

Modele symulacyjne w ePEDlab – wirtualnej platformie wspomagającej nauczanie napędu elektrycznego

Streszczenie. Artykuł dotyczy tematyki nauczania napędu elektrycznego za pomocą opracowanej wirtualnej platformy nauczania ePEDlab. Stanowi on uzupełnienie do artykułu prezentującego ideę platformy ePEDlab [1]. W pracy skupiono się na zagadnieniach związanych z środowiskiem do modelowania GeckoCIRCUITS oraz opracowanymi modelami symulacyjnymi. Przedstawiono tu właściwości oprogramowania GeckoCIRCUITS, opisano udostępnione za pomocą platformy modele symulacyjne, a następnie, dla wybranych dwóch przykładów, przedstawiono program ćwiczeń symulacyjnych i dostępne w GeckoCIRCUITS narzędzia do analizy wyników symulacji.

Abstract. The article presents selected aspects of electric drive teaching methods using ePEDlab – the developed virtual educational platform. It is a supplement to an article presenting the idea of an ePEDlab platform [1]. The work focuses on issues related to GeckoCIRCUITS software and developed simulation models. It presents GeckoCIRCUITS software features and provided by the platform simulation models. Then two selected models are described. (*Simulation models in ePEDlab - a virtual platform supporting electric drive teaching*).

Słowa kluczowe: dydaktyka, modelowanie, napęd elektryczny, wirtualna platforma edukacyjna

Keywords: electrical drives learning, simulation models, e-learning platform

Wstęp

Napędy elektryczne stanowią jeden z najpowszechniej występujących w przemyśle rodzajów odbiorników. Z tego względu od studentów kończących studia I i II stopnia na wydziałach o profilach elektrycznych wymaga się znajomości zagadnień z zakresu napędu elektrycznego. Wiedza ta powinna być stosunkowo szeroka i obejmować zarówno właściwości samych silników, ich układów zasilania oraz algorytmów sterowania. Dodatkowo przydatna jest wiedza z zakresu parametryzacji rozwiązań przemysłowych. Zastosowanie w napędach przekształtników energoelektronicznych oraz zaawansowanych metod sterowania implementowanych w sterownikach mikroprocesorowych umożliwiło zdecydowaną poprawę ich właściwości statycznych: sterowanie parametrami urządzeń napędzanych jak i często zmniejszenie energochłonności napędu, oraz dynamicznych: uzyskanie pełnej kontroli nad pracą napędu w stanach przejściowych, realizację algorytmów ze sterowaniem polowo-zorientowanym i szybkiego pozycjonowania). Niestety z drugiej strony spowodowało to zwiększenie negatywnego wpływu napędu na sieć zasilającą oraz znacząco podniosło stopień komplikacji układu jako całości. Aby w pełni zrozumieć współczesne rozwiązania napędowe nie wystarczy sama wiedza teoretyczna, czy nawet badania laboratoryjne napędu pod kątem jego właściwości ruchowych, np. charakterystyk mechanicznych. W celu dogłębnego poznania możliwości bardziej złożonych rozwiązań napędowych, należy skupić się głównie na realizacji algorytmu sterowania. Najlepszym rozwiązaniem jest możliwość samodzielnej implementacji algorytmu sterowania napędu i przebadanie go w rzeczywistym układzie [2]. Ograniczeniem może być dostęp do laboratorium wyposażonego w odpowiednie przekształtniki i maszyny, a także aparaturę pomiarową. Na szczęście pewną część tych informacji można pozyskać poprzez badania symulacyjne napędów elektrycznych. Z tego względu w ramach wirtualnej platformy do nauki napędu elektrycznego ePEDlab, przy użyciu oprogramowania GeckoCIRCUITS, opracowane zostały modele symulacyjne pozwalające na badanie typowych układów napędowych [3]. Modele zamieszczono na stronie internetowej projektu w formie Apletów Javy, co oznacza, że do ich działania nie jest konieczne posiadanie programu GeckoCIRCUITS, wymagana jest jedynie obecność w systemie operacyjnym JRE (ang. *Java Runtime*

Environment – środowisko uruchomieniowe Javy). Środowisko to jest często zainstalowane we współczesnych komputerach osobistych, a zatem uruchamianie modeli symulacyjnych jest proste i może odbywać się bezpośrednio z poziomu przeglądarki internetowej, bez konieczności podejmowania dodatkowych czynności. Przygotowane modele odpowiadają ćwiczeniom laboratoryjnym prowadzonym w Laboratorium Energoelektronicznych Układów Napędowych na Wydziale Elektrycznym Politechniki Śląskiej, a zakresem obejmują one większość współczesnych rozwiązań napędowych. Modele te pozwalają szczegółowo zapoznać się z możliwościami i specyficznymi właściwościami układów napędowych. Pozwalają one zarówno badać właściwości mechaniczne i elektryczne napędów, jak i zapoznać się z algorytmem i sygnałami występującymi w układzie sterowania oraz w wybranych układach również z wpływem napędów na sieć zasilającą. Pliki symulacyjne podzielone zostały na dwie grupy, odpowiadające zagadnieniom związanym z I i II stopniem studiów. Obejmują one swoim zakresem: napędy z silnikami obcowzbudnymi prądu stałego (zasilane z prostownika tyrystorowego, przekształtnika tranzystorowego czterokwadrantowego oraz problematykę doboru nastaw regulatorów), napęd z silnikiem BLDC, napęd z silnikiem PMSM, serwonapęd z silnikiem PMSM, napędy z silnikami indukcyjnymi klatkowymi (układy rozruchu, sterowanie skalarne, sterowanie polowo-zorientowane bezpośrednio DFOC i pośrednie IFOC, bezpośrednie sterowanie momentem DTC, bezpośrednie sterowanie momentem z modulatorem wektorowym DTC-SVM, odtwarzanie strumienia/prędkości), skompensowane przekształtniki częstotliwości AC/DC/AC, oraz napęd z maszyną asynchroniczną dwustronnie zasilaną. Dla każdego z modeli symulacyjnych opracowano instrukcję pozwalającą na samodzielne przeprowadzanie badań symulacyjnych.

