Redukcja binarnych tablic decyzyjnych z wykorzystaniem automatycznego wnioskowania

Marian Adamski, Jacek Tkacz, Beata Wrzesińska

Streszczenie: W artykule przedstawiono sposób redukcji binarnych tablic decyzyjnych z wykorzystaniem symbolicznej metody komputerowego wnioskowania. Wszystkie przekształcenia formuł logicznych są przeprowadzane w zdaniowym rachunku sekwentów. Wykorzystano w tym celu program GENTZEN opracowany w Instytucie Informatyki i Elektroniki Uniwersytetu Zielonogórskiego. Metoda redukcji tablicy decyzyjnej jest ilustrowana przykładem minimalizacji słabo określonych funkcji logicznych wielu zmiennych. Opiera się ona na wyznaczaniu transwersali podzbiorów skończonego zbioru elementów symboliczną metodą przekształcania sekwentów z wykorzystaniem trzech reguł wnioskowania w monotonicznym rachunku zdaniowym.

Słowa kluczowe: tablice decyzyjne, funkcje logiczne słabo określone, automatyczne dowodzenie twierdzeń, rachunek sekwentów Gentzena, synteza logiczna.

1. WPROWADZENIE

W artykule przedstawiono sposób redukcji binarnych tablic decyzyjnych z wykorzystaniem systemu komputerowego dowodzenia twierdzeń w zdaniowym rachunku Gentzena [1, 4]. Binarna tablica decyzyjna przedstawia słabo określoną funkcję logiczną wielu zmiennych [6, 7]. Przykładową tablicę zaczerpnięto z książki [5], ze względu na zamieszczony w niej pełny i systematyczny opis wszystkich kroków minimalizacji funkcji i stopniowo uzyskiwanych rezultatów częściowych. Proponowany sposób minimalizacji symbolicznej postaci funkcji logicznej umożliwia w stosunkowo sprawny sposób otrzymywanie rezultatów dokładnych, wraz z formalnym dowodem ich gwarantowanej poprawności.

Za pomocą komputerowego systemu wnioskującego otrzymuje się, albo tylko jedno z pierwszych, najkorzystniejszych rozwiązań, albo na żądanie wszystkie dokładne rozwiązania. Badania eksperymentalne wykazały, że akceptowalne rozwiązania uzyskiwane są już w początkowej fazie komputerowego wnioskowania po przeszukaniu jednej z pierwszych gałęzi drzewa dowodu. Wnioskowanie gentzenowskie pozwala usprawnić proces redukcji poprzez wprowadzenie uprzednio sprecyzowanych elementów heurystyki, a nawet intuicji. Dodatkowo drzewo dowodu przeprowadzanego w głąb, jest na bieżąco redukowane (obcinane) metodą Thelena, co w znaczący sposób zmniejsza jego rozmiary [11].

W pracy celowo naśladowano sposób postępowania podczas manualnej minimalizacji funkcji logicznych słabo określonych przedstawiony w popularnych podręcznikach syntezy logicznej, między innymi w książkach [6, 7]. W ten sposób pokazano, że można pogodzić intuicję i przyzwyczajenia projektanta z formalnym projektowaniem metodą systematycznego dowodzenia twierdzeń [1].

Motywacją dla autorów jest przede wszystkim wypromowanie formalnej metody postępowania polegającej na przekształcaniu symbolicznym opisu specyfikacji na równoważny logicznie opis implementacji. Celem dalszych badań jest pełniejsza integracja metod komputerowego wnioskowania z metodami syntezy logicznej. Przykładowa specyfikacja kombinacyjnego układu cyfrowego przetwarzana jest na zbiór aksjomatów specyficznych i powiązana ze zdaniowym rachunkiem sekwentów Gentzena [1, 10]. Reguły wnioskowania Gentzena etykietujące wraz podsekwentami kolejno uzyskiwane wierzchołki drzewa wnioskowania świadczą o formalnej poprawności przekształceń wyrażeń logicznych.

Rezultaty otrzymane w sposób symboliczny mogą służyć do weryfikacji poprawności wyników różnorodnych programów heurystycznych z obszaru techniki cyfrowej i sztucznej inteligencji. W wielu przypadkach praktycznych możliwa jest ocena jakości uzyskanych heurystycznie rezultatów, na podstawie przeglądu wszystkich dokładnych rozwiązań.

