

OPIS PRZEDMIOTU ZAMÓWIENIA

Dostawa licencji specjalistycznego oprogramowania Matlab/Simulink Campus Wide Suite na okres 12 miesięcy

Wieloparadygmatyczny język programowania i środowisko obliczeń umożliwiające manipulowanie macierzami, wykreślanie funkcji i danych, implementację algorytmów, tworzenie interfejsów użytkownika i współpracę z programami napisanymi w innych językach.

- 1) Możliwość instalacji na systemach operacyjnych Windows 10/11, Linux oraz MacOS i wspierać 64-bitową architekturę procesorów.
- 2) Zintegrowany język wysokiego poziomu do obliczeń numerycznych, wizualizacji i tworzenia aplikacji.
- 3) Interaktywne środowisko dla iteracyjnej analizy i rozwiązywania problemów.
- 4) Wbudowane funkcje matematyczne wspomagające obliczenia z zakresu algebry liniowej, statystyki, analizy Fouriera, filtrowania, optymalizacji oraz rozwiązywania równań różniczkowych zwyczajnych.
- 5) Interaktywne narzędzia do eksploracji i wizualizacji danych (2D i 3D).
- 6) Narzędzia służące utrzymaniu przejrzystości oraz poprawności kodu a także maksymalizacji jego wydajności.
- 7) Narzędzia do tworzenia interfejsu graficznego dla aplikacji (GUI).
- 8) Funkcje integrujące algorytmy opracowane w środowisku z zewnętrznymi aplikacjami oraz językami programowania tj. C, Java, Python, .NET, and Microsoft® Excel®.
- 9) Zestaw narzędzi umożliwiający:
 1. Generacje sygnałów uplink i downlink zgodnych z 5G.
 2. Graficzna aplikacja do interaktywnej generacji modeli testowych 5G (NR-TM) oraz sygnałów FRC uplink i downlink wraz z uwzględnieniem zakłóceń w sygnale, takich jak AWGN, offset fazy, częstotliwości, składowej stałej, *IQ imbalance*, nieliniowości. Wizualizacja wyników na diagramach konstelacji, analizatorach widma, siatce OFDM i wykresach czasowych.
 3. Symulacje na poziomie łącza, symulacje BLER z modelami kanału propagacji TR 38.901.
 4. Charakteryzacja i symulacja linia modeli kanałów *tapped delay line* (TDL) i *cluster delay line* (CDL).
 5. Pomiar przepustowości kanałów PDSCH i PUSCH.
 6. Estymacja i equalizacja odebranych sygnałów 5G NR.
 7. Ocenę wydajności nadajników 5G NR oraz testowanie odbiorników w obecności interferencji.
 8. Pomiar parametrów łącza – ACLR i EVM.
 9. Funkcje przetwarzania sygnałów, w tym kodowanie kanałów (LDPC i kody biegunowe), dekodowanie MIB, estymacja kanału, synchronizacja i wyrównanie.
 10. Symulacje na poziomie systemu strategii planowania w warstwie MAC w trybach FDD i TDD.
 11. Pełny dostęp do jawnego kodu modułu z możliwością jego edycji.
 12. Wsparcie generacji kodu C i C++.
 13. Bloki Simulinka do modelowania, symulacji i analizy działania statków powietrznych, w tym ich układów napędowych, systemów sterowania, siłowników, przy uwzględnieniu właściwości masowych.

14. Uwzględnienie dynamiki lotu, w tym modeli równań ruchu z trzema oraz sześcioma stopniami swobody o stałej lub zmiennej masie.
15. Wizualizacja parametrów lotu z użyciem standardowych przyrządów pokładowych.
16. Możliwość wizualizacji maszyny w trakcie lotu z uwzględnieniem zjawisk dynamicznych dzięki interfejsowi z symulatorem lotu FlightGear oraz silnikowi Unreal Engine.
17. Dostęp do standardowych modeli zachowania pilotów, a także modeli środowiska, uwzględniających warunki atmosfery, grawitacji, wiatru i pola magnetycznego.
18. Import współczynników aerodynamicznych z kompendium danych cyfrowych US Air Force (DATCOM).
19. Predefiniowane narzędzia do konwersji jednostek, przekształcenia układów współrzędnych i reprezentacji przestrzennych oraz do wyznaczania parametrów lotu.
20. Możliwość wysokopoziomowego planowania misji dla statków kosmicznych i niewielkich satelit.
21. Analiza ruchu statków powietrznych za pomocą dedykowanych funkcji i narzędzi.
22. Wizualizacja parametrów lotu z użyciem standardowych przyrządów pokładowych.
23. Dostęp do standardowych modeli środowiska, uwzględniających warunki atmosfery, grawitacji, wiatru i pola magnetycznego.
24. Predefiniowane narzędzia do konwersji jednostek, przekształcenia układów współrzędnych i reprezentacji przestrzennych oraz do wyznaczania parametrów lotu.
25. Możliwość wizualizacji maszyny w trakcie lotu z uwzględnieniem zjawisk dynamicznych dzięki interfejsowi z symulatorem lotu FlightGear.
26. Import współczynników aerodynamicznych z kompendium danych cyfrowych US Air Force (DATCOM).
27. Projektowanie, analiza i wizualizacja anten z użyciem predefiniowanych lub niestandardowych elementów.
28. Projektowanie liniowych, prostokątnych, konforemnych i niestandardowych szyków antenowych.
29. Analiza dużych szyków z wykorzystaniem modelu szyków nieskończonych lub podejścia wykorzystującego wbudowane wzorce elementów.
30. Specyfikacja nieskończonej płaszczyzny uziemienia do analizy anten montowanych na bardzo dużych strukturach.
31. Wykorzystanie metody momentów do analizy właściwości portów anten i szyków antenowych, takich jak impedancja, straty odbiciowe czy parametry rozpraszania.
32. Analiza pola promieniowania wzorca, pola elektromagnetycznego oraz szerokości wiązki anten i szyków antenowych dla niestandardowych danych.
33. Analiza powierzchniowa anten i szyków antenowych: rozptywu prądu, rozkładu ładunku oraz siatki powierzchni.
34. Możliwość generacji plików Gerbera z zaprojektowanych rozwiązań, do celów projektowania anten PCB.
35. Wizualizacja pokrycia anteny na powierzchniowych mapach terenu 3D w oparciu o różne modele propagacji.
36. Estymacja efektywności i strat w antenach o skończonej konduktywności i grubości z wykorzystaniem katalogu materiałów takich jak m.in. miedź, cyna, złoto.
37. Projektowanie i testowanie systemów przetwarzania audio.

38. Algorytmy przetwarzania audio (filtrowanie, procesory dynamiczne, efekty audio), źródła sygnału (oscylatory audio, syntezy tablicowe), pomiary akustyczne (estymacja odpowiedzi impulsowej, filtry oktafowe, filtry korekcyjne ważone krzywymi A i C).
39. Algorytmy do ekstrakcji cech z sygnałów mowy (m.in. melowe współczynniki cepstralne MFCC).
40. Interfejsy dla sterowników audio o niskich latencych do obsługi kart dźwiękowych.
41. Interaktywne aplikacje graficzne do testowania i strojenia wtyczek audio, pomiarów odpowiedzi impulsowej oraz do etykietowania sygnałów audio.
42. Interfejsy do kontrolerów MIDI.
43. Wytrenowany system rozpoznawania mówcy z możliwością modyfikacji.
44. Funkcje do przetwarzania dźwięku przestrzennego.
45. Generacja wtyczek VST i AU dla środowisk DAW (*Digital Audio Workstations*).
46. Projektowanie, symulacja i testowanie systemów ADAS oraz systemów autonomicznej jazdy.
47. Interaktywna aplikacja do wykrywania i etykietowania obiektów na obrazach zarejestrowanych przez czujnik kamerowy.
48. Algorytmy przetwarzania obrazów do celów detekcji i klasyfikacji oznaczeń na jezdni, detekcji pojazdów i transformacji współrzędnych obraz-pojazd.
49. Narzędzia do wizualizacji danych zarejestrowanych przez czujniki, pokrycia czujników oraz śledzonych obiektów, w różnych układach odniesienia.
50. Możliwość importu i analizy map wysokiej rozdzielczości HERE HD oraz sieci map otwartego formatu OpenDRIVE.
51. Algorytmy fuzji danych z wielu czujników, śledzenia wielu obiektów równocześnie, detekcji toru jazdy.
52. Tworzenie scenariuszy jazdy (definiowanie dróg, aktorów, pojazdów, czujników i wizualizacja całości) w prostym środowisku graficznym w MATLABie oraz w realistycznym środowisku 3D Unreal Engine® firmy Epic Games®.
53. Możliwość importu predefiniowanych scenariuszy jazd testowych Euro NCAP.
54. Uruchamianie scenariuszy jazdy w ramach testów otwartej i zamkniętej pętli.
55. Możliwość generacji komponentów programowych na podstawie schematów blokowych algorytmów dla architektury AUTOSAR klasycznej oraz adaptacyjnej (w tym R19-03).
56. Interfejs do edycji komponentów AUTOSAR (porty, interfejsy i inne konfiguracje).
57. Wsparcie dla wybranych narzędzi do symulacji systemów dynamicznych, do symulacji działania komponentów programowych (wraz z funkcjonalnością) oraz oprogramowania podstawowego (BSW) i usług jak np. menadżer zdarzeń diagnostycznych czy NVRAM.
58. Wsparcie dla wybranych narzędzi do automatycznej generacji kodu C i C++, testowania i analizy pokrycia testami, zarządzania wymaganiami, weryfikacji algorytmów pod względem zgodności z wybranymi normami DO, analizy formalnej algorytmów oraz statycznej analizy kodu źródłowego
59. Możliwość generacji plików ARXML zgodnych ze standardem AUTOSAR Schema 4.4.

60. Wsparcie dla workflow *bottom-up*, *top-down* i *roundtrip*, możliwość importu oraz eksportu dokumentacji ARXML, wsparcie dla innych środowisk programistycznych do projektowania zgodnego ze standardem AUTOSAR.
61. Generowanie aplikacji wykonywalnych na systemy Linux z komponentów tworzonych na platformę AUTOSAR Adaptive
62. Analizy i wizualizacje związane z technologią NGS umożliwiającą określanie sekwencji DNA/RNA, włączając w to funkcje wspomagające dopasowanie dwóch lub więcej sekwencji oraz wykrywanie szczytów.
63. Analizy danych z mikromacierzy obejmujące odczytywanie, filtrowanie, normalizowanie oraz wizualizacje.
64. Zestaw funkcji wspomagających spektrometrię mas, w tym przetwarzanie, klasyfikacja i identyfikacja markerów.
65. Analizy drzewa filogenetycznego oraz sekwencji białka.
66. Wbudowane funkcje związane z teorią grafów.
67. Obsługa genomicznych, proteomicznych i genowych formatów plików, włączając w to pliki: SAM, FASTA, FASTQ, CEL i CDF oraz bazy danych takie jak NCBI oraz GenBank.
68. Zapewnia oparte na standardach narzędzia do projektowania, symulowania i weryfikowania systemów komunikacji Bluetooth.
69. Obsługuje generowanie przebiegów testowych, weryfikację „golden reference” i modelowanie sieci Bluetooth.
70. Umożliwia konfigurowanie, symulowanie i analizowanie łącz komunikacji Bluetooth.
71. Umożliwia również tworzenie i ponowne wykorzystywanie bramek testowych w celu sprawdzenia projektów, prototypów i implementacji pod kątem zgodności ze standardem Bluetooth, włącznie z Bluetooth Low Energy (LE) i Bluetooth Classic.
72. Pozwala na ocenę współistnienia, zakłócenia, lokalizacji i scenariuszy LE Audio poprzez zamodelowanie wielu warstw protokołu Bluetooth.
73. C2000 Microcontroller Blockset umożliwia modelowanie algorytmów do układów cyfrowej konwersji mocy oraz sterowania napędami elektrycznymi opartych o mikrokontrolery Texas Instruments C2000.
74. Blockset zawiera gotowe bloki peryferiów sprzętowych procesora np. wejścia/wyjścia cyfrowe, ADC, ePWM aby symulować kompletne algorytmy sterowania, które szczególnie w zastosowaniach przemysłowych i automotive wymagają synchronizacji na poziomie wejść/wyjść (np. ADC/PWM).
75. Blockset umożliwia również projektowanie algorytmów z uwzględnieniem wieloprotocowej architektury sprzętowej biorąc pod uwagę komunikację pomiędzy rdzeniami (IPC) oraz koprocesorem np. Control Law Accelerator.
76. Utworzony model może zostać bezpośrednio połączony z wspieranymi platformami sprzętowymi, aby pozyskiwać rzeczywiste dane używając wejść/wyjść mikrokontrolera lub użyć go jako szybki prototyp tworzonego rozwiązania.
77. Z zaprojektowanego algorytmu możliwe jest wygenerowanie kodu produkcyjnego zoptymalizowanego dla procesorów z rodziny C2000 wykorzystującego biblioteki np. IQ Math oraz pozwalającego na monitorowanie sygnałów w czasie rzeczywistym, strojenie parametrów oraz wykonywanie testów processor-in-the-loop (PIL) z użyciem narzędzia Embedded Coder .

78. Algorytmy służące do analizy, projektowania, symulacji i weryfikacji warstwy fizycznej systemów komunikacyjnych.
79. Uwzględnienie kodowania źródłowego i kanałowego, modulacji, technologii MIMO i OFDM w celu realizacji standardowych i zmodyfikowanych systemów bezprzewodowej komunikacji.
80. Narzędzia do analizy i wizualizacji parametrów kanału i sygnału, takie jak parametr BER, EVM, diagramy oka i diagramy konstelacji.
81. Narzędzia do generacji różnego typu sygnałów –dostosowanych przez użytkownika lub zgodnych z określonymi standardami (włącznie z modulacjami OFDM, QAM i PSK).
82. Modele kanałów, w tym AWGN, Multipath Rayleigh Fading, Rician Fading, WINNER II, MIMO Multipath Fading oraz LTE MIMO Multipath Fading.
83. Podstawowe modele zniekształcenia sygnału RF, w tym nieliniowości, zakłócenia fazy, szum termiczny, a także algorytmy kompensacji zniekształceń.
84. Możliwość wykorzystania GPU do algorytmów wymagających dużych nakładów obliczeniowych, takich jak Turbo, LDPC czy dekodery Viterbiego.
85. Wsparcie dla stałoprzecinkowego modelowania oraz generacji kodu C/C++ i HDL.
86. Wraz z instrumentami RF lub pakietami wsparcia sprzętowego dla danych platform – możliwość podłączenia modeli systemów do urządzeń radiowych i weryfikacja projektów poprzez generację i rejestrację rzeczywistych sygnałów.
87. Możliwość projektowania i testowania systemów komunikacji Bluetooth.
88. Możliwość kosymulacji warstw PHY i MAC.
89. Funkcje i aplikacje do projektowania oraz testowania algorytmów analizy i rozpoznawania obrazów, obrazów przestrzennych 3D oraz systemów przetwarzania wideo.
90. Algorytmy wykrywania obiektów, w tym algorytmy Viola-Jones, ACF i inne.
91. Śledzenie obiektów przy wykorzystaniu m.in. algorytmu Kanade-Lucas-Tomasi (KLT) i filtru Kalmana.
92. Funkcje wykrywania, ekstrakcji cech oraz dopasowania obrazów, w tym detektory: FAST, BRISK, MSER i HOG.
93. Kalibracja pojedynczej lub kilku kamer, w tym automatyczne wykrywanie wzorca szachownicy i aplikacje do automatyzacji pracy.
94. Stereowizja, w tym obliczenia dysparycji, rekonstrukcja 3D i rektyfikacja.
95. Wsparcie generacji kodu C, wraz z arytmetyką stałoprzecinkową, a także generacji kodu CUDA.
96. Rozpoznawanie tekstu.
97. Przetwarzanie wideo, adnotacje obiektów, wyświetlanie wideo, nakładki graficzne i tworzenie kompozycji.
98. Rozpoznawanie obrazów oraz wykrywanie obiektów na obrazach z wykorzystaniem głębokich sieci neuronowych (*deep learning*), m.in. konwolucyjne sieci neuronowe, sieci R-CNN (wymagany moduł Deep Learning Toolbox).
99. Interfejs do biblioteki OpenCV.
100. Przetwarzanie i analiza danych w postaci chmury punktów oraz danych z lidar 3D.
101. Reprezentacja systemów liniowych za pomocą transmitancji, równań stanu, postaci wzmocnienia zer i biegunów (*zero-pole-gain*) oraz odpowiedzi częstotliwościowej.

