemożliwe, gdyż oba systemy pracowały bezbłędnie.

Na rys. 10 i 11 reprezentowane są wyniki dla SNR = 15 dB. Obszar zastosowania badanej metody pokrywa się z obszarem jak dla przypadku gdy SNR = 13 dB.



Rys. 10. Zależność stopy błędów od stosunku y_0/y_1 dla SNR= 15 dB

Fig. 10. The bit error rate dependence on y_0/y_1 ratio for SNR = 15 dB

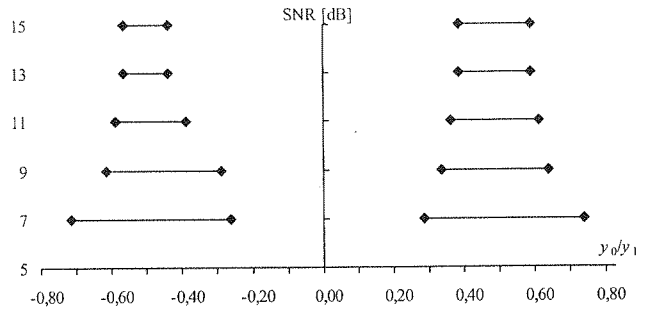


Rys. 11. Zależność K_{en} od y_0/y_1 dla SNR= 15 dB

Fig. 11. The dependence of K_{en} on y_0/y_1 for SNR = 15dB

Natomiast efektywność systemu z przewidywaniem decyzji spada o ok. 70%. Przedział, w którym nie było możliwe określenie obszaru zastosowania, poszerzył się do $y_0/y_1 \in (-0.34, 0.27)$.

Na rys. 12 dla każdej z prezentowanych wyżej zależności elementowej stopy błędów od stosunku y_0/y_1, zostały wykreślone zalecane przedziały zastosowania systemu z przewidywaniem decyzji. Dla ustalonego SNR zaznaczono odcinki określające przedziały wartości y_0/y_1, dla których w systemach z przewidywaniem decyzji otrzymano mniej przekłamań niż w systemach z korektorem wstępnym. Łatwo zauważyć, że im bardziej $|y_0/y_1|$ jest bliższe wartości 0.5, tym lepiej zachowuje się system z przewi-



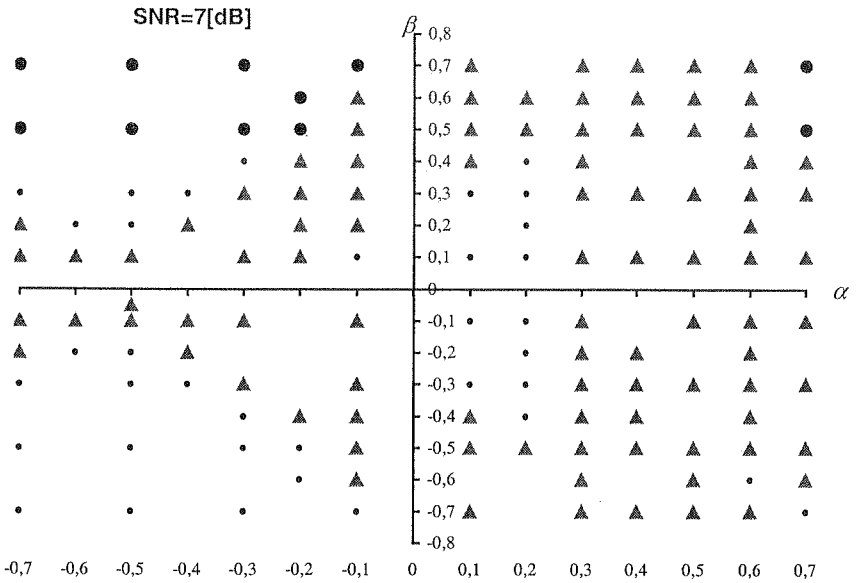
Rys. 12. Przedziały y_0/y_1 dla różnych wartości SNR

Fig. 12. The value intervals of y_0/y_1 for different SNR values

dywaniem decyzji. Im większe są zakłócenia, tym szerszy jest obszar zastosowania opisanego systemu. Na przykład kanał $y_0 = y_2 = \pm\frac{1}{\sqrt{6}}$, $y_1 = \frac{2}{\sqrt{6}}$, czyli kanał o dyskretnej transmitancji $Y(z) = \frac{\pm 1}{\sqrt{6}}\left(1 \pm z^{-1}\right)^2$ będzie korygowany lepiej korektorem z przewidywaniem decyzji nawet przy małych zakłóceniach.

Wykresy na rys. od 13 do 17 ilustrują dla kanałów opisanych pięcioma próbkami wyniki porównania elementowej stopy błędów uzyskanej w obu systemach w zależności od α oraz β przy ustalonej wartości SNR. Na układ współrzędnych $\alpha\beta$ naniesiono punkty w kształcie trójkątów lub kółek. Trójkąt umieszczony w punkcie (α, β) oznacza, że dla kanału o tych parametrach system drugi z badaną regułą decyzyjną jest lepszy niż system pierwszy z liniowym korektorem wstępnym.

Na rys. 13 prezentowane są rezultaty porównań stopy błędów uzyskane w systemach z rozważanymi korektorami dla ustalonego poziomu sygnału nad szumem SNR = 7 dB. W pierwszej ćwiartce prezentowane są wyniki dla dodatnich wartości α i β. System z przewidywaniem decyzji daje gorsze wyniki (punkty okrągłe) niż system z korektorem wstępnym dla większych α (np. $\alpha = 0.7$) i gdy β przyjmuje wartości większe od 0.5 oraz dla mniejszych α ($\alpha \leqslant 0.2$) i gdy $\beta \in (0.1, 0.3)$. W pozostałym obszarze system z przewidywaniem decyzji jest lepszy (punkty trójkątne). Analizując IV ćwiartkę zauważamy, że obszar zastosowania korektora z przewidywaniem decyzji jest symetryczny względem osi α. Gdy $\alpha < 0$, czyli dla II i III ćwiartki stosując system z przewidywaniem decyzji otrzymujemy mniejszą stopę błędów w porównaniu z systemem z korektorem wstępnym gdy $\beta \leqslant 0.2$ i $\alpha \in (-0.1, -0.7)$ albo gdy $\alpha \geqslant -0.2$ i $\beta < 0.7$. Wnioskiem wynikającym z symetrii obszarów na rys. 13 jest to, że skuteczność korekcji opartej na przewidywaniu decyzji w odniesieniu do korekcji tradycyjnej w istotny sposób zależy od znaku α a więc od zgodności polaryzacji prążków interferencyjnych y_0 i y_4 z prążkiem głównym y_2, natomiast skuteczność ta nie zależy w sposób znaczący od znaku β, czyli nie zależy od polaryzacji prążków y_1 i $y3$.



Rys. 13. Akceptowane obszary (α,β) dla SNR = 7 dB

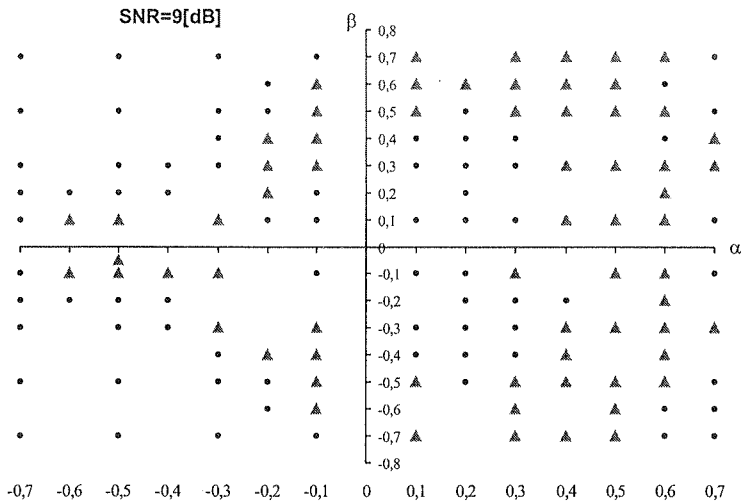
Fig. 13. The acceptable area (α,β) for SNR = 7 dB

Podobnie jak dla SNR = 7 dB (rys. 13), tak i dla SNR = 9 dB (rys. 14) można zauważyć symetrię obszarów względem osi α.

W pierwszej ćwiartce ($\alpha > 0$ i $\beta > 0$) korektora z przewidywaniem decyzji okazuje się lepszy dla większych α lub większych β (gdy $\alpha \geqslant 0.4$ lub $\beta \geqslant 0.5$) z wyłączeniem sytuacji gdy oba te parametry są równocześnie większe od 0.5 (np. $\alpha \geqslant 0.6$ i $\beta \geqslant 0.6$). Gdy $\alpha < 0$, to korektor z przewidywaniem decyzji jest lepszy albo gdy β jest bliskie zeru ($\beta \leqslant 0.1$) i $\alpha < -0.2$, albo gdy α jest bliskie zeru ($\alpha \geqslant -0.2$) i $\beta > 0.2$.

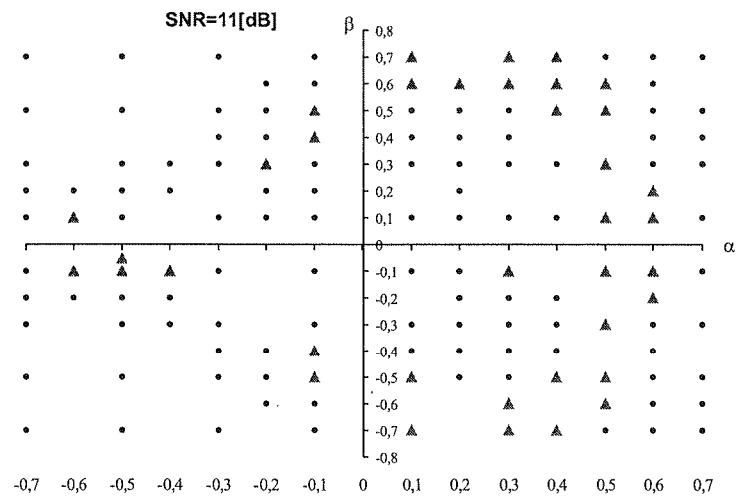
Rezultaty prezentowane na rys. 15 określają obszar zastosowania korektora z przewidywaniem decyzji dla SNR = 11 dB. Również w tym przypadku występuje symetria obszarów względem osi α. W pierwszej ćwiartce korektor z przewidywaniem decyzji jest lepszy gdy $\alpha \leqslant 0.5$ a $\beta \geqslant 0.5$ oraz dla średnich wartości α ($\alpha \approx 0.6$) i małych β ($\beta \leqslant 0.2$). System z przewidywaniem decyzji dla SNR = 11 dB warto stosować gdy α i β nie przyjmują jednocześnie małych albo dużych wartości. Dla $\alpha < 0$ korektor z przewidywaniem decyzji praktycznie nie powinien być stosowany, za wyjątkiem nielicznych punktów, gdzie α jest małe ($\alpha < 0.2$), a β średnie ($\beta \approx 0.4$) lub gdy β jest małe ($\beta \leqslant 0.1$), a α średnie ($\alpha \approx 0.5$).

Na rys. 16 i 17 przedstawiono wyniki kolejno dla SNR = 13 dB oraz SNR = 15 dB. Jak widać obszar zastosowania korektorów z przewidywaniem decyzji kurczy się wraz ze wzrostem SNR. Dla ujemnych wartości α, czyli dla ujemnych y_0 oraz y_4, zaleca się stosować tradycyjne systemy z korektorami wstępnymi. Dla dodatnich α obszar



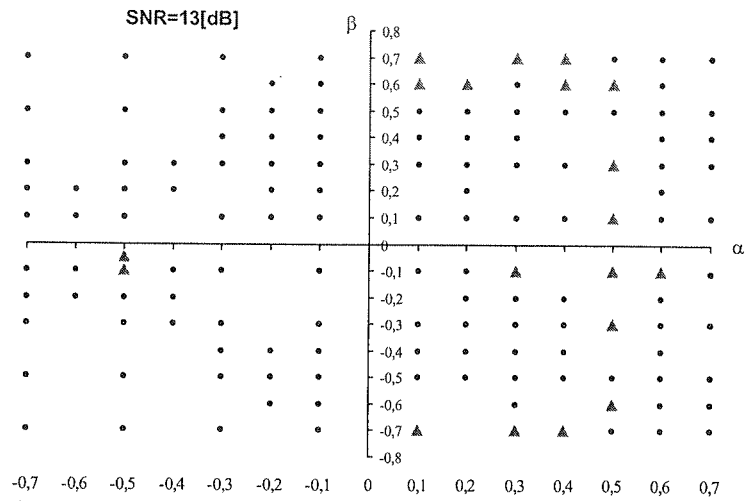
Rys. 14. Akceptowane obszary (α,β) dla SNR = 9dB

Fig. 14. The acceptable area (α,β) for SNR = 9dB



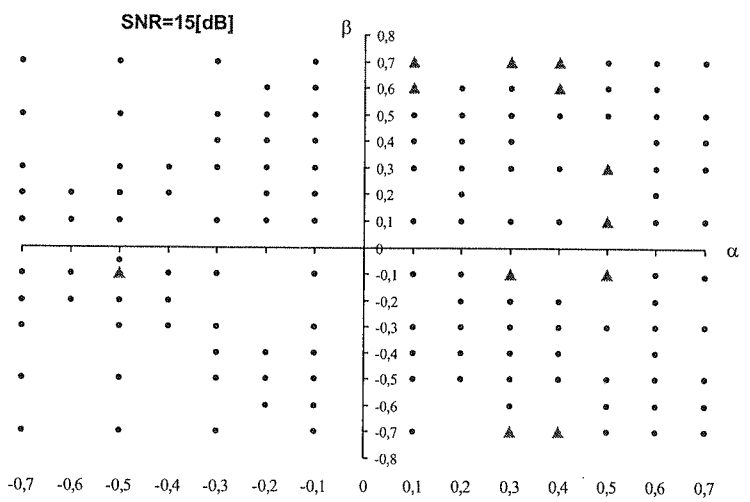
Rys. 15. Akceptowane obszary (α,β) dla SNR = 11 dB

Fig. 15. The acceptable area (α,β) for SNR = 11 dB



Rys. 16. Akceptowane obszary (α,β) dla SNR = 13 dB

Fig. 16. The acceptable area (α,β) for SNR = 13 dB



Rys. 17. Akceptowane obszary (α,β) dla SNR = 15 dB

Fig. 17. The acceptable area (α,β) for SNR = 15 dB

zastosowania korektorów z przewidywaniem decyzji skupia się w otoczeniu wartości $\alpha = 0.5$ i $\beta \leqslant 0.3$.

5. PODSUMOWANIE

Przedstawione wyniki świadczą, że zarówno dla kanałów charakteryzujących się trzema jak i pięcioma próbkami odpowiedzi impulsowej, im mniejszy jest stosunek mocy sygnału do mocy szumu (większe zakłócenia), tym większą poprawę elementowej stopy błędów uzyskuje się stosując algorytm z przewidywaniem decyzji i rezygnując z korektora wstępnego. Gdy stosunek SNR rośnie (zakłócenia maleją), obszar zastosowania systemu z przewidywaniem decyzji maleje.

Analizując rezultaty obliczeń dla kanałów o trzech próbkach odpowiedzi impulsowej na wykresach od rys. 2 do rys. 12, można zauważyć, że przy ustalonym poziomie stosunku SNR największe zastosowanie system z przewidywaniem decyzji ma dla wartości stosunku y_0/y_1 równych co do modułu 1/2.

Z zależności charakteryzujące kanały o pięciu próbkach odpowiedzi impulsowej wynika, że najlepsze rezultaty uzyskuje się w I i IV ćwiartce układu $\alpha\beta$, gdy wartości y_0 oraz y_1 są dodatnie. Obszary zastosowania wykazują symetrię względem osi α (nieznaczne różnice). Zdecydowanie mniejszy obszar zastosowania korektorów z przewidywaniem decyzji uzyskujemy dla ćwiartki II i III gdzie y_0 oraz y_1 przyjmują wartości ujemne. Gdy poziom sygnału nad szumem SNR $\geqslant$ 13 dB, dla systemu z przewidywaniem decyzji w prawie całym obszarze badania uzyskujemy gorszą stopę błędów niż dla korektorów tradycyjnych.

W przypadku kanałów o pięciu próbkach odpowiedzi impulsowej, z symetrii obszarów względem osi α można wnioskować, że niezależnie od stosunku mocy sygnału do mocy szumu, wyniki porównania systemu stosującego przewidywanie decyzji z systemem tradycyjnym zależą od polaryzacji prążków interferencyjnych y_0 i y_4 bardziej odległych od prążka głównego y_2, natomiast w niewielkim stopniu zależą od polaryzacji prążków y_1 i y_3 leżących bezpośrednio przed i za prążkiem głównym y_2. Na zachowanie się systemu z przewidywaniem decyzji mają wpływ również wartości prążków y_0 i y_4 oraz y_1 i y_3, przy czym wpływ ten zależy od poziomu szumu. Przy mniejszych szumach im większa jest różnica wartości tych prążków, tym system z przewidywaniem decyzji jest lepszy w porównaniu z systemem tradycyjnym.

6. BIBLIOGRAFIA

1. S.A. Altekar, A.E. Vityaev, J.K. Wolf: *Decision-Feedback Equalization via Separating Hyperplanes*. IEEE Trans. on Communications, March 2001, vol. 49, no 3, pp. 480-486.
2. N.C. Beaulieu: *Bounds on variances of Recovery Times of Decision Feedback Equalizers*. IEEE Trans. on Inform. Theory, Sept. 2000, vol. 46, no 3, pp. 2249- 2256.
3. W.R. Bennett, J.R. Davey: *Data Transmission*. New York, McGraw- Hill, 1965.
4. A.B. Carlson: *Communication Systems*. Tokyo, McGraw-Hill, 1975.

5. S. Chen, L. Hanzo, B. Mulgrew: *Decison-Feedback Equalization Using Multiple-Hyperplane Partitioning for Detecting ISI-Corrupted M-arry PAM Signals*. IEEE Trans. on Communications, May 2001, vol. 49, no 5, pp. 760-764.
6. A.P. Clark: *Advanced Data-Transmission Systems*. London, Pentech Press, 1977.
7. A.P. Clark: *Principles of Digital Data Transmission*. London, Pentech Press, 1976.
8. A. Dąbrowski: *Odbiór sygnałów transmisji danych w obecności zniekształceń interferencyjnych i zakłóceń*. Rozdz.4, Pr. zb. pod kier. Z. Barana: Podstawy transmisji danych, Warszawa, WKŁ, 1982.
9. K.O. Holdsworth, D.P. Taylor, R.T. Pullman: *On Combined Equalization and Decoding for Multilevel Coded Modulation*. IEEE Trans. on Communications, June 2001, vol. 49, no 6, pp. 943-947.
10. J. Kisilewicz: *Problemy poprawy wierności transmisji danych w sieciach komputerowych*. Prace Naukowe Instytutu Sterowania i Techniki Systemów Politechniki Wrocławskiej, Seria: Monografie nr 6, Wrocław, WPWr, 1991.
11. J. Kisilewicz: *Nieliniowa korekcja interferencji z przewidywaniem przyszłej decyzji*. Kwartalnik Elektroniki i Telekomunikacji, 1994, z. 4, ss. 489-497.
12. J. Kisilewicz: *Korekcja interferencji metodą przewidywania nadchodzących danych*. Kwartalnik Elektroniki i Telekomunikacji, 1997, z. 3, ss. 407-420.
13. J.Labat, C. Laot: *Blind Adaptive Multiple-Input Decision-Feedback Equalizer with a Self-Optimized Configuration*. IEEE Trans. on Communications, April 2001, vol. 49, no 4, pp. 646-654.
14. R.W. Lucky, J. Salz, E.J. Weldon: *Principles of Data Communication, New York*. McGraw-Hill, 1968.
15. W. Sobczak (red.): *Problemy teleinformatyki*. Warszawa, WKŁ, 1984.
16. M. Sternad, A. Ahlen: *The Structure and Design of Realizable Decision Feedback Equalizers for IIR Channels with Colored Noise*, IEEE, July 1990, vol. IT-36, no. 4, pp. 848-858.
17. J. Tsimbions, L.B. White: *Error Propagation and Recovery in Decision- Feedback Equalizers for Nonlinear Channels*. IEEE Trans. on Communications, Febr. 2001, vol. 49, no 2, pp. 239-242.
18. T. Willink, P.H. Wittke, L.L. Campbell: *Evaluation of the Effect of Intersymbol Interference in Decision-Feedback Equalizers*. IEEE Trans. on Communications, April 2000, vol. 48, no 4, pp. 629-636.

J. KISILEWICZ, A. GRZYBOWSKI

APPLICATION AREA OF FURTHER DECISIONS PREDICTION METHOD FOR THE INTERSYMBOL INTERFERENCE EQUALIZATION

Summary

The most common equalizers are the decision feedback equalizers with linear feedforward transversal filter at its input. This linear filter gives additional time delay to the transmission path and is ineffective for some kind of channels. So the new method of the intersymbol interference equalization was proposed. This method uses the decision feedback equalizer without linear feedforward transversal filter at its input. The linear filter was swapped by decisions prediction algorithm. The decision is worked out to maximize the probability of received data. This algorithm gives better results only for some channels. The application area of this new equalization method is described in this paper. The data transmission systems with new equalizer and common equalizer with linear transversal filter where simulated. Each of he simulated transmission paths contain: the binary data source, the channel, the white Gaussian noise source and two equalizers with data detectors. The bit error rate on these two detectors was compared for different noise

power and for two classes of channels. The channels were simulated by linear feedforward transversal filter that gives two or four symmetric interference samples with respect to the main sample. The channels giving less bit error rate for system with described new equalizer are found. The pulse response of the first class channel contains three samples y_0, y_1 and y_2. The energy of the response is normalized to one and symmetric with respect to y_1, so $y_0^2 + y_1^2 + y_2^2 = 1$ and $y_0 = y_2$. Assuming the absolute value of the main sample y_1 have to be maximum, the value of the y_0 and y_2 was changed from –0.577 to 0.577. The pulse response of the second class channel contains five samples and two of them $y_0 = y_4$ and $y_1 = y_3$ were changed independently with similar assumptions. For each channel the signal to noise ratio (SNR) was changed from 7 to 15 dB. Depends on the channel and SNR, the equalizer with the further decisions prediction method gives the less or greater bit error rate then the common used equalizer does. The values of y_0/y_1 for first class channels and values y_0/y_2 and y_0/y_2 for second class channels giving less bit error rate are presented on the figures.

Keywords: data transmission, transversal equalizer, decision feedback.

KWARTALNIK ELEKTRONIKI I TELEKOMUNIKACJI, 2004, 50, z. 1, ss. 49–63

Kompresja falkowa tła obrazów z obszarami zainteresowania

WALDEMAR RAKOWSKI

Politechnika Białostocka, Wydział Informatyki
ul. Wiejska 45A, 15-351 Białystok
e-mail: waldi@ii.pb.bialystok.pl

Otrzymano 2003.04.24
Autoryzowano 2003.08.30

Obrazy, w kórych można wyróżnić fragmenty wymagające kodowania bez zmiany nawet jednego piksela (obszary zainteresowania) podczas gdy w pozostałych częściach obrazu dopuszcza się kodowanie ze stratą jakości, mogą być kompresowane za pomocą dwóch metod: metody niewnoszącej strat i metody wnoszącej straty. Pierwsza metoda służy do zakodowania obszarów zainteresowania, druga do zakodowania pozostałych części obrazu (tła). Efektywną metodą kodowania tła jest metoda falkowa z nieodwracalną transformacją falkową. Nieodwracalna transformacja falkowa pozwala uzyskać lepszą kompresję obrazu ze stratą jakości niż odwracalna transformacja falkowa. Dodatkowe podwyższenie jakości kompresji tła można uzyskać poprzez modyfikację piramidy falkowej obrazu, polegającą na wyzerowaniu tych współczynników piramidy falkowej, które nie biorą udziału w rekonstrukcji tła. W wyniku takiej modyfikacji piramidy falkowej powstają dodatkowe drzewa zer, które w metodach kodowania osadzającego, są kodowane za pomocą pojedyńczych bitów. Modyfikacja piramidy falkowej może polepszyć jakość kompresji tła w sensie kryterium zniekształceń PSNR nawet o parę decybeli, w zależności od wielkości obszarów zainteresowania. W pracy przedstawiono algorytm wyznaczania położenia modyfikowanych współczynników falkowych oraz wyniki eksperymentów na paru obrazach, które ilustrują efektywność zaproponowanej metody hybrydowej kompresji obrazów z obszarami zainteresowania kodowanymi bez strat.

Słowa kluczowe: transformata falkowa, kompresja falkowa obrazów, obszary zainteresowania, kodowanie osadzające

1. WPROWADZENIE

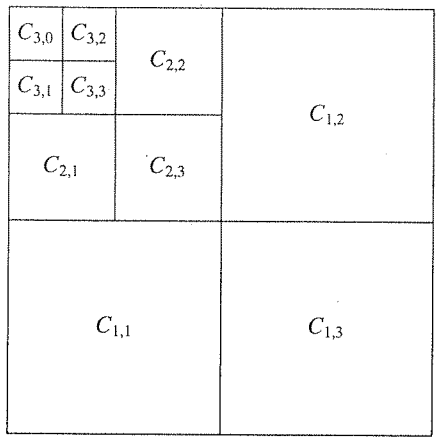
Obraz cyfrowy może być kompresowany metodą niewprowadzającą zniekształceń lub metodą wprowadzającą zniekształcenia. Metody kompresji niewprowadzają-

ce zniekształceń umożliwiają osiąganie współczynników kompresji nie przekraczających najczęściej wartości 2; metody wprowadzające zniekształcenia pozwalają osiągać współczynniki rzędu kilkudziesięciu i wyższe, przy akceptowalnym z punktu widzenia użytkownika obrazu poziomie wprowadzonych zniekształceń. Drogą do osiągnięcia kompromisu pomiędzy jakością kompresji i współczynnikiem kompresji może być określenie w obrazie fragmentów, które muszą być skompresowane bez jakiejkolwiek zmiany nawet jednego piksela, podczas gdy w pozostałej części obrazu dopuszcza się zniekształcenia. Fragmenty obrazu wymagające kodowania bez utraty jakości nazywane są *obszarami zainteresowania* (Region of Interest, ROI), i pozostała część obrazu nazywana jest *tłem* (Background). Obszary zainteresowania mogą mieć różne kształty; reprezentacje kształtów obszarów zainteresowania omówiono w [12]. Najprostsze do zakodowania są obszary w formie prostokątów. Do ich kompresji można wykorzystać takie metody jak JPEG-LS [1, 2] albo JPEG 2000 z odwracalną transformacją falkową [3, 6, 8]. Do kompresji tła można zastosować metodę falkową z nieodwracalną transformacją falkową i kodowaniem osadzającym piramidy falkowej. W pracy przedstawiono koncepcję i wyniki eksperymentów kompresji falkowej tła obrazu z adaptacyjną kwantyzacją piramidy falkowej, pozwalającą polepszyć jakość kompresji tła poprzez modyfikację współczynników falkowych w miejscach odpowiadających obszarom zainteresowania. Układ pracy jest następujący. W punkcie 2 przedstawiono ideę kompresji falkowej obrazów z obszarami zainteresowania. W punkcie 3 podano szczegółowe informacje dotyczące położenia obszarów zainteresowania w piramidzie falkowej obrazu. Punkt 4 przedstawia nową metodę kompresji falkowej tła obrazów z obszarami zainteresowania z adaptacyjnym kwantowaniem piramidy falkowej. Wyniki eksperymentów przedstawiono w punkcie 5, i w punkcie 6 wnioski końcowe.

2. METODA FALKOWA KOMPRESJI OBRAZÓW

Kompresja falkowa obrazów [9, 12] składa się z dwóch etapów. Pierwszy etat polega na obliczeniu piramidy falkowej obrazu poprzez wykonanie kilku transformacji falkowych obrazu. Struktura przykładowej piramidy falkowej obrazu pokazana jest na rys. 1.

Drugim etapem kompresji jest kwantowanie współczynników piramidy falkowej i zakodowanie entropijne skwantowanych współczynników. W praktyce te dwie ostatnie czynności są wykonywane jednocześnie w ramach tzw. kodowania osadzającego piramidy falkowej [15, 14, 10]. Tło obrazu, po wycięciu obszarów zainteresowania, stanowi obraz z „dziurami". Metoda falkowa nie dopuszcza dziur w obrazie i kompresji falkowej musi być poddany pełny obraz. Ponieważ obszary zainteresowania będą na końcu procesu rekonstrukcji obrazu odtworzone z oddzielnego kodu, więc na etapie kompresji tła współczynniki falkowe obszarów zainteresowania niebiorące udziału w rekonstrukcji tła mogą być zmodyfikowane. Kluczową sprawą jest wyznaczenie położenia w piramidzie falkowej współczynników biorących udział wyłącznie w rekonstrukcji obszarów zainteresowania. Położenie tych współczynników zależy nie tylko od położenia



Rys. 1. Przykład piramidy falkowej obrazu (3-poziomowa transformata falkowa obrazu). Indeks j w $C_{j,k}$ oznacza numer transformacji, indeks k numer podobrazu; obraz oryginalny jest oznaczony jako $C_{0,0}$

Fig. 1. 3-level image wavelet pyramid. $C_{0,0}$ — original image, $C_{j,k}$ — subimage

obszarów zainteresowania ale i od długości filtrów falkowych użytych do obliczenia transformaty falkowej.

3. OBSZARY ZAINTERESOWANIA W DZIEDZINIE TRANSFORMATY FALKOWEJ

Indeksy współczynników falkowych należących do obszarów zainteresowania w piramidzie falkowej obrazu można wyznaczyć, w związku z separowalnością dwuwymiarowej transformaty falkowej obrazu, analizując wzór określający współczynniki odwrotnej jednowymiarowej transformaty biortogonalnej, który dla filtrów o skończonej odpowiedzi impulsowej ma postać [7]

$$a_j[n] = \sum_{k=k_1^{\tilde{h}}}^{k_2^{\tilde{h}}} \tilde{h}[n-2k]a_{j+1}[k] + \sum_{k=k_1^{\tilde{g}}}^{k_2^{\tilde{g}}} \tilde{g}[n-2k]d_{j+1}[k] \tag{1}$$

gdzie $\tilde{h}$ oznacza filtr dolnoprzepustowy syntezy, i $\tilde{g}$ filtr górnoprzepustowy syntezy. Współczynniki $a_{j+1}[k]$ biorące udział w rekonstrukcji współczynnika $a_j[n]$ mają indeksy k, spełniające warunek

$$L_1^{\tilde{h}} \leqslant n - 2k \leqslant L_2^{\tilde{h}},$$

Tabela 1

Granice indeksów współczynników falkowych biorących udział w obliczaniu współczynnika odwrotnej transformaty falkowej w przypadku banku filtrów 5/3

Wavelet coefficients taken into account in pixel reconstruction for 5/3 filter banks

N	$k_1^{\tilde{h}}$	$k_2^{\tilde{h}}$	$k_1^{\tilde{g}}$	$k_2^{\tilde{g}}$
$2p$	P	p	$p-1$	p
$2p+1$	P	$p+1$	$p-1$	$p+1$

gdzie $L_1^{\tilde{h}}$ i $L_2^{\tilde{h}}$, odpowiednio najniższy i najwyższy indeks współczynników odpowiedzi impulsowej filtru dolnoprzepustowego syntezy $\tilde{h}$. Z powyższej nierówności wynika, że granice sumowania $k_1^{\tilde{h}}$ i $k_2^{\tilde{h}}$ we wzorze (1) można wyrazić wzorami

$$\begin{aligned} k_1^{\tilde{h}} &= \left\lfloor \left(n - L_2^{\tilde{h}} + 1\right)/2 \right\rfloor, \\ k_2^{\tilde{h}} &= \left\lfloor \left(n - L_1^{\tilde{h}}\right)/2 \right\rfloor. \end{aligned}$$

Analogiczne zależności dotyczą indeksów k współczynników $d_{j+1}[k]$ biorących udział w rekonstrukcji współczynnika $a_j[n]$, co oznacza, że granice sumowania $k_1^{\tilde{g}}$ i $k_2^{\tilde{g}}$ we wzorze (1) wyrażają wzory

$$\begin{aligned} k_1^{\tilde{g}} &= \left\lfloor \left(n - L_2^{\tilde{g}} + 1\right)/2 \right\rfloor, \\ k_2^{\tilde{g}} &= \left\lfloor \left(n - L_1^{\tilde{g}}\right)/2 \right\rfloor, \end{aligned}$$

gdzie gdzie $L_1^{\tilde{g}}$ i $L_2^{\tilde{g}}$ odpowiednio najniższy i najwyższy indeks współczynników odpowiedzi impulsowej filtru górnoprzepustowego syntezy $\tilde{g}$.

W przypadku filtrów biortogonalnych 5/3 gdzie $L_1^{\tilde{h}} = -1$, $L_2^{\tilde{h}} = -1$, $L_1^{\tilde{g}} = -1$, $L_2^{\tilde{g}} = 3$. Zakresy indeksów k współczynników falkowych $a_{j+1}[k]$ i $d_{j+1}[k]$ biorących udział w rekonstrukcji współczynnika $a_{j+1}[n]$ przedstawione są w tabeli 1.

W l-poziomowej jednowymiarowej piramidzie falkowej każdemu obszarowi zainteresowania, którym jest odcinek, odpowiada $l+1$ podobszarów. Indeksy brzegowych współczynników falkowych tworzących te podobszary można obliczyć iteracyjnie dla $j = 1, 2, \ldots, l$ na podstawie poniższych wzorów:

$$\begin{aligned} m_{j,1}^{\tilde{h}} &= \left\lfloor \left(m_{j-1,1}^{\tilde{h}} - L_2^{\tilde{h}} + 1\right)/2 \right\rfloor, \\ m_{j,2}^{\tilde{h}} &= \left\lfloor \left(m_{j-1,2}^{\tilde{h}} - L_1^{\tilde{h}}\right)/2 \right\rfloor, \\ m_{j,1}^{\tilde{g}} &= \left\lfloor \left(m_{j-1,1}^{\tilde{h}} - L_2^{\tilde{g}} + 1\right)/2 \right\rfloor, \\ m_{j,2}^{\tilde{g}} &= \left\lfloor \left(m_{j-1,2}^{\tilde{h}} - L_1^{\tilde{g}}\right) \right\rfloor, \end{aligned}$$

Tabela 2

Rodzaj filtru biorącego udział w wyznaczaniu położenia obszaru zainteresowania w podpasmie $C_{j,k}$ dwuwymiarowej piramidy falkowej: $\tilde{h}$ — filtr dolnoprzepustowy syntezy, $\tilde{g}$ — filtr górnoprzepustowy syntezy

Filters determining position of ROI in subimage $C_{j,k}$ of 2D wavelet pyramid: $\tilde{h}$ — lowpass filter of synthesis, $\tilde{g}$ — highpass filter of synthesis

k	Podpasmo	x	y
0	LL	$\tilde{h}$	$\tilde{h}$
1	LH	$\tilde{h}$	$\tilde{g}$
2	HL	$\tilde{g}$	$\tilde{h}$
3	HH	$\tilde{g}$	$\tilde{g}$

gdzie $m_{0,1}^{\tilde{g}}$ i $m_{0,2}^{\tilde{h}}$ są indeksami odpowiednio lewego i prawego skrajnego piksela obszaru zainteresowania.

