

CZASOOPTYMALNY ODPORNY ALGORYTM STEROWANIA SERWONAPĘDU ELEKTRYCZNEGO

STRESZCZENIE

W artykule przedstawiono stosunkowo prosty aplikacyjnie algorytm sterowania serwośilnika elektrycznego DC, mający zapewnić jego sterowanie czasooptymalne w sytuacji zmian momentu zewnętrznego (obciążenia) i momentu bezwładności całego napędu. Działanie algorytmu zaprezentowano w porównaniu z ogólnie znanym kaskadowym algorytmem regulacji kąta położenia serwośilnika. Działanie obu algorytmów porównawczo przebadano symulacyjnie w pakiecie MATLAB/Simulink.

Słowa kluczowe: serwonapęd, sterowanie odporne, sterowanie czasooptymalne

THE TIME-OPTIMAL ROBUST CONTROL ALGORITHM OF ELECTRIC SERVODRIVE

This paper shows quite simple to apply algorithm of DC/DC brushless servodrive (actuator) control. The algorithm should guarantee time-optimal control while load disturbance and moment of inertia variations. Working of this algorithm is presented in comparison with working of well known cascade algorithm of DC servomotor control. Working of both algorithms was compared by MATLAB/Simulink simulations.

Keywords: servodrive, robust control, time-optimal control

1. WSTĘP

Artykuł przedstawia czasooptymalny odporny algorytm regulacji położenia serwomechanizmu elektrycznego wykorzystujący szczotkowy silnik prądu stałego (silnik magnetoelektryczny) lub bezszczotkowy silnik prądu stałego o trapezowym przebiegu siły elektromotorycznej. Celem opracowania przedstawianego algorytmu było maksymalne ograniczenie wpływu zmian momentu zewnętrznego (obciążenia) i momentu bezwładności napędu na jakość regulacji.

Za najważniejsze kryteria jakości regulacji uznano:

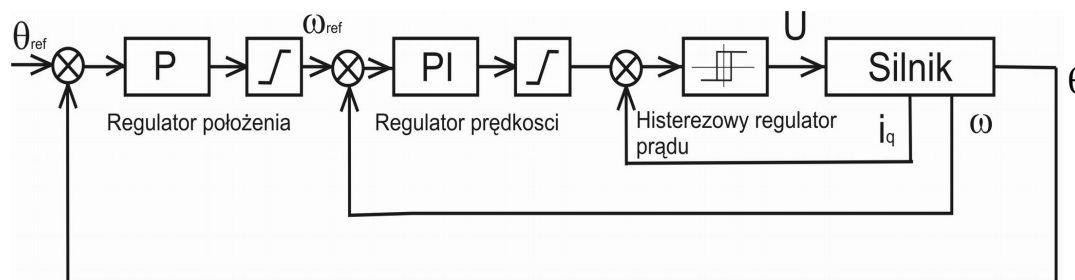
- dokładność osiągnięcia zadanego położenia przy braku przeregulowań,
- minimalizację czasu regulacji.

Ważnym warunkiem dookreślającym tworzony algorytm był postulat jego aplikowalności na prostych, 8-bitowych mikrokontrolerach ogólnego zastosowania, przy sterowaniu serwośilnikiem za pośrednictwem półprzewodnikowych przyrządów mocy – tranzystorów mocy. Rezultaty działania prezentowanego algorytmu przedstawiono na tle

rezultatów działania ogólnie znanego algorytmu regulacji położenia, w którym reguluje się położenie i jego kolejne pochodne (prędkość, przyspieszenie) w układzie kaskadowym (rys. 1).

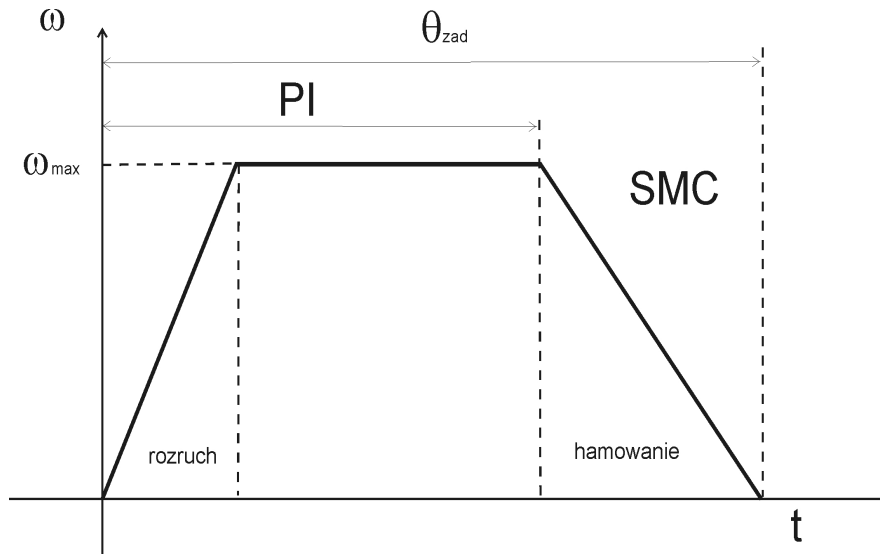
2. ALGORYTM REGULACJI

Zasadą działania przedstawianego algorytmu jest odtworzenie czasooptymalnego, w ogólnym przypadku trapezowego, przebiegu prędkości serwonapędu. W dwóch pierwszych fazach takiego ruchu (fazie przyspieszania i fazie ruchu ze stałą prędkością) wykorzystywany jest jako regulator prędkości regulator typu PID (PI), o prędkości zadanej równej maksymalnej dopuszczalnej prędkości napędu ($\omega_{ref} = \omega_{max}$). Położenie napędu nie jest tu bezpośrednio kontrolowane. Przy przejściu do trzeciej fazy ruchu (fazy hamowania) następuje zmiana algorytmu regulacji prędkości na algorytm ściśle odtwarzający czasooptymalny (a więc malejący liniowo w czasie) przebieg prędkości. Algorytm ten został określony na rysunkach 2 i 3 jako SMC (*Sliding Mode Control*) – tryb regulacji ślizgowej.