- 102.Odpowiedź skokowa, charakterystyka Nyquista, a także inne narzędzia do badania stabilności i wydajności w dziedzinie czasu i częstotliwości.
- 103.Metoda root locus, charakterystyka Bodego, LQR, LQG, a także inne techniki projektowania systemów: klasyczne oraz wykorzystujące równania stanu.
- 104.Automatyczne strojenie regulatorów PID oraz harmonogramowanie wzmocnienia.
- 105.Konwersja reprezentacji modeli, dyskretyzacja modeli czasu ciągłego, aproksymacja niskiego rzędu dla systemów wysokiego rzędu.
- 106.Algorytmy estymacji stanu (filtracja Kalmana) oraz narzędzia do projektowania regulatorów LQR/LQG.
- 107.Współpraca z wybranymi narzędziami do identyfikacji systemów w celu modelowania obiektów za pomocą danych eksperymentalnych.
- 108.Interaktywny interfejs graficzny użytkownika, unifikujący podstawowe zadania dopasowywania krzywych i płaszczyzn.
- 109.Liniowe i nieliniowe równania regresji z możliwością użycia równań niestandardowych.
- 110.Biblioteka modeli regresji ze zoptymalizowanymi punktami startowymi i parametrami solvera.
- 111.Dopasowywanie nieparametryczne, za pomocą interpolacji, funkcji sklepanych i średnich ruchomych.
- 112.Procedury wstępnego przetwarzania danych: skalowanie danych, podział na podgrupy, wygładzanie, usuwanie błędnych punktów.
- 113.Procedury po przetworzeniu: interpolacja, ekstrapolacja, estymacja przedziałów ufności, obliczanie całek i pochodnych.
- 114.Funkcje i aplikacje do obsługi przemysłowych urządzeń do akwizycji danych.
- 115.Wsparcie dla urządzeń DAQ wykorzystujących połączenie USB, PCI, PCI Express, PXI, PXI Express od różnych producentów, w tym National Instruments.
- 116.Obsługa analogowych wejść i wyjść, liczników, timerów oraz cyfrowych wejść i wyjść.
- 117.Bezpośredni dostęp do parametrów sprzętowych urządzeń pomiarowych, takich jak pomiar jedno- i wielokanałowy, pojedynczy pomiar punktowy i pomiary buforowane.
- 118.Automatyczne uruchamianie procedur poprzez zdarzenia sprzętowe i programowe.
- 119.Możliwość tworzenia własnych interfejsów dla nieobsługiwanego sprzętu.
- 120.Aplikacje graficzne do konfiguracji ustawień akwizycji i generacji danych.
- 121.Możliwość interaktywnej pracy na bazach danych bez znajomości języka SQL z wykorzystaniem aplikacji Database Explorer.
- 122.Połączenia z bazami danych poprzez interfejs JDBC.
- 123.Połączenia z bazami danych poprzez interfejs ODBC wraz z opcją szybkiego dostępu przez natywny sterownik ODBC.
- 124.Funkcje pozwalające na wykonywanie skomplikowanych zapytań do baz danych z wykorzystaniem plików i poleceń SQL.
- 125.Import z oraz eksport do wielu baz danych w czasie jednej sesji.
- 126.Pobieranie dużych bloków danych w jednej transakcji lub w wielu transakcjach z automatycznym podziałem danych na mniejsze części.
- 127.Wsparcie dla nierelacyjnych baz danych, takich jak Cassandra, MongoDB i Neo4j.
- 128.Natywne interfejsy dla baz MySQL[®] i PostgreSQL[®].
- 129.Dostęp do danych rynkowych bieżących, intraday, historycznych i czasu rzeczywistego.

130. Możliwość zdefiniowania rynku, okresu oraz rodzaju danych do pobrania (cena otwarcia, cena zamknięcia, wolumen, itd.).
131. Możliwość pobierania danych tickowych intraday jako szereg czasowy.
132. Obsługa połączeń z Bloomberg desktop, B-PIPE oraz server.
133. Wykonywanie zleceń na platformach Bloomberg EMSX, Wind Data Feed Services, Wind Financial Terminal.
134. Obsługa połączeń z RMDS, Datastream® Web Services, NewsScope, oraz Reuters Tick History.
135. Wsparcie dla pobierania danych ekonomicznych z Haver Analytics oraz Federal Reserve Economic Data (FRED).
136. Modelowanie wpływu dużych transakcji na ceny aktywów oraz generowanie krzywych kosztowych z wykorzystaniem modeli Kissell Research Group.
137. Szacowanie kosztów transakcyjnych, miar wrażliwości oraz analizy potransakcyjne.
138. Dostęp do danych bieżących, intraday i czasu rzeczywistego dla instrumentów będących przedmiotem handlu.
139. Filtrowanie danych według instrumentów oraz giełd.
140. Definiowalne typy zleceń oraz instrukcje ich wykonania.
141. Wycena instrumentów, wykonywanie zleceń oraz pobieranie cen historycznych na platformach Trading Technologies X_TRADER, CQG Integrated Client, IB Trader Workstationsm.
142. Dostęp do systemów transakcyjnych za pomocą silnika FIX FlyerTM.
143. Projektowanie i symulacja aplikacji DDS (Data Distribution Service).
144. Aplikacje i bloki do modelowania i symulacji aplikacji, które publikują lub subskrybują oprogramowanie pośredniczące DDS.
145. Biblioteka DDS do zarządzania, tworzenia i edytowania definicji DDS w Simulinku.
146. Import specyfikacji DDS jako pliki IDL lub XML, w celu utworzenia szkieletowego modelu Simulinka jako punkt wyjścia do opracowywania algorytmów dla aplikacji DDS.
147. Bloki do publikowania i subskrybowania próbek do DDS, wraz z odpowiadającym im Quality of Service (QoS). W pełni integrowalne się ze stosami RTI Connnext DDS i eProsima DDS.
148. Generacja kodu C ++ i plików XML z modeli Simulinka (wraz z licencją na moduł Embedded Coder).
149. Funkcje i narzędzia do prototypowania oraz implementacji głębokich sieci neuronowych na układach FPGA i SoC.
150. Gotowe bitstreamy do celów uruchamiania różnego typu głębokich sieci neuronowych na wspieranych układach FPGA i SoC firm Intel oraz Xilinx.
151. Narzędzia do profilowania i estymacji umożliwiające dostosowanie parametrów wykorzystywanej głębokiej sieci neuronowej.
152. Możliwość dostosowywania sprzętowej implementacji głębokiej sieci neuronowej oraz generacji przenośnego, syntezywalnego kodu Verilog/VHDL do celów zaprogramowania wybranego układu FPGA (do tego celu potrzebna licencja na moduły Simulink i HDL Coder).
153. Płytkie sieci neuronowe (ang. *shallow networks*):

- a. sieci nadzorowane (uczenie z nauczycielem), w tym wielowarstwowe, z funkcjami o symetrii kołowej (radialne), LVQ, z opóźnieniem czasowym, NARX, LRN.
 - b. Sieci bez nadzoru, w tym mapy samoorganizujące i warstwy neuronów współzawodniczących.
154. Uczenie głębokie, w tym konwolucyjne sieci neuronowe (CNN), sieci DAG oraz LSTM, autoenkodery i sieci o architekturze typu GAN.
 155. Graficzny interfejs użytkownika do tworzenia, uczenia i symulowania sieci neuronowych.
 156. Wsparcie dla obliczeń równoległych oraz z wykorzystaniem GPU w celu przyspieszenia uczenia się (z wykorzystaniem modułu Parallel Computing Toolbox).
 157. Zwiększanie efektywności uczenia się dzięki funkcjom przetwarzającym dane przed i po uczeniu sieci.
 158. Modularna reprezentacja sieci, pozwalająca na zadawanie dowolnej liczby warstw wejściowych i dowolnej liczby połączeń między warstwami.
 159. Zbiór bloków Simulinka do budowania i oceny płytkich sieci neuronowych wraz z dokumentacją i demonstracyjnymi aplikacjami systemów sterowania.
 160. Eksportowanie wytrenowanych modeli do Simulinka
 161. Importowanie gotowych, wytrenowanych modeli sieci głębokich (AlexNet, VGG-16, VGG-19, Caffe Model Zoo).
 162. Biblioteka bloków do implementacji algorytmów przetwarzania systemów DSP na fizycznych urządzeniach.
 163. Wysokowydajne algorytmy przetwarzające dane z częstotliwościami wyrażonymi w GS/s
 164. Eksploracja rozwiązań projektowych i porównywanie wartości takich jak moc, wydajność, zużycie zasobów dla różnych wyborów architektonicznych.
 165. Modelowanie, symulacja i wdrażanie rzeczywistych systemów do zastosowań takich jak radary czy komunikacja bezprzewodowa na docelowych platformach FPGA i SoC.
 166. Prototypowanie zweryfikowanych algorytmów na FPGA, ASIC i SoC przy użyciu HDL Coder.
 167. Modelowanie systemów DSP czasu rzeczywistego w aplikacjach komunikacji, radarów, audio, urządzeń medycznych, IoT i innych.
 168. Wbudowane metody do projektowania filtrów, w tym zaawansowanych filtrów wielostopniowych, filtrów typu *multirate* i filtrów adaptacyjnych.
 169. Szybka transformata Fouriera, estymacja widmowa, wykorzystanie okien, statystyki sygnałów i algebra liniowa.
 170. Graficzne analizatory widma sygnałów oraz dynamiczne wykresy sygnałów w domenie czasu.
 171. Algorytmy wspierające zmiennoprzecinkowe, stałoprzecinkowe i całkowite typy danych.
 172. Wsparcie dla modelowania stałoprzecinkowego oraz generacji kodu C/C++ i HDL.
 173. Modele jednoczynnikowe ARMAX/GARCH, w tym EGARCH, GJR i inne warianty, modele regresyjne, ARIMA.
 174. Wieloczynnikowa symulacja i prognozowanie VAR, VEC i modele kointegracyjne.
 175. Modele przestrzeni stanów i filtry Kalmana do estymacji parametrów.

176. Testy pierwiastka jednostkowego (Dickeya-Fullera, Phillipsa-Perrona) i stacjonarności (Leybourne'a-McCabe'a, KPSS).
177. Diagnostyka pre- i postestymacyjna oraz testy statystyczne, w tym test wskaźnika prawdopodobieństw, test LM, test Walda, test Engle'a efektu ARCH i test Q Ljung-Boxa.
178. Testy kointegracji, w tym Engle'a-Grangera i Johansena.
179. Narzędzia diagnostyczne i wspomagające m.in. wybór modelu z wykorzystaniem kryteriów informacyjnych AIC/BIC oraz autokorelację, korelację krzyżową i częściową.
180. Filtr Hodricka-Prescotta do analizy cyklu koniunkturalnego.
181. Analiza struktury i ewolucji modeli Markova używających dyskretnych łańcuchów Markova.
182. Generacja czytelnego i kompaktowego kodu ANSI/ISO C/C++ optymalnego do zastosowań w urządzeniach wbudowanych na podstawie schematu blokowego algorytmu.
183. Możliwość optymalizacji generowanego kodu pod względem wydajności wykonywania lub ustawień dotyczących wybranej platformy docelowej.
184. Wsparcie dla standardów AUTOSAR, MISRA C i ASAP2.
185. Tworzenie dokumentacji generowanego kodu wraz z *traceability* oraz integracja z narzędziami do weryfikacji zgodności z normami DO-178, IEC 61508 i ISO26262.
186. Wsparcie dla typów danych zmiennoprzecinkowych oraz stałoprzecinkowych.
187. Umożliwienie przeprowadzania testów typu software-in-the-loop (SIL) oraz processor-in-the-loop (PIL).
188. Możliwość generacji kodu działającego w trybie jedno- i wielowątkowym lub asynchronicznym.
189. Możliwość ręcznego edytowania wygenerowanego kodu i integracji w zewnętrznych środowiskach programistycznych.
190. Wsparcie dla dyktów SIMD do wykonywania obliczeń z użyciem instrukcji SSE, AVX256/512 dla kodu przeznaczonego na platformy Intel oraz ARM NEON
191. Generacja przenośnego i syntezywalnego kodu VHDL i Verilog do implementacji filtrów stałoprzecinkowych na układach ASIC i FPGA.
192. Wsparcie generacji kodu z filtrów o różnej strukturze, w tym filtrów FIR, IIR (sekcje drugiego rzędu), typu *multirate* oraz *fractional delay*.
193. Kontrola zawartości, optymalizacji i stylu generowanego kodu.
194. Generacja testów benchów w celu szybkiej weryfikacji i walidacji automatycznie wygenerowanego kodu na drodze symulacji w Simulinku.
195. Możliwość przeprowadzenia kosymulacji modelu filtru w Simulinku z wygenerowanym kodem uruchomionym w syntezytorach firm trzecich, takim jak Cadence® Incisive® i Xcelium™ lub Mentor® ModelSim® i Questa® (wymagany moduł HDL Verifier).
196. Dopasowywanie krzywych dochodowości z wykorzystaniem bootstrappingu oraz metod parametrycznych, dodatkowo możliwość analizowania terminowej struktury stóp procentowych.
197. Modele wyceny Blacka- Scholesa, Blacka, Garmana-Kohlhagena, Rolla-Geske'go-Whaleya, Bjerksunda-Stenslanda, Nengjiu Ju, Stulza, Longstaffa-Schwartz, SABR, wykorzystujące drzewa dwu- i trójmianowe oraz wykorzystujące symulację Monte Carlo.
198. Możliwość szacowania cen, dochodowości, stopy dyskontu, harmonogramu przepływów pieniężnych, spreadów, zmienności implikowanej, OAS oraz

