

Wiktor Hudy, Kazimierz Jaracz

Metodyka badań symulacyjnych obiektów opisanych nieliniowym układem równań różniczkowych przy zastosowaniu pakietu MATLAB/Simulink

Wstęp

Symulacje komputerowe zjawisk, układów czy obiektów sterowania stosowane są w celu poznania zachowania się badanego obiektu na różne wymuszenia. Jednym z narzędzi służących do tego celu jest pakiet MATLAB/Simulink [1, 2, 3, 4, 8]. Program nadrzędny, MATLAB, służy do obliczeń inżynierskich, symulacji, interpretacji i wizualizacji wyników w trybie tekstowym i graficznym. Jest językiem programowania wysokiego poziomu. Simulink, toolbox MATLABa, służy wyłącznie do symulacji, przy czym tworzenie symulacji odbywa się wyłącznie w trybie graficznym. W niniejszym artykule przedstawiono proces tworzenia symulacji w Simulinku przy wykorzystaniu bibliotek Simulinka. Można natomiast modelować w trybie mieszanym, zapisując modele obiektów za pomocą s-funkcji. Nie jest to jednak przedmiotem tej pracy. Badanym obiektem jest model matematyczny silnika indukcyjnego, którego układ równań różniczkowych zapisany w układzie współrzędnych Odq sformułowany, przy następujących uproszczeniach [5, 6, 7]:

- nie uwzględnia się nasycenia obwodu magnetycznego silnika i przyjmuje się monoharmoniczne pole magnetyczne w szczelinie,
- nie uwzględnia się strat mocy w rdzeniu obwodu magnetycznego,
- przyjmuje się symetryczne uzwojenie trójfazowe dla stojana,
- uzwojenia wirnika klatkowego zastępuje się równoważnym uzwojeniem 3-fazowym,
- sprzężenie magnetyczne pomiędzy uzwojeniami fazowymi stojana oraz zastąpionymi uzwojeniami wirnika zachodzą tylko dzięki monoharmonicznemu polu głównemu,

ma postać:

$$\begin{bmatrix} \frac{d}{dt} i_1^d \\ \frac{d}{dt} i_1^q \\ \frac{d}{dt} i_2^d \\ \frac{d}{dt} i_2^q \end{bmatrix} = \begin{bmatrix} -L_2 b u_1^d \\ -L_2 b u_1^q \\ \frac{3}{2} L_{12} b u_1^d \\ \frac{3}{2} L_{12} b u_1^q \end{bmatrix} +$$

$$+ \begin{bmatrix} R_1 L_2 b & \left(\frac{3}{2} L_{12}\right)^2 b p_b \Omega - \omega & -\frac{3}{2} L_{12} R_2 b & \frac{3}{2} L_{12} L_2 b p_b \Omega \\ -\left(\frac{3}{2} L_{12}\right)^2 b p_b \Omega + \omega & R_1 L_2 b & -\frac{3}{2} L_{12} L_2 b p_b \Omega & -\frac{3}{2} L_{12} R_2 b \\ -\frac{3}{2} L_{12} R_1 b & -\frac{3}{2} L_{12} L_1 b p_b \Omega & R_2 L_1 b & -\omega - L_2 b p_b \Omega \\ \frac{3}{2} L_{12} L_1 b p_b \Omega & -\frac{3}{2} L_{12} R_1 b & \omega + L_1 L_2 b p_b \Omega & R_2 L_1 b \end{bmatrix} \begin{bmatrix} i_1^d \\ i_1^q \\ i_2^d \\ i_2^q \end{bmatrix}$$

(1)

$$\frac{d}{dt} \Omega = \frac{1}{J} \left(-D\Omega - M_{obc} + \frac{3}{2} L_{12} p_b (i_1^q i_2^d - i_1^d i_2^q) \right)$$

gdzie:

$$b = \frac{1}{\left(\frac{3}{2} L_{12}\right)^2 - L_1 L_2},$$

$$\Omega = \frac{d\theta}{dt},$$

ω – prędkość wirowania układu współrzędnych względem stojana,

L_1 – suma indukcyjności głównej fazy stojana i indukcyjności rozproszenia czołowego i żłobkowego fazy stojana,

L_{12} – indukcyjność między fazami stojana i wirnika sprowadzona na stronę stojana,

L_2 – suma indukcyjności głównej fazy wirnika sprowadzona na stronę stojana i indukcyjności rozproszenia czołowego i żłobkowego fazy wirnika sprowadzona na stronę stojana,

Θ – kąt położenia wirnika względem stojana,

p_b – liczba par biegunów maszyny,

i_1^d, i_1^q – prądy stojana w układzie współrzędnych ortogonalnych Odq ,

i_2^d, i_2^q – prądy wirnika sprowadzone na stronę stojana w układzie współrzędnych ortogonalnych Odq ,

u_1^d, u_1^q – napięcia stojana w układzie współrzędnych ortogonalnych Odq ,

J – moment bezwładności maszyny,

D – współczynnik tarcia lepkiego,

M_{obc} – moment obciążenia maszyny.

Do dalszej analizy przyjęto parametry modelu matematycznego silnika Sg90L-6 takie jak w tabeli 1.

Tab. 1. Parametry modelu matematycznego silnika klatkowego Sg90L-6

Wielkość	Wartość
R_1	0,3747 Ω
R_2	0,112 Ω
L_1	0,07355 H
L_2	0,02836 H
L_{12}	0,0295 H
D	0,8 N · m · s/rad
J	1,2 kg · m ²
p_b	2
M_{obc}	0 N · m
f	50 Hz
U_{zsk}	380 V

