

Transformacja rozmytej interpretowanej sieci Petriego na schemat układu logicznego

Lesław Gniewek

Streszczenie: W pracy przedstawiono formalne podstawy rozmytej interpretowanej sieci Petriego opisujące jej budowę i działanie. Zaproponowano metodę transformacji tej sieci na schemat układu logicznego, który oparto na rozmytych układach kombinacyjnych i sekwencyjnych. Schemat ten może być podstawą generowania kodu programu sterującego dla sterowników przemysłowych i prowadzić do częściowej automatyzacji tego procesu.

Słowa kluczowe: logika rozmyta, modelowanie, sieci Petriego

1. WPROWADZENIE

Obecnie wśród licznych prac na temat sieci Petriego można wyróżnić dwa główne nurty badań. Pierwszy z nich, najbardziej popularny, związany jest z sieciami wysokiego poziomu (*high-level PN*), które są połączeniem teorii sieci i języków programowania wysokiego poziomu. Drugi poświęcony jest sieciom niskiego poziomu (*low-level PN*), zwykle wykorzystywanym do modelowania i weryfikacji systemów opartych na reprogramowalnych układach sprzętowych czy też sterownikach przemysłowych. Rozmyte interpretowane sieci Petriego opisane w niniejszej pracy należą do sieci niskiego poziomu. Można odnaleźć w nich pewne cechy sieci interpretowanych, ciągłych, hybrydowych i tych sieci rozmytych, które działają w logice wielowartościowej.

W pracy zostaną przedstawione formalne podstawy rozmytej interpretowanej sieci Petriego, której działanie i możliwości zastosowania w układach sterowania i diagnostyki przedstawiono w [1, 2]. Następnie zostaną pokazane podstawowe komponenty, które będą wykorzystane do przekształcenia sieci na schemat układu logicznego, jak i sam sposób tej transformacji. Zaproponowana metoda budowy schematu układu logicznego pozwala na częściowe zautomatyzowanie procesu tworzenia kodu programu dla sterowników przemysłowych.

2. ROZMYTA INTERPRETOWANA SIĘĆ PETRIEGO

Formalne podstawy rozmytej interpretowanej sieci Petriego opisują trzy definicje.

Definicja 1. Rozmyta interpretowana sieć Petriego jest dwunastką:

$$FIPN = (P, T, D, G, R, \Delta, K, W, \Gamma, \Theta, M_0, e),$$

gdzie: $P = P' \cup P''$ - niepusty, skończony zbiór miejsc; $P' = \{p'_1, p'_2, \dots, p'_{a'}\}$ - zbiór miejsc związanych z modelowaniem działań; $P'' = \{p''_1, p''_2, \dots, p''_{a''}\}$ - zbiór miejsc związanych z modelowaniem zasobów; $T = \{t_1, t_2, \dots, t_b\}$ - niepusty, skończony zbiór tranzycji; $D = \{d_1, d_2, \dots, d_{a'+a''}\}$ - niepusty, skończony zbiór stwierdzeń; $G = \{g_1, g_2, \dots, g_b\}$ - niepusty, skończony zbiór warunków; P, T, D, G - zbiory parami rozłączne; $R \subseteq (P \times T) \cup (T \times P)$ - relacja incydencji, która dla każdej tranzycji $t_i \in T$, $i = 1, 2, \dots, b$, spełnia następujący warunek: $((p', t_i) \in R \text{ lub } (t_i, p') \in R)$; $\Delta: P \rightarrow D$ - funkcja przypisująca każdemu miejscu stwierdzenie; $K: P' \rightarrow 1$ i $P'' \rightarrow N \setminus \{1\}$ - funkcja przypisująca każdemu miejscu pojemność, gdzie $N = \{1, 2, \dots\}$; $\Gamma: T \rightarrow G$ - funkcja przypisująca każdej tranzycji warunek; $\Theta: T \rightarrow [0, 1]$ - funkcja określająca stopnie spełnienia warunków związanych z tranzycjami t ; $W: R \rightarrow N$ - funkcja wagowa spełniająca warunki: $W(p, t) \leq K(p)$ i $W(t, p) \leq K(p)$

(w skrócie p oznacza p' lub p''); $M_0: P \rightarrow \frac{z}{K}$, gdzie $z \in N \cup \{0\}$ i $z \leq K$ - funkcja znakowania początkowego; e - zdarzenie synchronizujące działanie wszystkich tranzycji.