W dalszej części artykułu skupiono się na oprogramowaniu symulacyjnym GeckoCIRCUITS [4] oraz szczegółowym omówieniu możliwości symulacyjnych wybranych modeli napędów.

Oprogramowanie GeckoCIRCUITS

Oprogramowanie GeckoCIRCUITS, proponowane przez firmę Gecko-Simulations, zorientowane jest głównie na modelowanie wybranych zjawisk w układach energoelektronicznych, ale pozwala również na modelowanie na-

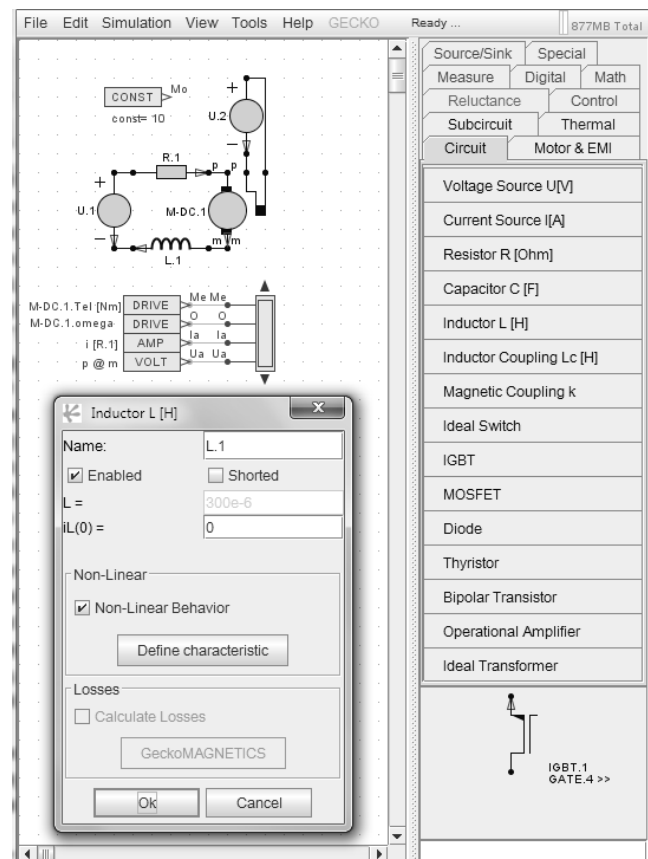
pędów elektrycznych oraz układów elektrycznych i elektronicznych [5,6]. Podobnie jak Matlab-Simulink, Simplorer, P-SIM czy LabVIEW należy do programów wykorzystujących języki graficzne w procesie tworzenia modeli symulacyjnych. Istotną zaletą programu jest to, że do niedawna dystrybuowany był również w wersji darmowej, a aktualnie uniwersytecka pełna wersja tego programu kosztuje 540 euro rocznie, przy nieograniczonej liczbie licencji.

Oprogramowanie GeckoCIRCUITS nie wymaga instalacji i zajmuje mało miejsca na dysku w porównaniu do innych programów symulacyjnych – poniżej 50 MB, z czego około 5,5 MB stanowią dostarczone przykładowe modele symulacyjne. Dodatkowo wymagane jest również zainstalowanie na komputerze środowiska uruchomieniowego Javy. Istotną cechą oprogramowania GeckoCIRCUITS jest również duża szybkość wykonywania obliczeń, co związane jest ze stosunkowo prostymi modelami zaworów energoelektronicznych, nie uwzględniającymi ich dynamiki. W modelach zaworów możliwe jest jedynie określanie ich parametrów statycznych, np. spadków napięcia na diodzie i tranzystorze. Uproszczenie modelu przekształtnika, polegające na braku uwzględnienia dynamiki przełączania zaworów, nie powoduje zbytnej utraty informacji z punktu widzenia analizy pracy przekształtnika jako całości (np. przy projektowaniu układu sterowania przekształtnika przeznaczonego do napędu). Uwzględnienie dynamiki zaworów w przypadku modelowania napędów przekształtnikowych prowadzi do niepotrzebnego wydłużenia czasu trwania symulacji. Tak dzieje się na przykład w oprogramowaniu Matlab-Simulink, gdzie na czas wykonywania obliczeń oraz uzyskane wyniki niebagatelny wpływ ma wartość rezystancji i pojemności w układzie tłumienia prądów (snubber). W celu przyspieszenia obliczeń można w programie Matlab-Simulink zastosować model przekształtnika będący źródłem napięcia sterowanym stanami tranzystorów (switching-function based VSC), jednak powoduje to utratę możliwości badania przekształtnika w stanach awaryjnych (np. uszkodzony jeden tranzystor z działającą diodą zwrotną). Możliwe jest jednak wykorzystanie modeli opracowanych w oprogramowaniu GeckoCIRCUITS bezpośrednio w programie Matlab-Simulink.