Po niewielkich modyfikacjach opisane formalne procedury wnioskowania mogą być zastosowane do analizy danych w obszarze metod syntezy logicznej i sztucznej inteligencji [6, 7, 8, 9, 10, 11].

2. MONOTONICZNY RACHUNEK SEKWENTÓW

Monotoniczny rachunek sekwentów [2] jest podzbiorem ogólniejszego zdaniowego rachunku opisanego między innymi w pracach [1, 4, 11].

Sekwentem nazywa się uporządkowaną parę (Γ, Δ) skończonych zbiorów formuł $\Gamma=\{A1,A2,\ldots,Am\}$, $\Delta=\{B1,B2,\ldots,Bn\}$. Definiując sekwent (1)

$$A1, A2, \ldots, Am \mid\!- B1, B2, \ldots, Bn \qquad (1)$$

zakłada się, że jest on spełniony dla tych wartościowań, dla których formuła (2)

$$(A1 * A2 * \ldots * Am) \text{ -> } (B1 + B2 + \ldots + Bn) \qquad (2)$$

rozpatrywana w klasycznym rachunku zdań jest spełnialna. W przyjętej notacji symbol '->' oznacza implikację, symbol '+' dysjunkcję, natomiast symbol '*' jest symbolem koniunkcji. We wnioskowaniu wykorzystywane są reguły: lewo i prawostronnej eliminacji spójników koniunkcji (3) i dysjunkcji (4).

$$\frac{\Lambda \mid -\Theta,\Phi*\Psi,\Gamma}{\Lambda \mid -\Theta,\Phi,\Gamma \quad \Lambda \mid -\Theta,\Psi\Gamma} \qquad \frac{\Theta,\Phi*\Psi,\Gamma \mid -\Pi}{\Theta,\Phi,\Psi,\Gamma \mid -\Pi} \tag{3}$$

$$\frac{\Theta,\Phi+\Psi,\Gamma \mid -\Pi}{\Theta,\Phi,\Gamma \mid -\Pi \quad \Theta,\Psi,\Gamma \mid -\Pi} \qquad \frac{\Lambda \mid -\Theta,\Phi+\Psi,\Gamma}{\Lambda \mid -\Theta,\Phi,\Psi,\Gamma} \tag{4}$$

Do eliminacji zbędnych formuł logicznych w procesie wnioskowania wykorzystuje się regułę cięcia (5), jako uogólnioną formę pochłaniania w procesie rezolucji.

$$\frac{\Theta,\Phi,\Gamma \mid -\Psi \quad \Theta,\Gamma \mid -\Psi,\Phi}{\Theta,\Gamma \mid -\Psi} \tag{5}$$

W monotonicznym rachunku sekwentów przyjmuje się, że w formułach logicznych zawartych w sekwencie nie występują równocześnie te same literały w postaci afirmacji i negacji.

System wnioskowania GENTZEN uzupełniony jest mechanizmem wyszukiwania podsekwentów już zdominowanych lub tych, które zostałyby zdominowane przez kolejno powstające podsekwenty. Są one natychmiastowo wycinane z drzewa dowodu. Pełny opis systemu komputerowego wnioskowania w zdaniowym rachunku sekwentów można znaleźć w pracy doktorskiej [11]. Po niewielkiej modyfikacji system GENTZEN umożliwi w pełni automatyczną redukcję nadmiarowego zbioru zmiennych wejściowych funkcji skończonych, stąd może zostać zastosowany także podczas redukcji wielowartościowych tablic decyzyjnych.

2. WYZNACZANIE TRANSWERSALI Z WYKORZYSTANIEM KOMPUTEROWEGO WNIOSKOWANIA METODĄ GENTZENA

Pojęcie transwersali wykorzystywane jest w kombinatoryce i często powiązane z teorią hipergrafów [3, 8].