- współczynników (tzw. współczynników greckich). ryzyka
199. Ryzyko kredytowe kontrpartnera, modelowanie CVA oraz wsparcie dla instrumentów kredytowych, w tym dla instrumentów hipotecznych (pule kredytowe oraz kredyty balonowe) i swapów na zwłokę w spłacie kredytu (CDS).
 200. Wsparcie dla instrumentów stopy procentowej, w tym: obligacji, kontraktów futures, opcji klasycznych, opcji bermudzkich, obligacji z wbudowanymi opcjami, swapów, swapów opóźnionych, swapów amortyzowanych, swapcji, capów i floorów.
 201. Wsparcie dla instrumentów typu equity, w tym: akcji, opcji klasycznych, opcji bermudzkich, opcji azjatyckich, opcji wstecznych, opcji barierowych, opcji typu digital, opcji tęczyowych, opcji koszykowych, opcji na opcje oraz opcji wyboru.
 202. Wsparcie dla instrumentów pochodnych na energię i commodity, w tym: opcji azjatyckich, opcji bermudzkich, opcji wstecznych, opcji typu swing, opcji typu spread oraz opcji klasycznych o europejskim i amerykańskim stylu wykonania.
 203. Wsparcie dla wyceny różnych rodzajów instrumentów finansowych indywidualnie lub jako portfel, stosując nowy framework obiektowy.
 204. Narzędzia do optymalizacji portfela, w tym optymalizacja typu średnia-wariancja oraz CVaR.
 205. Analizy przepływów finansowych, analizy ryzyka, modelowanie finansowych szeregów czasowych, operacje na danych.
 206. Podstawowe analizy instrumentów o stałym dochodzie (zgodnie z SIA).
 207. Podstawowe modele wyceny opcji tj. Blacka-Scholesa, Blacka i model dwumianowy.
 208. Narzędzia wspomagające regresję oraz estymację z brakującymi danymi.
 209. Podstawowe narzędzia wspomagające estymację, symulacje oraz prognozowanie w modelach GARCH.
 210. Narzędzia analizy technicznej (wskaźniki oraz wykresy).
 211. Symulacja Batesa, Hestona, CIR za pomocą dyskretyzacji kwadratowo-wykładniczej.
 212. Modelowanie i symulacje procesów stochastycznych z wykorzystaniem stochastycznych równań różniczkowych.
 213. Tworzenie algorytmów stałoprzecinkowych i konwersja istniejących algorytmów zmiennoprzecinkowych na stałoprzecinkowe.
 214. Optymalizacja algorytmów w zależności od wymaganej dokładności numerycznej i ograniczeń docelowej platformy sprzętowej.
 215. Przyspieszone symulacje 'Bit-true' algorytmów stałoprzecinkowych.
 216. Testowanie i debugowanie efektów kwantyzacji, takich jak przepiętnienie bądź utrata precyzji.
 217. Automatyczna konwersja z typu zmiennoprzecinkowego na stałoprzecinkowy bądź zmiennoprzecinkowy o zredukowanej precyzji, w oparciu o analizę zakresu danych.
 218. Możliwość porównania wyników stało- i zmiennoprzecinkowych.
 219. Biblioteka w Simulinku bloków do efektywnych operacji matematycznych i macierzowych stałoprzecinkowych.
 220. Narzędzia do optymalizacji tabel wyszukiwań (*Lookup tables*).
 221. kodu C i HDL.
 222. Funkcje, aplikacje i bloki Simulinka do projektowania i symulacji systemów decyzyjnych opartych na logice rozmytej, możliwość graficznej edycji zmiennych wejściowych i wyjściowych, funkcji przynależności i reguł rozmytych.

223. Możliwość projektowania systemów rozmytych typu Mamdani i Sugeno oraz systemów złożonych.
224. Wsparcie dla systemów rozmytych typu 2 z funkcjami przynależności uwzględniającymi niepewność.
225. Wsparcie dla narzędzi pozwalających na dostrajanie algorytmów rozmytych metodami optymalizacyjnymi jak np. algorytmy genetyczne.
226. Możliwość tworzenia algorytmów adaptacyjnych w postaci neuro-rozmytych.
227. Wsparcie dla narzędzi do generacji kodu C/C++ oraz kodu Structured Text.
228. Interaktywne narzędzia pozwalające na definicję i rozwiązanie problemów optymalizacyjnych oraz na monitorowanie postępu prac z tym związanych.
229. Solvery globalnego wyszukiwania oraz typu multistart do znajdowania jednego lub wielu optimum globalnych.
230. Narzędzia do stosowania algorytmów genetycznych (w tym wielokryterialnych), wspierających liniowe i nieliniowe ograniczenia.
231. Metoda numeryczna „pattern search” dla ograniczeń liniowych, nieliniowych oraz brzegowych.
232. Symulowane wyżarzanie z zaimplementowaną metodą losowego wyszukiwania, wbudowane narzędzia dające możliwość definiowania procesu wygrzewania, temperatury i kryteriów akceptacji.
233. Wsparcie dla obliczeń równoległych.
234. Generacja zoptymalizowanego kodu CUDA z wykorzystaniem wywołania bibliotek NVIDIA® CUDA, takich jak TensorRT, cuDNN, cuFFT, cuSolver i cuBLAS.
235. Prototypowanie kodu CUDA do zastosowań w uczeniu głębokim i przetwarzaniu obrazów na procesorach graficznych NVIDIA Tesla® oraz NVIDIA Tegra®.
236. Możliwość integracji w zewnętrznych projektach wygenerowanego kodu CUDA jako kodu źródłowego lub bibliotek statycznych bądź dynamicznych i jego kompilacji na desktopach, serwerach oraz procesorach GPU wbudowanych na platformy NVIDIA Jetson, NVIDIA DRIVE i inne.
237. Integracja kodu CUDA napisanego ręcznie z kodem wygenerowanym w sposób automatyczny.
238. Generacja kodu CUDA z modeli Simulinka na procesory graficzne NVIDIA.
239. Generacja kodu CUDA z głębokich sieci neuronowych opracowywanych w Simulinku, celem programowania procesorów graficznych NVIDIA.
240. Możliwość weryfikacji numerycznej wygenerowanego kodu CUDA za pomocą testów software-in-the-loop i processor-in-the-loop (do tego celu wymagana licencja na Embedded Coder).
241. Generacja niezależnego od platformy docelowej, syntezywalnego kodu VHDL i Verilog na układy FPGA, ASIC i SoC.
242. Implementacja maszyn skończenie stanowych Moore’a i Melay’ego oraz logiki sterującej z użyciem Stateflow.
243. Przewodnik w procesie programowania aplikacji dla płyt rozwojowych Xilinx®, Intel® i Microsemi®.
244. Kontrola architektury i implementacji kodu, estymacja ścieżki krytycznej.
245. Współdzielenie zasobów oraz retiming w celu spełnienia wymagań szybkości i zajętości.
246. Możliwość natywnej obsługi zmiennorzecinkowych typów danych.

247. Możliwość dwukierunkowego śledzenia pomiędzy modelem a wygenerowanym kodem, wymaganego przez standardy bezpieczeństwa funkcjonalnego takie jak DO-254, ISO 26262 i IEC 61508.
248. Możliwość integracji kodu napisanego ręcznie.
249. Połączenie z oprogramowaniem Cadence® Incisive® i Xcelium™, Mentor Graphics® ModelSim® i Questa® oraz Synopsys® VCS®.
250. Weryfikacja FPGA-in-the-loop na płytkach rozwojowych Xilinx®, Microsemi® i Intel®.
251. Generacja test benchy UVM z modeli Simulinka.
252. Generacja modeli IEEE® 1666 SystemC TLM 2.0 kompatybilnych na poziomie transakcji.
253. Automatyzacja procesu weryfikacji kodu HDL dzięki wsparciu narzędzia HDL Coder.
254. Możliwość integracji zewnętrznego kodu HDL.
255. Wsparcie dla standardów przemysłowych, w tym DCAM, GenICam GenTL, Camera Link i GigE Vision.
256. Wsparcie dla popularnych interfejsów operacyjnych dla kamer, w tym Direct Show QuickTime oraz video4linux2.
257. Wsparcie dla kamer 3D, w tym kamery Kinect for Windows.
258. Wsparcie dla kamer Hamamatsu oraz FLIR Spinnaker.
259. Wiele trybów akwizycji danych i opcji zarządzania buforem.
260. Synchronizacja z urządzeniami do zbierania danych multimodalnych wyzwalanych sprzętowo.
261. Aplikacja Image Acquisition Tool do szybkiej konfiguracji sprzętu, rejestracji obrazu i podglądu wideo na żywo.
262. kodu C w Simulinku.
263. Kompleksowy zestaw wzorcowych algorytmów i aplikacji do przetwarzania, analizy, wizualizacji i rozpoznawania obrazów.
264. Algorytmy przetwarzania obrazów: filtry liniowe i nieliniowe, wyostżanie, detekcja krawędzi, poprawa jakości. Operacje blokowe (ROI) i kontekstowe.
265. Algorytmy analizy obrazu, w tym segmentacji, operacji morfologicznych, wyznaczania histogramów, statystyk i innych pomiarów.
266. Przetwarzanie obrazów medycznych, dostęp do danych w formacie DICOM.
267. Przetwarzanie danych wolumetrycznych 3D.
268. Transformacje geometryczne, metody dopasowania obrazów oparte o cechy charakterystyczne.
269. Transformacje obrazu, w tym FFT, DCT, Radon i typu fan-beam.
270. Metody do blokowego przetwarzania dużych obrazów.
271. Przetwarzanie wsadowe dużych zbiorów danych obrazowych.
272. Narzędzia do wizualizacji, w tym Image Viewer i Video Viewer.
273. Wykorzystanie głębokich sieci neuronowych (*deep learning*) do przetwarzania obrazów (wymagany moduł Deep Learning Toolbox)
274. kodu C/C++ z wielu dostępnych funkcji. Możliwość wykorzystania GPU do przyspieszania obliczeń.
275. Pozwala odczytywać, zapisywać i rejestrować dane OPC Unified Architecture (UA) z urządzeń, takich jak rozproszone systemy sterowania, systemy nadzoru i akwizycji danych oraz programowalne sterowniki logiczne.
276. Możliwe jest również uzyskanie dostępu do danych maszynowych i produkcyjnych bezpośrednio z serwerów OSIsoft® PI i wykorzystanie tych danych do monitorowania oraz doskonalenia procesów i aplikacji predykcyjnego utrzymania ruchu.

277. Pozwala na pracę z danymi z aktywnych serwerów i archiwów danych, które są zgodne ze standardami OPC UA, OPC Data Access (DA) i OPC Classic Historical Data Access (HDA). Podczas komunikacji z serwerami OPC UA bezpieczne połączenie zapewnione jest poprzez użycie rozmaitych trybów bezpieczeństwa, algorytmów szyfrowania i metod uwierzytelniania użytkowników.
278. Wsparcie dla sterowników IVI i VXI *plug&play*.
279. Obsługa protokołów GPIB i VISA (GPIB, GPIB-VXI, VXI, USB, TCP/IP, serial).
280. Wsparcie dla protokołów szeregowych TCP/IP, UDP, I2C oraz Bluetooth do zdalnej komunikacji z innymi komputerami i płytkami drukowanymi PCB.
281. Wsparcie dla protokołu MODBUS, umożliwiające komunikację z przemysłowym sprzętem automatyzującym, takim jak sterowniki PLC i PAC.
282. Funkcje pozwalające na zapis i odczyt danych binarnych oraz ASCII z i do instrumentów pomiarowych.
283. Graficzny interfejs użytkownika dla identyfikacji urządzeń, konfiguracji i komunikacji.
284. Trenowanie modeli w celu przeprowadzenia detekcji, segmentacji semantycznej i klasyfikacji przy pomocy algorytmów uczenia maszynowego i głębokiego takich jak PointPillars, SqueezeSegV2 i PointNet++.
285. Interaktywne aplikacje, w tym narzędzie Lidar Viewer pozwalające na wizualizację, analizę i wstępne przetwarzanie danych lidarowych oraz narzędzie Lidar Labeler do etykietowania obiektów na obrazach z czujników LiDAR z możliwością użycia wbudowanych lub własnych algorytmów do automatycznego oznaczania.
286. Wbudowane algorytmy do przeprowadzania wzajemnej kalibracji między czujnikami LIDAR oraz kamerami oraz funkcje do fuzji danych z czujników wizyjnych oraz LIDAR.
287. Tworzenie dwu- i trójwymiarowych map otoczenia oraz implementacja algorytmów SLAM.
288. Strumieniowanie chmur punktów LIDAR dla czujników firmy Velodyne.
289. Możliwość importowania danych LIDAR z zewnętrznych plików np. PCAP, LAS, ibeo, PCD, PLY oraz eksportu do plików PLY i PCD.
290. Funkcje i aplikacje do projektowania, symulacji i weryfikacji systemów komunikacyjnych LTE, LTE-Advanced oraz LTE-Advanced Pro.
291. Generacja sygnałów zgodnych ze standardami LTE, konfiguracja sygnałów uplink, downlink, informacji sterującej oraz kanałów.
292. Symulacja na poziomie łącza i funkcje przetwarzania odbieranych danych.
293. Modele testowe (E-TM) i referencyjny kanał pomiarowy (RMC) dla generatorów fali LTE, LTE-A, UMTS.
294. Interaktywne narzędzia dla testów zgodności i analizy bitowej stopy błędów (BER).
295. Możliwość pomiarów uplink i downlink, w tym EVM i ACLR.
296. Odzyskiwanie parametrów z przechwyconych sygnałów, w tym Cell Identifier, MIB oraz SIB1.
297. Estymacja kanału, synchronizacja oraz modelowanie odbiorników z wielodostępem (MIMO).
298. Modelowanie technologii radiowych NB-IoT oraz LTE-M (wersja 13 i 14).
299. Import i eksport danych wektorowych i rastrowych.
300. Szerokie spektrum obsługiwanych formatów plików GIS i geoprzestrzennych oraz serwerów map webowych.
301. Pobieranie niestandardowych map rastrowych z serwerów Web Map Service (WMS).
302. Wyświetlanie map z OpenStreetMap i innych źródeł.

