

Szczegółowy opis przedmiotu zamówienia- zadanie 2

Zadanie 2: Dostawa oprogramowania do projektowania sterowania oraz oprogramowania do obliczeń inżynierskich do Pracowni Programowania Układów Mikroprocesorowych w Centrum Innowacji i Transferu Wiedzy Techniczno-Przyrodniczej

Czas dostawy – 2 miesiące od podpisania umowy.

Gwarancja – opisana dla każdej pozycji

I. Elementy składowe stanowiska:

L.p.	Opis	Ilość
1	Oprogramowanie do projektowania sterowania	1
2	Oprogramowanie do obliczeń inżynierskich	1

II. Szczegółowy opis przedmiotu zamówienia:

Ad. 1 Oprogramowanie do projektowania sterowania (szt. 1)

Minimalne wymagania
<p>Elementy składowe nie gorsze niż:</p> <ol style="list-style-type: none"> 1. Pakiet Matlab, Simulink i Toolboxy (1x licencja badawcza, wersja min. 2012a) lub równoważne: <ol style="list-style-type: none"> a) Matlab b) Simulink c) Control System Toolbox d) DSP System Toolbox e) Signal Processing Toolbox f) Simulink Control Design g) System Identification Toolbox h) Symbolic Math Toolbox <p>Oprogramowanie dostarczone na oryginalnym nośniku producenta oprogramowania. Gwarancja na nośnik min. 3 mc.</p> <p>Ad. a)</p> <p>Program musi posiadać następujące rozwiązania programistyczne:</p> <ol style="list-style-type: none"> 1. Język programowania wysokiego poziomu - indeksacja niejawną pozwalająca na efektywną wektoryzację obliczeń i zwarty zapis (konstrukcje z użyciem dwukropka, nawiasów kwadratowych, połączenie kropki i operatorów arytmetycznych, indeksacja za pomocą warunków logicznych, wektoryzacja funkcji wbudowanych), domyślne określanie typów i wymiarowanie tablic umożliwiające szybkie prototypowanie aplikacji w środowisku interpretera programu, automatyczna generacja kodu C wykorzystywana do budowania niezależnych aplikacji na bazie kodu programu, różnorodne typy zmiennych: numeryczne - stałoprzecinkowe i zmiennoprzecinkowe, tablice wielowymiarowe, zmienne tekstowe, zmienne strukturalne - dostęp do pola przez nazwę (notacja kropkowa), tablice wskaźnikowe - cellarrays (notacja klamrowa) 2. Otwartą architekturę - rozszerzalność (możliwość budowania własnych funkcji użytkownika: pliki, skrypty i funkcje, dołączanie skompilowanego kodu w C lub w Fortranie: biblioteki DLL - MEX-pliki), przenośność (jednokodowy kod programu - kod dla różnych platform PC: Windows 95/98/NT, Linux, Mac i UNIX: Sun Solaris, HP UX, IBM AIX, SGI Irix i innych), wymiana danych z innymi aplikacjami poprzez pliki i różne protokoły komunikacyjne: DDE, TCP/IP (internet)- Web Server, ODBC/JDBC (bazy danych), współpraca z innymi aplikacjami w środowisku wielozadaniowym (możliwość uruchamiania innych aplikacji z poziomu programu - polecenia poprzedzane znakiem!, udostępnianie usług Engine Services) 3. Orientację obiektową - grafika uchwytów (hierarchiczna struktura Handle Graphics - pobieranie (get) i nadawanie (set) wartości atrybutów określanych zmiennymi tekstowymi), obiekty standardowe (np. LTI - liniowy, stacjonarny system dynamiczny, obiekt sym - wyrażenie symboliczne), narzędzia programowania obiektowego (hermetyzacja, dziedziczenie, przeciążanie). <ol style="list-style-type: none"> 1. Narzędzie do projektowania systemów opartych na logice rozmytej. Wyspecjalizowane graficzne interfejsy użytkownika (GUI) do budowy rozmytych systemów wnioskujących oraz podglądu i analizy wyników; funkcje

