

Transient engine simulations using the stochastic reactor model for driving cycle performance studies



**Silesian University
of Technology**

Reddy Babu Siddareddy, MSc

Rozszerzone streszczenie

Academic Supervisor

Prof. dr hab. inż. Wojciech Adamczyk



Industry Supervisor

dr inż. Michał Pasternak



September 2025

1 Zakres i cel

Optymalizacja silników spalinowych pozostaje kluczowym priorytetem inżynierskim w zakresie ekonomii paliwowej, wykorzystania paliw alternatywnych oraz redukcji emisji spalin, ponieważ normy emisji na całym świecie stają się coraz bardziej rygorystyczne. Proces optymalizacji wymaga szeroko zakrojonych testów w celu oceny różnych strategii pracy silnika. Strategie te muszą być odpowiednie dla pracy silnika w rzeczywistych, przejściowych warunkach jazdy, takich jak te odwzorowane przez Worldwide harmonised Light vehicles Test Procedure (WLTP) [1] lub Non-Road Transient Cycle (NRTC) [2]. Gdy rozwój i optymalizacja silników spalinowych odbywają się wyłącznie na stanowiskach testowych, proces ten jest kosztowny i czasochłonny, co powoduje potrzebę stosowania wiarygodnych narzędzi symulacyjnych wspierających ten proces.

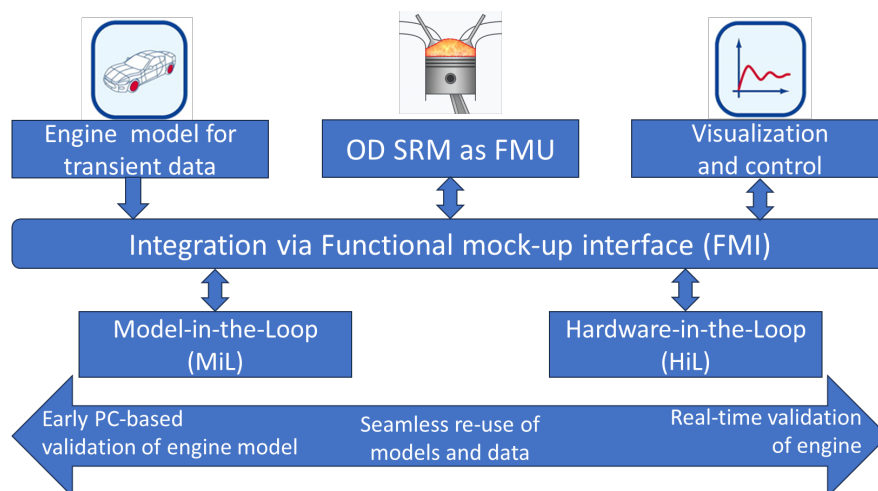
Dokładne modelowanie i symulacja zmiennych w czasie warunków pracy silnika podczas WLTP lub NRTC wymagają odpowiednich metod symulacji spalania i emisji zanieczyszczeń w tych rzeczywistych scenariuszach. Metody takie muszą być wydajne obliczeniowo, aby umożliwić skuteczne zastosowanie w warunkach przejściowych. Z drugiej strony muszą być wystarczająco dokładne, aby przewidywać powstawanie zanieczyszczeń, co jest wymagane w procedurach WLTP lub NRTC. Stochastyczny model reaktora (SRM) procesów zachodzących w cylindrze silnika wydaje się być odpowiednim podejściem modelowym do osiągnięcia tego celu. W kontekście symulacji przejściowych silnika interesujące jest nie tylko symulowanie samych procesów zachodzących w cylindrze silnika, ale także symulowanie zachowania całego układu napędowego silnika, wraz z układem oczyszczania spalin, aby umożliwić opracowanie strategii sterowania silnikiem lub symulacji, które ściśle naśladują warunki podczas rzeczywistych testów jezdnych do celów certyfikacji. Taką zdolność można osiągnąć poprzez integrację modelu SRM z platformami symulacyjnymi typu Model-in-the-Loop (MiL) lub Hardware-in-the-Loop (HiL).

MiL odnosi się do reprezentacji modeli systemów fizycznych w formie oprogramowania. W odniesieniu do silników spalinowych, testy MiL zazwyczaj obejmują walidację algorytmów sterowania silnikiem względem szczegółowych modeli symulacyjnych silnika, zanim zostanie wprowadzone jakiejkolwiek fizyczne urządzenie. Metoda ta ma szczególne znaczenie w symulacjach przejściowych pracy silnika, ponieważ umożliwia opracowywanie i udoskonalanie strategii sterowania w warunkach przejściowych, w bezpiecznym i w pełni wirtualnym środowisku [3, 4]. Z kolei HiL odnosi się do systemu reprezentującego fizyczny sprzęt, Electronic Control Unit (ECU) oraz modele oprogramowania. W odniesieniu do silników spalinowych, testy HiL stanowią metodę walidacji, w której rzeczywisty sprzęt sterujący silnikiem jest testowany względem modeli symulacyjnych działających w czasie rzeczywistym. Dla symulacji przejściowych silników, HiL umożliwia testowanie sprzętu silnikowego oraz identyfikację jego mocnych i słabych stron podczas pracy w warunkach przejściowych, bez konieczności stosowania w pełni zmontowanego stanowiska testowego, co pozwala upewnić się, że jednostki sterujące działają niezawodnie w realistycznych warunkach pracy [5, 6].

Zastosowanie SRM jako zintegrowanego łańcucha narzędzi z platformami MiL/HiL wydaje się obiecujące ze względu na oczekiwaną elastyczność w pracy z różnymi silnikami i paliwami oraz możliwość współsymulacji z modelami innych komponentów układów pojazdów. Ponadto, 0D formułowanie SRM zapewnia niski koszt obliczeniowy przy jednoczesnym zachowaniu wysokiej dokładności przewidywania tworzenia się zanieczyszczeń dzięki zastosowaniu szczegółowej kinetyki reakcji chemicznych. Dlatego taki zintegrowany łańcuch narzędzi powinien przyczynić

się do rozwoju czystszych i bardziej efektywnych technologii silników spalinowych, co stanowiło motywację do podjęcia niniejszej pracy.

SRM znalazł zastosowanie jako efektywny model do symulacji procesów silnikowych w warunkach ustalonych [7, 8, 9, 10], jednak dotychczas nie był stosowany do symulacji w warunkach przejściowych, takich jak te występujące w WLTP lub NRTC. Ponadto, SRM jak do tąd nie był kompatybilny ze standardami Functional Mock-up Interface (FMI) oraz Functional Mock-up Unit (FMU), które są kluczowe dla zastosowania SRM w środowiskach symulacyjnych takich jak HiL i MiL. Ta luka w interoperacyjności ograniczała zakres zastosowań SRM do symulacji w czasie rzeczywistym, wymaganych przez platformy HiL. Rozwiązanie tych problemów jest kluczowe dla pełnego wykorzystania potencjału SRM w kompleksowych symulacjach warunków przejściowych. W odniesieniu do tych luk, celem niniejszej pracy jest opracowanie łańcucha narzędzi symulacyjnych umożliwiającego symulacje przejściowe silnika oparte na technologii SRM. Zdefiniowany cel został graficznie przedstawiony na Rysunku 1.