303. Wyświetlanie interaktywnych map 2D i 3D z możliwością ich dostosowywania.
304. Narzędzia do transformacji układów współrzędnych.
305. Narzędzia do operacji na danych geograficznych, takich jak przycinanie, interpolacje, resampling i innych.
306. Wbudowane funkcje wspomagające analizę cyfrowych modeli terenu.
307. Funkcje geodezji geometrycznej, w tym wspomagające transformację 2D i 3D.
308. Ponad 60 rodzajów odwzorowań kartograficznych.
309. Generowanie przenośnego i czytelnego kodu C i C++ (zgodnego z ANSI/ISO) z kodu MATLABa.
310. Generowanie funkcji MEX (*MATLAB executable*).
311. Wbudowane narzędzia do zarządzania projektami pozwalające na określenie punktów wejścia, właściwości danych wejściowych oraz innych opcji związanych z generowanym kodem.
312. Statyczny lub dynamiczny przydział pamięci dla danych o zmiennej wielkości.
313. Wsparcie generacji kodu dla wielu większości funkcji w języku MATLAB oraz funkcji w modułach rozszerzających, m.in. Communications Toolbox, Computer Vision Toolbox, DSP System Toolbox, Deep Learning Toolbox oraz Phased Array System Toolbox.
314. Obsługa funkcji języka MATLAB, w tym działania na macierzach, indeksowania, instrukcji sterujących (*if, switch, for, while*), klas i struktur.
315. Współpraca z modułami Simulink Coder oraz Embedded Coder w zakresie generowania kodu C na podstawie modeli Simulinka, które zawierają kod MATLABa.
316. Współpraca z modułem Embedded Coder w zakresie dostosowania kodu, optymalizacji kodu pod docelową platformę, śledzenia zależności na linii kod MATLABa – kod C/C++ oraz weryfikacji software-in-the-loop (SIL) i processor-in-the-loop (PIL).
317. Generowanie kodu na systemy wieloprocesorowe w standardzie OpenMP.
318. Automatyczna konwersja własnej aplikacji stworzonej w środowisku MATLAB do samodzielnego pliku wykonywalnego lub aplikacji webowej.
319. Możliwość bezpłatnej dystrybucji aplikacji wśród użytkowników, którzy nie posiadają licencji na oprogramowanie MATLAB.
320. Integracja aplikacji z aplikacjami do *Big Data* – MapReduce i Spark™.
321. Możliwość hostowania aplikacji webowych, dostępnych dla użytkowników z poziomu przeglądarek, dzięki wersji deweloperskiej modułu MATLAB Web App Server, zawartej w module MATLAB Compiler.
322. Szyfrowanie kodu MATLAB w celu ochrony własności intelektualnej.
323. Możliwość bezpłatnej dystrybucji nakładek na Excela (*Excel add-ins*) na komputerach bez zainstalowanego MATLABa.
324. Automatyczna konwersja typów danych między Excelem i MATLABem.
325. Proste tworzenie makr VBA komunikujących się z MATLABem z poziomu Excela.
326. Rozszerzenie funkcjonalności modułu MATLAB Compiler o możliwość zbudowania komponentów takich jak współdzielone biblioteki C/C++, komponenty Microsoft® .NET, klasy języka Java i moduły języka Python z programów napisanych w MATLABie. Komponenty te mogą zostać zintegrowane z większymi aplikacjami, przeznaczonymi do wdrożenia w środowiskach desktopowych, webowych i *enterprise*.
327. Możliwość bezpłatnej dystrybucji klas Javy w środowiskach desktopowych i webowych.

328. Interfejs WebFigures umożliwiający zoomowanie, obracanie i przesuwanie wykresów tak jak w środowisku MATLAB.
329. Możliwość testowania i debuggowania kodu aplikacji oraz wtyczek Excela przed wdrożeniem ich w środowisku webowym i *enterprise*.
330. API dla automatycznej konwersji pomiędzy typami danych w Javie/Pythonie/.NET/COM i MATLABie.
331. Możliwość bezpłatnej dystrybucji, w środowiskach desktopowych oraz webowych, obiektów .NET i COM na komputerach bez zainstalowanego MATLABa.
332. Możliwość wywoływania komponentów z poziomu języków zgodnych z CLS (Common Language Specification), włączając w to: C#, F#, VB.NET oraz ASP.NET, a także zgodnych z technologią COM (Visual Basic®, ASP, Excel®).
333. Wsparcie Windows Communication Foundation (WCF) dla architektury zorientowanej na usługi (SOA) oraz architektury webowej.
334. .NET remoting do komunikacji między procesami.
335. Biblioteka przykładowych zadań z wielu dyscyplin naukowych z możliwością modyfikacji oraz opcją tworzenia własnych zadań, ćwiczeń, egzaminów.
336. Możliwość oceniania kształtującego i sumującego oraz skalowania ocen.
337. Automatyczna ocena rozwiązań, z opcją ważenia wyniku i informacją zwrotną dla studenta.
338. Statystyka wyników indywidualnych i zbiorczych, wraz z historią ich uzyskania.
339. Środowisko pracy oparte na przeglądarce internetowej.
340. Integracja z systemem LMS oraz platformą MOOC.
341. Wykonywanie funkcji wspierających obliczenia z wykorzystaniem GPU na klastrze obliczeniowym.
342. Szeregowanie i uruchamianie zadań za pomocą wbudowanego schedulera lub rozwiązań firm trzecich.
343. Uruchamianie oprogramowania stworzonego z wykorzystaniem języka MATLAB w środowisku produkcyjnym bez konieczności rekodowania bądź tworzenia specyficznej infrastruktury.
344. Skalowalna wydajność i zarządzanie pakietami aplikacji stworzonymi w MATLABie.
345. Lekka biblioteka kliencka pozwalająca na uruchamianie złożonych obliczeń.
346. Środowisko obliczeniowe dostępne za pośrednictwem interfejsu RESTful API lub bezpośrednio z .NET, Javy, C/C++ i Pythona.
347. Wbudowana baza danych REDIS do przechowywania stanu pomiędzy wywołaniami funkcji.
348. Możliwość komunikacji z innymi bazami danych wspieranymi przez moduł Database Toolbox.
349. Możliwość strumieniowania danych z urządzeń pomiarowych z użyciem platform takich jak Azure® IoT Hub, Azure Event Hub i Apache Kafka.
350. Hostowanie interaktywnych aplikacji webowych stworzonych w MATLABie i w Simulinku z pomocą narzędzia App Designer, oderwanych od oprogramowania z użyciem modułów MATLAB Compiler i Simulink Compiler.
351. Dostęp do aplikacji po stronie użytkowników końcowych z poziomu przeglądarek internetowych bez konieczności instalacji dodatkowego oprogramowania
352. Automatyczne zarządzanie wieloma aplikacjami webowymi w ramach jednego ekosystemu IT w danym przedsiębiorstwie.

353. Możliwość kontroli dostępu do aplikacji webowych dzięki integracji narzędzia ze standardami uwierzytelniania, takimi jak OpenID Connect i LDAP.
354. Kontrola zakresu dostępu do aplikacji webowych poprzez nadawanie uprawnień autora bądź użytkownika końcowego.
355. Liczba hostowanych i uruchamianych aplikacji webowych na serwerze ograniczona jedynie przez możliwości sprzętowe serwera.
356. Możliwość instalacji kilku instancji serwera w celu rozdzielenia dostępu do określonych aplikacji dla poszczególnych działów/zespołów w organizacji.
357. Zapewnia narzędzia, funkcje i środowisko do projektowania i testowania aplikacji do obrazowania diagnostycznego..
358. Pozwala wykonać renderowanie i wizualizację 3D, rejestrację multimodalną oraz segmentację i etykietowanie obrazów radiologicznych. Zestaw narzędzi umożliwia również trenowanie wstępnie zdefiniowanych sieci głębokich (z wykorzystaniem modułu Deep Learning Toolbox)
359. Możliwy jest import, wstępne przetwarzanie i analiza obrazów radiologicznych z różnych metod obrazowania, w tym obrazowania rentgenowskiego rzutowego, tomografii komputerowej (CT), rezonansu magnetycznego (MRI), ultradźwięków (US) i medycyny nuklearnej (PET, SPECT)..
360. Narzędzie *Medical Image Labeler* umożliwia półautomatyzację etykietowania 2D i 3D do użytku w cyklu projektowania aplikacji inteligencji maszynowej. Możliwe jest wykonanie multimodalnej rejestracji obrazów medycznych, w tym obrazów 2D, powierzchni i objętości 3D. Zestaw narzędzi zapewnia zintegrowane środowisko do kompleksowej, wspomaganej komputerowo diagnostyki i analizy obrazów medycznych.
361. Modelowanie, symulacja i weryfikacja systemów przetwarzających sygnały analogowe i mieszane, takich jak pętle PLL czy przetworniki ADC.
362. Możliwość skorzystania z gotowych części systemów oraz modyfikacji modeli poprzez uwzględnienie dodatkowych zakłóceń, takich jak szumy, nieliniowości i efekty kwantyzacji.
363. Możliwość wspólnej symulacji komponentów systemów mieszanych wraz z komponentami algorytmów DSP i logiki sterującej.
364. Gotowe scenariusze testowe do weryfikacji działania opracowywanych systemów.
365. Dedykowana aplikacja do analizy i wizualizacji sygnałów mieszanych oraz wyników symulacji pochodzących ze środowiska Cadence® Virtuoso®.
366. Projektowanie i symulacja regulatorów predykcyjnych.
367. Narzędzia wspierające dostosowanie ograniczeń i wag i innych parametrów projektowanego regulatora.
368. Możliwość dostosowywania ograniczeń i wag regulatorów oraz monitorowania sygnałów w trakcie działania skryptów MATLABa oraz modeli Simulinka.
369. Możliwość zmiany wewnętrznego modelu obiektu w czasie trwania symulacji w adaptacyjnych algorytmach sterowania typu MPC.
370. Kontrola modelu w szerokim zakresie warunków pracy, poprzez przełączanie się pomiędzy wieloma regulatorami predykcyjnymi.
371. Interaktywna aplikacja do projektowania regulatorów predykcyjnych.

372. Gotowe przykłady regulacji MPC w systemach ADAS.
373. kodu C oraz kodu Structured Text IEC 61131-3 do celów szybkiego prototypowania i implementacji na systemach wbudowanych.
374. kodu kontrolerów liniowych MPC do uruchamiania na GPU zarówno z poziomu MATLAB jak i Simulink
375. Interfejs użytkownika (aplikacja) do projektowania i kalibracji złożonych systemów nieliniowych używanych w przemyśle samochodowym (m.in. silników spalinowych, urządzeń elektrycznych, pomp oraz wentylatorów) bądź innych systemów jak silniki odrzutowe, napędy jednostek wodnych czy sprzęt wiertniczy.
376. Możliwość przeprowadzania eksperymentów i testów do kalibracji i strojenia systemów na podstawie modeli statystycznych.
377. Możliwość eksportu stworzonych skalibrowanych modeli do środowisk do symulacji systemów dynamicznych w dziedzinie czasu oraz eksportu tablic (*Lookup Tables* - LUT) kalibracji do standardów ETAS INCA oraz ATI Vision.
378. Projektowanie klasyczne, wypełniania przestrzeni oraz optymalne, oparte na metodzie projektowania eksperymentalnego do tworzenia zoptymalizowanych planów testowych.
379. Możliwość wykorzystania statystycznych technik tworzenia nieliniowych modeli o wysokiej wierności odwzorowania z danych testowych.
380. Możliwość modelowania regresji liniowej i podstawowych funkcji radialnych dla dokładnego dopasowania modelu do danych.
381. Możliwość modelowania warunków brzegowych w celu utrzymania wyników optymalizacji w przestrzeni definiowanej przez punkty pracy systemu.
382. Projektowanie algorytmów sterowania wektorowego (FOC) silników bezszczotkowych.
383. Programowa obsługa czujników Halla, enkoderów i resolverów.
384. Obliczenia transformat Parka i Clarka.
385. Generowanie sygnałów wektorowych składowych wypadkowych prądu.
386. Możliwość użycia przykładowych rozwiązań dotyczących algorytmów sterowania silnikami PMSM.
387. Zgodność z wytycznymi MISRA C
388. Wsparcie dla narzędzi automatycznie generujących kod.
389. Wsparcie dla narzędzi do automatycznego strojenia regulatorów.
390. Wsparcie dla narzędzi do identyfikacji obiektów i estymacji parametrycznej.
391. Algorytmy i narzędzia do projektowania systemów planowania ruchu i nawigacji.
392. Tworzenie reprezentacji map 2D i 3D z wykorzystaniem własnych danych lub generacji map przy użyciu algorytmów jednoczesnej lokalizacji i mapowania (SLAM).
393. Zastosowanie algorytmów SLAM w systemach z wieloma czujnikami oraz opartymi o punkty orientacyjne.
394. Generowanie metryk do porównywania testów optymalności, płynności i wydajności ścieżki.
395. Modele czujników i algorytmy do estymacji pozycji na bazie fuzji czujników.
396. Symulowanie systemów globalnej nawigacji satelitarnej – lokalizowanie satelitów i odbiorników GPS.
397. Aplikacja SLAM Map Builder umożliwiająca interaktywną wizualizację i debugowanie generowania wirtualnych map.
398. Wsparcie generacji

399. Dostęp do zestawu interaktywnych kursów,
400. Praca w przeglądarce internetowej, bez konieczności instalacji programu,
401. Możliwość generacji imiennego certyfikatu po pozytywnym ukończeniu kursu,
402. Możliwość samodzielnej organizacji pracy – czasu, tempa, liczby podejść itp.
403. Aplikacja w Live Editor z graficznym interfejsem użytkownika do definiowania (typ zadania, zmienne decyzyjne, funkcja celu, ograniczenia, parametry metody) i rozwiązywania problemów optymalizacji oraz monitorowania procesów z tym związanych.
404. Optymalizacja nieliniowa i wielokryterialna.
405. Solvery dla regresji nieliniowej metodą najmniejszych kwadratów, dopasowywania danych i równań nieliniowych.
406. Rozwiązywanie zadań programowania liniowego, także mieszanych (z ciągłymi i całkowitoliczbowymi zmiennymi decyzyjnymi) oraz zadań programowania kwadratowego.
407. Przyspieszenie działania solverów nieliniowych z ograniczeniami z wykorzystaniem modułu Parallel Computing Toolbox.
408. Generacja kodu C/C++ dla problemów programowania kwadratowego (quadprog) oraz nieliniowych i liniowych problemów optymalizacji z wykorzystaniem modułu MATLAB Coder.
409. Narzędzia do automatycznego różnicowania funkcji celu oraz ograniczeń, mające za zadanie uzyskanie dokładniejszych rozwiązań w krótszym czasie.
410. Możliwość rozwiązywania intensywnych obliczeniowo zadań z wykorzystaniem procesorów wielordzeniowych, kart GPU oraz klastrów komputerowych.
411. Zrównoleglone pętle for (parfor) do uruchamiania równoległych zadań na wielu procesorach.
412. Wsparcie dla procesorów graficznych NVIDIA z obsługą architektury CUDA.
413. Pełne wykorzystanie procesorów wielordzeniowych.
414. Interaktywne i wsadowe wykonywanie aplikacji.
415. Tablice rozproszone oraz równoległe wykonywanie identycznych podzadań dla różnych danych (przetwarzanie współbieżne) do obsługi dużych zestawów danych.
416. Metody numeryczne do rozwiązywania układów równań różniczkowych cząstkowych: statycznych, w dziedzinie czasu, w dziedzinie częstotliwości, wartości własnych.
417. Możliwość rozwiązywania problemów eliptycznych, parabolicznych i hiperbolicznych.
418. Specyfikacje warunków brzegowych: Dirichleta, uogólniona Neumanna i mieszane.
419. Funkcje pozwalające na tworzenie geometrii 2D oraz import geometrii 3D z plików STL.
420. Zautomatyzowane tworzenie siatki obliczeniowej z ziarnem trójkątnym lub czworobocianym.
421. Jednoczesna wizualizacja wielu własności rozwiązania, nakładanie siatki FEM i animacje.
422. Modelowanie wielofunkcyjnych szyków antenowych, zarówno aktywnych (AESAs), jak i pasywnych (PESAs).
423. Modele otoczenia radarów z uwzględnieniem targetów, ich parametrów i trajektorii, kanałów propagacji, jammerów i zakłóceń.
424. URA, ULA, UCA i konforemne szyki czujników z efektami perturbacyjnymi i polaryzacyjnymi.
425. Ciągłe i impulsowe przebiegi modulowane częstotliwościowo i kodowane fazowo.

