Sprawozdanie z numerycznych metod rozwiązywania cząstkowych równań rózniczkowych z czasem

Karol Urbański

16 czerwca 2011

1 Podstawy teoretyczne

1.1 Rozwiązywany problem

Rozwiązywany problem to zadanie 1 z laboratorium, w którym znajdujemy rozwiązanie *równania dyfuzji temperatury w czasie* w przestrzeni dwuwymiarowej $(x, y) \in [0, 1]^2$ i czasie $t \in [0, t_{end}]$:

$$\frac{\delta^2 T}{\delta x^2} + \frac{\delta^2 T}{\delta y^2} - \frac{\delta T}{\delta t} = 0 \tag{1}$$

z jakimś początkowym rozkładem temperatur w czasie $t = 0$ oraz T na brzegu równym stałej T_b.

1.2 Dyskretyzacja problemu

Problem dyskretyzujemy tak samo jak problem z wcześniejszego zadania.

1.3 Zastosowanie ilorazów różnicowych do zadanego problemu

W MRS korzystamy z zastąpienia pochodnych ilorazem różnicowym korzystającym do obliczenia przybliżonej wartości pochodnej przy użyciu punktów siatki znajdujących się w bezpośrednim otoczeniu badanego punktu. W przedstawionym problemie korzystamy z drugiej różnicy centralnej dla kroku h w przestrzeni oraz pierwszej różnicy prawostronnej/lewostronnej dla kroku l w czasie. Różnica prawostronna skutkuje metodą explicite:

$$\frac{lT(x_{i-nj})}{h^2} + \frac{lT(x_{i-1j})}{h^2} + \left[1 - \frac{4l}{h^2}\right] T(x_{i-nj}) + \frac{lT(x_{i+1j})}{h^2} + \frac{lT(x_{i+nj})}{h^2} = T(x_{ij+1})$$

w której element w kroku czasowym $j+1$ wyliczamy bezpośrednio z wcześniejszego kroku. Różnica lewostronna daje metodę implicite:

$$\frac{lT(x_{i-nj})}{h^2}+\frac{lT(x_{i-1j})}{h^2}-\left[1+\frac{4l}{h^2}\right]T(x_{i-nj})+\frac{lT(x_{i+1j})}{h^2}+\frac{lT(x_{i+nj})}{h^2}=-T(x_{ij+1})$$

w której w każdym kroku musimy rozwiązać układ równań z macierzą piecioprzekątniową, co daje ogromny narzut obliczeniowy, jednak metoda ta jest znacznie dokładniejsza i bezpieczniejsza od metody explicite.[1]

2 Warstwa implementacyjna

2.1 Kernel obliczeniowy

2.1.1 Wykonanie

Kernel obliczeniowy to programy w C, przyjmujący na wejściu następujące argumenty:

1. wartość brzegowa T_b
2. ilość punktów siatki wzdłuż jednego boku
3. plik ze stanem początkowym
4. t_{end}
5. ilość kroków czasowych

2.2 Komunikacja z interfejsem graficznym

Interfejs graficzny otrzymuje dane poprzez strumień danych bezpośrednio z programu. Możliwe jest również użycie prekalkulowanych wartości podanych na wejściu interfejsu. Pierwsza linia danych podana do interfejsu to typ dawanych danych (IMPMODE jeżeli wywołujemy program w C dla metody implicit, EXPMODE dla metody explicit, PREMODE jeżeli wykorzystujemy stare obliczenia). Następna linia w EXP/IMPMODE to argumenty wywołania programu w C, a w PREMODE gotowa do wyświetlenia tablica punktów. Po nazwie trybu powinna znaleźć się liczba określająca opóźnienie wyświetlania nowego kroku (w klatkach).

[1] Dokładny sposób rozwiązania jest taki sam jak równania stacjonarnego z poprzedniego zadania, pomiędzy krokami czasowymi musimy jedynie uaktualniać wartość wektora wyrazów wolnych tak, by elementy znajdujące się na brzegach miały odjętą odpowiednią ilość $\frac{lT_b}{h^2}$, zależną od ilości brzegów, z którymi dany element się styka.