- przynależności do tworzenia systemów rozmytego wnioskowania. Obsługa logiki AND, OR i NOT w zdefiniowany przez użytkownika sposób. Standardowy system wnioskowania rozmytego typu Mamdani i Sugeno. Zautomatyzowane funkcje przynależności kształtujące poprzez techniki uczenia neuroadapttywnej rozmytej analizy skupień. Zdolność umieszczania systemu rozmytego wnioskowania w modelu. Zdolność do generowania kodu C lub samodzielnie wykonywalnego enginu rozmytego wnioskowania.
2. Narzędzie do optymalizacji algorytmów genetycznych i algorytmów przeszukiwania bezpośredniego. Graficzny interfejs użytkownika i dostęp do funkcji z linii poleceń do szybkiego ustawiania parametrów problemu i opcji algorytmu oraz monitorowania postępu działania. Narzędzia do stosowania algorytmów genetycznych z możliwością dopasowania wielu opcji, w tym krzyżowanie, mutacja, selekcja. Algorytmy przeszukiwania bezpośredniego z możliwością zdefiniowania rozmiaru siatki. Funkcje integrujące rozszerzenie i wbudowane procedury programu z algorytmami genetycznymi i przeszukiwania bezpośredniego. Obsługa automatycznej generacji kodu programu.
 3. Narzędzie do przetwarzania obrazów, analizy, wizualizacji i rozwijania własnych algorytmów. Poprawianie obrazu, w tym filtrowanie liniowe i nieliniowe, projektowanie filtrów, wyostanie i automatyczna regulacja kontrastu. Analiza obrazu, w tym analiza teksturowa, detekcja linii, morfologia, detekcja krawędzi, segmentacja, przetwarzanie ROI (Rejon Zainteresowania), pomiar parametrów. Przetwarzanie obrazów barwnych, konwersja palety kolorów, import i eksport profili ICC. Transformacje przestrzenne i rejestracja obrazów, w tym narzędzie graficzne do wyboru punktów kontrolnych. Transformacje obrazów, takie jak FFT, DCT, Radon i rzutowanie wiązki wachlarzowej. Import i eksport DICOM. Interaktywny podgląd obrazów i modułarne narzędzia do tworzenia własnego Graficznego Interfejsu Użytkownika (GUI) dla wybranych obrazów. Obsługa wielowymiarowego przetwarzania obrazu.
 4. Narzędzie do projektowania, implementacji, wizualizacji i symulacji sieci neuronowych. Graficzny interfejs użytkownika do tworzenia, uczenia i symulowania sieci neuronowych. Wsparcie dla najpowszechniejszych architektur (nadzorowanych i bez nauczyciela). Kompletny zestaw funkcji uczących sieć. Zbiór bloków programu wraz z dokumentacją i demonstracjami aplikacji systemów sterowania. Automatyczne tworzenie modeli programu z obiektów sieci neuronowych. Modułarna reprezentacja sieci, pozwalająca na zadawanie dowolnej liczby warstw wejściowych i dowolnej liczby połączeń między warstwami. Zwiększanie skuteczności uczenia się dzięki funkcjom przetwarzającym dane przed i po uczeniu sieci. Procedury poprawiające generalizację. Funkcje wizualizacyjne pokazujące skuteczność sieci.
 5. Narzędzie do wielowymiarowej optymalizacji średniej i wielkiej skali. Funkcje dostępne z linii poleceń programu powinny posiadać łatwy do modyfikacji otwartego kodu źródłowego. Rozszerzenie powinno zawierać funkcje do: definiowania, rozwiązywania i oceny problemów optymalizacji. Optymalizacji nieliniowej i wielokryterialnej. Obliczeń regresji średniokwadratowej i nieliniowej. Rozwiązywania zadań programowania kwadratowego i liniowego.
 6. Narzędzie do tworzenia aplikacji programu i uruchomienie ich na klastrach komputerów, bez opuszczania własnego stanowiska pracy. Rozszerzenie powinno prototypować aplikacje w programie i wykorzystać funkcjonalność rozszerzenia aby zdefiniować niezależne i współdziałające zadania. Algorytmy, które wymagają współdziałających zadań, wykorzystują interfejs MPI (Message Passing Interface). Program powinien szeregować i uruchamiać zadania na wielu zdalnych sesjach programu, redukując czas wykonania w porównaniu z pojedynczą sesją.
 7. Narzędzie do rozwiązywania równań różniczkowych cząstkowych (PDE) w dwóch wymiarach i w czasie, metodą elementów skończonych. Kompletny GUI do pre- i post-procesowania dwuwymiarowych PDE. Automatyczny i adaptacyjny wybór siatki. Tworzenie geometrii za pomocą Constructive Solid Geometry. Specyfikacja warunków brzegowych: Dirichleta, Neumanna lub mieszane. Wykorzystanie składni programu do elastycznej specyfikacji współczynników i PDE. W pełni zautomatyzowane generowanie siatki i jej oczyszczanie. Jednoczesna wizualizacja wielu własności rozwiązania, nakładanie siatki FEM i animacje.
 8. Narzędzie do przetwarzania sygnałów analogowych i cyfrowych. Zestaw modeli sygnałowych i systemów liniowych. Narzędzia do projektowania, analizy i implementacji filtrów cyfrowych FIR i IIR. Narzędzia do projektowania filtrów analogowych. Dostęp do najczęściej używanych transformat, jak FFT czy DCT. Narzędzia do analizy spektralnej i statystycznego przetwarzania sygnału. Funkcje do modelowania parametrycznych ciągów czasowych. Generowanie przebiegów falowych, w tym generator impulsów gaussowskich, przebiegów sinusoidalnych i prostokątnych. Interfejsy graficzne do projektowania, analizy i wizualizacji sygnałów, filtrów i okien.
 9. Narzędzie do analizy historii danych, modelowania danych, symulowania systemów, rozwijania algorytmów statystycznych oraz nauczania statystyki. Obliczanie i dopasowywanie rozkładów prawdopodobieństwa. Modelowanie liniowe i nieliniowe. Statystyka wielu zmiennych. Statystyka opisowa. Analiza wariancji (ANOVA). Testowanie hipotez. Statystyka przemysłowa. Wizualizacja danych statystycznych.

Ad. b)

Platformą do wielodomenowej symulacji i projektowania systemów dynamicznych z wykorzystaniem modeli.

Stanowiąca interaktywne środowisko graficzne i zbiór bibliotek konfigurowalnych z blokami pozwalającymi dokładnie projektować, symulować, implementować i kontrolować przetwarzanie sygnałów, telekomunikację i inne systemy

zdefiniowane w dziedzinie czasu.

Dodatkowe moduły rozszerzają środowisko o narzędzia do modelowania i projektowania specyficznych zadań oraz generacji kodu, implementacji algorytmów, testowania i weryfikacji. Integracja z MATLABem zapewnia bezpośredni dostęp do szerokiego wachlarza narzędzi do budowy algorytmów, analizy i wizualizacji danych oraz obliczeń numerycznych. Platforma musi umożliwiać stworzenie, modelowanie i zarządzanie szczegółowym diagramem bloków systemu używając obszernych zestawów zdefiniowanych bloków. Musi zapewnić narzędzia do hierarchizacji modelowania, zarządzania danymi i konfiguracji podsystemów sprawiając, że tworzenie zwiezłych, dokładnych prezentacji jest proste niezależnie od złożoności modelowanego systemu.

Musi posiadać ponad 1000 bloków i zaimplementowanych funkcji używanych potocznie w modelowaniu systemów.

Wbudowane bloki można konfigurować do własnych potrzeb a nawet tworzyć nowe. Dodatkowe zestawy bloków (dostępne oddzielnie) rozszerzają platformę o specyficzne właściwości do symulacji systemów przestrzeni powietrznej, telekomunikacji, przetwarzania sygnałów i innych. SimMechanics i SimPowerSystems (dostępne oddzielnie) pozwalają modelować systemy fizyczne zapewniając rozszerzone możliwości do modelowania systemów odpowiednio mechanicznych i elektrycznych.

Budowanie modelu polega na przesuwaniu i upuszczaniu bloków z przeglądarki bibliotek do edytora graficznego i łączeniu ich liniami w celu ustanowienia powiązań matematycznych między blokami.

Platforma pozwala zorganizować model w przejrzyste, łatwe w zarządzaniu poziomy hierarchii używając podsystemów. Podsystem obejmuje grupę bloków i sygnałów w pojedynczym bloku. To podstawowy sposób na wprowadzenie hierarchii do modelu. Można również podzielić model na segmenty by modelować, symulować i weryfikować każdą część oddzielnie. Komponenty mogą być zapisane oddzielnie jako osobne modele lub podsystemy w bibliotece. Można użyć ponownie zaprojektowane komponenty w wielokrotnych projektach w łatwy sposób zarządzając kontrolą i historią korekt. Organizowanie modelu w ten sposób pozwala wybrać poziom szczegółowości właściwy do zadania projektowego. Na przykład można użyć prostych relacji do modelu specyfikacji wysokiego poziomu i dodawać bardziej szczegółowe relacje przed implementacją.