Dbając o przejrzystą interpretację położenia rozmytych znaczników, w definicji sieci zapisano, że w chwili początkowej w miejscach p' znaczniki są „nierozmyte” („są w stopniu 1” lub „są w stopniu 0”), a znaczniki w miejscach p'' są wielokrotnością przypisanego im współczynnika normalizacji $\frac{1}{K}$ lub zerem. Ponadto założono, że każda tranzycja $t \in T$ musi posiadać co najmniej jedno miejsce wejściowe lub wyjściowe typu $p' \in P'$, gdyż znakowanie tych miejsc pozwala jednoznacznie określić położenie znacznika w czasie jego procesu przemieszczania przez uaktywnioną tranzycję.

Koncesję na uaktywnienie tranzycji otrzymuje w zależności od wartości znakowania jej miejsc wejściowych i wyjściowych.

Definicja 2. Tranzycja $t \in T$ jest przygotowana do uaktywnienia (ma koncesję) dla znakowania $M: P \rightarrow [0, 1]$ od momentu, gdy spełnione są warunki:

$$\forall p \in \bullet t, M(p) \geq \frac{W(p,t)}{K(p)} \quad \text{ i } \\ \forall p \in t^\bullet, M(p) \leq \frac{K(p)-W(t,p)}{K(p)}$$

do momentu, gdy:

$$\forall p' \in \bullet t, M(p) = 0 \quad \text{ i } \quad \forall p' \in t^\bullet, M(p) = 1,$$

gdzie: $\bullet t = \{p \in P \mid (p,t) \in R\}$ oznacza zbiór miejsc wejściowych tranzycji t , a $t^\bullet = \{p \in P \mid (t,p) \in R\}$ - zbiór jej miejsc wyjściowych.

Czas aktywności tranzycji zależy od szybkości zmian warunku związanego z tą tranzycją. Sposób wyznaczania nowego znakowania sieci opisuje Definicja 3.

Definicja 3. Jeżeli dla znakowania M tranzycja $t \in T$ jest przygotowana do uaktywnienia, stopień spełnienia warunku związanego z przygotowaną tranzycją $\Theta(t) = \Theta \in [0,1]$ zwiększy się o $\Delta\Theta$ i wystąpi zdarzenie e synchronizujące działanie wszystkich tranzycji, to nowe znakowanie sieci M' można wyznaczyć za pomocą następującej reguły:

$$M'(p) = \begin{cases} M(p) - \Delta\Theta \cdot \frac{W(p,t)}{K(p)} & \Leftrightarrow p \in \bullet t, \\ M(p) + \Delta\Theta \cdot \frac{W(t,p)}{K(p)} & \Leftrightarrow p \in t^\bullet, \\ M(p) & \Leftrightarrow p \notin \bullet t \cup t^\bullet. \end{cases}$$

Zgodnie z tą definicją uaktywnienie tranzycji jest synchronizowane zewnętrznym sygnałem e , który pełni rolę sygnału taktującego pracę sieci. Opisywana sieć Petriego należy więc do klasy sieci synchronizowanych.

3. TRANSFORMACJA SIECI

Sposób implementacji sieci Petriego działającej w logice dwuwartościowej opisano w [5]. Według tej metody każdemu miejscu sieci przyporządkowuje się klasyczny przerzutnik JK, a każdej tranzycji - bramkę AND. W rezultacie otrzymuje się schemat układu logicznego realizujący binarną sieć Petriego. Rozszerzenie tej metody do logiki rozmytej zaproponowano w [3] i użyto do transformacji rozmytej sieci Petriego na schemat układu logicznego. Klasyczne układy kombinacyjne i sekwencyjne zastąpiono w sposób naturalny elementami rozmytymi. W ten sposób możliwym stało się przechowywanie nie tylko informacji binarnej, ale i sygnałów analogowych znormalizowanych do przedziału [0,1]. Według metody opisanej w [3] każdemu miejscu sieci przyporządkowano rozmyty przerzutnik JK. Każdej tranzycji przypisano układ zawierający rozmytą bramkę iloczynu ograniczonego lub bramkę „min” oraz dwustanowy przerzutnik typu T. Pokazano również praktyczną implementację takiego układu w sterowniku PLC. W [4] zaproponowano zastąpienie rozmytych przerzutników JK przez rozmyte przerzutniki SR. Dzięki temu uzyskano zmniejszenie kosztów realizacji i szybkości działania w układach FPGA, które wykorzystano do implementacji sieci.

