

Łukasz NIEWIARA*
Krzysztof ZAWIRSKI*

AUTO-STROJENIE REGULATORA TYPU PID Z WYKORZYSTANIEM LOGIKI ROZMYTEJ

Zagadnienia związane z automatycznym doбором nastaw regulatorów typu PID znane są już od kilkunastu lat. W niniejszym artykule opisano koncepcję auto-strojenia wykorzystującego Blok Logiki Rozmytej, który w całym procesie pełni rolę układu nadrzędnego, reprezentującego wiedzę ekspercką. Opisano tutaj zalety zastosowania logiki rozmytej, strukturę zaprojektowanego układu symulacyjnego, algorytm strojenia, sposób zmiany nastaw oraz przedstawiono wyniki badań symulacyjnych wykonanych w środowisku MatLab.

1. WSTĘP

Proces auto-strojenia regulatorów typu PID pozwala na minimalizację wpływu czynnika ludzkiego przy doborze jego nastaw, ogranicza się on do uruchomienia procesu i ewentualnego jego nadzorowania. W świetle szybkości rozwoju postępu technologicznego oraz przemysłu zastosowanie algorytmów „*auto tuningu*” daje możliwość szybszej realizacji projektów. Konwencjonalne podejście do procesu doboru nastaw regulatora jest często procesem żmudnym i czasochłonnym, dodatkowo proces ten jest ograniczony dostępnością kryteriów. Procesy samo-strojenia realizują się według ściśle określonego algorytmu, w trakcie kolejnych cykli iteracyjnych. Procesem kieruje odpowiedni sterownik, który znacznie szybciej analizuje otrzymane dane procesowe, aniżeli człowiek. Dzięki temu uzyskuje się znaczne przyspieszenie procesu strojenia, co pozwala na szybszy rozwój aplikacji przemysłowych.

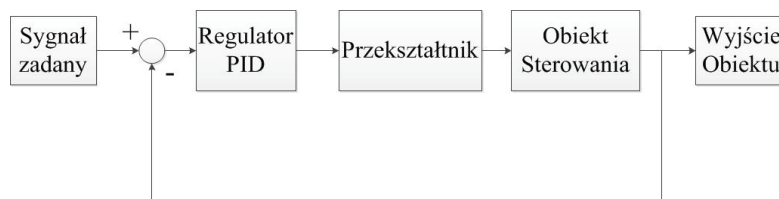
Zastosowanie logiki rozmytej do przeprowadzenia procesu strojenia pozwoliło na jego bardziej ogólny opis, gdzie częściowo uniezależniono się od ściśle matematycznego opisu algorytmu. Podejście to wykorzystuje zalety nieprecyzyjnego opisu wskaźników jakości procesu regulacji za pomocą odpowiednich zmiennych lingwistycznych oraz baz reguł. Blok Logiki Rozmytej (BLR) w zaprojektowanym algorytmie nie wpływa bezpośrednio na proces regulacji, odpowiada on jedynie za aktualizację nastaw regulatora. Takie podejście pozwoliło na zastosowanie tradycyjnego algorytmu regulacji PID, gdzie BLR jest blokiem nadrzędnym odpowiadającym wiedzy eksperckiej.

* Politechnika Poznańska.

2. OPIS SYMULOWANEGO UKŁADU

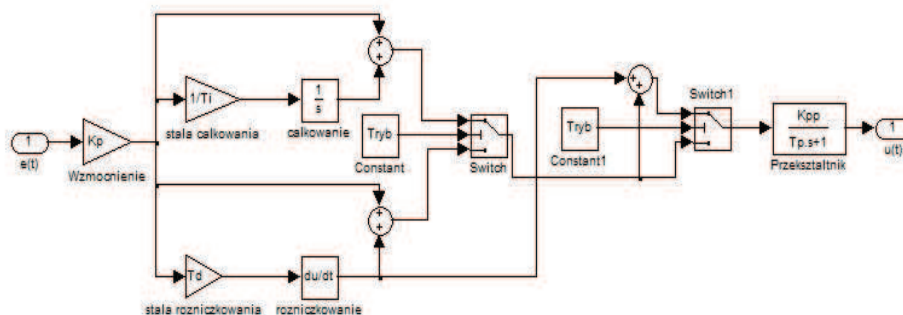
Proces samo-strojenia symulowano przy pomocy środowiska MatLab stosując utworzone w tym celu m-skrypty, gdzie model układu regulacji zbudowano w Simulinku.