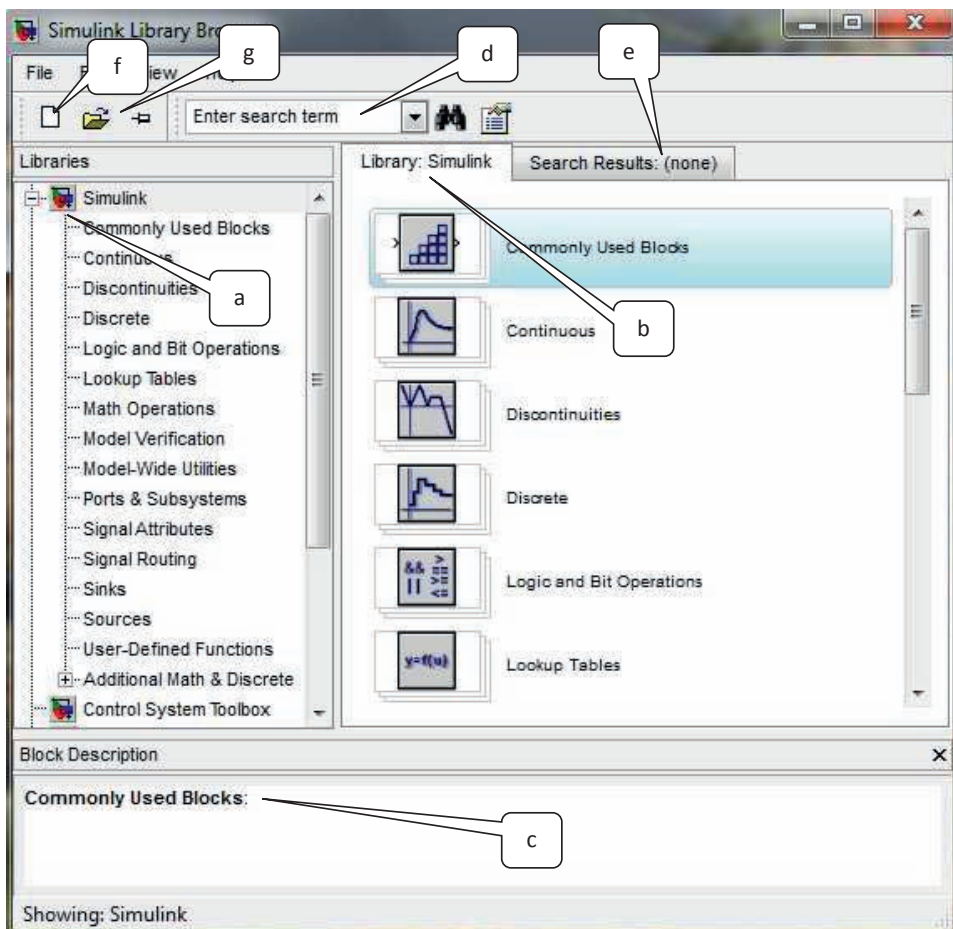
W badanym przypadku wymuszeniem jest wektor o współrzędnych $u_1^d, u_1^q M_{obc}$. Sygnałami wyjściowymi są: $i_1^d, i_1^q, u_2^d, u_2^q, \Omega$. Jak pokazano w wyrażeniu (1) układ równań różniczkowych jest układem nieliniowym ze względu na występowanie iloczynów zmiennej Ω ze zmiennymi prądów $i_1^d, i_1^q, u_2^d, u_2^q$. Analityczne rozwiązanie tego układu równań nie jest możliwe.

Simulink – tworzenie symulacji

Jedynym sposobem analizy tego układu równań jest analiza numeryczna. W celu zasymulowania działania silnika klatkowego w Simulinku niezbędna jest znajomość jego bloków funkcyjnych [8]. Simulinka uruchamia się, wpisując w główne tekstowe okno MATLABa komendę *simulink* i zatwierdzając ją *Enterem*. Otworzy się nowe okno jak na rys. 1.

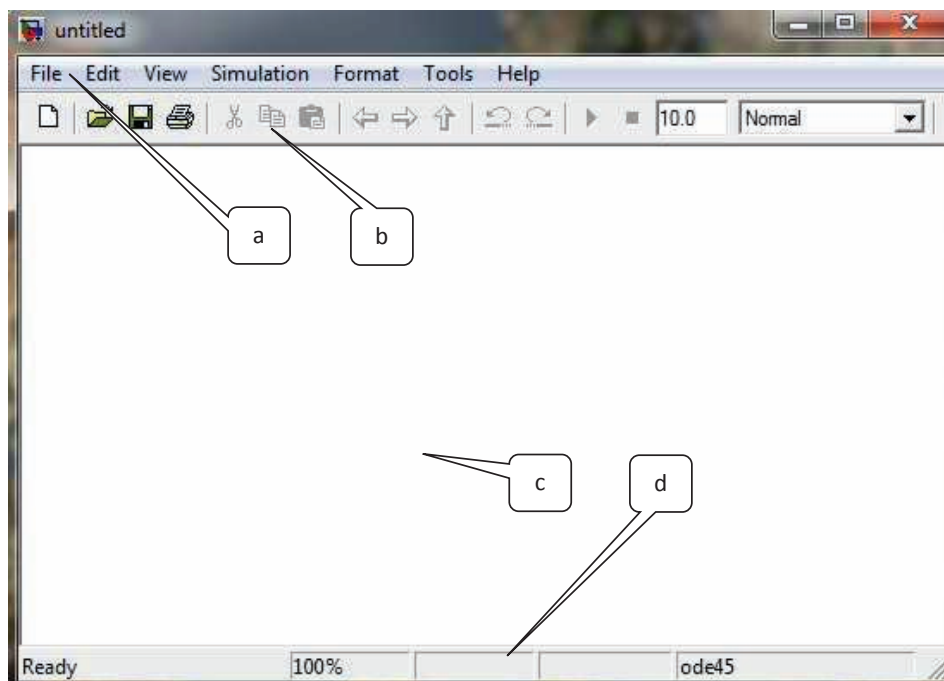
By utworzyć nową symulację, należy nacisnąć przycisk *New model*, oznaczony literą „f”. Pojawi się okno z rys. 2.

Zainicjowana została nowa symulacja. Do tego okna przeciąga się elementy bibliotek z głównego okna Simulinka. Elementy te łączy się za pomocą linii. Każdy element ma wejście (wejścia) lub wyjście (wyjścia) lub oba typy (wejścia i wyjścia).