Konfigurowalne podsystemy pozwalają łączyć różne warianty projektu z podsystemami wewnątrz modelu. Upraszcza to tworzenie i zarządzanie projektami, które współdzielą komponenty, gdyż jeden model może reprezentować całą rodzinę projektów.

Podsystemy sterowane zdarzeniami pozwalają na zmianę dynamiki systemu umożliwiając lub blokując dostęp do niektórych sekcji projektu poprzez sygnały kontrolne. Platforma pozwala tworzyć sygnały kontrolne umożliwiające lub wywołujące uruchomienie podsystemu bazując na określonym czasie lub zdarzeniu. Bloki logiczne pozwalają modelować proste komendy kontrolujące włączanie lub wywołanie podsystemów. Bardziej złożona logika jak również modelowanie systemów sterowanych zdarzeniami jest możliwe z modułem Stateflow (dostępnym oddzielnie).

Ad. c)

Oprogramowanie zapewniające wyspecjalizowane narzędzia modelowania, projektowania i analizy systemów ze sprzężeniami, obejmując zarówno klasyczne jak i nowoczesne metody projektowe :

- LTI Viewer - interaktywny graficzny interfejs użytkownika (GUI) do analizy i porównywania liniowych systemów niezmiennych w czasie (LTI systems).
- Wykresy w dziedzinie czasu odpowiedzi na skok jednostkowy, impulsowej i zero-biegunowej oraz odpowiedzi w dziedzinie częstotliwości (Bode, Nyquist i Nichols) w jednym oknie.
- Narzędzie projektowania systemów z pojedynczym wejściem i pojedynczym wyjściem (SISO Design Tool) - interaktywny GUI do analizy i konfiguracji systemów kontroli SISO używając wykresów linii pierwiastkowych, Bodego i Nicholasa.
- Graficzny kompensator ograniczający w SISO Design Tool.

- Interaktywny wykres odpowiedzi w zamkniętej pętli ciągle aktualizowany wraz z modyfikacją kompensatora w SISO Design Tool.
- Opcje zapisu i odczytu umożliwiające archiwizację projektów z wielokrotnymi kompensatorami.
- Pełna kontrola nad wizualizacją w dziedzinie czasu i częstotliwości.
- Specjalne struktury danych zwane obiektami LTI (LTI Objects) dla związanej reprezentacji formatów danych funkcji przejścia, przestrzeni stanu, zer i biegunów i odpowiedzi częstotliwościowych.
- Obsługa systemów wielowejsściowych i wielowyjściowych (MIMO), systemów czasu ciągłego i próbkowanych danych oraz systemów z opóźnieniami czasowymi.
- Obsługa różnorodnych metod przekształceń dyskretnych na ciągłe.
- Narzędzia nowoczesnych technik kontroli projektowania (umieszczanie biegunów, regulacje LQR-LQG, projektowanie filtrów Kalmana, wyznaczanie obserwowalności i sterowalności i rozwiązania równań Riccatiego i Lapunowa).

Ad. d)

Moduł rozszerzający środowisko MATLAB & Simulink o efektywne oparte na ramkach przetwarzanie oraz algorytmy do projektowania, implementacji i weryfikacji systemów przetwarzania sygnałów.

DSP System Toolbox udostępnia swoją funkcjonalność zarówno w postaci funkcji środowiska MATLAB, jak i tzw. MATLAB System Objects oraz bloków środowiska Simulink. Moduł pozwala na modelowanie strumieni danych i systemów o zmiennej częstotliwości próbkowania telekomunikacyjnych, audio-wideo, kontroli cyfrowej, radarowych i sonarowych, elektronicznych konsumenckich i medycznych oraz innych obszarów zastosowań obliczeń numerycznych:

- Obsługa symulacji opartej na ramkach i próbkach systemów wielokanałowych i o zmiennej częstotliwości próbkowania oraz stałoprzecinkowych i zmiennoprzecinkowych typach danych.
- Algorytmy transformacji (FFT i DCT), bloki estymacji widma i funkcji okien.
- Algorytmy projektowania i realizacji filtrów cyfrowych.
- Filtry Adaptatywne (RLS, LMS i Kalmana) i filtry (FIR decymacyjne/interpolacyjne i dyskretnej transformaty falkowej).
- Filtry estymacji parametrów predykcji liniowej i konwersji między reprezentacjami takimi jak parametry odbicia i pary linii widmowych lub częstotliwościami.
- Biblioteka matematyczna zawierająca macierze, algebrę liniową, sploty i funkcje wielomianów.
- Algorytmy statystyczne do operacji takich jak wartość minimalna, maksymalna, mediana, korelacja, sortowanie, histogram.
- Źródła (sinusowe i losowe) i ujścia (sinks) dla wprowadzania i eksportowania danych do i z przestrzeni roboczej MATLABa.
- Obsługa operacji stałoprzecinkowych z przepelnieniem, wartościami min-max i autoskalowaniem.
- Algorytmy optymalizacyjne automatycznie generujące kody ANSI/ISO C używane w systemach wbudowanych lub do przyspieszania wielkoskalowych symulacji.

Ad. e)

Kolekcja algorytmów do przetwarzania sygnałów analogowych i cyfrowych. Zawiera interfejs graficzny do interaktywnego projektowania i analizy, oraz funkcje dostępne z linii poleceń do rozwijania algorytmów zaawansowanych. Do zastosowań tego narzędzia należy przetwarzanie mowy i dźwięku, komunikacja przewodowa i bezprzewodowa, modelowanie i analiza dla inżynierii finansowej, aparatury medycznej i wizualizacji.

- Zestaw modeli sygnałowych i systemów liniowych.
- Narzędzia do projektowania, analizy i implementacji filtrów cyfrowych FIR i IIR.
- Narzędzia do projektowania filtrów analogowych.
- Dostęp do najczęściej używanych transformacji, jak FFT czy DCT.
- Narzędzia do analizy spektralnej i statystycznego przetwarzania sygnału.