RYSUNEK 1: Koncepcja zintegrowanego łańcucha narzędzi obejmującego model OD SRM, MiL/HiL oraz technologię FMI/FMU do symulacji w stanach nieustalonych.

W szczególności praca koncentruje się na opracowaniu środowiska (ang. framework) FMU poprzez FMI dostosowanego do SRM, w celu umożliwienia jego integracji z zewnętrznymi środowiskami symulacyjnymi, platformami współsymulacji oraz środowiskami testów w czasie rzeczywistym, takimi jak HiL i MiL. Aby zrealizować to zadanie, istniejąca baza kodowa SRM została zaktualizowana, aby umożliwić symulacje przejściowe poprzez uwzględnienie zmiennych w czasie parametrów silnika, takich jak prędkość obrotowa, obciążenie, strategia wtrysku oraz fazy rozrządu, które ulegają zmianie podczas pracy przejściowej. Ponadto konieczne było opracowanie frameworka umożliwiającego generowanie FMU dla SRM z wykorzystaniem standardu FMI oraz implementację interfejsu zgodnego z FMI. Ostatecznie framework musiał zostać zoptymalizowany w celu minimalizacji kosztów obliczeniowych, zapewniając możliwość pracy w czasie rzeczywistym na platformach stand-alone, MiL i HiL, przy jednoczesnym zachowaniu zdolności SRM do przewidywania procesu spalania i emisji dla silników Compression-Ignition (CI) oraz Spark-Ignition (SI).

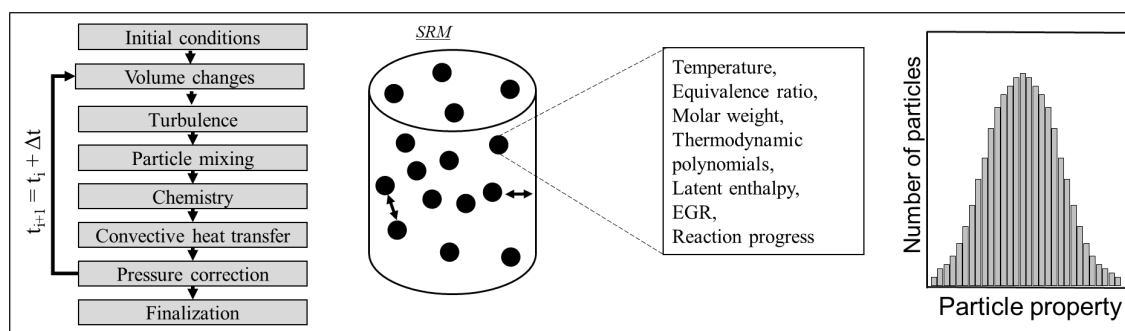
Ogólnie, framework przedstawiony schematycznie na Rysunku 1 został zaprojektowany z myślą o przenośności i adaptowalności w środowiskach MiL i HiL, zapewniając standardowe interfejsy FMI/FMU dla płynnej integracji z przepływami pracy w czasie rzeczywistym. Takie podejście łączy dokładność fizyczną z efektywnością obliczeniową, czyniąc symulacje SRM dotyczące osiągnięć silnika i emisji spalin praktycznym narzędziem dla współczesnych zastosowań w testowaniu silników, kalibracji oraz wirtualnej walidacji w warunkach przejściowych.

2 Metody i narzędzia

Łańcuch narzędzi zaproponowany w niniejszej pracy integruje 0D SRM procesów zachodzących w cylindrze silnika, wykorzystujący tablicowaną chemię dla spalania i tworzenia emisji, z platformami MiL i HiL przy użyciu standardu FMI. Poprzez enkapsulację SRM w FMU, opracowany framework umożliwia symulacje międzyplatformowe, łącząc dokładność modelowania fizycznego z praktyczną użytecznością w procesie rozwoju silników [11, 12, 13].

2.1 Stochastic Reactor Model (SRM)

SRM jest zerowymiarowym modelem procesów fizycznych i chemicznych zachodzących w cylindrze silnika spalinowego. Model ten traktuje gaz znajdujący się w cylindrze jako zbiór hipotetycznych cząstek. Cząstki te mogą oddziaływać ze sobą poprzez mieszanie oraz wymianę ciepła ze ściankami cylindra. Każda z takich cząstek hipotetycznych charakteryzuje się temperaturą, masą oraz stężeniem poszczególnych składników. W pewnym sensie każda cząstka może być interpretowana jako realizacja przepływu w określonym momencie i miejscu rzeczywistej komory spalania. Cząstki poddawane są reakcjom chemicznym zgodnie z zastosowanym mechanizmem reakcji. Wszystkie procesy oparte na cząstkach są obliczane sekwencyjnie z wykorzystaniem metody operator splitting loop. Mieszanie między cząstkami jest procesem modelowanym, z zastosowaniem różnych typów modeli mieszania. Wspólną cechą tych modeli jest konieczność określenia skali czasowej mieszania, która determinuje częstotliwość, z jaką cząstki się mieszają, a w konsekwencji wpływa na niejednorodność mieszaniny pod względem temperatury i składu. To z kolei oddziałuje na przebieg procesu spalania oraz formowanie emisji [14]. Schematyczna wizualizacja koncepcji SRM przedstawiona została na Rysunku 2.



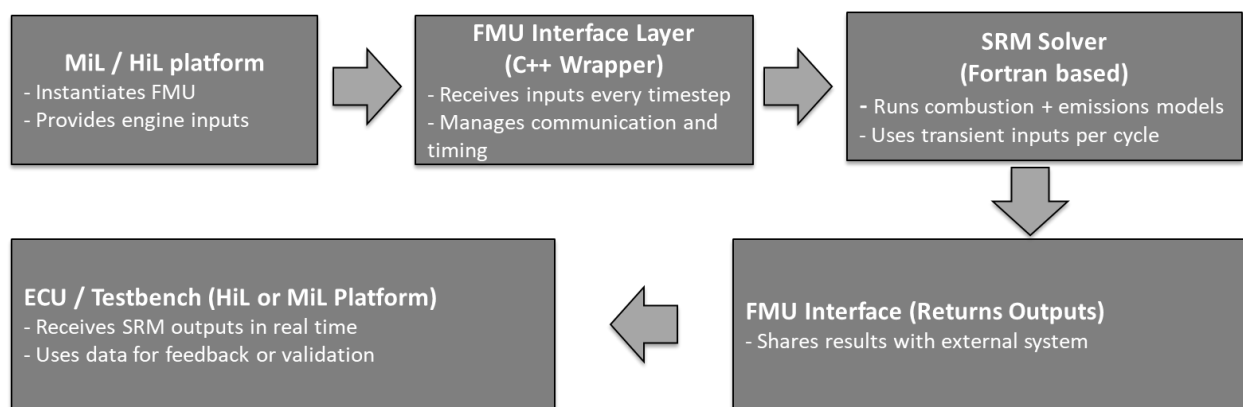
RYSUNEK 2: Schemat koncepcji zastosowania SRM do symulacji silników spalinowych [15].