W dalszej części rozdziału omówiona zostanie metoda transformacji rozmytej interpretowanej sieci

Petriego na schemat układu logicznego. Wykorzystany do tego celu będzie zmodyfikowany rozmyty przerzutnik SR.

3.1. ROZMYTE PRZERZUTNIKI SR

Klasyczne przerzutniki SR działające w logice dwuwartościowej opisywane są za pomocą równań charakterystycznych:

$$Q_{set}(n+1) = S \vee (\bar{R} \wedge Q), \quad (1)$$

$$Q_{reset}(n+1) = \bar{R} \wedge (S \vee Q) \quad (2)$$

(dla uproszczenia zapisu po prawej stronie równań użyto oznaczeń S, R, Q zamiast $S(n), R(n), Q(n)$). Przerzutniki te mogą zostać uogólnione do logiki rozmytej za pomocą t-norm, s-norm i rozmytej negacji. Najczęściej do tego celu wykorzystywane są dwie grupy operacji:

- grupa operacji Zadeha:

$$x \wedge y = \min(x, y) - \text{t-norma}, \quad (3)$$

$$x \vee y = \max(x, y) - \text{s-norma}, \quad (4)$$

$$\bar{x} = 1 - x - \text{rozmyta negacja}, \quad (5)$$

- grupa operacji ograniczonych:

$$x \otimes y = 0 \vee (x + y - 1) - \text{t-norma}, \quad (6)$$

$$x \oplus y = 1 \wedge (x + y) - \text{s-norma}, \quad (7)$$

rozmyta negacja zdefiniowana jak w (5).

Rozmyte przerzutniki SR bazujące na grupie operacji Zadeha doczekały się realizacji sprzętowych w układach FPGA. Opracowano i zaimplementowano 136 równań opisujących te przerzutniki [6]. Ponadto w [4] pokazano realizację synchronicznego i asynchronicznego rozmytego przerzutnika SR.

W dalszej części rozdziału do syntezy sieci zostanie wykorzystany rozmyty synchroniczny przerzutnik SR opisany następującym równaniem:

$$Q(n+1) = [(aS \oplus Q) \otimes \bar{bR}], \quad (8)$$

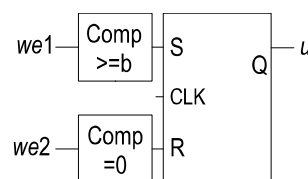
gdzie

$$a = \frac{W(t,p)}{K(p)} \leq 1, \quad b = \frac{W(p,t)}{K(p)} \leq 1.$$

Równanie (8) powstało poprzez uogólnienie równania (2) za pomocą operacji ograniczonych (5), (6), (7) oraz wprowadzenie dwóch współczynników a i b , które umożliwią zmianę stanu przerzutnika zgodnie z parametrami sieci i Definicją 3.

3.2. UKŁAD AKTYWACJI

Kolejnym elementem, który będzie użyty do transformacji sieci jest układ A odpowiedzialny za aktywację tranzycji zgodnie z Definicją 2. Składa się on z dwóch komparatorów i klasycznego przerzutnika SR (Rys. 1).



Rys. 1. Układ aktywacji A

Sposób działania tego układu opisuje jego funkcja charakterystyczna:

$$u(n+1) = \begin{cases} 0 & \text{dla } we2 = 0 \\ u(n) & \text{dla } we2 \neq 0 \text{ i } we1 < b \\ 1 & \text{dla } we2 \neq 0 \text{ i } we1 \geq b \end{cases} \quad (9)$$