- a – biblioteki komponentów
- b – zawartość zaznaczonej biblioteki z pkt. a
- c – opis zaznaczonej biblioteki lub komponentu
- d – okienko Szukaj
- e – wyniki przeszukiwania
- f – nowe okno/projekt
- g – otwórz okno/projekt

Rys. 1. Główne okno Simulinka

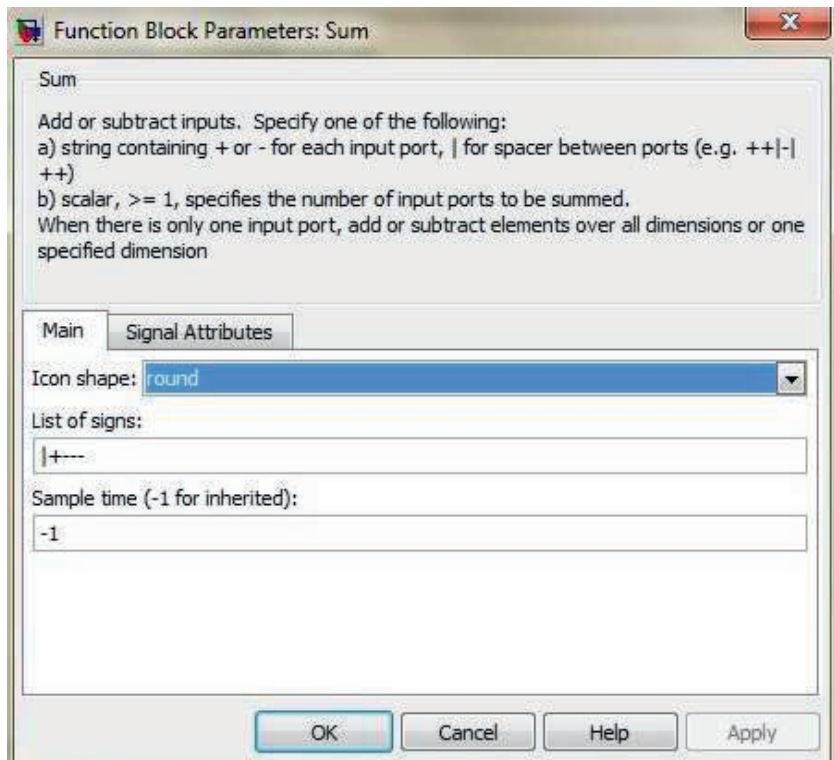


- a – menu górne
- b – pasek ikon
- c – okno z projektem
- d – pasek statusu

Rys. 2. Nowe okno symulacji

Podstawowe bloki funkcyjne Simulinka

Węzeł sumacyjny – oznaczony w bibliotece Simulinka jako *Sum* – służy do sumowania wartości sygnałów wejściowych. Domyślnie węzeł sumacyjny ma dwa wejścia dodatnie (+). Na rys. 3 przedstawiono symbol bloku funkcyjnego węzła sumacyjnego z opisem opcji.



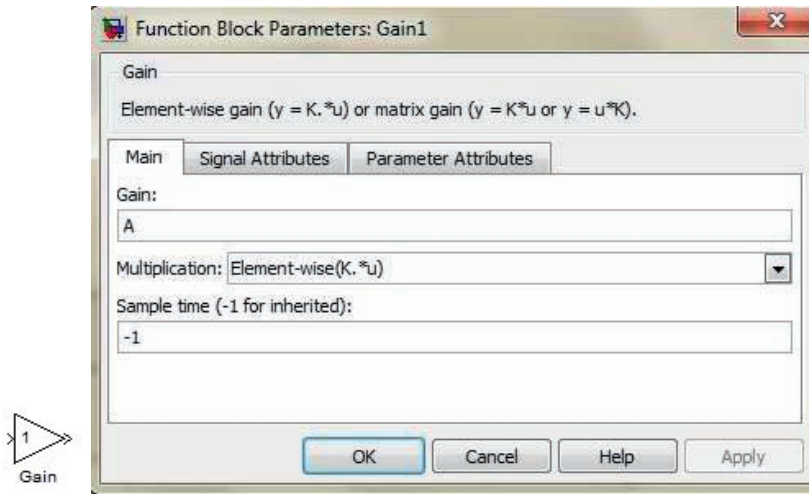
Rys. 3. Symbol bloku funkcyjnego węzła sumacyjnego z opisem jego opcji

Opis najczęściej wykorzystywanych opcji:

List of signs – liczba i rodzaj wejść węzła sumacyjnego; w rozpatrywanym przykładzie jest jedno wejście ze znakiem plus (+) oraz trzy wejścia ze znakiem minus (-),

Icon shape – rodzaj wykorzystywanej ikony; w rozpatrywanym przykładzie jest ikona okrągła; do wyboru ma użytkownik: *round* – ikona okrągła lub *rectangular* – ikona prostokątna.

Mnożenie przez wartość – oznaczony w bibliotece Simulinka jako *Gain* – służy do przemnożenia sygnału przez stałą wartość. Na rys. 4 przedstawiono symbol bloku funkcyjnego *Gain* z opisem opcji.

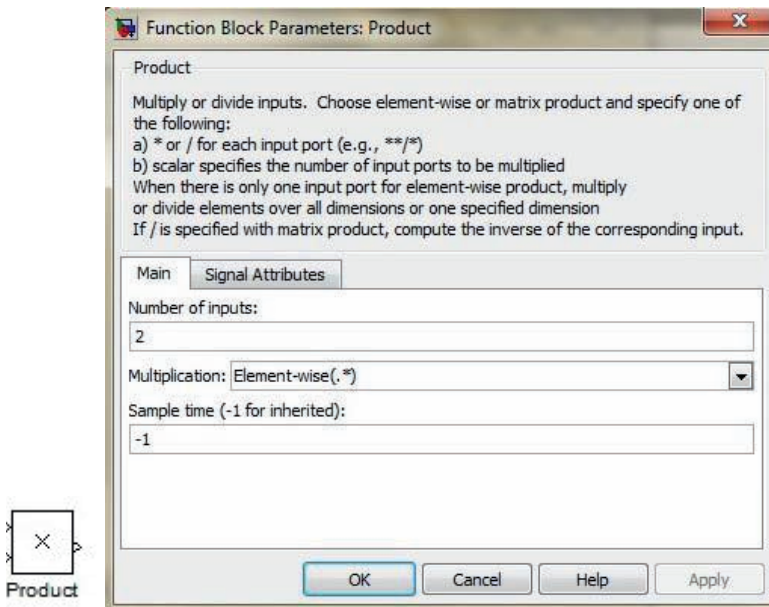


Rys. 4. Symbol bloku funkcyjnego *Gain* z opisem jego opcji

Opis najczęściej wykorzystywanych opcji:

Gain – współczynnik, przez który przemnażany jest sygnał wejściowy. Domyślną wartością jest wartość 1. W polu tym można stosować również zmienne będące skalarami, pod warunkiem wcześniejszego ich zdefiniowania w oknie głównym MATLABa.

