inesc id lisboa

Recursos Humanos

RELATÓRIO DE ACTIVIDADES DE BOLSA DE INVESTIGAÇÃO

(A preencher pelo Bolseiro)

RELATÓRIO INTERCALAR ☐ RELATÓRIO FINAL ☒

Identificação do bolseiro

Nome completo: Manuel Luís Henriques de Araújo

Identificação da bolsa

Tipo de bolsa: Bolsa de Introdução à Investigação Referência: BII_2008

Período: De: 2008 - 10 - 31 a: 2009 - 10 - 31

Nome do projecto: Desenvolvimento e análise de algoritmos para modelação de sistemas dinâmicos não-lineares

Área de trabalho: Engenharia Electrotécnica e de Computadores

Orientador científico: Prof. Luís Miguel Silveira

Actividades desenvolvidas

Objectivo

O objectivo deste trabalho foi a implementação de algoritmos de regressão em GPU's procurando comparar os desempenhos destes com o das implementações clássicas de CPU.

Conceitos Básicos

GPUs

GPUs, ou Graphics Processing Units, são unidades de processamento especificamente desenhadas para a computação gráfica e estão presentes na maioria das placas gráficas modernas. Devido à crescente exigência das aplicações gráficas, especialmente os videojogos, estas unidades têm hoje em dia um enorme poder de processamento, bem como arquitecturas altamente paralelizadas (podem conter mais de 900 processadores).

É possível utilizar todo este poder de processamento para outro tipo de aplicações, nomeadamente aplicações científicas de grande exigência computacional, conseguindo-se, em certos casos, desempenhos muito superiores aos obtidos com a utilização do CPU.

CUDA

CUDA (Computer Unified Device Architecture) é uma arquitectura de processamento paralelo desenvolvida pela NVIDIA [1] para permitir a programação dos GPUs existentes nas suas placas gráficas. Permite que estes sejam programados nas linguagens mais comuns, acrescentadas de algumas extensões específicas.

A NVIDIA disponibiliza uma biblioteca de funções em CUDA (a CUBLAS) que, apesar de ser já bastante extensa, tem algumas limitações próprias de uma linguagem recente.

Método dos Mínimos Quadrados

O método dos mínimos quadrados é um método de regressão que consiste em, dada uma função f dependente de um conjunto de parâmetros e um conjunto de pontos (x_i, y_i), determinar os valores dos parâmetros que minimizam a soma dos quadrados das diferenças $y_i - f(x_i)$.

Na prática, o método permite obter uma função que dê uma boa aproximação dos valores y_i num conjunto de pontos x_i, que pode ser utilizada para prever valores correspondentes a outros pontos x.

Neste trabalho foi utilizada uma versão do método em que a função f é um polinómio e os parâmetros a ajustar são os seus coeficientes. Neste caso, a resolução do problema de minimização referido reduz-se à resolução de um sistema linear de matriz simétrica definida positiva.

(continuar em folhas adicionais, se necessário)

Actividades desenvolvidas *(continuação)*

Decomposição de Cholesky

Se A for uma matriz simétrica definida positiva com entradas reais (ou, em geral, Hermitiana) é possível escrever A = L*L', onde L é uma matriz triangular inferior e L' denota a matriz transposta de L. A esta decomposição de A dá-se o nome de decomposição de Cholesky.

Em particular, esta decomposição é útil na resolução do sistema linear A × **x** = **b**, com A simétrica definida positiva, pois reduz a sua resolução à resolução de dois sistemas lineares de matriz triangular.

Como a aplicação do método dos mínimos quadrados implica a resolução de um sistema do tipo mencionado, a decomposição de Cholesky torna-se bastante útil no ataque ao problema.

SVM

Support Vector Machines (SVMs) são um método de regressão e classificação de dados baseado em aprendizagem. O método divide-se em duas fases: aprendizagem e regressão/classificação. Na fase da aprendizagem, um conjunto de dados é utilizado para criar um modelo que será depois utilizado no processo de regressão/classificação.

