

Praca magisterska

*Szybkie metody znajdowania pól własnych
osiowosymetrycznych rezonatorów
mikrofalowych*

inż. Patryk Guba

Promotor: *dr inż. Bartosz Chaber*

**Politechnika
Warszawska**



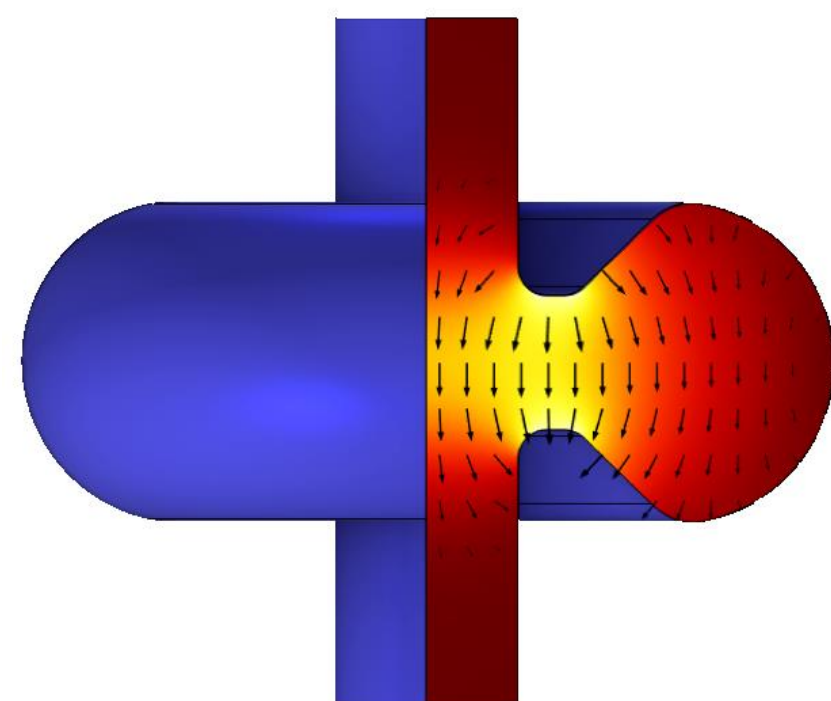
Motywacja

Wsparcie procesu projektowania wnęk rezonansowych wykorzystywanych w generatorach mikrofalowych, np. klistronach.



Cel pracy

Celem pracy jest opracowanie kodu w Julia, który przy pomocy metody elementów skończonych pozwoli na szybkie i dokładne znajdowanie pól własnych oscylatorów mikrofalowych. Zadanie polega na znalezieniu metody pozwalającej na rozwiązanie uogólnego problemu własnego (Generalized Eigenproblem) znajdując tylko wybrane pola własne we wnęce/rezonatorze. Projekt wymaga weryfikacji wyników z dostępnym oprogramowaniem komercyjnym, np. COMSOL Multiphysics.



