

Raport Capacitor

Mikołaj Wielebnowski Krzysztof Kubień

4 stycznia 2021

1 Wstęp

W niniejszym raporcie znajduje się opis prac przeprowadzonych w celu sporządzenia programu do symulacji potencjału pomiędzy naładowanymi płytkami. Projekt został oparty o zaproponowany podręcznik i został stworzony w celu zaliczenia projektu na przedmiot *"Wstęp do modelowania komputerowego"*.

2 Opis projektu

Etapy prac:

1. Wstęp teoretyczny.
2. Implementacja kodu zaproponowanego w opracowaniu.
3. Rozwinięcie programu- błędna metoda.
4. Właściwa implementacja metod zawartych w opracowaniu.
5. Próba rozwoju programu- układ cylindryczny.
6. Podsumowanie.

2.1 Wstęp teoretyczny.

Symulacja przykładowa w podręczniku pokazywała jak zmienia się napięcie z odległością od płytki. Naszym zadaniem było rozwinięcie tego projektu, aby symulował on napięcie występujące pomiędzy dwoma płytkami. Metoda, która została zaproponowana do wykonania symulacji to metoda relaksacji Gaussa (ang. SOR, Successive over- relation), która polega na wykonywaniu kolejnych iteracji i otrzymywanie coraz to dokładniejszego wyniku licząc średnią z sąsiadujących pól. Kluczowe dla działania tej metody jest określenie warunków brzegowych Dirichleta. Oznacza to, że musimy arbitralnie ustalić warunki brzegowe i nie mogą one ulec zmianie, skutkuje to tym, że nasza symulacja musi być "zamknięta" w pudełku o danym potencjale oraz potencjał płytek nie może ulegać zmianie (zastosowanie płytek powoduje podzielenie obszaru na osobne pudełka i liczenie ich w zasadzie jako osobne przypadki). By była możliwość liczenia w ten sposób układu wymagane jest określenie siatki, której rozmiar definiuje rozdzielczość obliczanego potencjału. Przy pomocy tej siatki powstają komórki, dla których wykonywane są obliczenia.

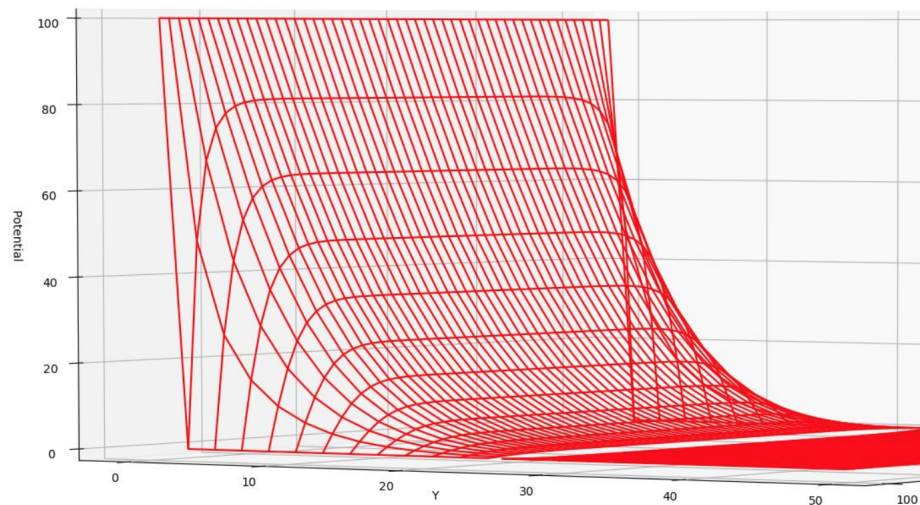
32.9987	33.9821	35.0397	36.0663	37.0254
35.4958	36.7035	38.0216	39.2377	40.3543
37.6976	39.3212	41.1125	42.5156	43.787
39.3244	41.7765	44.5972	45.9308	47.2827
39.966	43.867	49.5729	49.332	50.5429
39.6679	44.1549	60.4981	51.2838	52.1574
44.2733	32.5875	96.9815	43.1485	46.296
39.9295	44.4285	60.7826	51.5779	52.4598
40.4952	44.4205	50.1484	49.9269	51.1547
40.1307	42.6197	45.4737	46.8369	48.2144
38.7936	40.4673	42.3037	43.7467	45.0528
36.8973	38.1688	39.5443	40.8112	41.9719
34.7165	35.7862	36.914	38.0029	39.0159

Rysunek 1: Na zielono zaznaczono obszar ze stałą wartością potencjału, następnie kolorem czerwonym zaznaczono pole, którego wartość jak obliczana z pól oznaczonych na pomarańczowo. Proces ten jest powtarzany dla każdego pola, przy każdej iteracji.

