

Bernard WYRWOŁ

POLITECHNIKA ŚLĄSKA, INSTYTUT ELEKTRONIKI

Sprzętowa implementacja algorytmów dekompozycji lingwistycznej opartych na podziale bazy wiedzy w układzie FPGA

Dr inż. Bernard WYRWOŁ

Ukończył studia na Wydziale Automatyki, Elektroniki i Informatyki Politechniki Śląskiej. Pracę doktorską obronił w 2004 roku. Jest adiunktem w Zakładzie Urządzeń Cyfrowych i Mikroprocesorowych Instytutu Elektroniki Politechniki Śląskiej. Jego zainteresowania naukowe obejmują zagadnienia związane z zastosowaniem i sprzętową realizacją systemów wnioskowania przybliżonego, układami logiki programowalnej, językami opisu sprzętu oraz systemami wieloprocessorowymi.

e-mail: Bernard.Wyrwol@polsl.pl



Streszczenie

Układowe realizacje systemów wnioskowania przybliżonego wymagają często znacznych nakładów. Zmniejszenie ich jest możliwe poprzez zastosowanie metody dekompozycji Gupty i przedstawieniu systemu jako struktury hierarchicznej. W celu wyeliminowania jej niekorzystnych własności konieczny jest wstępny podział bazy wiedzy. Zaproponowana została metoda najlepszego wyboru wykorzystująca wybrane algorytmy podziału, zaimplementowana w sprzętowym systemie wnioskowania przybliżonego FPGA-FIS.

Słowa kluczowe: funkcja przynależności, reguła rozmyta, reguła sprzeczna, relacja rozmyta, baza wiedzy, dekompozycja relacyjna, dekompozycja lingwistyczna, system regułowy FITA, system relacyjny FATI, wnioskowanie przybliżone, FPGA.

Hardware implementation of linguistic de-composition algorithms based on partitioning the knowledge base in the FPGA chip

Abstract

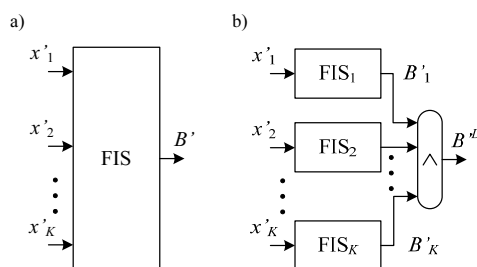
The hardware cost of a fuzzy inference system can be reduced using the Gupta's relational decomposition technique [1]. The system can be represented as a hierarchical architecture that comprises a set of Single Input Single Output subsystems (Fig. 1). The decomposition has some disadvantages, computation of the global relation \mathcal{R} is an extremely time-consuming process and a large memory is necessary to store it. They can be eliminated if projection is expanded on linguistic level and decomposition is used for the knowledge base (1), (Fig. 2) [2]. The projection operation (on relational or linguistic level) in some cases can lead to inevitable loss of information because of its approximate nature [3]. To avoid the inference error (the output result is more fuzzy than that obtained in the classical system architecture (3)) methods for partitioning (5) the knowledge base $KB[Y, X_K, \dots, X_1]$ into p subbases without inconsistent rules (4) are proposed [4]. In Section 3 the methods based on partitioning towards a defined input linguistic variable (Fig. 3) and elimination of the inconsistent rules (Fig. 4) are described [5, 6]. The algorithms are simple and fast but the results are not optimal in all cases (hardware cost depends on the number of subsystems p , Tab. 1). Thus, the method of the best choice is proposed and implemented in the FPGA fuzzy inference system as a DMU (Decomposition Management Unit) module (Fig. 6).

Keywords: membership function, fuzzy rule, inconsistent rule, fuzzy relation, knowledge base, relational decomposition, linguistic decomposition, First Inference Then Aggregation system (FITA), First Aggregation Then Inference system (FATI), fuzzy inference, FPGA.