Mnożenie sygnałów – oznaczono w bibliotece Simulinka jako *Product* – służy do mnożenia dwóch lub więcej sygnałów przez siebie. Na rys. 5 przedstawiono symbol bloku funkcyjnego *Product* z opisem opcji.

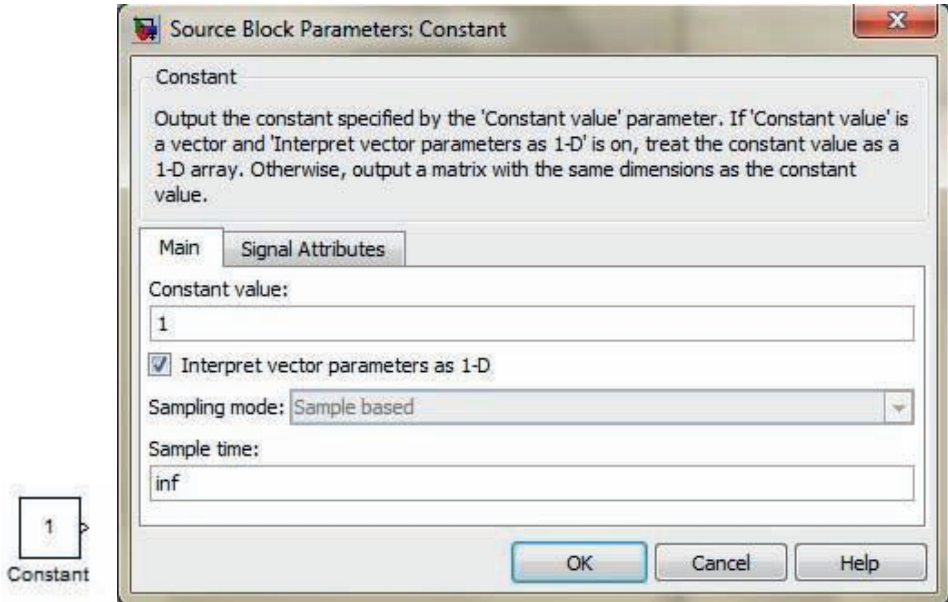


Rys. 5. Symbol bloku funkcyjnego *Product* z opisem jego opcji

Opis najczęściej wykorzystywanych opcji:

Number of inputs – liczba wejść. Domyślną wartością jest wartość 2.

Wartość stała – oznaczono w bibliotece Simulinka jako *Constant* – służy do wprowadzenia wartości stałej. Na rys. 6 przedstawiono symbol bloku funkcyjnego *Constant* z opisem opcji.

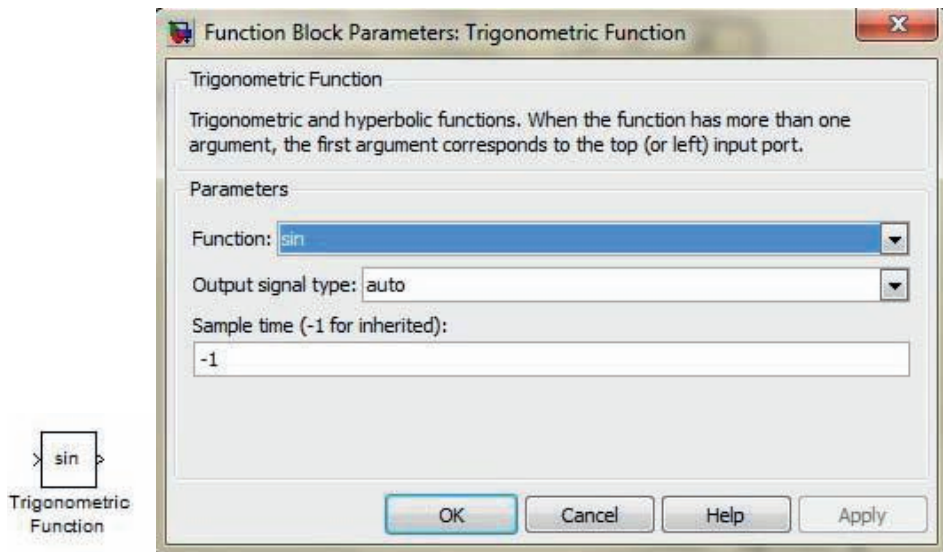


Rys. 6. Symbol bloku funkcyjnego *Constant* z opisem jego opcji

Opis najczęściej wykorzystywanych opcji:

Constant value – wprowadzana wartość. Domyślną wartością jest wartość 1.

Stosowanie funkcji trygonometrycznych – oznaczono w bibliotece Simulinka jako *Trigonometric Function* – służy do obliczania w czasie symulacji wartości zdefiniowanej funkcji trygonometrycznej. Na rys. 7 przedstawiono symbol bloku funkcyjnego *Trigonometric Function* z opisem opcji.