Wykorzystanie transwersali do wyznaczania sekwentów rozróżnialności szczegółowo omówiono w rozdziale 3. Niech A1 jest zbiorem literałów *{x1, /x2, x4, x5, x7}*. Zbiór A1 został podzielony na następujące podzbiory: *{/x4}, {x1 , x6 , /x7}, {x1 , /x2 , x5 , x6}*. Transwersalą rozpatrywanej rodziny podzbiorów jest podzbiór zbioru A1, powstały poprzez wybranie po jednym elemencie z każdego z podzbiorów. Jedną z minimalnych transwersali o najmniejszej mocy jest transwersala *I1={x1,/x4}*. Wszystkie minimalne transwersale I1÷I4 wyznaczono przekształcając sekwent:

$$/x4,(x1 + x6 + /x7),(x1 + /x2 + x5 + x6) \mid\text{-}A1 \tag{6}$$

do dualnego sekwentu:

$$(x1*/x4),(/x4*x6),(/x2*/x4*/x7),(/x4,x5,/x7) \mid\text{-}A1 \tag{7}$$

Sposób wykorzystania transwersali do wybierania niezbędnych implikantów pokazano w rozdziale 4. Niech A jest zbiorem wszystkich rozpatrywanych implikantów *{I1,I2,...,I22}*. Rozróżnia się następujące ich podzbiory *A1={I1, I2 , I3, I4}, A2={I1, I2, I5, I6, I7, I8, I9, I10}, A3={I5, I6, I11, I12, I13, I14, I15}, A4={I2, I7, I9, I10, I16, I17, I18, I19, I20}, A5={I21, I22}*. Tylko jeden element z każdego podzbioru jest niezbędny. Transwersalami są między innymi podzbiory *{I1, I5, I16}* oraz *{I1, I15, I17}*. Szczegółowe przykłady wyznaczania transwersali metodą wnioskowania gentzenowskiego zostały zamieszczone w rozdziale 4. Złożoność obliczeniowa wnioskowania redukcyjnego dla stosowanych tutaj sekwentów monotonicznych, zawierających *m* literałów, mierzona liczbą węzłów w drzewie dowodu ma charakter wielomianowy [2, 4].

3. WYZNACZANIE SEKWENTÓW ROZRÓŻNIALNOŚCI

Jako przykład rozpatrywana jest tablica decyzyjna (Tabela 1) opisująca funkcję logiczną Z = *f(x1, x2, ..., x7).*

Tabela 1.
Przykładowa tablica decyzyjna [4]

	X1	x2	x3	x4	x5	x6	x7	Z
1	1	0	0	0	1	1	0	1
2	1	0	0	0	0	1	1	1
3	1	0	0	1	0	0	0	1
4	0	1	1	0	0	1	1	1
5	1	0	1	0	0	0	0	1
6	1	0	0	1	1	1	0	0
7	0	0	0	0	1	0	1	0
8	0	1	0	1	1	0	1	0
9	0	0	1	0	1	0	1	0
10	0	1	1	0	0	0	0	0
11	0	1	0	0	0	0	0	0

Ze względu na ograniczone ramy w artykule nie omówiono sposobu uzyskiwania sekwentów rozróżnialności metodą symboliczną w sposób automatyczny. Zakłada się, że znany jest hipergraf rozróżnialności otrzymany sposobem przestawionym w pracy [9]. Przez sekwent rozróżnialności dla *i*-tego wiersza tablicy decyzyjnej rozumiany będzie sekwent *Ai*, który jest spełniony dla wartościowania zmiennych przedstawionego wierszem *i*. Literały występujące w poszczególnych sekwentach *Ai, i=1,...,5* nie występują równocześnie w postaci afirmacji i negacji tej samej formuły elementarnej. Sekwent rozróżnialności może występować w dwóch komplementarnych, dualnych formach. Transformacja sekwentu (6) na dualny sekwent (7) jest równoważna przekształceniu koniunkcji trzech dysjunkcji na postać dysjunkcji czterech koniunkcji.

Tabela 2.
Sekwenty rozróżnialności

/x4,(x1 + x6 + /x7),(x1 + /x2 + x5 + x6)|-A1;
(/x4+/x5+ x7),(x1 + /x5 + x6),(x1 + /x2 + x6 + x7)|-A2;
(/x5 + /x6),(x1 + x4 + /x5 + /x7),(x1 + /x2 + /x5 + /x7),(x1 + /x2 + x4)|-A3;
(/x1 + x2 + x3 + /x4 + /x5 + x7),(x3 + /x4 + /x5 + x6),(x2 + /x5 + x6),(x6 + x7)|-A4;
(x3 + /x4 + /x5 + /x6),(x1 + /x5 + /x7),(x1 + /x2)|-A5;