W niniejszej pracy zastosowano model mieszania typu coalescence-dispersion z losowym (stochastic-based) doбором cząstek do mieszania. Historia skali czasowych mieszania jest używana z modelu turbulencji jako stosunek energii kinetycznej turbulencji k do jej dyssypacji ϵ . Procesy spalania i tworzenia emisji są obliczane na podstawie szczegółowych mechanizmów kinetyki reakcji chemicznych, wykorzystywanych w postaci tablicowanej chemii (tabulated chemistry) z zastosowaniem zmiennej postępu spalania (Combustion Progress Variable, CPV). Zastosowanie tablicowanej chemii pozwala na redukcję kosztów obliczeniowych do poziomu milisekund dla pełnego cyklu pracy silnika, bez pogorszenia jakości wyników. Model SRM może symulować spalanie wstępnie zmieszane z propagacją płomienia, charakterystyczne dla silników Spark-Ignition (SI), oraz spalanie niewstępnie zmieszane, typowe dla silników Compression-Ignition (CI) [14].

2.2 Integracja FMU i SRM

W niniejszej pracy model SRM został skompilowany jako FMU w celu umożliwienia integracji ze środowiskami MiL/HiL. Zastosowano współsymulację (co-simulation), w której SRM działa jako niezależny solver komunikujący się z zewnętrznym masterem symulacji (np. MATLAB/Simulink lub dSPACE ASM) poprzez standaryzowaną wymianę sygnałów. Takie podejście umożliwia modułowe połączenie SRM z innymi podsystemami silnika przy zachowaniu pracy w czasie rzeczywistym oraz wysokiej efektywności obliczeniowej. FMU pełni rolę kontenera dla solvera SRM i jest określany jako FMU/SRM.

FMU został zbudowany zgodnie ze standardem FMI 2.0 dla współsymulacji (co-simulation). W tym układzie SRM działa niezależnie, przyjmując sygnały wejściowe i wysyłając zmienne wyjściowe w stałych odstępach czasu komunikacji. Do sygnałów wejściowych należą: prędkość obrotowa silnika, fazy rozrządu, ciśnienia i temperatury w układach dolotowym i wylotowym, współczynnik recyrkulacji spalin (EGR), masa paliwa na wtrysk oraz współczynnik nadmiaru powietrza (air-fuel ratio), a także sygnały sterujące fazami rozrządu i strategią wtrysku. Sygnały wyjściowe obejmują parametry pracy silnika rozwiązywane w ujęciu cyklowym, takie jak maksymalne ciśnienie, IMEP, moment obrotowy oraz emisje spalin.



RYSUNEK 3: Przepływ pracy symulacyjnej z modelem SRM opartym na technologii FMU.

FMU jest tworzony poprzez połączenie kodu solvera SRM z plikiem manifestu XML zgodnym ze standardem FMI oraz biblioteką dynamiczną. Master współsymulacji (co-simulation master) zarządza synchronizacją czasu i wyrównywaniem kroków obliczeniowych solvera, zapewniając dokładną komunikację z narzędziami zewnętrznymi (Rysunek 3). FMU jest zaimplementowany w języku C++ jako wrapper (osłona) wokół solvera SRM opartego na języku Fortran. Wrapper dostarcza standardowe interfejsy FMI 2.0 dla inicjalizacji, obsługi sygnałów wejściowych/wyjściowych, kroków czasowych oraz raportowania błędów. Solver napisany w Fortranie wykonuje obliczenia cyklu spalania i zwraca wyniki dotyczące osiąągów silnika oraz emisji spalin. Buforowanie danych zapewnia, że jeśli SRM opóźni się podczas cyklu, przekazywane są najnowsze wyniki w celu utrzymania ciągłości działania. Taka modułowa struktura umożliwia testowanie różnych solverów SRM bez konieczności ponownej kompilacji FMU.

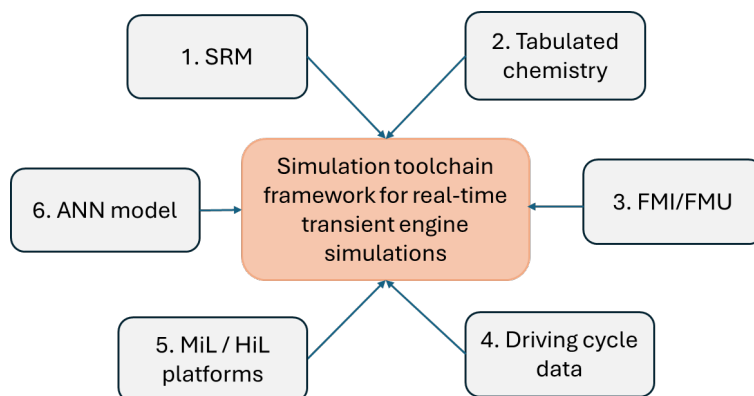
2.3 Metamodel oparty na sztucznej sieci neuronowej (ANN)

Aby dodatkowo obniżyć koszty obliczeniowe poniżej czasu rzeczywistego, opracowano metamodel symulatora cyklu jezdny oparty na sztucznej sieci neuronowej (Artificial Neural Network, ANN). Sieć ANN jest trenowana z wykorzystaniem danych generowanych przez symulator oparty na modelu fizycznym wykorzystującym SRM symulacji jazdy, co czyni ten etap rozwoju

wstępnym. Trójwarstwowa sieć została zaimplementowana w języku Python z użyciem bibliotek Keras, TensorFlow oraz scikit-learn [16, 17, 18], z 128 neuronami w warstwie wejściowej, 64 w warstwie ukrytej oraz warstwą wyjściową odpowiadającą liczbie przewidywanych wielkości. Funkcja aktywacji ReLU wprowadza nieliniowość, natomiast optymalizator Adam [19] realizuje stochastyczny spadek gradientu (stochastic gradient descent). Schemat działania obejmuje trenowanie sieci na danych z przejściowych warunków pracy silnika, przewidywanie wyników „engine-out” oraz iteracyjne aktualizowanie modelu do momentu, aż błąd predykcji spadnie poniżej założonego progu 5%.

2.4 Łańcuch narzędzi symulacyjnych

Cały łańcuch narzędzi zaproponowany w niniejszej pracy wraz z jego komponentami został schematycznie przedstawiony na Rysunku 4. Ten zunifikowany framework symulacyjny umożliwia przeprowadzanie symulacji przejściowych silnika w czasie rzeczywistym poprzez połączenie SRM, tablicowanej chemii, interfejsów FMU, meta-modelu opartego na ANN oraz ich integrację w platformach testowych MiL i HiL.