Rys. 7. Symbol bloku funkcyjnego *Trigonometric Function* z opisem jego opcji

Opis najczęściej wykorzystywanych opcji:

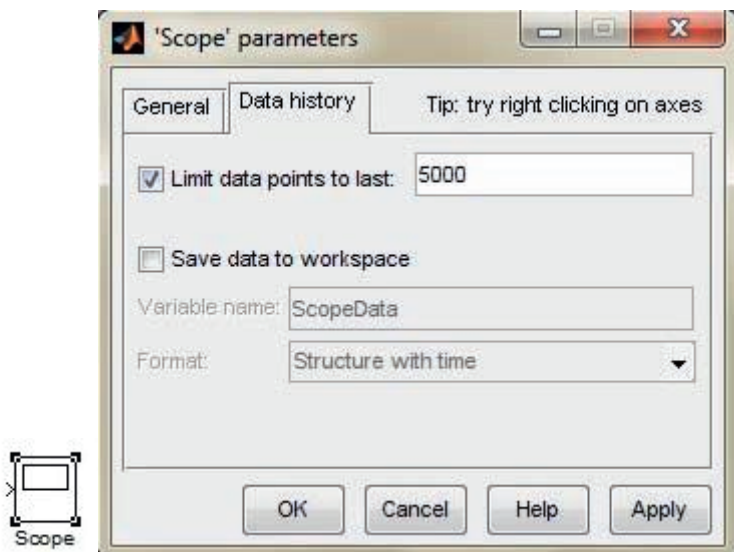
Function – definicja funkcji trygonometrycznej. Domyślną funkcją jest funkcja sinus. Możliwe jest zdefiniowanie: *sin* – funkcja sinus; *cos* – cosinus; *tan* – tangens; *asin* – arcus sinus; *sinh* – sinus hiperboliczny; *asinh* – arcus sinus hiperboliczny itp.

Zegar – oznaczono w bibliotece Simulinka jako *Clock* – służy do wprowadzenia wartości zegara symulacji. Symbol zegara przedstawiono na rys. 8.



Rys. 8. Symbol bloku funkcyjnego zegara

Wizualizacja wyników (oscyloskop) – oznaczono w bibliotece Simulinka jako *Scope* – służy do wykreślania przebiegów. Domyślnie ma jedno wejście. Symbol bloku funkcyjnego *Scope* przedstawiono na rys. 9.

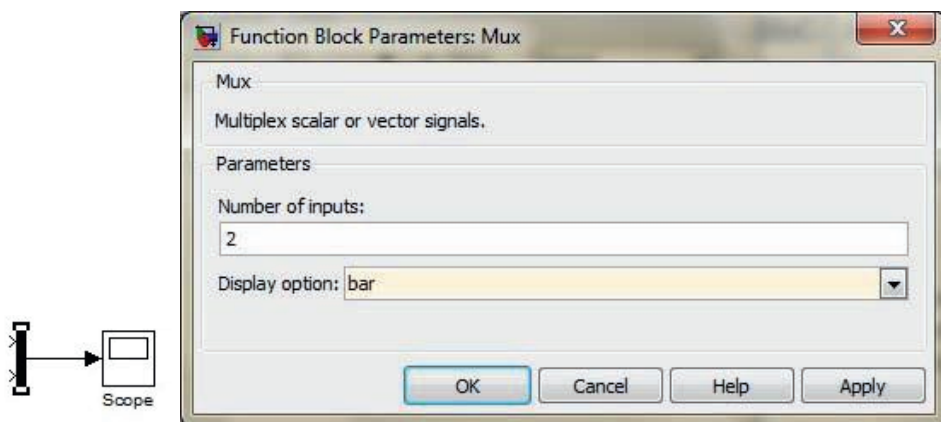


Rys. 9. Symbol bloku funkcyjnego *Scope* oraz jego opcje

Opis najczęściej wykorzystywanych opcji:

Limit data points to last – określenie liczby ostatnich zarejestrowanych punktów pokazywanych w aktywnym oknie oscyloskopu. Podczas dłuższych symulacji należy upewnić się, że okienko jest niezaznaczone, spowoduje to rejestrację wszystkich punktów symulacji.

Multiplekser – oznaczono w bibliotece Simulinka jako *Mux* – służy do zastąpienia dwóch lub więcej linii sygnału jedną. Domyślnie ma dwa wejścia sygnałowe. W połączeniu z oscyloskopem powoduje pokazanie więcej niż jednego przebiegu. Symbol bloku funkcyjnego *Mux* wraz z oscyloskopem przedstawiono na rys. 10.



Rys. 10. Symbol bloku funkcyjnego *Mux* oraz jego opcje

Opis najczęściej wykorzystywanych opcji:

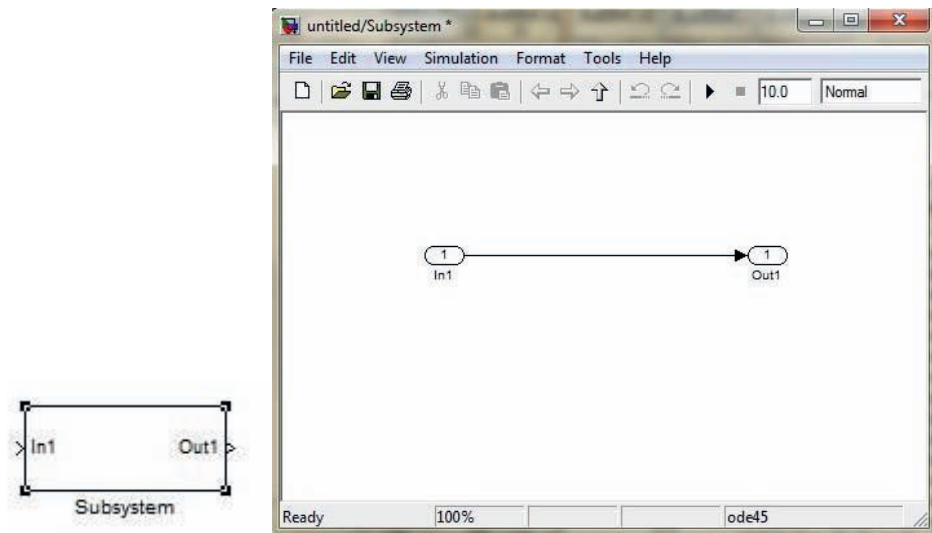
Number of inputs – liczba wejść multipleksera.

Zakończenie linii sygnałowej – oznaczono w bibliotece Simulinka jako *Terminator* – służy do zakończenia dowolnej linii sygnałowej. Symbol bloku funkcyjnego *Mux* przedstawiono na rys. 11.