Liczba podobszarów w l-poziomowej dwuwymiarowej piramidzie falkowej, odpowiadających jednemu obszarowi zainteresowania w dziedzinie obrazu wynosi $3l+1$. Do wyznaczenia położenia podobszaru zainteresowania w podpasmie $C_{j,k}$ (rys. 1) używa się filtru, którego rodzaj zależy od typu podpasma (LL, LH, HL, HH) oraz kierunku (poziomy, pionowy). W tabeli 2 podane są filtry biorące udział w wyznaczaniu podobszaru zainteresowania w określonym podpasmie.

Najprostszym obszarem zainteresowania w dziedzinie obrazu jest prostokąt, zdefiniowany następująco:

$$x_1 \leqslant x \leqslant x_2,$$
$$y_1 \leqslant y \leqslant y_2.$$

W przypadku prostokąta granice podobszarów zainteresowania w piramidzie falkowej obrazu można wyznaczyć rozpatrując tylko dwa przeciwległe rogi prostokąta. Dla każdego z dwóch pikseli stanowiących wymienione rogi należy wyznaczyć indeksy współczynników falkowych biorących udział w rekonstrukcji rozpatrywanego piksela, niezależnie w kierunku poziomym i pionowym. Poniżej pokazano indeksy (i, j) współczynników falkowych należących do podobszarów zainteresowania w poszczególnych podobrazach transformaty falkowej obrazu.

$C_{1,0}$:

$$\left\lfloor \left(y_1 - L_2^{\tilde{h}} + 1\right)/2 \right\rfloor \leqslant i \leqslant \left\lfloor \left(y_2 - L_1^{\tilde{h}}\right)/2 \right\rfloor$$
$$\left\lfloor \left(x_1 - L_2^{\tilde{h}} + 1\right)/2 \right\rfloor \leqslant j \leqslant \left\lfloor \left(x_2 - L_1^{\tilde{h}}\right)/2 \right\rfloor$$

$C_{1,1}$:

$$\lfloor (y_1 - L_2^{\tilde{g}} + 1)/2 \rfloor \leqslant i \leqslant \lfloor (y_2 - L_1^{\tilde{g}})/2 \rfloor$$
$$\lfloor (x_1 - L_2^{\tilde{g}} + 1)/2 \rfloor \leqslant j \leqslant \lfloor (x_2 - L_1^{\tilde{g}})/2 \rfloor$$

$C_{1,2}$:

$$\lfloor (y_1 - L_2^{\tilde{h}} + 1)/2 \rfloor \leqslant i \leqslant \lfloor (y_2 - L_1^{\tilde{h}})/2 \rfloor$$
$$\lfloor (x_1 - L_2^{\tilde{g}} + 1)/2 \rfloor \leqslant j \leqslant \lfloor (x_2 - L_1^{\tilde{g}})/2 \rfloor$$

$C_{1,3}$:

$$\lfloor (y_1 - L_2^{\tilde{g}} + 1)/2 \rfloor \leqslant i \leqslant \lfloor (y_2 - L_1^{\tilde{g}})/2 \rfloor$$
$$\lfloor (x_1 - L_2^{\tilde{g}} + 1)/2 \rfloor \leqslant j \leqslant \lfloor (x_2 - L_1^{\tilde{g}})/2 \rfloor$$

Położenie podobszarów zainteresowania w l-poziomowej piramidzie falkowej obrazu może być wyznaczone w sposób iteracyjny. Przed rozpoczęciem pętli należy wykonać następujące podstawienia:

$$\begin{aligned} x_{0,1} &= x_1 \\ x_{0,2} &= x_2 \\ y_{0,1} &= y_1 \\ y_{0,2} &= y_2, \end{aligned}$$

a następnie w pętli dla $m = 1, 2, \ldots l$ obliczyć indeksy dla poszczególnych podobrazów według poniższych wzorów:

$C_{m,1}$:

$$\lfloor (y_{m-1,1} - L_2^{\tilde{g}} + 1)/2 \rfloor \leqslant i \leqslant \lfloor (y_{m-1,2} - L_1^{\tilde{g}})/2 \rfloor$$
$$\lfloor (x_{m-1,1} - L_2^{\tilde{h}} + 1)/2 \rfloor \leqslant i \leqslant \lfloor (x_{m-1,2} - L_1^{\tilde{h}})/2 \rfloor$$

$C_{m,2}$:

$$\lfloor (y_{m-1,1} - L_2^{\tilde{h}} + 1)/2 \rfloor \leqslant i \leqslant \lfloor (y_{m-1,2} - L_1^{\tilde{h}})/2 \rfloor$$
$$\lfloor (x_{m-1,1} - L_2^{\tilde{g}} + 1)/2 \rfloor \leqslant i \leqslant \lfloor (x_{m-1,2} - L_1^{\tilde{g}})/2 \rfloor$$

$C_{m,3}$:

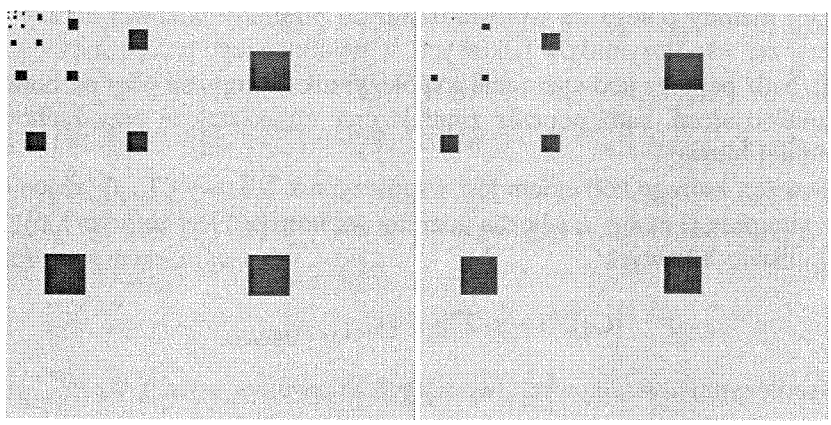
$$\lfloor (y_{m-1,1} - L_2^{\tilde{g}} + 1)/2 \rfloor \leqslant i \leqslant \lfloor (y_{m-1,2} - L_1^{\tilde{g}})/2 \rfloor$$
$$\lfloor (x_{m-1,1} - L_2^{\tilde{g}} + 1)/2 \rfloor \leqslant i \leqslant \lfloor (x_{m-1,2} - L_1^{\tilde{g}})/2 \rfloor$$

$$
\begin{aligned}
x_{m,1} &= \left\lfloor \left(x_{m-1,1} - L_2^{\tilde{h}} + 1 \right)/2 \right\rfloor \\
x_{m,2} &= \left\lfloor \left(x_{m-1,2} - L_1^{\tilde{h}} \right)/2 \right\rfloor \\
y_{m,1} &= \left\lfloor \left(y_{m-1,1} - L_2^{\tilde{h}} + 1 \right)/2 \right\rfloor \\
y_{m,2} &= \left\lfloor \left(y_{m-1,2} - L_1^{\tilde{h}} \right)/2 \right\rfloor
\end{aligned}
$$

Po wyjściu z pętli położenie podobszaru zainteresowania w podobrazie „wygładzonym" (na najwyższym poziomie piramidy) $C_{l,0}$ określają indeksy (i, j):

$$
\begin{aligned}
y_{l,1} \leqslant i \leqslant y_{l,2} \\
x_{l,1} \leqslant j \leqslant x_{l,2}
\end{aligned}
$$

Rysunek 2(a) ilustruje maskę obszarów zainteresowania w 6-poziomowej piramidzie falkowej obrazu o rozdzielczości 512×512 z obszarem zainteresowania o współrzędnych 512×512 z obszarem zainteresowania o współrzędnych $100 \leqslant x \leqslant 200$



(a) (b)

Rys. 2. (a) Ilustracja maski podobszarów zainteresowania w 6-poziomowej piramidzie falkowej obrazu o rozdzielczości $100 \leqslant x < 200$ i $100 \leqslant y < 200$ dla filtrów 5/3. Kolorem czarnym oznaczono podobszary zainteresowania. (b) Ilustracja położenia zerowanych współczynników falkowych w 6-poziomowej piramidzie falkowej obrazu o rozdzielczości 512×512 z obszarem zainteresowania o współrzędnych $100 \leqslant x \leqslant 200$ i $100 \leqslant y \leqslant 200$ dla filtrów 5/3. Kolorem czarnym oznaczono modyfikowane podobszary

Fig. 2. (a) Mask of ROI in 6-level pyramid for image resolution 512×512 and ROI defined as $100 \leqslant x \leqslant 200$, $100 \leqslant y \leqslant 200$, for 5/3 filters. (b) Places of modified wavelet coefficients for ROI shown in Fig. 2a

i $100 \leqslant y \leqslant 200$ dla filtru 5/3. Położenie współczynników falkowych biorących udział w rekonstrukcji obszaru zainteresowania oznaczono kolorem czarnym.

Z rysunku 2(a) widać, że podobszary odpowiadające kwadratowemu obszarowi zainteresowania w dziedzinie obrazu są kwadratami w podmacierzach $C_{j,0}$ i $C_{j,3}$ i prostokątami w podmacierzach $C_{j,1}$ i $C_{j,2}$, co wynika z zastosowania w $C_{j,1}$ i $C_{j,2}$ filtrów o różnych długościach w kierunkach poziomym i pionowym. Z rysunku widać również, że im podobraz $C_{j,k}$ jest na wyższym poziomie piramidy (większe j), tym procentowy udział współczynników falkowych należących do podobszaru zainteresowania jest większy.

Ogólnym sposobem wyznaczenia maski podobszarów zainteresowania w przypadku obszaru zainteresowania o dowolnym kształcie jest wyznaczenie za pomocą wyżej podanych wzorów podobszarów w piramidzie falkowej dla każdego piksela należącego do obszaru zainteresowania. Suma mnogościowa podobszarów w piramidzie falkowej odpowiadających wszystkim pikselom należącym do obszaru zainteresowania stanowi poszukiwaną maskę podobszarów zainteresowania w dziedzinie falkowej.

4. ADAPTACYJNE KWANTOWANIE PIRAMIDY FALKOWEJ OBRAZU Z OBSZARAMI ZAINTERESOWANIA

Idea tej metody polega na wprowadzeniu do piramidy falkowej obrazu dodatkowych drzew zer współczynników falkowych, w wyniku czego polepszają się parametry kompresji, bądź poprzez podwyższenie współczynnika kompresji przy zachowaniu tych samych zniekształceń, bądź poprzez zmniejszenie zniekształceń przy zachowaniu tej samej średniej bitowej.

Drzewo zer, którego korzeniem jest współczynnik falkowy $C[i, j]$ leżący na k-tym poziomie piramidy ($1 \leqslant k \leqslant l$, l-liczba poziomów piramidy) jest zbiorem następujących współczynników falkowych:

$$\left\{C\left[2^m i + u,\ 2^m j + v\right]_{0 \leqslant u,v < 2^m}\right\}_{0 \leqslant m < k}. \tag{2}$$

Liczbę współczynników falkowych tworzących drzewo zer wyraża wzór

$$\sum_{m=0}^{k-1} 2^{2m} = (4^k - 1)/3. \tag{3}$$

W algorytmie kodowania piramidy falkowej SPIHT [14, 10] drzewo zer reprezentowane jest w strumieniu wyjściowym kodera przez jeden symbol binarny. Widać stąd, że tworząc w piramidzie falkowej drzewo zer można zaoszczędzić bardzo wiele bitów niezbędnych do zakodowania wartości współczynników falkowych nietworzących drzewa zer. Nowe drzewa zer można tworzyć poprzez kwantowanie skalarne współczynników piramidy falkowej. W szczególności, jeśli współczynnik kwantowania jest równy $+\infty$, to skwantowane współczynniki przyjmują wartość zero. Utworzone z takich współczynników drzewo zer jest nieznaczące względem wszystkich progów kodowania i nie wprowadza do strumienia wyjściowego żadnych symboli. Poniżej przedstawiony

jest algorytm wyznaczania położenia współczynników falkowych, które można poddać kwantowaniu mającemu na celu utworzenie nowych drzew zer bez pogarszania jakości tła w pobliżu granicy z obszarem zainteresowania. Wymóg niepogarszania jakości tła na granicy z obszarem zainteresowania zawęża liczbę współczynników falkowych odpowiadających obszarom zainteresowania możliwych do modyfikacji tylko do tych, które nie biorą udziału w rekonstrukcji tła. Redukcja zależy oczywiście od długości filtrów użytych do obliczenia transformaty falkowej. Im filtry dłuższe, tym mniejszą liczbę współczynników falkowych podobszarów zainteresowania można zmodyfikować.

Niech obszar zainteresowania w dziedzinie obrazu tworzą piksele o współrzędnych (i, j) takich, że

$$x_1 \leqslant j \leqslant x_2,$$
$$y_1 \leqslant i \leqslant y_2.$$

gdzie x_1 ,x_2 ,y_1 ,y_2 granice obszaru zainteresowania. W celu iteracyjnego wyznaczenia indeksów współczynników falkowych podlegających adaptacyjnemu kwantowaniu należy nadać wartości początkowe zmiennym

$$\begin{aligned} x_{0,1} &= x_1 \\ x_{0,2} &= x_2 \\ y_{0,1} &= y_1 \\ y_{0,2} &= y_2, \end{aligned}$$

a następnie w pętli:
dla $m = 1, 2, \ldots l$ obliczyć indeksy dla poszczególnych podobrazów $C_{m,k}$, $k = 1, 2, 3$ według poniższych wzorów:

$C_{m,1}$:

$$\left\lfloor \left(y_{m-1,1} - L_1^{\tilde{g}} - 1\right)/2 \right\rfloor + 1 \leqslant i \leqslant \left\lfloor \left(y_{m-1,2} - L_2^{\tilde{g}}\right)/2 \right\rfloor$$
$$\left\lfloor \left(x_{m-1,1} - L_1^{\tilde{h}} - 1\right)/2 \right\rfloor + 1 \leqslant j \leqslant \left\lfloor \left(x_{m-1,2} - L_2^{\tilde{h}}\right)/2 \right\rfloor$$

$C_{m,2}$:

$$\left\lfloor \left(y_{m-1,1} - L_1^{\tilde{h}} - 1\right)/2 \right\rfloor + 1 \leqslant i \leqslant \left\lfloor \left(y_{m-1,2} - L_2^{\tilde{h}}\right)/2 \right\rfloor$$
$$\left\lfloor \left(x_{m-1,1} - L_1^{\tilde{g}} - 1\right)/2 \right\rfloor \leqslant j \leqslant \left\lfloor \left(x_{m-1,2} - L_2^{\tilde{g}}\right)/2 \right\rfloor$$

$C_{m,3}$:

$$\left\lfloor \left(y_{m-1,1} - L_1^{\tilde{g}} - 1\right)/2 \right\rfloor + 1 \leqslant i \leqslant \left\lfloor \left(y_{m-1,2} - L_2^{\tilde{g}}\right)/2 \right\rfloor$$
$$\left\lfloor \left(x_{m-1,1} - L_1^{\tilde{g}} - 1\right)/2 \right\rfloor \leqslant j \leqslant \left\lfloor \left(x_{m-1,2} - L_2^{\tilde{g}}\right)/2 \right\rfloor$$

Granice podobszaru kwantowania w podobrazie $C_{m,0}$:

$$\begin{aligned} x_{m,1} &= \left\lfloor \left(x_{m-1,1} - L_1^{\tilde{h}} - 1\right)/2 \right\rfloor + 1 \\ x_{m,2} &= \left\lfloor \left(x_{m-1,2} - L_2^{\tilde{h}}\right)/2 \right\rfloor \\ y_{m,1} &= \left\lfloor \left(y_{m-1,1} - L_1^{\tilde{h}}\right)/2 + 1 \right\rfloor \\ y_{m,2} &= \left\lfloor \left(y_{m-1,2} - L_2^{\tilde{h}}\right)/2 \right\rfloor \end{aligned}$$

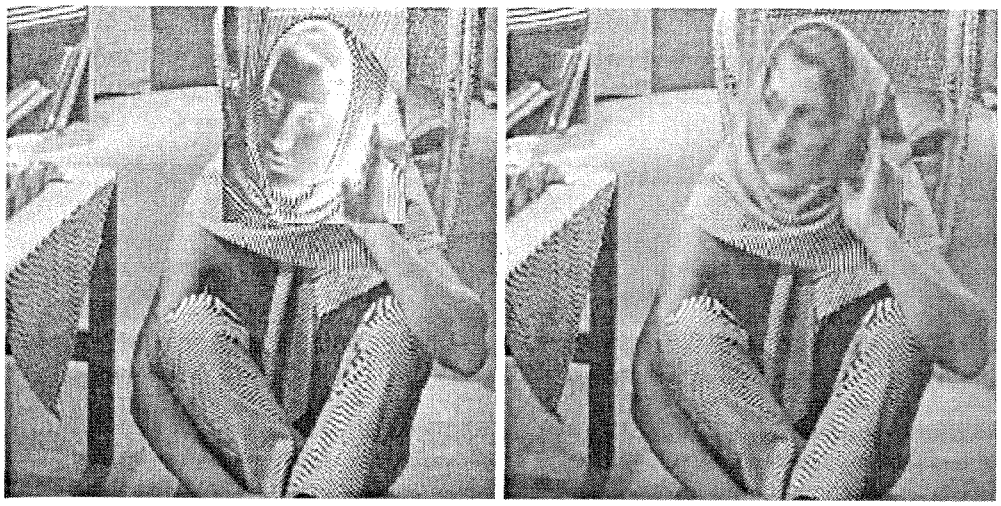
Jeżeli z obliczeń wynika, że indeks i lub j ma należeć do zbioru $\{a, a+1, \ldots, b-1, b\}$ i a jest większe od b, oznacza to, że w tym podobrazie podobszar modyfikacji jest pusty. Rysunek (b) ilustruje położenie modyfikowanych współczynników falkowych w 6-poziomowej piramidzie falkowej obrazu o rozdzielczości 512×512 z obszarem zainteresowania o współrzędnych $100 \leqslant x \leqslant 200$ i $100 \leqslant y \leqslant 200$ dla filtrów 5/3. Położenie wymienionych wyżej współczynników oznaczono kolorem czarnym.

Z rysunku 2(b) widać, że podobszary modyfikowania odpowiadające kwadratowemu obszarowi zainteresowania w dziedzinie obrazu są kwadratami w podmacierzach $C_{j,0}$ i $C_{j,3}$ i prostokątami w podmacierzach $C_{j,1}$ $C_{j,2}$, co wynika z zastosowania w $C_{j,1}$ i $C_{j,2}$ filtrów o różnej długości w kierunku poziomym i pionowym. Z rysunku widać również, że im podobraz $C_{j,k}$ jest na wyższym poziomie piramidy (większe j), tym procentowy udział współczynników falkowych należących do modyfikowanego podobszaru jest mniejszy. W omawianym przykładzie podobszary kwantowania zniknęły na piątym poziomie piramidy falkowej.

W przypadku podobszarów zainteresowania o dowolnym kształcie, np. będących obrazami chromosomów, ogólnym sposobem określenia położenia kwantowanych współczynników falkowych jest wyznaczenie dla każdego piksela należącego do tła współczynników piramidy falkowej, biorących udział w rekonstrukcji tego piksela. Położenie wyznaczonych współczynników falkowych może być zapamiętane w masce binarnej o rozmiarach takich jak obraz. Te współczynniki falkowe, które nie biorą udziału w rekonstrukcji ani jednego piksela tła, nie zostaną zaznaczone w masce i mogą zostać zmodyfikowane.

5. WYNIKI EKSPERYMENTÓW

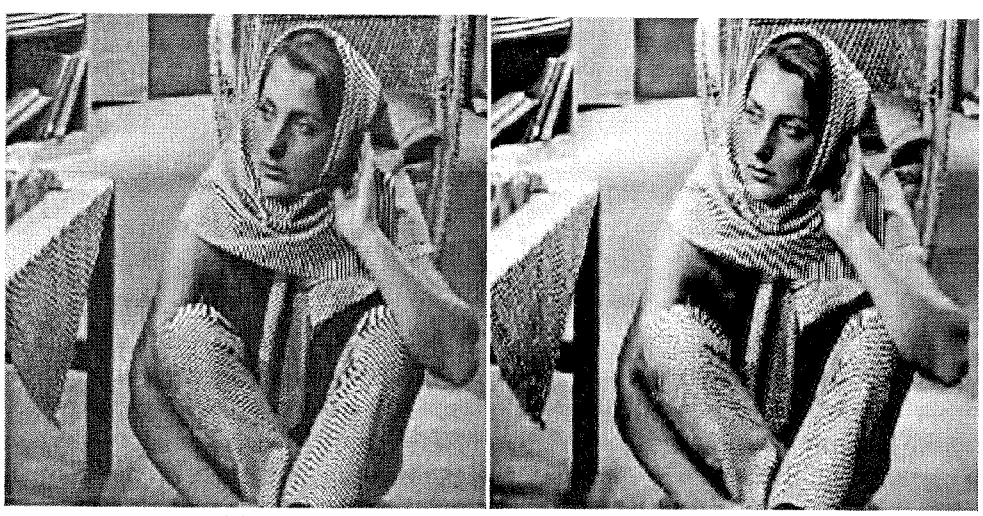
Przedstawiony w poprzednim punkcie algorytm poprawy jakości kompresji tła obrazów z obszarami zainteresowania zastosowano do kilku popularnych w literaturze obrazów. Do kompresji tła wykorzystano metodę falkową z nieodwracalną transformatą falkową, obliczaną z zastosowaniem filtrów FBI 9/7. Wyniki porównano z wynikami kompresji obrazów z obszarami zainteresowania metodą Maxshift zaimplementowaną w systemie kompresji JPEG 2000 [4, 5, 3]. Na rys. 3(a) pokazano obraz oryginalny *Barbara* z wyświetlonym w negatywie obszarem zainteresowania. Na rys. 3(b) widoczne jest wyraźne „rozmycie" obszaru zainteresowania spowodowane modyfikacją



(a) (b)

Rys. 3. Obraz *Barbara* 512 × 512, 8 *bpp*. (a) Oryginał z pokazanym w negatywie obszarem zainteresowania ($224 \leqslant x < 416$, $0 \leqslant y < 224$). (b) Tło skompresowane z modyfikacją piramidy falkowej ze średnią bitową 0,5 *bpp*

Fig. 3. Image *Barbara* 512 × 512, 8 *bpp*. (a) Original image with ROI ($224 \leqslant x < 416$, $0 \leqslant y < 224$) shown as negative subimage. (b) Background compressed with modified wavelet pyramid, bit rate 0.5 *bpp*



(a) (b)

Rys. 4. (a) Obraz *Barbara* zrekonstruowany, z obszarem zainteresowania bez strat i z tłem skompresowanym z modyfikacją piramidy falkowej; średnia bitowa 1,16 *bpp*, PSNR tła równy 32,39 *dB*. (b) Obraz *Barbara* skompresowany metodą Maxshift ze średnią bitową 1,16 *dB*, PSNR tła równy 28,41 *dB*

Fig. 4. (a) Image *Barbara* (with ROI losslessly codded) reconstructed from the modified wavelet pyramid; bite rate equal to 1.16 *bpp*, PSNR of the background equal to 32.39 *dB*. (b) Image *Barbara* with ROI losslessly compressed by the Maxshift method; bite rate — 1.16 *bpp*, PSNR of the background — 28.41 *dB*

piramidy falkowej obrazu, tj. wprowadzeniem dodatkowych drzew zer. Rys. 4(a) pokazuje obraz zrekonstruowany hybrydowo, tj. z „wklejonym" do tła (rys. 3(b)) obszarem zainteresowania. Obszar zainteresowania zakodowano bez strat metodą JPEG-LS ze średnią bitową 4,04 *bpp*. Wypadkowa średnia bitowa wyniosła 1,16 *bpp*, i PSNR tła 32,39 dB. Dla porównania, na rys. 4(b) pokazano wynik kompresji omawianego obrazu metodą Maxshift. Dla tej samej średniej bitowej (1,16 *bpp*) PSNR tła wyniósł 28,41 dB. W tabeli 3 zestawiono wartości różnych miar zniekształceń[1] kompresji tła metodą falkową bez modyfikacji i z modyfikacją piramidy falkowej dla kilku wartości średnich bitowych. Z tabeli widać, że modyfikacja piramidy falkowej poprawia PSNR o ponad 1 *dB*.

Tabela 3

Wielkości zniekształceń kompresji falkowej tła obrazu *Barbara* bez modyfikacji piramidy falkowej (górna część tabeli) i z modyfikacją piramidy falkowej (dolna część tabeli)

Background distortions resulting from nonmodified pyramid method (upper part of the table) and from modified pyramid method (lower part)

średnia bitowa [bpp]	*PSNR* [dB]	*SNR*	*MSE*	*MaxErr*
1.0	36.57	30.16	14.33	21
0.5	31.28	24.88	48.47	48
0.25	26.93	20.53	131.76	79
1.0	37.96	31.55	10.40	20
0.5	32.39	25.99	37.52	38
0.25	28.08	21.68	101.12	72

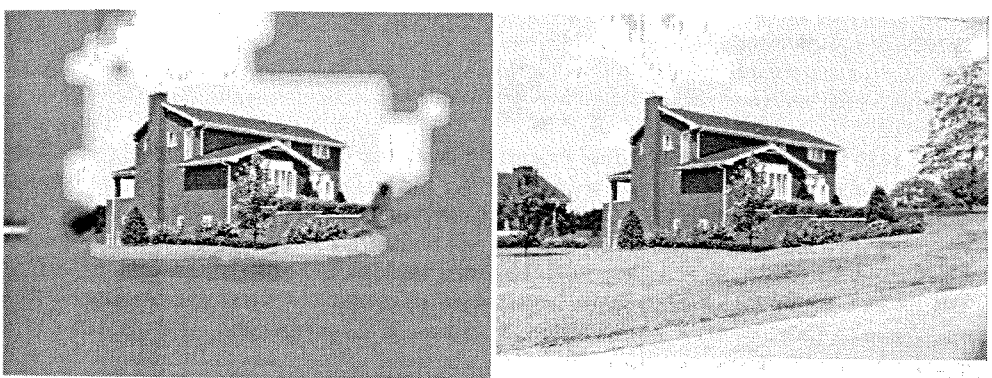
Na rys. 5(a) pokazano obraz *Home* 800 × 600, 24 *bpp*, z zaznaczonym linią przerywaną obszarem zainteresowania. Obszar zainteresowania skompresowano bez strat metodą JPEG-LS ze średnią bitową 12,24 *bpp*. Tło skompresowano z modyfikacją piramidy falkowej ze średnią bitową 0,3214 *bpp*. Wypadkowa średnia bitowa wyniosła 3,00 *bpp*, i PSNR tła 32,67 *dB*. Obraz po rekonstrukcji pokazano na rys. 5(b). Dla porównania, na rys. 6 pokazano efekty zastosowania do omawianego obrazu metody Maxshift. Z rys. 6(a) widać, że przy średniej bitowej 3,00 *bpp* zabrakło bitów na zakodowanie tła. Przy średniej bitowej 3,50 *bpp*, tło zostało zakodowane, ale z niższą jakością (PSNR tła równy 32,29 *bpp*) niż w przypadku metody hybrydowej ze średnią bitową 3,00 *bpp* (PSNR tła 32,67 *dB*).

[1] SNR(Signal to Noisy Ratio), MSE(Mean Square Error), MaxErr — największy moduł różnicy wartości pikseli w obrazie oryginalnym i zrekonstruowanym.

(a) (b)

Rys. 5. Obraz *Home* 800 × 600, 24 *bpp*. (a) Oryginał z zaznaczonym w ramce obszarem zainteresowania ($160 \leqslant x < 580$, $128 \leqslant y < 384$). (b) Obraz zrekonstruowany, z obszarem zainteresowania bez strat i z tłem skompresowanym z modyfikacją piramidy falkowej; średnia bitowa 3,0 *bpp*, PSNR tła równy 32,67 *dB*

Fig. 5. Image *Home* 800 × 600, 24 *bpp*. (a) Original image with ROI defined as $160 \leqslant x < 580$, $128 \leqslant y < 384$. (b) Image reconstructed from modified pyramid; bit rate — 3.0 *bpp*, PSNR of the background — 32.67 *dB*



(a) (b)

Rys. 6. (a) Obraz *Home* skompresowany metodą Maxshift z obszarem zainteresowania bez strat, średnia bitowa 3,0 *bpp*. (b) Obraz *Home* skompresowany metodą Maxshift z obszarem zainteresowania bez strat, średnia bitowa 3,5 *bpp*, PSNR tła równy 31,29 *dB*

Fig. 6. (a) Image *Home* compressed by the Maxshift method; ROI losslessly compressed, bit rate — 3.0 *bpp*. (b) Image *Home* compressed by the Maxshift method; ROI losslessly compressed, bit rate — 3.5 *dB*, PSNR of the background — 31.29 *dB*

6. WNIOSKI KOŃCOWE

W pracy przedstawiono nową metodę kompresji tła obrazu z obszarami zainteresowania, opartą na metodzie falkowej z adaptacyjną modyfikacją piramidy falko-

wej. Zamieszczone w pracy wyniki eksperymentów pokazują, że połączenie dwóch metod: niewnoszącej strat do obszarów zainteresowania i wnoszącej straty metody falkowej z nieodwracalną transformacją falkową i z adaptacyjnym kwantowaniem piramidy falkowej do tła obrazu, dają znaczące percepcyjnie polepszenie jakości kompresji w stosunku do metody Maxshift z systemu kompresji JPEG 2000. Uzupełniające dane dotyczące porównania omówionych tu metod zawierają prace [13, 11, 12].

Praca finansowana częściowo z grantu Rektora Politechniki Białostockiej W/WI/8/03.

7. BIBLIOGRAFIA

1. ISO/IEC JTC 1/SC 29/WG 1. ISO/IEC 14495-1: *Information technology — Lossless and nearlossless compression of continuous-tone still images: Baseline.* December 1999.
2. ISO/IEC JTC 1/SC 29/WG 1. ISO/IEC 14495-2: *Information technology — Lossless and nearlossless compression of continuous-tone still images* — Part 2: Extention, December 1999.
3. ISO/IEC JTC 1/SC 29/WG 1. ISO/IEC 15444-1: *Information technology — JPEG 2000 image coding system: Core coding system [WG 1 N 1890].* September 2000.
4. J. Askelöf, M. L. Carlander, C. Christopoulos: *Region of interest coding in JPEG 2000.* Signal Processing: Image Communication, 17, 2002, pp. 105–111.
5. Ch. Christopoulos, J. Askelof, M. Larsson: *Efficient methods for encoding regions of interest in the upcoming JPEG2000 still image coding standard.* IEEE Signal Processing Letters, 7(9), September 2000, pp. 247–249.
6. Ch. Christopoulos, A. Skodras, T. Ebrahimi: *The JPEG2000 still image coding system: an overview.* IEEE Transactions on Consumer Electronics, 46(4), November 2000, pp. 1103–1127.
7. S. Mallat: *A Wavelet Tour of Signal Processing.* Academic Press, San Diego, 1998.
8. M. W. Marcellin, M. J. Gormish, A. Bilgin, M. P. Boliek: *An overview of JPEG-2000.* Proc. of IEEE Data Compression Conference, 2000, pp. 523–541.
9. W. Rakowski: *Metoda falkowa.* W. Skarbek, red. Multimedia — algorytmy i standardy kompresji, Akademicka Oficyna Wydawnicza PLJ, Warszawa, 1998, ss. 207–256.
10. W. Rakowski: *Kodowanie transformat falkowych dla obrazów.* VI Sympozjum Nowości w Technice Audio — Technika Audio i Multimedia, , Warszawa, 22-23 października 1999, pp. 101–111.
11. W. Rakowski: *Kompresja falkowa obrazów z obszarami zainteresowania kodowanymi bez strat.* Konferencja Naukowo-Techniczna: Systemy i Technologie Telekomunikacji Multimedialnej (STM), Łódź, 2000, ss. 125–131.
12. W. Rakowski: *Metody falkowe kompresji obrazów z obszarami zainteresowania.* Rozprawy Naukowe Nr 99, Wydawnictwo Politechniki Białostockiej, Białystok, 2002.
13. W. Rakowski, W. Skarbek: *Compression of images with regions of interest by adaptive quantisation in wavelet domain.* 11 Portuguese Conference on Pattern Recognition (RECPAD 2000), Porto, Portugal, 11-12 May 2000, pp. 367–371.
14. A. Said, W. A. Pearlman: *A new fast and efficient image codec based on set partitioning in hierarchical trees.* IEEE Transactions on Circuits and Systems for Video Technology, 6, June 1996, pp. 243–250.
15. J. M. Shapiro: *Embedded image coding using zerotrees of wavelet coefficients.* IEEE Transactions on Signal Processing, 41(12), December 1993, pp. 3445–3462.

W. RAKOWSKI

A WAVELET COMPRESSION METHOD OF THE BACKGROUND FOR IMAGES WITH ROI

Summary

There are images in which some regions, called regions of interest (ROI), must be coded losslessly while other parts, called background, can be coded with data loss. These images can be compressed using two different methods: lossless method for ROI and lossy one for background. The wavelet method based on non-reversible wavelet transformation is one the most effective for background compression. Non-reversible wavelet transformations give more effective compressions than reversible transformations. A significant improvement of the background quality can be achieved by modification of the wavelet image pyramid. The modification can be carried out by zeroing these wavelet coefficients which are not involved in background reconstruction. As result of such modification, additional zerotrees are created which can be coded by embedded methods with only one or couple of bits and, in consequence, the image quality in terms of PSNR can increase even by few decibels, depending on the size of ROI. In the paper a way of determining the position of modified wavelet coefficients and experimental results for i couple of images with ROI are presented. The results show significant efficiency of the proposed hybrid method of compression of images with ROI.

Keywords: wavelet transform, image wavelet compression, regions of interest, embedded coding

KWARTALNIK ELEKTRONIKI I TELEKOMUNIKACJI, 2004, 50, z. 1, ss. 65–86

P-warstwowa synteza logiczna dedykowana dla struktur typu PAL

DARIUSZ KANIA

Politechnika Śląska, Instytut Elektroniki,
ul. Akademicka 16, 44-100 Gliwice
e-mail: kdariusz@zeus.polsl.gliwice.pl

Otrzymano 2003.01.10
Autoryzowano 2003.03.05

Artykuł przedstawia *p*-warstwową metodę syntezy dedykowaną dla struktur typu PAL z trójstanowymi buforami wyjściowymi. Metoda ta prowadzi do realizacji układów cyfrowych w postaci struktur *p*-warstwowych. Opracowany algorytm, zaimplementowany w systemie PALDec został wykorzystany do podziału układów testowych. Uzyskane wyniki pokazują wyższość zaprezentowanej metody w porównaniu z metodą klasyczną.