RYSUNEK 4: Komponenty opracowanego łańcucha narzędzi do symulacji nieustalanej pracy silnika.

TABELA 1: Mapowanie elementów łańcucha narzędzi symulacyjnych na cele techniczne.

Element diagramu	Rola w łańcuchu narzędzi
SRM	Model 0D procesów w cylindrze silnika
Chemia tabelaryczna	Modelowanie spalania i emisji Formulacja paliwa zastępczego
FMI/FMU	Interfejsowanie, modułowość i kompatybilność systemowa
Dane cyklu jazdy	Wejścia do symulacji nieustalonego cyklu pracy silnika
Platformy MiL/HiL	Integracja z platformami testowymi stron trzecich
Metamodel ANN	Metamodel do redukcji kosztów obliczeniowych

Framework jest modułowy, co pozwala każdemu komponentowi działać niezależnie, komunikując się poprzez interfejs FMU. Dane dotyczące cyklu jazdy dostarczają wejść przejściowych, które są przetwarzane przez SRM, natomiast FMU zapewnia synchronizację oraz przenośność pomiędzy platformami symulacyjnymi. W centrum uwagi symulacji przejściowych silnika znajduje

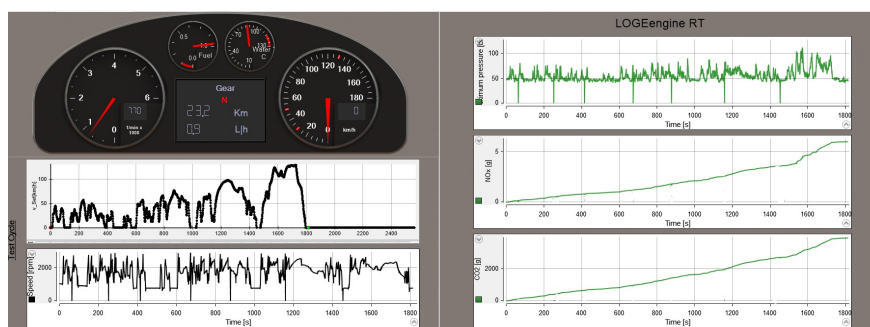
się interfejs FMU, który integruje wszystkie komponenty w jednolite, skalowalne i przenośne środowisko symulacyjne. Ten łańcuch narzędzi zapewnia metodyczne podejście do modelowania silników opartego na fizyce, odpowiednie do zastosowań w czasie rzeczywistym oraz na poziomie systemowym.

3 Wyniki

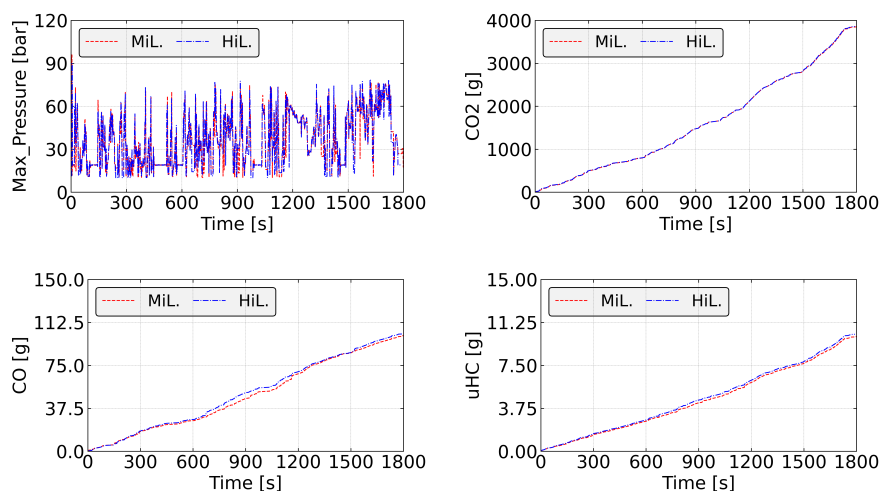
Opracowany łańcuch narzędzi wykorzystujący SRM został zastosowany zarówno w trybie stand-alone, jak i w zintegrowanych środowiskach MiL/HiL dla silników CI i SI w przejściowych warunkach jazdy zgodnych z cyklami WLTP i NRTC. Treść tego rozdziału opiera się na opublikowanych wynikach [15, 14, 20, 21]. Rozdział rozpoczyna się od integracji modelu silnika opartego na SRM poprzez technologię FMU/FMI z platformami MiL i HiL, tworząc kompletny symulator cyklu jazdy oparty na fizyce. Następnie przedstawiono przewidywania dotyczące osiąarów silnika i emisji spalin dla silnika CI w warunkach cyklu WLTP, w tym zmienność emisji spalin pomiędzy kolejnymi cyklami jazdy. Kolejno zaprezentowano symulacje silnika SI podczas faz rozruchu zimnego oraz współsymulacje silnika i katalizatora. Na końcu przedstawiono wyniki meta-modelu opartego na ANN dla symulatora sterowanego przez SRM.

3.1 Symulacje silnika oparte na platformach MiL/HiL

W tej sekcji przedstawiono wydajność w czasie rzeczywistym, funkcjonalność oraz integralność opracowanego łańcucha narzędzi w środowiskach MiL i HiL podczas standardowego cyklu jazdy WLTP. Model silnika zbudowany w SRM oraz dane dotyczące cyklu jazdy wykorzystane w tym badaniu pochodzą z pakietu dSPACE ASM. Silnik jest silnikiem SI zasilanym benzyną i wyposażonym w system wtrysku portowego. Rysunek 5 przedstawia wykonanie symulacji MiL/HiL w czasie rzeczywistym w dSPACE ControlDesk, które rejestruje emisje „engine-out” oraz kluczowe parametry pracy silnika podczas przejściowego cyklu WLTP. Natomiast Rysunek 6 prezentuje wyniki symulacji dla maksymalnego ciśnienia w cylindrze oraz skumulowanych emisji CO₂, CO i uHC w trakcie 1800-sekundowego cyklu WLTP, przy użyciu zarówno frameworków MiL, jak i HiL.



RYСУNEK 5: Widok pulpitu symulacji MiL/HiL w dSPACE ConfigurationDesk [6, 22].