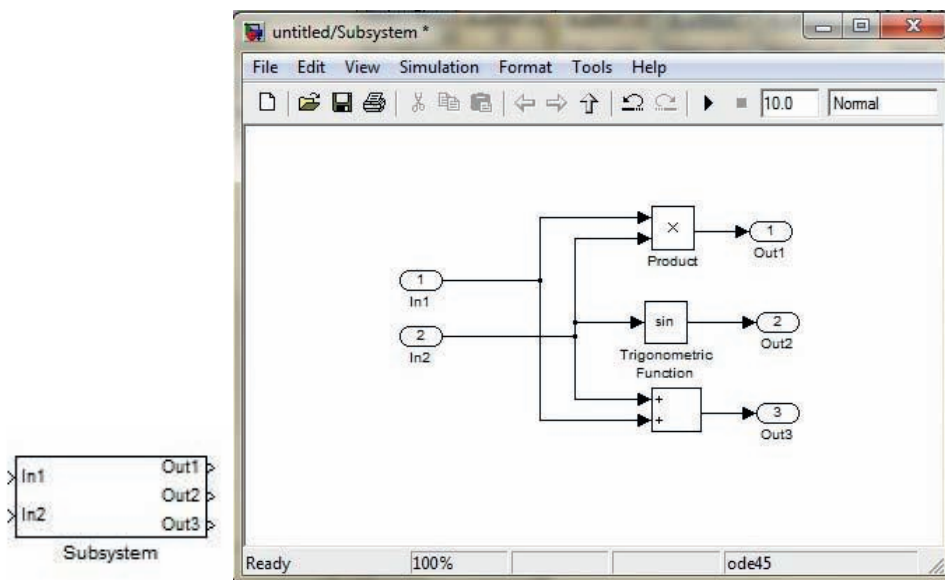
Rys. 11. Symbol bloku funkcyjnego *Terminator*

Podsystem – oznaczono w bibliotece Simulinka jako *Subsystem* – służy do grupowania bloków funkcyjnych stanowiących fragment złożonego układu. Symbol bloku funkcyjnego *Subsystem* przedstawiono na rys. 12.



Rys. 12. Symbol bloku funkcyjnego *Subsystem*

Na rys. 12 przedstawiono symbol bloku funkcyjnego oraz domyślny schemat blokowy nowego tworzonego podsystemu. Domyślnie sygnał z wejścia pierwszego *In1* jest przekazywany na wyjście pierwsze *Out1*, co jest symbolizowane przez połączenie linią wejścia *In1* z wyjściem *Out1*. Grot strzałki symbolizuje kierunek przepływu sygnału. Pomiedzy wejścia i wyjścia użytkownik ma możliwość wstawiania dowolnych bloków funkcyjnych, w tym dalszych podsystemów, zachowując kierunek przepływu sygnału. W przypadku potrzeby wykorzystania dwóch wejść, należy wejście *In1* skopiować, używając np. klawiszy funkcyjnych *Ctrl+c* oraz *Ctrl+v* lub skorzystać z biblioteki Simulinka. Cyfry zawarte wewnątrz elips oznaczających wejścia lub wyjścia są ich numerami. Przykład podsystemu z dwoma wejściami i trzema wyjściami przedstawiono na rys. 13.

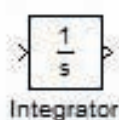


Rys. 13. Podsystem z dwoma wejściami i trzema wyjściami

Analiza podsystemu z rys. 13:

- ma dwa wejścia oznaczone $In1$, $In2$,
- ma trzy wyjścia oznaczone $Out1$, $Out2$, $Out3$,
- ciąg znaków $In1$, $In2$, $Out1$, $Out2$, $Out3$ jest komentarzem do wejścia/wyjścia, który można zmienić,
- sygnał wyjścia pierwszego jest równy iloczynowi sygnałów wejść pierwszego i drugiego,
- sygnał wyjścia drugiego jest równy sinusowi wejścia drugiego,
- sygnał wyjścia trzeciego jest równy sumie sygnałów wejść pierwszego i drugiego.

Całkowanie – oznaczono w bibliotece Simulinka jako *Integrator* – służy do obliczania w czasie rzeczywistym całki sygnału wejściowego. Symbol bloku funkcyjnego *Integrator* przedstawiono na rys. 14.



Rys. 14. Symbol bloku funkcyjnego *Integrator*

Metodyka konstrukcji schematu blokowego opisanego równaniami różniczkowymi

Jako przykład posłużą równanie ruchu układu równań (1):

$$\frac{d}{dt}\Omega = \frac{1}{J} \left(-D\Omega - M_{obc} + \frac{3}{2}L_{12}p_b(i_2^d i_1^q - i_1^d i_2^q) \right) \quad (2)$$

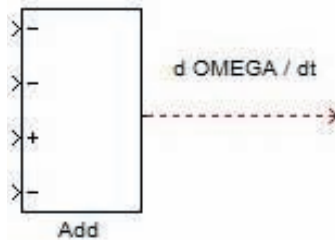
Rozpatrywane równanie należy przedstawić w postaci (3), tj. po jednej stronie równania ma być najwyższy stopień pochodnej badanego sygnału, po drugiej stronie pozostałe elementy równania przedstawione jako suma.

$$\frac{d}{dt}\Omega = -\frac{D}{J}\Omega - \frac{M_{obc}}{J} - a_{1_1^d 1_2^q} + a_{2_2^d 1_1^q} \quad (3)$$

gdzie

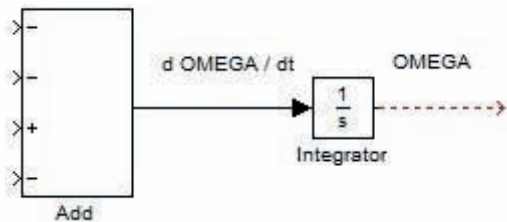
$$a = \frac{3}{2} \frac{L_{12}p_b}{J}$$

Równanie (3) ma po prawej stronie znaku równości cztery elementy sumy. Budowę symulacji zaczyna się od bloku sumatora (*Sum*) o czterech wejściach, co pokazano na rys. 15. Należy zwrócić uwagę na fakt, że pierwszy, drugi i czwarty składnik sumy mają znaki ujemne.