426. Modelowanie i analiza polaryzacji fal elektromagnetycznych.
427. Cyfrowe algorytmy kształtowania wiązki dla szerokopasmowych i wąskopasmowych przebiegów.
428. Algorytmy estymacji kierunku propagacji (DOA) takie jak algorytmy *monopulse*, *beamscan*, MVDR, MUSIC, 2D MUSIC oraz root-MUSIC.
429. Generacja danych z radaru, sonaru i EW do celów analizy działania systemów oraz do trenowania algorytmów uczenia maszynowego.
430. Wizualizacja detekcji na wykresach zależności: zasięg-Doppler, zasięg-kąt, zasięg-czas-intensywność (RTI) i Doppler-czas-intensywność (DTI).
431. Algorytmy adaptacyjnego przetwarzania przestrzenno-czasowego (STAP).
432. Modelowanie kanałów MIMO z uwzględnieniem rozpraszaczy i warunków środowiskowych, takich jak deszcz, mgła i gazy.
433. Możliwość przyspieszania obliczeń poprzez wykorzystanie GPU oraz wsparcie generacji kodu C/C++, w tym MEX
434. Wykrywanie błędów czasu wykonania, problemów z przepływem danych i innych defektów w kodzie źródłowym C i C++.
435. Szybka analiza dużych baz kodu.
436. Weryfikacja zgodności ze standardami kodowania takimi jak MISRA-C, MISRA-C++, JSF++, CERT C, CERT C++, ISO/IEC TS 17961:2013 lub zdefiniowanymi konwencjami nazewnictwa.
437. Możliwość śledzenia (*traceability*) rezultatów weryfikacji kodu do bloków programu nadrzędnego.
438. Integracja ze środowiskiem Eclipse™ IDE.
439. Udowodnianie braku błędów czasu wykonania w kodzie C i C++.
440. Wykryte błędy czasu wykonania, oznaczone kolorem bezpośrednio w kodzie.
441. Identyfikacja zmiennych, które przekraczają określony limit.
442. Wyliczenie zakresów dla zmiennych oraz wartości zwracanych przez funkcje.
443. Metryki jakości kodu.
444. Wsparcie w procesie przeglądania kodu w celu klasyfikacji wyników i błędów czasu wykonania (*run-time errors*).
445. Graficzny interfejs do zapisu i odczytu zmiennych.
446. Kompletny przykładowe modele układów napędowych (benzynowe, diesla, hybrydowe, elektryczne) do symulacji tych systemów.
447. Biblioteki komponentów silników, skrzyń biegów, silników trakcyjnych i akumulatorów oraz podstawowe algorytmy sterowania do podzespołów napędowych.
448. Bazowanie na standardowych danych cyklu jazdy (FTP75, NEDC i JC08).
449. Możliwość przeprowadzania analizy kompromisów w kontekście np. emisji spalin, spalania czy sprawności układu napędowego oraz wsparcie dla wybranych narzędzi wspomagających optymalizację parametrów układu napędowego.
450. Wsparcie dla wybranych narzędzi do przeprowadzania testów typu hardware-in-the-loop (HIL).
451. Możliwość integracji z wybranymi narzędziami do modelowania fizycznego systemów rzeczywistych.
452. Wsparcie dla plików MDF do importu danych kalibracyjnych.
453. Modele przeżycia, podobieństwa i szeregi czasowe dla oceny pozostałego okresu użytkowania (RUL).

454. Metody ekstrakcji cech czasowych, częstotliwościowych i czasowo-częstotliwościowych do projektowania wskaźników stanu.
455. Organizacja danych z czujników importowanych z lokalnych plików, usług Amazon S3™, Windows Azure® Blob Storage i Hadoop® Distributed File System.
456. Organizacja symulowanych danych maszynowych z modeli Simulinka.
457. Przykłady opracowania algorytmów konserwacji predykcyjnej dla silników, skrzyń biegów, akumulatorów i innych maszyn.
458. Generacja kodu MATLABa z aplikacji do zautomatyzowania przetwarzania sygnałów, czy wyodrębniania cech (Diagnostic Feature Designer App)
459. Algorytmy i narzędzia do projektowania, symulacji, analizy i testowania wielofunkcyjnych systemów radarowych.
460. Projektowanie systemów radarowych stosowanych w samolotach, łodziach, samochodach i na lądzie.
461. Aplikacja Radar Designer do analizy kosztów i przykładowych rozwiązań na podstawie zadanych równań radaru.
462. Biblioteka modeli nadajników, odbiorników, zagłuszczy, targetów i kanałów propagacji.
463. Możliwość modelowania fizycznych i abstrakcyjnych sensorów samochodowych i symulacji danych dla różnych zjawisk takich jak efekt micro-Dopplera.
464. Generacja kodu C w celu wykonania obliczeń na rzeczywistych platformach.
465. Trenowanie za pomocą algorytmów DQN, DDPG, A2C i innych.
466. Parametryzacja strategii za pomocą głębokich sieci neuronowych, liniowych funkcji bazowych.
467. Automatyczne generowanie funkcji nagród na podstawie specyfikacji kontrolera
468. Import zasad ze środowiska Keras i formatu modelu ONNX (z modułem Deep Learning Toolbox).
469. Zrównoleglanie symulacji środowiska i obliczenia gradientu na procesorach graficznych i wielordzeniowych procesorach w celu trenowania
470. Aplikacja do interaktywnego tworzenia, trenowania i symulowania agentów uczenia ze wzmocnieniem.
471. Wdrażanie wytrenowanych modeli na urządzenia wbudowane poprzez automatyczną generację kodu dla procesorów CPU i GPU.
472. Zarządzanie, tworzenie oraz edycja wymagań projektu.
473. Możliwość tworzenia wymagań tekstowych o zadanych atrybutach i powiązania ich z komponentami modeli Simulinka, kodem bądź testami.
474. Podgląd modeli wraz z naniesioną na nie informacją o poszczególnych wymaganiach.
475. Śledzenie zmian między wymaganiami, projektem, generowanym kodem i testami.
476. Import wymagań opracowanych w zewnętrznych narzędziach takich jak IBM Rational DOORS, Microsoft Word i Microsoft Excel, a także import plików ReqIF.
477. Eksport wymagań do plików formatu ReqIF.
478. Metryki stanu implementacji i weryfikacji wymagań.
479. Wsparcie dla wybranych standardów przemysłowych dzięki współpracy z modułami IEC Certification Kit (dla ISO 26262 i IEC 61508) oraz DO Qualification Kit (dla DO-178).
480. Dedykowany interfejs do tworzenia i analizy macierzy śledzenia (Traceability Matrix) obrazujący powiązania między wymaganiami.

481. Biblioteka bloków Simulinka do projektowania systemów komunikacji RF i systemów radarowych.
482. Możliwość symulacji nieliniowych wzmacniaczy sygnałów w torze RF oraz modelowania efektów pamięci do celów estymacji wzmocnienia, szumów oraz zniekształceń intermodulacyjnych parzystych i nieparzystych rzędów.
483. Modelowanie niedopasowania impedancyjnego.
484. Parametryzacja modeli z użyciem danych z dokumentacji bądź danych uzyskanych w pomiarach.
485. Możliwość importu plików Touchstone® i wykorzystania parametrów rozpraszania do modelowania parametrów systemów.
486. Algorytmy adaptacyjne, takie jak automatyczna kontrola wzmocnienia (AGC), cyfrowej predystorcji (DPD) oraz kształtowania wiązki (ang. *beamforming*).
487. Modelowanie systemów RF na różnych poziomach abstrakcji, dzięki technikom *circuit envelope* i *equivalent baseband*.
488. Symulacja obwiedni układu dla wielu modeli częstotliwości nośnej.
489. Generacja jarzm testowych z aplikacji RF Budget Analyzer.
490. Komponenty pasywne, w tym elementy RLC, linie transmisyjne, filtry, przełączniki, złącza i ogólne bloki impedancji.
491. Ulepszone wysoce nieliniowe modele mikserów 3-portowych i wzmacniaczy 2-portowych określane przez współczynnik szumów, IP2, IP3 oraz pliki danych.
492. Tworzenie modeli z użyciem języka Simscape.
493. Funkcje i aplikacje do projektowania, analizy i wizualizacji sieci komponentów RF.
494. Filtry RF, linie transmisyjne, wzmacniacze i mieszacze częstotliwości określone przez dane pomiarowe, parametry sieci lub właściwości fizyczne.
495. Obliczanie parametrów rozpraszania dla komponentów sieci RF oraz możliwość konwersji między parametrami sieci S, Y, Z, ABCD, h, g i T.
496. Obliczanie opóźnienia fazowego i grupowego.
497. Aplikacja RF Budget Analyzer umożliwiająca wyznaczenie wartości współczynnika szumów, wzmocnienia i IP3 dla transceiverów RF oraz wygenerowanie jarzm testowych wykorzystywanych w module RF Blockset w symulacjach *circuit envelope*.
498. Metoda dopasowywania funkcji wymiernych do budowania modeli złącz *backplane* i eksportu ich jako bloki Simulinka, netlisty SPICE lub moduły Verilog-A.
499. Wizualizacja z wykorzystaniem wykresów kartezjańskich, biegunowych, a także wykresów Smitha.
500. Projektowanie płytek PCB z wbudowanymi filtry, transformatorami impedancji, sprzęgaczami, rozdzielaczami i ścieżkami działającymi na częstotliwościach RF.
501. Tworzenie różnych układów płytek PCB używając połączeń takich jak ścieżki, zagięcia, krzywe i ścięte krawędzie.
502. Analiza elektromagnetyczna projektu płytek – obliczanie S-parametrów, impedancji a także dystrybucji ładunków i natężenia prądu
503. Możliwość użycia różnych materiałów – dielektryków i metali – do projektowania płytek PCB. Wybór materiałów z katalogu lub o określonej grubości, przewodności, przenikalności i wartości tangensa strat.
504. Generacja plików Gerber gotowych do przekazania wytwórcom.
505. Analiza przesłuchów i optymalizacja ścieżek w celu minimalizacji szumu i sprzężeń.

506. Aplikacja Binning Explorer do tworzenia i modelowania kart scoringowych dla ryzyka kredytowego.
507. Narzędzia symulacyjne bazujące na kopułach dla portfeli instrumentów kredytowych.
508. Modele: binomialny, traffic light, Kupca, Christoffersena oraz Haasa do testowania jakości miary wartości zagrożonej (VaR) w przypadku ryzyka rynkowego.
509. Interaktywny edytor scen 3D wykorzystywanych w symulacji i testowaniu systemów automatycznej jazdy.
510. Tworzenie i dostosowywanie scen drogowych, w tym sieci dróg złożonej m.in. ze skrzyżowań, rond i mostów.
511. Kontrola zakrzywienia i nachylenia dróg.
512. Dostosowywanie nawierzchni dróg, oznaczeń na drodze oraz uszkodzeń nawierzchni.
513. Możliwość wstawiania komponentów z biblioteki RoadRunner Asset Library.
514. Konfiguracja czasowa sygnalizacji świetlnej umieszczonej w scenie 3D.
515. Konfiguracja ścieżki pojazdów na skrzyżowaniach dróg.
516. Wsparcie wizualizacji chmury punktów lidar, zdjęć lotniczych oraz danych GIS.
517. Import i eksport sieci dróg w formacie OpenDRIVE®.
518. Eksport scen 3D do plików w formatach FBX®, glTF™, OpenFlight, OpenSceneGraph, OBJ i USD.
519. Możliwość wykorzystania wyeksportowanych scen 3D w symulatorach automatycznej jazdy i w silnikach do gier takich jak CARLA, Vires VTD, NVIDIA DRIVE Sim®, Metamoto®, LGSVL, Baidu Apollo®, Unity® i Unreal® Engine.
520. Biblioteka komponentów 3D wykorzystywanych w tworzeniu scen 3D w narzędziu RoadRunner.
521. W bibliotece m. in. modele dróg, znaków drogowych pochodzących z różnych krajów, sygnalizacji świetlnej, oznaczeń na nawierzchni, drzew i krzewów, budynków, barier, uszkodzeń nawierzchni.
522. Możliwość dostosowywania komponentów, np. tworzenia własnych znaków drogowych.
523. Interaktywny edytor rozszerzający możliwości środowiska RoadRunner do tworzenia dynamicznych scenariuszy na potrzeby systemów autonomicznych w pojazdach.
524. Możliwość umieszczania pojazdów wraz z trajektorią i logiką ruchu na stworzonych wcześniej scenach.
525. Automatyzacja zadań takich jak generowanie, import i eksport scenariuszy z poziomu interfejsu programistycznego wspierającego język MATLABa, C++ czy Python.
526. Wbudowany odtwarzacz pozwalający na przegląd zaprojektowanego scenariusza.
527. Eksport do formatu ASAM OpenSCENARIO
528. Narzędzie do importu danych drogowych z usługi map wysokiej rozdzielczości HERE HD Live Map oraz automatycznej syntezy sieci dróg 3D z tych danych w środowisku RoadRunner.
529. Możliwość wizualizacji oraz edycji automatycznie zsyntezowanej sieci dróg, poprzez dodawanie komponentów takich jak ulice, roślinność, znaki drogowe i inne obiekty 3D.
530. Możliwość eksportu automatycznie zsyntezowanych sieci dróg do formatów OpenDrive®, FBX®, glTF™, OpenFlight, OpenSceneGraph, OBJ i USD.