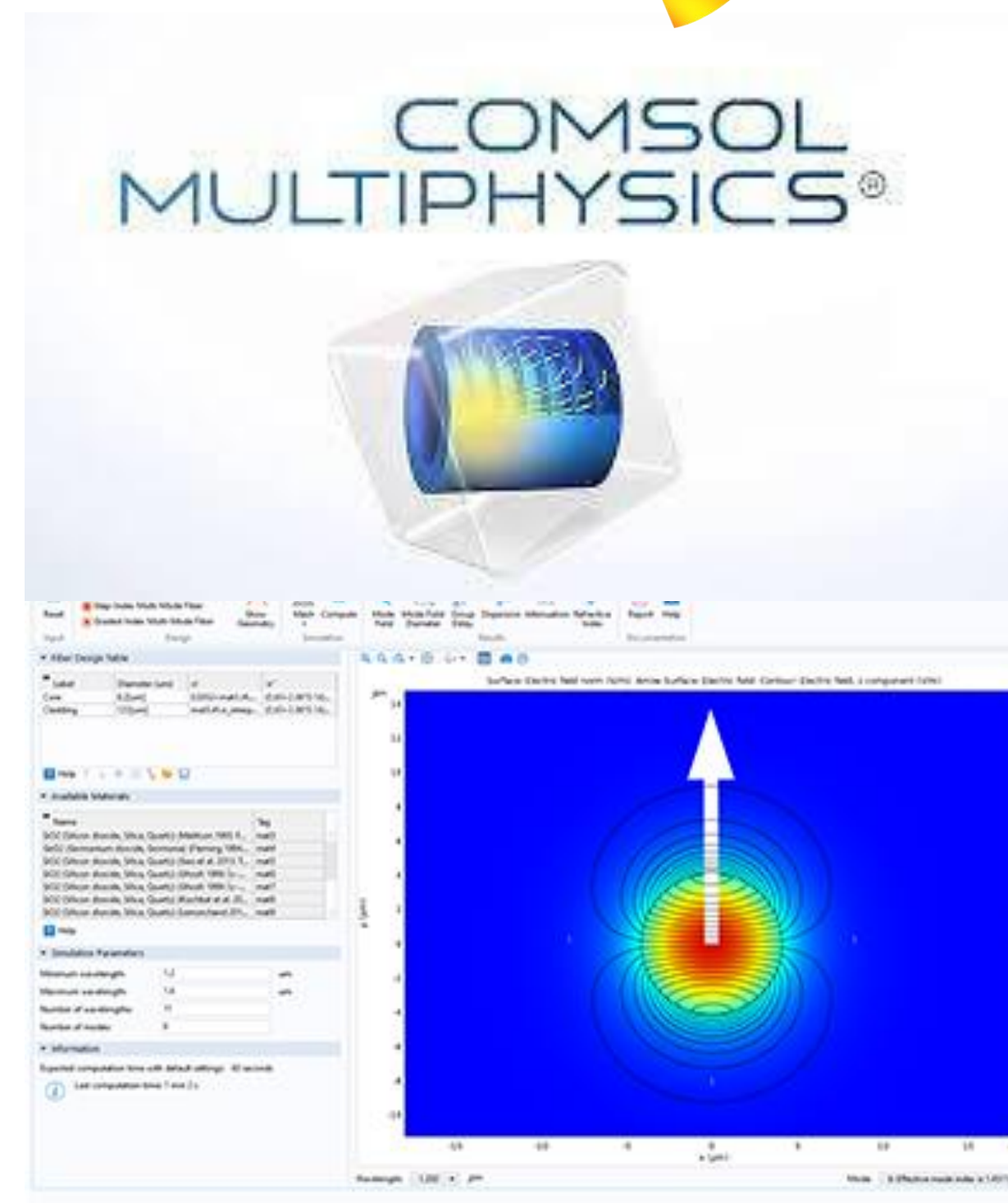
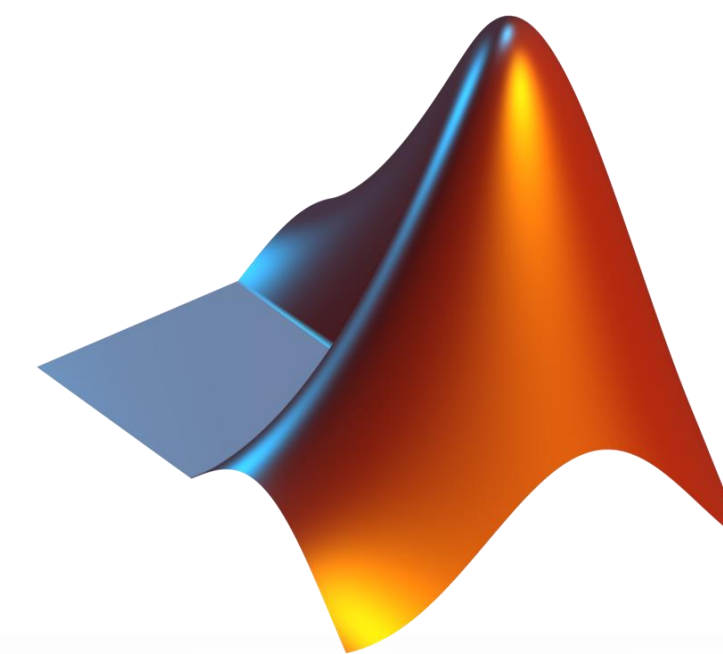
Wykorzystane metody i narzędzia