Słowa kluczowe: synteza logiczna, podział, minimalizacja, PLD

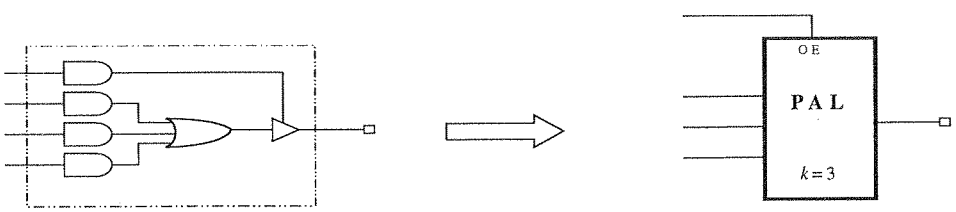
1. WPROWADZENIE

Sposób efektywnego wykorzystania iloczynów stanowi jeden z kluczowych problemów syntezy logicznej, dedykowanej dla struktur programowalnych. Problem ten został zauważony przez producentów układów programowalnych, którzy do struktur CPLD typu PAL wprowadzili sprzętowe mechanizmy umożliwiające nierównomierny rozdział iloczynów pomiędzy komórki wyjściowe [2, 21]. Należą do nich, występujące w strukturach MACH firmy AMD tzw. programowalne rozdzielacze (ang. logic allocator), w układach firmy Altera oraz Xilinx mechanizmy umożliwiające lokalną ekspansję liczby iloczynów (np. MAX 7000, MAX 9000 — shareable, parallel expanders; XC 9500 — product term allocator). Sprzętowe zasoby układów programowalnych umożliwiające nierównomierny podział iloczynów, prowadzące do możliwości tworzenia bloków logicznych typu PAL zawierających różną liczbę iloczynów, mimo swoich niewątpliwych zalet, nie zapewniają realizacji każdej funkcji w jednym bloku

logicznym typu PAL. Konieczne jest wprowadzanie sprzężeń zwrotnych prowadzących do realizacji układów w postaci struktur wielowarstwowych.

Problem zwiększania liczby warstw logicznych towarzyszy nie tylko metodzie klasycznej. Zaawansowane metody syntezy przedstawione między innymi w pracach [1, 16, 17, 18, 20] prowadzą do efektywnego wykorzystywania zasobów logicznych struktur programowalnych lecz redukcja liczby bloków często okupiona jest zwiększeniem liczby warstw logicznych. Tak dzieje się w przypadku metod syntezy wykorzystujących różnorodne modele dekompozycji [1, 5, 6, 7, 8, 10, 11, 15, 16] oraz specjalizowanych metod dedykowanych dla struktur typu PAL [3, 9, 14, 19].

Okazuje się, że problem ekspansji liczby warstw logicznych można rozwiązać wykorzystując występujące w strukturach programowalnych trójstanowe bufory wyjściowe. Elementarne bloki logiczne typu PAL, składające się z iloczynów dołączonych do sumy, często zawierają dodatkowe elementy. W większości struktur programowalnych występują trójstanowe bufory wyjściowe [2]. Wprowadzenie do struktury programowalnej tych elementów przede wszystkim wprowadza możliwość elastycznego konfigurowania wyprowadzeń układu jako wyprowadzenia wejściowe, wyjściowe lub dwukierunkowe [4, 21]. Struktura bloku logicznego typu PAL wyposażonego w sterowany trójstanowy bufor wyjściowy oraz jego schemat blokowy przedstawione są na rysunku 1.



Rys. 1. Struktura oraz schemat blokowy bloku logicznego typu PAL zawierającego trójstanowy bufor wyjściowy

Fig. 1. The structure and symbol of PAL-based logic block with three-state output buffer

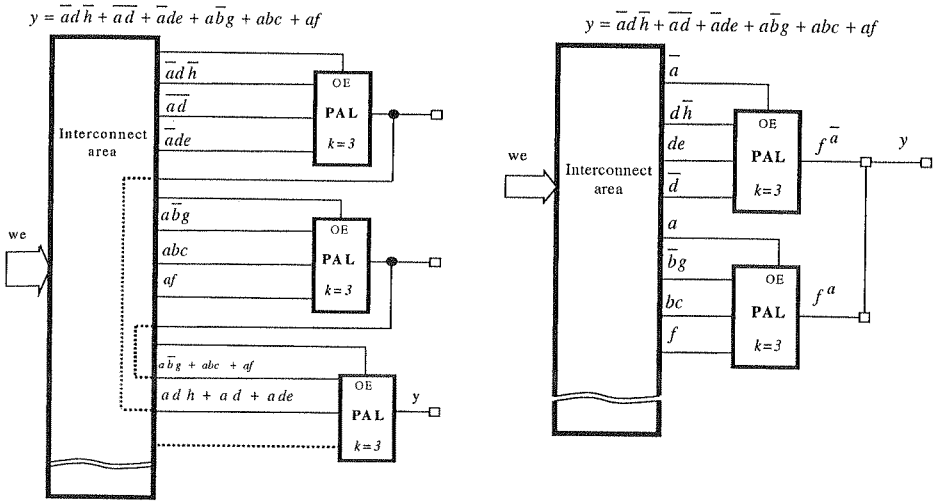
Ze względu na wspólny proces technologiczny, wszystkie elementy struktur programowalnych, posiadają zbliżone parametry dynamiczne. Stwarza to możliwość nietypowego wykorzystania trójstanowych buforów wyjściowych, włączając je w proces syntezy logicznej.

Przykład 1:

Rozważmy funkcję $f : B^8 \rightarrow B^1$, którą po minimalizacji można przedstawić w postaci sumy implikantów $y = f(a,b,c,d,e,f,g,h) = \overline{a}d\overline{h} + \overline{a}\overline{d} + \overline{a}de + a\overline{b}g + abc + af$. Klasyczna metoda realizacji funkcji z minimalizacją liczby warstw logicznych, przedstawiona na rysunku 2a, zajmuje 3 bloki logiczne typu PAL zawierające 3 iloczyny. Możliwa jest inna realizacja tej funkcji, wykorzystująca trójstanowe bufory wyjściowe. Zmienna a dzieli zbiór implikantów funkcji $f : B^8 \rightarrow B^1$ na dwa podzbiory:

jeden zawierający implikanty, w których występuje literał $\bar{a}$ oraz drugi zawierający implikanty z literałem a. Sygnał odpowiadający zmiennej a, zwanej zmienną podziału implikantów, może być wykorzystany do sterowania odpowiedniego trójstanowego bufora wyjściowego.

Realizacja funkcji wykorzystująca trójstanowe bufory wyjściowe, sterowane przez sygnał odpowiadający zmiennej podziału implikantów a, przedstawiona jest na rysunku 2b.



Rys. 2. a.) Klasyczna realizacja przykładowej funkcji ; b.) Realizacja przykładowej funkcji wykorzystująca trójstanowe bufory wyjściowe

Fig. 2. a.) The classical implementation of the exemplary function; b.) The implementation of the exemplary function using three-state output buffers

Przykład 1 pokazuje, że wykorzystanie trójstanowych buforów wyjściowych, doprowadziło do zmniejszenia liczby warstw logicznych oraz liczby użytych bloków logicznych typu PAL.

Głównym problemem syntezy logicznej wykorzystującej trójstanowe bufory wyjściowe jest sposób doboru zmiennych, w ogólnym przypadku wektorów, podziału implikantów. W pracach [12, 13] przedstawione są różnorodne metody doboru wektorów podziału. W pracy [12] przedstawiono syntezę logiczną wielopoziomowych układów, bazującą na odpowiednim doborze zmiennej podziału. W metodzie tej wyszukiwano tylko jedną zmienną podziału, po czym poszczególne podukłady realizowano metodą klasyczną. Krańcowo odmienne podejście zaprezentowano w pracy [13]. Tym razem poszukiwano wektorów podziału zapewniających realizację układów w postaci struktur

jednowarstwowych. Doprowadziło to do znacznego skomplikowania algorytmu wyznaczania wektorów podziału. Niniejsza praca jest kontynuacją poprzednich prac, łączącą w sobie prostotę pierwszego podejścia z zaletami drugiej metody. Z pierwszego pomysłu zaczerpnięto prostotę wyboru zmiennej podziału. Okazuje się, że poprzez odpowiedni wybór zmiennych podziału, wykonywany w kolejnych krokach zaproponowanego algorytmu, można uzyskiwać również rozwiązania jednowarstwowe. Dodatkowo zaproponowana metoda pozwala znajdować rozwiązania o zadanej liczbie warstw logicznych.

Celem pracy jest przedstawienie metody syntezy logicznej dedykowanej dla struktur programowalnych zawierających wyjściowe bufory trójstanowe, stanowiącej rozszerzenie metod zaprezentowanych w pracach [12, 13]. Metoda ta umożliwia realizację układów w postaci struktur o zadanych właściwościach dynamicznych tzn. liczbie warstw logicznych.

2. PODSTAWY TEORETYCZNE

Niech f będzie n-wejściową i 1-wyjściową funkcją logiczną odwzorowującą zbiór B^n w zbiór B^1 tzn. $f : B^n \rightarrow B^1$, gdzie $B = \{0, 1\}$. Funkcja f może być również przedstawiona w postaci $y = f(i_n, \ldots, i_2, i_1)$, przyporządkowując każdemu wektorowi wejściowemu $I = (i_n, \ldots, i_2, i_1)$ wartość y równą 0 lub 1. Nadając zmiennym $i_n, \ldots, i_2, i_1$ wartości zerojedynkowe otrzymujemy stany wejściowe $I_j \subset B^n$; $(j = 0, 1, \ldots, 2^n - 1)$.

Zminimalizowana postać funkcji może być opisana przez zbiór jednowyjściowych implikantów będących zbiorem par wektorów o wymiarach n oraz 1 nazywanych odpowiednio częścią wejściową oraz częścią wyjściową. Część wejściowa składa się z elementów $\{0, 1, -\}$ i reprezentuje iloczyn literałów, dla którego funkcja przyjmuje wartość $\{1\}$, stanowiącą część wyjściową jednowyjściowego implikantu [17].

Niech $\mathbf{A}_f = \{A_{y(l-1)}, \ldots, A_{y1}, A_{y0}\}$ będzie zbiorem części wejściowych jednowyjściowych implikantów, zwanych dalej wektorami wejściowymi. Każdy wektor wejściowy $A_y = (a_n, \ldots, a_2, a_1)$ należący do zbioru $\mathbf{A}_f$ składa się z n-elementów, odpowiadających zmiennym wejściowym i_n, ..., i_2, i_1, przy czym każdy element $a_i \in \mathbf{A}_{0,1,-} = \{0, 1, -\}$, $(i = 1, 2, \ldots, n)$.

Przykład 2: Rozważmy funkcję $f : B^8 \rightarrow B^1$ z przykładu 1. Zminimalizowana postać funkcji $y = f(a, b, c, d, e, f, g, h) = \overline{a}d\overline{h} + \overline{a}\overline{d} + \overline{a}de + a\overline{b}g + abc + af$ może być opisana przez zbiór jednowyjściowych implikantów. Z każdym jednowyjściowym implikantem skojarzony jest wektor wejściowy $A_{yi} = (a, b, c, d, e, f, g, h)$.

a	b	c	d	e	f	g	h	y		
0	-	-	1	-	-	-	0	1	$\Rightarrow$	$A_{y0} = (0--1---0)_{a,b,c,d,e,f,g,h}$
0	-	-	0	-	-	-	-	1	$\Rightarrow$	$A_{y1} = (0--0----)_{a,b,c,d,e,f,g,h}$
0	-	-	1	1	-	-	-	1	$\Rightarrow$	$A_{y2} = (0--11---)_{a,b,c,d,e,f,g,h}$
1	0	-	-	-	-	1	-	1	$\Rightarrow$	$A_{y3} = (10----1-)_{a,b,c,d,e,f,g,h}$
1	1	1	-	-	-	-	-	1	$\Rightarrow$	$A_{y4} = (111-----)_{a,b,c,d,e,f,g,h}$
1	-	-	-	-	1	-	-	1	$\Rightarrow$	$A_{y5} = (1----1--)_{a,b,c,d,e,f,g,h}$

Zbiór wektorów wejściowych, dla których funkcja przyjmuje wartość 1 zawiera sześć elementów $\mathbf{A_f} = \{\mathbf{A_{y5}}, \mathbf{A_{y4}}, \mathbf{A_{y3}}, \mathbf{A_{y2}}, \mathbf{A_{y1}}, \mathbf{A_{y0}}\}$

$$\mathbf{A}_f = \left\{ \begin{array}{l} (0\ -\ -\ 1\ -\ -\ -\ 0)_{abcdefgh} \\ (0\ -\ -\ 0\ -\ -\ -\ -)_{abcdefgh} \\ (0\ -\ -\ 1\ 1\ -\ -\ -)_{abcdefgh} \\ (1\ 0\ -\ -\ -\ -\ 1\ -)_{abcdefgh} \\ (1\ 1\ 1\ -\ -\ -\ -\ -)_{abcdefgh} \\ (1\ -\ -\ -\ -\ 1\ -\ -)_{abcdefgh} \end{array} \right\} \equiv \mathbf{A}_f = \left\{ \begin{array}{l} (0\ -\ -\ 1\ -\ -\ -\ 0\ ) \\ (0\ -\ -\ 0\ -\ -\ -\ -\ ) \\ (0\ -\ -\ 1\ 1\ -\ -\ -\ ) \\ (1\ 0\ -\ -\ -\ -\ 1\ -\ ) \\ (1\ 1\ 1\ -\ -\ -\ -\ -\ ) \\ (1\ -\ -\ -\ -\ 1\ -\ -\ ) \end{array} \right\}_{abcdefgh}$$

Każdą funkcję $f(i_n, \ldots, i_2, i_1)$ można podzielić na dwa składniki względem każdej zmiennej (rozwinięcie Shannona) [4], na przykład

$$f(i_n, \ldots, i_2, i_1) = i_1 f(i_n, \ldots, i_2, 1) + \overline{i_1} f(i_n, \ldots, i_2, 0) = i_1 f_{i_1} + \overline{i_1} f_{\overline{i_1}}$$

Kontynuując rozwijanie można przykładowo otrzymać postać

$$f(i_n, \ldots, i_2, i_1) = i_1 i_2 f_{i_1 i_2} + i_1 \overline{i_2} f_{i_1 \overline{i_2}} + \overline{i_1} i_2 f_{\overline{i_1} i_2} \overline{i_1}\ \overline{i_2} f_{\overline{i_1}\ \overline{i_2}} \qquad (1)$$

Niech funkcja f^x opisuje zachowanie wyjścia bloku logicznego typu PAL z trójstanowym buforem wyjściowym sterowanym przez sygnał odpowiadający zmiennej x.

Ponieważ

$$f^x = \left\{ \begin{array}{ll} f_x & \text{dla } x = 1 \\ \text{stan wysokiej impedancji} & \text{dla } x = 0 \end{array} \right.$$

stąd funkcję $f(i_n, \ldots, i_2, i_1)$ przedstawioną w postaci (1) można zrealizować łącząc wyjścia bloków logicznych z trójstanowymi buforami wyjściowymi (rys. 3), których zachowanie wyjść opisują funkcje $f^{i_1 i_2}$, $f^{i_1 \overline{i_2}}$, $f^{\overline{i_1} i_2}$, $f^{\overline{i_1}\ \overline{i_2}}$.

Powyższą zasadę rozwinięć można wykorzystać do podziału zbioru $\mathbf{A}_f$ i realizacji uzyskanych w ten sposób grup implikantów na blokach logicznych o zadanej liczbie iloczynów.

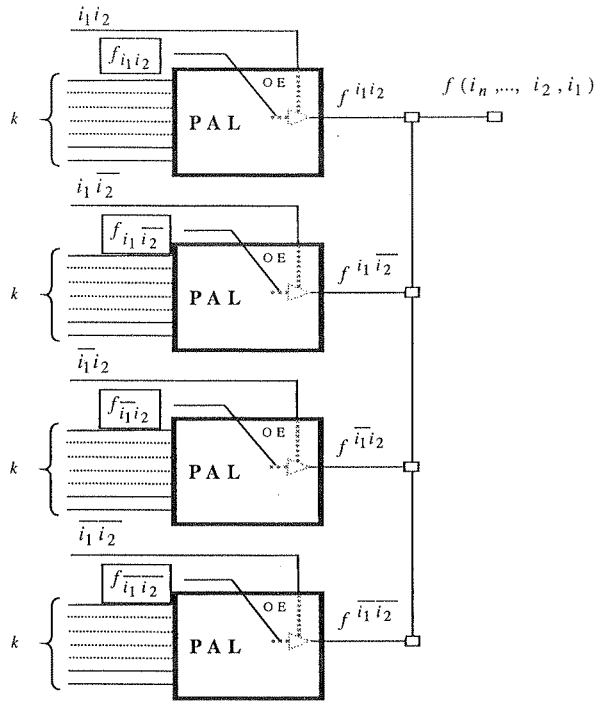
Niech $X = (x_p, \ldots, x_2, x_1)$ będzie p-elementowym ($p < n$) wektorem, w którym poszczególne elementy odpowiadają wybranym elementom wektorów wejściowych $A_y = (a_n, \ldots, a_2, a_1)$, przy czym każdy element $x_i \in \mathbf{X_{0,1}} = \{0, 1\}$, $(i = 1, 2, \ldots, p)$.

Dowolny i-ty element wektora wejściowego $A_y = (a_n, \ldots, a_2, a_1)$ i wektora $X = (x_p, \ldots, x_2, x_1)$, gdzie $(i = 1, 2, \ldots, p)$ odpowiada tej samej zmiennej. Uporządkowane pary $\langle a_i, x_i \rangle$ należą do zbioru par $\mathbf{A}$ będącego iloczynem kartezjańskim zbiorów $\mathbf{A_{0,1,-}}$ i $\mathbf{X_{0,1}}$; $(\mathbf{A} = \mathbf{A_{0,1,-}} \times \mathbf{X_{0,1}})$.

Definicja 1:

Dwa wektory $A_y = (a_n, \ldots, a_2, a_1)$ i $X = (x_p, \ldots, x_2, x_1)$ takie, że $p \leqslant n$ i $a_i \in \mathbf{A}_{0,1,-} = \{0, 1, -\}$, $x_i \in \mathbf{X}_{0,1} = \{0, 1\}$ nazywamy ***wektorami niesprzecznymi*** wtedy i tylko wtedy, gdy zbiór uporządkowanych p-par $\langle a_i, x_i \rangle$ nie zawiera elementów $\langle 0, 1 \rangle$ i $\langle 1, 0 \rangle$.

np.: wektory $A_y = (1\,0\,-\,1\,-)_{abcde}$ i $X = (0\,1\,1)_{bcd}$ są wektorami niesprzecznymi



Rys. 3. Realizacja funkcji $f(i_n, \ldots, i_2, i_1)$ wykorzystująca bloki logiczne typu PAL z trójstanowymi buforami wyjściowymi

Fig. 3. The implementation of function $f(i_n, \ldots, i_2, i_1)$ using PAL-based logic blocks with three-state output buffers

Definicja 2:

Dwa wektory $A_y = (a_n, \ldots, a_2, a_1)$ i $X = (x_p, \ldots, x_2, x_1)$ takie, że $p \leqslant n$ i $a_i \in \mathbf{A}_{0,1,-} = \{0, 1, -\}$, $x_i \in \mathbf{X}_{0,1} = \{0, 1\}$ nazywamy ***wektorami niepseudorównoważnymi*** wtedy i tylko wtedy, gdy w zbiorze uporządkowanych p-par $\langle a_i, x_i \rangle$ istnieje para nie należąca do zbioru $\{\langle 0, 0\rangle, \langle 1, 1\rangle\}$.

np.: wektory $A_y = (1\,0\,\text{-}\,1\,\text{-})_{abcde}$ i $X = (1\,0\,0)_{bcd}$ są wektorami niepseudorównoważnymi

Dowolny wektor $X = (x_p, \ldots, x_2, x_1)$ nazywany ***wektorem podziału*** implikuje podział zbioru $\mathbf{A}_f$ wektorów wejściowych, dla których funkcja przyjmuje wartość 1 na dwa podzbiory $\mathbf{A}_{fx}$ oraz ${}^*\mathbf{A}_{fx}$, pierwszy zawierający wektory niesprzeczne i drugi wektory niepseudorównoważne z wybranym wektorem podziału.

W sytuacji, gdy wektor podziału zredukowany jest do pojedynczej zmiennej x, zachodzą następujące zależności

$$\mathbf{A}_{f_x} = {}^*\mathbf{A}_{f_{\bar{x}}} \qquad \mathbf{A}_{f_{\bar{x}}} = {}^*\mathbf{A}_{f_x}$$

Przykład 3:

Rozważmy funkcję $f\colon B^8 \to B^1$ z przykładu 1. Zbiór wektorów, dla których funkcja przyjmuje wartość 1 wynosi

$$\mathbf{A}_f = \left\{ \begin{array}{l} (0\ -\ -\ 1\ -\ -\ -\ 0\ ) \\ (0\ -\ -\ 0\ -\ -\ -\ -\ ) \\ (0\ -\ -\ 1\ 1\ -\ -\ -\ ) \\ (1\ 0\ -\ -\ -\ -\ 1\ -\ ) \\ (1\ 1\ 1\ -\ -\ -\ -\ -\ ) \\ (1\ -\ -\ -\ -\ 1\ -\ -\ ) \end{array} \right\}_{abcdefgh}$$

Niech wektor podziału $X = (1\,0)_{ab}$. Zbiory wektorów niesprzecznych oraz niepseudorównoważnych z zadanym wektorem podziału przedstawione są poniżej

Wektory niesprzeczne z wektorem $X = (10)_{ab}$

$$^*\mathbf{A}_f = \left\{ \begin{array}{l} (1\ 0\ -\ -\ -\ -\ 1\ -\ ) \\ (1\ -\ -\ -\ -\ 1\ -\ -\ ) \end{array} \right\}_{abcdefgh}$$

Wektory niepseudorównoważne z wektorem $X = (10)_{ab}$

$$\mathbf{A}_f = \left\{ \begin{array}{l} (0\ -\ -\ 1\ -\ -\ -\ 0\ ) \\ (0\ -\ -\ 0\ -\ -\ -\ -\ ) \\ (0\ -\ -\ 1\ 1\ -\ -\ -\ ) \\ (1\ 1\ 1\ -\ -\ -\ -\ -\ ) \\ (1\ -\ -\ -\ -\ 1\ -\ -\ ) \end{array} \right\}_{abcdefgh}$$

Powyższy podział zbioru $\mathbf{A}_f$ odpowiada następującemu rozwinięciu funkcji

$$f(a,b,c,d,e,f,g,h) = a\bar{b}f_{a\bar{b}} + \overline{a\bar{b}}f_{\overline{a\bar{b}}}$$

gdzie

$$\overline{a\bar{b}}f_{\overline{a\bar{b}}} = abf_{ab} + \bar{a}bf_{\bar{a}b} + \bar{a}\bar{b}f_{\bar{a}\bar{b}}$$

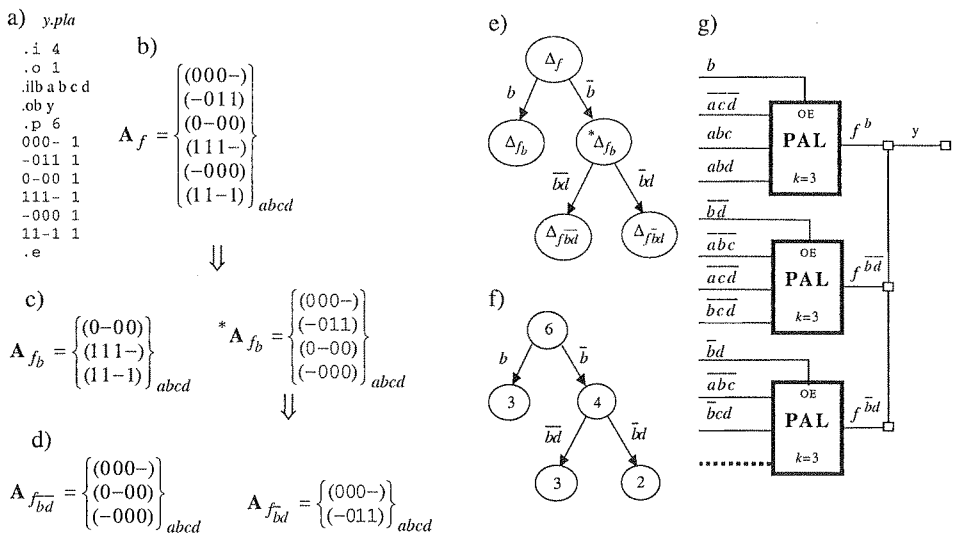
Niech wyróżnik $\Delta_{fx} = \overline{\mathbf{A}_{fx}}$ oraz wyróżnik sprzężony $^*\Delta_{fx} = \overline{^*\mathbf{A}_{fx}}$. W przypadku, gdy wektor podziału zredukowany jest do pojedynczej zmiennej x zachodzą następujące równości pomiędzy wyróżnikami: $\Delta_{f_x} = {}^*\Delta_{f_{\bar{x}}}$ oraz $\Delta_{f_{\bar{x}}} = {}^*\Delta_{f_x}$.

Realizacja funkcji logicznej w oparciu o bloki logiczne typu PAL z trójstanowymi buforami wyjściowymi jest ściśle związana z odpowiednim podziałem zbioru wektorów wejściowych (implikantów) w taki sposób, aby poszczególne części były realizowalne na blokach logicznych zawierających k-iloczynów. Proces podziału zbioru $\mathbf{A}_f$ bezpośrednio łączy się z doborem odpowiednich wektorów podziału.

Niech $G\langle \mathbf{Y}, \vec{\mathbf{U}} \rangle$ będzie grafem skierowanym, gdzie $\mathbf{Y}$ jest zbiorem wszystkich wierzchołków skojarzonych z wartościami wyróżników Δ_{fx} oraz $^*\Delta_{fx}$ opisujących kolejne kroki podziału, natomiast $\vec{\mathbf{U}}$ zbiorem krawędzi łączących wierzchołki (Y_i, Y_j) takie, że wierzchołek Y_j skojarzony jest ze zbiorem będącym wynikiem podziału zbioru, odpowiadającego wierzchołkowi Y_i. Z każdą krawędzią grafu związany jest wektor podziału.

Przykład 4:

Rozważmy funkcję $f : B^4 \rightarrow B^1$, która po minimalizacji przyjmuje postać przedstawioną w pliku *f.pla* (rys. 4a). Załóżmy, że poszukujemy realizacji funkcji na blokach logicznych typu PAL z trójstanowymi buforami wyjściowymi zawierających 3 iloczyny. Zbiór wektorów wejściowych $\mathbf{A}_f$, dla których funkcja $y = f(a, b, c, d)$ przyjmuje wartość 1 przedstawiony jest na rysunku 4b. Zmienna podziału b implikuje podział zbioru $\mathbf{A}_f$ na dwa podzbiory $\mathbf{A}_{f_b}$ oraz ${}^*\mathbf{A}_{f_b} = \mathbf{A}_{f_{\bar{b}}}$ przedstawione na rysunku 4c. Ponieważ, $\overline{{}^*\mathbf{A}_{f_b}} = {}^*\Delta_{f_b} > k$ stąd konieczne jest dalszy podział zbioru ${}^*\mathbf{A}_{f_b}$. Dalszy podział implikowany przez zmienną d przedstawiony jest na rysunku 4d. Proces podziału zbioru $\mathbf{A}_f$ o mocy Δ_f można przedstawić za pomocą grafu podziału (rys. 4e, 4f). Ostateczna realizacja przykładowej funkcji uwidoczniona jest na rysunku 4g.



Rys. 4. Realizacja przykładowej funkcji wykorzystująca bloki typu PAL z trójstanowymi buforami wyjściowymi; a) opis funkcji w postaci pliku *y.pla*; b) zbiór wektorów wejściowych $\mathbf{A}_f$, dla których funkcja $y = f(a, b, c, d)$ przyjmuje wartość 1; c,d) etapy podziału zbioru $\mathbf{A}_f$; e,f) grafy podziału; g) struktura układu

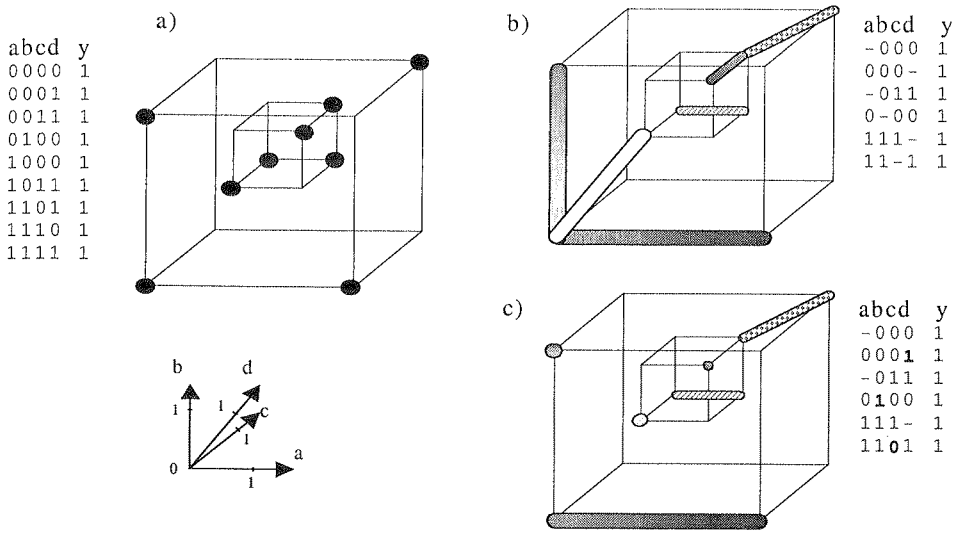
Fig. 4. The implementation of the exemplary function using PAL-based logic blocks with three-state output buffers; a) Description of the function with the aid of *y.pla* file; b) The set of input vectors $\mathbf{A}_f$, for which the function becomes 1 value; c,d) The steps of partition of $\mathbf{A}_f$ set; e,f) The partition graph; g) The structure of circuits

Wybór zmiennej (wektora) podziału znacząco wpływa na liczbę wykorzystywanych bloków logicznych typu PAL z trójstanowymi buforami wyjściowymi. Różnorodne strategie doboru zmiennych zostały przedstawione w pracach [9, 12, 13]. Okazuje

się, że nie tylko dobór zmiennych (wektorów) podziału wpływa na efektywność wykorzystywania bloków logicznych typu PAL. Niezwykle istotne znaczenie ma sposób minimalizacji.

2.1. MINIMALIZACJA Z ROZŁĄCZANIEM IMPLIKANTÓW

Celem minimalizacji funkcji logicznych jest zredukowanie wyrażeń będących sumą iloczynów lub iloczynem sum. W przypadku struktur programowalnych, opartych na matrycach AND-OR, istotna jest optymalizacja wyrażenia będącego sumą iloczynów.



Rys. 5. Koncepcja minimalizacji dwupoziomowej z rozłączaniem implikantów a) przedstawienie funkcji przed minimalizacją; b) wynik klasycznej minimalizacji dwupoziomowej; c) wynik minimalizacji dwupoziomowej z rozłączaniem implikantów

Fig. 5. Concept of two-level splitting minimization; a) The function before minimization; b.) Result of classical minimization; c) Result of two-level splitting minimization

Opracowane przez Quine'a i McCluskey'a algorytmy minimalizacji związane są z minimalizacją wyrażeń dwupoziomowych. W późniejszym okresie zaproponowano kilka heurystycznych algorytmów bazujących na strategiach iteracyjnych ulepszeń. Przykładem programów wykorzystujących heurystyczne algorytmy minimalizacji są MINI, PRESTO oraz ESPRESSO, będący obecnie standardowym narzędziem optymalizacji dwupoziomowych wyrażeń logicznych.

Celem minimalizacji dwupoziomowej funkcji logicznych w algorytmach Quine'a McCluskey'a oraz Espresso jest minimalizacja liczby iloczynów oraz liczby literałów występujących w poszczególnych iloczynach. W przypadku realizacji funkcji w strukturach programowalnych redukcja literałów jest procesem nieistotnym. Dodatkowo znacznie utrudnia dobór wektorów podziału i bardzo niekorzystnie wpływa na ostateczną

realizację funkcji w oparciu o bloki logiczne typu PAL z trójstanowymi buforami wyjściowymi.

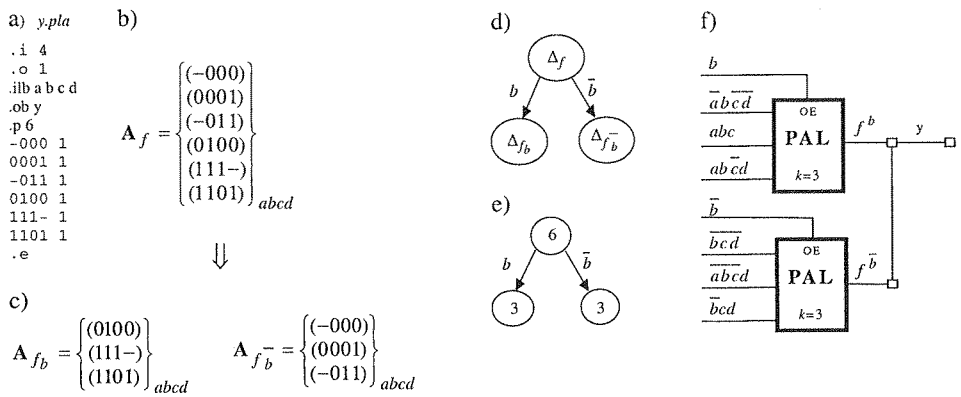
Z punktu widzenia optymalizacji podziału zbioru implikantów na grupy realizowane w poszczególnych blokach logicznych typu PAL celem minimalizacji powinna być redukcja liczby iloczynów i maksymalizacja liczby literałów w poszczególnych iloczynach. Powyższy cel spełnia tzw. dwupoziomowa minimalizacja z rozłączaniem implikantów [13].

Idea rozłączania dla przykładowej funkcji przedstawiona jest na rysunku 5.

Dokładny opis algorytmu rozłączania implikantów można znaleźć w pracy [13].

Przykład 5:

Rozważmy ponownie realizację funkcji analizowanej w przykładzie 4 i na rysunku 5. W wyniku dwupoziomowej minimalizacji z rozłączaniem wektorów (rys. 5) pojawiła się możliwość doboru zmiennej podziału wektorów b zapewniającej realizację funkcji tylko na dwóch blokach logicznych typu PAL z trójstanowymi buforami wyjściowymi, zawierających 3 iloczyny. Poszczególne etapy syntezy, graf podziału oraz ostateczna realizacja przedstawione są na rysunku 6.



Rys. 6. Realizacja przykładowej funkcji wykorzystująca bloki typu PAL z trójstanowymi buforami wyjściowymi; a) opis funkcji w postaci pliku *y.pla*; b) zbiór wektorów wejściowych $\mathbf{A}_f$, dla których funkcja $y = f(a, b, c, d)$ przyjmuje wartość 1; c) podział zbioru $\mathbf{A}_f$; implikowany przez zmienną b; d,e) grafy podziału; f) struktura układu

Fig. 6. Implementation of exemplary function using PAL-based logic block with three-state output buffers; a) Description of the function with the aid of *y.pla* file; b) The set of input vectors $\mathbf{A}_f$, for which the function $y = f(a, b, c, d)$ becomes 1 value; c) Partition of $\mathbf{A}_f$ set implied by b variable; d,e) The partition graphs; f) The structure of circuits

3. KONCEPCJA IMPLEMENTACJI FUNKCJI W STRUKTURACH TYPU PAL Z TRÓJSTANOWYMI BUFORAMI WYJŚCIOWYMI

Synteza logiczna prowadząca do realizacji funkcji $f : B^n \to B^m$ w strukturach typu PAL z trójstanowymi buforami wyjściowymi składa się z kilku etapów. Proces syntezy rozpoczyna klasyczna dwupoziomowa minimalizacja wykorzystująca algorytm Espresso. Minimalizacja wykonywana jest dla każdej funkcji $f_i : B^n \to B^1$ $(i = 1, 2, \ldots, m)$ oddzielnie (espresso -Dso).