4. REDUKCJA NADMIAROWYCH SEKWENTÓW – IMPLIKANTÓW

Koniunkcje _I1=x1*/x4, I2=/x4*x6, I3=/x2*/x4*/x7, I4=/x4*x5*/x7_ występujące po lewej stronie sekwentu (7) należą do podzbioru _A1={I1, I2, I3, I4}_ wszystkich implikantów rozpatrywanej funkcji _Z_. Wyszukując również inne podzbiory _A2-A5_ zbioru implikantów _{I1-I22}_, otrzymano rezultaty zamieszczone w tabeli 3. Redukcja powtarzających się sekwentów została przeprowadzone w sposób automatyczny w systemie wnioskowania GENTZEN. Sekwenty elementarne opisujące rozpatrywane implikanty funkcji logicznej Z zostały powiązane z implikantami _I={I1, I2, ..., I21, I22}_, a następnie pogrupowane zgodnie z podzbiorami _A1-A5_.

Tabela 3.
Sekwenty pokrywające implikanty funkcji Z

Grupa	Implikanty w postaci sekwentów	Symbole implikantów
A1= {I1, I2, I3, I4}	x1,/x4|-A1;	I1
	/x4,x6|-A1;	I2
	/x2,/x4,/x7|-A1;	I3
	/x4,x5,/x7|-A1;	I4
A2= {I1, I2, I5, I6, I7, I8, I9, I10}	x1,/x4|-A2;	I1
	/x4,x6|-A2;	I2
	x1,/x5|-A2;	I5
	/x2,/x5|-A2;	I6
	/x5,x6|-A2;	I7
	x1,x7|-A2;	I8
	/x5,x7|-A2;	I9
	x6,x7|-A2;	I10
A3= {I5, I6, I11, I12, I13, I14, I15}	x1,/x5|-A3;	I5
	/x2,/x5|-A3;	I6
	x4,/x5|-A3;	I11
	x1,/x6|-A3;	I12
	/x2,x4,/x6|-A3;	I13
	/x2,/x6,/x7|-A3;	I14
	x4,/x6,/x7|-A3;	I15
A4= {I2, I7, I9, I10, I16, I17, I18, I19, I20}	/x1,x6|-A4;	I16
	x2,x6|-A4;	I17
	x3,x6|-A4;	I18
	/x4,x6|-A4;	I2
	/x5,x6|-A4;	I7
	x2,x3,x7|-A4;	I19
	x2,/x4,x7|-A4;	I20
	/x5,x7|-A4;	I9
	x6,x7|-A4;	I10
A5= {I2, I7, I16, I17, I18}	x1,x3|-A5;	I21
	/x2,x3,/x7|-A5;	I22
	x1,/x4|-A5;	I1
	/x2,/x4,/x7|-A5;	I3
	x1,/x5|-A5;	I5
	/x2,/x5|-A5;	I6
	x1,/x6|-A5;	I12
	/x2,/x6,/x7|-A5;	I14

Metoda może być wykorzystywana do oceny jakości wyników redukcji tablic decyzyjnych wykonywanych manualnie lub komputerowo, poprzez sterowany przegląd akceptowalnych, alternatywnych wariantów.

5. WYZNACZANIE NAJKORZYSTNIEJSZEGO PODZBIORU IMPLIKANTÓW

Wybranie najkorzystniejszego, minimalnego podzbioru implikantów spośród wszystkich rozpatrywanych implikantów selekcjonowanych z nadmiarowego zbioru polega na wyszukaniu transwersali podzbiorów _A1={I1, I2, I3, I4}, ..., A5={I1, I3, I5, I6, I12, I14, I21, I22}_, spełniającej równocześnie inne dodatkowe kryteria.

Tabela 4.
Sekwent wyznaczający tranwersale pokrycia

Z |- I1*I2*I3*I4 , I1*I2*I5*I6*I7*I8*I9*I10, I5*I6*I11*I12*I13*I14*I15, I16*I17*I18*I2*I7*I19*I20*I9*I10, I21*I22*I1*I3*I5*I6*I12*I14;

Wyznaczając minimalne transwersale podzbiorów _A1-A5_, posłużono się sekwentem, zamieszczonym w tabeli 4. Poprzez redukcję tego sekwentu, a tym samym po wyszukaniu wszystkich minimalnych pokryć funkcji Z implikantami ze zbioru _I_, otrzymano aż 208 pojedynczych sekwentów (Tabela 5).