W procesie symulacji auto-strojenia przyjęto układ kaskadowo połączony regulator z przekształtnikiem jako członem wykonawczym oraz modelem obiektu regulacji w domkniętej pętli ujemnego sprzężenia zwrotnego (Rys. 1).



Rys. 1. Schemat blokowy struktury układu sterowania

Blok regulatora zaprojektowano tak, aby w zależności od klasy modelu obiektu sterowania możliwy był wybór jego pracy w trybach: PD, PI oraz PID.



Rys. 2. Schemat bloku PID wykorzystanego do symulacji

Proces aktualizacji wartości nastaw regulatora realizowany był przez odpowiedni algorytm. Rola czynnika ludzkiego ogranicza się w tym przypadku do wstępnego doboru nastaw regulatora, określenia wymagań w stosunku do wskaźników jakości oraz uruchomienia procesu.

Kierunek zmian nastaw regulatora uzależniono od wartości wskaźników jakości odpowiedzi układu na wymuszenie skokowe, przyjęto następujące wskaźniki jakości:

- maksymalne przeregulowanie ϵ ,
- znormalizowany czas regulacji T_w dla 2% toru regulacji,

$$T_w = \frac{T_r}{100T_p} \quad (1)$$

Wartości zmian nastaw wyliczono przy pomocy BLR na podstawie wskaźników jakości.



Rys. 3. Blok rozmyty – dane wejściowe i wyjściowe

Zmiany nastaw regulatora realizowano według następujących zależności:

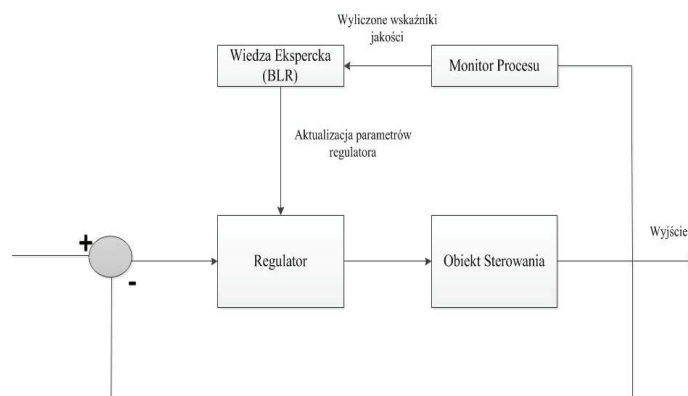
$$K_p(i+1) = K_p(i)C^{\Delta K_p} \quad (2)$$

$$T_I(i+1) = T_I(i)C^{\Delta T_I} \quad (3)$$

$$T_D(i+1) = T_{DI}(i)C^{\Delta T_D} \quad (4)$$

gdzie: C – podstawa potęgi determinująca szybkość zmian nastaw, $K_p(i+1)$ – nowa wartość wzmacnienia regulatora, $T_I(i+1)$ – nowa wartość czasu całkowania regulatora, $T_D(i+1)$ – nowa wartość czasu różniczkowania regulatora, ΔK_p , ΔT_I , ΔT_D – obliczone wartości przyrostów nastaw regulatora dla i -tej iteracji.