2.3 Interfejs

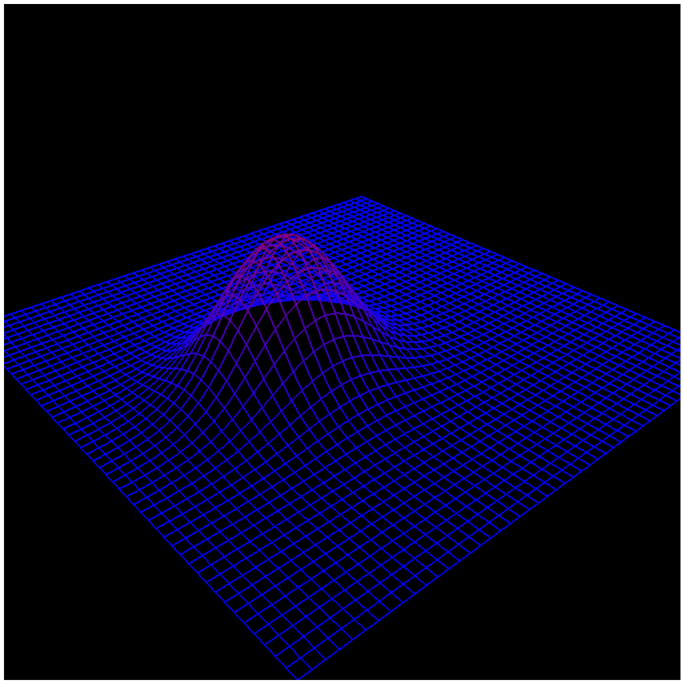
Interfejs napisany jest w perlu, z użyciem OpenGL. Wyświetla płytkę z odpowiednim gradientem kolorów (niebieski dla małych temperatur, czerwony dla dużych). Może wyświetlać w trybie wireframe (klawisz w), trybie punktowym (klawisz m), oraz z wypukłościami dla wartości różnych od zera (wyświetlanie w 3D, klawisz p). Spacja uruchamia/stopuje animację, przycisk r restartuje animację.

3 Pomiary, obserwacje

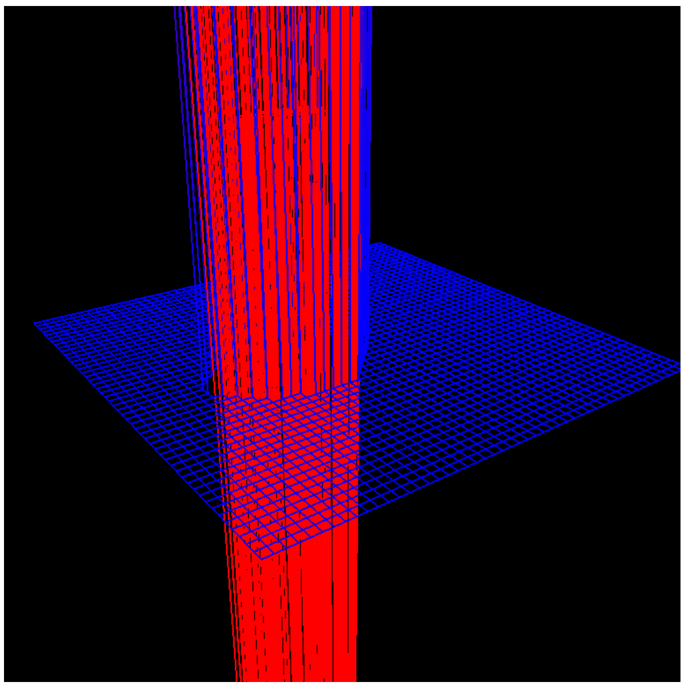
3.1 Testy czasu wykonania

Zdecydowanie szybciej działała metoda explicite - niewiątpliwie dlatego, że nie ma dodatkowego narzutu obliczeniowego. Metoda implicite działa znacznie wolniej, szczególnie dla dużych siatek, jednak jest dokładniejsza. Do tego metoda implicite nie powoduje powstawania ogromnych błędów dla małej ilości kroków czasowych, co jest problemem w metodzie explicite.