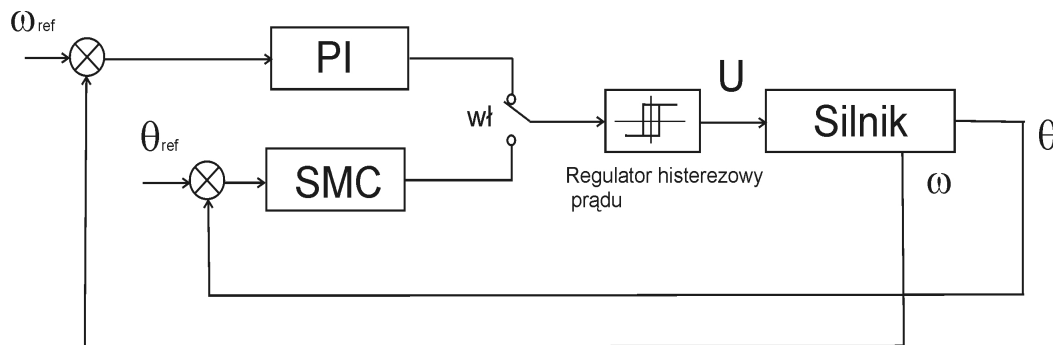


Rys. 1. Schemat blokowy klasycznego, kaskadowego układu automatycznej regulacji kąta położenia

* Katedra Maszyn Elektrycznych, Akademia Górniczo-Hutnicza w Krakowie



Rys. 2. Referencyjna trajektoria ruchu serwonapędu sterowanego według prezentowanego algorytmu



Rys. 3. Schemat blokowy prezentowanego algorytmu sterowania serwonapędu

Działa on bowiem na zasadzie pośredniej kontroli prędkości napędu ω w funkcji jego położenia φ , co stanowi zasadę regulacji ślizgowej [4]. Zależność położenia napędu od jego prędkości jest tutaj kwadratowa. Chwila przejścia do trzeciej fazy ruchu określona jest na podstawie równań ruchu jednostajnie opóźnionego. W każdej chwili pracy napędu, także w dwóch pierwszych fazach ruchu, obliczane jest opóźnienie a_{hobl} , z którym w danej chwili powinien zacząć hamować napęd, aby dojść do zadanego położenia po trajektorii czasooptymalnej. Efektywne przejście do fazy hamowania następuje wówczas, gdy wartość ta wzrośnie do wartości opóźnienia granicznego a_{max} , możliwego do uzyskania przez silnik rozwijający największy dopuszczalny moment hamujący T_{emax} :

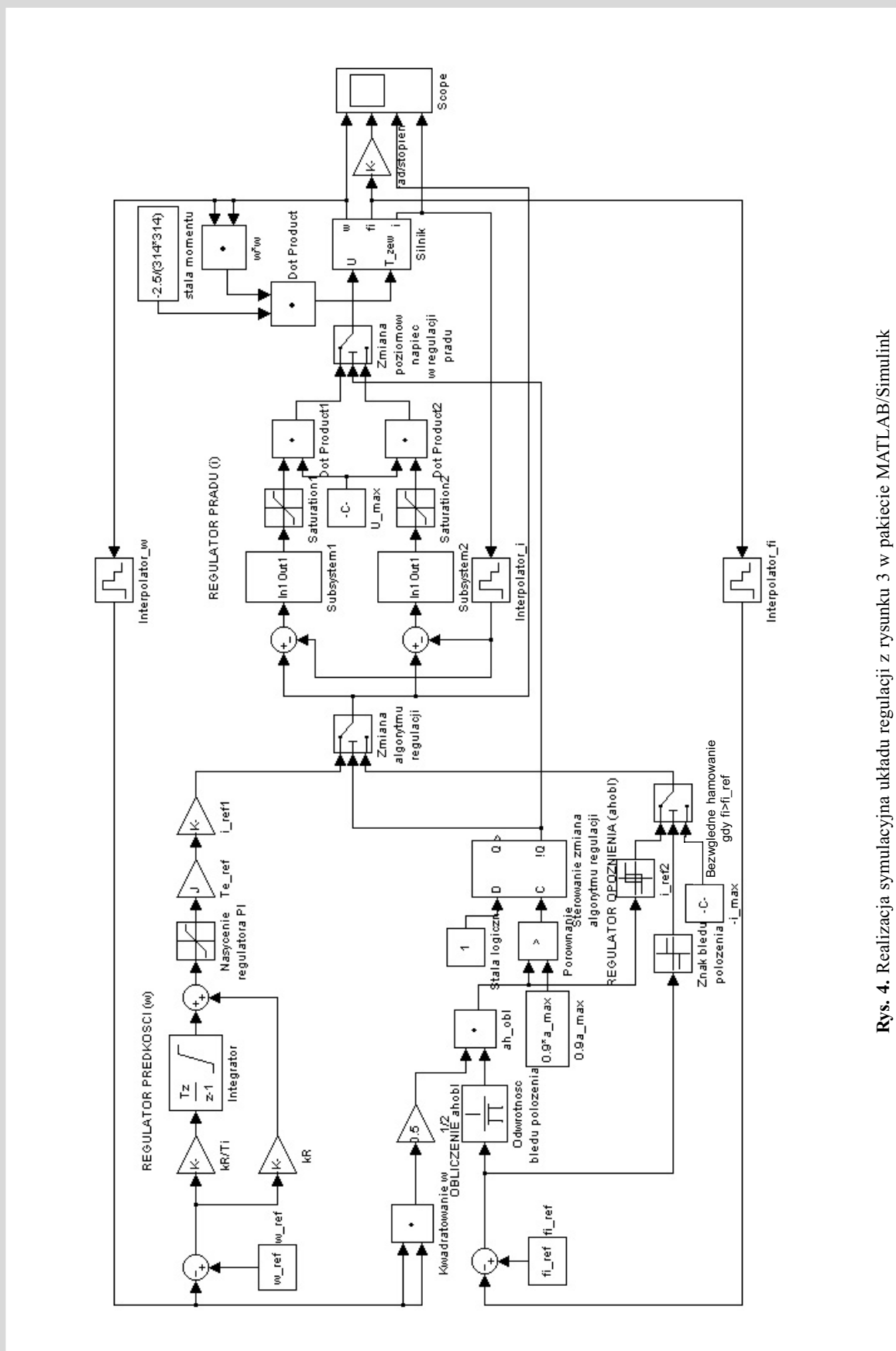
$$a_{hobl} = \frac{\omega^2}{2 \cdot (\varphi_{ref} - \varphi)} \quad (1)$$