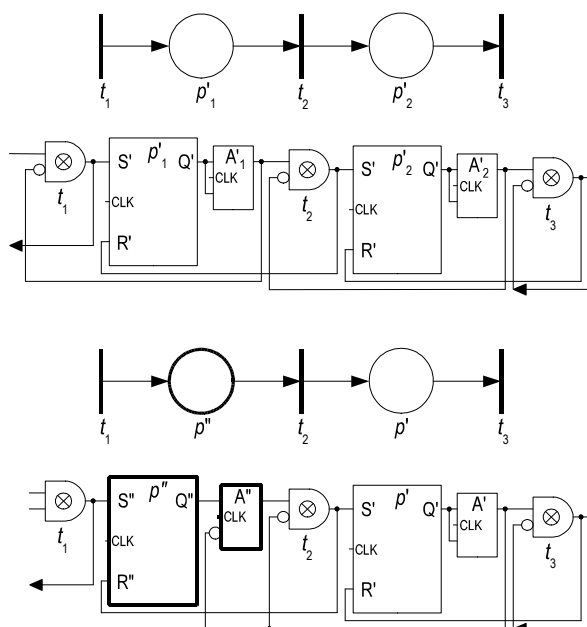
Dla uproszczenia proponowanej metody założono, że pojemności miejsc typu p" są na tyle duże, że nie ma potrzeby sprawdzania, czy po uaktywnieniu tranzycji zostaną one przekroczone.

3.3. SPOSÓB ŁĄCZENIA ELEMENTÓW

Według proponowanej metody transformacji każdemu miejscu sieci należy przyporządkować rozmyty przerzutnik SR opisany równaniem (8) i na jego wyjściu podłączyć układ aktywacji A przedstawiony na Rys. 1. W układach przypisanych do miejsc typu p" trzeba zastosować współczynniki a i b zależnie od pojemności modelowanego miejsca i przepustowości łuków z nim związanych. W układach przyporządkowanych miejscom typu p' należy przypisać współczynniki jednostkowe (a=b=1).

Na obydwie wejścia układów aktywacji A' podawany jest ten sam sygnał pochodzący z wyjścia przerzutnika SR. W układach aktywacji A" podawany jest on tylko na wejście we1. Jeśli układ A" związany jest z miejscem wejściowym tranzycji t, to na we2 podawane jest rozmyte zaprzeczenie wartości wyjściowej układu aktywacji A' związanego z miejscem wyjściowym tej tranzycji.

Tranzycjom przypisuje się rozmyte bramki iloczynu ograniczonego. Na Rys. 2 przedstawiono przykładowe fragmenty sieci, w których wszystkie tranzycje są bezwarunkowe i posiadają najwyżej jedno miejsce wejściowe i jedno wyjściowe.

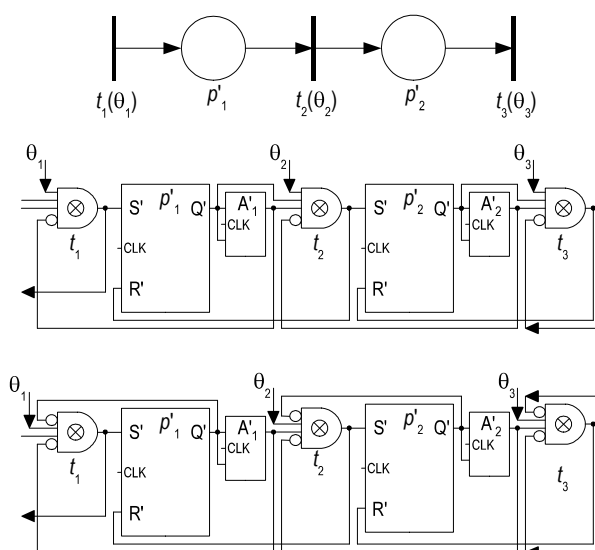


Rys. 2. Fragmenty sieci zawierające wyłącznie tranzycje bezwarunkowe oraz odpowiadające im schematy

Wszystkie przerzutniki SR są taktowane wspólnym sygnałem CLK. Układy aktywacji A powinny być taktowane tym samym sygnałem, ale z pewnym przesunięciem (fazą), aby były uaktywniane z nowymi

wartościami ustawianymi na wyjściach przerzutników. Dla zachowania czytelności schematu pominięto połączenia pomiędzy wejściami zegarowymi, jak również dodatkowe parametry wejściowe a i b przerzutników modelujących miejsca p" i układów aktywacji A".