Ciekawą właściwością programu GeckoCIRCUITS jest integracja w jednym modelu, zawierającym przekształtnik energoelektroniczny, części elektrycznej i sterowania, a także zagadnień cieplnych i magnetycznych. Dzięki temu możliwa jest na przykład analiza strat mocy w przekształtnikach, jak i badanie procesu nagrzewania. Poprzez kształtowanie charakterystyk modeluje się zarówno straty mocy przewodzenia, jak i straty mocy przełączania zaworów energoelektronicznych. Przykładową analizę z wykorzystaniem modelowania strat mocy przedstawiono w [7].

Na rysunku 1 przedstawiono wygląd podstawowego okna z modelem przygotowanym w programie GeckoCIRCUITS. Po prawej stronie widoczne są zakładki do wyboru elementów elektrycznych i energoelektronicznych, elementów układu sterowania, termicznych (Thermal) i magnetycznych (Reluctance). Poniżej zestawione zostały elementy dostępne w zakładce Circuit, zawierającej elementy układów elektrycznych i energoelektronicznych. W oknie projektu przedstawiono przykładowy obwód elektryczny z silnikiem obcowzбудnym prądu stałego. Łączenie elementów w oknie schematu odbywa się za pomocą myszki (prawy klawisz), przy czym należy pamiętać o występowaniu odpowiednich interfejsów do łączenia elektrycznych obwodów mocy z obwodami sterowania (np. widoczne układy pomiarowe oraz układy sterowania tranzystorów). Istnieje również możliwość

zastąpienia połączenia w formie linii etykietą nadawaną dowolnemu z zacisków danego elementu. Pozwala to uprościć schemat całego układu. Atutem jest również to, że w zakładce do tworzenia układów sterowania znajdują się takie elementy jak: układy transformacji współrzędnych, układy wyzwalania zaworów czy układy do wizualizacji wektorów przestrzennych. Na rysunku 1 pokazano również zakładkę do ustawiania parametrów dławika. Możliwe jest tu nadanie wartości indukcyjności i warunku początkowego prądu oraz zdefiniowanie nieliniowej charakterystyki opisanej w postaci indukcyjność-prąd. Na górnej listwie znajdują się zakładki do sterowania procesem symulacji, wyglądem i tworzeniem modelu. Program wyposażony jest dodatkowo w pakiet GeckoSCRIPT (zakładka GECKO) pozwalający na proste sterowanie procesem symulacji oraz wstępne ustawianie parametrów symulacji. Pozwala to np. nadawać wartości poszczególnym parametrom elementów w modelu symulacyjnym (polecenie setParameter) oraz uruchamianie symulacji (runSimulation()). GeckoSCRIPT przypomina przygotowywanie m-plików do obliczeń prowadzonych w programie Matlab-Simulink. Pod schematem obwodów elektrycznych znajduje się oscyloskop pozwalający na wyprowadzanie wyników symulacji. Mimo, że samo ustawianie parametrów związanych z wizualnym wyglądem wyników symulacyjnych jest stosunkowo skomplikowane, należy zaznaczyć, że w programie przygotowane zostało menu pozwalające na np. zebranie danych w tablicach w postaci czas-wartość, transformata Fouriera czy analiza parametrów jakości energii elektrycznej. Jako wadę można uznać brak wystarczającej dokumentacji i pomocy w programie.



Rys. 1. Przykładowe okno programu GeckoCIRCUITS

W programie GeckoCIRCUITS możliwa jest realizacja algorytmu sterowania poprzez tworzenie schematu graficznego z elementów odpowiadających typowym funkcjom lub poprzez wykorzystanie bloków do przygotowywania kodu w

języku JAVA (Java-function) lub języku C (C/C++ Function). Ze względu na fakt, że do działania programu GeckoCIRCUITS niezbędna jest instalacja środowiska JAVA wraz z bibliotekami, lepszym rozwiązaniem jest stosowanie bloku Java-function. Na rysunku 2 pokazano użycie tego bloku na przykładzie dyskretnej realizacji regulatora PI. W przypadku bloku Java-function wymagane jest przygotowanie kodu, jednak kod ten jest prosty i przejrzysty. Stosowanie takiego bloku umożliwi większą elastyczność (np. wprowadzanie nieliniowych współczynników, blokowanie części całkującej itd.). Opracowany w ten sposób kod może być bazą dla późniejszej mikroprocesorowej realizacji sterowania. W przypadku bloku Java-function należy zachować określoną składnię, gdzie xIN[n] oznacza n-te wejście bloku, a yOUT[m] oznacza m-te wyjście bloku. Dodatkowo możliwe jest wykorzystanie aktualnego czasu symulacji time oraz kroku symulacji dt. W pliku Java-function tworzone są zmienne globalne, którym mogą być nadawane początkowe wartości.

