

Metody kreacji nowoczesnego interfejsu graficznego dla kodu symulacji elektromagnetycznej FDTD

Streszczenie. W artykule przedstawiono dwie koncepcje zrealizowanego praktycznie interfejsu użytkownika dla elektromagnetycznego kodu obliczeniowego FDTD. Interfejsu, który może konkurować z najlepszymi programami tego typu na współczesnym, globalnym rynku.

Abstract. Two concepts of the modern GUI realization in practice for FDTD electromagnetic simulation code are presented. Such interface solutions can compete with the best CAD programs on contemporary global market. (**Modern GUI creation for FDTD electromagnetic simulation code**).

Słowa kluczowe: interfejs graficzny, GUI, symulacje elektromagnetyczne, metoda FDTD, CAD, formaty danych CAD.

Keywords: graphical interface, GUI, electromagnetic simulation, FDTD method, CAD, CAD file formats.

Wprowadzenie

O powodzeniu rynkowym innowacyjnego kodu obliczeniowego symulatora elektromagnetycznego decyduje wiele czynników. Jednym z ważniejszych w kategorii składników technicznych jest interfejs graficzny (GUI - **G**raphical **U**ser **I**nterface). Projektant układów wysokich częstotliwości wymagających pełno falowej analizy elektromagnetycznej zwykle rozpoczyna pracę od zdefiniowania struktury geometrycznej i wprowadzenia danych tej struktury do programu umożliwiającego późniejszą symulację elektromagnetyczną (EM). Parametryzacja struktury geometrycznej jest bardzo istotna, gdyż proces projektowania zwykle nie kończy się na pojedynczej symulacji, ale wymaga wielokrotnych obliczeń w celu optymalizacji, badań wrażliwości zmiany wyników obliczeń na zmiany parametrów czy nawet dużych modyfikacji związanych ze zmianami koncepcji działania układu. Z drugiej strony dla inżyniera w.cz. czynności związane z definiowaniem i wprowadzaniem geometrii układu stanowią działalność drugoplanową wymaganą do otrzymania wyników elektrycznych. Inżynier w.cz. zwykle nie jest mechanikiem i dlatego w programie przeznaczonym do symulacji definiuje geometrię układu elektrycznego intuicyjnie, możliwie prosto, ucząc się stopniowo jak realizować to optymalnie i w najkrótszym czasie.

Najnowocześniejsze, współcześnie używane programy CAD (**C**omputer **A**ided **D**esign), umożliwiające definiowanie struktury geometrycznej, rozwijane są od kilkunastu lat, a ich „silniki” graficzne od ponad 30 lat. Stworzenie od podstaw kompletnego oprogramowania CAD wraz z jądrem graficznym najnowszej generacji zapewniającym odpowiedni poziom niezawodności nie wydaje się być realne. W artykule przedstawiono alternatywne możliwości wykorzystania istniejących już narzędzi software’owych do stworzenia elektromagnetycznego środowiska symulacyjnego z interfejsem użytkownika, który można obecnie zaliczyć do najnowocześniejszych zarówno od strony użytkownika jak i od strony programistycznej.

Współczesne systemy CAD

Systemy CAD rozwijane są od wielu lat. Za pierwszy system z graficznym interfejsem użytkownika, będącym podstawą systemu CAD, uznaje się zwykle SketchPAD [1] z 1963 roku. Z drugiej strony systemy CAD, oprócz GUI, wymagają zastosowania modułu modelowania geometrycznego GMK (**G**eometric **M**odeling **K**ernel), który we współczesnym świecie projektowania komputerowego jest oprogramowaniem strategicznym rozwijanym od lat 80. XX wieku głównie w USA, Francji, Niemczech i Rosji [2]. Moduł ten to duża, zaawansowana programowa biblioteka

procedur, która umożliwia definiowanie, przekształcanie, przechowywanie oraz prezentację skomplikowanych figur i kształtów geometrycznych w przestrzeni trójwymiarowej. „Silniki geometryczne” takie jak ACIS [3], Parasolid [4] czy C3D [5] są dostępne na rynku i mogą być wykorzystane do opracowania własnego programu CAD. Należy jednak zauważyć, że tworzenie nowoczesnego GUI od podstaw, na bazie istniejących już modułów modelowania geometrycznego, jest ze względu na duże oczekiwania użytkowników bardzo pracochłonne i wymaga wielu lat pracy. Wieloletnia konkurencja na rynku systemów GUI doprowadziła do unifikacji możliwości, przez co użytkownicy mogą obecnie łatwiej niż wiele lat temu korzystać z programów CAD oferowanych przez różne firmy.