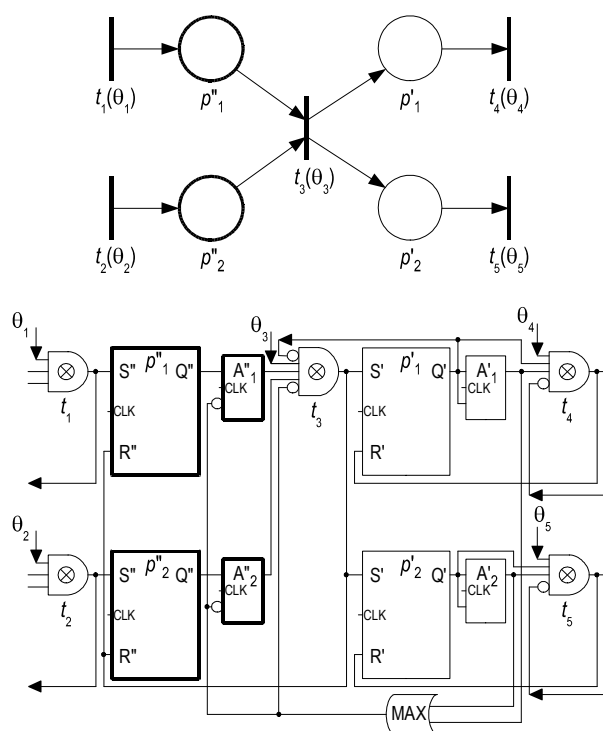
Jeżeli z modelowaną tranzycją związany jest dodatkowy warunek, to na wejście rozmytej bramki należy dodatkowo podać 2 sygnały. Pierwszym z nich jest stopień spełnienia tego warunku. Drugim sygnałem jest albo sygnał wyjściowy z dowolnego przerzutnika SR modelującego miejsce wejściowe typu p' albo zaprzeczony sygnał wyjściowy z dowolnego przerzutnika SR modelującego miejsce wyjściowe typu p' tej tranzycji. Dzięki temu w czasie aktywności tranzycji t na wyjściu rozmytej bramki, która ją modeluje, podawany będzie przyrost $\Delta\theta$ stopnia spełnienia warunku związanego z tą tranzycją. Na Rys. 3 przedstawiono dwa alternatywne sposoby podawania sygnałów na wejście rozmytej bramki modelującej tranzycję.



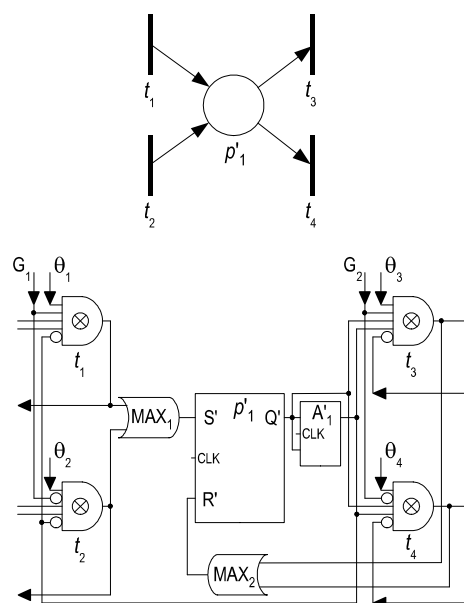
Rys. 3. Fragment sieci zawierający tranzycje warunkowe i dwa schematy pokazujące alternatywne sposoby podawania sygnałów na bramkę iloczynu ograniczonego

W przypadku, gdy tranzycja posiada kilka miejsc wejściowych i wyjściowych (Rys. 4), na wejścia bramki odpowiadającej tej tranzycji podaje się sygnały ze wszystkich układów aktywacji związanych z miejscami wejściowymi oraz zaprzeczenie maksymalnego sygnału ze wszystkich układów aktywacji związanych z miejscami wyjściowymi typu p' tej tranzycji.

W rozmytej interpretowanej sieci Petriego dopuszcza się możliwość wystąpienia konfliktu. W takich przypadkach tylko jedna tranzycja wejściowa lub wyjściowa miejsca typu p' może być aktywna. Miejsce typu p" może posiadać równocześnie jedną aktywną tranzycję wejściową i jedną aktywną tranzycję wyjściową. Pamiętając o tych ograniczeniach, do tranzycji mogących brać udział w konflikcie należy przypisać warunki wzajemnie wykluczające się, np. wprowadzając dodatkowe wejścia do bramek iloczynu ograniczonego, na które podawany będzie sygnał rozstrzygający konflikt (Rys. 5). Wartość tego sygnału nie może ulec zmianie w czasie aktywności którejkolwiek tranzycji biorącej udział w rozstrzyganym konflikcie.