Rys.2. Okno bloku Java-Function odpowiadające regulatorowi PI

W dalszej części przedstawione zostaną, opracowane w darmowej wersji oprogramowania GeckoCIRCUITS, przykładowe modele układów napędowych. Wynikowe pliki, zapisane w formie Apletów, zawierają w sobie zarówno

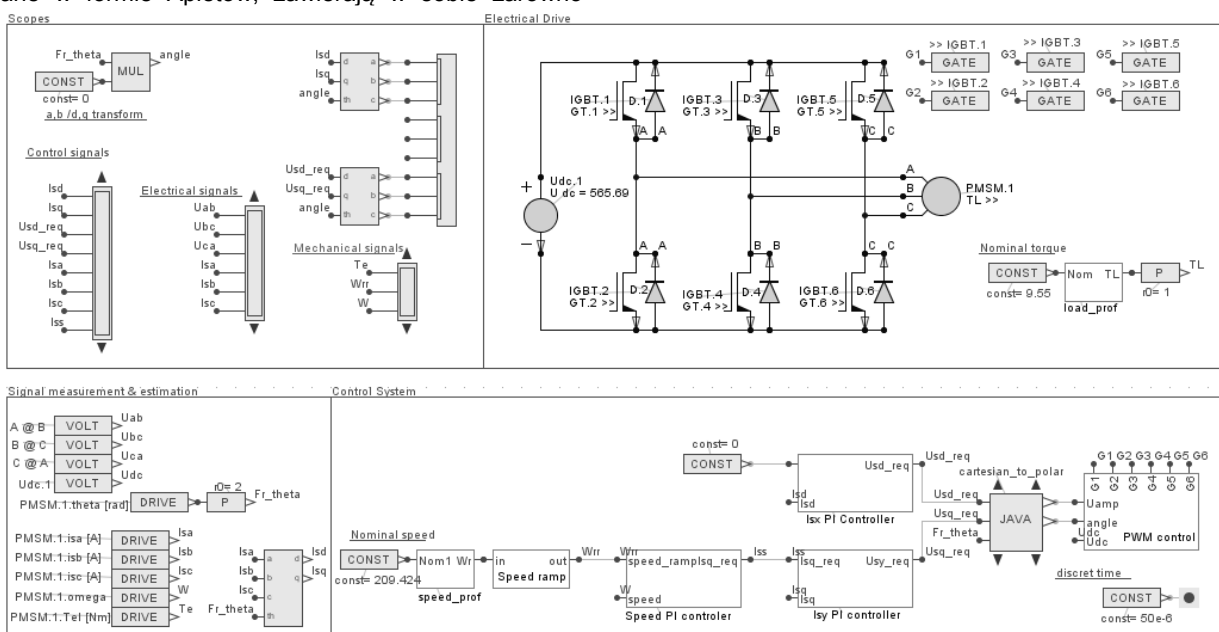
same modele, jak i narzędzia pozwalające na przeprowadzenie symulacji, zapoznanie się z wynikami oraz dokonywanie zmian wartości parametrów. Wadą jest brak możliwości podglądania wnętrza bloków Java-Function i zapisu zmienionych plików.

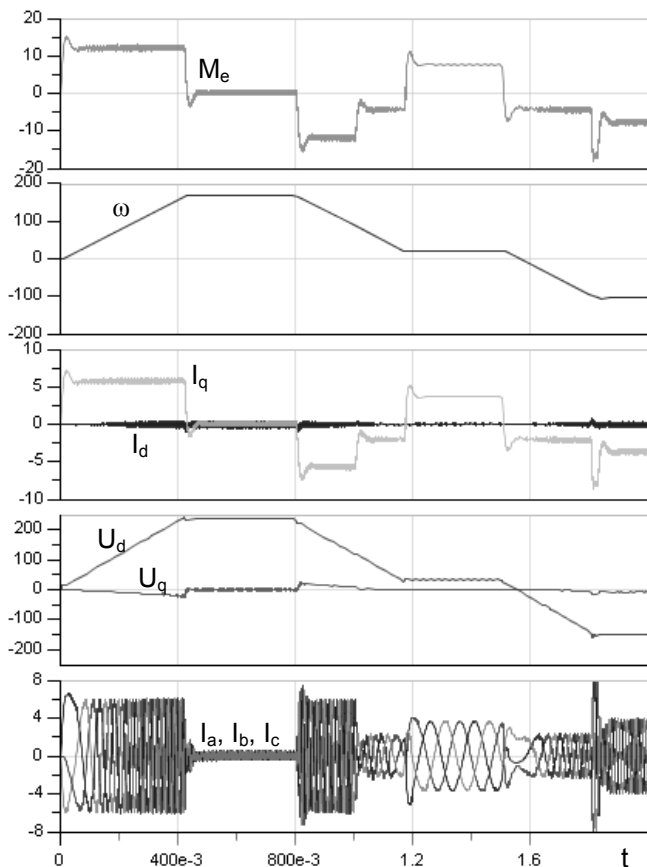
Napęd z silnikiem PMSM

Jako pierwszy zostanie omówiony model symulacyjny z silnikiem synchronicznym o magnesach trwałych PMSM. Na rysunku 3 pokazano model symulacyjny, zbudowany wg przyjętego wzorca. W lewej górnej części umieszczono bloki związane z wizualizacją wyników symulacji, w prawej górnej części – obwody mocy przekształtnika wraz z blokiem odpowiedzialnym za kształtowanie profilu obciążenia *load_prof*, w dolnym lewym rogu – bloki związane z pomiarami i układem transformacji współrzędnych dla prądu, natomiast w prawym dolnym rogu znajduje się układ sterowania napędu.