1. Wstęp

Systemy wnioskowania przybliżonego, ze względu na sposób reprezentacji wiedzy w systemie oraz metodę wyznaczania wyniku, podzielić można na regułowe FITA (ang. *First Inference then Aggregate*) oraz relacyjne FATI (ang. *First Aggregate then Inference*) [7]. W pierwszej wiedzy reprezentowana jest przez

zbiór reguł rozmytych *Jeżeli-To*, natomiast w drugim reguły te przekształcane są do postaci funkcji tzw. relacji rozmytej, której argumenty zwykle mają charakter nierozmyty, natomiast wartości – rozmyte [8, 9]. Praktyczne realizacje tych systemów wymagają znacznych zasobów sprzętowych (w szczególności dla systemów FATI). W celu ich obniżenia system wnioskujący o K wejściach (rys. 1a) można przedstawić w postaci struktury hierarchicznej (rys. 1b) złożonej z K podsystemów rozmytych SISO (ang. *Single Input Single Output*) o takiej samej architekturze, ale różniących się zawartością swoich baz wiedzy, w oparciu o które przeprowadzany jest proces wnioskowania [2, 4].



rys. 1. Logiczna architektura systemu wnioskowania przybliżonego: a) klasyczna, b) hierarchiczna

Fig. 1. Logical architecture of the fuzzy inference system: a) classical, b) hierarchical

Jedną z metod pozwalającą przekształcić pierwotną bazę wiedzy do postaci baz wiedzy opisujących zachowanie podsystemów rozmytych w strukturze hierarchicznej jest technika dekompozycji zaproponowana przez Guptę [1]. Wykorzystana ona została w realizacji wyłącznie systemów FATI, a oparta została na tworzeniu subrelacji $\mathcal{R}_K \dots \mathcal{R}_1$ przy wykorzystaniu operacji projekcji globalnej relacji rozmytej \mathcal{R} . Charakteryzuje się ona znacznymi nakładami zarówno sprzętowymi jak i obliczeniowymi związanymi z koniecznością wyznaczenia relacji rozmytej \mathcal{R} , niezbędnej do utworzenia subrelacji $\mathcal{R}_K \dots \mathcal{R}_1$, a nie wykorzystywanej w czasie normalnej pracy systemu tj. podczas procesu wnioskowania.

Ze względu na to, że globalna relacja rozmyta \mathcal{R} tworzona jest na podstawie zawartości bazy wiedzy regułowego systemu wnioskującego FITA $KB[Y, X_K, \dots, X_1]$, gdzie Y, X_K, \dots, X_1 są odpowiednio wyjściową i wejściowymi zmiennymi lingwistycznymi, zaproponowana została metoda, w której to procesowi dekompozycji poddawana jest nie relacja, a baza wiedzy

$$KB_i[Y, X_i] = \text{proj}_{X_K, \dots, X_{i+1}, X_{i-1}, \dots, X_1} KB[Y, X_K, \dots, X_1], \quad (1)$$

gdzie *proj* jest operacją projekcji przeniesioną na poziom lingwistyczny [2]. Podejście to ma dwie podstawowe zalety: nie jest wymagane obliczanie globalnej relacji rozmytej \mathcal{R} , a wyniki uzyskane po przeprowadzeniu dekompozycji lingwistycznej mogą posłużyć zarówno do realizacji regułowego, relacyjnego lub regułowo-relacyjnego systemu wnioskującego o strukturze hierarchicznej.

Metoda dekompozycji Gupty nie jest pozbawiona wad. Operacja projekcji, niezależnie od tego czy zastosowana na poziomie relacyjnym czy też lingwistycznym, powoduje utratę informacji z pierwotnej bazy wiedzy podczas przenoszenia jej i konwersji z systemu o klasycznej architekturze do podsystemów SISO w systemie o architekturze hierarchicznej. Z tego powodu wyniki wnioskowania B' , dla tych samych wartości wielkości wejściowych x'_K, \dots, x'_1 , uzyskiwane z obydwu systemów mogą się różnić. Tę niekorzystną własność można całkowicie wyeliminować.

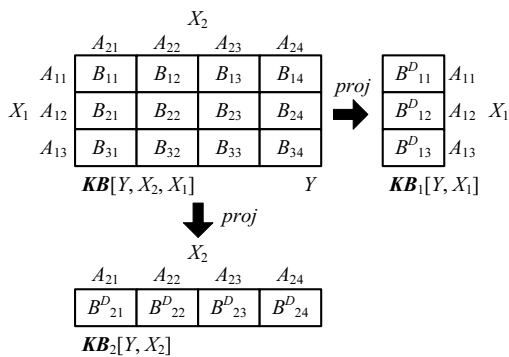
wać dokonując wstępnego podziału pierwotnej bazy wiedzy na bazy wiedzy o tym samym wymiarze, dla których to dekompozycja będzie operacją całkowicie bezstratną [2, 4, 5, 6].