Systemy GUI, podobnie jak moduły modelowania geometrycznego, są najczęściej otwarte na poziomie bibliotek programowych (API – **A**pplication **P**rogramming **I**nterface) i zaawansowany użytkownik/programista może dostosować GUI do własnych potrzeb bez konieczności tworzenia całego interfejsu od początku. Większość procedur modelowania zawartych w GUI jest uniwersalna i nie zależy od docelowego przeznaczenia interfejsu. Dlatego wydaje się, że tworzenie od podstaw GUI, w którym 90%-95% stanowią operacje takie jak definiowanie i przekształcanie elementów geometrycznych 3D, obsługa plików, naprawa geometrii, tworzenie przekrojów itp. nie jest potrzebne.

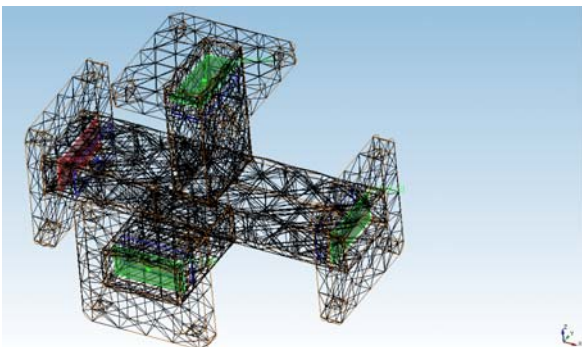
W zakresie otwartego oprogramowania dostępny jest zarówno moduł modelowania graficznego – Open Cascade [6] jak i interfejs graficzny GUI FreeCAD [7], które mogą stanowić kompletną platformę CAD, dzięki której pracochłonność wykonania systemu np. do projektowania elektromagnetycznego, przy zachowaniu wysokiej jakości i bezpieczeństwa rozwiązania, będzie stosunkowo mała. Otwarte oprogramowanie ma tę zaletę, że w przypadku napotkania specyficznych problemów dotyczących modelowania (pojawiających się najczęściej w przypadku dużych, skomplikowanych modeli) istnieje szansa rozwiązania ich we własnym zakresie dzięki dostępności kodów źródłowych i możliwości dokonania odpowiednich poprawek.

Systemy CAD do modelowania elektromagnetycznego wysokich częstotliwości

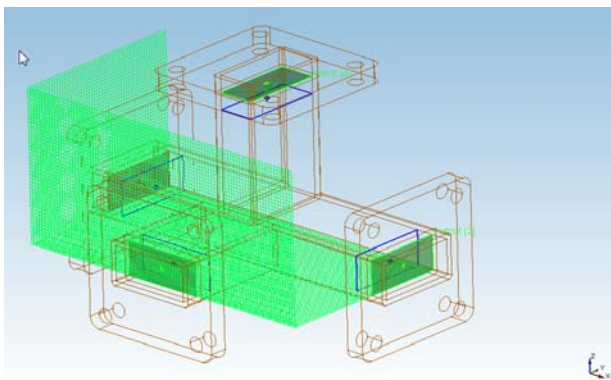
Dyskretyzacja wprowadzonego modelu geometrycznego stanowi ważny etap wstępnego modelowania elektromagnetycznego. Sposób dyskretyzacji zależy od metody obliczeniowej. W przypadku metody elementów skończonych FEM (**F**inite **E**lement **M**ethod – dyskretyzacja za pomocą czworościanów – rys. 1a) dostępność oprogramowania dyskretyzacji struktur geometrycznych jest znacznie większa niż ma to miejsce dla metody FDTD (**F**inite **D**ifference **T**ime **D**omain – dyskretyzacja za pomocą

prostopadłościanów rys. 1b). Wynika to z faktu, że metoda FEM jest stosowana znacznie dłużej, szczególnie w zakresie obliczeń wytrzymałości konstrukcji mechanicznych.