Następnie, kolejno rozpatrywane są poszczególne, zminimalizowane funkcje $f_i : B^n \to B^1$. Dla każdej z nich wykonana jest procedura rozłączanie implikantów [13]. Celem następnego etapu jest podział zbiorów jednowyjściowych implikantów opisujących poszczególne funkcje, na podzbiory realizowalne w blokach logicznych typu PAL o zadanej liczbie iloczynów. Procedura podziału opiera się na odpowiednim doborze zmiennych (wektorów) podziału. Sposób wyboru zmiennych (wektorów) podziału ściśle związany jest z ostateczną implementacją funkcji w strukturach typu PAL z trójstanowymi buforami wyjściowymi.

3.1. METODA WYSZUKIWANIA ZMIENNEJ PODZIAŁU, WYKORZYSTYWANA W *P*-WARSTWOWEJ IMPLEMENTACJI FUNKCJI W STRUKTURACH TYPU PAL Z TRÓJSTANOWYMI BUFORAMI WYJŚCIOWYMI

Klasyczna realizacja funkcji f_i: $B^n \to B^1$ będącej sumą Δ_{f_i} - implikantów wymaga użycia $\delta_{f_i} = \left\lceil \frac{\Delta_{f_i} - k}{k-1} \right\rceil + 1$ bloków logicznych typu PAL zawierających k-iloczynów i prowadzi do struktury co najmniej $\xi_{f_i} = \left\lceil \frac{\log(\Delta_{f_i})}{\log k} \right\rceil$ warstwowej (realizacja klasyczna z minimalizacją liczby warstw). Wykorzystanie trójstanowych buforów wyjściowych stwarza możliwość ograniczenia liczby warstw. Zmienna podziału x dzieli zbiór wektorów wejściowych $\mathbf{A}_{f_i}$ na dwa podzbiory $\mathbf{A}_{f_{i_x}}$ i $\mathbf{A}_{f_{i_{\bar{x}}}}$ o mocy Δf_{i_x} oraz $\Delta f_{i_{\bar{x}}}$. Jeżeli wartości wyróżników Δf_{i_x} oraz $\Delta f_{i_{\bar{x}}}$ są większe od liczby iloczynów zawartych w blokach logicznych typu PAL, zachodzi konieczność wprowadzania sprzężeń zwrotnych (klasyczna realizacja wektorów wejściowych należących do zbiorów $\mathbf{A}_{f_{i_x}}$ oraz $\mathbf{A}_{f_{i_{\bar{x}}}}$). Prowadzi to do wykorzystania $\delta_{f_i} = \delta_{f_{i_x}} + \delta_{f_{i_{\bar{x}}}} = \left(\left\lceil \frac{\Delta_{f_{i_x}} - k}{k-1} \right\rceil + 1\right) + \left(\left\lceil \frac{\Delta_{f_{i_{\bar{x}}}} - k}{k-1} \right\rceil + 1\right)$ bloków logicznych typu PAL zawierających k-iloczynów i uzyskania struktury $\xi_{f_i} = \max(\xi_{f_{i_x}}, \xi_{f_{i_{\bar{x}}}}) = \max\left(\left\lceil \frac{\lg(\Delta_{f_{i_x}})}{\lg k} \right\rceil, \left\lceil \frac{lg(\Delta_{f_{i_{\bar{x}}}})}{\lg k} \right\rceil\right)$ warstwowej.

Dla każdej funkcji f_i: $B^n \to B^1$ można wybrać n-zmiennych podziału. Głównym kryterium wyboru zmiennej podziału jest minimalna liczba warstw logicznych uzyskanej struktury. W sytuacji, gdy dla kilku zmiennych uzyskuje się identyczną liczbę warstw wybierana jest zmienna, która zapewnia minimalizację liczby użytych bloków logicznych typu PAL.

Przykład 6:

Rozważmy funkcję $f\colon B^7 \to B^1$, która po dwupoziomowej minimalizacji z rozłączaniem implikantów przyjmuje postać przedstawioną w pliku *f.pla* rys. 7a). Załóżmy, że poszukujemy realizacji funkcji na blokach logicznych typu PAL z trójstanowymi buforami wyjściowymi, zawierających 3 iloczyny. W tabeli przedstawionej na rysunku 7b zawarte są wartości wyróżników oraz parametry charakteryzujące podział dla wszystkich możliwych zmiennych podziałów.

a) *f.pla*

```
.i 7
.o 1
.ilb abcdegh
.ob. y
.p 12
----101 1
----010 1
0-0-110 1
1-1-001 1
0-10110 1
110-001 1
0011110 1
000-011 1
11-1100 1
0010011 1
1110100 1
0101000 1
.e
```

b)

	a	b	c	d	e	g	h
	–	–	–	–	1	0	1
	–	–	–	–	0	1	0
	0	–	0	–	1	1	0
	1	–	1	–	0	0	1
	0	–	1	0	1	1	0
	1	1	0	–	0	0	1
	0	0	1	1	1	1	0
	0	0	0	–	0	1	1
	1	1	–	1	1	0	0
	0	0	1	0	0	1	1
	1	1	1	0	1	0	0
	0	1	0	1	0	0	0
	$\Delta_{f_{\bar{a}}}=8$	$\Delta_{f_{\bar{b}}}=8$	$\Delta_{f_{\bar{c}}}=7$	$\Delta_{f_{\bar{d}}}=9$	$\Delta_{f_{\bar{e}}}=6$	$\Delta_{f_{\bar{g}}}=6$	$\Delta_{f_{\bar{h}}}=7$
	$\Delta_{f_a}=6$	$\Delta_{f_b}=9$	$\Delta_{f_c}=8$	$\Delta_{f_d}=9$	$\Delta_{f_e}=6$	$\Delta_{f_g}=6$	$\Delta_{f_h}=5$
	$\xi_{f_{\bar{a}}}=2$	$\xi_{f_{\bar{b}}}=2$	$\xi_{f_{\bar{c}}}=2$	$\xi_{f_{\bar{d}}}=2$	$\xi_{f_{\bar{e}}}=2$	$\xi_{f_{\bar{g}}}=2$	$\xi_{f_{\bar{h}}}=2$
	$\xi_{f_a}=2$	$\xi_{f_b}=2$	$\xi_{f_c}=2$	$\xi_{f_d}=2$	$\xi_{f_e}=2$	$\xi_{f_g}=2$	$\xi_{f_h}=2$
	$\delta_{f_{\bar{a}}}=4$	$\delta_{f_{\bar{b}}}=4$	$\delta_{f_{\bar{c}}}=3$	$\delta_{f_{\bar{d}}}=4$	$\delta_{f_{\bar{e}}}=3$	$\delta_{f_{\bar{g}}}=3$	$\delta_{f_{\bar{h}}}=3$
	$\delta_{f_a}=3$	$\delta_{f_b}=4$	$\delta_{f_c}=4$	$\delta_{f_d}=4$	$\delta_{f_e}=3$	$\delta_{f_g}=3$	$\delta_{f_h}=2$
ξ_f	2	2	2	2	2	2	2
δ_f	7	8	7	8	6	6	**5**

Rys. 7. Metoda wyszukiwania zmiennej podziału, prowadząca do p-warstwowej implementacji funkcji w strukturach typu PAL z trójstanowymi buforami wyjściowymi a) opis funkcji w postaci pliku *f.pla*; b) tabela zawierająca podstawowe parametry

Fig. 7. Method of searching for the partitioning variable providing to realization of the function by means of PAL-based logic blocks with three-state output buffers; a) Description of the function with the aid of *y.pla* file; b) Table with essential parameters

Ponieważ dla wszystkich zmiennych podziału uzyskiwana jest identyczna liczba warstw logicznych ($\xi_f = 2$), stąd jako zmienna podziału wybrana zostaje zmienna h zapewniająca wykorzystanie minimalnej liczby bloków logicznych ($\delta_f = 5$). Wynik podziału zbioru $\mathbf{A}_f$, implikowany przez zmienną h, graf podziału oraz ostateczną realizację przykładowej funkcji przedstawia rys. 8.

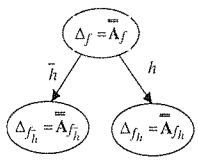
Możliwe jest oczywiście poszukiwanie kolejnych zmiennych podziału, dla wektorów tworzących zbiory $\mathbf{A}_{f_h}$ i $\mathbf{A}_{f_{\bar{h}}}$ umożliwiając ostatecznie realizację funkcji w postaci jednowarstwowej struktury. Graf podziału oraz odpowiadająca mu jednowarstwowa realizacja przykładowej funkcji przedstawione są na rysunku 9.

a) *f.pla*

```
.i 7
.o 1
.ilb abcdegh
.ob. y
.p 12
----101 1
----010 1
0--0-110 1
1-1-001 1
0-10110 1
110-001 1
0011110 1
000-011 1
11-1100 1
0010011 1
1110100 1
0101000 1
.e
```

b)

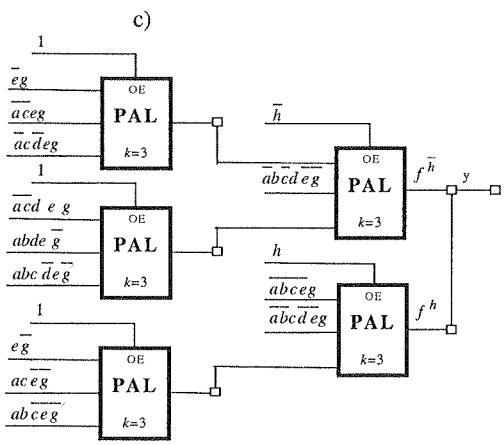
$$\mathbf{A}_f = \left\{ \begin{array}{ccccccc} (- & - & - & - & 1 & 0 & 1) \\ (- & - & - & - & 0 & 1 & 0) \\ (0 & - & 0 & - & 1 & 1 & 0) \\ (1 & - & 1 & - & 0 & 0 & 1) \\ (0 & - & 1 & 0 & 1 & 1 & 0) \\ (1 & 1 & 0 & - & 0 & 0 & 1) \\ (0 & 0 & 1 & 1 & 1 & 1 & 0) \\ (0 & 0 & 0 & - & 0 & 1 & 1) \\ (1 & 1 & - & 1 & 1 & 0 & 0) \\ (0 & 0 & 1 & 0 & 0 & 1 & 1) \\ (1 & 1 & 1 & 0 & 1 & 0 & 0) \\ (0 & 1 & 0 & 1 & 0 & 0 & 0) \end{array} \right\}_{abcdegh}$$



$$\mathbf{A}_{f_{\bar{h}}} = \left\{ \begin{array}{c} (----010) \\ (0-0-110) \\ (0-10110) \\ (0011110) \\ (11-1100) \\ (1110100) \\ (0101000) \end{array} \right\}_{abcdegh}$$

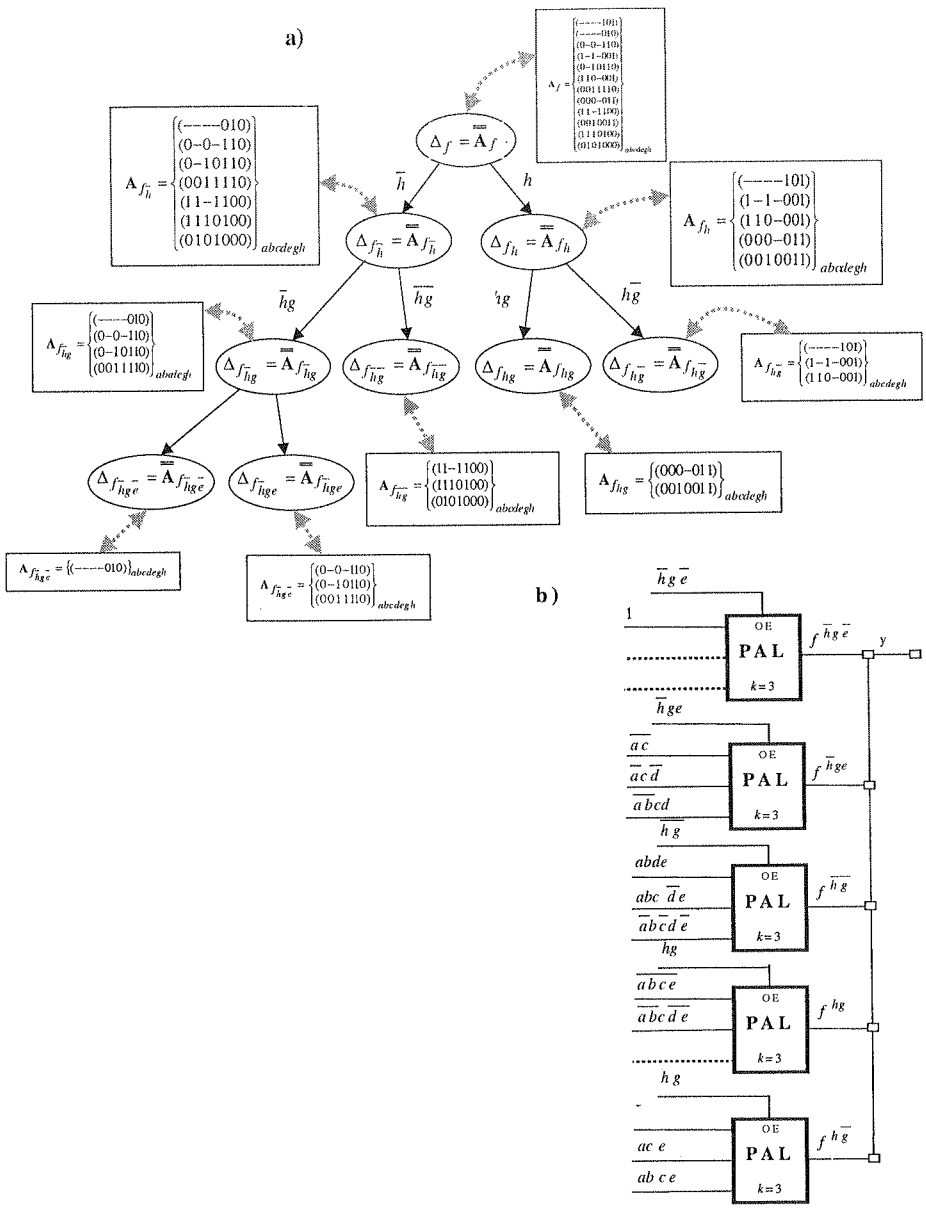
$$A_{f_h} = \left\{ \begin{array}{c} (----101) \\ (1-1-001) \\ (110-001) \\ (000-011) \\ (0010011) \end{array} \right\}_{abcdegh}$$

c)



Rys. 8. Realizacja przykładowej funkcji $f\colon B^7 \to B^1$ wykorzystująca bloki typu PAL z trójstanowymi buforami wyjściowymi; a) opis funkcji w postaci pliku *f.pla*; b) podział zbioru $\mathbf{A}_f$; c) struktura układu

Fig. 8. Implementation of exemplary function $f\colon B^7 \to B^1$ using PAL-based logic block with three-state output buffers; a) Description of the function with the aid of *y.pla* file; b) Partition of the $\mathbf{A}_f$ set; c) The structure of circuits



Rys. 9. Jednowarstwowa realizacja przykładowej funkcji; a) graf podziału; b) struktura układu

Fig. 9. One-level implementation of the exemplary function; a) The partition graphs; b) The structure of circuits

3.2. ALGORYTM P-WARSTWOWEJ IMPLEMENTACJI FUNKCJI NA BLOKACH LOGICZNYCH TYPU PAL Z TRÓJSTANOWYMI BUFORAMI WYJŚCIOWYMI (METODA B_PW)

1. Minimalizacja funkcji $f\colon B^n \to B^m$; każda funkcja $f_i\colon B^n \to B^1$ $(i = 1,2,\ldots,m)$ minimalizowana jest oddzielnie (espresso -Dso)
   W dalszej części algorytmu kolejno rozpatrywane są wszystkie zminimalizowane funkcje $f_i\colon B^n \to B^1$ $(i = 1,2,\ldots,m)$
2. Rozłączanie implikantów [13]
3. Dla wszystkich n-zmiennych wejściowych wyznaczane są wartości wyróżników $\Delta_{fi_{\overline{x_j}}}$ oraz $\Delta_{fi_{x_j}}$ $(j = 1,2,\ldots,n)$
4. Wybór zmiennej podziału x, dla której

$$\xi_{f_i} = \max(\xi_{f_{i_x}}, \xi_{f_{i_{\bar{x}}}}) = \max\left(\left\lceil \frac{\lg(\Delta_{f_{i_x}})}{\lg k} \right\rceil, \left\lceil \frac{lg(\Delta_{f_{i_{\bar{x}}}})}{\lg k} \right\rceil\right)$$

   Jeżeli istnieje kilka zmiennych, dla których $\xi_{fi} = \min$, to spośród nich wybierana jest ta zmienna, dla której $\delta_{f_i} = \delta_{f_{i_x}} + \delta_{f_{i_{\bar{x}}}} = \left(\left\lceil \frac{\Delta_{f_{i_x}} - k}{k-1} \right\rceil + 1\right) + \left(\left\lceil \frac{\Delta_{f_{i_{\bar{x}}}} - k}{k-1} \right\rceil + 1\right)$.
5. Podział zbioru wektorów wejściowych A_{fi}, dla których funkcja przyjmuje wartości 1 na dwa podzbiory $\mathbf{A}_{fi_x}$ oraz $\mathbf{A}_{fi_{\bar{x}}}$.
6. Jeżeli uzyskana liczba warstw jest mniejsza lub równa od założonej (parametr wejściowy) wartości p $(\xi_{f_i} \leqslant p)$, to realizacja implikantów w blokach logicznych typu PAL funkcji opisanych przez zbiory wektorów powstających po podziale tzn. $\mathbf{A}_{fi_x}$ oraz $\mathbf{A}_{fi_{\bar{x}}}$. W przypadku konieczności wprowadzania sprzężeń zwrotnych wykorzystywana jest strategia klasyczna z minimalizacją liczby warstw.
7. Jeżeli uzyskana liczba warstw jest większa od założonej wartości p $(\xi_{f_i} > p)$, to poszukiwane są zmienne podziałów dla uzyskanych podzbiorów $\mathbf{A}_{fi_x}$ oraz $\mathbf{A}_{fi_{\bar{x}}}$ zgodnie z zasadami zawartymi w punkcie 4. Zmienne podziału poszukiwane są dla odpowiednich (wymagających podziału) podzbiorów, aż do spełnienia warunku punktu 6 tzn. $\xi_{f_i} \leqslant p$.

4. WYNIKI EKSPERYMENTÓW

Zaprezentowana w artykule metoda syntezy jest szczególnie atrakcyjna ze względu na uzyskiwaną liczbę warstw. Prowadząc eksperymenty zwrócono również uwagę na liczbę wykorzystywanych bloków logicznych.

W pierwszej kolejności porównano zaproponowaną p-warstwową realizację funkcji na blokach typu PAL z trójstanowymi buforami wyjściowymi o zadanej liczbie iloczynów (metoda B_PW) z klasyczną realizacją funkcji z minimalizacją liczby warstw (metoda K_LW). Po to, aby porównanie było jak najbardziej obiektywne dwie porównywane metody syntezy zostały zaimplementowane w systemie PALDec.

Tabela 1

Wynik syntezy wykorzystującej bloki logiczne typu PAL z trójstanowymi buforami wyjściowymi K_LW — metoda klasyczna z minimalizacją liczby warstw; B_PW — zaproponowana metoda wykorzystująca trójstanowe bufory wyjściowe; B — liczba bloków; W — liczba warstw, p — liczba warstw — parametr wejściowy w metodzie B_PW

Results of synthesis making use of PAL-based logic block with three-state output buffers K_LW — classical method with level minimization; B_PW — proposed approach using three-state output buffers; B — the number of logic blocks; W — the number of levels; p — the number of level — input parameter of B_PW method

	K_LW												B_PW																							
	k=3		k=4		k=5		k=6		k=7		k=8		p = 1						p = 2						p = 3						p = 4				p = 5	
	B	W	B	W	B	W	B	W	B	W	B	W	k=	k=	k=	k=	k=	k=	k=	k=	k=	k=	k=	k=	k=	k=	k=	k=	k=	k=	k=	k=	k=	k=	k=	k=
													3	4	5	6	7	8	3	4	5	6	7	8	3	4	5	6	7	8	3	4	5	6	3	4
5xp1	35	3	26	3	22	2	18	2	16	2	15	2	32	25	20	18	17	15	33	26	22	18	16	15	35	26										
9sym	43	5	29	4	22	3	18	3	15	3	13	3	x	101	83	66	56	41	112	62	43	27	25	16	79	38	22	18	15	13	56	29			43	
9symml	42	5	28	4	21	3	17	3	14	3	12	3	x	103	79	57	48	38	109	65	42	31	24	16	82	38	21	17	14	12	56	28			42	
b12	24	3	16	2	14	2	13	2	10	2	10	2	30	21	16	13	10	10	24	16	14	13	10	10												
bw	49	2	38	2	30	2	28	1	28	1	28	1	48	38	30	28	28	28	49	38	30															
clip_pla	73	4	49	3	38	3	30	3	26	2	22	2	x	67	56	44	34	31	77	49	37	30	26	22	72	49	38	30			73					
con1	4	2	3	2	2	1	2	1	2	1	2	1	4	3	2	2	2	2	4	3																
Duke2	99	3	72	3	61	2	49	2	44	2	41	2	x	x	x	x	51	50	x	72	61	49	44	41	99	72										
ex1010	242	4	165	3	124	3	101	3	85	3	74	2	x	467	358	290	235	202	511	312	212	153	118	74	390	165	124	101	85		242					
f51m	37	3	27	3	22	2	18	2	16	2	15	2	34	26	22	17	16	15	35	17	22	18	16	15	37	27										
Inc	23	3	17	2	16	2	13	2	11	2	11	2	24	18	16	14	11	11	24	17	15	13	11	11	23											
Ldd	34	2	25	2	25	2	21	2	19	1	19	1	38	29	25	21	19	19	34	25	25	21														
Misex1	14	2	12	2	8	2	7	1	7	1	7	1	13	12	8	7	7	7	14	12	8															
Misex2	19	2	19	2	18	1	18	1	18	1	18	1	19	19	18	18	18	18	19	19																
Pcle	10	2	9	1	9	1	9	1	9	1	9	1	10	9	9	9	9	9	10																	
rd53	15	3	11	2	8	2	6	2	6	2	6	2	18	10	9	7	7	5	16	11	8	6	6	6	15											
rd73	70	4	47	3	36	3	29	3	24	3	20	2	x	x	x	37	28	76	49	37	30	25	20	71	47	36	29	24	70							
rd84	142	5	95	4	72	4	58	3	49	3	42	3	x	x	96	81	79	58	162	103	78	63	52	42	150	97	72	58	49	42	142	95	72		142	
Root	35	4	24	3	18	3	16	2	13	2	11	2	38	26	22	19	15	13	36	24	18	16	13	11	35	24	18				35					
Sao2	36	3	24	3	19	2	15	2	14	2	11	2	x	x	x	47	30	25	61	29	19	15	14	11	36	24										
Sqn	21	3	15	3	11	2	10	2	8	2	7	2	25	17	13	12	10	8	22	15	11	10	8	7	21	15										
Sqr6	31	3	24	2	20	2	16	2	16	2	15	2	33	25	20	17	16	15	30	24	20	16	16	15	31											
Squar5	14	2	11	2	9	2	9	2	9	2	8	1	13	10	9	9	9	8	14	11	9	9	9													
Table3	264	4	181	4	135	3	110	3	94	3	80	3	x	x	140	116	98	85	247	179	135	111	94	81	262	181	135	110	94	80	264	181				
vg2	53	4	37	3	28	3	22	3	20	2	19	2	x	67	x	36	30	x	70	46	30	24	20	19	56	37	28	22			53					
x2	13	2	10	2	10	2	7	1	7	1	7	1	14	11	10	7	7	7	13	10	10															
Xor5_pla	8	2	5	2	4	2	3	2	3	2	3	2	8	4	4	4	4	2	8	5	4	3	3	3	8											
z4ml	29	4	19	3	15	3	13	2	11	2	9	2	27	18	17	15	10	9	27	20	16	13	11	9	29	19	15				29					
z5xp1	37	3	27	3	23	2	19	2	17	2	15	2	34	27	22	19	16	16	35	27	23	19	17	15	37	27										
z9sym	42	5	28	4	21	3	17	3	14	3	12	3	x	107	82	62	47	42	118	67	42	28	24	16	83	38	21	17	14	12	56	28			42	

W pierwszej części tabeli 1 przedstawione są wyniki syntezy układów testowych (ang. benchmark) [22] metodą klasyczną z minimalizacją liczby warstw (metoda K_LW).

W kolumnach, w których występuje litera „B" podano liczbę wykorzystywanych bloków logicznych zawierających k-iloczynów, natomiast w kolumnach z literą „W" — liczbę warstw wynikowego układu. W kolejnych częściach tabeli zawarte są wyniki syntezy zaproponowaną p-warstwową ($p = 1, \ldots, 5$) realizacją funkcji wykorzystującą wyjściowe bufory trójstanowe. Ze względu na założoną (parametr wejściowy) liczbę warstw, w tej części tabeli zawarte są tylko liczby wykorzystywanych bloków logicznych zawierających k-iloczynów. Symbole „x" oznaczają brak rozwiązania. Konstrukcja algorytmu syntezy wykorzystywanego w metodzie B_PW, zapewnia znalezienie rozwiązania o nie większej liczbie warstw niż liczba warstw uzyskiwana metodą klasyczną z minimalizacją liczby warstw (K_LW). Puste kratki odpowiadają sytuacjom, w których w pierwszym kroku algorytmu uzyskiwano mniejszą liczbę warstw od zadanej.

Analizując uzyskane wyniki można zauważyć, że w większości przypadków (85%) metodą B_PW można znaleźć rozwiązanie, wykorzystujące mniejszą liczbę warstw niż metodą klasyczną z minimalizacją liczby warstw (K_LW). Przykładowo układ testowy *rd84* metodą K_LW można zrealizować wykorzystując 142 bloki logiczne zawierające 3-iloczyny w postaci struktury 5-warstwowej. Metoda B_PW umożliwia realizację tego samego układu testowego w postaci struktury 2-warstwowej, używając jednak 162 bloki logiczne.

Istnieją przypadki (8%), w których wraz z oczywistym zmniejszeniem liczby warstw w stosunku do metody klasycznej, dochodzi również do minimalizacji liczby wykorzystanych bloków logicznych. Przykładowo tak dzieje się dla układu testowego *5xp1*. Metodą K_LW można zrealizować ten układ testowy wykorzystując 35 bloków logicznych zawierających 3-iloczyny w postaci struktury 3-warstwowej. Metoda B_PW umożliwia realizację tego samego układu testowego w postaci struktury jednowarstwowej, używając tylko 32 bloki logiczne. Dlaczego tak się dzieje? Rozważmy prosty przykład. Załóżmy, że chcemy zrealizować funkcję będącą sumą 6 implikantów. Metoda klasyczna z minimalizacją liczby warstw pozwala na realizację dwuwarstwową, wykorzystującą trzy bloki logiczne typu PAL zawierające 3 iloczyny. Jeżeli istnieje zmienna podziału implikująca podział zbioru implikantów na dwa podzbiory zawierające po trzy elementy, to możliwa jest realizacja jednowarstwowa, wykorzystująca tylko 2 bloki logiczne typu PAL (patrz przykład 5). Powyższy prosty przykład pokazuje istotę minimalizacji liczby wykorzystywanych bloków w metodzie B_PW oraz wskazuje na warunki jakie powinna spełniać funkcja, aby jej realizacja metodą B_PW doprowadziła do minimalizacji liczby bloków logicznych w stosunku do metody K_LW. Warunki te spełnione są w kilku przypadkach, dla układów testowych takich jak: *5xp1, bw, clip, f51m, squar5, xor5 i z5xp1*.

Niestety, dla założonej wartości liczby warstw, metoda B_PW nie zawsze znajduje rozwiązanie. W przypadku realizacji jednowarstwowej metoda B_PW nie doprowadziła do rozwiązania w 23 przypadkach (9,5%). Najczęściej nie znajdywano

rozwiązań (37% przypadków) dla małych bloków logicznych typu PAL ($k = 3$). Nie jest to jednak wielkim problemem ponieważ ostateczna metoda syntezy może być skomponowana w taki sposób, że w przypadku braku rozwiązania w postaci układu i-warstwowego można rozpocząć poszukiwania struktury $(i + 1)$-warstwowej. Tak skomponowana strategia syntezy i tak w większości przypadków (85%) prowadzi do rozwiązania wykorzystującego mniejszą liczbę warstw niż strategia klasyczna. W tabeli 2 przedstawiono porównanie klasycznej metody syntezy z minimalizacją liczby warstw (metoda K_LW) z realizacją wykorzystującą wyjściowe bufory trójstanowe (metoda B_PW), którą znaleziono rozwiązanie zawierające możliwie minimalną liczbę warstw, tzn. w przypadku braku rozwiązania w postaci układu i-warstwowego poszukiwano struktury $(i + 1)$-warstwowej itd. Jak widać w tabeli 2 tak skomponowana metoda syntezy zawsze doprowadziła do rozwiązania.

Analizując sumaryczną liczbę bloków oraz sumaryczną liczbę warstw można zauważyć, że znaczne ograniczenie sumarycznej liczby warstw występujące w metodzie B_PW pociąga za sobą procentowo mniejsze zwiększenie sumarycznej liczby wykorzystywanych bloków logicznych (np. $k = 7$ zmniejszenie sumarycznej liczby warstw o 100% oraz zwiększenie sumarycznej liczby bloków o 56%). Podobnie wygląda sytuacja w dla innych wartości k, przy czym większe korzyści uzyskuje się dla mniejszych bloków logicznych.

Podsumowaniem powyższych rozważań mogą być wykresy uwypuklające mocną i słabą stronę zaproponowanej strategii syntezy wykorzystującej wyjściowe bufory trójstanowe. Wartości przedstawione na wykresie 1 wyznaczone zostały z zależności $\frac{\sum bloków_{(B\_PW)} - \sum bloków_{(K\_LW)}}{\sum bloków_{(K\_LW)}}$ 100%, poprzez wstawienie za wartości $\sum bloków_{(K\_LW)}$ i $\sum bloków_{(B\_PW)}$ sumarycznej liczby bloków, wykorzystywanych odpowiednimi metodami syntezy. Liczby te zawarte są w najniższym wierszu tabeli 2. Podobnie wyznaczone zostały wartości przedstawione na wykresie 2 na podstawie zależności $\frac{\sum warstw_{(K\_LW)} - \sum warstw_{(B\_PW)}}{\sum warstw_{(B\_PW)}}$ 100%, gdzie $\sum warstw_{(K\_LW)}$ i $\sum warstw_{(B\_PW)}$ oznaczają sumaryczne liczby warstw logicznych, dla poszczególnych wartości k.

Z powyższych rozważań wynika, że wykorzystując wyjściowe bufory trójstanowe można uzyskiwać układy pracujące około dwa razy szybciej niż układy, uzyskiwane metodą klasyczną. Niestety okupione jest to wykorzystaniem większej (około 50%) liczby bloków logicznych typu PAL. Średni czas syntezy analizowanych układów testowych metodą B_PW jest około pięć razy większy od czasu syntezy metodą klasyczną z minimalizacją liczby warstw logicznych (metoda K_LW). Najwięcej czasu zajmuje procedura rozłączania implikantów. Maksymalny, sumaryczny czas syntezy, bez uwzględnienia czasu minimalizacji (Espresso-Dso) wynosi dla układu testowego *ex1010* około 9 s (Celeron 433MHz).