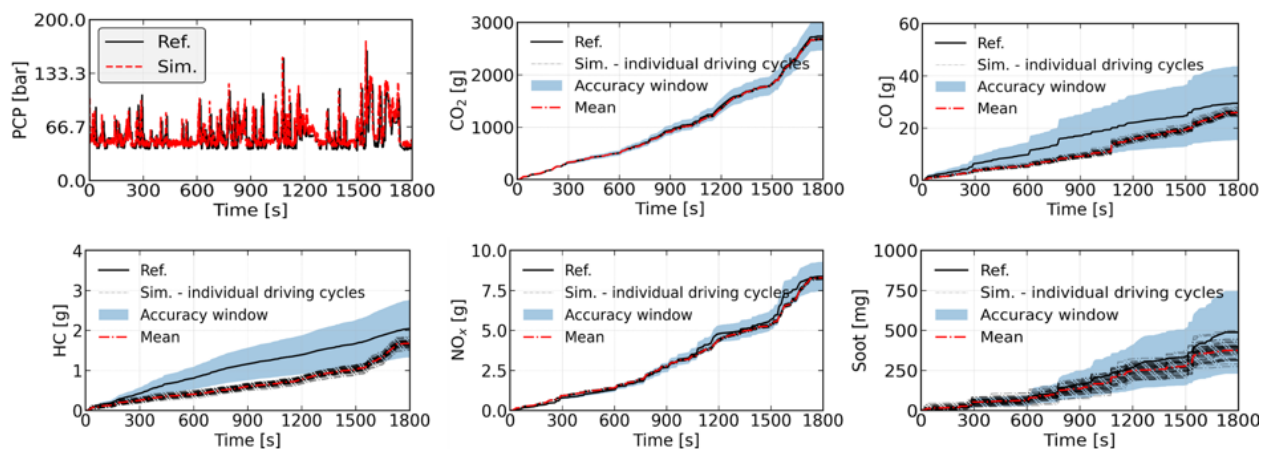


RYSUNEK 6: Wyniki symulacji MiL i HiL wykorzystujących model SRM.

Wyniki potwierdzają elastyczność i niezawodność opracowanego łańcucha narzędzi w różnych środowiskach symulacyjnych, przy czym obserwuje się spójne wyniki między MiL i HiL. Niewielkie rozbieżności w skumulowanych emisjach przypisuje się stochastycznej zmienności pomiędzy cyklami, która prowadzi do różnic w wynikach między kolejnymi cyklami jazdy; w niniejszej symulacji przeprowadzono pojedynczą realizację cyklu jazdy.

3.2 Symulacje silnika CI w cyklu jazdy WLTP

Opracowany łańcuch narzędzi został zastosowany do silników CI w celu symulacji zmienności emisji spalin pomiędzy kolejnymi cyklami jazdy, obejmującej NO_x , sadzę, niespalone węglowodory (HC) oraz CO w silniku zasilanym olejem napędowym.



RYSUNEK 7: Maksymalne ciśnienie w cylindrze oraz skumulowane przebiegi emisji spalin z kolejnych pięćdziesięciu cykli jazdy oraz ich średni przebieg w porównaniu z danymi referencyjnymi, przedstawione w oknie dokładności pomiaru stanowiska testowego.

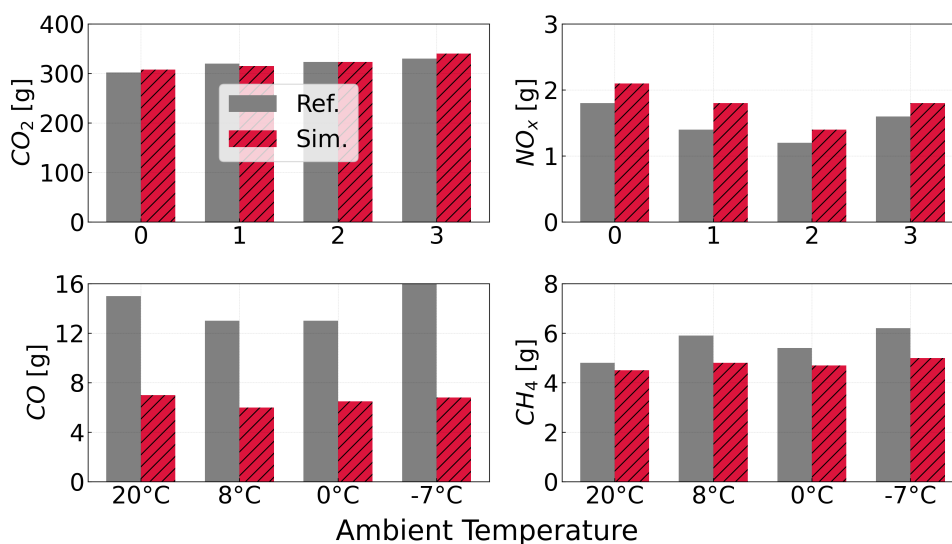
Model wytrenowany w warunkach ustalonych został wykorzystany do symulacji przejściowego zachowania silnika w cyklu WLTP. Metryki spalania wykazują dobrą zgodność z danymi referencyjnymi. Trendy emisji zostały dobrze odwzorowane. Wartości CO_2 i CO są zbliżone do wyników eksperymentalnych. Podobnie, NO_x jest przewidywane z dużą dokładnością. Niektóre rozbieżności między danymi symulowanymi a referencyjnymi przypisuje się efektom aktywacji

niskociśnieniowego EGR w tych fazach cyklu. HC i sadza są niedoszacowane w większości cyklu, jednak ich skumulowane wartości końcowe mieszczą się w ustalonych oknach dokładności opartych na wynikach przedstawionych w [23].

Aby ilościowo określić stochastyczną zmienność wyników uzyskanych w symulacjach przejściowych, przeprowadzono 50 kolejnych symulacji cyklu WLTP przy identycznych danych silnika. Wyniki pokazują, że stabilną średnią dla CO_2 , CO i NO_x można osiągnąć przy 15 symulacjach, podczas gdy dla HC i sadzy wymagana jest liczba odpowiednio 25 i 35 symulacji. Symulowane zanieczyszczenia mieszczą się w eksperymentalnie określonych oknach dokładności, potwierdzając niezawodność modelu w przewidywaniu emisji w warunkach przejściowych w porównaniu do wyników eksperymentalnych. Czas symulacji wyniósł 30 minut dla 30-minutowego cyklu jazdy, co dowodzi potencjału zastosowania tej metody w czasie rzeczywistym do przewidywania emisji oraz wirtualnej kalibracji.

3.3 Symulacja rozruchu na zimno silnika CNG w cyklach WLTP i NRTC

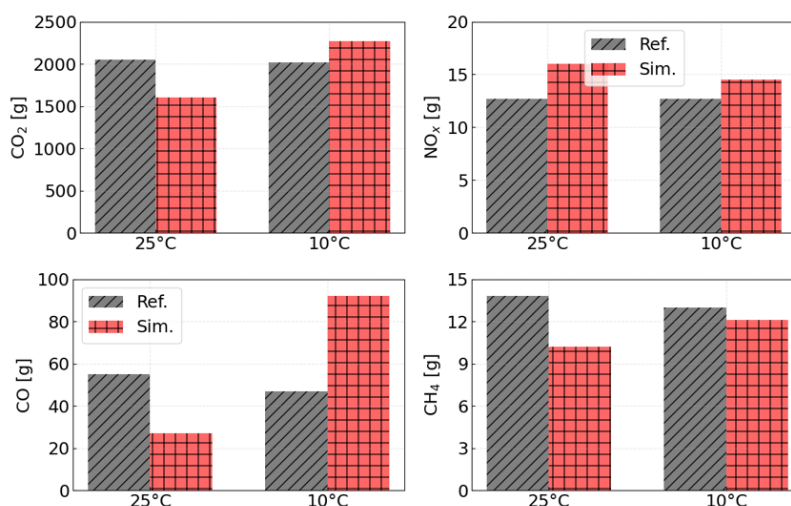
Opracowany łańcuch narzędzi został zastosowany do silników SI w celu analizy zachowania silnika zasilanego CNG podczas fazy rozruchu zimnego w cyklach jazdy: WLTP dla samochodów osobowych oraz NRTC dla zastosowań w ciągnikach rolniczych. Badania obejmowały analizę pracy silnika przy zmiennych temperaturach otoczenia, składach paliwa oraz współczynnikach nadmiaru powietrza (air-fuel ratio).