Jak widać metoda ta wymaga wielu iteracji, gdyż tylko żeby dane wartości przypisać do komórki w danej odległości od zadanego warunku, należy wykonać liczbę iteracji równą liczbie pól leżących pomiędzy nimi. Dopiero następnym takim cyklem zwiększają dokładność wyniku.

2.2 Implementacja kodu zaproponowanego w opracowaniu.

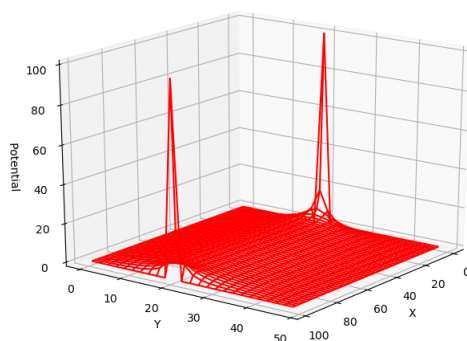
Początkowo wykorzystano część kodu, który znajdował się w przykładowym programie **LaplaceLine.py**, kod ten symulował rozkład potencjału dla pojedynczej płytki. Proponowaną metodą do wizualizacji problemu był gnuplot ze względu na możliwość obrotu otrzymanego wykresu. Metoda ta nie została zaimplementowana, ponieważ zespół napotkał trudność z implementacją kodu użytego w podręczniku. Początkowo aby manipulować wykresem zespół ręcznie wprowadzał kąt o który miał się obrócić wyświetlany wykres. Metoda ta była wystarczająca aby znajdować niedoskonałości programu, lecz wielokrotne uruchamianie programu nie było odpowiednie dla prezentacji otrzymanych wyników. W celu usprawnienia wizualizacji wykresu znaleziono sposób aby program Spyder wyświetlał w osobnym oknie wykresy którymi można było swobodnie manipulować.



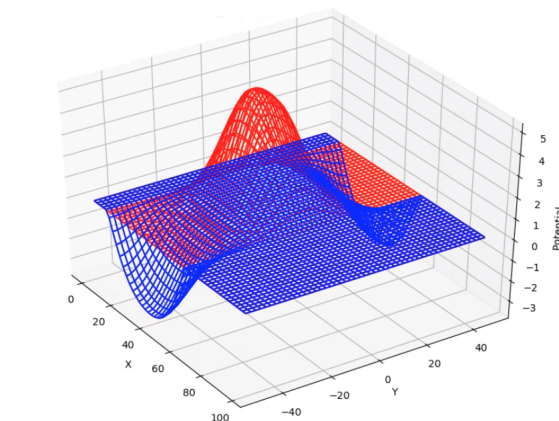
Rysunek 2: Symulacja pojedynczej naładowanej płytki.

2.3 Rozwinięcie programu- błędna metoda.

Początkowym podejściem wybranym przez zespół było wykonanie osobnych symulacji dla dwóch płytek, a następnie zsumowanie otrzymanych poencjałów i traktowanie go jako wypadkowy. Jednak okazało się, że takie podejście jest błędne i początkowo otrzymane wyniki były pozbawione jakiegokolwiek sensu. Stosując takie podejście należało zmienić granice liczonych przez program pól, gdyż naruszały one zadane warunki i w wyniku otrzymywano zanikającą płytkę, co wynika bezpośrednio z zasady działania metody.



Rysunek 3: Zanikanie płytki w wyniku naruszenia warunku brzegowego.

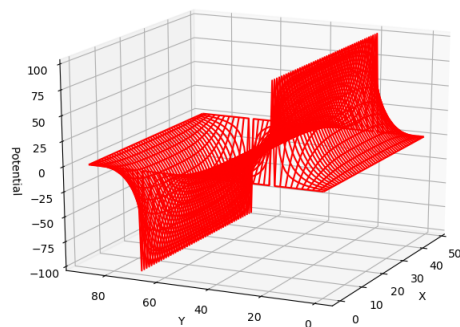


Rysunek 4: Błędny wykres otrzymany w wyniku obliczania dwóch osobnych pól.