Układ sterowania ma typową strukturę kaskadową z wewnętrznymi pętlami regulacji składowych prądu silnika, nadrzędnym regulatorem prędkości, ogranicznikiem szybkości zmian prędkości zadanej oraz blokiem odpowiedzialnym za generację profilu prędkości zadanej *speed_prof*. Układ regulacji został zrealizowany w wersji dyskretnej z czasem dyskretyzacji równym 50 μ s. Bloki regulatorów, modulatora oraz profili obciążeń i prędkości zadanych złożone są z podukładów, zawierających nieedytowalne bloki Java-Function i stałe, pozwalające na ich parametryzację.

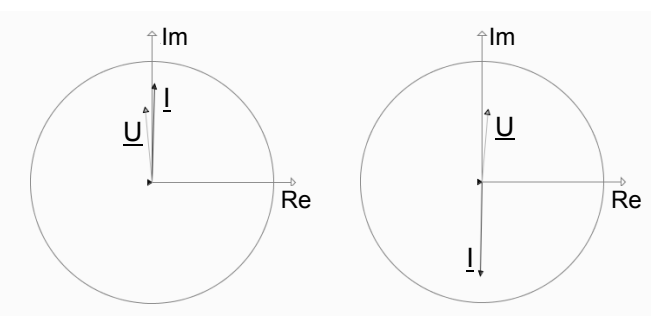
W ramach badań symulacyjnych student może zapoznać się z pracą napędu w różnych stanach statycznych (przy wybranych wartościach prędkości i momentu obciążenia), lub zadać parametry profilu zmian prędkości i momentu obciążenia. Następnie możliwe jest sprawdzenie wpływu nastaw regulatorów i ograniczeń na właściwości dynamiczne napędu. Przykładowe przebiegi wielkości mechanicznych i elektrycznych, a także wielkości występujących w układzie sterowania, pokazano na rysunku 4. Proces obejmuje rozruch do prędkości równej 80 % wartości znamionowej, a następnie pracę z prędkością równą 10 % i -50 % wartości znamionowej, ze skokowymi zmianami momentu obciążenia wynoszącymi 80 % i -80 % wartości znamionowej.





Rys.4. Wyniki badań symulacyjnych dla modelu z silnikiem PMSM (od góry: moment elektromagnetyczny, prędkość rzeczywista, składowe prądy d,q, składowe napięcia zadanych d,q, prądy fazowe silnika)

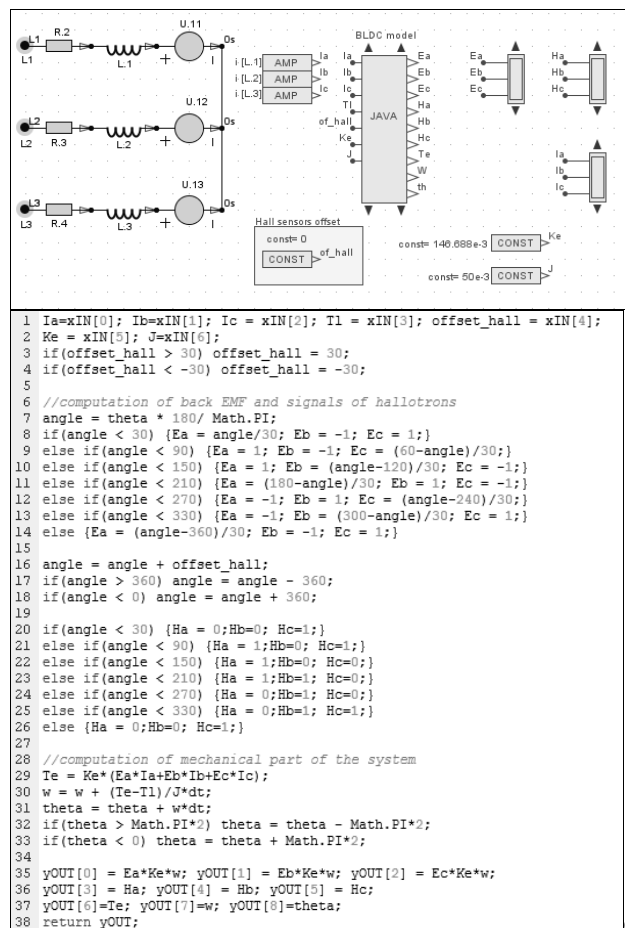
Na rysunku 3 w lewym górnym rogu pokazano bloki transformacji współrzędnych $a,b/d,q$ transform, pozwalające na analizę wektorów przestrzennych w układzie wirującym d,q lub stacjonarnym α,β . Wizualizacje wektorów przestrzennych dla pracy napędowej i hamulcowej pokazano na rysunku 5.



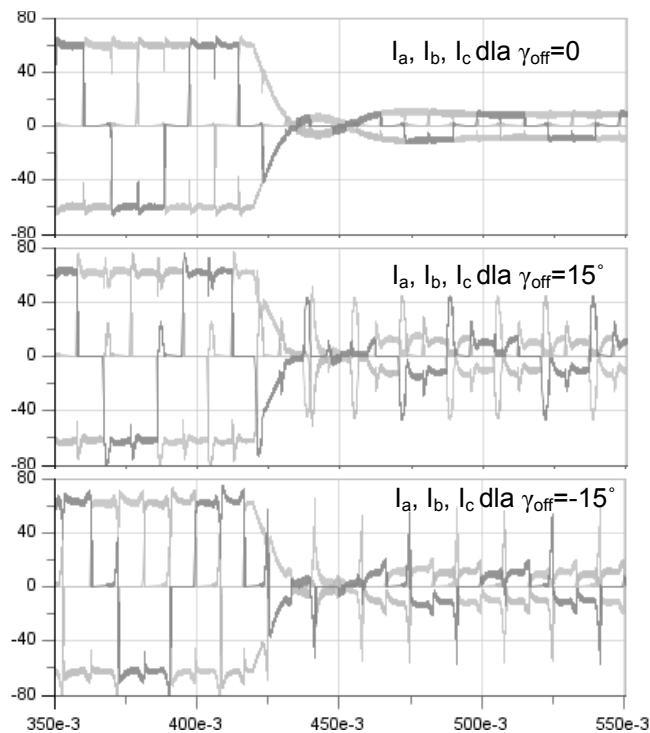
Rys. 5. Wektory przestrzenne prądu (czerwony) i napięcia (zielony) dla pracy napędu przy napędzaniu i hamowaniu w układzie wirującym d,q zorientowanym względem magnesów wirnika.