RYSUNEK 8: Symulowane i referencyjne emisje przy EVO oraz dla różnych temperatur otoczenia w cyklu WLTP.

Ogólnie model odwzorowuje trend emisji CO_2 , NO_x i CO dla różnych temperatur otoczenia w cyklu WLTP (Rysunek 8). W przypadku CH_4 model wykazuje niewielki wzrost emisji przy obniżających się temperaturach otoczenia, jednak nie jest w stanie dokładnie odtworzyć trendu danych eksperymentalnych. Wprowadzenie efektu gaszenia płomienia przy ścianach (flame-wall quenching) mogłoby poprawić przewidywalność modelu silnika przy niskich temperaturach otoczenia.

Podobne badania przeprowadzono dla zastosowań off-road w cyklu NRTC (Rysunek 9). Wzrost temperatury otoczenia z 10°C do 25°C skutkował zmniejszeniem całkowitej emisji CO_2 . Przewidywana skumulowana emisja NO_x wzrasta wraz ze wzrostem temperatury otoczenia. Skumulowana



RYSUNEK 9: Porównanie skumulowanych emisji CO₂, CO, NO_x i CH₄ z eksperymentu i symulacji dla przejściowego rozruchu na zimno przy temperaturach otoczenia 25°C i 10°C w cyklu NRTC.

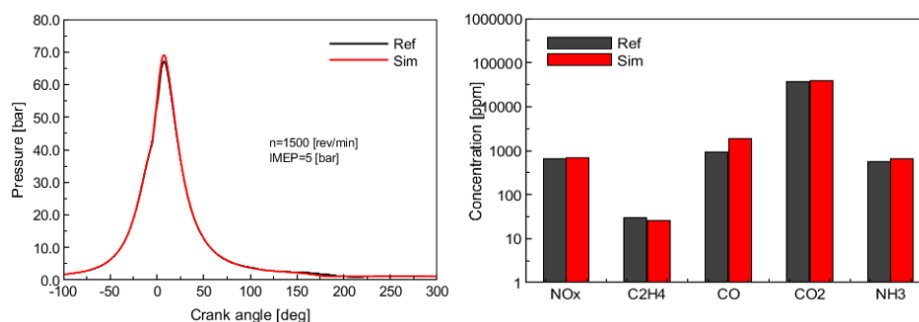
przewidywana emisja CO maleje wraz z obniżeniem temperatury otoczenia. Skumulowana przewidywana emisja CH₄ zmniejsza się wraz ze wzrostem temperatury otoczenia z 10°C do 25°C, ponieważ przy wyższej temperaturze rośnie zużycie paliwa, co powoduje zmniejszenie ilości paliwa pozostającego w spalinach.

3.4 Łańcuch narzędzi symulacyjnych z układami oczyszczania spalin

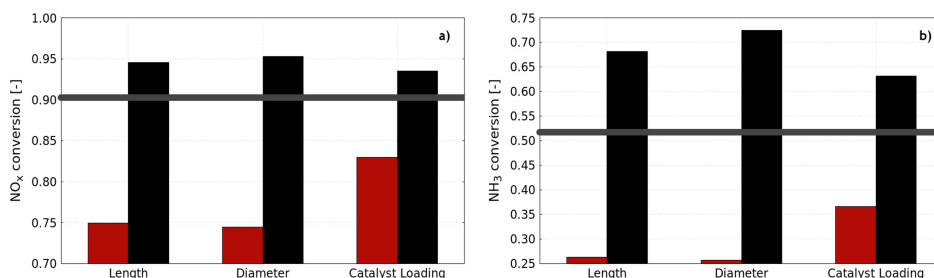
Możliwość współsymulacji opracowanego łańcucha narzędzi z systemami post-processingowymi została zweryfikowana na podstawie sprzężenia z modelem SCR zastosowanym do symulacji silnika zasilanego amoniakiem i biodieslem, tworząc wirtualne stanowisko testowe. W szczególności praca koncentrowała się na numerycznym badaniu wpływu właściwości katalizatora SCR na redukcję amoniaku w warunkach pracy silnika. System post-processingowy został zamodelowany przy użyciu jednowymiarowego modelu reaktora katalitycznego (1D Catalytic Reactor Model, CRM) reprezentującego katalizator SCR [24] i wykorzystującego szczegółową kinetykę reakcji powierzchniowych istotną do analizy SCR. Technologia FMI została użyta do skompilowania zarówno modelu silnika opartego na SRM, jak i CRM w postaci FMU. FMU zostały połączone w MATLAB/Simulink. W każdym kroku czasowym symulacji SRM przekazuje do CRM ciśnienie spalin, temperaturę, masowy strumień przepływu oraz stężenia gatunków takich jak NO, NO_x, NH₃, CO, CO₂, H₂, H₂O, C₃H₆. CRM zwraca ciśnienie i skład gazów na wylocie, jeśli wymagane jest działanie w pętli zamkniętej.

SRM działający w ramach współsymulacyjnego łańcucha narzędzi odwzorował ciśnienie w cylindrze z dobrą zgodnością z danymi eksperymentalnymi. Podobną zgodność zaobserwowano dla emisji spalin (Rysunek 10). Symulacja SCR przewidywała bazową konwersję NO_x na poziomie 90,1% oraz konwersję NH₃ wynoszącą 51,6% przy danej temperaturze na wlocie (550 K) i składzie gazów. Wartości te były zgodne z trendami literaturowymi dla katalizatorów Fe-ZSM-5 [25]. Aby zilustrować elastyczność łańcucha narzędzi, przeprowadzono krótką analizę wrażliwości poprzez zmianę długości katalizatora, jego średnicy oraz załadunku katalizatora. Zwiększenie długości katalizatora o 50% spowodowało wzrost konwersji NO_x do 94,6%, podczas gdy jej zmniejszenie o 30% znacząco obniżyło konwersję. Podobne zależności zaobserwowano dla średnicy katalizatora — zwiększenie o 30% podniosło konwersję NO_x o kilka punktów procentowych, a zmniejszenie o 30% znacząco ją obniżyło. Zwiększenie załadunku katalizatora o 30% poprawiło konwersję NO_x.

o 3,8%, natomiast jej zmniejszenie o 30% obniżyło sprawność o 8%. Choć redukcja NO_x była skuteczna, zaobserwowano istotny „slip” amoniaku, co przypisuje się częściowo konkurencyjnym reakcjom z udziałem CO i niespalonych węglowodorów w spalinach. W praktyce można to ograniczyć, integrując katalizator do redukcji amoniaku (ammonia slip catalyst) za katalizatorem SCR.