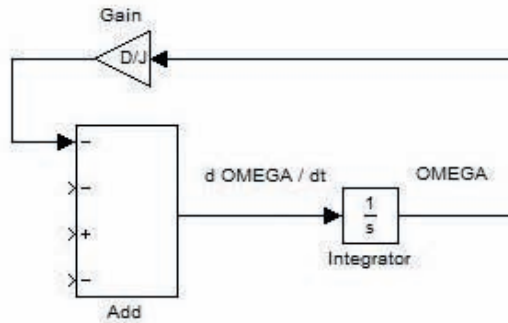
Rys. 15. Początek budowy symulacji

Sygnał wyjściowy sumatora jest równy wg wzoru (3) pochodnej prędkości obrotowej. We wzorze (3) znajduje się pierwszy składnik sumy zależny od prędkości obrotowej, a nie od jej pochodnej. Sygnał wyjściowy sumatora należy scałkować (rys. 16).



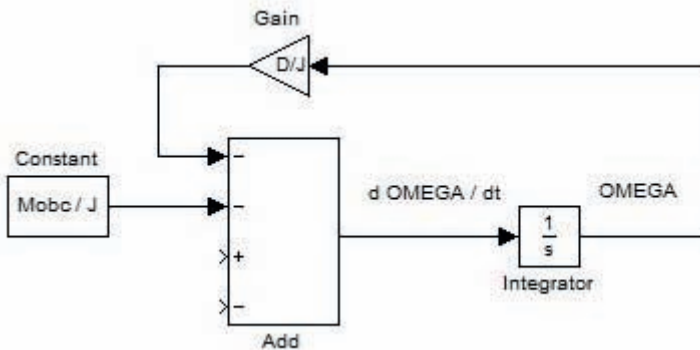
Rys. 16. Scałkowany sygnał prędkości obrotowej

Pierwszy składnik sumy równania (3) zależy od prędkości obrotowej, współczynnika tarcia lepkiego oraz momentu bezwładności. Sygnał prędkości obrotowej należy pomnożyć przez składnik D/J i wprowadzić do bloku funkcyjnego sumatora ze znakiem ujemnym (rys. 17).



Rys. 17. Wprowadzony sygnał prędkości obrotowej przemnożony przez współczynnik D/J

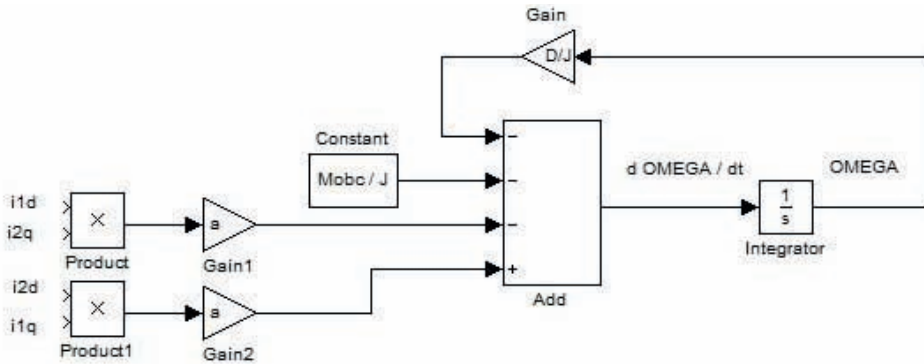
Drugi składnik sumy równania (3) jest stałym współczynnikiem. Modyfikację schematu przedstawiono na rys. 18.



Rys. 18. Tworzony schemat uzupełniony o drugi współczynnik sumy równania (3)

Trzeci i czwarty składnik sumy zależy od tego samego współczynnika. Sygnały prądów są sygnałami zmiennymi reprezentowanymi przez pozostałe równania układu równań (1). Modyfikację schematu przedstawiono na rys. 19.

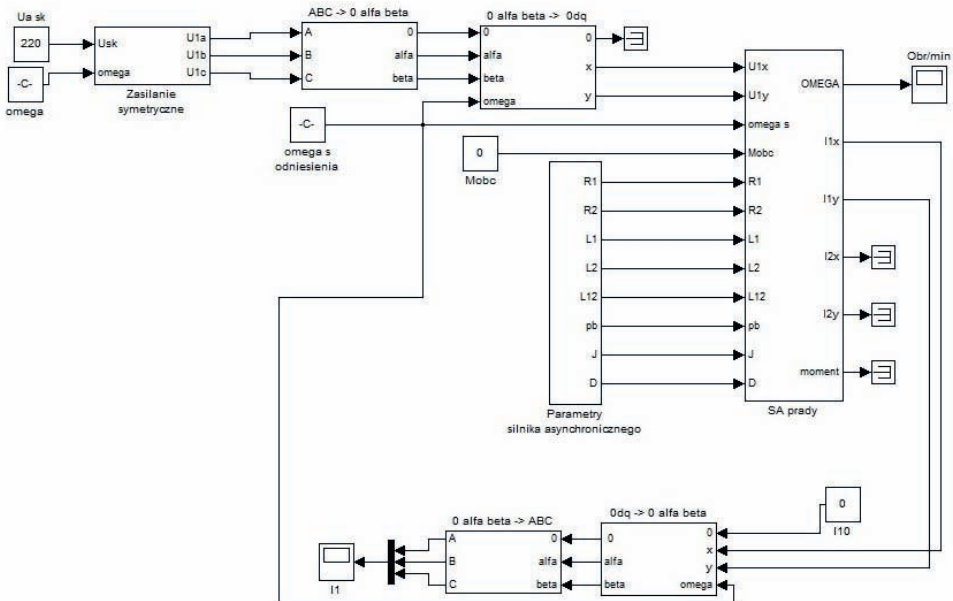
Analogicznie tworzy się strukturę bloków funkcyjnych pozostałych czterech równań różniczkowych, sygnałów prądów $i_1^d, i_1^q, i_2^d, i_2^q$. W dalszej kolejności sygnały prądów wprowadza się do fragmentu układu z rys. 19. Sygnał prędkości obrotowej Ω wprowadza się do fragmentów układu odpowiadających równaniom od pierwszego do czwartego układu równań (1).