Napęd z silnikiem BLDC

W programie GeckoCIRCUITS brak jest gotowego modelu silnika BLDC (Brushless DC). Z tego względu w ramach realizacji projektu opracowano taki model z wykorzystaniem bloku Java-Function. Na rysunku 6 pokazano model elektryczny składający się z rezystancji, indukcyjności i sterowanych źródeł napięcia, odtwarzających siły elektromotoryczne oraz kod zawarty w bloku Java-Function.



Java-Function BLDC model.

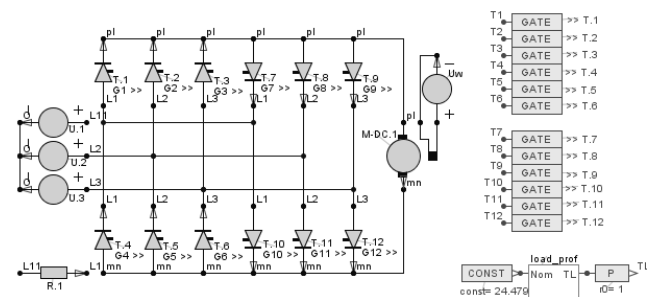


Rys.7. Przebiegi prądów silnika BLDC przy poprawnym ustawieniu hallotronów i przesunięciu hallotronów o 15° w obie strony
Opracowany model silnika pozwala na kształtowanie sił elektromotorycznych w funkcji kąta położenia wału, generuje sygnały hallotronów oraz wielkości mechaniczne.

Istotną jego właściwością jest możliwość wprowadzania przesunięcia położenia hallotronów *hall sensor offset* i badania wpływu tego przesunięcia na pracę napędu. Na rysunku 7 pokazano przebiegi prądów silnika BLDC przy prawidłowym ustawieniu czujników położenia oraz przy zbyt wczesnej i zbyt późnej komutacji zaworów w stosunku do siły elektromotorycznej (przesunięcie o +/-15°). Widoczne są w tym przypadku charakterystyczne dodatkowe uderzenia prądu wynikające z braku możliwości kompensacji wpływu sił elektromotorycznych.

Napęd z silnikiem obcowzbudnym prądu stałego zasilanym z prostownika tyrystorowego

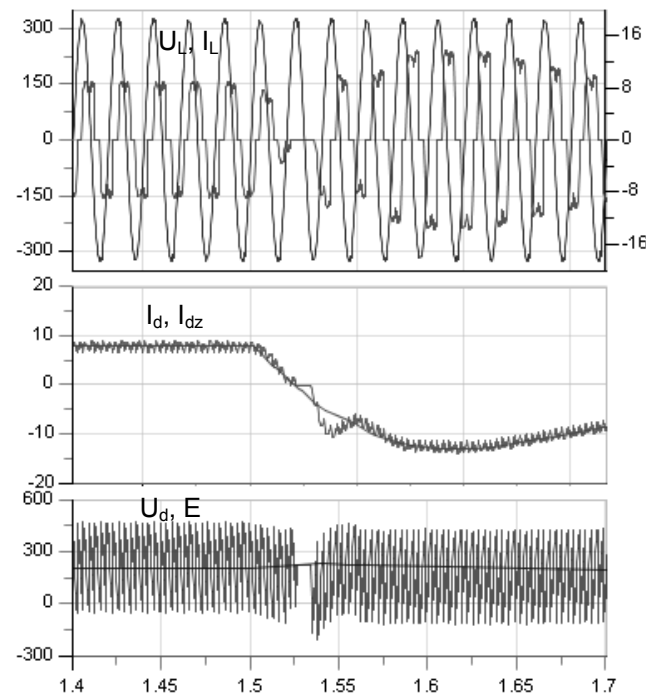
Jako ostatni omówiony zostanie model napędu z silnikiem obcowzbudnym prądu stałego, zasilanym z prostownika tyrystorowego. Obwody mocy modelu tego napędu pokazano na rysunku 8. Jest to rozwiązanie napędu nawrotnego z dwoma prostownikami tyrystorowymi o sterowaniu rozdzielonym. Przygotowanie modelu wymagało opracowania układów wyzwalania tyrystorów, ponieważ dostępne w programie GeckoCIRCUITS bloki nie zapewniały poprawnej pracy prostowników. Dodatkowo opracowano w modelu układ zapewniający rozdzielone sterowanie każdym z prostowników z wprowadzeniem przerwy bezprądowej, niezbędnej ze względu na potrzebę odzyskania właściwości zaworowych przez tyrystory. Na rysunku 9 pokazano przebiegi napięcia i prądu sieci, prądu zadanego i rzeczywistego oraz siły elektromotorycznej i napięcia wyjściowego z prostownika dla zmiany kierunku momentu obciążenia. W przebiegach widoczne są związane z tym zmiany przesunięcia między napięciem i prądem, chwila wyłączenia obu prostowników oraz zmiany kształtu napięcia wyjściowego z prostownika, przy przejściu z pracy prostownikowej jednego prostownika do falownikowej drugiego.