Os dados consistem num conjunto de vectores de n coordenadas (pontos de treino) aos quais estão associados valores de um conjunto discreto (classificação) ou contínuo (regressão). Pretende-se determinar um hiperplano que separe os pontos de diferentes classes (classificação) ou que se aproxime o mais possível dos valores associados aos pontos de treino (regressão).

O processo anteriormente descrito gera apenas modelos lineares. No caso de estes não serem apropriados, podemos aplicar uma transformação não linear ao conjunto dos dados (a esta transformação dá-se o nome de função de Kernel), que passam a estar num espaço de dimensão superior à original. É então possível utilizar o processo já descrito para criar um modelo linear nesse espaço, que corresponderá a um modelo não linear no espaço original.

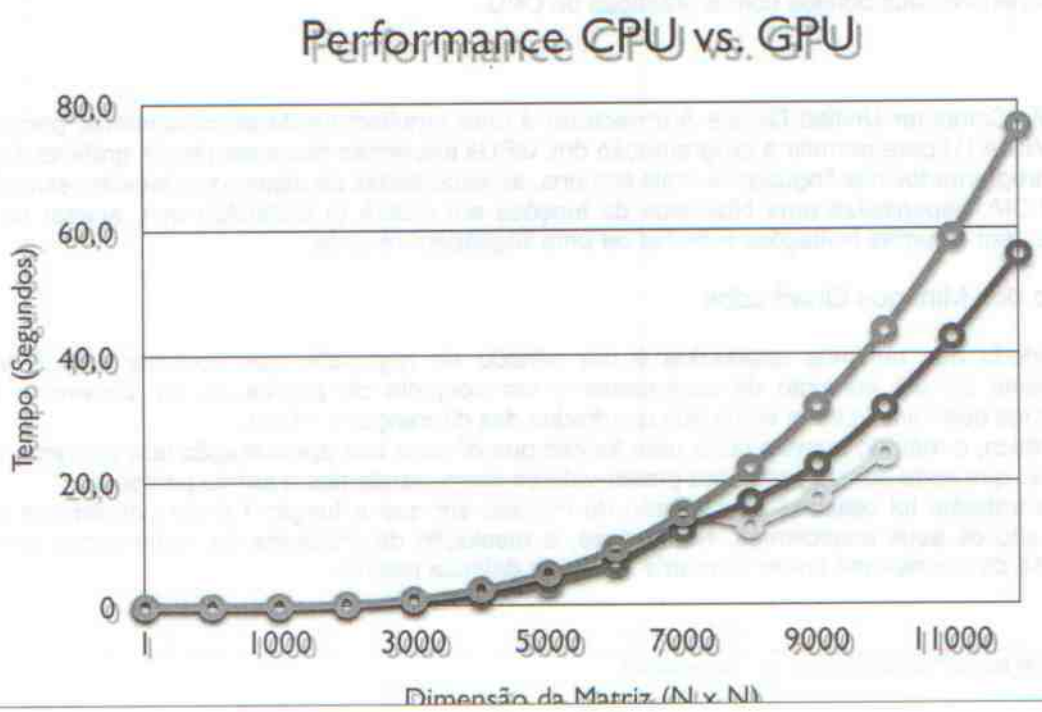
Trabalho Desenvolvido

O trabalho desenvolvido pode ser dividido em duas partes: implementação e teste de um algoritmo de regressão pelo método dos mínimos quadrados em CUDA e implementação e teste de uma SVM em CUDA.

Método dos Mínimos Quadrados

Foi desenvolvido código em CUDA que implementa o método dos mínimos quadrados, sendo que na resolução do sistema linear referido na secção introdutória foi utilizado código para a decomposição de Cholesky disponibilizado por Steven Gratton em [1], que foi posteriormente adaptado. Foram desenvolvidas versões do código em precisão single e double.