2. Dekompozycja lingwistyczna

Dekompozycja lingwistyczna bazuje na operacji projekcji (1), która jest odpowiednikiem operacji projekcji wielowymiarowych relacji rozmytych [2, 4]. Zastosowana w odniesieniu do bazy wiedzy systemu o architekturze klasycznej prowadzi do utworzenia baz wiedzy podsystemów SISO $KB_i[Y, X_i]$ ($i=1, \dots, K$), w których to eliminuje się wszystkie zmienne lingwistyczne wejściowe z bazy wiedzy $KB[Y, X_K, \dots, X_1]$ za wyjątkiem zmiennej X_i . W wyniku projekcji lingwistycznej nie ulegają zmianie poprzedniki reguł, natomiast wartości lingwistyczne następników reguł nowo tworzonej bazy wiedzy otrzymuje się poprzez kombinację następników pierwotnej bazy wiedzy zgodnie z zależnością

$$B_{ij}^D = \bigvee_{j_k=1}^{N_K} \dots \bigvee_{j_{i+1}=1}^{N_{i+1}} \bigvee_{j_{i-1}=1}^{N_{i-1}} \dots \bigvee_{j_1=1}^{N_1} B_{j_K \dots j_1} \tag{2}$$

gdzie N_K, \dots, N_1 określają liczbę wartości lingwistycznych jakie może przyjmować każda zmienna lingwistyczna wejściowa X_K, \dots, X_1 , natomiast \bigvee oznacza operator sumy typu MAX [1, 7, 8]. Graficzna ilustracja dekompozycji, dla przykładowej bazy wiedzy, została pokazana na rys. 2.



Rys. 2. Dekompozycja lingwistyczna
Fig. 2. Linguistic decomposition

Dla pewnej klasy systemów wnioskujących, w oparciu o bazy wiedzy $KB_i[Y, X_i]$, nie jest możliwe odtworzenie bazy wiedzy $KB[Y, X_K, \dots, X_1]$ systemu pierwotnego [5]. W tym przypadku wyniki wnioskowania uzyskiwane z systemu klasycznego i hierarchicznego mogą się różnić. Porównując zawartość pierwotnej i odtworzonej bazy wiedzy można zauważyć, że niektóre z reguł ulec mogą zdeformowaniu tworząc nowe reguły $R^{ce}_{j_K \dots j_1}$ ($j_K=1, \dots, N_K; \dots; j_1=1, \dots, N_1$), w których wartość lingwistyczną następnika opisana jest zbiorem rozmytym

$$B_{j_K \dots j_1}^{ce} = B_{j_K \dots j_1} \bigvee \left(\bigvee_{j_k=1}^{N_K} \dots \bigvee_{j_1=1}^{N_1} (B_{j_K \dots j_2 j_1} \wedge \dots \wedge B_{j_K \dots j_2 j_1}) \right), \tag{3}$$

gdzie N_K, \dots, N_1 określają liczbę wartości lingwistycznych jakie mogą przyjmować zmienne lingwistyczne wejściowe X_K, \dots, X_1 . Jeżeli wartość wyrażenia (3), występująca w nawiasie, jest różna od zera, wówczas reguła może zafalszować wynik wnioskowania (dla rozpatrywanej reguły $R_{j_K \dots j_1}$ wszystkie pozostałe reguły, dla których iloczyn ten jest niezerowy określone są jako sprzeczne [2, 5]). Rozpatrując bazy wiedzy $KB_1[Y, X_1]$ i $KB_2[Y, X_2]$ pokazane na rys. 2, odtworzona na ich podstawie baza wiedzy $KB^{ce}[Y, X_2, X_1]$ zawiera m.in. regułę R^{ce}_{23} , której następnik (przy założeniu, że iloczyny odpowiednich zbiorów rozmytych występujących

w wierszu A_{12} i kolumnie A_{23} bazy wiedzy są zbiorami pustymi, za wyjątkiem iloczynu zbiorów B_{13} i B_{21}) wyrazić można jako