$$a_{max} = \frac{T_{emax}}{J_{max}} \quad (2)$$

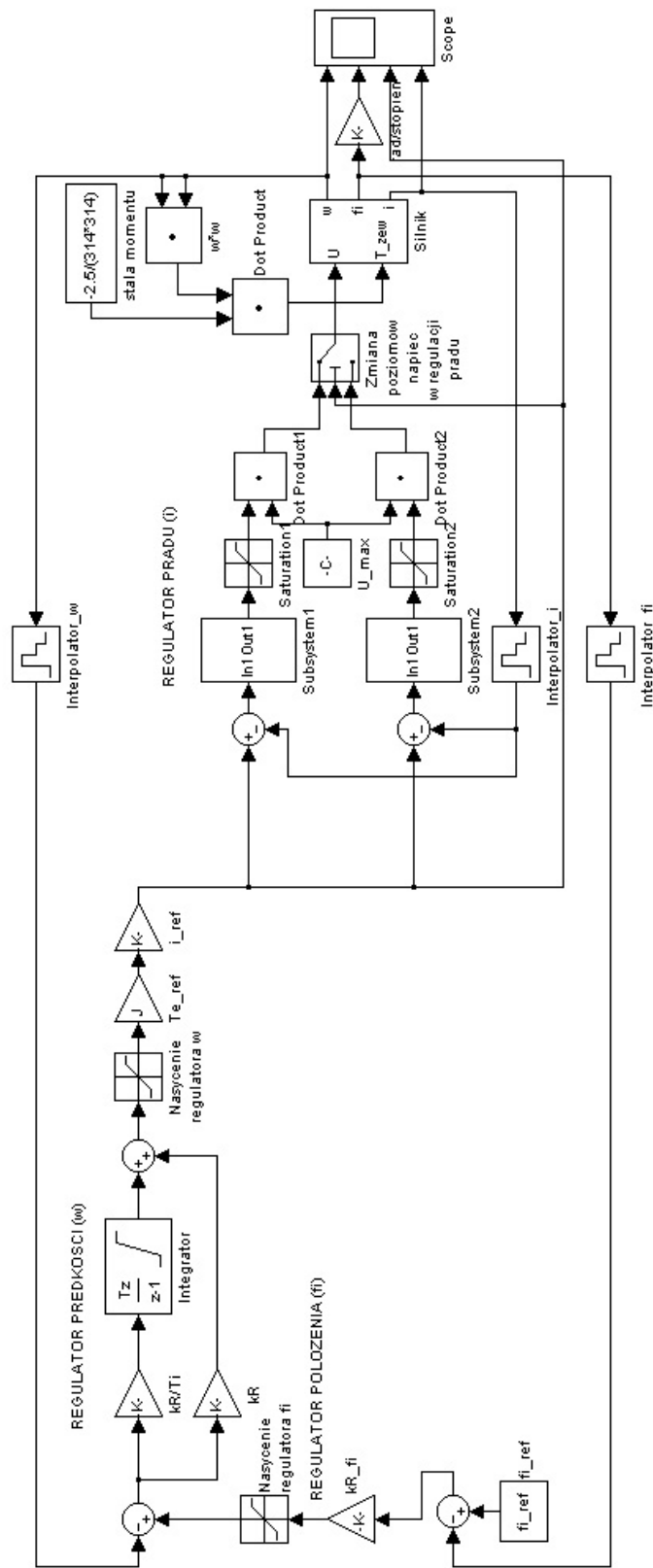
gdzie:

- ω – chwilowa wartość prędkości serwonapędu,
- φ – chwilowa wartość mierzonego kąta położenia serwonapędu,
- φ_{ref} – wartość referencyjna kąta położenia serwonapędu,
- T_{emax} – maksymalny dopuszczalny moment hamujący rozwijany przez silnik,
- J_{max} – założona z góry maksymalna możliwa wartość momentu bezwładności całego napędu, wprowadzonego na poziom prędkości silnika.

Praktycznie przyjęto jako wartość graniczną $0,9a_{max}$ (rys. 4 i 5). Odtwarzanie liniowego przebiegu prędkości podczas hamowania prowadzone jest przez regulator histerezyowy, kontrolujący chwilową wartość a_{hobl} w granicach $(0,98 \div 1,02) a_{max}$. Na wyjściu regulatora pojawia się jako prąd zadany dla regulatora prądu silnika pełny dopuszczalny prąd hamowania, odpowiadający momentowi T_{emax} ($i_{ref} = -i_{max}$) lub prąd zerowy ($i_{ref} = 0$). Regulacja ta prowadzona jest do uzyskania zerowej prędkości, co oznacza zakończenie procesu sterowania. Położenie napędu φ powinno wówczas osiągnąć wartość zadaną φ_{ref} .