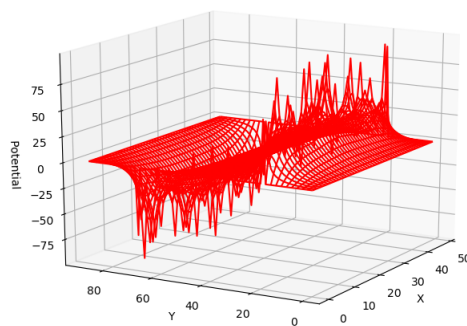
Kolejne próby modyfikacji tego podejścia skutkowały w błędnych wizualizacjach, przez co zespół doszedł do wniosku o zaczęciu pracy nad nowym programem nie bazującym na przykładzie z podręcznika.

2.4 Właściwa implementacja metod zawartych w opracowaniu.

Zespół przygotował program, który popranie liczył dwie płytki, gdzie w celu weryfikacji poprzedniego podejścia pierwsza wersja programu liczyła osobno obszary pomiędzy jak i poza płytkami, by finalnie przejść na liczenie potencjału jako całości z pominięciem warunków brzegowych (w tym płytek). Program został następnie rozwinięty o możliwość modyfikacji rozłożenia ładunku, czyli w tym wypadku rozłożenia potencjału na płytce. Aby zaprezentować taką możliwość zdecydowano o zastosowaniu losowo wybranych liczb, z przedziału określonego od 0 do napięcia, które było wybrane dla płytki o stałym rozłożeniu potencjału. Przeciwnie płytki mają potencjał o przeciwnych znakach, jednak łatwo można to zmienić.



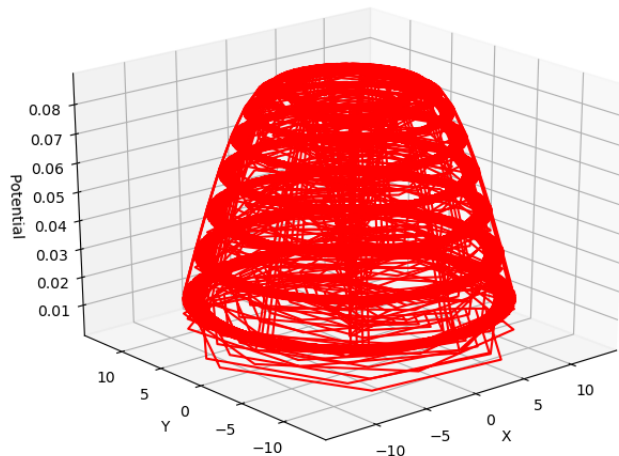
Rysunek 5: Wykres potencjału płytek o jednorodnym rozłożeniu ładunku.



Rysunek 6: Wykres potencjału płytek o niejednorodnym rozłożeniu ładunku i losowych wartościach.

2.5 Próba rozwoju programu- układ cylindryczny.

Kolejnym planowanym krokiem zespołu było przedstawienie symulacji kondensatora posiadającego dwie cylindryczne okładki. Podczas prac nad tym problemem zespół napotkał liczne problemy z konwersją z układu cylindrycznego na kartezjański dlatego podjęto decyzję o porzuceniu prac nad rozwojem programu.



Rysunek 7: Błędny wykres otrzymywany w symulacji płytek cylindrycznych.

3 Podsumowanie

Dołączony do raportu program **Capacitor.py** poprawnie symuluje układ dwóch równoległych okładek w polu. Dodatkowo przedstawia sytuację w której rozkład pola nie jest jednorodny. Dodatkowymi możliwościami rozwoju programu może być przedstawienie różnych wariantów okładek kondensatora. Dodatkowym programem **CapacitorCyl.py** dołączonym do raportu jest niedokończona wersja projektu, której na celu miało być zaimplementowanie układu cylindrycznego.

Podziękowania

Dziękujemy Panu Prof. Sebastianowi Kubisowi za udzielenie wskazówek odnośnie pracy nad programem.

Załączniki

Do raportu załączone są pliki **Capacitor.py** oraz **CapacitorCyl.py**.

Literatura

[1] /ElektrostatykaICiepło.pdf