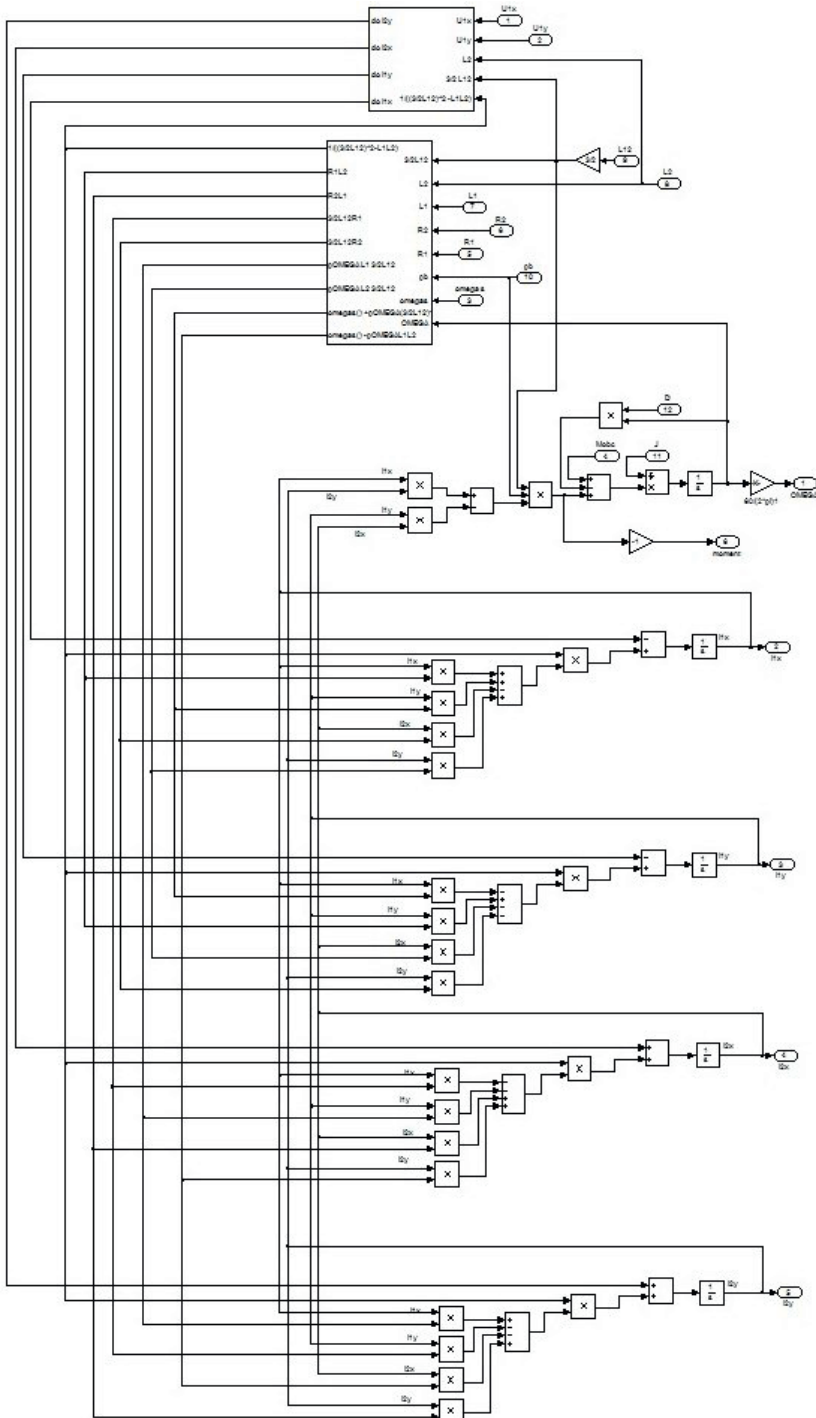
Rys. 19. Tworzony schemat uzupełniony o trzeci i czwarty współczynnik sumy równania (3)

Ukończony schemat blokowy

Ukończony układ przedstawiono na rys. 21. Uzupełniając model matematyczny silnika klatkowego o bloki funkcyjne reprezentujące zasilanie oraz wizualizację wyników otrzymuje się schemat blokowy gotowy do przeprowadzania symulacji rozruchu (rys. 20).



Rys. 20. Schematy blokowe reprezentujące układ zasilania silnika klatkowego



Rys. 21. Schemat blokowy modelu matematycznego silnika klatkowego przygotowany do badań symulacyjnych z zastosowaniem języka MATLAB/Simulink

Literatura

- [1] Brzózka J., Dorobczyński L., *Programowanie w Matlab*, Zakład Nauczania Informatyki „MIKOM”, Warszawa 1998
- [2] Cegiela R., Zalewski A., *Matlab – obliczenia numeryczne i ich zastosowania*, Wydawnictwo NAKOM, Poznań 2000
- [3] Jaracz K., *Symulacyjne metody badania elementów i układów regulacji*, Annales Academiae Paedagogicae Cracoviensis 74. Studia Technica III, Wydawnictwo Naukowe AP, Kraków 2010
- [4] Mrozek B., Mrozek Z., *MATLAB i Simulink. Poradnik użytkownika*, wyd. III, Helion, Warszawa 2010
- [5] Rusek J., *Elektrotechnika z elementami napędów*, wyd. 3, Wydawnictwo AGH, Kraków 1996
- [6] Skwarczyński J., Tertel Z., *Maszyny elektryczne*, Uczelniane Wydawnictwa Naukowo-Dydaktyczne AGH, Kraków 1999
- [7] Sobczyk T.J., *Metodyczne aspekty modelowania matematycznego maszyn indukcyjnych*, WNT, Warszawa 2004
- [8] System pomocy programu MATLAB/Simulink (ang.)

Method for simulation objects described by set of non-linear differential equations with using MATLAB/Simulink

Abstract

In this paper the way of simulation implementation of complex objects is presented. These object are described by differential equations, which are non-linear in many cases. Project with using MATLAB/Simulink was realized.

Key words: Matlab, Simulink, induction motor

Wiktor Hudy, Kazimierz Jaracz
UP – Kraków
Instytut Techniki