Rys. 4. Realizacja symulacyjna układu regulacji z rysunku 3 w pakiecie MATLAB/Simulink



Rys. 5. Realizacja symulacyjna układu regulacji z rysunku 1 w pakiecie MATLAB/Simulink

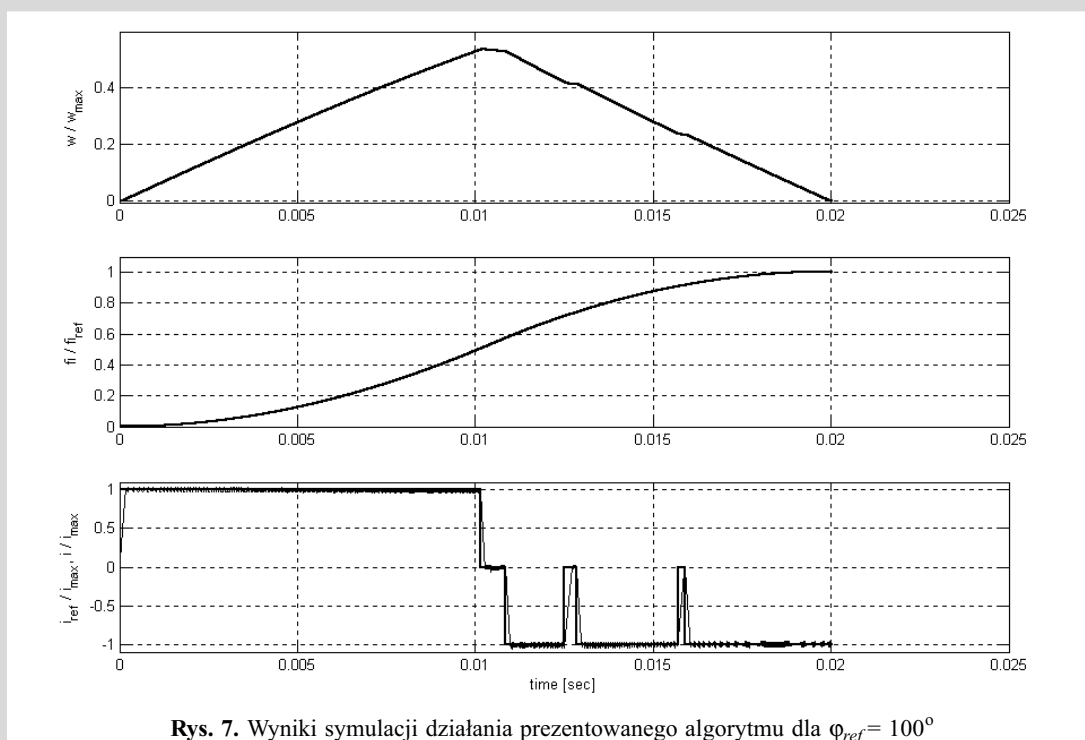
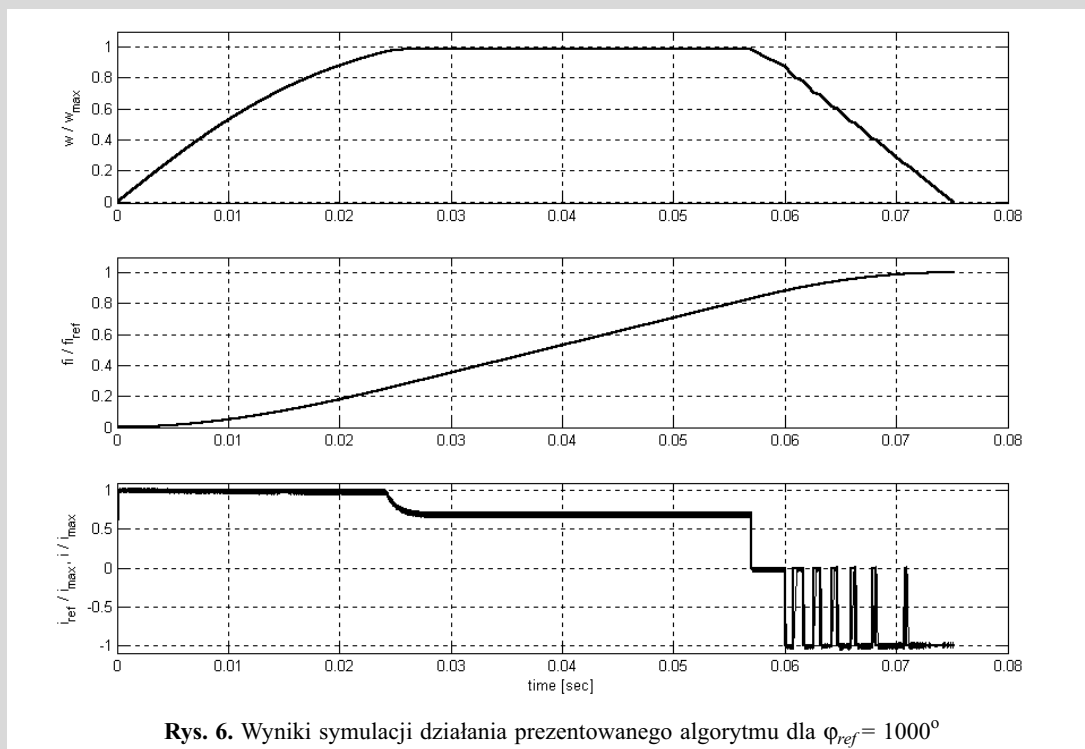
3. BADANIA SYMULACYJNE