Wyznaczone minimalne symboliczne reprezentacje funkcji Z zawierają po 2, 3 lub nawet 4 implikanty. Uzyskane komputerowo rezultaty minimalizacji funkcji zamieszczono zgodnie z kolejnością ich otrzymywania. Po podstawieniu za symbole implikantów odpowiadających im formuł logicznych (tabela 3) bez zbędnych przekształceń wprost uzyskuje się postać dysjunkcyjno-koniunkcyjną. Wyliczone jako pierwsze dwie symboliczne reprezentacje funkcji Z zamieszczono w tabeli 6.

Tabela 5.
Akceptowalne wyniki pokrycia funkcji implikantami

|-I1, I16, I5;	|-I1, I17, I5;	|-I1, I18, I5;	|-I1, I5, I7;
|-I1, I19, I5;	|-I1, I20, I5;	|-I1, I5, I9;	|-I1, I10, I5;
|-I1, I16, I6;	|-I1, I17, I6;	…………..	|-I10,I15,I22,I4;

Tabela 6.
Dwa pierwsze wyniki minimalizacji funkcji logicznej Z

|-I1,I16,I5;	Z = x1*/x4 + /x1*x6 + x1*/x5
|-I1,I17,I5;	Z = x1*/x4 + x2*x6+ x1*/x5

7. WYNIKI EKSPERYMENTÓW

Najkorzystniejsze wyniki minimalizacji ze względu na liczbę implikantów oraz zmiennych zamieszczono w tabeli 7. Biorąc pod uwagę minimalną liczbę implikantów wyznaczono cztery minimalne postacie funkcji logicznej _Z_. Następnie jako dodatkowe kryterium przyjęto liczbę różnych literałów wchodzących do wyrażeń logicznych eksponując symboliczną postać funkcji _w62_: Z = _f(x1 ,x4 ,x6)_ wykorzystującą tylko trzy z pośród siedmiu zmiennych. Metodą symboliczną eliminuje się więc zbyteczne (pozorne) argumenty funkcji.

Pełne drzewo dowodu zawiera 208 liści. Warto zauważyć, że pierwsze akceptowalne rozwiązanie (trzy implikanty) skojarzone z liściem _w1_ uzyskano już po ośmiu krokach wnioskowania. Wszystkie korzystne rozwiązania ze względu na minimalną liczbę implikantów równą dwa, które są zamieszczone w tabeli 7 przypisane są stosunkowo wcześnie liściom o numerach: w57, w58, w62 i w67. Jedno z najgorszych rozwiązań uzyskano w ostatnim 208 liściu drzewa dowodu.

Tabela 7.
Najkorzystniejsze wyniki minimalizacji funkcji Z

(w57)	|-I2,I5;	Z = /x4*x6 + x1*/x5
(w58)	|-I2,I6;	Z = /x4*x6 + /x2*/x5
(w62)	|- I2,I12;	Z = /x4*x6 + x1*/x6
(w67)	|- I2,I14;	Z = /x4*x6 + /x2*/x6*/x7

Korzystne rozwiązanie *w57* jest całkowicie zgodne z wynikiem minimalizacji przeprowadzonej w książce [4]. Identyczny wynik uzyskano minimalizując funkcję Z z wykorzystaniem programu ESPRESSO. Najlepszym rozwiązaniem zarówno ze względu na najmniejszą liczbę implikantów, jak i liczbę literałów jest jednak formuła *w62: Z = f (x1 ,x4 ,x6) = /x4*x6 + x1*/x6*.

Podatność algorytmu na heurystykę związaną z porządkiem formuł w sekwencie sprawdzono porządkując podzbiory *A1-A5* zgodnie z ich mocą. Umowną podstawą oceny jest numer liścia, w którym uzyskano ten sam wynik, co stosując program ESPRESSO.