$$B_{23}^{ce} = B_{23} \vee (B_{13} \wedge B_{21}). \tag{4}$$

Reguły, w których następniki opisane są za pomocą zbiorów rozmytych B_{13} i B_{21} , określane są jako sprzeczne w stosunku do rozpatrywanej reguły R_{23} [6]. Przeprowadzając proces wnioskowania w oparciu o zniekształcone reguły zawarte w bazach wiedzy podsystemów SISO struktury hierarchicznej, uzyskuje się wynik zniekształcony, przy czym może on charakteryzować się zwiększoną rozmytością w porównaniu z wynikiem otrzymywanym w systemie o strukturze klasycznej [3, 5].

3. Metody podziału bazy wiedzy

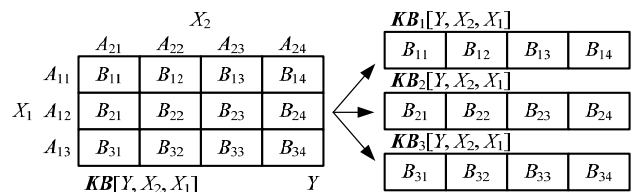
Wyeliminowanie zniekształcania wyniku wnioskowania w systemie o strukturze hierarchicznej można uzyskać poprzez usunięcie reguł sprzecznych z pierwotnej bazy wiedzy, a dopiero w następnym etapie zastosować technikę dekompozycji Gupty. Aby w wyniku tej operacji zachowanie systemu nie uległo zmianie pierwotną bazę reguł należy przedstawić jako złożenie p baz reguł

$$KB[Y, X_K, \dots, X_1] = \bigvee_p KB_p[Y, X_K, \dots, X_1], \tag{5}$$

z których każda może zawierać wszystkie lub tylko wybrane reguły $R_{j_K \dots j_1}$ ($j_K=1, \dots, N_K; \dots, j_1=1, \dots, N_1$) z podstawowej bazy wiedzy [2, 5]. Należy spełnić tutaj warunek, aby każda z reguł występowała przynajmniej raz w jednej z baz wiedzy $KB_p[Y, X_K, \dots, X_1]$. Powyższe działanie opisuje dekompozycję systemu wnioskującego na podsystemy o tej samej liczbie wejść i wyjść, ale o różniących się zawartościach baz reguł. Dla tak utworzonych baz wiedzy zastosowanie techniki dekompozycji Gupty na poziomie lingwistycznym nie powoduje utraty informacji, a tym samym wynik wnioskowania uzyskany z systemu o strukturze hierarchicznej jest dokładnie taki sam jak dla struktury klasycznej.

Kluczowym zagadnieniem jest tutaj optymalny podział pierwotnej bazy wiedzy tj. taki podział, który przy najmniejszej możliwie liczbie nowo utworzonych baz wiedzy $KB_p[Y, X_K, \dots, X_1]$ zapewni w każdej z nich brak reguł sprzecznych tzn. suma iloczynów zbiorów rozmytych występujących w nawiasie równania (3) jest zbiorem $B_{j_K \dots j_1}$ lub zbiorem pustym. Od wartości współczynnika p zależy nakład sprzętowy, którego zwiększenie jest tutaj nieuniknione [2].

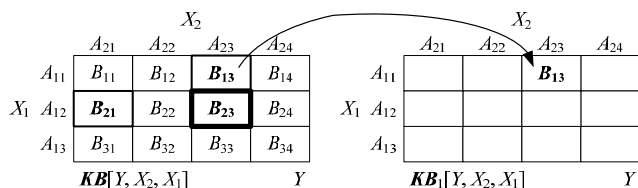
W [4] zaproponowana została metoda oparta na podziale pierwotnej bazy wiedzy względem wyróżnionej zmiennej wejściowej, której graficzna ilustracja przedstawia rys. 3.