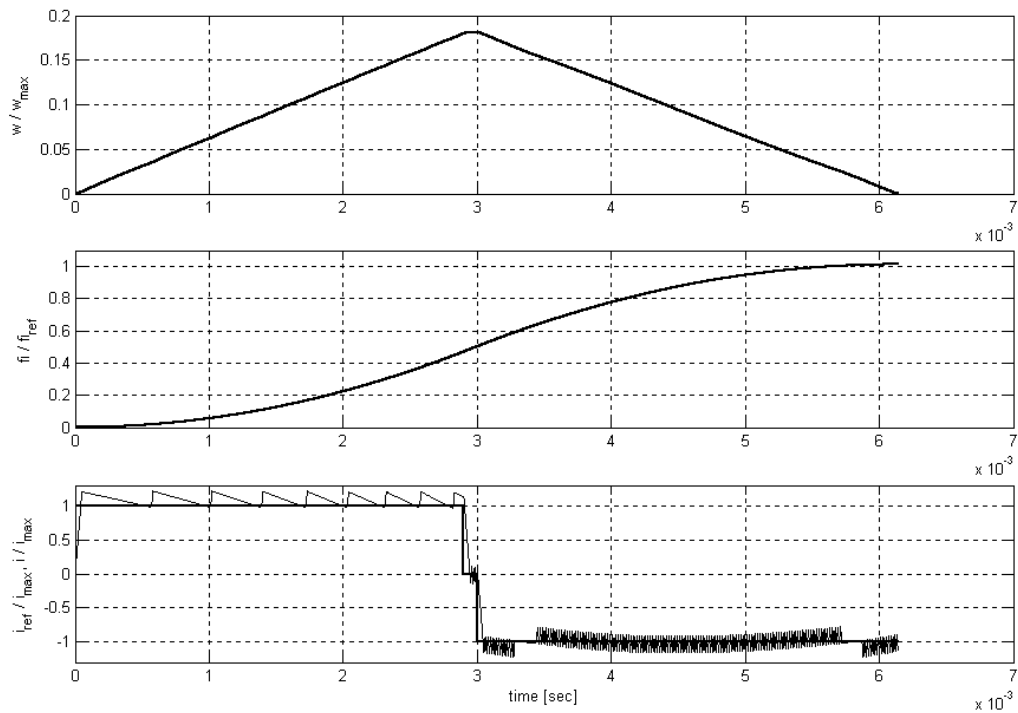
Symulacje działania algorytmu przeprowadzono dla bezszczotkowego silnika prądu stałego o trapezowym przebiegu SEM fazy twornika, skojarzonego w gwiazdę, o danych katalogowych:

napięcie znamionowe DC $U_N = 240$ V,
 indukcyjność dwóch faz $2L = 11,5$ mH,
 rezystancja dwóch faz $2R = 3,89$ Ω ,
 stała czasowa twornika $T_t = 2L/2R = 2,956$ ms,
 stała prądowa momentu $K_t = 0,857$ N·m/A_{sk},

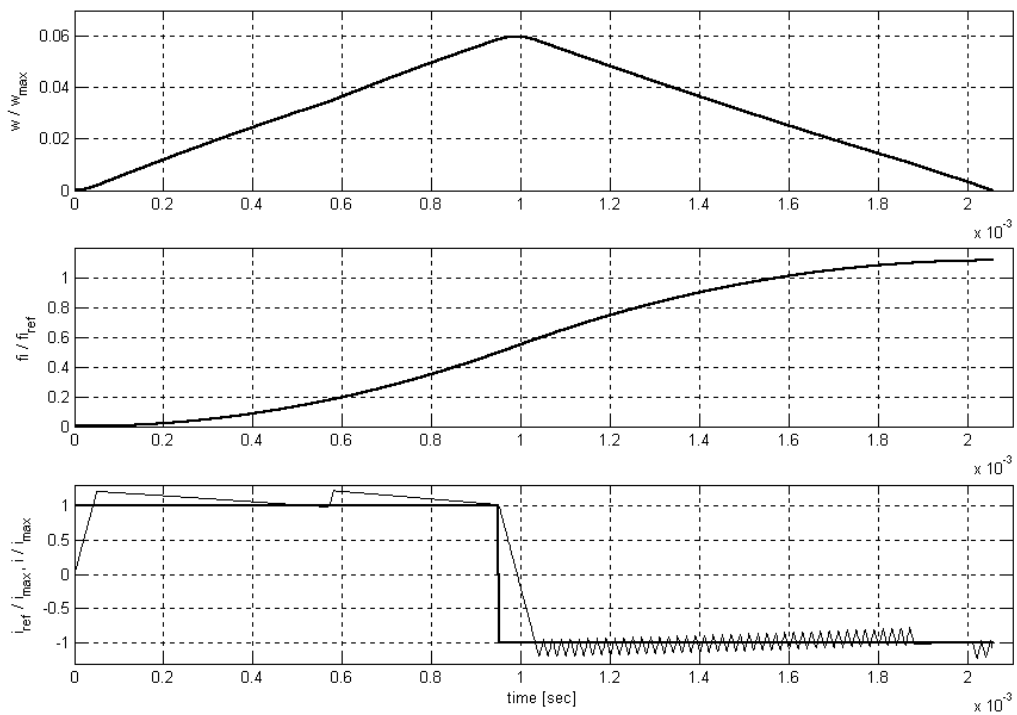
prąd znamionowy skuteczny $I_{Nsk} = 4,2$ A,
 moment znamionowy $T_e = K_t \cdot I_{Nsk} = 3,6$ N·m,
 prędkość znamionowa $n_n = 3000$ obr./min,
 moc znamionowa $P_n = 1130$ W,
 moment bezwładności (silnika) $J = 0,0002$ kg·m².