Rys. 4. Fragment sieci zawierający tranzycję z dwoma miejscami wejściowymi typu p'' wraz z odpowiadającym schematem



Rys. 5. Fragment sieci zawierający miejsce typu p' oraz odpowiadający mu schemat

Gdy jedno miejsce typu p'' ma kilka tranzycji wejściowych i wyjściowych, a do tego różne wagi łuków łączących to miejsce z tranzycjami, to należy dodatkowo zastosować bloki selekcjonujące, które nie tylko dokonają wyboru tranzycji, ale również zmieniają parametry wejściowe przerzutnika związane z wagami łuków.

4. PODSUMOWANIE

Formalny opis algorytmu sterowania można przedstawić w postaci rozmytej interpretowanej sieci Petriego, co umożliwia dokonanie analizy jego własności już na etapie syntezy abstrakcyjnej. Ma to duże znaczenie

przy stale rosnącej złożoności algorytmów i wzroście wymagań dotyczących ich formalnej weryfikacji. Sieć pozwala uwzględnić istnienie zarówno sygnałów binarnych, jak i analogowych oraz opisać ilościowe zmiany zachodzące w systemie nie tracąc jednocześnie naturalnej interpretacji położenia znacznika w sieci. Zaproponowana metoda transformacji sieci na schemat układu logicznego umożliwia tworzenie jej realizacji programowych, w klasycznych językach konfiguracji sterowników PLC (np. jako schemat drabinkowy), poprzez przygotowanie odpowiednich bloków programowych realizujących wszystkie komponenty otrzymanego schematu. Wykorzystując tą metodę można częściowo zautomatyzować proces przejście od rozmytej interpretowanej sieci Petriego jako algorytmu sterowania, przez schemat układu logicznego, do programu sterującego zaimplementowanego na sterowniku PLC. Stworzenie środowiska narzędziowego wspomagającego ten proces znacznie zwiększyłoby szansę zastosowania opisanych sieci w warunkach przemysłowych.

Praca naukowa częściowo finansowana ze środków na naukę w latach 2008 - 2010 jako projekt badawczy N N514 413934.

LITERATURA

- [1] Gniewek L., „Modelowanie układu sterowania i diagnostyki za pomocą rozmytej interpretowanej sieci Petriego”, Kowalczyk Z. (red.): *Systemy wykrywające, analizujące i tolerujące usterki*, PWN, Gdańsk 2009, ss. 105-112.
- [2] Gniewek L., „Rozmyta interpretowana sieć Petriego jako układ sterowania”, *Pomiary Automatyka Kontrola*, 2009, vol. 55, nr 7, ss. 494-497.
- [3] Gniewek L., Kluska J., „Hardware implementation of fuzzy Petri net as a controller”, *IEEE Trans. on Systems, Man, and Cybernetics, Part B: Cybernetics*, 2004, vol. 34, no. 3, pp. 1315-1324.
- [4] Hajduk Z., „Sprzętowa implementacja rozmytych sieci Petriego jako układów sterowania”, rozprawa doktorska, *Uniwersytet Zielonogórski*, 2005.
- [5] Misiurewicz P., „Zagadnienia projektowania cyfrowych układów sterowania binarnego”, *VIII Krajowa Konferencja Automatyki*, Szczecin 1980, t. 1, ss. 664-670.
- [6] Yoshida S., Hirota K., „Lattice Structure of D, T, and SR Fuzzy Flip-Flops Under Max-Min Logic”, *Journal of Advanced Computational Intelligence and Intelligent Informatics*, 2005, vol. 9, no. 6.



dr inż. Lesław Gniewek
Politechnika Rzeszowska
Wydział Elektrotechniki i Informatyki
Katedra Informatyki i Automatyki

ul. W. Pola 2
35-959 Rzeszów

tel.: 17 865 15 36
e-mail: lgniewek@prz-rzeszow.pl