- Funkcje do modelowania parametrycznych ciągów czasowych.
- Generowanie przebiegów falowych, w tym generator impulsów gaussowskich, przebiegów sinusoidalnych i prostokątnych.
- Interfejsy graficzne do projektowania, analizy i wizualizacji sygnałów (Signal Processing Tool), filtrów (Filter Design & Analysis Tool) i okien (Window design & Analysis Tool).

Ad. f)

Oprogramowanie zapewnia zaawansowaną funkcjonalność i analizę liniową modeli nieliniowych. Pozwala na aproksymację liniową modelu do analizy charakterystyk, takich jak odpowiedzi czasowe i częstotliwościowe i dynamikę zero-biegunową. Graficzny interfejs użytkownika (GUI) i możliwość programowania redukują złożoność i wymagany czas otrzymania modelu liniowego.

- Menadżer narzędzi kontroli i estymacji (Control and Estimation Tools Manager) stanowi graficzny interfejs użytkownika do zarządzania i organizacji procesu analizy liniowej.
- Nieingerująca analiza punkt po punkcie w otwartej pętli.
- Inspektor linearyzacji (LinearizationInspector) sprawdzający rezultaty linearyzacji blok po bloku.
- Obliczenia i zarządzanie punktem pracy.
- Wiersz poleceń dla wywoływania automatycznych skryptów linearyzujących i uruchamiania skryptów linearyzujących.

Ad. g)

Oprogramowanie umożliwiające budowanie i używanie modeli systemów dynamicznych z wejściowych i wyjściowych danych pomiarowych. Umożliwia użycie technik w dziedzinie czasowej i częstotliwościowej do dopasowania modeli do jedno- i wielokanałowych danych. Wszystkie funkcje modułu napisane zostały przez profesora Lennarta Ljunga, uznanego lidera na polu identyfikacji systemów.

- Parametryczna i nieparametryczna identyfikacja modeli wielowejsściowych i wielowyjściowych (MIMO) z wykorzystaniem danych czasowych i częstotliwościowych.
- Wyszczególnione narzędzia do identyfikacji modeli dynamicznych pierwszego, drugiego i trzeciego rzędu.
- Funkcje doradcze do oceny danych testowych i identyfikacji modeli.
- Narzędzia wstępnej analizy czasowych i częstotliwościowych danych testowych
- Narzędzia estymacji opóźnień czasowych i detekcji sprzężeń zwrotnych z danych testowych.

Ad. h)

Platforma dzięki której użytkownicy MATLAB mogą w jednym środowisku korzystać z obliczeń numerycznych i symbolicznych.

Zakres funkcjonalności modułu obejmuje:

- Algebrę liniową, transformaty, rachunek całkowy i różniczkowy.
- Upraszczenie wyrażeń symbolicznych.
- Rozwiązywanie równań.
- Specjalistyczne funkcje matematyczne.
- Podstawowe operacje symboliczne.
- Arytmetykę o zmiennej precyzji.
- Konwersję wyrażeń symbolicznych do kodu C, Fortranu lub LaTeX.
- Dostęp do jądra obliczeniowego MuPAD.

Ad. 2. Oprogramowanie do obliczeń inżynierskich (szt. 1)**Minimalne wymagania**

Elementy składowe:

1. Pakiet Matlab, Simulink i Toolboxy (1x licencja badawcza, wersja min. 2012a) lub równoważne:
 - a) Embedded Coder
 - b) Fixed-Point Toolbox
 - c) FuzzyLogicToolbox
 - d) MATLAB
 - e) MATLAB Coder
 - f) SimulinkCoder
 - g) xPC Target
 - h) SimulinkFixed Point

- i) Control System Toolbox
- j) DSP System Toolbox
- k) Signal Processing Toolbox
- l) Simulink
- m) Simulink Control Design
- n) System IdentificationToolbox

Oprogramowanie dostarczone na oryginalnym nośniku producenta oprogramowania. Gwarancja na nośnik min. 3 mc.

Ad. a)

Oprogramowanie musi zapewnić tworzenie kod C/C++ ze skryptów w języku MATLAB, z modeli Simulinka oraz diagramów Stateflow, które mają czytelność i efektywność profesjonalnych ręcznie pisanych kodów. Wygenerowany kod musi być zwięzły i szybki, co jest zasadniczym wymaganiem w systemach wbudowanych, docelowych płytach szybkiego prototypowania, mikroprocesorach używanych w produkcji masowej i symulatorach czasu rzeczywistego.

Funkcje programu:

- Tworzenie kod C/C++ i pliki wykonywalne ze skryptów MATLAB, z modeli Simulinka i Stateflow z porównywalnym zużyciem pamięci, prędkością uruchamiania i czytelnością do ręcznie pisanego kodu.
- Rozszerzenie MATLAB Coder i Simulink Coder o optymalizację i właściwości konfiguracyjne nieodzowne do rozmieszczenia produkcji.
- Obsługuje wszystkie obiekty danych i cechy słownika danych Simulinka włącznie ze zdefiniowanymi przez użytkownika klasami pamięci, typami i aliasami.
- Zapewnia intuicyjny graficzny interfejs użytkownika do tworzenia niestandardowych danych.
- Dzielnie wielostopniowe kodów dla efektywnego szeregowania z lub bez RTOS (system operacyjny czasu rzeczywistego).
- Zawiera rozszerzalne właściwości modułu pakującego.
- Zapewnia bogaty zbiór możliwości komentowania pomagający śledzić kod w modelach i wymaganiach.
- Weryfikuje kod automatycznie importując go do Simulinka do testowania metodą Software-in-the-loop.
- Tworzy dokumentację kodu zintegrowaną z eksploratorem modelu (Model Explorer) Simulinka i połączoną z modelem.
- Tworzy zdefiniowany przez użytkownika program główny opisujący dokładnie jak umieścić kod w docelowym środowisku.

Ad. b)

Oprogramowanie powinno umożliwić wykorzystywanie w tworzonych algorytmach i sekwencjach testowych stałoprzecinkowych typów danych i odpowiadających im funkcji, celem symulowania (model Simulink) i weryfikacji sprzętu wykorzystującego arytmetykę stałoprzecinkową.

Funkcje programu:

- Obsługa stałoprzecinkowych typów danych w środowisku MATLAB.
- Arytmetyka stałoprzecinkowa i operatory logiczne.
- Wsparcie dla wymiany danych między MATLABem a Simulinkiem, zapisanych w formacie stałoprzecinkowym.