531. Możliwość wykorzystania wyeksportowanych scen 3D w symulatorach automatycznej jazdy i w silnikach do gier takich jak CARLA, Vires VTD, NVIDIA DRIVE Sim®, Metamoto®, LGSVL, Baidu Apollo®, Unity® i Unreal® Engine.
532. Narzędzia i algorytmy do projektowania, symulacji i testowania manipulatorów, robotów mobilnych i robotów humanoidalnych.
533. Algorytmy sprawdzania kolizji, generowania trajektorii, kinematyki prostej i odwrotnej oraz dynamiki bryły sztywnej.
534. Algorytm mapowania, lokalizacji, planowania ścieżki, śledzenia ścieżki i sterowania ruchem.
535. Biblioteka modeli komercyjnych robotów przemysłowych.
536. Możliwość kosymulacji z symulatorem Gazebo.
537. Sprawdzanie kolizji, definiowanie kształtów kolizji i wykrywanie kolizji pomiędzy siatkowymi geometriami.
538. Możliwość importu plików URDF oraz modeli modułu Simscape Multibody do celów tworzenia własnych modeli robotów.
539. Wsparcie generacji kodu C/C++ do celów szybkiego prototypowania oraz testów hardware-in-the-loop
540. Funkcje i bloki do analizy i strojenia systemów sterownia pod kątem ich wydajności i niezawodności.
541. Tworzenie modeli z niepewnościami (niepewność parametrów, niezamodelowana dynamika).
542. Analizy najgorszych przypadków zapasu stabilności i wrażliwości na zakłócenia.
543. Generowanie kontrolerów ze zredukowaną czułością na wariację parametrów i błędy modelowania.
544. Automatyczne strojenie regulatorów SISO i MIMO, zdecentralizowane, o stałej strukturze oraz z wieloma pętlami sprzężenia zwrotnego.
545. Interfejs komunikacyjny dla narzędzia do obliczeń inżynierskich oraz symulacji dynamicznych systemów w domenie czasu z systemami ROS oraz ROS 2.
546. Możliwość obustronnej wymiany danych z systemami ROS oraz ROS 2.
547. Wsparcie dla wybranych narzędzi do automatycznej generacji kodu dla tworzenia aplikacji do sieci ROS (ROS Nodes).
548. Wsparcie dla funkcjonalności ROS jak: Publishers, Subscribers, Services, Actions, Parameter Server, rosbag i niestandardowe wiadomości (messages) ROS.
549. Wsparcie dla systemów ROS/ROS 2 działających pod różnymi systemami operacyjnymi, takimi jak Windows, Mac OS i Linux.
550. Generacja węzłów C++ ROS Nodes
551. Symulowanie, analizowanie oraz testowanie systemów i łącz komunikacji satelitarnej.
552. Modelowanie i wizualizacja orbit satelitów.
553. Analizowanie połączeń i obliczeń dostępu.
554. Projektowanie algorytmów warstwy fizycznej wraz z komponentami RF i odbiornikami stacji naziemnej.
555. Generowanie przebiegów testowych i przeprowadzanie weryfikacji typu „golden reference design”.

556. Weryfikowanie zgodności ze standardami komunikacji satelitarnej i nawigacji, DVB-S2X, DVB-S2, CCSDS i GPS.
557. Algorytmy fuzji czujników oraz śledzenia obiektów, w tym filtry fuzji czujników i *multi-object trackers*.
558. Generacja danych syntetycznych z modeli czujników, w tym czujników RF, akustycznych, EO/IR i GPS/IMU do celów testowania systemów fuzji sensorów i śledzenia obiektów.
559. Metryki i narzędzia wizualizacyjne do oceny dokładności i wydajności systemów fuzji sensorów i śledzenia obiektów, w tym metryki OSPA i GOSPA.
560. Przykłady referencyjne fuzji sensorów i śledzenia w systemach nadzoru lotniczego, naziemnego, podwodnego, nawigacji i w systemach autonomicznych.
561. Import i generowanie scenariuszy i trajektorii ruchu.
562. Wsparcie generacji kodu C do celu przyspieszenia symulacji i szybkiego prototypowania
563. Śledzenie obiektów z użyciem algorytmów RFS (Random Finite Set).
564. Narzędzia i aplikacje do projektowania i weryfikacji systemów serializera/deserializera
565. Analiza statystyczna z użyciem aplikacji SerDes Designer do celów szybkiego projektowania nadajników i odbiorników w systemach komunikacji przewodowej.
566. Modele typu *white-box*, takie jak DFE, CTLE, FFE i CDR do opracowywania adaptacyjnych equalizerów.
567. Automatyczna generacja modeli dualnych IBIS-AMI do wykorzystania w symulatorach kanałów dostarczanych przez firmy trzecie, takich jak SiSoft QCD i QSI, Keysight™ ADS, Synopsys® HSPICE, Mentor Graphics® HyperLynx® czy Cadence® Sigrity SystemSI.
568. Gotowe przykłady typu *white-box* modelowania standardowych protokołów komunikacyjnych, takich jak PCI Express, DDR i Ethernet.
569. Możliwość analizy jitteru IBIS-AMI.
570. Funkcje i aplikacje do analizy, preprocessingu i ekstrakcji cech z jednostajnie i niejednostajnie próbkowanych sygnałów.
571. Transformaty sygnałów, m.in. szybka transformata Fouriera (FFT), krótkookresowa transformata Fouriera (STFT), transformata Hilberta.
572. Metody projektowania filtrów FIR i IIR, ich analiza i implementacja.
573. Możliwość projektowania filtrów analogowych: Butterwortha, Czebyszewa, Bessla i eliptycznych oraz ich konwersji na postać cyfrową za pomocą metod transformacji biliniowej i niezmienności odpowiedzi impulsowej.
574. Aplikacja Filter Designer do interaktywnego projektowania i analizy filtrów o określonych charakterystykach.
575. Funkcje do generacji sygnałów takich jak sinus, prostokąt, piła, delta Kroneckera.
576. Pomiary i analizy statystyczne sygnałów.
577. Algorytmy estymacji widmowej gęstości mocy, m.in. periodogram, funkcje Welch, Burga, Yule-Walkera.
578. Pomiary widma mocy sygnału i parametrów takich jak SNR, THD i SINAD.
579. Narzędzia do analizy okien czasowych.
580. Modelowanie parametryczne i predykcyjne systemów liniowych.
581. Narzędzia do etykietowania fragmentów sygnałów do celów trenowania i walidacji modeli uczenia maszynowego.

582. Możliwość przeprowadzenia analizy modalnej oraz analizy rzędu sygnałów wibracyjnych.
583. Wsparcie generacji kodu C/C++ oraz zoptymalizowanego kodu CUDA.
584. Funkcje do projektowania szybkich łączy szeregowych i równoległych, przeprowadzania eksperymentów uwzględniających zróżnicowane parametry, wyodrębniania metryk projektu oraz wizualizacji przebiegów i rezultatów.
585. Aplikacje Serial Link Designer i Parallel Link Designer do kompleksowej analizy łączy z wykorzystaniem modeli IBIS-AMI, płytek PCB, ścieżek i konektorów.
586. Aplikacja Signal Integrity Viewer do wizualizacji rezultatów z projektów wykonanych w wymienionych aplikacjach pod kątem integralności sygnału.
587. Przebiegi i wykresy oczkowe oraz pomiar jakości kanału na podstawie zjawisk takich jak interferencja międzysymbolowa, jitter i szum.
588. Możliwość oszacowania kompromisów i zoptymalizowania łączy szeregowych i równoległych pod kątem kosztów, wydajności i niezawodności.
589. Wykonywanie poprojektowej weryfikacji systemu i skorelowanie wyników symulacji z otrzymanymi danymi pomiarowymi dla projektów PCB.
590. Weryfikacja zgodności ze standardami przemysłowymi takimi jak IEEE 802.3, OIF, PCIe i DDR.
591. Modelowanie PK/PD (*pharmacokinetic/pharmacodynamic*), QSB, PBPk.
592. Importowanie istniejących modeli z plików SBML (Systems Biology Markup Language).
593. Solvery równań różniczkowych zwyczajnych (ODEs) oraz solvery stochastyczne.
594. Biblioteka modeli PK.
595. Analiza wrażliwości i wpływu parametrów na dynamikę systemu.
596. Wykresy diagnostyczne dla indywidualnych oraz populacyjnych dopasowań (*fits*).
597. Metody tworzenia i optymalizacji harmonogramów dozowania.
598. Badanie wpływu zmian na odpowiedź modelu (wskaźniki Sobola, wieloparametryczny GSA - *Global Sensitivity Analysis*).
599. Silnik symulacji zdarzeń dyskretnych dla wielodomenowego modelownia złożonych systemów kolejkowych.
600. Predefiniowane biblioteki bloków, w tym kolejek, serwerów, generatorów, routingu (trasowania) oraz bloki sumatorów i rozdzielaczy.
601. Wbudowana agregacja statystyki dla wyznaczenia opóźnienia, przepustowości, średniej długości kolejki i innych wskaźników.
602. Animacja modelu do celów wizualizacji działania modelu oraz debugowania.
603. Możliwość tworzenia niestandardowych animacji do obserwowania wejść i zdarzeń.
604. Jedno środowisko do modelowania i symulacji systemów mechanicznych, elektrycznych, hydraulicznych, termicznych, a także innych wielodomenowych systemów fizycznych.
605. Biblioteki bloków do modelowania fizycznego oraz elementy matematyczne dla opracowania własnych komponentów.
606. Jednostki fizyczne dla parametrów i zmiennych, z automatyczną obsługą konwersji wszystkich jednostek.
607. Automatyczna redukcja zmiennych w równaniach symbolicznych oraz metody numeryczne do rozwiązywania równań różniczkowo-algebraicznych (DAE), w tym obsługa zdarzeń.

- 608.Specjalne solwery pozwalające na symulację w czasie rzeczywistym oraz testy hardware-in-the-loop (HIL).
- 609.Możliwość symulacji modeli, które zawierają bloki pochodzące z innych produktów związanych z modelowaniem fizycznym, bez konieczności zakupu tych produktów.
- 610.Umożliwia projektowanie i symulowanie systemów baterii i magazynowania energii.
- 611.Dostarcza narzędzia projektowe i parametryzowane modele do projektowania systemów akumulatorowych. Pozwala na tworzenie cyfrowych bliźniaków, przeprowadzanie wirtualnych testów pakietów baterii, projektowanie systemów zarządzania bateriami i ocenę zachowań systemów w różnych warunkach.
- 612.Battery Pack Model Builder to narzędzie projektowe, które umożliwia interaktywną ocenę różnych układów zestawów akumulatorów. Narzędzie automatyzuje tworzenie modeli symulacyjnych, które odpowiadają pożądanej topologii pakietu oraz zawiera połączenia płyt chłodzących, dzięki czemu można obserwować reakcje elektryczne i termiczne.
- 613.Parametryzowane modele pakietów baterii i systemów zarządzania bateriami demonstrują działania, włączając w to równoważenie ogniw i szacowanie stanu naładowania.
- 614.Modele typowych konfiguracji przekładni, w tym planetarnych, różnicowych oraz ślimakowych.
- 615.Modele elementów pojazdu, w tym silnika, opon, przemiennika momentu obrotowego oraz modeli dynamiki pojazdu.
- 616.Modele elementów translacyjnych, w tym śruby pociągowej, mechanizmu zębatkowego oraz przełożenia tarcia.
- 617.Możliwość określenia jednostek dla parametrów i zmiennych, a także automatyczna konwersja jednostek.
- 618.
- 619.Biblioteki komponentów elektrycznych, w tym czujników, siłowników, silników, maszyn, urządzeń pasywnych i urządzeń półprzewodnikowych.
- 620.Możliwość zmiany dokładności modelu, w tym efektów nieliniowych, ograniczeń parametrów pracy, modelowania błędów i zachowań zależnych od temperatury.
- 621.Konwersja elementów SPICE do modeli Simscape z wykorzystaniem importera listy połączeń.
- 622.Modele dedykowane określonym zastosowaniom, w tym popularne napędy elektryczne AC i DC, inteligentne sieci energetyczne i systemy energii odnawialnej.
- 623.Idealne przełączanie, dyskretyzacja i symulacja fazowa dla szybszego wykonywania modeli.
- 624.Bloki silników PMSM i BLDC uwzględniające zależności temperaturowe i straty magnetyczne.
- 625.Generacja, wizualizacja i eksport danych dotyczących napięcia i mocy w stanie ustalonym dla trójfazowych systemów przesyłowych prądu przemiennego.
- 626.Język Simscape oparty na MATLABie do tworzenia niestandardowych modeli komponentów.
- 627.Wsparcie do generowania
- 628.Modele pomp, w tym odśrodkowe, strumieniowe (*jet*) oraz osiowo-tłokowe (*axial-piston*).

629. Modele zaworów kierunkowych, w tym zawory zwrotne a także typowe konfiguracje dla zaworów 2, 4 i 6-drożnych.
630. Modele obrotowych i liniowych elementów wykonawczych, z możliwością dodania sił odśrodkowych i tarcia.
631. Modele wymienników ciepła do modelowania systemów ogrzewania i chłodzenia.
632. Modele zbiornika i rur, z uwzględnieniem efektów elewacyjnych, do modelowania systemów transportu płynów.
633. Konfigurowalna biblioteka dla typowych płynów hydraulicznych.
634. Modelowanie układów hydraulicznych za pomocą równań masy w stałej temperaturze.
635. Modelowanie wymiany ciepła pomiędzy cieczami dwufazowymi (*two-phase fluid*) i wilgotnym powietrzem (*moist air*).
636. Wychwytywanie ciepłych stanów nieustalonych związanych z masą ściany wymiennika ciepła.
637. Jedno środowisko do symulacji systemów mechanicznych w 3D razem z wielodomenowymi systemami fizycznymi i algorytmami sterowania.
638. Definiowanie bryły sztywnej za pomocą standardowej geometrii oraz niestandardowych profili.
639. Połączenia i ograniczenia mechaniczne dotyczące standardowych i niestandardowych relacji kinematycznych.
640. Modele symulacyjne pozwalają na obliczanie ruchu i obliczanie siły.
641. Animacje 3D dynamiki systemów wielocłonowych.
642. Środowisko graficzne do tworzenia i zarządzania hierarchicznymi schematami blokowymi.
643. Biblioteki predefiniowanych bloków do modelowania systemów dyskretnych oraz ciągłych.
644. Silnik symulacji ze stało- i zmiennokrokowymi solverami ODE.
645. Bloki do wizualizacji wyników symulacji.
646. Narzędzia zarządzania projektem i danymi.
647. Blok umożliwiający import algorytmów MATLABa do modelu.
648. Narzędzia importu kodu C i C++ do modeli.
649. Implementacja algorytmów na tanich platformach sprzętowych (*low-cost hardware platforms*), takich jak Arduino, Raspberry Pi, LEGO Mindstorms EV3.
650. Możliwość tworzenia projektów ułatwiających zarządzanie pracą w środowisku oraz integrację z systemami kontroli wersji (np. SVN, GIT, Github).
651. Bloki, aplikacje i funkcje do połączenia modeli Simulinka i algorytmów MATLABa z obiektami grafiki 3D w scenach wirtualnej rzeczywistości.
652. Możliwość animacji dynamiki systemów modelowanych za pomocą danych modułów, np. Simscape Multibody, Robotics System Toolbox, Aerospace Blockset.
653. Narzędzia do podglądu, renderowania i interakcji ze sceną wirtualnej rzeczywistości.
654. Możliwość nagrywania i odtwarzania animacji.
655. Możliwość importu plików CAD i URDF wykorzystywanych w budowaniu scen 3D.
656. Interakcja z widokiem 3D za pomocą joysticka, myszki 3D lub innego sprzętu.
657. Możliwość uruchomienia animacji w przeglądarce internetowej.
658. Możliwość renderowania animacji dla systemów czasu rzeczywistego.
659. Metryki i zestawy sprawdzenia zgodności dla wytycznych MISRA C:2012, MAB (MathWorks Advisory Board) oraz wytycznych dla projektów o wysokiej integralności