3

Do modelowania wnęki rezonansowej wykorzystana została Metoda Elementów Skończonych, do rozwiązania uogólnionego problemu własnego wykorzystane zostały metody:

- QZ
- Arnoldiego
- Kryłowa-Schura

Do uzyskania wyników referencyjnych wykorzystane zostały programy: COMSOL Multiphysics i MATLAB.



Sposób rozwiązania problemu

- Na podstawie wnęki rezonansowej zamodelowanej z wykorzystaniem Metody Elementów Skończonych otrzymujemy macierzowy układ równań:

$$[S]\{e\} = k^2[T]\{e\},$$

gdzie $[S]$ to macierz sztywności, $[T]$ to macierz mas, $\{e\}$ to wektor współczynników skojarzonych ze stopniami swobody (krawędziami siatki)

- Układ równań ma postać uogólnionego problemu własnego, do którego możemy zastosować metody numeryczne takie jak rozkład QZ, metoda Arnoldiego, metoda Kryłowa-Schura
- Ponieważ interesują nas częstotliwości w konkretnym zakresie (np. okolice 2 GHz), w wymienionych metodach stosujemy modyfikacje takie jak przesunięcie i odwrócenie, skupiając obliczenia na uzyskaniu wartości w tej części spektrum



Osiągnięte rezultaty

Tabela 1: Porównanie otrzymanych częstotliwości poszukiwanych w pobliżu wartości 2 GHz w COMSOL Multiphysics i w programie w języku Julia