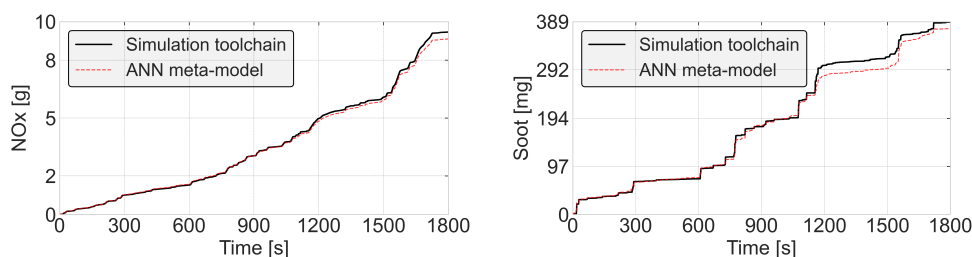
RYSUNEK 10: Przykładowa symulacja ciśnienia w cylindrze i emisja spalin porównane z pomiarami w wybranym punkcie pracy [14].



RYSUNEK 11: Wpływ długości, średnicy i zmiany wypełnienia katalizatora na konwersję NO_x i NH_3 [14].

3.5 Symulacje cyklu jazdy oparte na sztucznej sieci neuronowej (ANN)

W końcowym etapie przedstawionej pracy opracowano meta-model symulatora cyklu jazdy oparty na feedforward ANN, który pełni rolę modelu zastępczego (surrogate model), działającego w czasie szybszym niż rzeczywisty i osadzonego w ramach łańcucha narzędzi. Ocena wydajności wykazała bardzo dobrą zgodność między predykcjami ANN a wynikami SRM. Miary błędów, przy porównaniu wyników opartych na ANN z wynikami źródłowymi SRM, były poniżej 5% dla wszystkich metryk spalania i emisji łącznie.



RYSUNEK 12: Porównanie wyników uzyskanych za pomocą modelu ANN i modelu fizycznego wykorzystującego SRM.

Należy zaznaczyć, że w obecnym schemacie pracy model oparty na ANN jest trenowany przy użyciu wyników symulacji cyklu jazdy wykorzystujących model fizyczne SRM. W związku z tym ANN na tym etapie nie może działać niezależnie od SRM. Model oparty na ANN mógłby wspierać symulacje mające na celu testowanie innych podmodeli oraz ich algorytmów sterowania, jeśli mogłyby one być symulowane szybciej niż w czasie rzeczywistym i bez konieczności zmian w modelowaniu procesów zachodzących w cylindrze. W takich warunkach ANN, po treningu, mogłaby wyeliminować konieczność ponownego uruchamiania pełnego łańcucha narzędzi symulacyjnych, oszczędzając zasoby obliczeniowe — na przykład symulacja pełnego cyklu WLTP przy użyciu ANN trwa poniżej 10 sekund. Niemniej jednak, ze względu na wspomniane ograniczenia, obecnie to podejście modelowania jest uznawane za wstępne i wymaga dalszego rozwoju oraz ulepszeń.

4 Podsumowanie

Praca przedstawia framework do symulacji w czasie rzeczywistym parametrów pracy silnika, emisji spalin i efektów paliwa w silnikach z zapłonem samoczynnym (ZS) oraz zapłonem iskrowym (ZI) w przejściowych warunkach rzeczywistych cykli jazdy, takich jak WLTP i NRTC. Framework stanowi zunifikowany łańcuch narzędzi, który integruje 0D SRM ze szczegółową kinetyką chemiczną w formie tabelarycznej dla spalania i tworzenia zanieczyszczeń w platformach MiL i HiL przy użyciu standardu FMI/FMU. Takie rozwiązania są interesujące dla przemysłu motoryzacyjnego, ponieważ redukują koszty i czas eksperymentalnych badań i prototypowania. Główny nacisk położono na umożliwienie symulacji przejściowych w czasie rzeczywistym poprzez integrację 0D SRM w standardowych środowiskach współsymulacji MiL/HiL oraz symulacjach międzyplatformowych, co wcześniej nie było dostępne. Pozwala to połączyć dokładność modelowania fizycznego z praktycznym zastosowaniem w rozwoju silników.

Istniejący SRM został dostosowany do ciągłych zmian parametrów pracy silnika, takich jak prędkość, obciążenie, strategia wtrysku i fazy rozrządu. Umożliwiło to przejściowe symulacje oraz integrację oprogramowania wyższego poziomu wspierającego sterowanie w pętli zamkniętej i symulacje systemowe. Opracowano wrapper FMI/FMU zgodny ze standardem FMI 2.0 dla współpracy z legacy SRM w Fortranie i platformami MATLAB/Simulink, dSPACE ASM, VEOS i SCALEXIO, zapewniający tryb czasu rzeczywistego, synchronizowaną wymianę danych i standardyzację portów poprzez plik XML. Framework został rozszerzony o sprzężenie procesów silnika z modelami aftertreatment do współsymulacji, a także o meta-model ANN, wytrenowany na wynikach SRM, zapewniający bardzo szybkie symulacje zastępcze.

Łańcuch narzędzi zastosowano dla silników ZS oraz ZI w przejściowych cyklach WLTP i NRTC. Dla ZS przewidziano zmienność emisji NO_x , sadzy, HC i CO między cyklami, pokazując potrzebę 15–35 powtórzeń dla stabilności statystycznej i zgodności z danymi eksperymentalnymi. Dla ZI analizowano rozruch zimny silnika CNG, wykazując silną zależność emisji od temperatury, składu paliwa i air-fuel ratio. Niższe temperatury zwiększały CO_2 i CH_4 , natomiast wzbogacenie H_2 umożliwiało spełnienie limitów CO_2 Euro 6. Podobne trendy zaobserwowano w silniku CNG ciężkiego sprzętu rolniczego w NRTC.

Współsymulacja z układami oczyszczania spalin została zademonstrowana na modelu SCR sprzężonym z silnikiem amoniak-biodiesel, tworząc wirtualne stanowisko testowe. Analiza wrażliwości katalizatora wykazała jego wpływ na konwersję NO_x i wskazała obszary wymagające poprawy w zakresie ammonia slip. Modułowa struktura umożliwia zastąpienie SCR innymi modelami, wspierając digital twin i zintegrowaną optymalizację układu silnik-aftertreatment.