Tabela 2

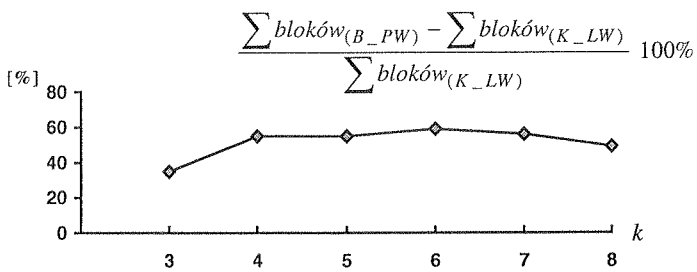
Wynik syntezy wykorzystującej bloki logiczne typu PAL z trójstanowymi buforami wyjściowymi; K_LW — metoda klasyczna z minimalizacją liczby warstw; B_PW — zaproponowana metoda wykorzystująca trójstanowe bufory wyjściowe — minimalna liczba warstw; B — liczba bloków ; W — liczba warstw

Results of synthesis making use of PAL-based logic block with three-state output buffers K_LW — classical method with level minimization; B_PW — proposed approach using three-state output buffers — minimal number of levels; B — the number of logic blocks; W — the number of levels

			K_LW												B_PW											
			k=3		k=4		k=5		k=6		k=7		k=8		k=3		k=4		k=5		k=6		k=7		k=8	
Przykład	*i*	*o*	B	W	B	W	B	W	B	W	B	W	B	W	B	W	B	W	B	W	B	W	B	W	B	W
5xp1	7	10	35	3	26	3	22	2	18	2	16	2	15	2	32	1	25	1	20	1	18	1	17	1	15	1
9sym	9	1	43	5	29	4	22	3	18	3	15	3	13	3	112	2	101	1	83	1	66	1	56	1	41	1
9symml	9	1	42	5	28	4	21	3	17	3	14	3	12	3	109	2	103	1	79	1	57	1	48	1	38	1
b12	15	9	24	3	16	2	14	2	13	2	10	2	10	2	30	1	21	1	16	1	13	1	10	1	10	1
bw	5	28	49	2	38	2	30	2	28	1	28	1	28	1	48	1	38	1	30	1	28	1	28	1	28	1
clip.pla	9	5	73	4	49	3	38	3	30	3	26	2	22	2	77	2	67	1	56	1	44	1	34	1	31	1
con1	7	2	4	2	3	2	2	1	2	1	2	1	2	1	4	1	3	1	2	1	2	1	2	1	2	1
duke2	22	29	99	3	72	3	61	2	49	2	44	2	41	2	99	3	72	2	61	2	49	2	51	1	50	1
ex1010	10	10	242	4	165	3	124	3	101	3	85	3	74	2	511	2	467	1	358	1	290	1	235	1	202	1
f51m	8	8	37	3	27	3	22	2	18	2	16	2	15	2	34	1	26	1	22	1	17	1	16	1	15	1
inc	7	9	23	3	17	2	16	2	13	2	11	2	11	2	24	1	18	1	16	1	14	1	11	1	11	1
ldd	9	19	34	2	25	2	25	2	21	2	19	1	19	1	38	1	29	1	25	1	21	1	19	1	19	1
misex1	8	7	14	2	12	2	8	2	7	1	7	1	7	1	13	1	12	1	8	1	7	1	7	1	7	1
misex2	25	18	19	2	19	2	18	1	18	1	18	1	18	1	19	1	19	1	18	1	18	1	18	1	18	1
pcle	19	9	10	2	9	1	9	1	9	1	9	1	9	1	10	1	9	1	9	1	9	1	9	1	9	1
rd53	5	3	15	3	11	2	8	2	6	2	6	2	6	2	18	1	10	1	9	1	7	1	7	1	5	1
rd73	7	3	70	4	47	3	36	3	29	3	24	3	20	2	76	2	49	2	37	2	30	2	37	1	28	1
rd84	8	4	142	5	95	4	72	4	58	3	49	3	42	3	162	2	103	2	96	1	81	1	79	1	58	1
root	8	5	35	4	24	3	18	3	16	2	13	2	11	2	38	1	26	1	22	1	19	1	15	1	13	1
sao2	10	4	36	3	24	3	19	2	15	2	14	2	11	2	61	2	29	2	19	2	47	1	30	1	25	1
sqn	7	3	21	3	15	3	11	2	10	2	8	2	7	2	25	1	17	1	13	1	12	1	10	1	8	1
sqr6	6	12	31	3	24	2	20	2	16	2	16	2	15	2	33	1	25	1	20	1	17	1	16	1	15	1
squar5	5	8	14	2	11	2	9	2	9	2	9	2	8	1	13	1	10	1	9	1	9	1	9	1	8	1
Table3	14	14	264	4	181	4	135	3	110	3	94	3	80	3	247	2	179	2	140	1	116	1	98	1	85	1
Vg2	25	8	53	4	37	3	28	3	22	3	20	2	19	2	70	2	67	1	30	2	36	1	30	1	19	2
x2	10	7	13	2	10	2	10	2	7	1	7	1	7	1	14	1	11	1	10	1	7	1	7	1	7	1
Xor5.pla	5	1	8	2	5	2	4	2	3	2	3	2	3	2	8	1	4	1	4	1	4	1	4	1	2	1
z4ml	7	4	29	4	19	3	15	3	13	2	11	2	9	2	27	1	18	1	17	1	15	1	10	1	9	1
z5xp1	7	10	37	3	27	3	23	2	19	2	17	2	15	2	34	1	27	1	22	1	19	1	16	1	16	1
z9sym	9	1	42	5	28	4	21	3	17	3	14	3	12	3	118	2	107	1	82	1	62	1	47	1	42	1
		Σ	1558	96	1093	81	861	69	712	63	625	60	561	57	2104	42	1692	35	1333	34	1134	32	976	30	836	31

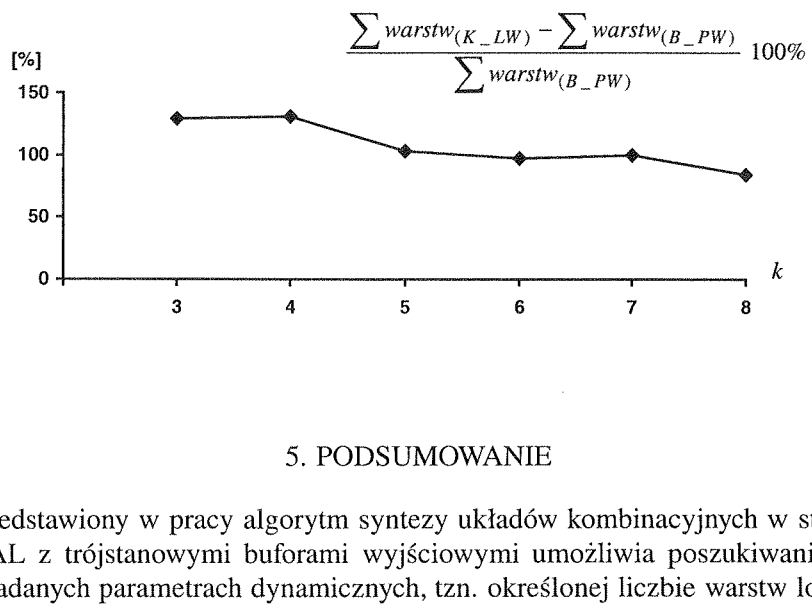
Wykres 1. Porównanie metody syntezy B_PW z metodą K_LW pod kątem liczby bloków (opis w tekście)

Chart 1. Comparison of B_PW synthesis method with K_LW method from the point of logic blocks (description in the text)



Wykres 2. Porównanie metody syntezy B_PW z metodą K_LW pod kątem liczby warstw logicznych (opis w tekście)

Chart 2. Comparison of B_PW synthesis method with K_LW method from the point of logic levels (description in the text)



5. PODSUMOWANIE

Przedstawiony w pracy algorytm syntezy układów kombinacyjnych w strukturach typu PAL z trójstanowymi buforami wyjściowymi umożliwia poszukiwanie rozwiązań o zadanych parametrach dynamicznych, tzn. określonej liczbie warstw logicznych. Istota zaproponowanej metody syntezy tkwi w nietypowym wykorzystaniu buforów trójstanowych. Umożliwia ona znaczne ograniczenie czasu propagacji sygnałów od wejść do wyjść w stosunku do innych, znanych autorowi, narzędzi komercyjnych jak i akademickich.

6. BIBLIOGRAFIA

1. P. Abouzeid, B. Babba, M. Crastes, G. Saucier: *Input-Driven Partitioning Methods and Application to Synthesis on Table-Lookup-based FPGAs.* IEEE Trans on CAD, 12, No. 7, 1993, pp. 913-925
2. AMD, Altera, Atmel, Lattice, Xilinx Data Book
3. B. Babba, M. Crastes, G. Saucier: *Input driven synthesis on PLDs and PGAs.* The European Conference on Design Automation, ECDA'92, March 1992, Brussels, Belgium, pp. 48-52
4. M. Bolton: *Digital Systems Design with Programmable Logic.* Addison-Wesley Publishing Company, 1990, p.133-140
5. R.K. Brayton, G.D. Hachtel, A.L. Sangiovanni-Vincentelli: *Multilevel logic synthesis.* Proceedings of the IEEE, 78(2), February 1990, pp. 264-300
6. S.D. Brown, R.J. Francis, J. Rose, Z.G. Vranesic: *Field Programmable Gate Arrays.* Boston, Kluwer Academic Publishers, 1993, pp. 45-86
7. M. J. Ciesielski, S. Yang, PLADE: *A two-stage PLA decomposition.* IEEE Transactions on Computer-Aided Design, 11(8), August 1992, pp. 943-954
8. L. Jóźwiak: *General decomposition and its use in digital circuit synthesis.* VLSI Design, 3 (3-4), 1995, pp. 225-248
9. D. Kania: *Two-level logic synthesis on PALs.* Electronics Letters, 1999, Vol. 35, No. 11, pp. 879-880
10. D. Kania: *Decomposition-based synthesis and its application in PAL-oriented technology mapping.* Proceedings of 26-th Euromicro Conference, IEEE Computer Society Press, Maastricht, 2000, pp. 138-145
11. D. Kania: *Synteza logiczna dla układów CPLD typu PAL wykorzystująca dekompozycję.* Kwartalnik Elektroniki i Telekomunikacji, 1999,45, z. 3-4, ss. 445-454
12. D. Kania, Synteza logiczna wielopoziomowych układów w strukturach typu PAL z trójstanowymi buforami wyjściowymi. Kwartalnik Elektroniki i Telekomunikacji, 2000, 46, z.1, ss. 81-90
13. D. Kania: *Synteza logiczna dla struktur typu PAL wykorzystująca bufory wyjściowe.* Kwartalnik Elektroniki i Telekomunikacji, 2002, 48, z.1, ss. 53-66
14. D. Kania: *Realizacja układów kombinacyjnych w strukturach MACH.* Kwartalnik Elektroniki i Telekomunikacji, 2001, 47, z. 1, ss. 65-74
15. R. Murgai, Y. Nishizaki, N.Shenay, R.K. Brayton, A.Sangiovanni-Vincentelli: *Logic Synthesis for Programmable Gate Array.* Proc. 27th DAC, June 1990, pp. 620-625
16. T. Łuba: *Multi-level logic synthesis based on decomposition.* Microprocessors and Microsystems, Vol. 18, No. 8, Octomber 1994, pp. 429-437
17. G. de Micheli: *Synthesis and optimization of digital circuits.* McGraw-Hill, 1994
18. P. Michel, U. Lauther, P. Duzy: *The Synthesis Approach to Digital System Design.* Boston, Kluwer Academic Publishers, 1993
19. G. Saucier, P. Sicard, L. Bouchet: *Multi-level synthesis on PAL's.* Proc. European Design Automation Conference, Glasgow, March 1990, pp. 542-546
20. T. Sasao: *FPGA Design by Generalized Functional Decomposition in Logic Synthesis and Optimization.* Boston, Kluwer Academic Publishers, 1993
21. K. Sharma: *Programmable Logic Handbook, PLDs, CPLDs,& FPGAs.* McGraw-Hill, 1998
22. www.cbl.ncsu/edu Collaborative Benchmarking Laboratory, Department of Computer Science at North Carolina State University

Pracę wykonano w ramach projektu badawczego KBN nr rej. 4T 11B 06022 pt.: „Synteza logiczna układów cyfrowych dedykowana dla macierzowych struktur programowalnych"

D. KANIA

A P-STAGE LOGIC SYNTHESIS FOR PAL-BASED DEVICES

Summary

The PAL-based logic block constitutes the kernel of many Programmable Logic Devices. A typical CPLD architecture often includes logic block, which are similar to a simple two-level PAL. Each PAL-based logic block contains a programmable AND-array that feeds its macrocells. These structures have exactly defined number of terms connected to the individual output. This feature of a PAL-based block affects significantly the synthesis process of digital circuits based on such devices.

The problem of partition of the whole devices under design into suitable parts, which can be implemented as single PAL-based logic blocks, containing the limited number of terms, is one of basic problems of the synthesis. The logic blocks included into CPLD structures contain usually additional logic resources that facilitate the partitioning process. A typical CPLDs usually contain three-state output buffers.

A new method of *p*-stage logic synthesis on PAL-based devices with three-state output buffers is presented. This method leads to the implementation of digital circuits in the form of *p*-stage structures.

Developed algorithms, implemented within the PALDec system, have been used for partitioning the benchmark circuits. The obtained results demonstrate the superiority of the presented synthesis methods compared to the classical approach.

Keywords: logic synthesis, partitioning, minimization, PLD

KWARTALNIK ELEKTRONIKI I TELEKOMUNIKACJI, 2004, 50, z. 1, ss. 87–97

Czy istnieją algorytmy równoległe o superliniowym przyspieszeniu obliczeń?

MIROSŁAW GAJER

Katedra Automatyki AGH
Al. Mickiewicza 30
30-059 Kraków
mgajer@ia.agh.edu.pl

Otrzymano 2003.01.07
Autoryzowano 2003.06.04

Rozwiązania wieloprocesorowe stają się coraz bardziej popularne, zwłaszcza w zastosowaniach związanych z systemami czasu rzeczywistego, gdzie istnieje duże zapotrzebowanie na moc obliczeniową, a czas realizacji wielu zadań stanowi parametr krytyczny. Ewentualny zysk, jaki może przynieść zastosowanie systemu wieloprocesorowego jest zdeterminowany tzw. prawem Amdahla, które podaje górną granicę wartości współczynnika przyśpieszenia obliczeń w zależności od liczby zastosowanych procesorów. Motywacją do napisania niniejszego artykułu było zapoznanie się autora z hipotezą istnienia algorytmów o przyśpieszeniu superliniowym, które stanowiłyby odstępstwo od prawa Amdahla. Niestety po wnikliwym zbadaniu hipoteza ta okazała się całkowicie błędna, a w artykule zamieszczono dowód twierdzenia pokazującego, że algorytmy o superliniowym przyśpieszeniu obliczeń nie istnieją. z zamieszczonych w artykule rozważań wynika w sposób jednoznaczny, że nie jest rzeczą możliwą uzyskanie efektu synergii w przypadku systemów wieloprocesorowych. Zatem dysponując np. systemem czteroprocesorowym dany algorytm można realizować co najwyżej cztery razy szybciej niż w przypadku systemu jednoprocesorowego, ale jest zarazem maksymalna wartość możliwego do uzyskani przyśpieszenia obliczeń.

Słowa kluczowe: Obliczenia równoległe, Prawo Amdahla, Współczynnik przyśpieszenia obliczeń

1. WPROWADZENIE

Obecnie coraz bardziej popularne stają się rozwiązania wieloprocesorowe. Uwaga ta dotyczy zwłaszcza komputerowych systemów czasu rzeczywistego, w których na realizację zadań obliczeniowych nałożone są ograniczenia czasowe [5]. Koniecz-

ność terminowego zakończenia zadań, przed upływem ich ograniczeń czasowych, powoduje z kolei powstanie dużego zapotrzebowania na moc obliczeniową zastosowanych procesorów [1]. W pewnych wypadkach, np. w przypadku pracujących w czasie rzeczywistym komputerowych systemów wizyjnych, zastosowanie nawet najszybszych procesorów nie pozwala na dochowanie ograniczeń czasowych [4]. W takiej sytuacji jedynym rozwiązaniem jest zastosowanie w miejsce pojedynczego procesora systemu wieloprocesorowego i dokonanie podziału zadania na zbiór podzadań, które mogą być wykonywane równolegle [6].

Miernikiem jakości pracy systemu wieloprocesorowego, podczas realizacji pewnego zadania obliczeniowego τ, jest tzw. współczynnik przyspieszenia obliczeń S zdefiniowany w sposób następujący:

$$S = \frac{T_S}{T_P} \tag{1}$$

We wzorze (1) T_S oznacza czas wykonania zadania τ przez system jednoprocesorowy, przy czym powinna być to najszybsza z możliwych realizacja sekwencyjna zadania τ, która jednakże często nie jest znana i musi w związku z tym podlegać w praktyce aproksymacji. z kolei T_P oznacza czas realizacji zadania τ przez badany system wieloprocesorowy.

Zatem współczynnik przyspieszenia obliczeń S mówi, ile razy szybciej zadanie τ zostanie policzone przez system wieloprocesorowy, w porównaniu z realizacją tego zadania na najszybszym z procesorów sekwencyjnych. Należy zauważyć, że wartość współczynnika przyspieszenia obliczeń S powinna być zawsze większa od jedności

$$S > 1 \tag{2}$$

gdyż w przypadku przeciwnym zastosowanie systemu wieloprocesorowego nie miałoby sensu — po prostu, rozważane zadanie mogłoby być wykonane szybciej przez system jednoprocesorowy.

W związku z powyższym powstaje ważne pytanie, czy wartość współczynnika przyśpieszenia obliczeń S jest również w jakiś sposób ograniczona od góry, czy też współczynnik ten może przyjmować dowolnie duże wartości.

2. PRAWO AMDAHLA A PRZYŚPIESZENIE SUPERLINIOWE

Odpowiedź na tak postawione pytanie przynosi tzw. prawo Amdahla [2], które pozwala wyliczyć wartość współczynnika przyśpieszenia obliczeń S w funkcji liczby procesorów n zastosowanego systemu wieloprocesorowego, jako

$$S(n) = \frac{n}{1 + (n-1)f} \tag{3}$$

We wzorze (3) f oznacza frakcję obliczeń, która nie może zostać wykonana w sposób równoległy. Wartość współczynnika f mieści się w zakresie od zera do jedności

$$f \in [0, 1] \tag{4}$$

Zerowa wartość współczynnika f oznacza, że zadanie obliczeniowe może zostać w całości wykonane w sposób równoległy. Zatem

$$S(n) = n, \qquad \text{gdy} \qquad f = 0 \tag{5}$$

Z kolei, gdy współczynnik f przyjmuje wartość równą jedności oznacza to, że zadanie obliczeniowe w ogóle nie może zostać wykonane w sposób równoległy i w związku z tym musi być realizowane na pojedynczym procesorze. Zatem

$$S(n) = 1, \qquad \text{gdy} \qquad f = 1 \tag{6}$$

Dla wszystkich pozostałych wartości współczynnika f, należących do przedziału $f \in (0, 1)$ współczynnik przyspieszenia obliczeń przyjmuje wartości z przedziału

$$S(n) \in (1, n) \tag{7}$$

Jak wynika z przeprowadzonych rozważań, zgodnie z prawem Amdahla wartość współczynnika przyspieszenia obliczeń dla n procesorów jest ograniczona od góry przez liczbę procesorów n. Zatem stosując system równoległy zbudowany na przykład z ośmiu procesorów dowolne zadanie obliczeniowe τ będzie wykonywane co najwyżej osiem razy szybciej.

Należy zauważyć, że prawidłowość ta nie ma bezwzględnego zastosowania we wszelkich dziedzinach techniki. Często bowiem daje się zaobserwować efekty synergii, polegające na tym, że grupa czynników wywołuje znacznie silniejszy efekt działania niż wynikałoby to z ich indywidualnych zdolności i możliwości. Ostatnio modnym terminem w dziedzinie informatyki zajmującej się systemami sztucznej inteligencji stała się tzw. inteligencja grupowa CI (ang. Collective Intelligence) [2]. Okazuje się bowiem, że zespół nawet stosunkowo prostych automatów może dzięki ich wzajemnej komunikacji tworzyć skomplikowane struktury społeczne zdolne do realizacji zadań, które nie byłyby możliwe do zrealizowania przez żadnego z członków tej zbiorowości działającego w pojedynkę. Przykładem występujących w sposób naturalny w przyrodzie systemów, w których szczególnie mocno zamanifestowała się inteligencja grupowa jest kolonia mrówek, pszczół, termitów i innych owadów wykazujących zachowania społeczne. Należy zauważyć, że żaden z tych owadów nie jest w stanie przeżyć samodzielnie i dopiero wytworzenie skomplikowanej struktury społecznej, wykorzystującej zjawisko synergii pozwala na przetrwanie gatunku.

W związku z powyższym nasuwa się pytanie, czy również w przypadku systemów wieloprocesorowych nie dało by się w jakiś sposób wytworzyć takiego zjawiska synergii, która pozwoliłaby na szybszą realizacje zadania obliczeniowego niż to wynika z prawa Amdahla. Zgodnie z prawem Amdahla wartość współczynnika przyspieszenia obliczeń S może wzrastać co najwyżej liniowo w funkcji liczby zastosowanych

procesorów n. Zatem wykorzystanie zjawiska synergii powinno pozwolić na szybszy (tzw. superliniowy) wzrost wartości współczynnika przyśpieszenia obliczeń w funkcji liczby procesorów, czyli powinno zachodzić

$$S(n) > n \tag{8}$$

3. POSZUKIWANIE SYNERGII W SYSTEMACH WIELOPROCESOROWYCH

Pewną wskazówkę w poszukiwaniu odpowiedzi na tak postawione pytanie podsuwa książka [3]. W książce tej na stronie 104 można przeczytać, że istnieją algorytmy, które charakteryzują się przyspieszeniem większym niż liniowe, czyli tzw. superliniowym. Autor książki [3] wyróżnia trzy możliwe przypadki pojawienia się takiego zjawiska, do których zalicza:

1. Zadania obliczeniowe, które po dekompozycji na podzadania mieszczą się w całości w pamięciach podręcznych procesorów
2. Zadania, które polegają na znalezieniu najkrótszego czasu rozwiązania problemu
3. Zadania, dla których można, w rzadkim przypadku, stworzyć algorytm równoległy dający w porównaniu z algorytmem sekwencyjnym przyspieszenie superliniowe

Jeżeli chodzi o przypadek wymieniony w punkcie pierwszym, to istotnie często jest tak, że kod programu realizującego pewne zadanie obliczeniowe jest zbyt duży, aby mógł się zmieścić w całości w pamięci podręcznej instrukcji (ang. instruction cache) procesora. Natomiast po podziale tego zadania na zbiór podzadań może się oczywiście tak zdarzyć, że kody programów realizujących rozważane podzadania będą miały mniejsze rozmiary i w związku z tym zmieszczą się w całości w pamięciach podręcznych procesorów systemu równoległego. W takim wypadku w trakcie wykonywania obliczeń nie będzie zachodziła konieczność dokonywania operacji odświeżania pamięci podręcznych, podczas której to operacji procesor wykonuje cykle jałowe w oczekiwaniu na zakończenie zapisu fragmentu kodu programu do podbloku pamięci podręcznej. W związku z powyższym może się tak w pewnym bardzo szczególnym przypadku zdarzyć, że ewentualne zyski wynikające z braku konieczności oczekiwania na odświeżenie pamięci podręcznych, przewyższą występujące w sposób nieuchronny w każdym systemie równoległym straty związane z koniecznością wzajemnej komunikacji i synchronizacji jednostek obliczeniowych, i w efekcie tego równoległa realizacja obliczeń wykaże ich superliniowe przyśpieszenie. Na przykład zastosowanie systemu ośmioprocesorowego spowoduje dziewięciokrotne przyśpieszenie obliczeń.

Z drugiej jednak strony nie jest to przykład zbytnio interesujący z naukowego i poznawczego punktu widzenia, ponieważ wszystko sprowadza się w tym wypadku do szczegółów natury wyłącznie technicznej takich, jak rozmiary pamięci podręcznych procesorów. Zastosowanie w przypadku sekwencyjnej realizacji rozważanego algorytmu jednostki obliczeniowej o większej pojemności podręcznej pamięci instrukcji spowoduje, że rozważany, w swej istocie iluzoryczny, zysk zniknie, a prawo Amdahla będzie ponownie spełnione.

Co do przypadku wymienionego w punkcie drugim, trudno jest się autorowi niniejszego artykułu wypowiedzieć, z tego prostego powodu, że nie udało mu się odgadnąć, co autor książki [3] miał tutaj na myśli.

Natomiast przypadek wymieniony w punkcie trzecim wygląda bardzo interesująco, ponieważ wynika z niego, że możliwe jest stworzenie, aczkolwiek jak autor książki [3] ostrożnie zaznaczył, w pewnych rzadkich przypadkach specjalnego algorytmu równoległego, który w porównaniu ze znanym wcześniej algorytmem sekwencyjnym dawałby superliniowe przyśpieszenie obliczeń. Niestety autor książki [3] nie podał żadnego przykładu takiego algorytmu równoległego wykazującego właściwości przyśpieszenia superliniowego, nie podał również żadnych odnośników do pozycji literaturowych, w których można było by odnaleźć jakiekolwiek informacje na ten temat. Skąd zatem wzięła się hipoteza istnienia algorytmów równoległych o superliniowym przyśpieszeniu obliczeń?

Autor niniejszego artykułu uważa taką hipotezę za całkowicie błędną i twierdzi, że algorytmy równoległe o superliniowym współczynniku przyśpieszenia obliczeń nie istnieją, co zostanie wykazane w kolejnym punkcie.

4. DOWÓD NA NIE ISTNIENIE ALGORYTMÓW O SUPERLINIOWYM PRZYSPIESZENIU OBLICZEŃ

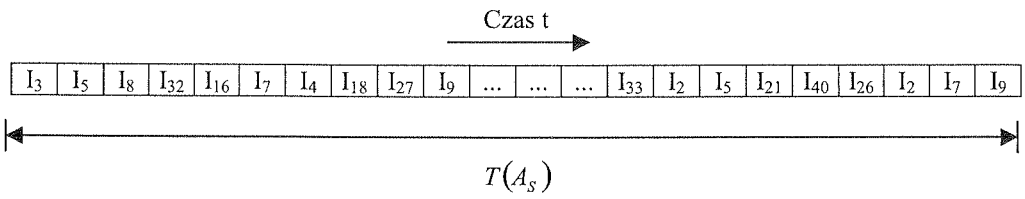
Załóżmy, że istnieje pewien sekwencyjny algorytm A_S, który stanowi najszybszy ze znanych w danej chwili algorytmów rozwiązania pewnego problemu obliczeniowego C. Niech czas wykonania tego algorytmu na najszybszym w danej chwili dostępnym procesorze sekwencyjnym P wynosi $T(A_S)$. Niech ponadto sposób realizacji tego algorytmu został zoptymalizowany na poziomie kodu instrukcji procesora P — problem C został zapisany w języku asemblera, w taki sposób aby realizacja algorytmu A_S była możliwie najszybsza.

Ponieważ procesor P jest w danej chwili najszybszym dostępnym procesorem, nie istnieje zatem żadna inna sekwencyjna realizacja algorytmu A_S o krótszym czasie realizacji. Zatem czas wykonania algorytmu A_S na sekwencyjnym procesorze P, czyli $T(A_S)$ spełnia wszelkie wymogi definicji (1). Zatem dalsze skrócenie czasu realizacji problemu C wymaga zastosowania systemu wieloprocesorowego.

Niech rozważany system wieloprocesorowy zbudowany będzie z N procesorów takiego samego typu, jak sekwencyjny procesor P. Zatem w danej chwili system wieloprocesorowy złożony z N procesorów P jest najlepszym ze wszystkich możliwych systemów wieloprocesorowych zbudowanych z N procesorów.

Sekwencyjna realizacja algorytmu A_S wymaga wykonania przez procesor P pewnej sekwencji instrukcji maszynowych procesora P. Załóżmy ponadto, że w trakcie wykonywania takiej sekwencji instrukcji pomiędzy poszczególne instrukcje nie są wstawiane żadne takty pracy jałowej procesora, związane np. z koniecznością oczekiwania na zakończenie operacji odczytu pamięci lub urządzenia I/O. Zatem procesor P przez cały czas trwania realizacji algorytmu A_S pracuje w sposób aktywny, wykonując kolejno

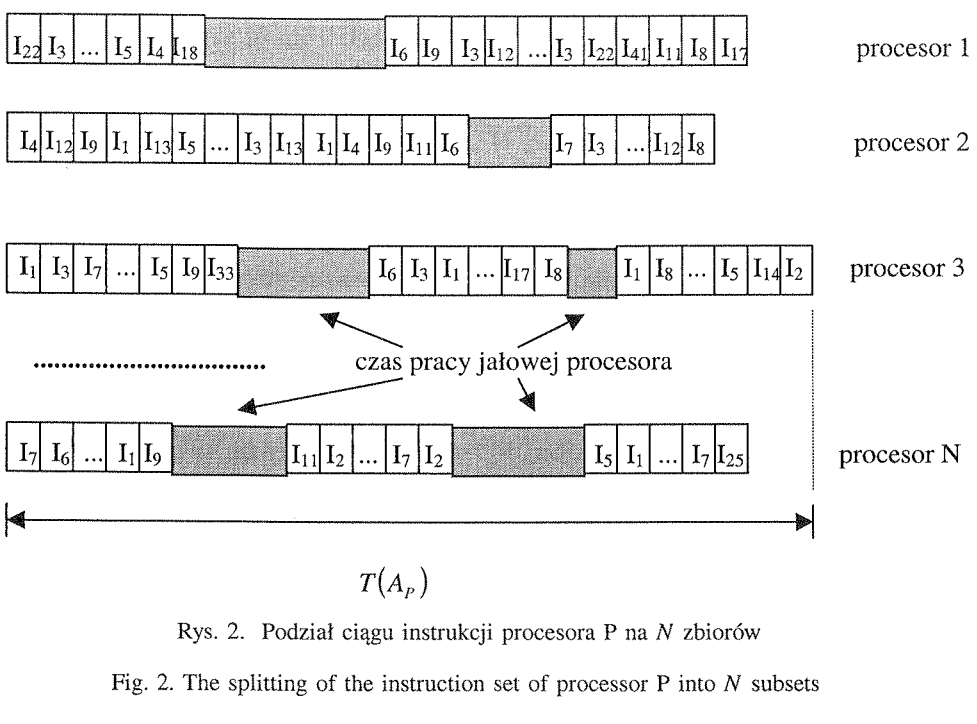
instrukcję za instrukcją. Jeżeli ponadto założymy, że instrukcje wykonywane przez procesor P należą do zbioru instrukcji $\{I_1, I_2, I_3, \ldots, I_K\}$ oraz, że każda z instrukcji posiada taki sam czas realizacji t, praca procesora P podczas realizacji algorytmu A_S może zostać zilustrowana tak, jak na rys. 1. Załóżmy ponadto, że procesor P wykonuje M instrukcji.



Rys. 1. Ilustracja sposobu pracy procesora P podczas realizacji algorytmu A_S

Fig. 1. The illustration of the way of the work of processor P during the realisation of algorithm A_S

Zaimplementowanie algorytmu A_S dla systemu wieloprocesorowego złożonego z N procesorów P wymaga dokonania podziału przedstawionego na rys. 1 ciągu instrukcji na N części, z których każda wykonywana będzie przez inny procesor. Proces ten został pokazany na rys. 2.



Rys. 2. Podział ciągu instrukcji procesora P na N zbiorów

Fig. 2. The splitting of the instruction set of processor P into N subsets

Czas wykonania algorytmu A_S w wersji równoległej, oznaczonej jako A_P, jest równy czasowi zakończenia pracy przez ostatni (pracujący najdłużej) z procesorów P (na rys. 2 jest to procesor 3) i wynosi $T(A_P)$. Zatem wartość współczynnika przyspieszenia obliczeń może zostać wyliczona ze wzoru

$$S = \frac{T(A_S)}{T(A_P)} \tag{9}$$

Gdyby liczba M instrukcji była podzielna przez liczbę N procesorów systemu wieloprocesorowego i udało się dodatkowo podzielić program sekwencyjny na N równych części o liczbie $\frac{M}{N}$ instrukcji każda, wówczas czas realizacji programu równoległego $T(A_P)$ byłby najkrótszy z możliwych i wynosił

$$T(A_P) = \frac{T(A_S)}{N} \tag{10}$$

Podstawiając zależność (10) do wzoru (9) otrzymuje się

$$S = N \tag{11}$$

Jest to górna granica wartości współczynnika przyspieszenia obliczeń dopuszczalna przez prawo Amdahla.

Niestety w praktycznych realizacjach algorytmów równoległych sprawy nie wyglądają aż tak dobrze, ponieważ:

- Nie zawsze można podzielić program na N równych części, które mogą być realizowane w sposób równoległy.
- Nie zawsze poszczególne części programu równoległego mogą być wykonywane w sposób nieprzerwany. Jak zostało pokazane na rys. 2 regułą jest raczej przerywanie ciągu wykonywanych instrukcji taktami pracy jałowej procesora oczekującego na synchronizację bądź uzyskanie danych z innych części programu realizowanych na pozostałych procesorach.

Jak widać w wyidealizowanym przypadku otrzymuje się liniową zależność wartości współczynnika przyśpieszenia obliczeń S od liczby zastosowanych procesorów N. Jak zatem można uzyskać zależność superliniową?

Trzeba w tym miejscu przypomnieć poczynione uprzednio założenie o jednakowym czasie wykonywania t wszystkich instrukcji procesora P należących do jego zbioru instrukcji $\{I_1, I_2, I_3, \ldots, I_K\}$. Sytuacja taka ma istotnie miejsce w wielu nowoczesnych procesorach o architekturze RISC, gdzie każda z instrukcji procesora wykonywana jest w jednym cyklu sygnału taktującego. W przypadku procesorów, dla których prawidłowość ta nie jest spełniona, rozkazy o dłuższym czasie wykonywania stanowiącym wielokrotność czasu t mogą zostać potraktowane jako ciąg odpowiedniej liczby instrukcji, z których każda realizowana jest w czasie t równym cyklowi sygnału taktującego.

Zatem jedynym sposobem na uzyskanie przyspieszenia superliniowego byłoby opracowanie nowego algorytmu równoległego A_P^*, który zawierałby w porównaniu

z algorytmem A_P mniejszą liczbę instrukcji potrzebnych do wykonania. Na rys. 3 przedstawiono porównanie algorytmu równoległego A_P o liniowym przyśpieszeniu obliczeń oraz hipotetycznego algorytmu A_P^* o przyspieszeniu superliniowym.

Ponieważ $T(A_P^*) < T(A_P)$, podstawiając do wzoru (9) $T(A_P^*)$ w miejsce $T(A_P)$ otrzymuje się

$$S(N) > N \tag{12}$$

Zatem wszystko wskazuje na to, że nowy algorytm równoległy A_P^* odznacza się superliniowym przyspieszeniem obliczeń. Nie można jednak zapominać tutaj o treści definicji współczynnika przyśpieszenia obliczeń, w której występuje czas realizacji algorytmu sekwencyjnego $T(A_S)$. Definicja ta mówi wyraźnie, że powinien być to czas najlepszej, znanej w danej chwili sekwencyjnej realizacji danego zadania obliczeniowego τ. Zatem zadanie τ powinno być realizowane za pomocą najszybszego (o najmniejszej złożoności obliczeniowej) znanego w danej chwili algorytmu oraz powinno być wykonywane na najszybszym (o największej mocy obliczeniowej) procesorze, jaki jest w danej chwili dostępny.

Powstaje pytanie, czy odkrycie nowego algorytmu równoległego A_P^* coś tutaj zmienia?

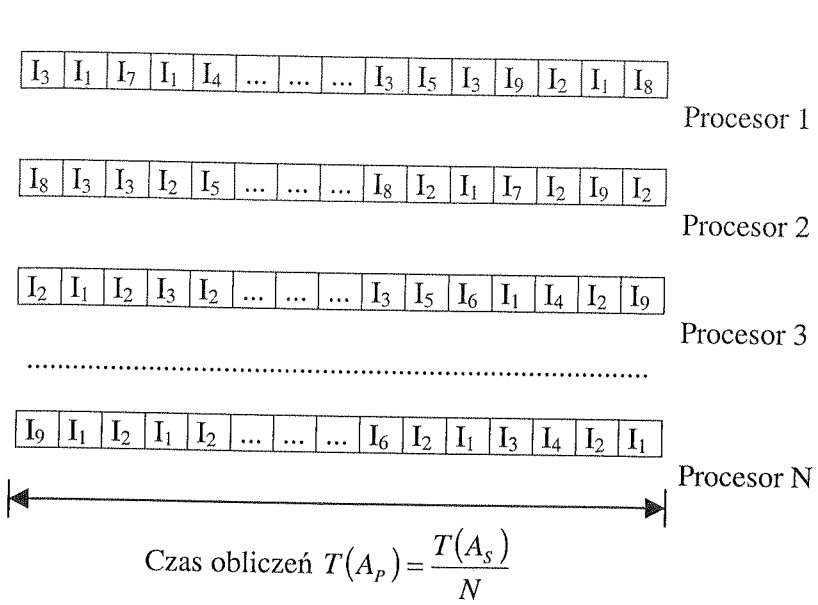
Okazuje się, że tak. Załóżmy, że realizacja sekwencyjnego algorytm A_S na procesorze P wymagała wykonania M instrukcji tego procesora. z kolei równoległa realizacja algorytmu A_P o liniowym współczynniku przyspieszenia obliczeń wymagała wykonania przez każdy z N procesorów systemu $\frac{M}{N}$ instrukcji należących do zbioru instrukcji procesora P. Ponieważ nowy algorytm równoległy wykazuje własności przyśpieszenia superliniowego całkowita liczba instrukcji wykonywanych przez procesory systemu wieloprocesorowego M^* musi być mniejsza od M, tak aby czas pracy każdego z procesorów był mniejszy od czasu realizacji $\frac{M}{N}$ instrukcji.