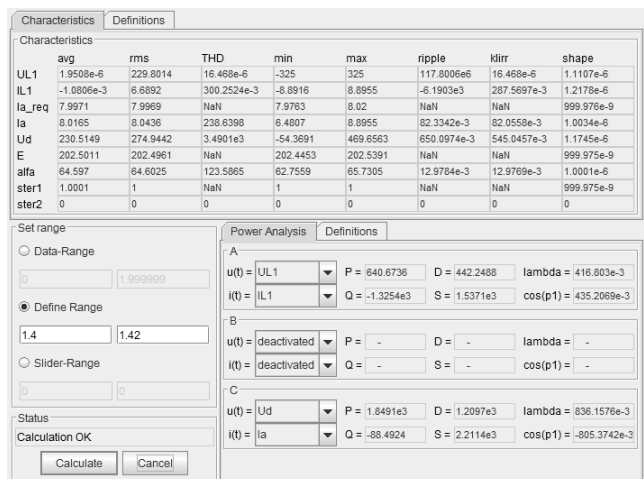
Rys.8. Obwody mocy modelu napędu z silnikiem obcowzbudnym zasilanym z prostownika tyrystorowego

W przypadku symulacji energoelektronicznych układów napędowych ważna jest analiza uzyskanych wyników, którymi są często przebiegi odkształcone (rys. 9). Oprogramowanie GeckoCIRCUITS posiada wbudowane narzędzie służące do tego celu (rys. 10). Jest ono dostępne w menu elementu *Scope*, służącego do wizualizacji przebiegów. Znajduje się tam zakładka *Analysis/Characteristics*, pozwalająca na wyznaczenie wartości średnich, skutecznych, współczynnika THD i innych współczynników określających właściwości wszystkich wielkości wizualizowanych na oscylogramach. Dodatkowo, przy zdefiniowaniu danych przebiegów jako napięcia i prądy, wyznaczane są moce: czynne, bierna, pozorne, deformacji oraz współczynniki mocy λ i $\cos\phi$ dla podstawowej harmonicznej. W przypadku wyznaczenia wszystkich współczynników ważne jest prawidłowe określenie zakresu czasu do analizy w celu określenia w programie częstotliwości podstawowej harmonicznej. Na rysunku 10 pokazano analizę przebiegów dla pracy prostownikowej napędu tyrystorowego. Uzyskany współczynnik THD prądu sieci ma wartość 30%, co jest wartością oczekiwaną. Analiza mocy pozwala określić przekazywaną moc

czynną, bierną i odkształceń w każdej fazie niezależnie, co stwarza dodatkowe możliwości - na rysunku 10 pokazano możliwość wykorzystania tego narzędzia do wyznaczania sprawności układu napędowego. Do fazy A przypisano przebiegi wejściowe jednej fazy prostownika, a do fazy C przypisano napięcie i prąd wyjściowy. W takim przypadku możliwa jest jedynie analiza mocy czynnej.



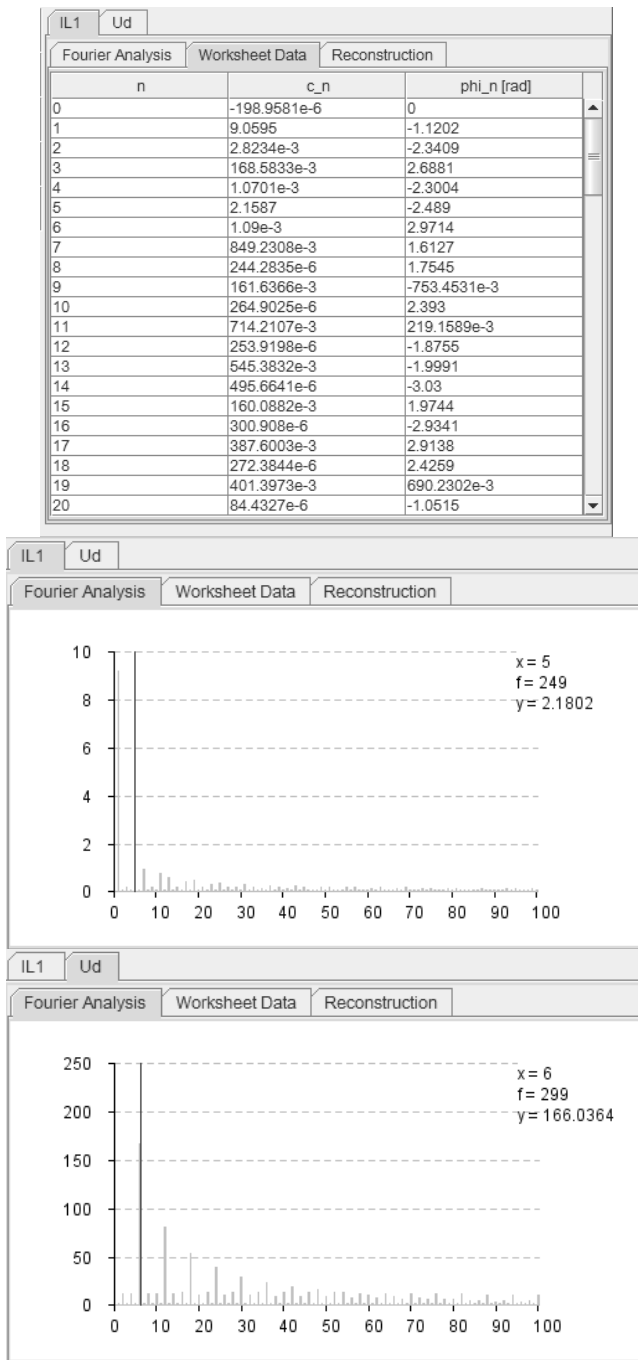
Rys.9. Przebiegi dla inwersji momentu: napięcie i prąd sieci, zadany i rzeczywisty prąd silnika, siła elektromotoryczna i napięcie silnika