Rys. 3. Metoda podziału względem zmiennej lingwistycznej wejściowej
Fig. 3. Partitioning method towards a defined input linguistic variable

Inna metoda podziału oparta została na przeszukiwaniu pierwotnej bazy wiedzy, odnajdywaniu reguł sprzecznych i ich przenoszeniu do nowo tworzonej bazy wiedzy [5]. Jeden z jej etapów, dla przykładowej bazy wiedzy, pokazany został na rys. 4. Dla rozpatrywanej reguły R_{23} konieczne jest odizolowanie pary reguł R_{12} i R_{21} (równanie (4)). O tym, która z nich zostanie przeniesiona do nowo tworzonej bazy wiedzy zależy od kierunku eliminowania

(tj. względem wierszy lub kolumn). W przypadku pokazanym na rys. 4 (eliminacji względem wierszy) usunięta zostaje reguła R_{13} .

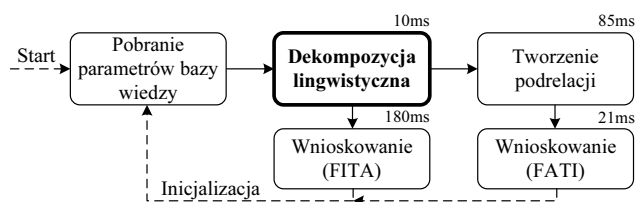


Rys. 4. Metoda podziału oparta na eliminacji reguł sprzecznych
Fig. 4. Partitioning method based on elimination of the inconsistent rules

Pierwsza z metod podziału charakteryzuje się niewielką złożonością, jednak nie w każdym przypadku daje wyniki optymalne, druga natomiast jest bardziej złożona, ale pozwala uzyskać lepsze wyniki podziału tzn. mniejszą wartość współczynnika p dla szerzej klasy systemów wnioskowania przybliżonego.

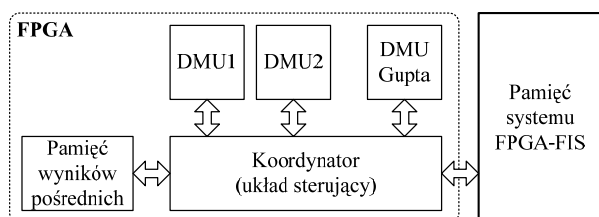
4. Sprzętowy system FPGA-FIS, moduł DMU

W ramach prowadzonych prac opracowany został sprzętowy system wnioskowania przybliżonego FPGA-FIS, zaimplementowany w układzie Spartan II XC2S200 [4, 10, 11], posiadający uniwersalną strukturę modułową. W zależności od potrzeb może on realizować dowolny algorytm wnioskujący, pracować jako system regułowy, relacyjny lub regułowo-relacyjny. Na rys. 5 przedstawiono cykl pracy systemu (z podanymi szacunkowymi czasami realizacji poszczególnych operacji), z wyróżnionym stanem, w którym przeprowadzana jest dekompozycja pierwotnej bazy wiedzy (tj. podział i dekompozycja metodą Gupty).



Rys. 5. Diagram pracy systemu wnioskowania przybliżonego FPGA-FIS
Fig. 5. Work diagram of the fuzzy inference FPGA-FIS system

Realizowana jest ona na drodze sprzętowej, z wykorzystaniem opracowanego modułu DMU (ang. *Decomposition Management Unit*). Zastosowano w nim zaprojektowane moduły realizujące wybrane strategie podziału zaproponowane w [4, 5] i wspomniane w rozdziale 3 (DMU1 i DMU2). Ze względu na to, iż pozwalają one uzyskać różne wartości współczynnika p (5), moduł koordynatora wybiera najlepszy rezultat i przekazuje go do modułu realizującego właściwą dekompozycję lingwistyczną Gupty (rys. 6). Wyniki końcowe zapisywane są w pamięci systemu FPGA-FIS i wykorzystywane do wyznaczenia wyniku (system FITA) lub do tworzenia podrelacji (baza wiedzy dla systemu FATI).



Rys. 6. Schemat blokowy modułu DMU
Fig. 6. Block diagram of the DMU module

5. Wnioski

W tabeli 1 zaprezentowano wyniki podziału przykładowych baz wiedzy systemów wnioskujących dla algorytmów opartych na podziale względem wybranej zmiennej (DMU1) oraz kierunku (DMU2) jak również zaproponowanej metody najlepszego wyboru (DMU). Do określania własności zaproponowanych metod wykorzystano bazy wiedzy następujących systemów: regulatory (oznaczone numerami 1 [12], 3 [13], 4 [9]), model bramki ENOR (2) [3], system parkowania ciężarówki (5) [14], regulator temperatury w systemie klimatyzacji (6) [15], system stabilizacji wahadła (7) [16], sterownik wentylatora (8) [17] oraz system identyfikacji układów nieliniowych (9) [18].