Każdą zmienną lingwistyczną opisano przy pomocy odpowiedniej zależności funkcyjnej, jej kształt i strukturę wyznaczono eksperymentalnie. Dla wskaźników jakości procesu regulacji określono po 4 funkcje reprezentowane przez następujący zbiór wartości lingwistycznych: $\{Z, S, M, B\}$, odpowiednio: Z – *Zero*, S – *Small*, M – *Medium*, B – *Big*.



Rys. 4. Budowa zaprojektowanego układu auto-strojenia

Dziedzinę zmiennej „Przeregulowanie” przyjęto w zakresie $\langle 0, 0.65 \rangle$, jako procentową wartość przeregulowania, zaś dla zmiennej „Czas regulacji” przyjęto zakres $\langle 0, 800 \rangle$.

Wyjściowym wartościom przyporządkowano po 5 funkcji zmiennych lingwistycznych: $\{NB, NS, Z, PS, PB\}$, odpowiednio: NB – *Negative Big*, NS –

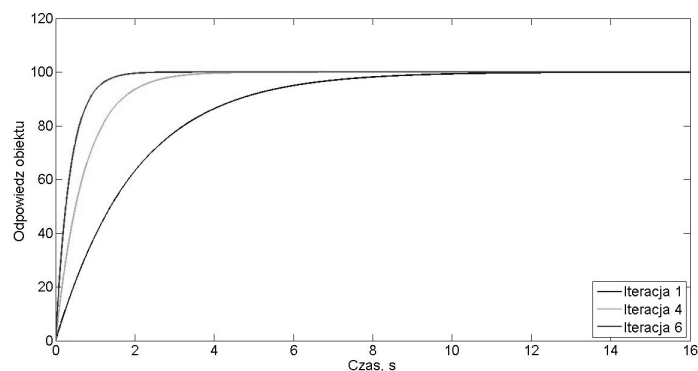
Negative Small, Z – Zero, PS – Positive Small, PB – Positive Big. Dla wszystkich zmiennych przyjęto dziedzinę w zakresie $\langle -1, 1 \rangle$.

Wnioskowanie rozmyte (bazy reguł) zdefiniowano przy pomocy opracowanych zmiennych lingwistycznych.

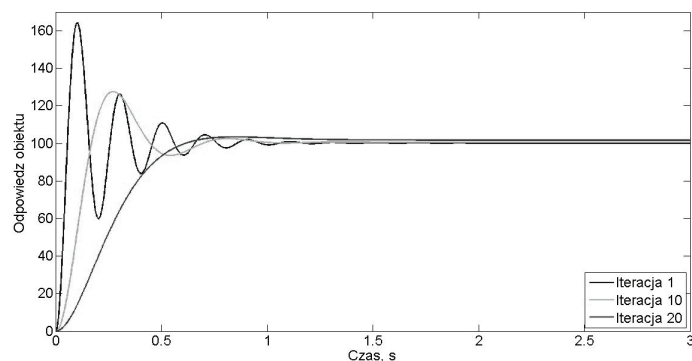
3. BADANIA SYMULACYJNE

Na wejście podawano sygnał skokowy o amplitudzie równej 100. Następnie badano odpowiedź układu i określano wskaźniki jakości procesu regulacji. Zatrzymanie procesu następowało po osiągnięciu odpowiednich kryteriów przez odpowiedź układu lub po osiągnięciu limitu ilości iteracji.