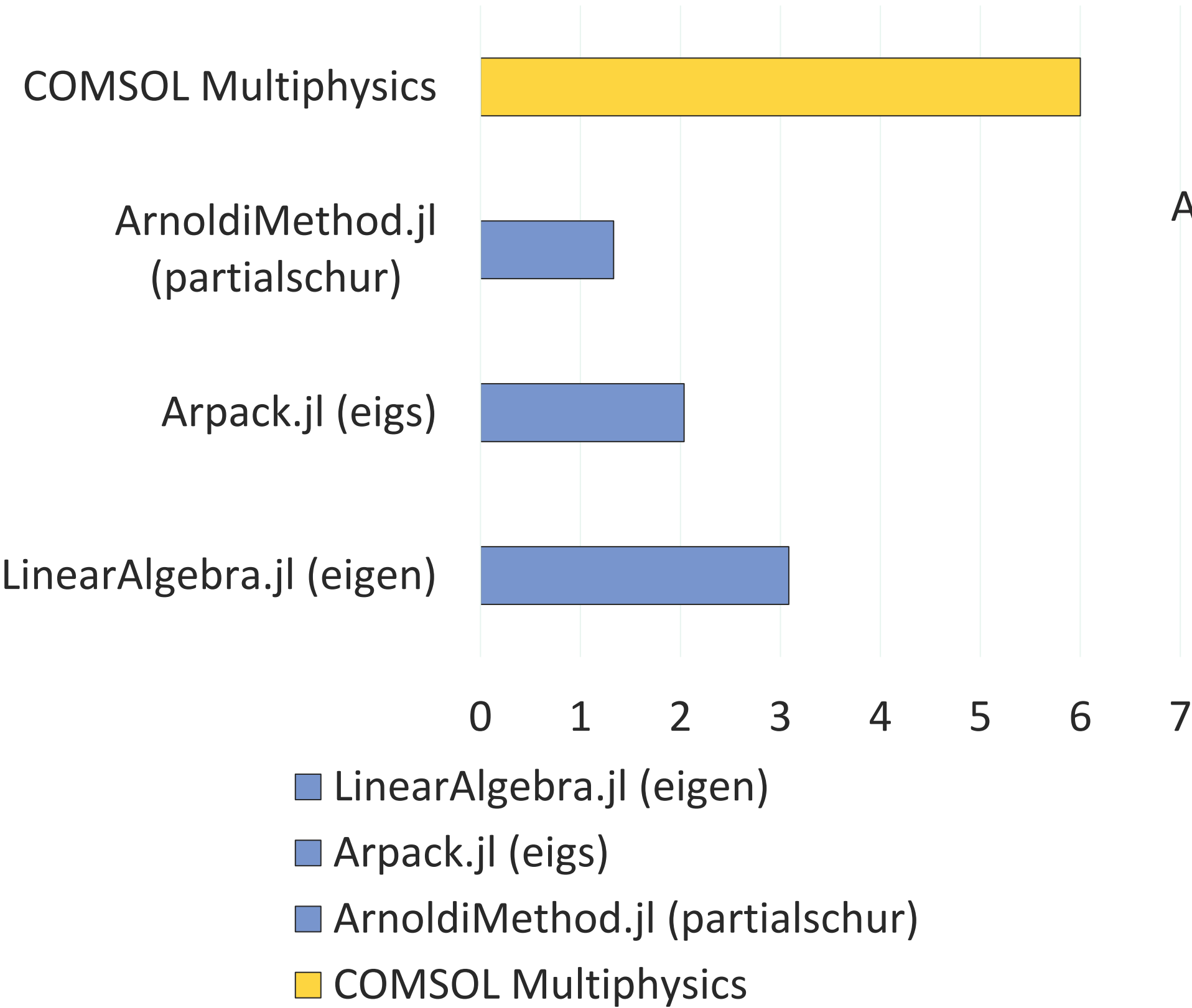
Częstotliwości w COMSOL Multiphysics [GHz]	Wartości własne w Julia	Częstotliwości w Julia [GHz]
2,000	1760,810	2,002
3,477	5323,330	3,481
3,986	8337,853	4,357
4,350	13956,172	5,637
5,628	14233,206	5,692
5,690	19942,799	6,738
5,997	24672,979	7,495
6,728	30766,389	8,369
7,485	34299,913	8,837
7,652	37886,465	9,287