Ad. c)

Oprogramowanie powinno pozwolić na rozszerzenie środowiska MATLABa o narzędzia do projektowania systemów opartych na logice rozmytej. Graficzny interfejs użytkownika (GUI) prowadzi poprzez kroki systemu projektowania wnioskowania rozmytego. Moduł powinien zapewnić funkcje dla wielu typowych metod logiki rozmytej.

Funkcje modułu:

- Wyspecjalizowane graficzne interfejsy użytkownika (GUI) do budowy rozmytych systemów wnioskujących oraz podglądu i analizy wyników.
- Funkcje przynależności do tworzenia systemów rozmytego wnioskowania.
- Obsługa logiki AND, OR i NOT w zdefiniowany przez użytkownika sposób.
- Standardowy system wnioskowania rozmytego typu Mamdani i Sugeno.
- Zautomatyzowane funkcje przynależności kształtujące poprzez techniki uczenia neuroadaptywne i rozmytej analizy skupień.
- Zdolność umieszczania systemu rozmytego wnioskowania w modelu Simulinka.
- Zdolność do generowania kodu C lub samodzielnie wykonywalnego enginu rozmytego wnioskowania.

Ad. d)

Program musi posiadać następujące rozwiązania programistyczne:

1. **Język programowania wysokiego poziomu** - indeksacja niejawna pozwalająca na efektywną wektoryzację obliczeń i zwarty zapis (konstrukcje z użyciem dwukropka, nawiasów kwadratowych, połączenie kropki i operatorów arytmetycznych, indeksacja za pomocą warunków logicznych, wektoryzacja funkcji wbudowanych), domyślne określanie typów i wymiarowanie tablic umożliwiające szybkie prototypowanie aplikacji w środowisku interpretera programu, automatyczna generacja kodu C wykorzystywana do budowania niezależnych aplikacji na bazie kodu programu, różnorodne typy zmiennych: numeryczne - stałoprzecinkowe i zmiennoprzecinkowe, tablice wielowymiarowe, zmienne tekstowe, zmienne strukturalne - dostęp do pola przez nazwę (notacja kropkowa), tablice wskaźnikowe - cellarrays (notacja klamrowa)
2. **Otwartą architekturę** - rozszerzalność (możliwość budowania własnych funkcji użytkownika: pliki, skrypty i funkcje), dołączanie skompilowanego kodu w C lub w Fortranie: biblioteki DLL - MEX-pliki), przenośność (jednokodowy kod programu - kod dla różnych platform PC: Windows 95/98/NT, Linux, Mac i UNIX: Sun Solaris, HP UX, IBM AIX, SGI Irix i innych), wymiana danych z innymi aplikacjami poprzez pliki i różne protokoły komunikacyjne: DDE, TCP/IP (internet)- Web Server, ODBC/JDBC (bazy danych), współpraca z innymi aplikacjami w środowisku wielozadaniowym (możliwość uruchamiania innych aplikacji z poziomu programu - polecenia poprzedzone znakiem!, udostępnianie usług Engine Services)
3. **Orientację obiektową** - grafika uchwytów (hierarchiczna struktura Handle Graphics - pobieranie (get) i nadawanie (set) wartości atrybutów określanych zmiennymi tekstowymi), obiekty standardowe (np. LTI - liniowy, stacjonarny system dynamiczny, obiekt sym - wyrażenie symboliczne), narzędzia programowania obiektowego (hermetyzacja, dziedziczenie, przeciążanie).
4. Narzędzie do projektowania systemów opartych na logice rozmytej. Wspecjalizowane graficzne interfejsy użytkownika (GUI) do budowy rozmytych systemów wnioskujących oraz podglądu i analizy wyników; funkcje przynależności do tworzenia systemów rozmytego wnioskowania. Obsługa logiki AND, OR i NOT w zdefiniowany przez użytkownika sposób. Standardowy system wnioskowania rozmytego typu Mamdani i Sugeno. Zautomatyzowane funkcje przynależności kształtujące poprzez techniki uczenia neuroadaptacyjnej rozmytej analizy skupień. Zdolność umieszczania systemu rozmytego wnioskowania w modelu. Zdolność do generowania kodu C lub samodzielnie wykonywalnego enginu rozmytego wnioskowania.
5. Narzędzie do optymalizacji algorytmów genetycznych i algorytmów przeszukiwania bezpośredniego. Graficzny interfejs użytkownika i dostęp do funkcji z linii poleceń do szybkiego ustawiania parametrów problemu i opcji algorytmu oraz monitorowania postępu działania. Narzędzia do stosowania algorytmów genetycznych z możliwością dopasowania wielu opcji, w tym krzyżowanie, mutacja, selekcja. Algorytmy przeszukiwania bezpośredniego z możliwością zdefiniowania rozmiaru siatki. Funkcje integrujące rozszerzenie i wbudowane procedury programu z algorytmami genetycznymi i przeszukiwania bezpośredniego. Obsługa automatycznej generacji kodu programu.
6. Narzędzie do przetwarzania obrazów, analizy, wizualizacji i rozwijania własnych algorytmów. Poprawianie obrazu, w tym filtrowanie liniowe i nieliniowe, projektowanie filtrów, wyostanie i automatyczna regulacja kontrastu. Analiza obrazu, w tym analiza teksturowa, detekcja linii, morfologia, detekcja krawędzi, segmentacja, przetwarzanie ROI (Rejon Zainteresowania), pomiar parametrów. Przetwarzanie obrazów barwnych, konwersja palety kolorów, import i eksport profili ICC. Transformacje przestrzenne i rejestracja obrazów, w tym narzędzie graficzne do wyboru punktów kontrolnych. Transformacje obrazów, takie jak FFT, DCT, Radon i rzutowanie wiązki wachlarzowej. Import i eksport DICOM. Interaktywny podgląd obrazów i modularne narzędzia do tworzenia własnego Graficznego Interfejsu Użytkownika (GUI) dla wybranych obrazów. Obsługa wielowymiarowego przetwarzania obrazu.
7. Narzędzie do projektowania, implementacji, wizualizacji i symulacji sieci neuronowych. Graficzny interfejs użytkownika do tworzenia, uczenia i symulowania sieci neuronowych. Wsparcie dla najpowszechniejszych architektur (nadzorowanych i bez nauczyciela). Kompletny zestaw funkcji uczących sieć. Zbiór bloków programu wraz z dokumentacją i demonstracjami aplikacji systemów sterowania. Automatyczne tworzenie modeli programu z obiektów sieci neuronowych. Modularna reprezentacja sieci, pozwalająca na zadawanie dowolnej liczby warstw wejściowych i dowolnej liczby połączeń między warstwami. Zwiększanie skuteczności uczenia się dzięki funkcjom przetwarzającym dane przed i po uczeniu sieci. Procedury poprawiające generalizację. Funkcje wizualizacyjne pokazujące skuteczność sieci.
8. Narzędzie do wielowymiarowej optymalizacji średniej i wielkiej skali. Funkcje dostępne z linii poleceń programu powinny posiadać łatwy do modyfikacji otwarty kod źródłowy. Rozszerzenie powinno zawierać funkcje do: definiowania, rozwiązywania i oceny problemów optymalizacji. Optymalizacji nieliniowej i wielokryterialnej. Obliczeń regresji średniokwadratowej i nieliniowej. Rozwiązywania zadań programowania kwadratowego i liniowego.
9. Narzędzie do tworzenia aplikacji programu i uruchomienie ich na klastrach komputerów, bez opuszczania własnego stanowiska pracy. Rozszerzenie powinno prototypować aplikacje w programie i wykorzystywać funkcjonalność rozszerzenia aby zdefiniować niezależne i współdziałające zadania. Algorytmy, które wymagają współdziałających zadań, wykorzystują interfejs MPI (Message Passing Interface). Program powinien szeregować i uruchamiać zadania na wielu zdalnych sesjach programu, redukując czas wykonania w porównaniu z pojedynczą