Tabela 8.
Podatność algorytmu na heurystykę

Porządek	Liczba węzłów drzewa	Pozycja w drze wie	Pozycja na liście rozw.	Czas uzyskania |-I2,I5
Przypadkowy	**2519**	**302**	**57**	**0,654s**
Niemalejący	**2520**	**161**	**57**	**0,375s**
Nierosnący	**1954**	**706**	**85**	**1,437s**

7. PODSUMOWANIE I WNIOSKI

Na podstawie konkretnego sposobu redukcji binarnej tablicy decyzyjnej, pokazano użyteczność pełniejszego zintegrowania metod syntezy logicznej z formalnymi metodami automatycznego wnioskowania metodą Gentzena. Mimo wyraźnej przydatności obliczeń symbolicznych w procesie minimalizacji funkcji logicznych bezsensowne jest konkurowanie w tym obszarze z uznanym na świecie heurystycznym programem ESPRESSO, jednak jak pokazano, nie zawsze dającym najlepszy wynik. Zaletą proponowanej procedury jest możliwość otrzymania pełnego zbioru akceptowalnych rezultatów wraz z ich komputerowym dowodem poprawności w logice sekwentów.

LITERATURA

[1] Adamski M., *Projektowanie układów cyfrowych systematyczną metodą strukturalną*. Monografie Nr 49. Wydawnictwo Wyższej Szkoły Inżynierskiej w Zielonej Górze, 1990, ISSN 0239-7390

[2] Atserias, A., Galesi, N., and Pudlák, P. 2002. *Monotone simulations of non-monotone proofs*. J. *Comput. Syst.* Sci. 65, 4 (Dec. 2002), 626-638.

[3] Eiter, T., Gottlob, G., and Makino, K. 2002. *New results on monotone dualization and generating hypergraph transversals*. Proc. of the 32th ACM Symp. STOC '02, 14-22.

[4] Gallier J.H., *Logic for computer science*, Foundations of Automatic Theorem Proving. Harper & Row, (1986).

[5] Gurtowcew L, Petrenko A.F., Czapenko W.P., *Logiczeskoje projektirowanie ustrojstw awtomatiki*. Izdatielstwo Zinatie, Riga 1978.

[6] Łuba T., *Synteza układów logicznych*, Wyższa Szkoła Informatyki Stosowanej i Zarządzania, Warszawa, 2000.

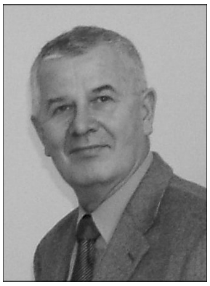
[7] Misiurewicz P., Perkowski M. *Teoria automatów*. Wydawnictwa Politechniki Warszawskiej, 1973

[8] Sapiecha P., *Algorytmy syntezy funkcji relacji boolowskich w aspekcie metod reprezentacji kompresji danych*. Rozprawa doktorska. Politechnika Warszawska, Wydział Elektroniki i Technik Informacyjnych, Warszawa, 1998.

[9] Sapiecha P., Pleban M., Łuba T, Zbierzchowski Dekompozycja funkcji i relacji boolowskich w syntezie logicznej i analizie danych. *Kwartalnik Elektroniki i Telekomunikacji*, 2000Vol.46(No.3)

[10] Tkacz J., Adamski M., Projektowanie sekwencyjnych układów cyfrowych z wykorzystaniem logiki sekwentów Gentzena, *Przegląd Elektrotechniczny*, 2009, nr 7, s. 196-199.

[11] Tkacz J., *Projektowanie układów sterowania binarnego wspomagane automatycznym wnioskowaniem Gentzena*, Rozprawa doktorska. Uniwersytet Zielonogórski, Wydział Elektrotechniki, Informatyki i Telekomunikacji, Zielona Góra, 2008.



prof. dr hab. inż. Marian Adamski
Uniwersytet Zielonogórski
Wydział Elektrotechniki, Informatyki i Telekomunikacji
Instytut Informatyki i Elektroniki

ul. Podgórna 50
65-246 Zielona Góra
tel.: 68 328 22 19
e-mail: M.Adamski@iie.uz.zgora.pl

dr inż. Jacek Tkacz
Uniwersytet Zielonogórski
Wydział Elektrotechniki, Informatyki i Telekomunikacji
Instytut Informatyki i Elektroniki

Podgórna 50
65-246 Zielona Góra
tel.: 68 328 25 26
e-mail: J.Tkacz@iie.uz.zgora.pl

inż. Beata Wrzesińska (studentka)
Uniwersytet Zielonogórski
Wydział Elektrotechniki, Informatyki i Telekomunikacji
Instytut Informatyki i Elektroniki

Podgórna 50
65-246 Zielona Góra
tel.: 68 328 25 26
e-mail:b.wrzesinska@poczta.onet.pl