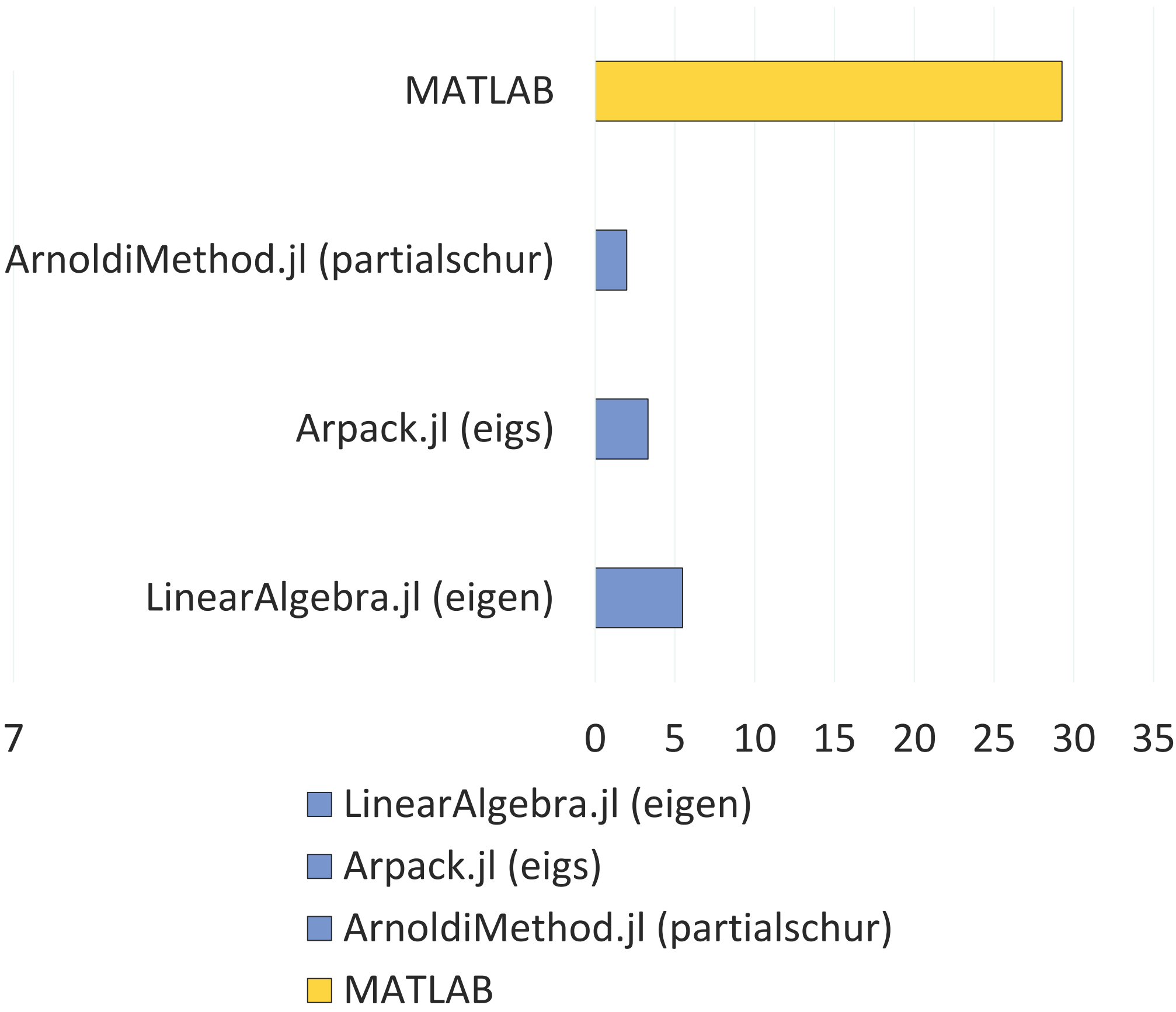
Wyniki

6

Porównanie czasu obliczeń dla problemu zamodelowanego w programie COMSOL Multiphysics i w Julii [s]



Porównanie czasu obliczeń w programie MATLAB i w Julii [s]



Podsumowanie

- Julia oferuje możliwość tworzenia zwięzłego kodu symulacji, jednocześnie oferując wydajność zbliżoną do niskopoziomowych języków programowania dzięki bibliotekom opartym na języku Fortran.
- Znajdowanie pól własnych z wykorzystaniem własnego solvera odbywa się szybciej niż w przypadku wykorzystania komercyjnego COMSOL Multiphysics, przy jednoczesnym otrzymywaniu porównywalnych wartości wyników.
- W porównaniu do komercyjnego oprogramowania MATLAB Julia oferuje także szerszy zakres ingerencji użytkownika w kod, oferując możliwość modyfikacji metod numerycznych w celu osiągnięcia lepszych rezultatów dla danego problemu.



Możliwości dalszego rozwoju

- Implementacja metody Jacobi-Davidsona QZ jako kolejnego sposobu rozwiązywania problemu własnego.
- Implementacja interfejsu GUI ułatwiającego pracę z programem.
- Dalsza optymalizacja programu poprzez implementację z wykorzystaniem bardziej dopasowanych instrukcji (np. AVX) lub wykorzystanie kart graficznych z architekturą CUDA.



Dziękuję za uwagę