sesją.

10. Narzędzie do rozwiązywania równań różniczkowych cząstkowych (PDE) w dwóch wymiarach i w czasie, metodą elementów skończonych. Kompletne GUI do pre- i post-procesowania dwuwymiarowych PDE. Automatyczny i adaptacyjny wybór siatki. Tworzenie geometrii za pomocą Constructive Solid Geometry. Specyfikacja warunków brzegowych: Dirichleta, Neumanna lub mieszane. Wykorzystanie składni programu do elastycznej specyfikacji współczynników i PDE. W pełni zautomatyzowane generowanie siatki i jej oczyszczanie. Jednoczesna wizualizacja wielu własności rozwiązania, nakładanie siatki FEM i animacje.
11. Narzędzie do przetwarzania sygnałów analogowych i cyfrowych. Zestaw modeli sygnałowych i systemów liniowych. Narzędzia do projektowania, analizy i implementacji filtrów cyfrowych FIR i IIR. Narzędzia do projektowania filtrów analogowych. Dostęp do najczęściej używanych transformat, jak FFT czy DCT. Narzędzia do analizy spektralnej i statystycznego przetwarzania sygnału. Funkcje do modelowania parametrycznych ciągów czasowych. Generowanie przebiegów falowych, w tym generator impulsów gaussowskich, przebiegów sinusoidalnych i prostokątnych. Interfejsy graficzne do projektowania, analizy i wizualizacji sygnałów, filtrów i okien.
12. Narzędzie do analizy historii danych, modelowania danych, symulowania systemów, rozwijania algorytmów statystycznych oraz nauczania statystyki.

Obliczanie i dopasowywanie rozkładów prawdopodobieństwa. Modelowanie liniowe i nieliniowe. Statystyka wielu zmiennych. Statystyka opisowa. Analiza wariancji (ANOVA). Testowanie hipotez. Statystyka przemysłowa.

Wizualizacja danych statystycznych

Ad. e)

Moduł powinien zapewnić generowanie niezależnego kodu C/C++ z kodu języka MATLAB. Wspieranie podzbiorów cech i właściwości języka MATLAB włączając w to:

- instrukcje sterujące przepływem programu
- funkcje
- operacje macierzowe
- wygenerowanie gotowej MEX-funkcji,

Ad. f)

Moduł powinien zapewnić generowanie i uruchamianie samodzielnego kodu C do rozwijania i testowania algorytmów modelowanych w Simulinku.

Funkcje modułu:

- Tworzenie kodu ANSI/ISO C, programy wykonywalne dla modeli dyskretnych, ciągłych i hybrydowych.
- Używanie bloków modelu do tworzenia przyrostowego i budowy kodu dla dużych aplikacji.
- Obsługuje cechy słownika danych Simulinka dla całkowitych, zmiennie- i stałoprzecinkowych typów danych.
- Tworzy kod dla modeli o jednej i zmiennej częstotliwości próbkowania oraz asynchronicznych.
- Obsługuje jedno- i wielozadaniowe systemy operacyjne oraz środowiska bezoperacyjne (bez systemu operacyjnego).
- Wykonuje optymalizację kodu polepszającą szybkość uruchamiania programu.
- Zapewnia możliwości do konfigurowania i legalnego ingerowania w kod.
- Pozwala interaktywnie dopasowywać i monitorować utworzony kod wewnątrz lub na zewnątrz Simulinka.

Ad. g)

Środowisko do prototypowania docelowego umożliwiające podłączenie modeli Simulinka i Stateflow do fizycznych systemów i uruchamianie ich w czasie rzeczywistym na urządzeniach kompatybilnych z architekturą PC. xPC Target posiada udowodnione możliwości do szybkiego prototypowania i symulacji metodą hardware-in-the-loop do systemów sterujących i przetwarzania sygnałów oraz jest wykorzystywany do projektów w każdej znaczącej branży.

Funkcje modułu:

- Uruchamia aplikacje generowane przez modele Simulinka i Stateflow używając jądra czasu rzeczywistego na dowolnym komputerze PC.
- Obsługuje dowolny komputer PC jako docelowy system czasu rzeczywistego.
- Uzyskuje próbki z częstotliwością do 100 KHz, w zależności od poziomu wydajności procesora i rozmiaru modelu.
- Obsługuje więcej niż 250 standardów płyt wejścia-wyjścia z rozległą biblioteką sterowników urządzeń wejścia-wyjścia wraz z kodem źródłowym sterowników.