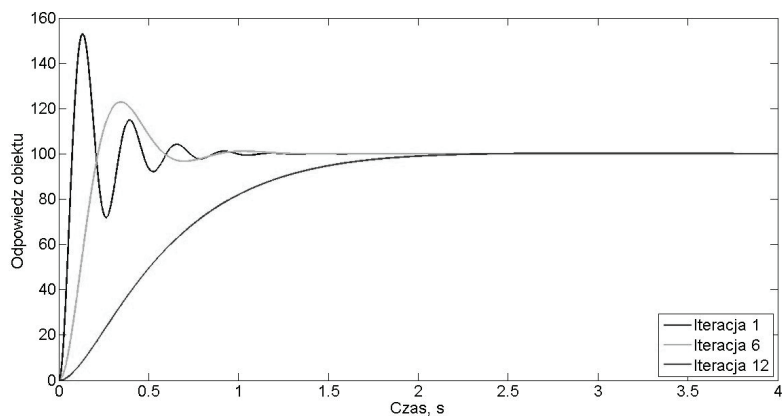
Wybrane wyniki badań przedstawiono na rysunkach 5 – 8. Rysunki ilustrują proces samo-strojenia dla czterech różnych obiektów regulacji. Z ilustracji tych wynika, że po kilku lub kilkunastu krokach procesu strojenia osiąga się poprawny proces regulacji tj. bez przeregulowania i względnie krótkim czasem regulacji.



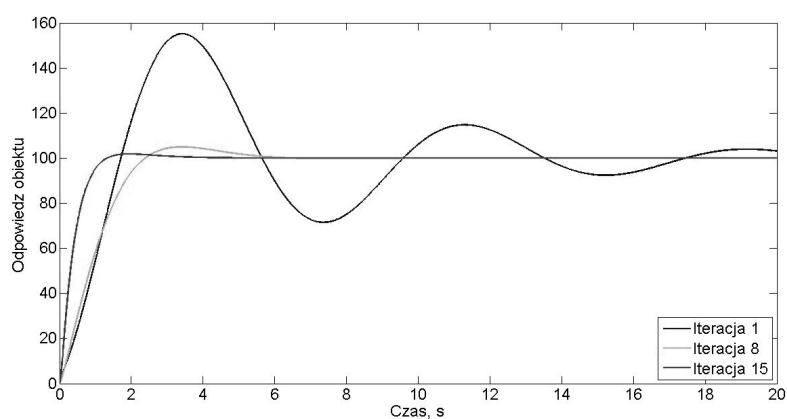
Rys. 5. Przebieg symulacji strojenia nr 1 – PD obiekt astatyczny



Rys. 6. Przebieg symulacji strojenia nr 2 – PI obiekt astatyczny



Rys. 7. Przebieg symulacji strojenia nr 3 – PI obiekt statyczny



Rys. 8. Przebieg symulacji nr 4 – PID obiekt statyczny

4. PODSUMOWANIE

Badania symulacyjne zaprojektowanego układu auto-strojenia wraz ze zdefiniowanymi funkcjami przynależności oraz bazami reguł okazały się pomyślne i zgodne z założeniami. Po podaniu wstępnych nastaw regulatora proces modyfikował je, aż do osiągnięcia odpowiedzi układu spełniającej postawione jej kryteria.

Wyniki badań pokazują, że proces przebiega pomyślnie dla różnego typu klasy modeli obiektów sterowania, a w przypadku wystąpienia oscylacji na wyjściu, po kilku iteracjach jest w stanie je wyeliminować, otrzymując ostatecznie łagodny przebieg odpowiedzi.

LITERATURA

- [1] R. Kaula „Podstawy automatyki” Wydawnictwo Politechniki Śląskiej, 2005.
- [2] A. Piegat „Modelowanie i sterowanie rozmyte” Akademicka Oficyna Wydawnicza EXIT, Warszawa 1999.
- [3] Ł. Niewiara „Samostrojenie regulatora PID dla obiektu o nieznanym parametrach z wykorzystaniem logiki rozmytej”, Politechnika Poznańska, Praca magisterska, 216/Z2/2012.

AUTOTUNING OF PID CONTROLLER BASED ON FUZZY LOGIC

Issues related to the automatic selection of the PID controller settings have been known for several years. This article describes the concept of autotuning using fuzzy logic block, which in the process serves as the master. It describes the advantages of the application of fuzzy logic simulation system designed structure, algorithm tuning, how to change the settings and the results of simulation studies carried out in MatLab environment.