Przydatność w MiL/HiL zweryfikowano na platformie dSPACE dla silnika benzynowego w cyklu WLTP. Symulowane przebiegi ciśnienia w cylindrze i emisje były zgodne w obu podejściach, przy zachowaniu czasu rzeczywistego, potwierdzając poprawność integracji. Meta-model ANN

odtworzył wyniki referencyjne z odchyleniem poniżej 5% i znacząco przyspieszył czas symulacji. W obecnej formie zastosowanie metamodelu wymaga wcześniejszych symulacji SRM jako bazy.

Podsumowując, praca demonstruje solidny, skalowalny framework do predykcyjnych symulacji w czasie rzeczywistym silników ZI oraz ZS w warunkach przejściowych cykli jezdnych. Integracja FMI/FMU umożliwia płynną pracę w MiL/HiL, sprzężenie z modelami aftertreatment oraz ma potencjał zastosowania jako digital twin, stanowiąc cenne narzędzie do badań i rozwoju w zastosowaniu do przemysłu motoryzacyjnego.

Bibliografia

- [1] European Union. Worldwide harmonised Light-duty vehicles Test Procedure (WLTP) and Real Driving Emissions (RDE), 2017. EUR-Lex.
- [2] TransportPolicy.net. International: Nonroad: Nrtc test. <https://www.transportpolicy.net/standard/international-nonroad-nrtc-test/>, n.d. Accessed July 2025.
- [3] Paul D. Walker. Model-based development and calibration of automotive engine control systems. *Control Engineering Practice*, 19(9):962–969, 2011.
- [4] Tianjun Zheng, Amir Khajepour, Shuaibing Chen, and Bakhtiar Litkouhi. A model-in-the-loop simulation framework for vehicle control system development. *IEEE Transactions on Vehicular Technology*, 67(5):3885–3899, 2018.
- [5] Werner Damm, Bernhard Josko, Jens Oehlerking, and Thomas Peikenkamp. Hardware-in-the-loop and real-time simulation for automotive applications. *ATZ Worldwide*, 110(9):28–33, 2008.
- [6] dSPACE. Fmi for pc-based simulation, dspace, 2024.
- [7] Tim Franken, Andrea Matrisciano, Rafael Sari, Álvaro Fogué Robles, Javier Monsalve-Serrano, Dario Lopez Pintor, Michal Pasternak, Antonio Garcia, and Fabian Mauss. Modeling of reactivity controlled compression ignition combustion using a stochastic reactor model coupled with detailed chemistry. In *SAE Technical Paper Series*, ICE2021. SAE International, September 2021.
- [8] Michal Pasternak, Fabian Mauss, Christian Klauer, and Andrea Matrisciano. Diesel engine performance mapping using a parametrized mixing time model. *International Journal of Engine Research*, 19(2):202–213, July 2017.
- [9] Michal Pasternak, Fabian Mauss, Cathleen Perlman, and Harry Lehtiniemi. Aspects of 0d and 3d modeling of soot formation for diesel engines. *Combustion Science and Technology*, 186(10–11):1517–1535, September 2014.
- [10] Corinna Netzer, Michal Pasternak, Lars Seidel, Frédéric Ravet, and Fabian Mauss. Computationally efficient prediction of cycle-to-cycle variations in spark-ignition engines. *International Journal of Engine Research*, 21(4):649–663, June 2019.
- [11] Andrea Matrisciano, Tim Franken, Laura Catalina Gonzales Mestre, Anders Borg, and Fabian Mauss. Development of a efficient tabulated chemistry solver for internal combustion engine optimization using stochastic reactor models. *Applied Sciences*, 10(24):8979, December 2020.
- [12] Alberto Falcone and Alfredo Garro. Distributed co-simulation of complex engineered systems by combining the high level architecture and functional mock-up interface. *Simulation Modelling Practice and Theory*, 8 2019.
- [13] Edmund Widl, Florian Judex, Katharina Eder, and Peter Palensky. Fmi-based co-simulation of hybrid closed-loop control system models. In *2015 International Conference on Complex Systems Engineering (ICCSE)*, page 1–6. IEEE, November 2015.
- [14] Reddy Babu Siddareddy, Michał Pasternak, Larisa León de Syniawa, Vivien Guenther, Lars Seidel, Fabian Mauss, Grzegorz Przybyła, and Wojciech Adamczyk. Simulations of the scr catalyst in ammonia-biodiesel fuelled ci engine using virtual test bench with detailed chemistry. *Renewable Energy*, 251:123169, October 2025.

-
- [15] Reddy Babu Siddareddy, Tim Franken, Michal Pasternak, Larisa Leon de Syniawa, Johannes Oder, Hermann Rottengruber, and Fabian Mauss. Real-time simulation of cng engine and after-treatment system cold start part 1: Transient engine-out emission prediction using a stochastic reactor model. *SAE International Journal of Advances and Current Practices in Mobility*, 5(6):2421–2443, April 2023.
- [16] François Chollet. Keras. <https://keras.io>, 2015.
- [17] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, et al. Tensorflow: Large-scale machine learning on heterogeneous systems. <https://www.tensorflow.org/>, 2015. Software available from tensorflow.org.
- [18] Fabian Pedregosa, Gael Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, and Olivier Grisel. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- [19] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014.
- [20] Reddy Babu Siddareddy, Tim Franken, Larisa Leon de Syniawa, Michal Pasternak, Sascha Prehn, Bert Buchholz, and Fabian Mauss. Simulation of cng engine in agriculture vehicles. part 1: Prediction of cold start engine-out emissions using tabulated chemistry and stochastic reactor model. In *SAE Technical Paper Series*, ICE2023. SAE International, August 2023.
- [21] Mario Picerno, Sung-Yong Lee, Michal Pasternak, Reddy Siddareddy, Tim Franken, Fabian Mauss, and Jakob Andert. Real-time emission prediction with detailed chemistry under transient conditions for hardware-in-the-loop simulations. *Energies*, 15:261, 12 2022.
- [22] LOGE AB. LOGEsoft, LOGEengine, LOGEcat. <http://www.logesoft.com>, 2024.
- [23] Mario Picerno, Sung-Yong Lee, Joschka Schaub, Markus Ehrly, Federico Millo, Mauro Scassa, and Jakob Andert. Co-simulation of multi-domain engine and its integrated control for transient driving cycles. *IFAC PapersOnLine*, 53-2:13982–13987, 12 2020.
- [24] Larisa Leon de Syniawa, Reddy Babu Siddareddy, Johannes Oder, Tim Franken, Vivien Guenther, Hermann Rottengruber, and Fabian Mauss. Real-time simulation of cng engine and after-treatment system cold start. part 2: Tail-pipe emissions prediction using a detailed chemistry based moc model. *SAE Technical Paper Series*, 1, 2023.
- [25] Larisa Leon de Syniawa, Reddy Babu Siddareddy, Johannes Oder, Tim Franken, Vivien Guenther, Hermann Rottengruber, and Fabian Mauss. Real-time simulation of cng engine and after-treatment system cold start part 2: Tail-pipe emissions prediction using a detailed chemistry based moc model, 4 2023.