- Umożliwia rejestrację sygnałów i konfigurowanie parametrów z lokalnego lub docelowego komputera PC.
- Rejestruje i zapisuje dane w czasie rzeczywistym do pamięci RAM lub systemu plików na docelowym komputerze PC.
- Wyświetla dane i ślady sygnałów na lokalnym, docelowym lub obu komputerach PC.
- Zapewnia lokalny zintegrowany graficzny interfejs użytkownika i zorientowany obiektowo interfejs linii poleceń MATLABa zapewniający dostęp do docelowej aplikacji uruchomionej na jednym lub kilku komputerach PC.
- Oferuje komunikację między komputerami poprzez RS-232 lub protokół TCP/IP (bezpośredni, LAN lub internet).
- Oferuje możliwość rozwoju własnego graficznego interfejsu użytkownika umożliwiającego dostęp do sygnałów i parametrów.

Ad. h)

Moduł umożliwiający zastosowanie istotnych właściwości stałoprzecinkowych w rodzinie produktów Simulinka pozwalając projektować systemy kontrolne i układy przetwarzania sygnałów, które będą implementowane używając stałoprzecinkowej arytmetyki.

Funkcje modułu

- Umożliwia symulacje stałoprzecinkowe w Simulinku, Stateflow i innych modułach.
- Umożliwia stałoprzecinkowe generowanie kodu.
- Obsługuje szybkie prototypowanie stałoprzecinkowe w xPC Target i innych systemach szybkiego prototypowania.
- Pozwala kontrolować stałoprzecinkowe typy danych i skalowanie.
- Zapewnia narzędzia do identyfikacji błędów przepełnienia i nasycenia.
- Zawiera narzędzia automatycznego skalowania zamieniające zakres i precyzję.
- Umożliwia automatyczną manipulację regulacją skali, zaawansowaną matematyką i innymi zadaniami.

Ad. i)

Wyspecjalizowane narzędzia modelowania, projektowania i analizy systemów ze sprzężeniami, obejmując zarówno klasyczne jak i nowoczesne metody projektowe.

Funkcje modułu:

- LTI Viewer - interaktywny graficzny interfejs użytkownika (GUI) do analizy i porównywania liniowych systemów niezmiennych w czasie (LTI systems).
- Wykresy w dziedzinie czasu odpowiedzi na skok jednostkowy, impulsowej i zero-biegunowej oraz odpowiedzi w dziedzinie częstotliwości (Bode, Nyquist i Nichols) w jednym oknie.
- Narzędzie projektowania systemów z pojedynczym wejściem i pojedynczym wyjściem (SISO Design Tool) - interaktywny GUI do analizy i konfiguracji systemów kontroli SISO używając wykresów linii pierwiastkowych, Bodego i Nicholasa.
- Graficzny kompensator ograniczający w SISO Design Tool.
- Interaktywny wykres odpowiedzi w zamkniętej pętli ciągle aktualizowany wraz z modyfikacją kompensatora w SISO Design Tool.
- Opcje zapisu i odczytu umożliwiające archiwizację projektów z wielokrotnymi kompensatorami.
- Pełna kontrola nad wizualizacją w dziedzinie czasu i częstotliwości.
- Specjalne struktury danych zwane obiektami LTI (LTI Objects) dla zwięzłej reprezentacji formatów danych funkcji przejścia, przestrzeni stanu, zer i biegunów i odpowiedzi częstotliwościowych.
- Obsługa systemów wielowejsściowych i wielowyjściowych (MIMO), systemów czasu ciągłego i próbkowanych danych oraz systemów z opóźnieniami czasowymi.
- Obsługa różnorodnych metod przekształceń dyskretnych na ciągłe.
- Narzędzia nowoczesnych technik kontroli projektowania (umieszczanie biegunów, regulacje LQR-LQG, projektowanie filtrów Kalmana, wyznaczanie obserwowalności i sterowalności i rozwiązania równań Riccatiego i Lapunowa).

Ad. j)

Funkcje modułu:

- Obsługa symulacji opartej na ramkach i próbkach systemów wielokanałowych i o zmiennej częstotliwości próbkowania oraz stałoprzecinkowych i zmiennoprzecinkowych typach danych.
- Algorytmy transformat (FFT i DCT), bloki estymacji widma i funkcji okien.
- Algorytmy projektowania i realizacji filtrów cyfrowych.
- Filtry Adaptatywne (RLS, LMS i Kalmana) i filtry (FIR decymacyjne/interpolacyjne i dyskretnej transformaty

falkowej).

- Filtry estymacji parametrów predykcji liniowej i konwersji między reprezentacjami takimi jak parametry odbicia i pary linii widmowych lub częstotliwościami.
- Biblioteka matematyczna zawierająca macierze, algebrę liniową, sploty i funkcje wielomianów.
- Algorytmy statystyczne do operacji takich jak wartość minimalna, maksymalna, mediana, korelacja, sortowanie, histogram.
- Źródła (sinusowe i losowe) i ujścia (sinks) dla wprowadzania i eksportowania danych do i z przestrzeni roboczej MATLABa.
- Obsługa operacji stałoprzecinkowych z przepełnieniem, wartościami min-max i autoskalowaniem.
- Algorytmy optymalizacyjne automatycznie generujące kody ANSI/ISO C używane w systemach wbudowanych lub do przyspieszania wielkoskalowych symulacji.

Ad. k)

Kolekcja algorytmów do przetwarzania sygnałów analogowych i cyfrowych.

Funkcje modułu:

- Zestaw modeli sygnałowych i systemów liniowych.
- Narzędzia do projektowania, analizy i implementacji filtrów cyfrowych FIR i IIR.
- Narzędzia do projektowania filtrów analogowych.
- Dostęp do najczęściej używanych transformat, jak FFT czy DCT.
- Narzędzia do analizy spektralnej i statystycznego przetwarzania sygnału.
- Funkcje do modelowania parametrycznych ciągów czasowych.
- Generowanie przebiegów falowych, w tym generator impulsów gaussowskich, przebiegów sinusoidalnych i prostokątnych.
- Interfejsy graficzne do projektowania, analizy i wizualizacji sygnałów (Signal Processing Tool), filtrów (Filter Design & Analysis Tool) i okien (Window design & Analysis Tool).

Ad. l)

Platforma do wielodomenowej symulacji i projektowania systemów dynamicznych z wykorzystaniem modeli. Stanowi interaktywne środowisko graficzne i zbiór bibliotek konfigurowalnych z blokami pozwalającymi dokładnie projektować, symulować, implementować i kontrolować przetwarzanie sygnałów, telekomunikację i inne systemy zdefiniowane w dziedzinie czasu.