- (DO-178, ISO 26262, IEC 61508, IEC 62304 oraz związane z nimi standardy branżowe), a także możliwość definiowania własnych wytycznych.
- 660.Sprawdzanie zgodności kodu ze standardami bezpieczeństwa CERT C, CWE, ISO/IEC TS 17961.
 - 661.Ewaluacja zgodności modelu ze standardami, szacowanie rozmiaru modelu, stopnia skomplikowania.
 - 662.Automatyczny refactoring modelu umożliwiający zlokalizowanie zduplikowanych komponentów modelu i zmniejszenie złożoności projektu.
 - 663.Interfejs umożliwiający wykrywanie błędów w trakcie edycji modelu
 - 664.API do tworzenia własnych wytycznych dot. standardów projektowania.
 - 665.Funkcjonalność porównania automatycznie wygenerowanego kodu C/C++ z modelem źródłowym, wymagana przez standardy wysokiej integralności, w tym standard DO-178.
 - 666.Sprawdzanie ekwiwalentności bloków, diagramów stanu, parametrów i ustawień modelu z wygenerowanym kodem.
 - 667.Import kodu C jako biblioteki Simulink wielokrotnego użytku
 - 668.Analiza i raport strukturalnej równoważności.
 - 669.Możliwość dwukierunkowego śledzenia (*traceability*).
 - 670.Niezależność od narzędzi generacji kodu.
 - 671.Współpraca z modułem DO Qualification Kit.
 - 672.Automatyczna generacja kodu ANSI/ISO C oraz C++, a także plików wykonywalnych z dyskretnych, ciągłych lub hybrydowych modeli Simulinka, diagramów Stateflow i bloków MATLAB Function.
 - 673.Użycie wygenerowanego kodu w aplikacjach czasu rzeczywistego i nierzeczywistego.
 - 674.Wsparcie dla całkowitych, zmiennie- i stałoprzecinkowych typów danych.
 - 675.Strojenie parametrów oraz monitoring sygnałów w trybie symulacji zewnętrznej.
 - 676.Udostępnianie modeli Simulinka wraz z powiązaniem z nim kodem MATLABa w formie samodzielnych plików wykonywalnych.
 - 677.Tworzenie samodzielnych aplikacji graficznych wykorzystujących modele Simulinka, opracowanych w narzędziu App Designer.
 - 678.Generacja niezależnych komponentów FMU (*Functional Mockup Units*) zgodnych ze standardem FMI (*Functional Mockup Interface*) z modeli Simulinka, co umożliwia przeprowadzenie kosymulacji z zewnętrznymi środowiskami symulacji wspierającymi standard FMI.
 - 679.Możliwość tworzenia i udostępniania aplikacji webowych (wraz z modułem MATLAB Web App Server).
 - 680.Możliwość spakowania modeli Simulinka jako komponentów software'owych w celu zintegrowania ich z innymi językami programowania (wraz z modułem MATLAB Compiler SDK).
 - 681.Automatyczne strojenie systemów o architekturze regulacji SISO i MIMO, w tym regulatorów PID.
 - 682.Strojenie regulatorów PID zaimplementowanych na docelowej platformie.
 - 683.Znajdowanie punktów pracy i linearyzacja modeli.
 - 684.Oparte na symulacji obliczenia częstotliwościowej odpowiedzi modelu.
 - 685.Funkcje do tworzenia skryptów automatycznej linearyzacji.
 - 686.Estymacja parametrów modelu na podstawie testowych danych.

687. Równoczesna optymalizacja odpowiedzi modeli w dziedzinie czasu i częstotliwości (z wykorzystaniem modułu Simulink Control Design).
688. Graficzna specyfikacja wymagań nałożonych na odpowiedź modelu i wizualne monitorowanie postępu optymalizacji.
689. Interfejs skryptowy dla programistycznego projektowania problemów optymalizacji.
690. Wykorzystanie metod formalnych do identyfikacji ukrytych błędów projektowych w modelach Simulink, ręcznie pisanym kodzie C/C++ (S-Funkcji), kodzie MATLAB oraz obiektach typu System Object
691. Wykrywanie błędów, w tym martwej logiki (*dead logic*), przepiętnień w liczbach całkowitych, dzielenia przez zero oraz błędnego dostępu do tablic.
692. Generowanie przypadków testowych spełniających zdefiniowane wymagania funkcjonalne lub uzupełniających zadaną metrykę pokrycia modelu, w tym metrykę warunkową, decyzyjną i MC/DC.
693. Narzędzie do analizy funkcjonalnych zależności i problemów w dużych modelach.
694. Programowalne tworzenie filtrów dla wygenerowanego kodu
695. Wsparcie dla standardów przemysłowych dzięki modułom IEC Certification Kit i DO Qualification Kit.
696. Symulacja modeli Simulinka w czasie rzeczywistym na systemie Microsoft Windows® oraz MacOS.
697. Wizualizacja sygnałów i strojenie paramentów podczas wykonywania modelu.
698. Wydajność zbliżona do 1 kHz w normalnym trybie wykonania oraz 20 kHz w zewnętrznym trybie wykonania (z wykorzystaniem dodatkowego narzędzia do generacji kodu – modułu Simulink Coder).
699. Bloki wspierające ponad 250 modułów I/O (analogowych i cyfrowych I/O, liczników, enkoderów i wyjść częstotliwościowych) a także protokołów komunikacyjnych (w tym UDP, RS-232, CAN, CAN FD).
700. Połączenie z urządzeniami I/O zainstalowanymi w komputerze oraz z urządzeniami Thunderbolt™.
701. Automatyczna generacja kodu w językach Structured Text i Ladder Diagrams zgodnego z normą IEC 61131-3 na urządzenia PLC i PAC.
702. Wsparcie generacji kodu z modeli Simulinka, diagramów Stateflow i bloków MATLAB Function.
703. Generacja kodu w formatach plików wspieranych przez zintegrowane środowiska programistyczne IDE takie jak 3S-Smart Software Solutions CODESYS®, Rockwell Automation® Studio 5000, Siemens® TIA Portal i Omron® Sysmac® Studio.
704. Obsługa wielu typów danych takich jak logiczne, całkowite, wyliczeniowe, zniennoprzecinkowe a także wektory, macierze, magistrale oraz parametry strojone.
705. Tworzenie test benchy umożliwiających weryfikację wygenerowanego kodu.
706. Raporty z generacji kodu zawierające statyczne metryki kodu oraz narzędzia śledzenia zależności model – kod (*traceability*).
707. Możliwość wsparcia wybranych standardów przemysłowych wraz z modułem IEC Certification Kit (dla IEC 61508 i IEC 61511).
708. Tworzenie aplikacji czasu rzeczywistego z modeli Simulinka i uruchamianie ich na platformach czasu rzeczywistego Speedgoat podłączonych do systemów fizycznych.
709. Wykonywanie symulacji czasu rzeczywistego oraz testów, w tym szybkiego prototypowania sterowników (RCP), systemów DSP i wizyjnych oraz testów hardware-in-the-loop.

710. Wielozadaniowe oraz wielordzeniowe jądro czasu rzeczywistego z wsparciem wykonania współbieżnego.
711. Możliwość wykorzystania modułów Simscape do tworzenia modeli systemów fizycznych na potrzeby testów HIL.
712. Bloki sterowników dla układów I/O, w tym we/wy analogowych i cyfrowych, enkoderów, przetworników, elementów biernych, szeregowych, audio, pamięci współdzielonej, a także rekonfigurowalnych układów FPGA.
713. Możliwość wykonywania symulacji czasu rzeczywistego na układach FPGA Speedgoat (wraz z modułem HDL Coder).
714. Bloki sterowników dla protokołów oraz magistral, w tym Raw Ethernet, real-time UDP, CAN, EtherCAT, Ethernet/IP, Lin, SAE J1939, FlexRay, Camera Link, USB video, ARINC 429, a także MIL-STD-1553.
715. Interakcje z aplikacją czasu rzeczywistego, strojenie jej parametrów i streamowanie danych z aplikacji do narzędzi wizualizacyjnych w Simulinku.
716. Możliwość tworzenia aplikacji graficznych w narzędziu App Designer do strojenia parametrów i wizualizacji sygnałów w aplikacji czasu rzeczywistego dla zewnętrznych środowisk takich jak Microsoft .NET.
717. Analiza aplikacji czasu rzeczywistego z wykorzystaniem narzędzi pomiarowych, diagnostycznych i kalibracyjnych autorstwa firm trzecich, bazujących na protokole XCP.
718. Możliwość opracowywania i wykonywania testów w czasie rzeczywistym (wraz z modułem Simulink Test).
719. Generowanie raportów zawierających elementy modeli Simulinka takie jak na przykład diagramy blokowe, Stateflow® charts, MATLAB® czy Function blocks.
720. Automatyczne przechwytywanie wyników symulacji oraz specyfikacji modelu.
721. Formatowanie raportów oparte o szablony Microsoft® Word® i HTML.
722. Tworzenie interaktywnych widoków modeli Simulinka, które mogą być wyświetlane w przeglądarce internetowej wśród użytkowników nieposiadających licencji na moduł Simulink.
723. Aplikacja pozwalająca na tworzenie niestandardowych szablonów w wielu formatach, w tym HTML, PDF, RTF, Microsoft® Word®, Microsoft PowerPoint® i XML.
724. Uwzględnienie w raportach artefaktów dla standardów DO-178, IEC 61508 oraz ISO 26262, a także informacji o wymaganiach powiązanych z modelem, wynikach generacji kodu C/C++ oraz wynikach testów.
725. Narzędzia do tworzenia, zarządzania i wykonywania testów bazujących na symulacjach modeli, wygenerowanego kodu oraz symulowanego bądź fizycznego sprzętu.
726. Szablony do testów symulacyjnych, równoważnych oraz testów typu *baseline* umożliwiające testowanie funkcjonalne, regresyjne i równoległe w trybach software-in-the-loop (SIL), processor-in-the-loop (PIL) oraz hardware-in-the-loop (HIL).
727. Możliwość stworzenia jarzm testowych izolujących wybrany komponent (*component under test*).
728. Importowanie i testowanie zewnętrznego kodu C/C++.
729. Blok Test Sequence do uruchamiania testów i ich oceny.
730. Kryteria *pass-fail* zawierające tolerancje, limity i warunki czasowe.
731. Skrypty *setup* i *cleanup* do dostosowywania wykonywania testów.
732. Test Manager do tworzenia, wykonywania i organizowania przypadków testowych i ich wyników.

733. Modyfikowalna generacja raportów w celu udokumentowania wyników testów.
734. Możliwość śledzenia zależności między testami a wymaganiami (wraz z modułem Requirements Toolbox) oraz generacji raportów zawierających informację o pokryciu modelu testami (wraz z modułem Simulink Coverage).
735. Wsparcie dla wybranych standardów przemysłowych dzięki współpracy z modułami IEC Certification Kit (dla ISO 26262 i IEC 61508) oraz DO Qualification Kit (dla DO-178).
736. Współpracuje w wybranych narzędziach do wykonywania obliczeń równoległych w celu przeprowadzania masowych testów na klastrach obliczeniowych.
737. Narzędzia do modelowania, symulacji i analizy architektury sprzętowej i programowej układów ASIC, FPGA i SoC (system on chip).
738. Tworzenie architektury systemu z wykorzystaniem modeli pamięci i magistrali i I/O oraz symulacja architektury wraz z właściwymi algorytmami.
739. Ocenianie wydajności pamięci oraz wykonanych zadań poprzez symulację i profilowanie na urządzeniu.
740. Generacja projektów referencyjnych i kodu RTL.
741. Generacja kodu C/C++ (dzięki współpracy z Embedded Coder).
742. Eksport projektów referencyjnych na układy FPGA i SoC firm Xilinx® oraz Intel®, w tym rodziny Zynq®-7000, Ultrascale+™ oraz układy Intel SoC.
743. Wstępne przetwarzanie danych, ich edycja i wizualizacja w tradycyjnym wierszowo-kolumnowym środowisku Excel®.
744. Interfejs graficzny do dostosowywania preferencji narzędzia Spreadsheet Link.
745. Środowisko do graficznego modelowania i symulacji algorytmów opartych o diagramy przepływu sygnału, maszyny stanowe, tablice przejść stanowych oraz tablice prawdy.
746. Wsparcie dla rozbudowanych funkcjonalności wprowadzających do modelowanego systemu hierarchii, wykonywania równoległego, operatorów logicznych oraz zdarzeń.
747. Możliwość projektowania standardowych maszyn stanowych w konwencji Mealy'ego oraz Moore'a wraz z regułami, które wprowadzają.
748. Narzędzia ułatwiające debugowanie algorytmów logicznych jak np. breakpointy, wykonywanie sekwencyjne, analiza przejść i warunków oraz wskazywanie konfliktów przejść, martwej logiki, niespójności stanów czy też przededefiniowania/niedodefiniowania tablic prawdy.
749. Możliwość wykorzystania architektury Stateflow do tworzenia komponentów w module System Composer
750. Posiada dedykowany interfejs do analizy pracy maszyny stanowej (liczba przejść, czas aktywności stanów, zliczanie liczby wywołań funkcji)
751. Wsparcie dla wybranych narzędzi do automatycznej generacji kodu, testowania i analizy pokrycia testami, zarządzania wymaganiami, weryfikacji algorytmów pod względem zgodności z wybranymi normami IEC/ISO/EN, analizy formalnej algorytmów oraz statycznej analizy kodu źródłowego.
752. Możliwość integracji projektowanych algorytmów z zewnętrznym ręcznie pisanym kodem C.
753. Techniki regresyjne, w tym regresja liniowa, regresja liniowa uogólniona, regresja nieliniowa, regresja odporna.
754. Analiza wariancji ANOVA jednoczynnikowa, wieloczynnikowa, analiza kowariancji ANCOVA, modele mieszane.
755. Jedno- i wielowymiarowe rozkłady prawdopodobieństwa.

756. Generatory liczb losowych i pseudolosowych oraz łańcuchy Markova.
757. Testowanie hipotez statystycznych.
758. Testy hipotez dla różnych rozkładów, miary rozproszenia i położenia a także techniki planowania eksperymentów (DOE) dla planów zoptymalizowanych, planów czynnikowych oraz planów powierzchni odpowiedzi.
759. Nieparametryczne i semiparametryczne metody analizy danych dotyczących niezawodności i przeżywalności.
760. Algorytmy nadzorowanego uczenia maszynowego, w tym algorytm Maszyny Wektorów Nośnych (SVMs), drzewa klasyfikacyjne i regresyjne boosted/bagged, algorytm k-najbliższych sąsiadów, naiwny klasyfikator bayesowski, analizy dyskryminacyjne.
761. Algorytmy nienadzorowanego uczenia maszynowego, w tym algorytm k-średnich (centroidów), grupowania hierarchicznego, mieszanina rozkładów Gaussa i ukryte modele Markova.
762. Algorytmy przetwarzania dużych zbiorów danych (Big Data) m.in. redukcja wymiarowa, statystyki opisowe, regresje liniowe, logistyczne i analiza dyskryminacyjna.
763. Możliwość opracowywania modeli predykcyjnych w sposób interaktywny z użyciem aplikacji Classification Learner App oraz Regression Learner App, a także w sposób programistyczny z użyciem procesu AutoML.
764. Możliwość zastosowania interpretowalnych technik, takich jak wykresów częściowej zależności oraz techniki LIME.
765. Narzędzie do rozwiązywania, wyświetlania i modyfikowania równań w postaci symbolicznej.
766. Algebra liniowa, transformaty, rachunek całkowy i różniczkowy.
767. Zaimplementowane metody numeryczne do rozwiązywania równań różniczkowo-algebraicznych (DAE) oraz zwyczajnych równań różniczkowych (ODE).
768. Płynne przechodzenie pomiędzy systemami liczbowymi US i SI.
769. Konwersja wyrażeń symbolicznych do kodu MATLABa, Simulinka, Simscape'a, C, Fortrana, MathML oraz TeX.
770. Arytmetyka o zmiennej precyzji.
771. Wykorzystanie Live Edytora do przeprowadzania i dokumentowania obliczeń symbolicznych.
772. Narzędzie do definiowania, analizy i specyfikowania architektury systemów dla zastosowań projektowania systemów w oparciu o modele.
773. Wsparcie dla wybranych narzędzi do zarządzania wymaganiami w celu tworzenia architektury systemu w oparciu o wymagania oraz śledzenia wymagań od źródła do implementacji (*traceability*).
774. Możliwość definiowania komponentów, specyfikacji interfejsów i sygnałów, architektury i hierarchii systemu na podstawie stereotypów.
775. Widok typu *spotlight* to analizy wybranych komponentów i ich najbliższego otoczenia.
776. API do importu istniejących szczegółów projektu czy też dokumentacji interfejsów (ICD) do tworzonego modelu architektury.
777. Zróżnicowany opis zachowania komponentów za pomocą diagramów interakcji, diagramów stanu oraz modeli Simulinka.