Rys.10. Narzędzie do analizy przebiegów odkształconych w oprogramowaniu GeckoCIRCUITS

W menu elementu *Scope* znajduje się również zakładka *Analysis/Fourier*, pozwalająca na analizę harmonicznych przebiegów prezentowanych na oscyloskopie, co pokazano na rysunku 11. W tym przypadku również należy poprawnie określić przedział czasowy oraz liczbę analizowanych harmonicznych. Wynik analizy Fouriera przedstawiony jest w postaci słupkowej, możliwe jest również przedstawienie wyników w formie tabeli. Na rysunku 11 pokazano rozkład harmonicznych prądu sieci w postaci słupkowej i tabelarycznej oraz w postaci słupkowej rozkład harmonicznych

napięcia wyjściowego. Widoczne jest tu, zgodnie z oczekiwaniami, występowanie głównie 5 i 7 harmonicznej w prądzie sieci oraz dużego udziału 6 harmonicznej w napięciu wyjściowym prostownika.



Rys.11. Narzędzie do analizy harmonicznych w oprogramowaniu GeckoCIRCUITS

Podsumowanie

W artykule przedstawiono stosunkowo nowe oprogramowanie GeckoCIRCUITS jako alternatywę dla innych, popularnych i powszechnie uznawanych narzędzi symulacyjnych. Oprogramowanie nadaje się znakomicie do modelowania energoelektronicznych układów napędowych. Środowisko to jest wygodne w użyciu, łatwe w obsłudze oraz wyróżnia się dużą szybkością działania. Wyposażone zostało w przydatne biblioteki elementów oraz bogaty zestaw narzędzi do wizualizacji i analizy wyników. O jego zaletach świadczy również niska cena (do niedawna oprogramowanie było darmowe). Możliwość zapisania zbudowanego modelu w postaci Apletów Javy pozwala na łatwe osadzenie go na stronie internetowej, co wykorzystano w ramach projektu ePEDlab.

Autorzy: dr hab. inż. Kazimierz Gierlotka, prof. Pol. Śl., Politechnika Śląska, Wydział Elektryczny, Katedra Energoelektroniki, Napędu Elektrycznego i Robotyki, ul. B. Krzywoustego 2, 44-100 Gliwice, E-mail: Kazimierz.Gierlotka@polsl.pl, dr inż. Grzegorz Jarek, Politechnika Śląska, Wydział Elektryczny, Katedra Energoelektroniki, Napędu Elektrycznego i Robotyki, ul. B. Krzywoustego 2, 44-100 Gliwice, E-mail: Grzegorz.Jarek@polsl.pl, dr inż. Michał Jeleń, Politechnika Śląska, Wydział Elektryczny, Katedra Energoelektroniki, Napędu Elektrycznego i Robotyki, ul. B. Krzywoustego 2, 44-100 Gliwice, E-mail: Michal.Jelen@polsl.pl, dr inż. Jarosław Michalak, Politechnika Śląska, Wydział Elektryczny, Katedra Energoelektroniki, Napędu Elektrycznego i Robotyki, ul. B. Krzywoustego 2, 44-100 Gliwice, Jaroslaw.Michalak@polsl.pl

LITERATURA

- [1] Gierlotka K., Jarek G., Jeleń M., Michalak J., Wspomaganie nauczania napędu elektrycznego z wykorzystaniem wirtualnej platformy ePEDlab, XII Konferencja SENE, Łódź, 18-20.11.2015
- [2] Jarek G., Jeleń M., Michalak J.: Zastosowanie technik szybkiego prototypowania w dydaktyce napędu elektrycznego, *Logistyka*, 6 (2014), 4846-4853
- [3] Witryna internetowa <http://kener.elekt.polsl.pl/epedlab/>
- [4] Witryna internetowa <http://www.gecko-simulations.com/>
- [5] Drogenik U., Miising A., Kolar J. W.: Novel online simulator for education of power electronics and electrical engineering, Proc. of the Int. Power Electronics Conf. (IPEC-ECCE Asia), Sapporo, Japan, June 2010
- [6] Miising A., Kolar J. W.: Successful Online Education — GeckoCIRCUITS as Open-Source Simulation Platform, Proceedings of the International Power Electronics Conference - ECCE Asia (IPEC 2014), Hiroshima, Japan, May 18-21, 2014
- [7] Michalak J., Zygmanski M., Biskup T., Kołodziej H.: Model przekształtnika trójfazowego NPC umożliwiającego zwrot energii do sieci zasilającej, *Logistyka*, 6 (2014), 7390-7399