Przykładowe wyniki symulacji działania algorytmu dla czterech istotnie różnych zadanych kątów obrotu: $\varphi_{ref} = 1000^\circ$, 100° , 10° , 1° , przedstawiono kolejno na rysunkach 6–9. Porównaniem dla nich są wyniki symulacji działania algorytmu z rysunków 1, 5 zamieszczone na rysunkach 10–13.

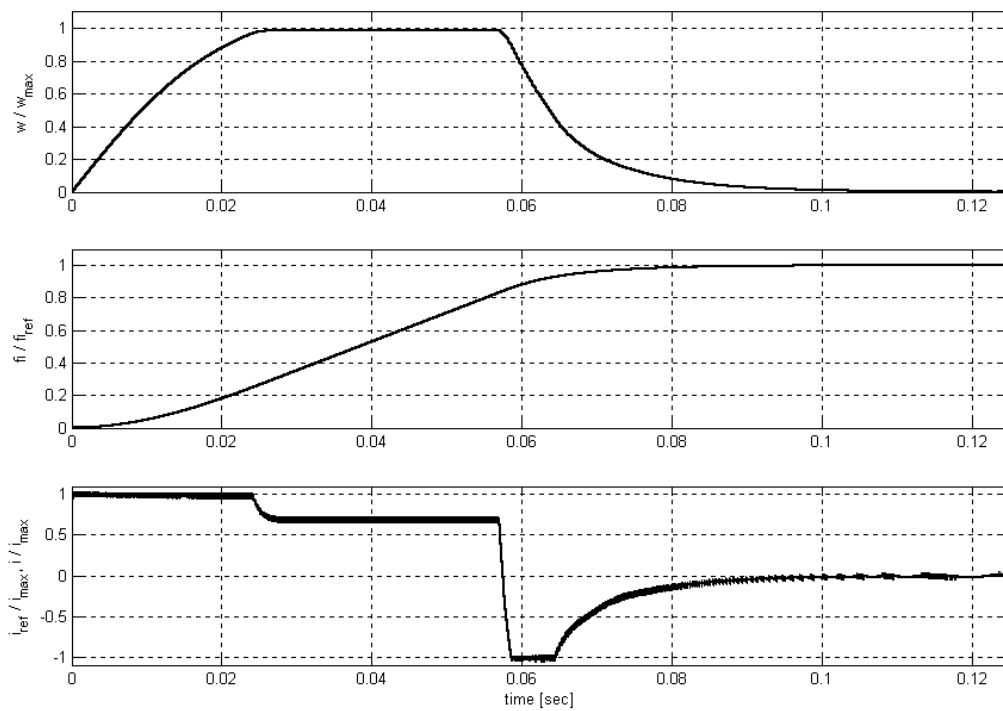




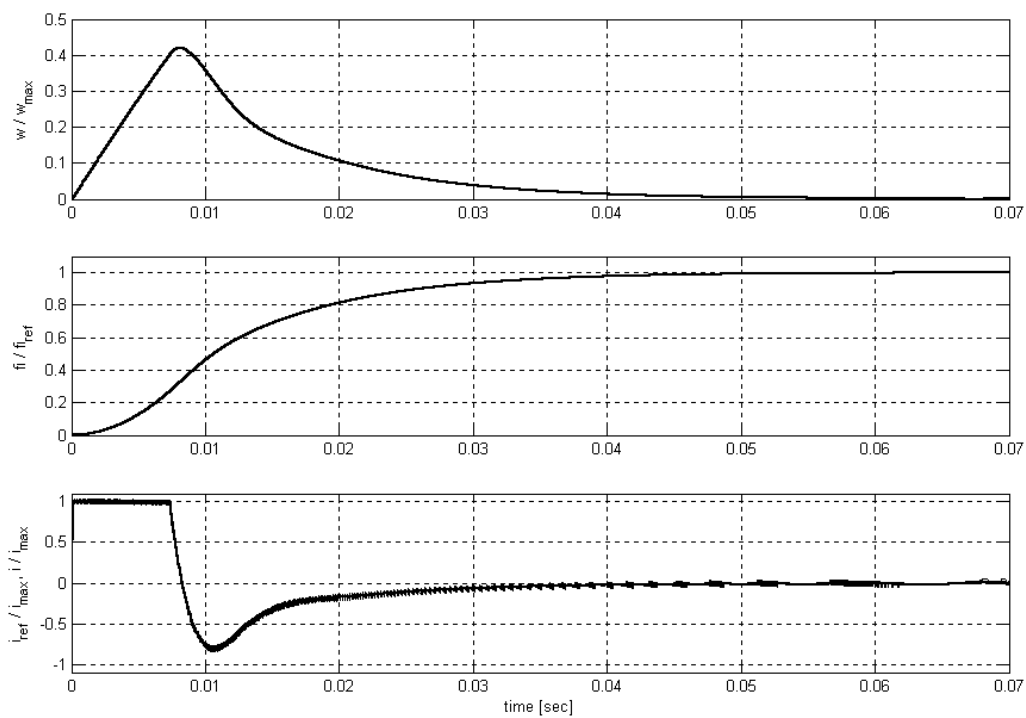
Rys. 8. Wyniki symulacji działania prezentowanego algorytmu dla $\varphi_{ref} = 10^\circ$