Tab. 1. Liczba baz wiedzy p , dla wybranych metod podziału
Tab. 1. The number of knowledge bases p for the selected partitioning method

Metoda podziału bazy wiedzy	System wnioskujący (benchmark)								
	1	2	3	4	5	6	7	8	9
Wartość optymalna	3	2	5	3	5	2	3	2	3
Podz. wzgl. zm. ling. wej. (DMU1)	5	2	5	3	5	3	5	3	5
Podz. wzgl. kier. X1 (DMU2)	3	2	5	4	6	3	3	2	3
Podz. wzgl. kier. X2 (DMU2)	4	2	5	4	5	3	3	3	4
Metoda najlepszego wyboru (DMU)	3	2	5	3	5	3	3	2	3

Można zauważyć, że metoda oparta na wyborze wyniku najlepszego spośród wyników uzyskiwanych metodami DMU1 i DMU2 tylko w jednym przypadku daje wynik gorszy od optymalnego (system (6)). Jej układowa implementacja charakteryzuje się jednak większym nakładem sprzętowym (tab. 2), ale uzyskiwane wyniki pozwalają ograniczyć do niezbędnego minimum liczbę podsystemów w strukturze hierarchicznej, a tym samym obniżyć całkowite nakłady wymagane w realizacji systemu wnioskującego oraz zminimalizować czas inicjalizacji (tworzenia podrelacji) dla systemu relacyjnego FATI [2, 10, 11].

Tab. 2. Zasoby sprzętowe wymagane do realizacji modułów DMU
Tab. 2. Hardware resources of the DMU modules

Metoda podziału bazy wiedzy (typ modułu)	Liczba			
	Slices	FF	LUTs	Bramek przel.
DMU1	42	44	54	640
DMU2	170	83	241	2494
DMU Gupta	112	65	207	1831
DMU	356	217	542	6713

6. Literatura

- [1] M. M. Gupta, J. B. Kiszka, G. M. Trojan, "Multivariable Structure of Fuzzy Control Systems", IEEE Transactions on Systems, Man, and Cybernetics, vol. 16, no. 5, 1986.
- [2] B. Wyrwoł, „Sprzętowa realizacja systemu wnioskowania przybliżonego z wykorzystaniem układów logiki programowalnej”, Rozprawa doktorska, Gliwice, 2004.
- [3] P. G. Lee, K. Lee Kyun, G. J. Jeon, „An Index of Applicability for the Decomposition Method of Multivariable Fuzzy Systems”, Fuzzy Systems, vol. 3, no. 3, 1995.
- [4] B. Wyrwoł, „Technika dekompozycji lingwistycznej oparta na podziale bazy wiedzy systemu wnioskowania przybliżonego”, VI KKE, Dąbrowka Wschodnie, tom 1, str. 207-212, 2007.
- [5] B. Wyrwoł, „Linguistic decomposition technique based on partitioning the knowledge base of the fuzzy inference system”, BULLETIN OF THE POLISH ACADEMY OF SCIENCES, TECHNICAL SCIENCES, Vol. 56, No. 1, pp. 71-76, 2008.

