

Optymalizacja obliczeń wartości minimalnej oraz maksymalnej na procesorze graficznym GPU

Filip Rynkiewicz & Marcin Daszuta

20 sierpnia 2018

1 Wstęp

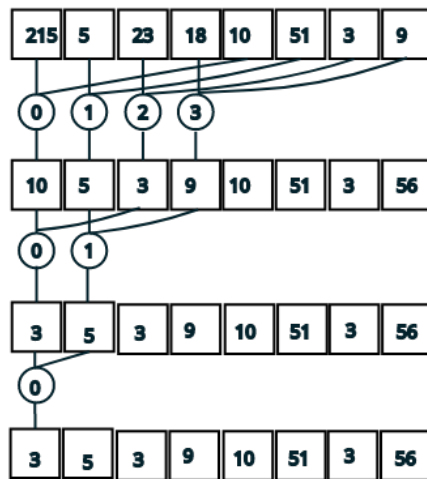
Zadania proste na procesorze CPU nie zawsze można w trywialny sposób zaimplementować na procesorze graficznym GPU. Zadania takie jak dodawanie elementów w tablicy, szukanie najmniejszej/największej liczby ze zbioru można tutaj dać jako przykład.

2 Treść

Karta graficzna z powodu swojej architektury wykonuje wiele zadań na raz. Z tego powodu tak trywialne zadanie jak wyszukiwanie najmniejszej wartości w tablicy można potraktować jako optymalizacyjne wyzwanie. Wprawdzie karty firmy Nvidia udostępnia tzw. *operacje atomowe*, w których widnieje funkcja *atomicMin()*, lecz jej zachowanie często jest zbyt powolne dla potrzeb algorytmów. Jest to uwarunkowane blokadą dostępu wszystkich możliwych wątków karty graficznej do pamięci w której są dane.

2.1 Adresowanie sekwencyjne

Jednym z pomysłów na przyspieszenie takich obliczeń są operacje *redukcji*. Polegają one na iteracyjnym wyłączeniu wątków na których wartości już zostały obliczone, czyli ich redukcje. Poprzez wyłączenie rozumie się nie wykonywanie obliczeń na tych wątkach. Przykład z Rysunku 1 pokazuje algorytm poszukiwania najmniejszego elementu w tablicy.



Rysunek 1: Przykład poszukiwania elementu najmniejszego w tablicy operacją redukcji

3 Bibliografia

- https://developer.download.nvidia.com/compute/cuda/1.1-Beta/x86_website/projects/reduction/doc/reduction.pdf