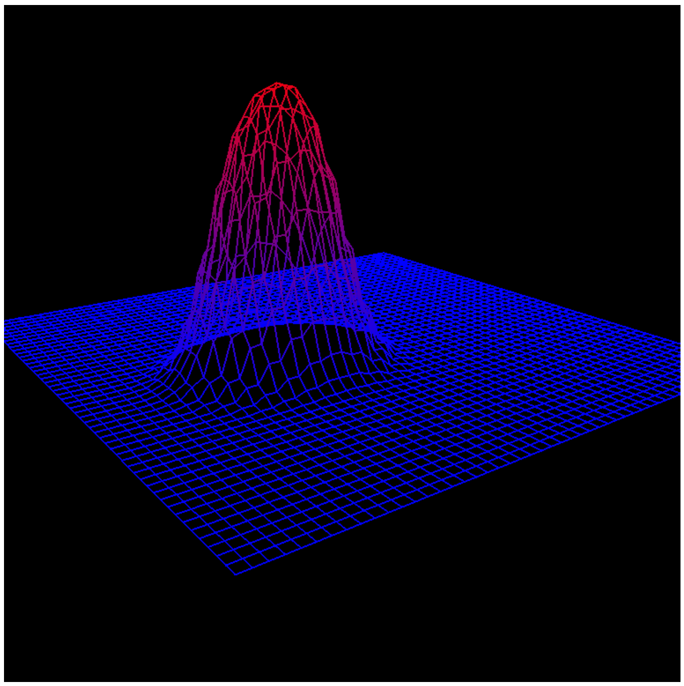
3.2 Kilka rozwiązań



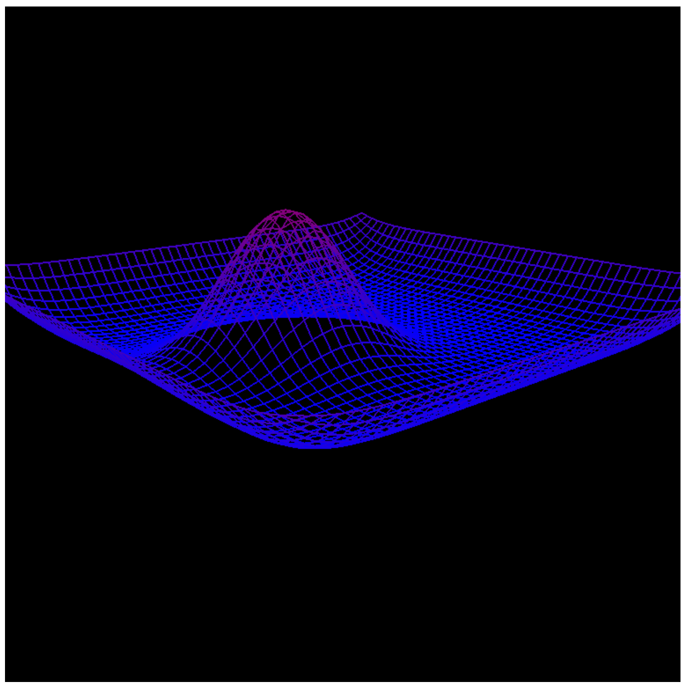
Rysunek 1: $T_b = 0$, $t_{end} = 0.1$, 100 kroków czasowych, metoda implicite, początkowa wartość to "wyspa" wysokich temperatur na środku. Ten krok czasowy to $t = 0.005$, rozwiązanie jest ciągłe i pokrywa się z przypuszczeniami co do poprawnego zachowania



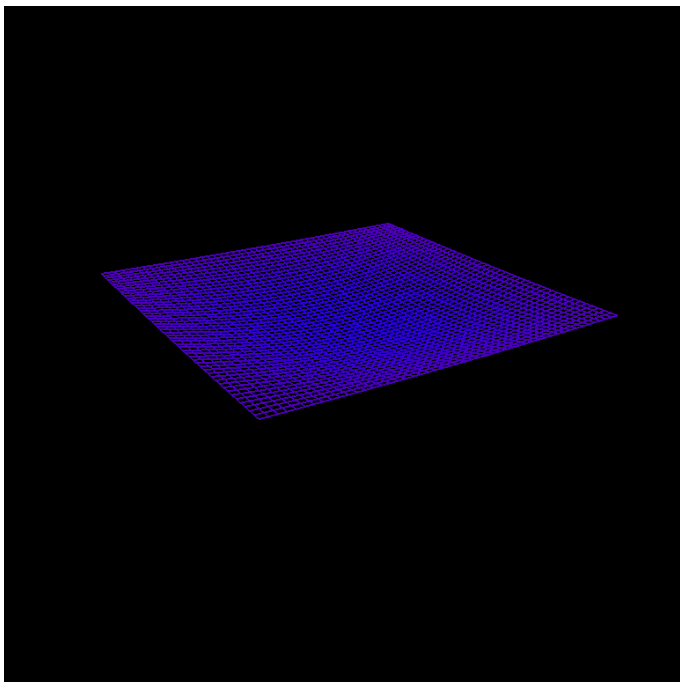
Rysunek 2: $T_b = 0$, $t_{end} = 0.1$, 100 kroków czasowych, metoda explicite, początkowa wartość to "wyspa" wysokich temperatur na środku. Ten krok czasowy to $t = 0.005$, mimo tego samego zadania problemu co w poprzednim rysunku, pojawiają się ogromne błędy. obliczeniowe i bezsensowne wyniki.



Rysunek 3: $T_b = 0$, $t_{end} = 0.1$, *1000 kroków czasowych, metoda explicite, początkowa wartość to "wyspa" wysokich temperatur na środku. Ten krok czasowy to* $t = 0.005$, *teraz wynik jest sensowny, ale nie jest tak ciągły jak rozwiązanie implicite, wydłużył się także znacznie czas obliczeń - choć wciąż jest wielokrotnie mniejszy od czasu obliczeń metody implicite. Uzyskanie tak samo ciągłego rozwiązania niweluje tę różnicę czasową. Jak widać, rozwiązanie implicite jest znacznie bardziej stabilne*



Rysunek 4: $T_b = 30$, $t_{end} = 0.1$, 100 kroków czasowych, metoda implicite, początkowa wartość to "wyspa" wysokich temperatur na środku. Ten krok czasowy to $t = 0.005$, rozwiązanie jest ciągłe i pokrywa się z przypuszczeniami co do poprawnego zachowania



Rysunek 5: $T_b = 30$, $t_{end} = 0.05$, 50 kroków czasowych, metoda implicite, ten sam problem co wcześniej, krok końcowy. Widać zbliżenie płytki do temperatury na brzegu.