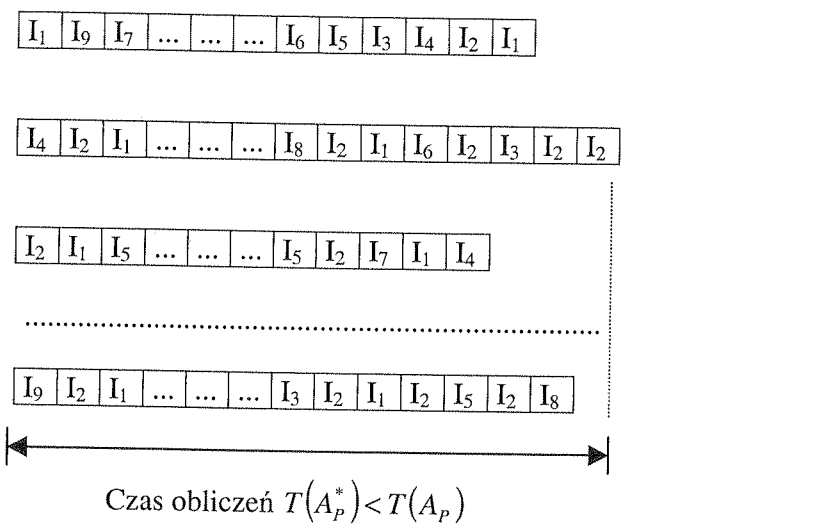
Czy w takim razie algorytm A_S może być nadal uważany za najszybszą ze znanych w danej chwili sekwencyjnych realizacji zadania obliczeniowego τ?

Niestety nie, ponieważ każdy algorytm równoległy może zostać również wykonany w sposób sekwencyjny (procesor sekwencyjny będzie po prostu wykonywał po kolei podzadania przewidziane dla procesorów systemu równoległego). Zatem sekwencyjna realizacja algorytmu A_P^* oznaczona jako A_S^* wymagała będzie wykonania takiej samej liczby M^* instrukcji procesora P. Ponieważ $M^* < M$, a wszystkie instrukcje procesora P posiadają jednakowy czas realizacji t, zatem musi w tym wypadku zachodzić

$$T\left(A_S^*\right) < T(A_S) \tag{13}$$

Co to oznacza? Otóż oznacza to, że został właśnie znaleziony nowy algorytm sekwencyjny A_S^* o mniejszej złożoności obliczeniowej, niż znany uprzednio algorytm A_S i to właśnie czas wykonania tego nowego algorytmu A_S^*, czyli TA_S^* należy podstawić do wzoru (9) zamiast czasu realizacji algorytmu A_S. Po dokonaniu takiego podstawie-





Rys. 3. Porównanie algorytmu o przyspieszeniu liniowym z algorytmem o przyspieszeniu superliniowym

Fig. 3. The comparison of algorithm of linear speedup with the algorithm of superlinear speedup

nia superliniowość algorytmu A_P^* okazuje się całkowicie iluzoryczna, ponieważ dla N procesorów zawsze będzie spełniony warunek

$$\frac{T\left(A_S^*\right)}{T\left(A_P^*\right)} \leqslant N \tag{14}$$

Wniosek końcowy nasuwa się wręcz sam — algorytmy równoległe o superliniowym współczynniku przyśpieszenia obliczeń nie istnieją.

5. PODSUMOWANIE

Bezpośrednią motywację do napisania niniejszego artykułu stanowiła napotkana w książce [3] hipoteza istnienia algorytmów równoległych wykazujących cechę superliniowości współczynnika przyspieszenia obliczeń. Istnienie takich algorytmów stanowiłoby w oczywisty sposób naruszenie znanego z literatury prawa Amdahla, które stwierdza, że wartość współczynnika przyspieszenia obliczeń może wzrastać co najwyżej liniowo wraz ze wzrostem liczby procesorów systemu równoległego.

Gruntowna analiza rozważanego problemu pozwoliła wykazać na drodze formalnej, że algorytmy równoległe o superliniowym współczynniku przyspieszenia obliczeń nie istnieją, a prawo Amdalha jest zawsze spełnione dla wszystkich algorytmów.

6. BIBLIOGRAFIA

1. J. Werewka, T. Szmuc: *Analiza i projektowanie systemów czasu rzeczywistego o różnym stopniu rozproszenia*, Kraków, Wydawnictwo D'EL ART., 2001
2. T. Szuba: *Compotational collective intelligence*, New York, John Willey & Sons, 2001
3. J. Kitowski: *Współczesne systemy komputerowe*, Kraków, Wydawnictwo CCNS, 2000
4. M. Gajer: *Przykłady zastosowań systemów wizyjnych w automatyzacji procesów przemysłowych*, IV Konferencja: Komputerowe systemy wspomagania nauki, przemysłu i transportu – TransComp, Zakopane 6-8 grudnia 2000, pp. 469-474
5. G. Motet, T. Szmuc: *Programowanie systemów czasu rzeczywistego*, Kraków, Wydawnictwo CCATIE, 2000
6. R. Negre: *EUROPRO: A new open VMEbus multiprocessor computer for signal processing and intensive data processing*, Real-Time Magazine 1/1997, pp. 16-20

M. GAJNER

DOES PARALLEL ALGORITHMS WITH SUPERLINEAR SPEEDUP COEFFICIENT EXIST?

Summary

Multiprocessor solutions are getting more and more popular especially in the applications concerned with real-time systems, which require the great amount of computational power and where the time of

tasks performance is a critical parameter. This concerns especially the real-time systems that require great computational power. Real-time systems are nowadays broadly used in telecommunication systems, industrial control systems, transport and space systems, and so on. In order to achieve greater computational power the real-time tasks are divided into a few subtasks that can be run in parallel.

Splitting an algorithm into some number of tasks that can be run in parallel is often a hard job to do for a computer programmer. It is so because this process cannot be fully automated and much of programmer's invention is necessary here, although the programmer can never know a priori how effective his parallel algorithm will be, because in most cases the communication overloads are a very important factor.

The profit that can be obtained by the usage of a multiprocessor system is determined by the Amdahl law, which gives the upper bound of the value of speedup coefficient depending on the number of processors being used. The motivation for having written this paper was the hypothesis of existence of parallel algorithms with superlinear speedup coefficient. If such algorithms existed they would in apparent way violate the Amdahl law. After the thorough examination this hypothesis appeared to be false. In the paper it was proved that parallel algorithms with superlinear speedup coefficients do not exist.

This fact also means that it is impossible to achieve a synergy effect in multiprocessor systems. This means that if one has a four-processor systems, such system can work only four times faster at the most. It is impossible that the system worked faster because if it were so, such parallel algorithm could also be realised sequentially and this sequential algorithm would be the base algorithm in that case. The execution time of the base algorithm is used in the definition of speedup factor for parallel systems.

Keywords: parallel computing, Amdahl law, speedup coefficient

KWARTALNIK ELEKTRONIKI I TELEKOMUNIKACJI, 2004, 50, z. 1, ss. 99–108

Czułość radiometru mikrofalowego z detektorem kwadratowym i liniowym

BRONISŁAW STEC, ANDRZEJ DOBROWOLSKI, WALDEMAR SUSEK

Wojskowa Akademia Techniczna, Wydział Elektroniki
ul. Kaliskiego 2, 00 – 908 Warszawa 49
ADobrowolski@wel.wat.edu.pl

Otrzymano 2002.11.26
Autoryzowano 2003.04.17

W artykule zaprezentowano analizę stochastyczną mikrofalowego radiometru modulacyjnego z kwadratowym i liniowym detektorem amplitudy. Przyjmując idealizowane charakterystyki detektorów wykazano, że w typowych zastosowaniach, tj. przy pomiarze bardzo małych mocy, rodzaj użytego detektora nie ma wpływu na wypadkową czułość radiometru. Ponadto przedstawiono inne uwarunkowania związane z zastosowaniem konkretnego typu detektora.

Słowa kluczowe: termograf mikrofalowy, radiometr, promieniowanie termiczne, czułość

1. WPROWADZENIE

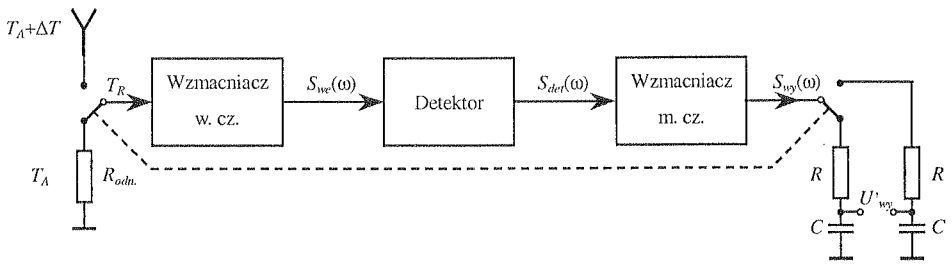
Pomiar temperatury obiektów fizycznych metodą radiometryczną polega na pomiarze mocy emitowanego przez nie promieniowania termicznego, które w zakresie mikrofalowym jest wprost proporcjonalne do temperatury badanego medium. Pomiar ten sprowadza się w istocie do pomiaru mocy sygnału szumowego pochodzącego z anteny, przy czym cechy probabilistyczne sygnału mierzonego i szumu generowanego termicznie wewnątrz radiometru są jednakowe.

W rozwiązaniach układowych radiometrów modulacyjnych wejście odbiornika przełączane jest z częstotliwością kilkuset Hz pomiędzy wejściem antenowym a źródłem sygnału odniesienia. Następuje w ten sposób „zaznaczenie" szumów odebranych przez antenę, co umożliwia ich odróżnienie od szumów sytemu odbiorczego, oraz określenie ich wielkości w relacji do znanego poziomu szumów odniesienia. Tak zmodulowany sygnał poddawany jest wzmocnieniu i detekcji. Sygnał wyjściowy niesie informację o wielkości i kierunku zmian mocy szumów wprowadzanych do anteny w relacji do

mocy szumów źródła odniesienia. W artykule przeanalizowano wpływ rodzaju zastosowanego detektora na podstawowe parametry użytkowe radiometru.

2. MODEL RADIOMETRU

Do analizy przyjęto układ radiometru modulacyjnego [3] przedstawiony na rys. 1.



Rys. 1. Uproszczony schemat blokowy radiometru modulacyjnego

Fig. 1. Simplified block diagram of the modulation radiometer

Wzmacniacz w. cz. charakteryzowany jest przez statystycznie równoważne pasmo przenoszenia $B_{w.cz.}$ definiowane następująco [1, 4, 5]

$$B_{w.cz.} \stackrel{df.}{=} \frac{1}{2\pi} \frac{\left[\int_{-\infty}^{\infty} G(\omega)\,\mathrm{d}\omega\right]^2}{2\int_{-\infty}^{\infty} G^2(\omega)\,\mathrm{d}\omega} \tag{1}$$

gdzie $G(\omega)$ jest dwustronną charakterystyką skutecznego wzmocnienia mocy wzmacniacza w. cz.

Pasmo $B_{w.cz.}$ jest pasmem hipotetycznego filtru o charakterystyce prostokątnej, który przenosi sygnał z takim samym błędem statystycznym wartości średniokwadratowej jak rzeczywisty filtr, w przypadku gdy do wejścia zostanie dołączony szum biały.

Wzmacniacz m. cz. charakteryzowany jest przez szumowe pasmo przenoszenia $B_{m.cz.}$ definiowane jako [1, 4]

$$B_{m.cz.} \stackrel{df.}{=} \frac{1}{2\pi} \frac{\int_{0}^{\infty} |H(\omega)|^2\,\mathrm{d}\omega}{|H(0)|^2} \tag{2}$$

gdzie $H(\omega)$ jest zdefiniowaną dla częstotliwości fizycznych funkcją transmitancji układu wzmacniacza m. cz. i filtru wyjściowego.

Pasmo $B_{m.cz.}$ jest pasmem przepustowym hipotetycznego filtru o charakterystyce prostokątnej, który przenosi sygnał z taką samą wartością średniokwadratową jak filtr

rzeczywisty, w przypadku gdy do wejścia zostanie dołączony szum biały. Zatem tak zdefiniowane pasmo jest wygodną miarą szerokości pasma stosowaną przy wąskopasmowych pomiarach wartości średniokwadratowej.

Jako detektor przyjęto nieliniowy układ bezinercyjny opisany dwupołówkową funkcją kwadratową o postaci

$$U_{det} = \beta U_{we}^2 \tag{3}$$

lub jednopołówkową funkcja liniową zdefiniowaną równaniem

$$U_{\text{det}} = \begin{cases} \gamma U_{we} & \text{dla } U_{we} \geqslant 0 \\ 0 & \text{dla } U_{we} < 0 \end{cases} \tag{4}$$

Sygnałem wejściowym detektora jest szum termiczny o równoważnej temperaturze szumowej T. Założono, że szum ten, uformowany przez wzmacniacz w. cz., jest wąskopasmowy i podlega rozkładowi normalnemu. Założono ponadto, że szum podawany na detektor nie posiada składowej stałej.

Dwustronną charakterystykę wzmocnienia skutecznego mocy wzmacniacza w. cz. aproksymowano funkcją eksponencjalną o postaci

$$G(\omega) = G_{ps\,\max}\left[e^{-\alpha(\omega+\omega_0)^2} + e^{-\alpha(\omega-\omega_0)^2}\right] \tag{5}$$

W konsekwencji dwustronną gęstość widmową mocy sygnału na wejściu detektora, przedstawioną w postaci unormowanej na rys. 2, można opisać zależnością

$$S_{we}(\omega) = \frac{N}{2}G(\omega) = \frac{N_{\max}}{2}\left[e^{-\alpha(\omega+\omega_0)^2} + e^{-\alpha(\omega-\omega_0)^2}\right]\left[\frac{\text{W}}{\text{Hz}}\right] \tag{6}$$

przy czym

$$N_{\max} = G_{ps\,\max}kT \tag{7}$$

Korzystając z definicji (1) oraz zależności (5), otrzymujemy

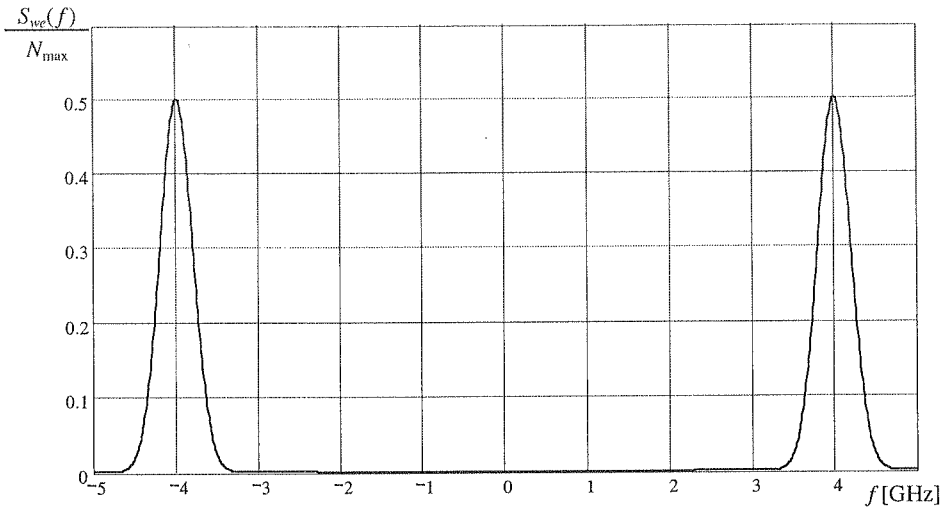
$$B_{w.cz.} = \frac{1}{\sqrt{2\pi\alpha}} \tag{8}$$

Zgodnie z twierdzeniem Wienera–Chinczyna funkcja autokorelacji szumu wejściowego jest równa

$$R_{we}(\tau) = \frac{1}{2\pi}\int_{-\infty}^{\infty} S_{\omega e}(\omega)e^{jw\tau}d\omega = \frac{N_{\max}}{2\sqrt{\pi\alpha}}e^{\frac{-\tau^2}{4\alpha}}\cos\omega_0\tau \tag{9}$$

Zakładając, że szum wejściowy jest procesem ergodycznym, ze względu na brak składowej stałej możemy wyznaczyć jego wariancję jako

$$\sigma_{we}^2 = R_{we}(0) = \frac{N_{\max}}{2\sqrt{\pi\alpha}} \tag{10}$$



Rys. 2. Dwustronna gęstość widmowa mocy szum wejściowego

Fig. 2. Double-sided spectral power density of input noise

Uwzględniając zależność (8), otrzymujemy związek łączący pasmo $B_{w.cz.}$ z wariancją szumu wejściowego

$$\sigma_{we}^2 = \frac{N_{\max} B_{w.cz.}}{\sqrt{2}} \tag{11}$$

Wprowadzając pojęcie obwiedni funkcji autokorelacji

$$\rho_{we}(\tau) = e^{\frac{-\tau^2}{4\alpha}} \tag{12}$$

otrzymujemy ostatecznie

$$R_{we}(\tau) = \sigma_{we}^2 \rho_{we}(\tau) \cos \omega_0 \tau \tag{13}$$

3. DETEKTOR KWADRATOWY

Rozpatrując przejście stacjonarnego szumu normalnego, bez składowej stałej, przez dwupołówkowy układ bezinercyjny o charakterystyce kwadratowej określonej wzorem (3) otrzymujemy wyrażenie opisujące funkcję autokorelacji procesu na wyjściu detektora [2]

$$R_{det}(\tau) = \beta^2 \sigma_{we}^4 + 2\beta^2 R_{we}^2(\tau) \tag{14}$$

Korzystając z relacji (13) otrzymujemy

$$R_{det}(\tau) = \beta^2 \sigma_{we}^4 \left[1 + \rho_{we}^2(\tau) + \rho_{we}^2(\tau) \cos 2\omega_0 \tau\right] \tag{15}$$

Pomijając składnik szybkozmienny, który nie występuje na wyjściu wzmacniacza m. cz. (pełniącego m. in. funkcję filtru podetekcyjnego) funkcję autokorelacji procesu wyjściowego można zapisać w postaci

$$R_{det}(\tau) = \beta^2 \sigma_{we}^4 \left[1 + \rho_{we}^2(\tau)\right] \tag{16}$$

Zgodnie z twierdzeniem Wienera–Chinczyna widmo szumu na wyjściu detektora ma postać

$$S_{det}(\omega) = \int_{-\infty}^{\infty} R_{det}(\tau) e^{j\omega\tau} = \beta^2 \sigma_{we}^4 \left[2\pi\delta(\omega) + \sqrt{2\pi\alpha}\, e^{\frac{-\alpha\omega^2}{2}}\right] \tag{17}$$

Pierwszy składnik opisuje moc składowej stałej, której wartość na wyjściu wzmacniacza m. cz. wynosi

$$P_{DET} = |H(0)|^2 \frac{1}{2\pi} \int_{-\infty}^{\infty} \beta^2 \sigma_{we}^4 2\pi\delta(\omega) d\omega = \frac{1}{2} |H(0)|^2 \beta^2 N_{max}^2 B_{w.cz.}^2 \tag{18}$$

Składnik drugi reprezentuje widmo składowej wolnozmiennej szumu wyjściowego zaprezentowane dla częstotliwości fizycznych na rys. 3. W bardzo wąskim w stosunku do $B_{w.cz.}$. paśmie przenoszenia wzmacniacza m.cz. (w praktyce $B_{m.cz.}/B_{w.cz.} \cong 10^{-8}$) wykładniczy czynnik zmienny jest praktycznie równy jedności, a gęstość widmową szumu wyjściowego można uznać za stałą —

$$S_{\det em.cz.}(\omega) = \beta^2 \sigma_{we}^4 \sqrt{2\pi\alpha}\, e^{\frac{-\alpha\omega^2}{2}} \approx \beta^2 \sigma_{we}^4 \sqrt{2\pi\alpha} = S_{det0} \tag{19}$$

Korzystając z (8) i (11) równanie(19) można przedstawić w postaci

$$S_{det0} = \frac{1}{2} \beta^2 N_{\max}^2 B_{w.cz.} \tag{20}$$

Moc składowej zmiennej szumów na wyjściu wzmacniacza m. cz. określona jest całką

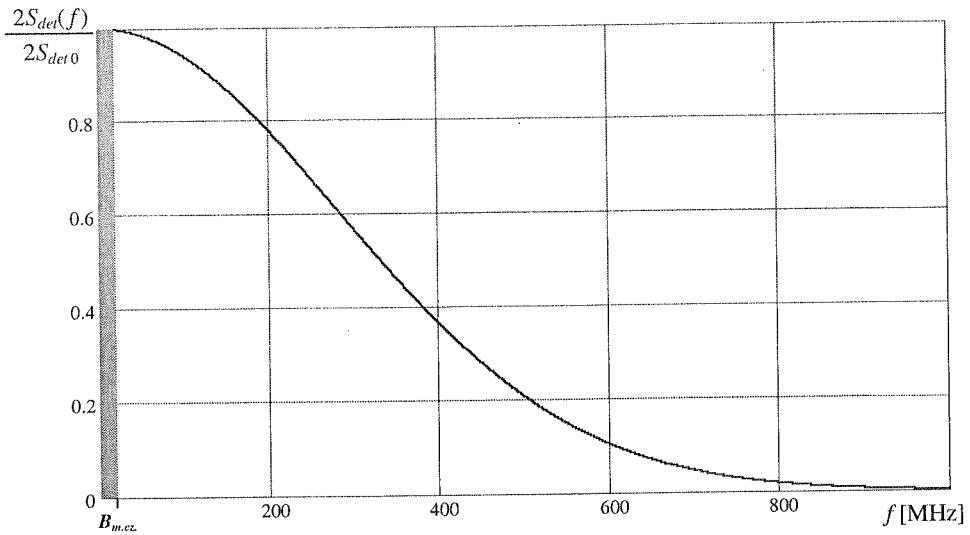
$$P_{det} = \frac{1}{2\pi} \int_0^{\infty} 2S_{det0} |H(\omega)|^2 \, d\omega = 2S_{det0} \frac{1}{2\pi} \int_0^{\infty} |H(\omega)|^2 \, d\omega \tag{21}$$

Korzystając z definicji (2) szumowego pasma przenoszenia toru m. cz., otrzymujemy

$$P_{det} = 2S_{det0} B_{m.cz.} |H(0)|^2 = |H(0)|^2 \beta^2 N_{\max}^2 B_{w.cz.} B_{m.cz.} \tag{22}$$

Stosunek mocy składowych stałej i zmiennej, na podstawie (18) i (21), wynosi

$$\rho_{kwad} = \frac{P_{DET}}{P_{det}} = \frac{1}{2} \frac{B_{w.cz.}}{B_{m.cz.}} \tag{23}$$



Rys. 3. Unormowana jednostronna gęstość widmowa mocy szumu na wyjściu detektora

Fig. 3. Normalised one-sided noise spectral density at detector output

Składową stałą napięcia na wyjściu wzmacniacza m. cz. można zapisać w postaci

$$U_{WY} = a_k \sqrt{P_{DET}} = \frac{b_k}{\sqrt{2}} N_{\max} B_{w.cz.}, \qquad b_k = a_k \beta \, |H(0)| \tag{24}$$

gdzie a_k jest współczynnikiem proporcjonalności zależnym od konkretnego rozwiązania układowego, natomiast b_k uwzględnia ponadto charakterystykę detektora oraz wzmocnienie toru m. cz. dla składowej stałej i bardzo małych częstotliwości.

Wartość skuteczna napięcia składowej fluktuacyjnej szumu na wyjściu wzmacniacza m. cz. – na skutek filtracji dolnoprzepustowej — jest równa

$$U_{wy} = a_k \sqrt{P_{det}} = b_k N_{\max} \sqrt{B_{w.cz.} B_{m.cz.}} \tag{25}$$

Wprowadzając stałą

$$c_k = b_k G_{ps\,\max} k \tag{26}$$

otrzymujemy końcowe wyrażenia opisujące składową stałą i wartość skuteczną składowej zmiennej napięcia na wyjściu wzmacniacza m. cz.

$$\begin{cases} U_{WY} = \frac{c_k}{\sqrt{2}} T B_{w.cz.} \\ U_{wy} = c_k T \sqrt{B_{w.cz.} B_{m.cz.}} \end{cases} \tag{27}$$

Równoważna temperatura szumowa sygnału wejściowego jest sumą skutecznej wejściowej temperatury szumów radiometru i temperatury szumów anteny oraz mierzonych zmian temperatury, gdy do wejścia podłączona jest antena

$$T = T_R + T_A + \Delta T = T_{sys} + \Delta T \tag{28}$$

lub wynosi

$$T = T_{sys} \tag{29}$$

gdy do wejścia podłączona jest odpowiednio dobrana rezystancja odniesienia $R_{odn.}$.

W odbiorniku modulacyjnym napięcie stałe związane z szumami systemu jest eliminowane w wyjściowym układzie różnicowym i w konsekwencji stałe napięcie wyjściowe zależy jedynie od przyrostu temperatury anteny i określone jest zależnością

$$U'_{WY} = \frac{c_k}{\sqrt{2}}\left(T_{sys} + \Delta T\right) B_{w.cz.} - \frac{c_k}{\sqrt{2}} T_{sys} B_{w.cz.} = \frac{c_k}{\sqrt{2}} \Delta T B_{w.cz.} \tag{30}$$

Z drugiej strony w wyrażeniu opisującym składową fluktuacyjną można, nie popełniając istotnego błędu, pominąć — jako bardzo mały — składnik ΔT i wówczas wartość skuteczna składowej zmiennej na wyjściu wzmacniacza m. cz. wynosi

$$U_{wy} = c_k(T_{sys} + \Delta T)\sqrt{B_{w.cz.} B_{m.cz.}} \overset{\Delta T \ll T_{sys}}{\approx} c_k T_{sys} \sqrt{B_{w.cz.} B_{m.cz.}} \tag{31}$$

Na wyjściu różnicowym, ze względu na sumowanie się nieskorelowanych składowych fluktuacyjnych, wartość skuteczna składowej zmiennej jest dwukrotnie większa i w konsekwencji

$$U'_{wy} = 2c_k T_{sys} \sqrt{B_{w.cz.} B_{m.cz.}} \tag{32}$$

Czułość radiometru definiowana jest jako minimalny przyrost sygnału wejściowego (w jednostkach temperatury) ΔT_{min}, który zapewnia równość składowej stałej i wartości skutecznej składowej zmiennej napięcia wyjściowego [4]:

$$U'_{wy}(sT_{min}) = U'_{wy} \tag{33}$$

a zatem, na podstawie (30) i (32), otrzymujemy

$$\Delta T_{min} = 2T_{sys} \sqrt{\frac{2B_{m.cz.}}{B_{w.cz.}}} \tag{34}$$

4. DETEKTOR LINIOWY

Jakościowo inna sytuacja występuje w przypadku jednopołówkowego detektora liniowego. Wiedząc, że detektor taki odtwarza obwiednię szumu wejściowego (w przypadku wąskopasmowego szumu normalnego obwiednia podlega rozkładowi Rayleigha)

i przyjmując nachylenie charakterystyki detektora 45° ($\gamma = 1$), otrzymuje się następujące wyrażenie na gęstość widmową mocy szumu wyjściowego (z pominięciem składowych zgrupowanych wokół częstotliwości środkowej wzmacniacza w. cz. i jej harmonicznych) [2]

$$S_{det}(\omega) = \frac{\pi\sigma_{we}^2}{2}\left[2\pi\delta(\omega) + \frac{1}{4}\sqrt{2\pi\alpha}\,\mathrm{e}^{\frac{-\alpha\omega^2}{2}}\right] \tag{35}$$

Podobnie jak poprzednio otrzymujemy wyrażenia opisujące moce składowej stałej oraz fluktuacyjnej

$$P_{DET} = |H(0)|^2 \frac{1}{2\pi}\int_{-\infty}^{\infty} \frac{\pi\sigma_{we}^2}{2} 2\pi\delta(\omega)\,\mathrm{d}\omega = \frac{\pi}{2\sqrt{2}}|H(0)|^2 N_{\max} B_{w.cz.} \tag{36}$$

$$P_{det} = \frac{1}{2\pi}\int_{0}^{\infty} 2\frac{\pi\sigma_{we}^2}{8}\sqrt{2\pi\alpha}\,|H(\omega)|^2\,\mathrm{d}\omega = \frac{1}{2}\frac{\pi}{2\sqrt{2}}|H(0)|^2 N_{\max} B_{m.cz.} \tag{37}$$

Na mocy równań (36) i (37) stosunek mocy składowych stałej i zmiennej wynosi

$$\rho_{lin} = \frac{P_{DET}}{P_{det}} = 2\frac{B_{w.cz.}}{B_{m.cz.}} \tag{38}$$

Analogicznie jak w poprzednim punkcie otrzymujemy:

$$\begin{cases} U_{WY} = a_l\sqrt{P_{DET}} = c_l\sqrt{TB_{w.cz.}} \\ U_{wy} = a_l\sqrt{P_{det}} = \dfrac{c_l}{\sqrt{2}}\sqrt{TB_{m.cz.}} \end{cases} \tag{39}$$

gdzie:

$$c_l = a_l|H(0)|\sqrt{\frac{\pi}{2\sqrt{2}}G_{ps\max}k} \tag{40}$$

Na wyjściu różnicowym:

$$\begin{cases} U'_{WY} = c_l\sqrt{\left(T_{sys}+\Delta T\right)B_{w.cz.}} - c_l\sqrt{T_{sys}B_{w.cz.}} \\ U'_{wy} = 2\dfrac{c_l}{\sqrt{2}}\sqrt{\left(T_{sys}+\Delta T\right)B_{m.cz.}} \approx \sqrt{2}c_l\sqrt{T_{sys}B_{m.cz.}} \end{cases} \tag{41}$$

Korzystając z definicji (33), otrzymujemy

$$\sqrt{1+\frac{\Delta T_{min}}{T_{sys}}} = 1 + \sqrt{\frac{2B_{m.cz.}}{B_{w.cz.}}} \tag{42}$$

Ze względu na typową w radiometrii bardzo małą wartość stosunku $\Delta T_{min}/T_{sys}$ lewą stronę powyższego wyrażenia można rozwinąć w szereg potęgowy i nie popełniając istotnego błędu ograniczyć się do dwóch pierwszych wyrazów:

$$\sqrt{1+\frac{\Delta T_{min}}{T_{sys}}} = 1+\frac{1}{2}\frac{\Delta T_{min}}{T_{sys}} - \frac{1}{8}\left(\frac{\Delta T_{min}}{T_{sys}}\right)^2 + \ldots \approx 1+\frac{1}{2}\frac{\Delta T_{min}}{T_{sys}} \tag{43}$$

W konsekwencji dostajemy

$$\frac{1}{2}\frac{\Delta T_{min}}{T_{sys}} \approx \sqrt{\frac{2B_{m.cz.}}{B_{w.cz.}}} \tag{44}$$

i ostatecznie wyrażenie na czułość radiometru z detektorem liniowym przyjmuje postać

$$\Delta T_{min} \approx 2T_{sys}\sqrt{\frac{2B_{m.cz.}}{B_{w.cz.}}} \tag{45}$$

5. PODSUMOWANIE

Porównanie zależności(23) i (38) wskazuje, że niezależnie od stosunku pasm wyjściowy stosunek składowych stałej do zmiennej w detektorze liniowym jest lepszy o 6dB niż w detektorze kwadratowym. Jednakże istotne w radiometrii czułości, określone zależnościami (34) i (45), są praktycznie jednakowe.

W rzeczywistych warunkach pracy radiometru jego czułość nie zależy od rodzaju użytego detektora, jest natomiast funkcją temperatury systemu i stosunku szerokości pasm torów po i przeddetekcyjnego.

Detektor liniowy charakteryzuje się liniową zależnością mocy wyjściowej od mierzonej temperatury, natomiast detektor kwadratowy charakteryzuje się liniową zależnością napięcia wyjściowego od temperatury. Ponieważ pomiar napięcia w stosunku do pomiaru mocy obarczony jest mniejszym błędem, oraz ze względu na szeroką dostępność detektorów kwadratowych, jakimi dla bardzo małych sygnałów są diody półprzewodnikowe, w radiometrach mikrofalowych o bezpośrednim wzmocnieniu wykorzystuje się praktycznie jedno lub dwupołówkowy detektor kwadratowy.

6. BIBLIOGRAFIA

1. J. S. Bendat, A. G. Piersol: *Random data: Analysis and measurement procedure.* New York, Wiley Interscience, 1971
2. I. S. Gonorowski: *Радиотехнические цепи и сигналы.* Moskwa, Радио и связь, 1986
3. B. Stec: *Microwave thermography: methods and equipment.* 11th International Conference on Microwaves, Radar and Wireless Communications (MIKON'96), Workshops, Warsaw 1996, pp. 97–114

4. M. E. Tiuri: *Radio astronomy receivers.* IEEE Transactions On Antennas And Propagation, 12, 1964, pp. 930–938
5. D. F. Wait: *The Sensitivity of the Dicke Radiometer.* Journal of research of the National Bureau of Standards — C. Engineering and Instrumentation, Vol. 71C, No. 2, 1967

B. STEC, A. DOBROWOLSKI, W. SUSEK

SENSITIVITY OF MICROWAVE RADIOMETER WITH SQUARE-LAW AND LINEAR DETECTOR

Summary

Stochastic analysis of modulation microwave radiometers with square-law and linear detectors preceded by a radio frequency amplifier is presented in the paper. Assuming ideal detector characteristics it is shown that in typical applications, i.e., in very low power measurements, a type of detector used is of no influence on total radiometer sensitivity. Other aspects of use of a particular detector are also presented.

Comparison of final relations shows that independently of bands ratio the output ratio of constant and variable components in linear detector is 6 dB better than for square detector. However, important in radiometry sensitivities are almost the same

In real conditions, radiometer sensitivity does not depend on the type of detector used, while it is a function of system temperature and a bandwidth ratio of pre- and post-detection channels, respectively.

Linear detector is characterised by linear dependence of output power on measured temperature while for square detector output voltage depends linearly on temperature. Since voltage measurement is related to lower measurement error than power measurement and because of good availability of square detectors such as semiconductor diodes for very low signals, single-half or double-half square detectors are common detectors applied in microwave radiometers with direct amplification.

Keywords: microwave thermograph, radiometer, thermal radiation, sensitivity

KWARTALNIK ELEKTRONIKI I TELEKOMUNIKACJI, 2004, 50, z. 1, ss. 109–130

Protokół mikropłatności i makropłatności w sieciach bezprzewodowych

GRZEGORZ RADZIKOWSKI†, JACEK WOJCIECHOWSKI††

Politechnika Warszawska, Instytut Radioelektroniki
ul. Nowowiejska 15/19, 00-665 Warszawa
†*G.Radzikowski@elka.pw.edu.pl* ††*jwojc@ire.pw.edu.pl*

Otrzymano 2002.12.06
Autoryzowano 2003.03.27

W artykule przedstawiono propozycję protokołu płatności w sieciach bezprzewodowych. Protokół zakłada dwie techniki realizacji transakcji: w trybie on-line z udziałem zaufanej strony — dla tzw. makropłatności oraz w trybie off-line przy użyciu elektronicznego pieniądza, głównie dla transakcji o małym nominale — tzw. mikropłatności. Omówiono scenariusze różnych zdarzeń i transakcji w protokole. Zwrócono uwagę na aspekty bezpieczeństwa płatności uwzględniając techniki kryptografii asymetrycznej i infrastruktury kluczy publicznych. Dla oceny protokołu wykonano testy, w których uwzględniono kryteria charakterystyczne dla środowiska bezprzewodowego.

Słowa kluczowe: protokół, mikropłatności, makropłatności, pieniądze cyfrowe, kryptografia asymetryczna, sieci bezprzewodowe

1. WPROWADZENIE

Internet i telefonia komórkowa wykreowały bogaty wachlarz usług. Jednocześnie pojawił się problem rozliczania oferowanych przez nie usług. W Internecie klient może dokonać płatności podając np. numer karty płatniczej. W sieci komórkowej opłaty za większość usług doliczane są do rachunku telefonicznego. Coraz bardziej różnorodna oferta usług wymusza wprowadzenie alternatywnych metod rozliczania płatności. Operatorzy nie będą mogli wykonywać transakcji finansowych ograniczonych licencjami, a także ponosić odpowiedzialności za przebieg zapłaty i obciążania rachunku klienta.

