

Identificação

Oswaldo Henrique Becker - 175963@upf.br

Implementação

O preenchimento das matrizes e vetores foi feito numa função kernel e algumas variáveis foram definidas usando a diretiva *#define*. Nas funções de soma de A com B, multiplicação de A com B e A com ESCALAR foi utilizado 1 grid de 200 blocos de 30 threads cada ambos bidimensionais e na multiplicação de B com V foi utilizado 1 grid com 200 blocos de 30 threads cada ambos unidimensionais.

Recursos de programação paralela utilizados

Foram utilizados recursos de CUDA, como o qualificador de tipo de variável *__device__* para alocar estaticamente as variáveis no device e o qualificador de tipo de função *__global__* para as funções kernel, além do tipo *dim3* para as variáveis que armazenam dimensões, a função *cudaSetDevice* e a função de sincronização *cudaDeviceSynchronize*.

Ambiente de execução

Os testes foram feitos em um computador com o sistema operacional Linux Mint 20 Cinnamon instalado nativamente e com o kernel 5.4.0-56-generic. Na parte de hardware, possui um processador Intel Core i7-7700HQ com 4 núcleos, memória RAM de 15.5 GiB e uma placa de vídeo Nvidia GeForce GTX1060 Max-Q.

Análise dos resultados dos testes

Um total de 5 repetições foram feitas para calcular a média do tempo de execução nas versões paralela e sequencial do algoritmo. Abaixo a tabela separada por etapa e o total.

	A + B	A * B	A * E	B * V	Total
Paralelo	0,012s	17,685s	0,008s	0,004s	18,546s
Sequencial	0,096s	25m54,726s	0.084s	0,104s	25m55,874s

Quando executado o sequencial, o tempo total aumentou em aproximadamente 8.289%, se comparado ao paralelo. O algoritmo paralelo ficou com uma aceleração próxima à 84.

Considerações finais

É possível concluir que a paralelização deste algoritmo utilizando CUDA teve resultados expressivamente positivos, visto que os tempos de execução reduziram significativamente quando comparado à execução sequencial.

Anexos:

Tempos de Execução

Abaixo os tempos de execução de todas as repetições, usados para o cálculo das médias.

Paralelo

	A + B	A * B	A * E	B * V	Total
Execução 1:	0,012s	17,695s	0,008s	0,004s	18,710s
Execução 2:	0,012s	17,466s	0,008s	0,004s	18,380s
Execução 3:	0,012s	17,646s	0,008s	0,004s	18,379s
Execução 4:	0,012s	17,884s	0,008s	0,004s	18,792s
Execução 5:	0,012s	17,732s	0,008s	0,004s	18,467s
Média:	0,012s	17,685s	0,008s	0,004s	18,546s

Sequencial

	A + B	A * B	A * E	B * V	Total
Execução 1:	0,098s	25m58,206s	0,084s	0,105s	25m59,389s
Execução 2:	0,093s	25m52,845s	0,083s	0,106s	25m53,745s
Execução 3:	0,093s	25m42,256s	0,085s	0,104s	25m43,017s
Execução 4:	0,098s	26m04,159s	0,084s	0,104s	26m05,030s
Execução 5:	0,099s	25m56,179s	0,083s	0,103s	25m58,189s
Média:	0,096s	25m54,726s	0,084s	0,104s	25m55,874s