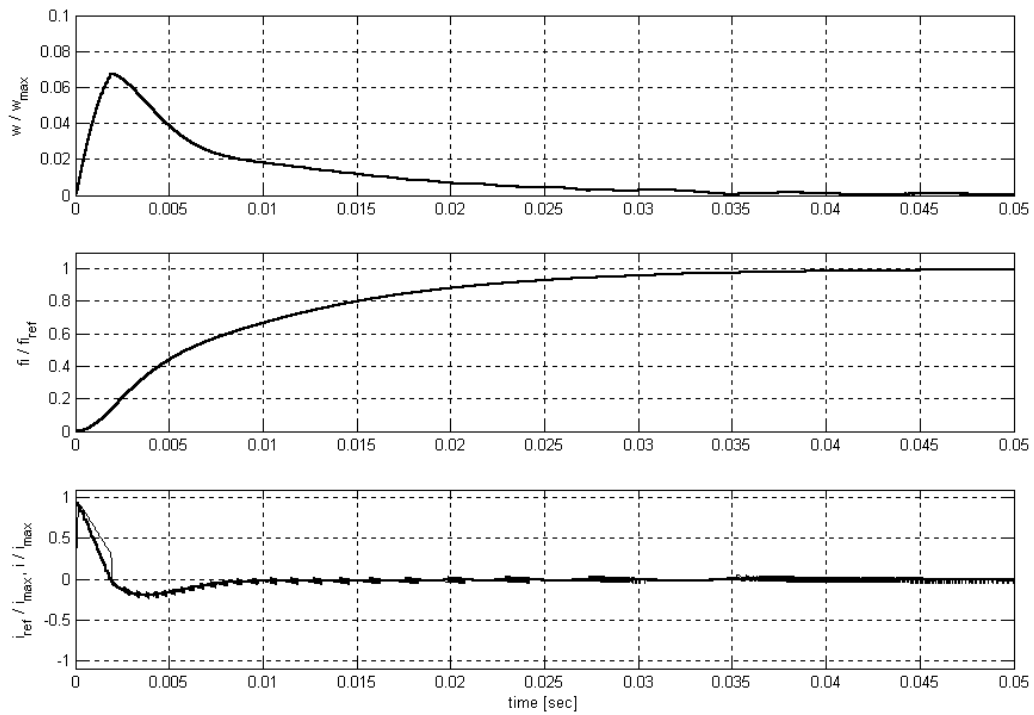
Rys. 9. Wyniki symulacji działania prezentowanego algorytmu dla $\varphi_{ref} = 1^\circ$



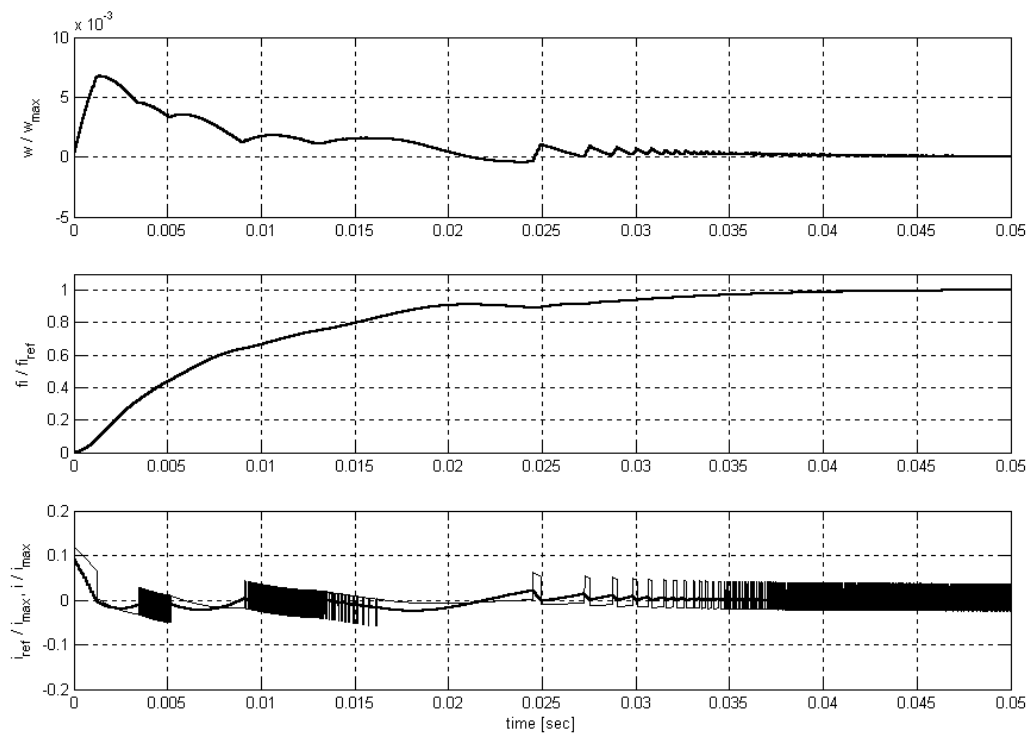
Rys. 10. Wyniki symulacji działania algorytmu porównawczego dla $\varphi_{ref} = 1000^\circ$



Rys. 11. Wyniki symulacji działania algorytmu porównawczego dla $\varphi_{ref} = 100^\circ$



Rys. 12. Wyniki symulacji działania algorytmu porównawczego dla $\varphi_{ref} = 10^\circ$



Rys. 13. Wyniki symulacji działania algorytmu porównawczego dla $\varphi_{ref} = 1^\circ$