Konstruując protokół płatności przyjmujemy, że abonent posiada telefon komórkowy i rachunek bankowy, a uczestniczące strony posiadają certyfikowane klucze kryp-

tograficzne. Klucze publiczne uczestników są znane stronom. Płatność przy użyciu telefonu komórkowego ma pozwolić na realizację transakcji na odległość z kontrolowanym przebiegiem zapłaty. Klient może także dokonywać płatności przez telefon osobiście w punktach handlowo-usługowych. Płatność w sieci komórkowej może być zrealizowana w oparciu o różne standardy: WAP czy STK lub nowsze, bardziej zaawansowane przewidziane dla systemów 3G, np. MExE czy UMS.

Celem artykułu jest przedstawienie protokołu płatności przy zakupie dóbr i usług poprzez sieci bezprzewodowe przy użyciu telefonu komórkowego wyposażonego w inteligentną kartę. W artykule zaprezentowano technikę płatności przy użyciu pieniędzy cyfrowych i w oparciu o konto bankowe. W szczególności zaprojektowano protokół mikropłatności monetami cyfrowymi w heterogenicznych sieciach bezprzewodowych. Rozważono identyfikację możliwych zdarzeń, wymianę komunikatów i transmisję danych. W celu określenia jakości i przydatności protokołu mikropłatności zostały przeprowadzone badania przy uwzględnieniu czynników charakterystycznych dla środowiska bezprzewodowego.

W rozdziale 2 artykułu dokonano przeglądu metod płatności elektronicznych. W sekcji 3 przedstawiono podstawowe właściwości płatności elektronicznych. Sekcja 4 mówi o stronie kryptograficznej protokołu. W części 5 został zaprezentowany model płatności. W punkcie 6 przedstawiono ogólne zagadnienia makropłatności, a w 7 protokół mikropłatności. W rozdziale 8 przeprowadzono ocenę protokołu, a sekcja 9 jest podsumowaniem artykułu.

2. PRZEGLĄD BADAŃ I IMPLEMENTACJI

Pierwszy system pieniędzy cyfrowych został zaproponowany w pracy [1], a następnie pojawiło się wiele rozwiązań płatności w trybie off-line [2–4] oraz w trybie on-line [5].

Popularność telefonii komórkowej i coraz większy zakres usług świadczonych przy jej użyciu, wymusiły wprowadzenie systemów płatności w sieciach komórkowych. Podjęto próby implementacji w środowisku bezprzewodowym systemów płatności obecnych dotąd w Internecie. Okazuje się, że charakterystyczne wymagania dla płatności w telefonii komórkowej postawiły wysokie wymagania przed konstruktorami protokołów.

Jedną z propozycji jest projekt ASPeCT [6], który dostarcza rozwiązań płatności przy wykorzystaniu zaufanej strony wspieranej przez infrastrukturę klucza publicznego. W pracach [7, 8] przedstawiono propozycję wdrożenia projektu ASPeCT w telefonii UMTS. W pracy [9] pokazano zastosowanie pieniądza elektronicznego wg schematu Braun'a. Płatność odbywa się w oparciu o usługę SMS. Inne podejście do płatności z udziałem wielu stron zaproponowano w [10]. W [11] została przedstawiona adaptacja protokołu SET w telefonii bezprzewodowej. Dalsze propozycje realizacji płatności przedstawiono w pracach [12–13].

Większość z cytowanych propozycji protokołów płatności w niedostateczny sposób jest dostosowana do sieci komórkowych. Protokół płatności przedstawiony w tym artykule ma charakter uniwersalny, gdyż dostarcza mechanizmów mikro- i makropłatności przy użyciu pieniędzy elektronicznych i w oparciu o konto bankowe.

3. CHARAKTERYSTYKA PŁATNOŚCI

W tej części zostaną przedstawione podstawowe kryteria dla systemu płatności elektronicznych [14–17] i dotychczasowe implementacje.

3.1. WŁAŚCIWOŚCI

Anonimowość — to sposób ochrony prywatności użytkownika pieniądza elektronicznego. Przeprowadzenie każdej transakcji powinno mieć charakter prywatny dla klienta, zaś ujawnienie danych (np. adres) tylko wtedy, gdy wymaga tego charakter usługi.

Tryb off-line — efektywność płatności mierzona jest m.in. czasem i prostotą usługi. Transakcje on- line wymagają połączenia się z bankiem i weryfikacji danych. Tryb off-line tego nie wymaga, przez co jest bardziej wydajny.

Transferowalność — podobnie do właściwości tradycyjnego pieniądza systemy płatności elektronicznych powinny zapewniać możliwość przeniesienia środków pieniężnych pomiędzy osobami bez ingerencji banku (płatności *peer-to-peer*).

Skalowalność — w projektowaniu elektronicznego systemu płatniczego należy uwzględnić globalną naturę handlu. System musi być przygotowany na duży ruch gotówki cyfrowej, funkcjonować bez utraty wydajności w transakcjach międzynarodowych oraz akceptować monety wystawiane przez banki zagraniczne.

Podwójne wydanie — cechy pieniądza elektronicznego implikują łatwość powielania monet cyfrowych. Najczęściej przytaczaną metodą zapobiegania podwójnemu wydaniu jest weryfikowanie każdej monety. Oznacza to konieczność czasochłonnej pracy on-line i każdorazowego łączenia się z instytucją wystawiającą monety. Innym sposobem przeciwdziałania podwójnemu wydaniu jest zastosowanie urządzeń wyposażonych w licznik nominału monet w wirtualnym portfelu.

Śledzenie nadużyć — środowisko pieniądza elektronicznego może być miejscem dla wielu nadużyć, m.in. prania brudnych pieniędzy. Należy stworzyć możliwości dla kontrolowania i wykrywania takich okoliczności.

Koszt — system płatności musi być tani w użytkowaniu i uwzględniać koszty transmisji radiowej realizowanej przez operatora sieci komórkowej.

Czas — szybki przebieg transakcji jest kluczem do powodzenia modelu płatności. Ważne jest dobranie narzędzi kryptograficznych ze względu na czas przetwarzania i wymagane rozmiary pamięci.

3.2. BEZPIECZEŃSTWO

Proponowany model płatności gwarantuje wysoki poziom bezpieczeństwa dzięki zastosowaniu algorytmów kryptografii asymetrycznej. Zapewnia uwierzytelnienie, a więc poprawną identyfikację; poufność gwarantującą, że przesyłane informacje są tajne; integralność, która zapewnia nienaruszalność danych; oraz niezaprzeczalność, czyli wykluczenie sytuacji wyparcia się jednej ze stron po dokonaniu transakcji.

Proponowany mechanizm pracy off-line oraz wykluczenie możliwości podwójnego wydania pieniędzy elektronicznych będą realizowane przez zastosowanie dwóch autonomicznych liczników nominałów gotówki elektronicznej, umieszczonych na inteligentnej karcie w urządzeniu bezprzewodowym. Bezpieczeństwo liczników jest strzeżone przez klucze kryptograficzne, które znane są tylko wystawcy karty.

3.3. ŚRODOWISKO BEZPRZEWODOWE

Implementacja płatności w sieciach bezprzewodowych wymaga rozważenia charakterystycznych dla tego środowiska parametrów.

Pamięć — obecnie wykorzystywane są karty o pojemności 32 kB. Wkrótce pojawią się karty 64 kB i 128 kB. Projektując pieniądz elektroniczny należy poszukiwać mechanizmów, które akceptują ograniczoną pamięć. Należy pamiętać, że użytkownik będzie przechowywał na karcie wiele kluczy publicznych i gotówkę elektroniczną.

Procesor — operacje związane z pieniądzem elektronicznym (obliczenia kryptograficzne) absorbują moc obliczeniową procesora na karcie wydłużając czas przebiegu operacji.

Transmisja — transmisja w sieciach telefonii komórkowej z podsystemem pakietowym GPRS osiąga szybkość 40 kbit/s. W systemach 3G szybkość wahać się będzie od 400 kbit/s do 2 Mbit/s. Szybkości gwarantowane przez GPRS w zasadzie spełniają wymagania, gdyż rozmiar przesyłanych danych na ogół nie będzie przekraczał 1 kbit.

Wyświetlacz — choć rozmiary ekranów w urządzeniach bezprzewodowych są coraz większe, to jego rozmiar nie będzie ciągle wzrastał — to kłóci się z ideą miniaturyzacji urządzeń przenośnych. Mały ekran terminala oznacza ubogą, uproszczoną wizualizację.

Terminal — malejące rozmiary terminali, obok oczywistych zalet (waga, wielkość, estetyka), prowadzą do trudności w posługiwaniu się aplikacjami i nawigacji. Trudniej jest wprowadzić tekst w telefonie komórkowym, niż przy użyciu tradycyjnej klawiatury komputera.

3.4. METODY PŁATNOŚCI

Cechy elektronicznych instrumentów płatniczych, rodzaj kupowanych dóbr i usług, ich wartość oraz charakter przeprowadzanych transakcji implikują dwojakie podejście do systemu płatności elektronicznych: w oparciu o konto bankowe i pieniądze cyfrowe.

Płatność w oparciu o konto polega na połączeniu się z bankiem, weryfikację salda na rachunku bankowym i przeniesieniu środków pieniężnych z jednego rachunku (obciążenie) na drugi (uznanie). Płatność odbywa się w trybie on-line i dotyczy przede wszystkim transakcji na większe kwoty — tzw. makropłatności.

Pieniądz cyfrowy wykorzystuje certyfikaty monet elektronicznych będące odwzorowaniem tradycyjnego pieniądza. Użytkownik pobiera z banku monety elektroniczne, umieszcza na inteligentnej karcie terminala i używa jako środek płatniczy. W proponowanym modelu transakcje z pieniądzem elektronicznym (poza wydaniem i depozytem) odbywają się w trybie off-line i dotyczą przede wszystkim operacji na małe sumy — tzw. mikropłatności.

3.5. PIENIĄDZ CYFROWY

Pieniądz cyfrowy jest elektronicznym ekwiwalentem tradycyjnej gotówki. Jest to pakiet informacji, który dzięki technikom kryptograficznym dostarczanym przez wystawcę pieniądza (m.in. podpis elektroniczny), jest rozpoznawany jako posiadający określoną wartość. Taka porcja informacji jest akceptowana przez sprzedającego w zamian za towary lub usługi nabywane przez kupującego, ale także przez drugą osobę, do której transferowane są cyfrowe monety. Podobnie jak pieniądz tradycyjny, pieniądze cyfrowe przechowują i przenoszą wartość nominalną same w sobie, nie reprezentują wartości pieniężnych zgromadzonych na koncie bankowym.

Najlepszym rozwiązaniem dla zabezpieczenia przed kopiowaniem monet elektronicznych jest wykorzystanie funkcji jednokierunkowej (nazywanej również funkcją skrótu lub hash) i podpisu kryptograficznego monet cyfrowych przez zaufaną stronę. Reprezentacja matematyczna pieniądza cyfrowego w omawianym podejściu jest następująca:

$$(a, H_1(a), b, H_2(b))^k \tag{1}$$

Pierwszą wartością jest numer seryjny monety a, następną wartość funkcji skrótu a–$H_1(a)$. Parametr b reprezentuje wartość nominalną monety, natomiast $H_2(b)$ jest skrótem tej wartości. Pakiet danych składający się z tych czterech elementów jest podpisywany kluczem prywatnym banku wystawiającego monety k[1].

Funkcja jednokierunkowa wykorzystywana do zabezpieczenia cyfrowych pieniędzy musi spełniać następujące wymagania [18, 19]:

- funkcja H jest określona dla wiadomości m o dowolnej długości,
- w wyniku przekształcenia H otrzymujemy skrót wiadomości $H(m)$ o ustalonej, długości n- bitów (np. n = 128 lub 160 bitów),
- skrót wiadomości $H(m)$ jest łatwy do obliczenia,
- jest obliczeniowo niewykonalne (tj. nieopłacalne w stosunku do nakładów) odtworzenie wiadomości m na podstawie jej skrótu $H(m)$,

[1] w tym artykule przyjęto notację, w której oznaczenie np. $(*)^k$ jest prywatnym kluczem strony k, natomiast oznaczenie: $(*)^{1/k}$ jest publicznym kluczem strony k

- funkcja H jest wolna od kolizji, tj. jest obliczeniowo niewykonalne wygenerowanie takiej pary wiadomości (m, m'), dla której $m \neq m'$ i $H(m) = H(m')$; w sensie matematycznym te warunki spełnia funkcja różnowartościowa.

Funkcje skrótu H_1, H_2 zapewniają integralność monet cyfrowych, zapobiegając naruszeniu ich parametrów: nie ma praktycznej możliwości otrzymania wartości a z wartości $H_1(a)$ lub b z $H_2(b)$.

Podpisanie pakietu informacji kluczem prywatnym banku k daje moc prawną i możliwość wykorzystania monety cyfrowej w środowisku płatności elektronicznych. Wartości nominalne monet cyfrowych są dyskretne, a ich nominał może wynosić np.: 0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1, 2, 5, 10, 20, 50, (...). Wielkość największego nominału zależy od wystawcy tj. banku lub wykładni prawnej, która może ograniczać wartości i liczbę monet cyfrowych przechowywaną w portfelu elektronicznym.

4. PROCEDURY I ALGORYTMY KRYPTOGRAFICZNE

Technika kryptografii asymetrycznej zapewnia wymienione w sekcji 3.2 wymogi bezpieczeństwa, dlatego została wykorzystana do protokołu płatności. Ten rozdział przedstawia procedury kryptograficzne szyfrowania i deszyfrowania informacji w trakcie realizacji płatności. W dalszej części przeprowadzono testy poszczególnych algorytmów w celu wyboru rozwiązania najlepiej dopasowanego do proponowanych metod płatności.

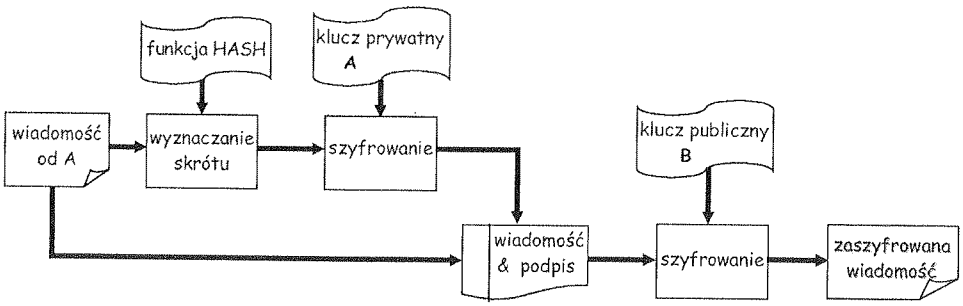
4.1. SZYFROWANIE INFORMACJI

Użytkownik, który chce przesłać pakiet danych do innego użytkownika (podmiotu) musi być w posiadaniu jego klucza publicznego [18, 19]. Aby transport informacji był bezpieczny, podczas szyfrowania wiadomość musi być poddana procesom zobrazowanym na poniższym schemacie.

W pierwszej kolejności powstaje skrót wiadomości $m_1 = H(m)$. Podpis skrótu $m_2 = (H(m))^a$ spełnia warunek uwierzytelnienia i niezaprzeczalności. Zaszyfrowany skrót jest łączony z oryginalnym komunikatem $m_3 = (H(m))^a \| m$ i tak otrzymany komunikat jest szyfrowany kluczem publicznym odbiorcy informacji, dzięki czemu uzyskujemy poufność. Notacja takiej procedury jest następująca:

$$m_4 = ((H(m))^a \| m)^{1/b} \tag{2}$$

gdzie: a — klucz prywatny nadawcy, $1/b$ — klucz publiczny odbiorcy, zaś symbol $\|$ oznacza konkatenację.

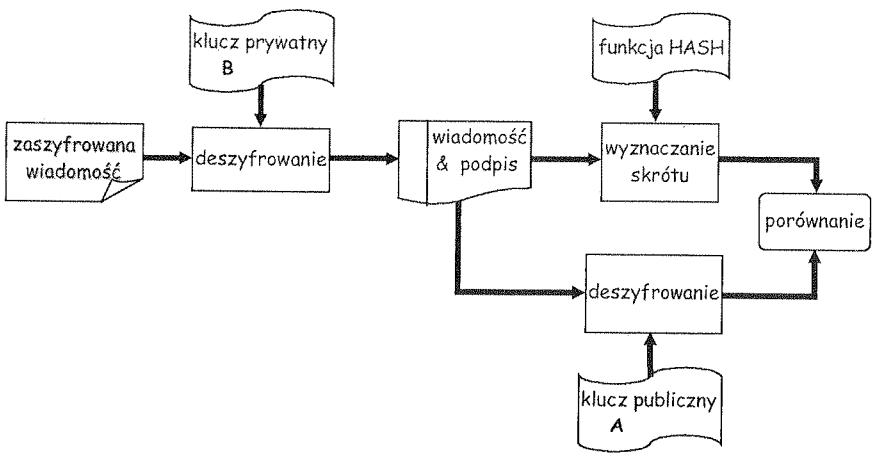


Rys. 1. Szyfrowanie

Fig. 1. Encryption

4.2. DESZYFROWANIE INFORMACJI

W procedurze deszyfrowania, należy wykonać poszczególne operacje w kolejności odwrotnej do szyfrowania oraz przeprowadzić sprawdzenie nienaruszalności informacji.



Rys. 2. Deszyfrowanie

Fig. 2. Decryption

Odebrana wiadomość $\bar{m}_4$ (rys. 2) jest deszyfrowana przy użyciu klucza prywatnego strony B: b (3), a następnie dzielona na dwie części: odebrana wiadomość $\bar{m}$ podpisany skrót $\bar{m}_2$ (4). Następnie skrót wiadomości jest deszyfrowany kluczem publicznym strony A: $1/a$ (5), zaś z wiadomości odebranej jest wyznaczany skrót $H(\bar{m})$. Wyniki obu tych operacji są porównywane (6). Jeśli oba skróty są takie same oznacza to, że

przesłane informacje nie zostały sfałszowane. Opis matematyczny schematu z rys. 2 jest następujący:

$$\bar{m}_3 = \left[\left(\overline{(H(m))^a}\,\|\bar{m}\right)^{1/b}\right]^b \overline{(H(m))^a}\,\|\bar{m} \tag{3}$$

$$\bar{m}_2 = \overline{(H(m))^a} \tag{4}$$

$$\bar{m}_1 = \left(\overline{(H(m))^a}\right)^{1/a} = \overline{H(m)} \tag{5}$$

$$\text{jeśli} \quad \overline{H(m)} = H(\bar{m}) \quad \Rightarrow \quad \text{wiadomość nie została naruszona} \tag{6}$$

$$\text{jeśli} \quad \overline{H(m)} \neq H(\bar{m}) \quad \Rightarrow \quad \text{wiadomość naruszona}$$

4.3. TESTY ALGORYTMÓW KRYPTOGRAFICZNYCH

W celu określenia przydatności poszczególnych algorytmów kryptograficznych do realizacji płatności w środowisku bezprzewodowym zbadano cztery algorytmy: RSA, ECC, Braid i NTRU [18, 21–23]. Zwrócono uwagę na parametry istotne w zastosowaniu do telefonii komórkowej: długość kluczy kryptograficznych, czasy generowania kluczy, kodowania i dekodowania informacji oraz bitowe rozszerzenie wiadomości. Badania wykonano w środowisku Linux na komputerze z procesorem Athlon 1 GHz i 512 MB pamięci RAM dla dwóch grup algorytmów o różnych długościach kluczy (kryterium A i B). Są one przedstawione w tabelach 1 i 2.

Najlepszym jest algorytm o najkrótszych czasach operacji kryptograficznych, małym rozmiarze kluczy i nie rozszerzający wiadomości. Z przeprowadzonych testów widać, że żaden z algorytmów nie spełnia wszystkich kryteriów.

Najszybszymi algorytmami są NTRU i Braid. Niestety, charakteryzują się znaczną długością kluczy i około czterokrotnym zwiększeniem rozmiaru wiadomości. Oznacza to większe wymagania wobec pamięci w telefonie komórkowym i wydłużenie czasu przesyłania informacji.

Algorytmy RSA i ECC są wolniejsze, ale lepiej spełniają postawione kryteria. W szczególności ECC wyróżnia się krótszymi kluczami kryptograficznymi. Pozwala to na umieszczenie w ograniczonej pamięci telefonu czy inteligentnej karty większej ilości danych i sprawia, że czas realizacji transakcji elektronicznych jest krótszy. Wadą algorytmu ECC jest dwukrotne rozszerzenie bitowe wiadomości. Jednak pozostałe parametry ECC są lepsze od innych algorytmów i dlatego ECC będzie wykorzystywany w naszym protokole płatności.

Operacje wyznaczania skrótów przesyłanych komunikatów mogą być wykonywane przy użyciu typowej procedury. W naszych testach posłużono się funkcją MD5 [18, 19], choć inne rozwiązania nie są wykluczone — szczegółowe analizy w tym zakresie będą przedmiotem dalszych badań. Czas realizacji procedury wyznaczania skrótu jest liczony w dziesiątkach μs i nie ma znaczącego wpływu na czas realizacji mikropłatności.

Tabela 1

Parametry operacji kryptograficznych dla różnych algorytmów — kryterium A

Estimates of operational parameters for various cryptographic algorithms — benchmark A

	RSA 1024	ECC 168[2]	NTRU 263	Braid 40
Czas generowania kluczy [ms]	716	32,5	9,9	4,25
Czas szyfrowania [ms]	2,15	70,5	0,95	14,9
Czas deszyfrowania [ms]	24,25	33,5	1,75	7,45
Rozmiar klucza publicznego (moduł) [bit]	1024	168	1841	1000
Rozmiar klucza prywatnego (moduł) [bit]	1024	168	814	167
Rozszerzenie rozmiaru wiadomości	1 – 1	2 – 1	4,5 – 1	4 – 1

Tabela 2

Parametry operacji kryptograficznych dla różnych algorytmów — kryterium B

Estimates of operational parameters for various cryptographic algorithms — benchmark B

	RSA 2048	ECC 196[2]	NTRU 503	Braid 90
Czas generowania kluczy [ms]	1969	57,2	35,6	11,3
Czas szyfrowania [ms]	6,0	127,8	3,3	41,3
Czas deszyfrowania [ms]	151,5	59,4	6,3	19,8
Rozmiar klucza publicznego (moduł) [bit]	2048	196	4024	2639
Rozmiar klucza prywatnego (moduł) [bit]	2048	196	1595	438
Rozszerzenie rozmiaru wiadomości	1 – 1	2 – 1	4,5 – 1	4 – 1

5. MODEL PŁATNOŚCI

W proponowanym modelu rozliczania usług i przepływu środków pieniężnych bierze udział pięciu uczestników. Relatywnie duża liczba stron wynika z założeń prezentowanego modelu: wysokiego stopnia bezpieczeństwa oraz wykorzystania podmiotów obecnych już na rynku, bez tworzenia i absorbowania nowych instytucji.

5.1. UCZESTNICY

Uczestnikami modelu są: klient, kupiec, operator sieci komórkowej, bank oraz organy autoryzacyjne. Poniżej jest przedstawiona krótka charakterystyka każdego z nich.

Klient — użytkownik telefonu komórkowego lub innego urządzenia bezprzewodowego (np. PDA), wyposażonego w inteligentną kartę (SIM, WIM, S@T, UIM, etc. — zależnie od operatora i rodzaju sieci), zdolną do realizacji bezpiecznych transakcji. Kryteria wysokiego stopnia bezpieczeństwa transakcji spełniane są przez wykorzystanie mechanizmów infrastruktury klucza publicznego (PKI).

Sprzedawca — jest stroną sprzedającą towary i usługi, za które klient uiszcza opłatę. Jest pierwszym i jedynym punktem kontaktu dla klienta. Zastosowanie kryptogra-

[2] bez kompresji

fii asymetrycznej we wzajemnych kontaktach, wsparte kwalifikowanymi certyfikatami uwiarygodnia strony transakcji.

Operator sieci komórkowej — jest dostawcą medium transmisyjnego oraz wystawcą lub „współwystawcą" inteligentnej karty. Chociaż jego udział w rozliczaniu transakcji sprowadza się do zabezpieczenia transmisji radiowej pomiędzy kupcem i klientem, to ma decydujący wpływ na skuteczność i poprawność kontaktu klienta z podmiotami biznesowymi.

Inteligentna karta w telefonie *de facto* jest własnością operatora sieci komórkowej. Aby nabrała cech środka płatniczego, operator będzie musiał kooperować z bankami (dlatego wyżej „współwystawca" karty), które dysponują licencją na świadczenie usług finansowych i wydawanie kart płatniczych. Innym rozwiązaniem jest wyposażenie telefonu w dodatkowe gniazdo przeznaczone dla drugiej karty, pełniącej wyłącznie funkcje płatnicze.

Bank — jego rola jest tradycyjna i polega na oferowaniu usług lokowania i kredytowania pieniędzy. Bank poprzez Centrum Autoryzacji Kart weryfikuje wysokość salda klienta i ważność karty oraz realizuje przepływ środków pieniężnych. Bierze udział w wystawianiu inteligentnych kart i zawiera umowy z operatorem sieci komórkowej oraz klientem banku na możliwość realizacji płatności przy wykorzystaniu telefonu komórkowego. Wystawienie karty przez bank będzie polegało na nadaniu karcie elektronicznej w telefonie uprawnień karty płatniczej.

Organ autoryzacyjny — to strona modelu, która wystawia certyfikaty kluczy publicznych wszystkich uczestników. Certyfikowanie podpisów daje gwarancję poprawnej identyfikacji stron transakcji.

5.2. OPIS DANYCH

Ta sekcja jest wstępem do opisu protokołu płatności i przedstawia dane wykorzystywane w transakcjach.

Rozmiary danych dla poszczególnych informacji zostały określone arbitralnie przy dbałości o minimalizację rozmiarów wiadomości. Przykładowe obliczenie rozmiaru bitowego monety wykonano przy następujących założeniach: nominał monety — 3 znaki, numer seryjny — 9 znaków, rozmiar funkcji skrótu 128 bitów, algorytm ECC oraz dwukrotne wydłużenie rozmiaru bitowego wiadomości (przy założeniu braku kompresji). Razem: $(3 \cdot 8 + 128 + 9 \cdot 8 + 128) \cdot 2 = 704$ bity.

6. MAKROPŁATNOŚCI

Makropłatności to transakcje pieniężne o dużej wartości. Taka technika realizacji transakcji jest nieefektywna pod względem kosztowym i czasowym dla małych płatności i dlatego jest dedykowana przede wszystkim do płatności o dużych nominałach.

Tabela 3

Dane używane w protokole płatności

Data used in the payment protocol

Dane	Opis	Maksymalny rozmiar danych [bit]
UserId	Dane identyfikujące użytkownika	80
BankId	Dane identyfikujące bank	80
MerchId	Dane identyfikujące sprzedawcę	80
UserCA	Certyfikat klucza publicznego użytkownika	880
BankCA	Certyfikat klucza publicznego banku	880
MerchCA	Certyfikat klucza publicznego sprzedawcy	880
TransId	Numer identyfikacyjny transakcji	80
ProdId	Identyfikatory alfanumeryczne produktów	240
UserAccNum	Numer konta bankowego użytkownika	208
MerchAccNum	Numer konta bankowego sprzedawcy	208
Address	Adres pocztowy lub e-mail klienta	320
TotAmount	Wartość nominalna transakcji	48
TimeStamp	Identyfikator czasowy dla transakcji	112
info	Informacje dotyczące transakcji: obciążenia, uznania rachunku, braku możliwości realizacji transakcji, etc.	400
SetCoinAsk	Prośba o wystawienie monet o podanych w poleceniu nominałach	216
SetCoin	Zestaw monet elektronicznych	7040
Coin(s)	Moneta(y) elektroniczna(e)	704
Ackn	Potwierdzenie zdarzenia	400

Makropłatności obejmują trzy podstawowe operacje: wydanie pieniędzy, płatność oraz transfer. Wydanie środków pieniężnych polega na podjęciu gotówki z automatycznego elektronicznego urządzenia (np. ATM), przy użyciu telefonu komórkowego. Wypłata fizycznego pieniądza oznacza bezpośredni kontakt użytkownika z urządzeniem. Funkcję banku pełni urządzenie ATM, które może odbierać informacje drogą radiową, a następnie komunikując się z bankiem przetwarzać je.

Procedurę makropłatności użytkownik może przeprowadzić na odległość lub będąc w bezpośrednim kontakcie z dostawcą towarów lub usług. Po wybraniu pożądanych produktów użytkownik inicjalizuje procedurę makropłatności. Wysyła przez sieć radiową zaszyfrowaną wiadomość do sprzedawcy. Sprzedawca dodaje pewne informacje i przesyła komunikat do banku. Zgodnie z prawem bankowym, bank musi zapytać klienta o potwierdzenie transakcji.

Transfer polega na przeniesieniu środków pieniężnych w banku z jednego rachunku na inny. Przelew środków pomiędzy rachunkami rozpoczyna użytkownik, którego rachunek jest obciążany. Właściciel rachunku uznawanego nie uczestniczy aktywnie w procesie.

W transakcje makropłatności zaangażowany jest bank. Każda transakcja związana jest z operacjami na koncie klienta czy sprzedawcy. Oznacza to, że strony zawsze muszą

autoryzować transakcję wykorzystując w tym celu klucz prywatny. Wówczas operacje szyfrowania i deszyfrowania komunikatów mają postać pokazaną w rozdziałach 4.1 i 4.2.

Makropłatności przy użyciu terminala bezprzewodowego doczekały się pierwszych projektów i wdrożeń. Dotychczasowe implementacje są dalekie od efektywności ze względu na niedoskonałości w obszarze bezpieczeństwa oraz relatywnie wysoki koszt transakcji (koszt kilku SMS). Propozycje makropłatności przedstawione w pracach [6, 10, 11] są znacznie lepsze, lecz również nie spełniają wszystkich wymagań stawianych przez środowisko bezprzewodowe (sekcja 3.3). Podstawowym problemem jest brak uniwersalnego modelu płatności dla telefonii komórkowej.

7. MIKROPŁATNOŚCI

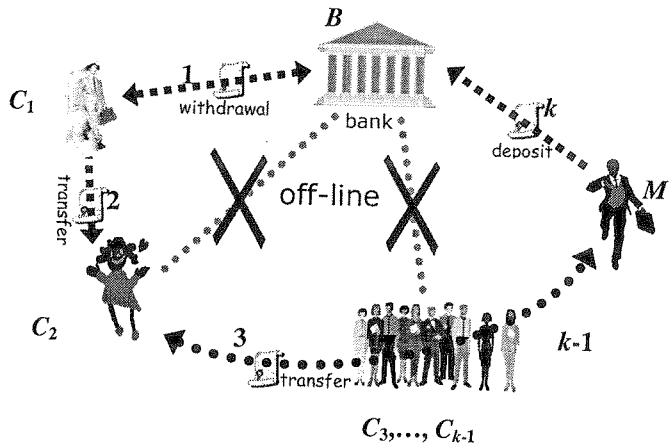
Mikropłatność reprezentuje transakcję elektroniczną o niskiej wartości. Określenie „niska wartość" ma wymiar względny i zależy od systemu mikropłatności, a także zapisów legislacyjnych. Wartość pojedynczej mikropłatności waha się w granicach od kilku centów (nawet ułamków centa) do kilku, czasami nawet kilkudziesięciu euro. Niektóre systemy mikropłatności dopuszczają płatności o wysokości ułamków centa. W artykule mikropłatności oznaczają płatności przy użyciu pieniądza elektronicznego.

7.1. ANONIMOWOŚĆ

W przypadku makropłatności użytkownik dokonując transakcji w oparciu o konto bankowe musi się przedstawić — nie jest więc realizowana zasada anonimowości. Transakcja z użyciem pieniądza cyfrowego ma być podobna do transakcji z wykorzystaniem pieniądza fizycznego, gdzie na ogół klient pozostaje anonimowy. Najbardziej znaną metodą realizacji anonimowości jest „zaślepienie" wydanych monet. Stosuje się tutaj techniki m.in. Okamoto-Schnorr'a [20] lub Chaum'a [1], wykorzystujące „zaślepianie" monet. Następnie przy użyciu protokołu „wytnij i wybierz" podpisuje losowo wybraną monetę cyfrową. Oznacza to, że użytkownik musi przygotować, „obliczyć" i wysłać każdą monetę — to jest nieefektywne dla środowiska bezprzewodowego, jak pokazano w sekcji 3.3.

Innym sposobem osiągnięcia anonimowości jest proponowany tutaj schemat transferu pieniądza cyfrowego w trybie off-line z zachowaniem prywatności (rys. 3). Poszczególne transakcje opatrzono numerami: $1, 2, \ldots, k-1, k$, natomiast uczestników notacją literową: B — bank, $C_1, C_2, \ldots, C_{k-1}$ — klienci, M — sprzedawca.

W zaproponowanym schemacie użytkownik nie przygotowuje monet do podpisu. Wysyła jedynie do banku informację o zbiorze monet, które chce nabyć. Bank wydaje monety. Dzięki możliwości transferu (por. sekcja 7.4) użytkownik może przekazywać monety innej osobie w trybie off-line, tj. bez ingerencji banku i bank nie może ich „śledzić". Dlatego transakcje oparte o tą technikę zapewniają anonimowość



Rys. 3. Transferowalność

Fig. 3. Transferability

użytkownikom. Pełny cykl obiegu pieniądza cyfrowego, tj. od wydania (1: $B \rightarrow C_1$) do zdeponowania (k : $M \rightarrow B$) może przebiegać z udziałem dowolnej liczby uczestników, bez pośrednictwa banku.

Dzięki anonimowości przez transferowalność, procesor urządzenia bezprzewodowego nie jest nadmiernie obciążany przez procedury kryptograficzne, tak jak w schematach Okamoto-Schnorr'a [20] czy Chaum'a [1], ponieważ monety przygotowywane są przez bank. Użytkownik nie wykonuje wielu skomplikowanych operacji. Czas realizacji mikropłatności jest krótszy, a sama operacja prostsza. Potwierdzają to wyniki z przeprowadzonych testów (rozdział 8). Szczegółowy opis procedur mikropłatności omówiono w sekcjach 7.2–7.5

7.2. WYDANIE PIENIĘDZY

W celu otrzymania pieniądza elektronicznego klient wysyła do banku zaszyfrowaną informację o zbiorze monet, które chce otrzymać. Klient musi uwierzytelnić się w banku, dlatego przesyłana wiadomość jest podpisana jego kluczem prywatnym $((H(m))^a \| m)^{1/b}$, gdzie m to prośba wydania monet 'AskWithdraw'. Na rys. 3 prośba o wydanie monet cyfrowych odpowiada zdarzeniu (1: $C_1 \rightarrow B$).

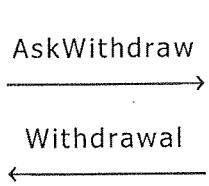
Polecenie prośby o monety zawiera następujące informacje:

AskWithdraw: [UserId, UserAccNum, SetCoinAsk]

W odpowiedzi bank przesyła monety elektroniczne (1: $B \rightarrow C1$):

Withdrawal: [SetCoin, TransId, TimeStamp]

obciążając jednocześnie rachunek bankowy klienta.