Dodatkowe moduły rozszerzają środowisko o narzędzia do modelowania i projektowania specyficznych zadań oraz generacji kodu, implementacji algorytmów, testowania i weryfikacji. Integracja z MATLABem zapewnia bezpośredni dostęp do szerokiego wachlarza narzędzi do budowy algorytmów, analizy i wizualizacji danych oraz obliczeń numerycznych.

Główne cechy programu:

- Obszerne i dające się powiększać biblioteki zdefiniowanych bloków.
- Interaktywny edytor graficzny pozwalający intuicyjnie asemblować i zarządzać diagramami bloków.
- Zdolność zarządzania złożonymi projektami poprzez segmentację modelu i hierarchizację komponentów projektu.
- Eksplorator modelu (Model Explorer) nawigujący, tworzący, konfigurujący i przeszukujący wszystkie sygnały, parametry i właściwości modelu.
- Zdolność do komunikowania się z innymi programami symulacyjnymi i dołączania ręcznie pisanego kodu włącznie z algorytmami MATLABa.
- Możliwości uruchamiania stało- lub zmiennokrokowych symulacji w systemach w dziedzinie czasu interaktywnie lub za pomocą symulacji batch.
- Funkcje interaktywnego definiowania wejść i wyjść podglądowych do oszacowywania zachowania się modelu.
- Graficzny debugger do sprawdzania wyników symulacji i diagnozowania nieoczekiwanego zachowania się projektu.
- Pełny dostęp do MATLABa w celu analizy i wizualizacji danych, poszerzenia graficznego interfejsu i definiowania danych i parametrów modelu.
- Narzędzia diagnozy i analizy modelu pozwalające upewnić się, co do spójności modelu i wykryć błędy modelowania.

Ad. m)

Moduł musi zapewnić zaawansowaną funkcjonalność i analizę liniową modeli nieliniowych. Pozwala na aproksymację liniową modelu do analizy charakterystyk, takich jak odpowiedzi czasowe i częstotliwościowe i dynamikę zero-biegunową. Graficzny interfejs użytkownika (GUI) i możliwość programowania redukują złożoność i wymagany czas otrzymania modelu liniowego.

Funkcje modułu:

- Menadżer narzędzi kontroli i estymacji (Control and Estimation Tools Manager) stanowi graficzny interfejs użytkownika do zarządzania i organizacji procesu analizy liniowej.
- Nieingerująca analiza punkt po punkcie w otwartej pętli.
- Inspektor linearyzacji (Linearization Inspector) sprawdzający rezultaty linearyzacji blok po bloku.
- Obliczenia i zarządzanie punktem pracy.
- Wiersz poleceń dla wywoływania automatycznych skryptów linearyzujących i uruchamiania skryptów linearyzujących.

Ad. n)

Moduł musi zapewnić budowanie i używanie modeli systemów dynamicznych z wejściowych i wyjściowych danych pomiarowych. Umożliwiać użycie technik w dziedzinie czasowej i częstotliwościowej do dopasowania modeli do jedno- i wielokanałowych danych.

Funkcje modułu:

- Parametryczna i nieparametryczna identyfikacja modeli wielowejsiowych i wielowyjściowych (MIMO) z wykorzystaniem danych czasowych i częstotliwościowych.
- Wyspecjalizowane narzędzia do identyfikacji modeli dynamicznych pierwszego, drugiego i trzeciego rzędu.
- Funkcje doradcze do oceny danych testowych i identyfikacji modeli.
- Narzędzia wstępnej analizy czasowych i częstotliwościowych danych testowych (usuwanie offsetu, detrending, wstępne filtrowanie i rekonstrukcja brakujących danych).
- Narzędzia estymacji opóźnień czasowych i detekcji sprzężeń zwrotnych z danych testowych.
- Bloki Simulinka do identyfikacji i obsługi estymowanych modeli i transportu danych do i z przestrzeni roboczej MATLABa.

Równoważność:

Zamawiający dopuszcza składanie ofert równoważnych.

Oferowane przez Wykonawców, składających oferty równoważne oprogramowania, muszą posiadać wszystkie parametry techniczne i funkcjonalne nie gorsze niż oprogramowanie wskazane w specyfikacji. Oferowane rozwiązania równoważne muszą poprawnie współpracować ze wszystkimi pozostałymi elementami proponowanego rozwiązania. W przypadku braku zgodności Wykonawca musi dostarczyć i wdrożyć elementy o odpowiedniej funkcjonalności, które poprawnie współpracują z resztą proponowanego rozwiązania. Wykonawca, który powołuje się na rozwiązania równoważne w stosunku do wskazanych przez Zamawiającego oprogramowań, obowiązany jest wykazać, że oferowane przez niego oprogramowania spełniają wszystkie wymagania określone przez Zamawiającego. W tym celu do oferty należy załączyć foldery, specyfikacje techniczne proponowanych oprogramowań lub inne dokumenty zawierające opisy techniczne i funkcjonalne.

Opisy techniczne i funkcjonalne, które będą stanowić podstawę do oceny równoważności.

Równoważne oprogramowanie winno być określone z nazwy oraz poprzez podanie producenta.

1. Równoważność oferowanego artykułu Wykonawca obowiązany jest wykazać poprzez wskazanie w przedkładanej ofercie właściwości oprogramowania (opis właściwości technicznych i funkcjonalności).

2. Wskazane przez Wykonawcę właściwości techniczne i funkcjonalne oprogramowania zamiennego musi potwierdzać załączona do oferty informacja pochodząca od producenta tego oprogramowania.

3. Zgodnie z art. 30 ust 4 ustawy z dnia 29 stycznia 2004r. Prawo zamówień publicznych, tekst jednolity: Dz. U. z 2010 r. nr 113, poz. 759 ze zm., Zamawiający, powołując się na oznaczenie normy, dopuszcza normalizację, specyfikacje techniczne, równoważne wskazanym w kolejności ważności zastosowania w wskazanym art. 30 ust 1 ww. Ustawy. W tym przypadku, gdy Wykonawca powoła się na system jakościowy równoważny, po jego stronie leży wykazanie dowodu równoważności.

Zamawiający przed wyborem najlepszej oferty ma prawo zażądać od Wykonawcy proponującego rozwiązanie równoważne udostępnienia na jego koszt oferowanego przez niego oprogramowania do sprawdzenia w celu potwierdzenia równoważności.