Zarówno oprogramowanie komercyjne (np. Catia®, SolidWorks®, Autodesk Inventor®) jak i otwarte (np. FreeCAD) umożliwiają generację siatki i dyskretyzację dla celów symulacji metodą elementów skończonych.



Rys. 1a. Siatka FEM



Rys. 1b. Siatka FDTD

W większości przypadków moduły FEM są stosowane przy obliczeniach dla konstrukcji mechanicznych. Jednakże dostępne są też symulatory HF EM (High Frequency ElectroMagnetic) np. HFWorks® [8], który jest symulatorem typu EM FEM wbudowanym w CAD SolidWorks®. Niektóre popularne symulatory elektromagnetyczne, takie jak QuickWave, CST Microwave Studio® czy ANSYS HFSS®, wykorzystują GMK (Graphical Modeling Kernel) np. ACIS, ale GUI jest już produkowane samodzielnie. Taka strategia ma swoje zalety i wady. Główną zaletą jest możliwość dostosowania GUI do specyficznych potrzeb symulacyjnych na niskim poziomie oprogramowania. Wadą jest konieczność dostosowywania GUI do wymagań rynku CAD, gdzie konkurencja jest bardzo duża, a zespoły ludzi tworzące CAD zwykle są większe i bardziej doświadczone w modelowaniu geometrycznym niż zespoły w firmach, gdzie głównym zadaniem jest modelowanie EM.

Wykorzystanie API współczesnych systemów CAD do budowy interfejsu symulatora elektromagnetycznego FDTD

Oprogramowanie CAD, takie jak Catia®, SolidWorks®, Autodesk Inventor® czy Siemens NX®, jest oprogramowaniem otwartym w zakresie dostępności do biblioteki oprogramowania API wysokiego poziomu, co pozwala na modyfikację interfejsu programu, rozszerzenie jego możliwości, zmianę funkcjonalności czy dostosowanie do specyficznych zadań, jak ma to miejsce w HFWorks® czy w QW-AddIn for Autodesk Inventor® [9]. Zwykle nie jest dopuszczalna ingerencja w GMK (Graphical Modeling

Kernel) oprogramowania CAD. Przystępując do wykorzystania API komercyjnego programu CAD niewątpliwie warto jest się zapoznać z powstałymi wcześniej rozszerzeniami (Addin, Workbench), których projektantami byli producenci API i CAD. Wykorzystując rozwiązania wzorcowe oraz eliminując ich zauważalne wady, można stworzyć wysokiej klasy GUI.

W przypadku interfejsu dla symulatora EM FDTD należy zapewnić obsługę dodatkowych elementów, charakterystycznych dla danej metody. Są to m.in.:

- Materiały i ich parametry elektryczne w zakresie wysokich częstotliwości (np. parametry metamateriałów)
- Wrota wejściowe i wyjściowe układów w.cz. (dla celów obliczeń parametrów obwodowych układu)
- Powierzchnie modelujące absorpcyjne warunki brzegowe (dla obliczeń układów otwartych – np. anten)
- Powierzchnie NTF (Near To Far) transformacji strefy bliskiej do dalekiej (dla obliczeń charakterystyk promieniowania)
- Kontrola i wizualizacja dyskretyzacji/siatki FDTD.

Dodatkowo, w szczególnych przypadkach, należy rozważyć następujące nowe składniki, specyficzne dla metody FDTD:

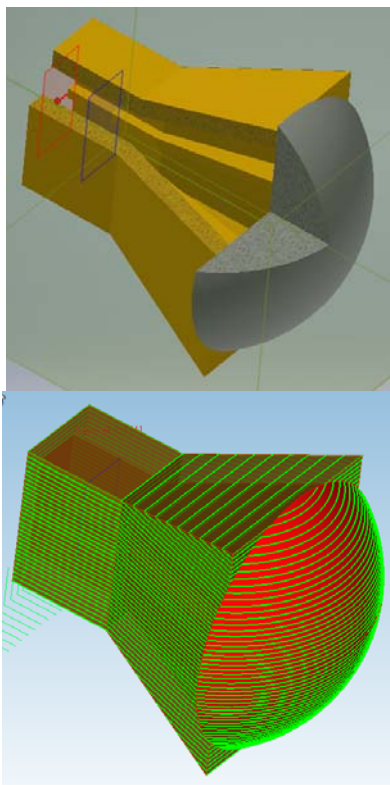
- Nieskończenie cienkie warstwy metalowe
 - Cienkie przewody metalowe
 - Płaszczyzny siatki
 - Obszary lokalnej zmiany siatki
 - Definicje transformacji elementów w przestrzeni (dla celów symulacji elementów ruchomych)
- oraz w zakresie wizualizacji wyników obliczeń:

- Kontury pola EM
 - Monitory pola EM
- W celu realizacji powyższych komponentów rozszerzeniu podlegają następujące elementy GUI:
- Zestaw nowych okien i dialogów
 - Menu, toolbar i drzewo komponentów
 - Specyficzna grafika – User graphics (np. wizualizacja siatki)
 - Komendy języka opisu projektu (np. Python)

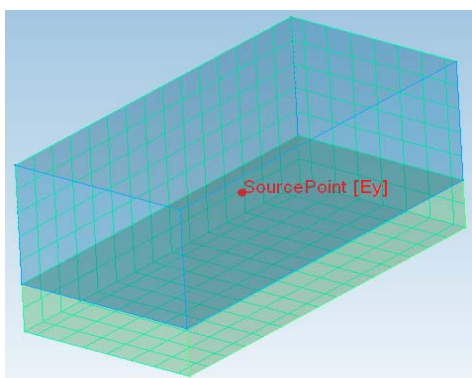
Ponadto, zmiana musi podlegać także zapis projektu do pliku i to najlepiej w taki sposób, aby część zapisanych danych, która jest specyficzna dla symulatora EM, nie zakłócała możliwości standardowego odczytu danych geometrycznych.

Podstawowym narzędziem API, służącym do przekształcenia modelu geometrycznego w dyskretny model elektryczny dla celów symulacji EM FDTD, jest możliwość wykonywania przekrojów geometrycznych – konturów (rys. 2), z zachowaniem informacji o materiałach zawartych między liniami konturu.

Każde API GMK posiada taką możliwość. Dane, uzyskane z przekrojów, stanowią podstawę do dalszej dyskretyzacji mającej na celu uzyskanie prostopadłościennej siatki (ang. structured grid/mesh) FDTD. W praktyce jednak, najczęściej w skomplikowanych geometrycznie przypadkach, zdarzają się sytuacje, gdy generacja konturu nie jest możliwa. Zwykle ujawniają się wtedy wady lub nawet błędy GMK, które nie mogą być rozwiązane przez użytkownika/programistę API. Wydaje się, że najlepszą strategią w takim przypadku jest znalezienie prawidłowego przekroju możliwie blisko poziomu, dla którego kontur nie może być wyznaczony. Doświadczenia autorów artykułu wskazują, że dużo częściej problemy związane z wyznaczeniem konturów/przekrojów występują dla modeli importowanych z innych programów CAD, co obok utrudnionej parametryzacji stanowi kolejną przyczynę, dla której użytkownicy CAD unikają modelowania z wykorzystaniem funkcji importu modeli.



Rys. 2. Generacja przekrojów modelu przeznaczonych do symulacji



```

QW_Modeller.newQWDocument("Q_cuboid1")
from FreeCAD import Base
import Part, PartGui
App.ActiveDocument.addObject("Part::Box", "Resonator")
QW_Modeller.addQWMedium("air_dens")
aktmed=QW_Modeller.getQWMedium("air_dens")
aktmed.materialtype = "Isotropic"
aktmed.Eps = 1
aktmed.Sigma = 0.00555556
App.ActiveDocument.addObject("Part::Box", "Dense")
QW_Modeller.addQWObject("QW_Modeller::PointPort",
"SourcePoint")
App.ActiveDocument.QW_Mesh.CellX = 1.00000
App.ActiveDocument.QW_Mesh.CellY = 1.00000
App.ActiveDocument.QW_Mesh.CellZ = 1.00000
App.ActiveDocument.recompute()
QW_Modeller.generateMesh("Q_cuboid1")

```

Rys. 3. Rezonator z uproszczonym opisem skryptowym w języku Python

Wykorzystanie platformy OPENSOURCE do budowy interfejsu symulatora elektromagnetycznego FDTD