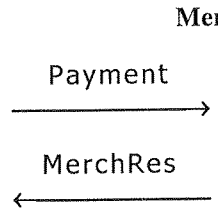
Rys. 4. Mikrowydanie

Fig. 4. Microwithdrawal

7.3. PŁATNOŚĆ

Protokół płatności z użyciem pieniądza elektronicznego ma następujący scenariusz
Klient płacąc za dobra lub usługi przesyła polecenie przy użyciu telefonu komórkowego:

Payment: [ProdId, Coins, Address]



Rys. 5. Mikropłatność

Fig. 5. Micropayment

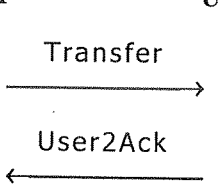
Parametr 'Coins' reprezentuje monety cyfrowe. Na rys. 3 procedura płatności występuje w zdarzeniu ($k-1$: $C_{k-1} \rightarrow M$). W odpowiedzi na mikropłatność, klient dostaje potwierdzenie ($k-1$: $M \rightarrow C_{k-1}$):

MerchRes: [TransId, TimeStamp, Ackn]

Po transakcji zapłaty następuje dostarczenie dobra do klienta. W zależności od charakteru towaru czy usługi (fizyczne, wirtualne) wykorzystywany jest adres tradycyjny lub e-mail. Klient nie musi się uwierzytelniać, dlatego wysyłany komunikat nie jest podpisywany kluczem prywatnym użytkownika. Zgodnie z notacją wprowadzoną w sekcji 4.1 komunikat będzie miał postać $(H(m)\|m)^{1/b}$, gdzie m to 'Payment'.

7.4. TRANSFER

Pieniądz elektroniczny może być przekazywany innej osobie bez udziału banku jak to zostało pokazane na rys. 3 w zdarzeniach (2: $C_1 \rightarrow C_2$), (3: $C_2 \rightarrow C_3$), ...,



Rys. 6. Mikrotransfer

Fig. 6. Microtransfer

$(k-1\colon C_{k-2} \to C_{k-1})$.

Klient 1 przesyła do klienta 2 wiadomość zaszyfrowaną kluczem publicznym odbiorcy (klienta 2), w której umieszczone są:

Transfer: [Coins]

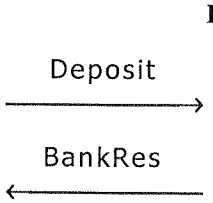
W odpowiedzi dostaje potwierdzenie transakcji:

User2Ack: [Ackn]

Podobnie jak w przypadku płatności, operacja transferu monet cyfrowych nie wymaga podpisu kluczem prywatnym użytkownika. Polecenie transferu jest szyfrowane kluczem publicznym odbiorcy według schematu $(H(m)\|m)^{1/b}$, gdzie m to 'Transfer'.

7.5. DEPOZYT

Posiadacz pieniądza elektronicznego może go zamienić na pieniądz tradycyjny, w postaci fizycznej gotówki (przez ATM) lub uznania rachunku bankowego. Procedura transakcji jest następująca. Klient przesyła do banku lub urządzenia ATM zaszyfrowaną



Rys. 7. Mikrodepozyt

Fig. 7. Microdeposit

wiadomość. Ponieważ stroną w transakcji jest bank użytkownik musi uwierzytelnić się, tj. podpisać komunikat swoim kluczem prywatnym: $((H(m))^a\|m)^{1/b}$, gdzie m to wiadomość 'Deposit' zawierająca następujące informacje:

Deposit: [UserId, UserAccNum, Coins]

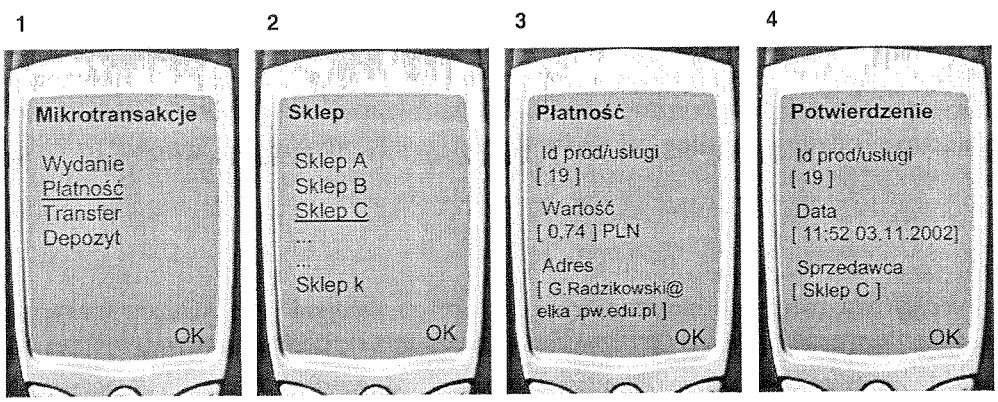
Po zdeponowaniu gotówki elektronicznej w banku lub urządzeniu ATM, użytkownik dostaje tradycyjne pieniądze lub jego konto w banku zostaje uznane na wysokość transakcji. Bank wysyła potwierdzenie zdarzenia:

BankRes: [TransId, TimeStamp, Ackn]

Stosownie do rys. 3 deponowanie cyfrowego pieniądza przez sprzedawcę zobrazowane jest w procesie (k: $M \rightarrow B$), zaś osoby prywatnej ($1, 2, \ldots, k-1$: $C_1, C_2, \ldots, C_{k-1} \rightarrow B$) — (nie pokazano na rys. 3).

7.6. PRAKTYCZNA REALIZACJA MIKROPŁATNOŚCI

W tej sekcji pokazano przykładową realizację mikropłatności. Użytkownik przy wykorzystaniu bezprzewodowego terminala zakupi usługę przy pomocy posiadanych monet cyfrowych. Przykładowy scenariusz mikropłatności zobrazowano na poniższym schemacie.



Rys. 8. Rzeczywista realizacja mikropłatności

Fig. 8. Micropayment realisation

Na rys. 8 pokazano widoki z ekranu telefonu komórkowego z przebiegu mikrotransakcji. Użytkownik dokonuje zakupu na odległość. Kupowanym dobrem jest np. plik muzyczny, który zostanie dostarczony pocztą elektroniczną.

W pierwszej kolejności (ekran 1) użytkownik wybiera rodzaj operacji — w tym przypadku płatność cyfrowym pieniądzem. Następnie wybierany jest dostawca usługi (ekran 2). Nazwa dostawcy związana jest z jego numerem dostępu (MSISDN) i kluczem publicznym. Po wyborze sprzedawcy, użytkownik wpisuje parametry kupowanej usługi czy towaru (ekran 3), tj. identyfikator produktu — 19, jego wartość — 0,74 PLN oraz adres e-mail, na który ma być dostarczona usługa. Po akceptacji wprowadzonych danych aktywowane są procedury kryptograficzne. Zaszyfrowana informacja jest przesyłana pod numer związany z nazwą Sklep C.

Po przetworzeniu informacji u dostawcy usługi, klient otrzymuje potwierdzenie zapłaty (ekran 4). W potwierdzeniu umieszczone są identyfikator usługi, data realizacji transakcji oraz nazwa dostawcy usługi.

8. OCENA PROTOKOŁU PŁATNOŚCI

U podstaw założeń przy konstruowaniu protokołu płatności w sieciach komórkowych leżała dbałość autorów o kilka podstawowych czynników: bezpieczeństwo i prostotę oraz szybkość transmisji i rozmiar danych przesyłanych przez sieć radiową podczas realizacji transakcji. Poniżej zostanie przeprowadzona analiza protokołu ze szczególnym uwzględnieniem tych czynników.

8.1. ANALIZA BEZPIECZEŃSTWA TRANSAKCJI

Bezpieczeństwo realizowane jest dzięki wykorzystaniu kryptografii asymetrycznej i infrastruktury klucza publicznego. Certyfikaty kluczy publicznych pozwalające na autoryzację stron mogą być przesyłane tylko przy pierwszym kontakcie stron. Zmniejsza to rozmiar transmitowanych danych.

Protokół jest odporny na ataki zewnętrzne naruszające bezpieczeństwo transakcji: podsłuchiwanie komunikatów (dzięki szyfrowaniu kluczem publicznym odbiorcy komunikatu), zmiana wiadomości (dzięki funkcji skrótu) oraz podszywanie się (wykluczone poprzez podpisanie wiadomości). System jest także bezpieczny od wewnątrz, tzn. żadna ze stron biorących udział w transakcji nie może oszukiwać. Klucze prywatne klienta, banku i kupca znane są tylko właścicielom i nikt inny nie może podpisać komunikatu. W przypadku przechwycenia transmitowanych danych, nie jest możliwa skuteczna ingerencja w ich zawartość (spełniona zasada integralności danych).

Klient nie może oszukać sprzedającego, gdyż przed otrzymaniem usługi musi uiścić opłatę. Klient nie może oszukać banku, ponieważ każda transakcja jest autoryzowana na podstawie salda rachunku i jego ważności. Klient nie może wyprzeć się transakcji, ponieważ istnieje podpisany przez niego komunikat. Bank może udowodnić poprawność podpisu cyfrowego klienta i potwierdzić autentyczność transakcji.

Sprzedawca nie ma możliwości oszukiwania podczas transakcji, lub jeśli się tego dopuści zostanie ono zidentyfikowane. Transakcja kończy się wysłaniem potwierdzenia do klienta. Klient może anulować transakcję, jeśli stwierdzi nieprawidłowości w otrzymanej usłudze.

Bank nie może oszukać klienta. W relacjach banku z klientem (wydanie, depozyt) wiadomość zawsze jest podpisywana. Klient na koniec transakcji otrzymuje podpisane potwierdzenie transakcji i jeśli zachodzi niezgodność może dochodzić swoich praw.

8.2. ROZMIAR DANYCH

Płatność musi charakteryzować się prostotą i łatwością obsługi. Jakość modelu płatności, obok poziomu zabezpieczeń, jest mierzona m.in. przez rozmiar przesyłanych danych (Tabela 4) i czas przebiegu poszczególnych operacji. Różne lub nieprecyzyjnie określone rozmiary transmitowanych danych są pochodną liczby przesyłanych monet cyfrowych. Przyjęto, że rozmiar jednorazowo przesłanego portfela będzie ograniczony do 20 monet, a operacja mikropłatności może być wykonana przy użyciu maksymalnie pięciu monet.

Tabela 4

Rozmiar danych transmitowanych podczas transakcji mikropłatności (dane w bitach)

Data sizes transmitted during micropayment transactions (bit)

Zdarzenie	Użytkownik	Bank	Sprzedawca	Użytkownik 2
Wydanie	1520	14972	—	—
Płatność	$2784 \div 8160$	—	1440	—
Transfer	$k * 1408 + 256$	—	—	1056
Depozyt	$k * 1408 + 1088$	1696	—	—

gdzie k jest liczbą monet cyfrowych $k = 1, 2, 3, \ldots 20$;

Rozmiar danych zależy od rodzaju operacji, liczby monet oraz obecności (lub nie) certyfikatu kluczy publicznych w komunikacie. Rozmiar bitowy poszczególnych operacji jest ograniczony do niezbędnego minimum i wynika z założeń przyjętych w tabeli 3, komunikatów wysyłanych w protokole mikropłatności oraz rodzaju algorytmu kryptograficznego wykorzystanego w protokole, tj. ECC.

8.3. CZASOCHŁONNOŚĆ TRANSAKCJI

Czas mikropłatności jest determinowany głównie przez trzy czynniki: procedury kryptograficzne, zestawienie połączenia radiowego oraz przesyłanie danych. Tabela 5 podaje sumaryczne dane dla procedur kryptograficznych. Liczba procedur kryptograficznych dla każdej operacji została określona przy uwzględnieniu informacji dotyczących szyfrowania i deszyfrowania (por. sekcje 4.1, 4.2) oraz samej struktury protokołu mikropłatności (rozdział 7). W kilku operacjach zrezygnowano z niektórych procedur kryptograficznych (np. podpis) na rzecz uproszczenia i przyspieszenia działania protokołu, tam gdzie nie jest potrzebne uwierzytelnienie. Na podstawie liczby procedur można było obliczyć czasy realizacji poszczególnych transakcji. Eksperymenty zostały wykonane na komputerze PC z procesorem 1 GHz i 512 MB pamięci RAM, w środowisku Linux. Do szyfrowania komunikatów wykorzystano algorytm ECC 168 i funkcję jednokierunkową MD5. Procedura szyfrowania wiadomości, a więc: wyznaczenie skrótu, podpis kluczem prywatnym i zaszyfrowanie kluczem publicznym odbiorcy, zajmuje w zależności od rozmiaru wiadomości i rodzaju operacji od 380 do 430 ms.

Tabela 5

Liczba operacji kryptograficznych podczas transakcji mikropłatności

The number of cryptographic operations during micropayment transactions

Zdarzenie	Operacja	Użytkownik	Bank	Sprzedawca
Wydanie	ECC podpis	—	—	—
	ECC szyfrowanie	1 na operację	—	—
	ECC deszyfrowanie	1 na operację	2 na operację	—
	Wyznaczenie skrótu	1 na operację	1 na operację	—
Płatność	ECC podpis	—	—	—
	ECC szyfrowanie	1 na operację	—	—
	ECC deszyfrowanie	1 na operację	—	1 na operację
	Wyznaczenie skrótu	1 na operację	—	1 na operację
Transfer	ECC podpis	—	—	—
	ECC szyfrowanie	1 na operację	—	—
	ECC deszyfrowanie	—	—	—
	Wyznaczenie skrótu	1 na operację	—	—
Depozyt	ECC podpis	1 na operację	—	—
	ECC szyfrowanie	1 na operację	—	—
	ECC deszyfrowanie	—	2 na operację	—
	Wyznaczenie skrótu	1 na operację	1 na operację	—

Transmisja w kanale radiowym (sygnalizacja i przesyłanie danych) była symulowana dla parametrów standardu GSM z podsystemem pakietowej transmisji danych GPRS i szybkości 40 kbit/s. Czas wykonania operacji jest obliczany od chwili inicjalizacji zdarzenia do chwili otrzymania potwierdzenia. Procedury sygnalizacyjne w sieci GSM, w zależności od wzajemnego położenia stron, tzn. w tym samym lub różnym Mobile Switching Centre, zajmują około 8 do 10 s.

W wyniku eksperymentów stwierdzono, że w zależności od przedstawionych czynników oraz rodzaju operacji, procedura mikropłatności może trwać od 20 do 25 sekund. W porównaniu do obecnych transakcji kartą płatniczą jest to około 3 — 5 razy krócej.

Z wykonanych testów można wywnioskować, iż czasy procedur kryptograficznych są pomijalnie małe w stosunku do czasu całej mikropłatności. Oznacza to, że czas szyfrowania (deszyfrowania) nie jest krytycznym parametrem. Ważniejsza staje się prostota implementacji sprzętowej i wymagania na pamięć dla protokołu.

9. PODSUMOWANIE I KIERUNKI PRZYSZŁYCH PRAC

System płatności obok zapewnienia bezpieczeństwa transakcji musi być przyjazny dla jego użytkowników. Przedstawiony model spełnia te oczekiwania poprzez możliwość transakcji na duże kwoty oraz mikropłatności. Dzięki realizacji anonimowości przez transfer użytkownik ma zapewnioną prywatność, a procedura nie absorbuje użyt-

kownika i procesora na karcie w skomplikowane i czasochłonne procesy kryptograficzne.

Projektując protokół płatności skoncentrowano się na minimalizacji liczby przesyłanych danych. Dzięki temu informacje przesyłane pomiędzy stronami mają zoptymalizowaną długość, dostosowaną do środowiska telefonii komórkowej. Natomiast użytkownik może w szybki i tani sposób wykonywać płatności.

Istotną kwestią jest wybór techniki kryptograficznej. Najbardziej powszechny na rynku algorytm RSA ma wady w postaci długich kluczy kryptograficznych oraz czasu wymaganego do szyfrowania i deszyfrowania. Lepsze wyniki w tym zakresie, co potwierdziły przeprowadzone badania, dostarczają algorytmy oparte na krzywych eliptycznych (ECC) i zostały wykorzystane w naszym protokole.

Zaproponowany protokół mikropłatności wymaga dalszych eksperymentów i testów w rzeczywistym środowisku. Punktem krytycznym jest problem fizycznego przenoszenia monet cyfrowych pomiędzy inteligentnymi kartami w warunkach zerwania połączenia lub awarii sieci radiokomunikacyjnej. Jak wskazano w 3.1, jedną z największych trudności w posługiwaniu się pieniądzem cyfrowym w trybie off-line jest podwójne wydanie. W przypadku zerwania połączenia podczas przenoszenia monet cyfrowych, pojawia się problem odzyskania i wykluczenie powielenia monet. Powielenie monety, ma miejsce, gdy w jednym portfelu wirtualnym nie został zmniejszony stan licznika monet, a w drugim — docelowym, już został zwiększony.

Dalszych eksperymentów wymaga implementacja i testy protokołu w środowisku bezprzewodowym. Umieszczenie aplikacji wirtualnego portfela monet cyfrowych na inteligentnej karcie procesorowej pozwoli na przebadanie protokołu w warunkach rzeczywistych.

Odrębnych badań wymaga także analiza ekonomiczna i prawna modelu, m.in. wpływ podaży monet cyfrowych na rynek monetarny i umocowania legislacyjne, rozstrzygające o tym, kto będzie wystawcą monet cyfrowych: banki komercyjne czy tylko bank centralny i w jakim stopniu będzie to wpływało na dystrybucję i użytkowanie pieniądza cyfrowego.

10. BIBLIOGRAFIA

1. D. Chaum: *Blind Signatures Untraceable Payments.* Advances Cryptology, Proceeding of Crypto'82, N.Y.
2. D. Chaum, den Boer, van Heyst, S. Mjolsnes, A. Steenbeek: *Efficient Offline Electronic Checks.* Proceedings of Eurocrypt '89, 1990, pp. 294-301.
3. S. Brands: *Untraceable Off-Line Cash in Wallets with Observers.* Crypto'93, pp. 302-318.
4. S. Brands: *Off-Line Electronic Cash Based on Secret-Key Certificates.* Proceedings of LATIN '95, 1995.
5. D. Chaum: *Online Cash Checks.* Proceedings of Eurocrypt '89, 1990, pp. 288-293.
6. G. Horn, H.J. Hitz: *ASPeCT: Advanced Security for Personal Communications Technologies.* Evaluating Report, June 1997.

7. G. Horn, B. Preneel: *Authentication and Payment in Future Mobile Systems.* Computer Security - ESORICS '98 Proceedings, Lecture Notes in Computer Science vol. 1485, Springer-Verlag, Berlin, 1998, pp. 277-293.
8. K.M. Martin, et al: *Secure Billing for Mobile Information Services in UMTS.* Proceeding of ISEN, 1998.
9. L. Pesonen: *A Comparison of Chaum's and Brand's Electronic Cash Schemes.* Tik-110.501, Seminar on Network Security, Department of Computer Science and Engineering, Helsinki University of Technology, 2000.
10. M. Peirce, D. O'Mahony: *Multi-Party Payments for Mobile Services.* Technical Report TCD-CS-1999-48, Department of Computer Science, Trinity College Dublin, Ireland, September 1999.
11. K. Wrona, G. Zavagli: *Adaptation of the SET Protocol to Mobile Networks and to the Wireless Application Protocol.* Ericsson Eurolab Deutschland Herzogenrath, 1998.
12. Q. Deng: *Mobile Payments.* in Advanced Seminar: Electronic Payments Systems, July 2001.
13. T. Jokinen: *Digital Signatures in Mobile Commerce.* Helsinki University of Technology, November 2000.
14. J.F. Hampe, P.M.C. Swatman, P.A. Swatman: *Electronic Commerce: The End of the Beginning.* Proc. of 13th International Bled Electronic Commerce Conference, June 2000.
15. S. Brands: *Electronic Cash in the Internet.* Proceeding of International Society Symposium, California, 1995.
16. N. Ferguson: *Single of Thermae Off-Line Coins.* Proceeding of Eurocrypt '93.
17. D. Bunitas, E. Mazuk: *Digital Cash and Electronic Commerce.* 1997.
18. A. Menezes, P. Van Oorschot, S. Vanstone: *Handbook of Applied Cryptography.* CRC Press, 1996.
19. B. Schneier: *Applied Cryptography.* J.W& S, 1996.
20. T. Okamoto: *Provably Secure and Practical Identification Schemes and Corresponding Signature Schemes.* Proceeding of Crypto '92, Springer Verlag, 1992.
21. E. D. Win, B. Preneel: *Elliptic Curve Public Key Cryptosystems — an Introduction.* Katholieke Universiteit Leuven.
22. K. Ko, S. Lee, J. Cheon: *New Public-key Cryptosystem Using Braid Groups.* Korea Advanced Institute of Science and Technology, Taejon, Korea.
23. The NTRU Public Key Cryptosystem: *Operating Characteristics and Comparison with RSA, ElGamal, and ECC Cryptosystems.* www.ntru.com.

G. RADZIKOWSKI, J. WOJCIECHOWSKI

MICROPAYMENT AND MACROPAYMENT PROTOCOL IN WIRELESS NETWORKS

Summary

The article proposes a payment protocol in wireless networks. The protocol assumes two techniques of transaction realisation with the division between on-line payments with a trust third party — for macropayments and transactions in the off-line mode with the use of electronic money, mainly for transactions of small denomination — the so-called micropayments. Various events and transactions in the protocol as well as the scope of information and announcements required for them are presented. An asymmetric cryptography and a public key infrastructure are exploited to provide payment security. In order to define

quality of the protocol under regard of the characteristic criteria for a wireless environment were carried out its examinations.

Usefulness of the payment protocol was evaluated using various measures. The best system has to demon-strate friendly proprieties for its users. An introduced model fulfils these expectations by delivering possibility of transactions of large amounts as well as of micropayments. Thanks to the use of digital cash transfer a user gets confidentiality. This makes the procedure simpler and a user does not absorb processor on a card through complicated and time-consuming cryptographic procedures.

One of the objectives in designing the protocol was to minimize quantity of data exchanged to meet requirements of wireless networks. Due to this a user can execute payment operations in a fast and cheap way. The desired properties of encryption and decryption are obtained by applying the elliptic curves cryptography (ECC). It was confirmed by researches and tests and why it one was used in our protocol.

Keywords: protocol, micropayment, macropayment, digital money, asymmetric cryptography, wireless networks.

INFORMACJE DLA AUTORÓW

Redakcja przyjmuje do publikowania prace oryginalne, przeglądowe i monograficzne wchodzące w zakres szeroko pojętej elektroniki. Ponieważ KWARTALNIK ELEKTRONIKI I TELEKOMUNIKACJI jest czasopismem Komitetu Elektroniki i Telekomunikacji Polskiej Akademii Nauk, w związku z tym na jego łamach znajdują się prace naukowe dotyczące podstaw teoretycznych i zastosowań z zakresu elektroniki, telekomunikacji, mikroelektroniki, optoelektroniki, radiotechniki i elektroniki medycznej.
Artykuły powinno charakteryzować oryginalne ujęcie zagadnienia, własna klasyfikacja, krytyczna ocena (teorii lub metod), omówienie aktualnego stanu, lub postępu danej gałęzi techniki oraz omówienie perspektyw rozwojowych. Artykuły publikowane w innych czasopismach nie mogą być kierowane do druku w Kwartalniku Elektroniki i Telekomunikacji w drugiej kolejności zgłoszenia.
Objętość artykułu nie powinna przekraczać 30 stron po około 1800 znaków na stronie, w tym rysunki i tabele.

Wymagania podstawowe.
Artykuły należy nadsyłać na wyraźnym, jednostronnym, czarno-białym wydruku komputerowym. Wydruk w formacie A4 powinien mieć znormalizowaną liczbę wierszy i znaków w wierszu (30 wierszy po 60 znaków w wierszu), w dwóch egzemplarzach, w języku polskim lub angielskim wybranym przez autora. Do wydruku powinna być dołączona dyskietka z elektronicznym tekstem artykułu. Preferowane edytory to WORD 6 lub 8.
Układ artykułu (w wersji podstawowej) musi być następujący:
- Tytuł.
- Autor (imię i nazwisko autora/ów).
- Miejsce pracy (nazwa instytucji, miejscowość, adres. + ew. adres elektroniczny (e-mail).
- Zwięzłe streszczenie powinno być w języku takim, w jakim jest pisany artykuł (wraz ze słowami kluczowymi).
- Tekst podstawowy powinien mieć następujący układ:
  1. WPROWADZENIE
  2. np. TEORIA
  3. np. WYNIKI NUMERYCZNE
  3.1. .....
  3.2. .....
  4. .....
  5. .....
  6. PODSUMOWANIE
  7. ew. PODZIĘKOWANIA
  8. BIBLIIOGRAFIA
- Układ streszczenia w dodatkowej wersji językowej powinien być następujący:
  AUTOR (inicjał imienia i nazwisko).
  TYTUŁ (w języku angielskim – o ile artykuł pisany jest w języku polskim i na odrwót).

Obszerne do 3600 znaków streszczenia (wraz z słowami kluczowymi) w języku:
  a. angielskim, gdy artykuł pisany jest w języku polskim.
  b. polskim, gdy artykuł pisany jest w języku angielskim.

Streszczenie to powinno pozwolić czytelnikowi na uzyskanie istotnych informacji zawartych w pracy. Z tego względu w streszczeniu tym mogą być cytowane numery istotnych wzorów, rysunków i tabel zawartych w podstawowej wersji językowej.
- Wszystkie strony muszą mieć numerację ciągłą.

Sposób pisania tekstu.
Tekst powinien być pisany bez używania wyróżnień, a w szczególności nie dopuszcza się spacjowania, podkreślania i pisania tekstu dużymi literami z wyjątkiem wyrazów, które umownie pisze się dużymi literami (np. FORTRAN). Proponowane wyróżnienia Autor może zaznaczyć w maszynopisie zwykłym ołówkiem za pomocą przyjętych znaków adjustacyjnych, np. podkreślenie linią przerywaną oznacza spacjowanie (rozstrzelenie), podkreślenie linią ciągłą – pogrubienie, podkreślenie wężykiem — kursywa.

Tekst powinien być napisany z podwójnym odstępem między wierszami, tytuły i podtytuły małymi literami. Marginesy z każdej strony powinny mieć około 35 mm. Wielkość czcionki wydruku powinna być zbliżona co najmniej co wielkości czcionki maszyny do pisania (minimum 12 punktów). Przy podziale pracy na rozdziały i podrozdziały cyfrowe ich oznaczenia nie powinny być większe niż II stopnia (np. 4.1.1.).

Sposób pisania tabel.
Tabele powinny być pisane na oddzielnych stronach. Tytuły rubryk pionowych i poziomych powinny być napisane małymi literami z podwójnym odstępem między wierszami. Przypisy (notki) dotyczące tabel należy pisać bezpośrednio pod tabelami. Tabele należy numerować kolejno liczbami arabskimi, u góry każdej tabeli podać tytuł dwujęzyczny. W pierwszej kolejności w podstawowej wersji językowej, a później w dodatkowej wersji językowej. Tabele umieścić na końcu maszynopisu. Przyjmowane są tabele algorytmów i programy na wydrukach komputerowych. W tym przypadku zachowany jest ich oryginalny układ. Tabele powinny być cytowane w tekście.

Sposób pisania wzorów matematycznych.
Rozmieszczenie znaków, cyfr, liter i odstępów powinno być zbliżone do rozmieszczenia elementów druku. Wskaźniki i wykładniki potęg powinny być napisane wyraźnie i być prawidłowo obniżone lub podwyższone w stosunku do linii wiersza podstawowego. Znaki nad literami i cyfram, całkami i in. symbolami (strzałki, linie, kropki, daszki) powinny być umieszczone dokładnie nad tymi elementami, do których się odnoszą. Numery wzorów cyframi arabskimi powinny być kolejne i umieszczone w nawiasach okrągłych z prawej strony. Nazwy jednotek, symbole literowe i graficzne powinny być zgodne z wytycznymi IEE (International Electronical Commision) oraz ISO (International Organization of Standarization).

Powołania.
Powołania na publikacje powinny być umieszczone na ostatnich stronach tekstu pod tytułem „Bibliografia", opatrzone numeracją kolejną bez nawiasów. Numeracja ta powinna być zgodna z odnośnikami w tekście artykułu. Przykłady opisu publikacji:

- periodycznej — 1. F. Valdoni: A new milimetre wave satelite. E.T.T. 1990, vol. 2, no 5, pp. 141–148
- nieperiodyczne — 2. K. Andersen: A resource allocation framework. XVI International Symposium, Stockholm (Sweden), may 1991, paper A 2,4
- książki — 3. Y.P. Tvidis: Operation and modeling of the MOS transistors. New York, McGraw-Hill, 1987, p. 553

Materiały ilustracyjne.
Rysunki powinny być wykonane wyraźnie, na papierze gładkim lub milimetrowym w formacie nie mniejszym niż 9 × 12 cm. Mogą być także w postaci wydruku komputerowego (preferowany edytor Corel Draw). Fotografie lub diapozytywy przyjmowane są raczej czarno-białe w formacie nie przekraczającym 10 × 15 cm. Na marginesie każdego rysunku i na odwrocie fotografii powinno być napisane ołówkiem imię i nazwisko Autora oraz skrót tytułu artykułu, do którego są przeznaczone oraz numer rysunku. Spis podpisów pod rysunki i fotografie powinien być umieszczony na oddzielnej stronie. Podpisy pod rysunkami (fotografiami) powinny być dwujęzyczne: w pierwszej kolejności w podstawowej wersji językowej, a później w dodatkowej wersji językowej. Rysunki powinny być cytowane w tekście.

Uwagi końcowe.
Na odrębnej stronie powinny być podane następujące informacje:
- adres do korespondencji z kodem pocztowym (domowy lub do miejsca pracy),
- telefon domowy i/lub do miejsca pracy,
- adres e-mailowy (jeśli autor posiada).

Autorowi przysługuje bezpłatnie 20 odbitek artykułu. Dodatkowe egzemplarze odbitek, lub cały zeszyt Autor może zamówić u wydawcy na własny koszt.

Autora obowiązują korekta autorska, którą powinien wykonać w ciągu 3 dni od daty otrzymania tekstu z Redakcji oraz zwrócić osobiście lub listownie pod adres Redakcji. Korekta powinna być naniesiona na przekazanych Autorowi szpaltach na marginesach ew. na osobnym arkuszu w przypadku uzupełnień tekstu większych niż dwa wiersze. W przypadku nie zwrócenia korekty w terminie, korektę przeprowadza Redakcja Techniczna Wydawcy. Redakcja prosi Autorów o powiadomienie ją o zmianie miejsca pracy i adresu prywatnego.

INFORMATION FOR AUTHORS OF K.E.T.

The editorial stuff will accept for publishing only original monographic and survey papers concerning widely understood electronics. Because of the fact that KWARTALNIK ELEKTRONIKI I TELEKOMUNIKACJI is a journal of the Committee for Electronics and Telecommunications of Polish Academy of Scince, it presents scientific works concerning theoretical bases and applications from the field of electronics, telecommunications, microelectronics, optoelectronics, radioelectronics and medical electronics.
Articles should be characterised by original depiction of a problem, its own classification, critical opinion (concerning theories or methods), discussion of an actual state or a progress of a given branch of a technique and discussion of development perspectives.
An article published in other magazines can not be submitted for publishing in K.E.T. TThe size of an article can not exceed 30 pages, 1800 character each, including figures and tables.

Basic requirements

The article should be submitted to the editorial stuff as a one side, clear, black and white computer printout in two copies. The article should be prepared in English or Polish. Floppy disc with an electronic version of the article should be enclosed. Preferred wordprocessors: WORD 6 or 8.
Layout of the article.

- Title.
- Author (first name and surname of author/authors).
- Workplace (institution, adress and e-mail).
- Concise summary in a language article is prepared in (with keywords).
- Main text with following layout:
  - Introduction
  - Theory (if applicable)
  - Numerical results (if applicable)
  - Paragraph 1
  - Paragraph 2
  - .....
  - .....
  - Conclusions
  - Acknowledgements (if applicable)
  - References
- Summary in additional language:
  - Author (firs name initials and surname)
  - Title (in Polish, if article was prepared in English and vice versa)
  - Extenstive summary, hawever not exceeching 3600 characters (along with keywords) in Polish, if artide wos prepared in English and vice versa). The summary should be prepared in a way allowing a reader to obtaoin essential information contained in the artide. For that reason in the summary author can place numbers of essential formulas, figures and tables from the article.

Pages should have continues numbering.

Main text

Main text can not contain formatting such as spacing, underlining, words written in capital letters (except words that are commonly written in capital letters). Author can mark suggested formatting with pencil on the margin of the article using commonly accepted adjusting marks.
Text should be written with double line spacing with 35 mro left and right margin. Titles and subtitles should be written with small letters. Titles and subtitles should be numberd using no mor than 3 levels (i.e. 4.1.1.).

Tables

Tables with their titles should be places on separate page at the end of the article. Titles of rows and columns should be written in small letters with double line spacing. Annotations concerning tables

should be placed directly below the table. Tables should be numbered with Arabic numbers on the top of each table. Table can consist algorithm and program listings. In such cases original layout of the table will be preserved. Table should be cited in the text.

Mathematical formulas

Characters, numbers, letters and spacing of the formula should be adequate to layout of main text. Indexes should be properly lowered or raised above the basic line and clearly written. Special characters such as lines, arrows, dots should be placed exactly over symbols which they ar attributed to. Formulas should be numbered with Arabic numbers place in brackets on the right side of the page. Units of measure, letter and graphic symbols should be printed according to requirements of IEE (International Electronical Commission) and ISO (International Organisation of Standardisation).

References

References should be placed at the and of the main text with th subtitle „References". Refrences should be numbered (without brackets) adequately to references placed in the text. Examples of periodical [@], non-periodical [2] and book [3] references:

1. F. Valdoni: A new milimetre wave satelite. E.T.T. 1990, vol. 2, no 5, pp. 141–148
2. K. Anderson: A resource allocation framework. XVI International Symposium (Sweden). May 1991. paper A 2.4
3. Y.P. Tvidis: Operation and modeling of the MOS transistors. Nw York. McGraw-Hill. 1987. p. 553

Figures

Figures should be clearly drawn on plain or milimetre paper in the format not smaller than 9 × 12 cm. Figures can be also printed (preferred editor – CorelDRAW). Photos or diapositives will be accepted in black and whit format not gratd than 10 × 15 cm. On th margin of each drawing and on th back side of each ohoto authorp name and abbreviation of the title of article should be placed. Figure's captures should be given in two longuaqes (first in the language the article is writtes in and then in additional language). Figure's captures should be also listed on separate page. Figures should be cited in the text.

Additional information

On the separate page following information should be placed:

- mailing address (home or office),
- phone (home or/and office),
- e-mail.

Author is entiled to free of charg 20 copies of articl. Additional copies or the whole magazine can be ordered at publisher at th one's expense.

Author is obliges to perform the author's correction, which should be accomplished within 3 days starting from the date of receiving of the text from the ditorial stuff. Corrected text should be return to the editorial stuff personally or by mail. Correction marks should be placed on th margin of copies received from the editorial stuff or if needed on separate pages. In the case when the correction is not returned in said time limit, correction will be performed by technical editorial stuff of the publisher.

In case of changing of workplace or home address Authors are asked to inform the editorial stuff.