778. Integracja z wybranymi narzędziami do modelowania i symulacji systemów dynamicznych oraz automatycznej generacji kodu, do rozbudowania modelu architektury systemu o funkcjonalność opisaną wykonywalnymi algorytmami.
779. Elastyczność w wyborze widoku architektury w zależności od indywidualnych potrzeb.
780. Możliwość hierarchicznej budowy modeli z ponownym wykorzystaniem komponentów bądź całych architektur systemów przy tworzeniu rozbudowanych i złożonych rozwiązań.
781. Wsparcie dla projektowania architektury systemów zgodnych ze standardem AUTOSAR.
782. Możliwość tworzenia alokacji pomiędzy architektuрами i komponentami w celu zaznaczenia bezpośrednich zależności.
783. Identyfikacja transmitancji, modelu procesu oraz modelu w postaci równań stanu z wykorzystaniem odpowiedzi w dziedzinie czasu i częstotliwości.
784. Estymacja online parametrów modelu.
785. Modelowanie szeregów czasowych (AR, ARMA) i prognozowanie.
786. Identyfikacja nieliniowych modeli ARX i modeli Hammerstein-Wienera z nieliniowościami wejścia-wyjścia, takimi jak nasycenie i martwa strefa.
787. Identyfikacja liniowych i nieliniowych systemów w postaci „szarej skrzynki”.
788. Estymacja opóźnienia, usuwanie trendu, filtrowanie, resampling oraz rekonstrukcja brakujących danych.
789. Aplikacja do interaktywnej estymacji modeli liniowych i nieliniowych systemu na bazie zmierzonych danych wejściowych i wyjściowych.
790. Analiza, wstępne przetwarzanie i wizualizacja informacji zawartych w tekście.
791. Narzędzia do przetwarzania surowych danych tekstowych pochodzących ze źródeł takich jak sprzętowe logi, kanały informacyjne, ankiety, raporty i *social media*.
792. Import danych tekstowych z pojedynczych plików bądź dużych kolekcji plików, w tym plików PDF, HTML, Microsoft® Word® i Excel®.
793. Funkcje filtracji tekstu, usuwające określoną zawartość, m.in. adresy URL, znaczniki HTML, znaki interpunkcyjne.
794. Trenowanie typu word embedding, metody uczenia maszynowego LSA, LDA i wsparcie dla modeli word2vec, skip-gram, FastText, GloVe.
795. Tworzenie wykresów typu Word Cloud i Text Scatter.
796. Statystyki częstotliwości występowania słów oraz generacja macierzy TF-IDF.
797. Ekstrakcja podsumowania z tekstu.
798. Metody uczenia głębokiego do analizy sentymentalnej, klasyfikacji tekstu oraz do generacji nowego tekstu na bazie tekstu analizowanego.
799. Usługa platformy analitycznej do projektów IoT.
800. Agregacja, wizualizacja i analiza danych strumieniowanych do chmury obliczeniowej.
801. Konfiguracja urządzeń do wysyłania danych na platformę ThingSpeak z wykorzystaniem REST API lub MQTT.
802. Zbieranie danych z urządzeń lub innych źródeł danych.
803. Natychmiastowa wizualizacja danych przychodzących na serwer oraz danych historycznych.
804. Wstępna obróbka i analiza zebranych danych, z wykorzystaniem zintegrowanego języka MATLAB.

805. Uruchamianie programów analizujących dane, bazujących na zaplanowanych chwilach czasu lub wyzwalanych zdarzeniowo.
806. Reakcje na pojawiające się dane i wyniki analiz z wykorzystaniem platform Twilio lub Twitter.
807. Narzędzia do projektowania, symulacji, analizy i wdrażania aplikacji dla dronów i bezzałogowych statków powietrznych
808. Aplikacje referencyjne dla popularnych zastosowań dronów i bezzałogowych statków powietrznych, takie jak autonomiczne dostarczanie paczek z wykorzystaniem wielowirnikowego bezzałogowego statku powietrznego
809. Aplikacja Flight Log Analyzer pozwalająca interaktywnie analizować ścieżki lotów 3D, informacje telemetryczne i odczyty z czujników w formatach takich jak TLOG, ULOG i innych.
810. Komunikacja ze sprzętem za pomocą protokołu MAVLink
811. Możliwość dodania opisu terenu i siatki w formacie DTED (Digital Terrain Elevation Data)
812. Możliwość tworzenia scenariuszy, wykorzystywania modeli czujników i generowania syntetycznych danych do testowania algorytmów lotu w symulowanym środowisku
813. Tworzenie, testowanie i wizualizowanie algorytmów autonomicznego lotu bezzałogowych statków powietrznych i dronów w symulowanym środowisku 3D renderowanym przy użyciu silnika Unreal Engine® firmy Epic Games®, z jednoczesnym generowaniem odczytów wysokiej jakości z kamery i czujnika LIDAR.
814. Możliwość dostosowania stanu pogody i położenia słońca w symulowanych scenach.
815. kodu C/C++ do szybkiego prototypowania, testów Hardware-in-the-Loop, wsparcie generacji kodu C++ dla systemu autopilota PX4 (generacja z wykorzystaniem Embedded Coder®).
816. Narzędzie do rozwoju systemów sterowania, testowania algorytmów ADAS, autonomicznej jazdy i planowania.
817. Gotowe modele do symulacji manewrów drogowych w środowisku 3D.
818. Interfejs ze środowiskiem 3D Unreal Engine® (silnikiem do gier firmy Epic Games®).
819. Modele ruchu zróżnicowanych pojazdów oraz przyczep jako bryły sztywnych o 1, 3 lub 6 stopniach swobody.
820. Skonfigurowane sceny wizualizacji drogi, znaków drogowych, budynków, roślinności i innych obiektów wokół pojazdu.
821. Predefiniowane sceny 3D, w tym proste i zakrzywione drogi, autostrady, parkingi.
822. Możliwość dostosowania stanu pogody i pozycji słońca w symulowanych scenach.
823. Biblioteka parametryzowalnych komponentów do modelowania napędu, układu kierowniczego, zawieszenia, nadwozia pojazdu, hamulców i opon.
824. Połączone dynamiczne modele opon o podłużnym i poprzecznym poślizgu.
825. Symulacja dynamiki pojazdu w celu analizy systemu pod kątem wymagań w normalnych i ekstremalnych warunkach drogowych.
826. Modele sterownika jazdy do generowania poleceń sterujących, które podążają za predefiniowaną ścieżką.
827. Funkcje MATLABa do transmitowania, odbierania, kodowania i dekodowania komunikatów magistrali CAN, CAN FD i J1939.
828. Bloki Simulinka do połączenia modelu z magistralą CAN, CAN FD lub J1939.

829. Wsparcie dla protokołu XCP do komunikacji z ECU z wykorzystaniem plików opisu A2L.
830. Obsługa baz danych CAN w formacie *.dbc*.
831. Aplikacje CAN Explorer oraz CAN FD Explorer pozwalające na dekodowanie i wizualizację danych oraz konfigurację urządzenia w przejrzysty sposób.
832. Monitorowanie, filtracja i analiza bieżących danych w magistrali CAN oraz opcja logowania i rejestracji komunikatów do celów późniejszej analizy.
833. Symulacja ruchu w wirtualnej sieci CAN.
834. Wsparcie dla urządzeń firm Vector, Kvaser, PEAK-System i National Instruments® pracujących na magistrali CAN.
835. Import plików MDF (w standardzie 3.0 i wyższych).
836. Obsługa plików ASAM CDFX.
837. Projektowanie systemów przetwarzania obrazów, wideo oraz wizji komputerowej na układy FPGA i ASIC.
838. Algorytmy do generacji czytelnego, syntetyzowalnego kodu VHDL i Verilog (wraz z modułem HDL Coder).
839. Sygnał kontroli synchronizacji wideo do obsługi niedopasowania czasowego i różnych rozdzielczości.
840. Konfigurowana ilość ramek i ich rozmiar, włączając w to filmy w technologii HFR (high frame rate) i rozdzielczość do 8k.
841. Przetwarzanie obrazów i wideo oraz algorytmy systemów wizyjnych z architekturą *pixel-streaming*, w tym poprawa jakości obrazu, filtrowanie, morfologia i statystyki.
842. Możliwość weryfikacji FPGA-in-the-loop na układach Xilinx®, Intel® i Microsemi® (wraz z modułem HDL Verifier).
843. Ciągła transformacja falkowa (CWT), skalogram i falki koherentne do celów analizy cech spektralnych w czasie, identyfikacji powtarzalnych zmiennych w czasie wzorów w dwóch sygnałach oraz filtracji zlokalizowanej w czasie.
844. Dyskretna analiza falkowa, w tym decymowana, podwójnego drzewa i transformat falkowych pakietowych – do celów analizy sygnałów i obrazów o różnych rozdzielczościach i znajdowania punktów zmiany, nieciągłości i innych artefaktów.
845. Zastosowanie transformacji stałego Q (CQT) oraz transformacji falkowej z przestrajalnym współczynnikiem Q (TQWT) pozwalającej na dyskretną analizę sygnałów z wykorzystaniem wybranego współczynnika Q dla dekompozycji i rekompozycji.
846. Kompresja i rekonstrukcja sygnałów i obrazów, w tym dopasowujące algorytmy poszukiwań.
847. Interaktywne aplikacje, w tym narzędzie Wavelet Signal Denoiser pozwalające na wizualizację i odsumowanie rzeczywistych sygnałów jednowymiarowych.
848. Banki rekonstrukcyjnych filtrów ortogonalnych i nieortogonalnych, w tym Daubechies, Coiflet, Haar, Fejer-Korovkin.
849. Wsparcie generacji kodu C/C++ oraz kodu CUDA z większości funkcji modułu.
850. Bloki Simulinka do projektowania i implementacji na układach FPGA, ASIC i SoC systemów komunikacji bezprzewodowej: LTE i 5G oraz własnej bazującej na modulacji OFDM.
851. Opracowane algorytmy gotowe do generacji z nich kodu VHDL lub Verilog (z modułem HDL Coder).
852. Algorytmy zoptymalizowane pod kątem zużycia zasobów oraz wydajności.
853. Implementacja enkoderów i dekoderów turbo, Viterbiego, splotowych oraz CRC.
854. Modulatory i demodulatory OFDM.

855. Algorytm *depuncturingu* i szybkiej transformaty Fouriera o zmiennej długości.
856. Synchronizacja sygnału (PSS i SSS) oraz dekodowanie (MIB oraz SIB).
857. Konwersja między sygnałami strumieniowanymi próbkami po próbkach, a przetwarzanymi w ramach.
858. Prototypowanie na platformach FPGA i SoC dostarczanych przez firmy Xilinx® oraz Intel®.
859. Możliwość testów sygnałów nadawanych i odbieranych przez rzeczywiste transceivery, dzięki kooperacji z modułem Communications Toolbox wraz z jego pakietami wsparcia sprzętowego.
860. Możliwość weryfikacji FPGA-in-the-loop na układach Xilinx®, Intel® i Microsemi® (wraz z modułem HDL Verifier).
861. Zawiera modele aplikacji referencyjnych pozwalające na uruchamianie powszechnie dostępnych urządzeń SDR (Software-Defined Radio) jak np. USRP.
862. Pozwala na łączenie ze wspieranymi urządzeniami SDR, konfigurowanie i wykonywanie przygotowanych bitstreamów FPGA oraz wykonywanie pomiarów w czasie rzeczywistym.
863. Transmisja i przechwytywanie wysokoczęstotliwościowych sygnałów (aż do 250Mps) na wspieranych platformach SDR z użyciem wewnętrznej pamięci.
864. Wsparcie dla urządzeń Ettus Research.
865. Możliwość wyboru czasów próbkowania dla każdej aplikacji tworzonej z Wireless Testbench umożliwiająca pracę z niestandardowymi sygnałami jak i również standardowymi jak np. 5G, WLAN oraz DVB-S2.
866. Modele systemów WLAN zgodne ze standardami IEEE 802.11ax/ac/ad/ah oraz IEEE 802.11b/a/g/n/j/p.
867. Modelowanie nadajników, kodowanie kanałów (BCC i LDPC), modulacje (OFDM, DSSS, CCK), mapowanie strumieni przestrzennych i odbiorniki MIMO.
868. Generacja przebiegów zgodnych ze standardami IEEE 802.11ax/ac/ad/ah/j/p/n/g/a/b.
869. Generacja i analiza przebiegów zgodnych ze standardem IEEE 802.11be dla pojedynczego użytkownika.
870. Pozycjonowanie i lokalizacja 3D we wnętrzu za pomocą danych w standardzie IEEE 802.11az i metod uczenia głębokiego.
871. Graficzna aplikacja do interaktywnej generacji przebiegów WLAN wraz z uwzględnieniem zakłóceń w sygnale, takich jak AWGN, offset fazy, częstotliwości, składowej stałej, *IQ imbalance*, nieliniowości. Wizualizacja wyników na diagramach konstelacji, analizatorach widma, siatce OFDM i wykresach czasowych.
872. Modele kanałów, w tym TGay, TGax, TGac, TGah i CCK.
873. Operacje na sygnałach po stronie odbiorczej – synchronizacja ramek, korekcja offsetu częstotliwości, estymacja i equalizacja kanału.
874. Generacja, analiza i dekodowanie ramek MAC zgodnych z IEEE 802.11 (MPDU, AMSDU, AMPDU).
875. Modelowanie warstw PHY i MAC oraz dzielonego kanału komunikacji.
876. Algorytmy kształtowania wiązki.
877. Możliwość pomiaru mocy kanału, maski spektralnej, EVM, PER, BER, przepustowości oraz zajętości pasma
878. Pomiar dokładności modulacji nadajnika oraz minimalnej czułości wejściowej odbiornika.
879. Generacja środowiska testowego do połączeń w symulacji komunikacji WLAN.