- [6] B. Wyrwoł, „Dekompozycja lingwistyczna bazy wiedzy systemu wnioskowania przybliżonego z eliminacją reguł sprzecznych”, VII KKE, Darłówko Wschodnie, tom 1, str. 237-242, 2008.
- [7] E. Czogała, W. Pedrycz, „Elementy i metody teorii zbiorów rozmytych”, skrypt uczelniany nr 1160, Gliwice 1983.
- [8] D. Driankov, H. Hellendoorn, M. Reinfrank, „Wprowadzenie do sterowania rozmytego”, WNT, Warszawa, 1996.
- [9] R. R. Yager, D. P. Filev, „Podstawy modelowania i sterowania rozmytego”, WNT, Warszawa, 1995.
- [10] B. Wyrwoł, „Regułowo-relacyjny system wnioskowania przybliżonego”, IV KKE, Darłówko Wschodnie, tom 2, str. 475-480, 2005.
- [11] B. Wyrwoł, D. Polok, „Hardware Implementation of the Linguistic Decomposition Technique in the FPGA-FIS System”, Scientific bulletin of the Politehnica University of Timisoara, Transactions on Electron. and Communications, Vol. 53 (67), Fascicola 1, pp. 161-166, '08.
- [12] I. Baturone, S. Sanchez-Solano, A. Barriga, J. L. Huertas, „Implementation of CMOS Fuzzy Controllers as Mixed-Signal Integrated Circuits”, IEEE Transactions on Fuzzy Systems, vol. 5, no. 1, 1997.
- [13] D. Kim, In-Hyun Cho, „An accurate and cost-effective COG defuzzifier without the multiplier and the divider”, Fuzzy Sets and Systems, vol. 104, 1999.
- [14] D. Rutkowska, M. Piliński, L. Rutkowski, „Sieci neuronowe, algorytmy genetyczne i systemy rozmyte”, PWN, Warszawa-Lódź, 1997.
- [15] A. Ollero, A. Garcia-Cerezo, „Direct digital control, auto-tuning and supervision using fuzzy logic”, Fuzzy Sets and Systems, vol. 30, 1989.
- [16] T. Yamakawa, „Stabilization of an inverted pendulum by a high-speed fuzzy logic controller hardware system”, Fuzzy Sets and Systems, vol. 32, 1989.
- [17] H. D. Hurdon, „Fuzzy logic fan controller”, nta.its.bldrdoc.gov/pub/fuzzy, 1993.
- [18] R. Rovatti, R. Guerrieri, G. Bacarani, „An enhanced Two-level Boolean Synthesis Methodology for Fuzzy Rules Minimization”, IEEE Trans. on Fuzzy Systems no. 3, 1995.

Artykuł recenzowany

INFORMACJE

Studia Podyplomowe

Wydział Elektryczny Politechniki Śląskiej w Gliwicach, Katedra Metrologii, Elektroniki i Automatyki
ogłasza nabór na Dwusemestralne Zaoczne Studia Podyplomowe

Systemy Pomiarowe i Sterowniki Programowalne (SPSP)

Cel Studiów

Celem studiów jest przekazanie wiedzy teoretycznej i umiejętności praktycznych w zakresie: projektowania, wdrażania i utrzymania ruchu systemów automatyki, programowania sterowników PLC oraz systemów nadrzędnych (SCADA), projektowania, programowania i eksploatacji automatycznych systemów pomiarowych w laboratoriach badawczych i przemysłowych, metod opracowania danych w systemach zapewnienia jakości procesów przemysłowych.

Profil uczestnika studiów

Studia przeznaczone są dla pracowników o różnych specjalnościach, z wyższym wykształceniem o kierunku elektrycznym, elektronicznym, informatycznym lub pokrewnym, zajmujących się organizacją pomiarów w laboratoriach badawczych i przemysłowych lub eksploatacją oraz modernizacją systemów sterowania. Ich ukończenie pozwoli uczestnikom na podwyższenie kwalifikacji niezbędnych do sprawnego opracowywania i wdrażania nowoczesnych systemów. Absolwent Studiów otrzymuje Świadectwo Ukończenia Studiów Podyplomowych w zakresie objętym nazwą studiów.

Studia prowadzone są na Wydziale Elektrycznym Politechniki Śląskiej w Gliwicach, w systemie zaocznym w każdą sobotę lub w co drugi weekend (do wyboru) przez dwa semestry. Zajęcia prowadzone są przez nauczycieli akademickich ze stopniem co najmniej doktora oraz przez zaproszonych Gości o uznanym dorobku i autorytecie. Studia obejmują 200 godzin dydaktycznych. Rozpoczęcie Studiów nastąpi po skompletowaniu odpowiedniej liczby kandydatów na dany rodzaj studiów.

Organizator studiów:

Katedra Metrologii, Elektroniki i Automatyki Politechniki Śląskiej, 44-100 Gliwice, ul. Akademicka 10, tel. 032 237 12 41, fax: 032 237 20 34, e-mail: re2@polsl.pl lub agnieszka.skorkowska@polsl.pl, <http://imeia.elekt.polsl.pl>

Kierownik studiów:

Prof. dr hab. inż. Tadeusz SKUBIS