Tabela 1

φ_{ref} [deg]	Algorytm badany		Algorytm porównawczy	
	Przeregulowanie [deg]	Czas regulacji [s]	Przeregulowanie [deg]	Czas regulacji [s]
1000	+0,30	0,075	≈ 0	0,125
100	+0,35	0,020	≈ 0	0,070
10	+0,54 dla $U=U_N$ +0,25 dla $U=2U_N$ +0,13 dla $U=4U_N$	0,0063	≈ 0	0,050
1	+0,11 dla $U=4U_N$ +0,012 dla $U=8U_N$	0,00206	≈ 0	0,050

Rysunki 6–13 przedstawiają kolejno: przebieg czasowy prędkości obrotowej silnika w odniesieniu do $\omega_{max} = \omega_N$, przebieg czasowy kąta położenia wirnika silnika w odniesieniu do φ_{ref} , przebiegi czasowe prądu zadane przez regulator prądu i prądu faktycznego, w odniesieniu do $i_{max} = I_{Nsk}$. Rezultaty działania obu algorytmów zebrano porównawczo w tabeli 1. Napięcie U podawane w tabeli to napięcie stałe podawane na silnik *DC brushless* z komutatorem elektronicznym przez histerezowy regulator prądu.

We wszystkich przypadkach zwracają uwagę mniejsze wartości czasu regulacji w badanym algorytmie. Względne przeregulowanie rośnie ze spadkiem wartości φ_{ref} a maleje ze wzrostem napięcia sterującego U . Oczywiście pojawia się tu pytanie, jak duże napięcie można podać na silnik i czy wartość di/dt może być dowolnie duża. Wartość przeregulowania jest również funkcją kroku czasowego realizacji algorytmu. W przedstawianych symulacjach wynosił on 100 μ s dla całego algorytmu, ale tylko 10 μ s dla samego regulatora prądu. Tak krótki okres pracy regulatora prądu sugeruje konieczność praktycznego wykonania go w technice analogowej.

We wszystkich przypadkach skokowe lub płynne zmiany momentu obciążenia i/lub momentu bezwładności silnika na tyle mało wpływają na jakość sterowania (przede wszystkim na wielkość przeregulowania kąta położenia φ), że zrezygnowano z zamieszczenia odpowiednich przebiegów. Zamieszczone rysunki 6–13 zawierają wyniki symulacji dla momentu bezwładności $J = 0,0002 \text{ kg}\cdot\text{m}^2$ i momentu obciążenia T_z kwadratowo zależnego od prędkości obrotowej silnika.

4. WNIOSKI

Zastosowana do hamowania serwonapędu zasada sterowania jest zbliżona do zasady regulacji ślizgowej. Należy zaznaczyć, że algorytmy sterowania nieciągłego (w tym SMC) są bardziej naturalne dla układów energoelektronicznych niż algorytmy ciągłe. Sygnał wyjściowy układu regulacji w tym przypadku jest bezpośrednio wykorzystany do sterowania elementami mocy przekształtnika energoelektronicznego. Jednakże w przypadku, gdy sygnał wyjściowy regulatora ślizgowego jest sygnałem zadaniem dla pozostałych regulatorów (np. prędkości lub momentu silnika), zjawisko szybkich przełączeń generowanych przez ten regulator może doprowadzić do wzbudzenia się drgań wysokiej częstotliwości, wynikających ze stosowania w procesie doboru regulatora uproszczonego modelu obiektu regulacji.

Pojawienie się mikroprocesorowych urządzeń sterowania i zaawansowanych narzędzi symulacyjnych stwarza możliwość opracowywania i implementacji praktycznie całkowicie dowolnych algorytmów sterowania i regulacji automatycznej. Otwiera to pole do tworzenia całkiem nowych, oryginalnych, silnie nieliniowych algorytmów sterowania i regulacji, nierozwijanych do tej pory, z uwagi na trudności obliczeniowe i brak praktycznych możliwości aplikacyjnych. Przedstawiony algorytm sterowania serwonapędu jest tego typu algorytmem – hybrydą klasycznej regulacji PID i regulacji silnie nieliniowej.

Literatura

- [1] Piątek P., Rosół M., Turnau A.: *Sterowanie serwomechanizmem z generatora PWM skonfigurowanego w logice XILINX z kompensacją zniekształceń*. Materiały Konferencyjne III Krajowej Konferencji „Metody i systemy komputerowe w badaniach naukowych i projektowaniu inżynierskim”, AGH, Kraków 19–21.11.2001, 285–290
- [2] Piłat A.: *Adaptacyjny regulator czasu rzeczywistego dla serwomechanizmu DC pracujący w środowiskach MS-Windows i Windows 95*. Materiały Konferencyjne I Krajowej Konferencji „Metody i systemy komputerowe w badaniach naukowych i projektowaniu inżynierskim”, Kraków 25–26.11.1997
- [3] Tutaj A.: *Adaptacyjne sterowanie serwomechanizmem prądu stałego*. Półrocznik Akademii Górniczo-Hutniczej Automatyka, t. 6, z. 2, 2002, 121–138
- [4] Warzała T.: *Algorytmy sterowania serwonapędów elektrycznych*. AGH, Kraków 2006 (praca magisterska)

Wpłynęło: 29.03.2007

Tomasz DRABEK



Urodził się w 1968 roku. Studia ukończył w roku 1991 na Wydziale Elektrotechniki, Automatyki i Elektroniki Akademii Górniczo-Hutniczej w Krakowie (kierunek: elektrotechnika). W roku 1991 rozpoczął pracę w Katedrze Maszyn Elektrycznych AGH. Doktorat uzyskał w roku 1999. Interesuje się głównie badaniami zjawisk elektromagnetycznych, towarzyszących dyskretnemu rozkładowi uzwojeń w uzębionych szczelinach powietrznych maszyn elektrycznych wirujących oraz maszynami elektrycznymi z magnesami trwałymi i ich sterowaniem.

e-mail: Tomasz.Drabek@agh.edu.pl