Metodologia tworzenia GUI dla symulatora FDTD w oparciu o oprogramowanie otwarte typu FreeCAD/Open Cascade jest podobna do wyżej opisanego wykorzystania API oprogramowania komercyjnego. Twórca

GUI powinien być w tym przypadku przygotowany na konieczność kompilacji całego dużego programu CAD lub jego dużej części. Z doświadczeń autorów artykułu wynika, że jakość końcowego interfejsu GUI dla oprogramowania otwartego może być porównywalna z rozwiązaniami komercyjnymi, a czasami nawet je przewyższać. Przykładem może być zastosowanie we FreeCAD języka Python jako języka opisu projektu (zarówno części geometrycznej jak i elektromagnetycznej) i możliwość użycia funkcji modelera GMK w trybie tekstowym. Ze względu na pełną powtarzalność tworzenia projektu, kontrolę nad projektem i łatwą parametryzację, część użytkowników GUI stosuje „skryptowy” tryb pracy (rys. 3).

Przyszłość elektromagnetycznego CAD

Aktualnie, poza stałym wprowadzaniem ulepszeń i ułatwień w zakresie definiowania skomplikowanych modeli 3D, systemy CAD rozwijają się w dwóch nowoczesnych kierunkach:

1. W kierunku PLM (Product Lifecycle Management)
2. W kierunku projektowania przy pomocy nowych narzędzi i interfejsów z zakresu rzeczywistości rozszerzonej (augmented reality).

Pierwszy z nich, system zarządzania cyklem życia produktu, związany jest zwykle z dużymi przemysłowymi projektami, gdzie liczba elementów, podzespołów oraz modeli jest duża i wymaga kontroli wspomaganą komputerowo.

Drugi natomiast jest o tyle ciekawy, że może doprowadzić do jeszcze łatwiejszego, bardziej intuicyjnego i szybszego wprowadzania precyzyjnej informacji geometrycznej projektowanego modelu EM. Są już wstępne prezentacje [10] użycia okularów typu HoloLens firmy Microsoft w współpracy z Autodesk Fusion 360 lub Autodesk Maya 3D do modelowania geometrycznego 3D. Możliwości i narzędzia rzeczywistości rozszerzonej mogą być także wykorzystane w prezentacji wyników 3D symulacji oraz jako nowoczesne narzędzie dydaktyczne.

Podsumowanie

W niniejszej pracy przedstawiono dwa sposoby wykonania interfejsu GUI dla celów symulacji elektromagnetycznych FDTD. Obie metody bazują na oprogramowaniu rozwijanym przez specjalistów z zakresu GUI i modelowania geometrycznego. Takie podejście umożliwia zminimalizowanie wysiłku związanego z czynnościami przygotowania modelu do symulacji, a nawet powoduje polepszenie jakości interfejsu użytkownika w stosunku do innych czołowych rozwiązań rynkowych.

Autorzy: dr inż. Maciej Sypniewski, Politechnika Warszawska, Wydział Elektroniki i Technik Informacyjnych, ul. Nowowiejska 15/19, 00-665 Warszawa, E-mail: m.sypniewski@ire.pw.edu.pl; mgr inż. Janusz Rudnicki, dr inż. Marzena Olszewska-Placha, QWED Sp. z o.o., ul. Krzywickiego 12 lok.1, 02-078 Warszawa.

LITERATURA

- [1] <https://en.wikipedia.org/wiki/Sketchpad>
- [2] https://en.wikipedia.org/wiki/Geometric_modeling_kernel
- [3] <http://www.spatial.com/products/3d-acis-modeling>
- [4] https://www.plm.automation.siemens.com/en_us/products/open/parasolid/
- [5] http://ascon.net/solutions/c3d_kernel/
- [6] <http://www.opencascade.com/>
- [7] <http://www.freecadweb.org/>
- [8] <https://www.emworks.com/product/preview/2/HFWorks>
- [9] <http://www.qwed.eu>
- [10] <https://www.youtube.com/watch?v=ywXWqNdVtjs> Autodesk Maya 3D HoloLens Demo at WPC
- [11] Keysight Which Electromagnetic Simulator Should I Use? – White Paper