Os resultados comparativos obtidos nesta parte do trabalho estão sintetizados nos seguintes gráficos (optou-se por representar o desempenho dos algoritmos de decomposição de Cholesky, visto ser esta a parte que mais afecta o desempenho do método dos mínimos quadrados, tanto em termos de velocidade como de precisão):



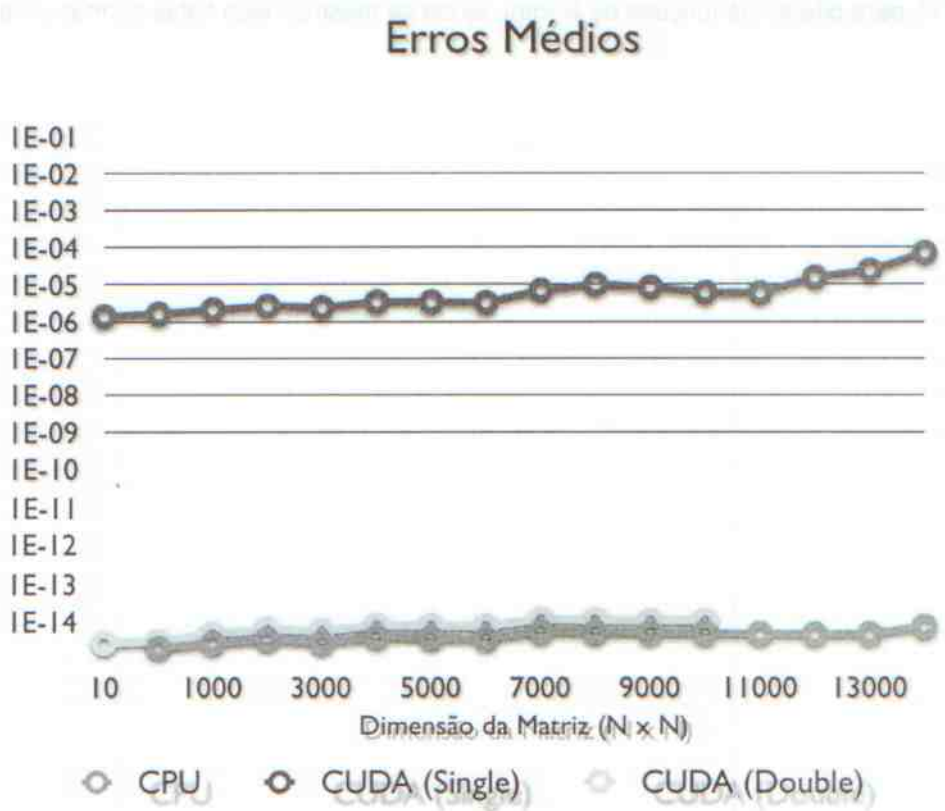


Fig. 2 - Erros médios cometidos na aplicação da decomposição de Cholesky (CPU vs GPU single vs GPU double)

Concluímos que a implementação em GPU, utilizando precisão double, permite obter resultados com um grau de correcção muito próximo ao do CPU, em tempos inferiores (com um ganho máximo de 1,5x).

SVM

Nesta segunda parte do trabalho, foram utilizadas as implementações *cuSVM,* disponível em [3] (implementação em CUDA) e *LIBSVM*, disponível em [4] (implementação em CPU).

O objectivo final desta parte é o estudo comparativo do desempenho das duas implementações, aplicadas a conjuntos de dados concretos, utilizando diferentes funções de Kernel e avaliando também os seus desempenhos relativos. Para tal, é necessário efectuar alterações ao código da implementação em CUDA, de modo a permitir a utilização de diferentes funções de Kernel, visto que, tal como é disponibilizado, este só permite a utilização da RBF (Radial Basis Function).

Esta parte do trabalho está em fase de conclusão, não havendo por isso ainda resultados a apresentar.

[1] - http://www.ast.cam.ac.uk/~stg20/cuda/cholesky/index.html
[2] - http://www.nvidia.com
[3] - http://patternsonascreen.net/cuSVM.html
[4] - http://www.csie.ntu.edu.tw/~cjlin/libsvm/

Desvios em relação ao planeado e respectiva justificação

Não foi possível ainda atingir o objectivo final de efectuar testes comparativos das implementações das SVMs em CUDA e GPU, para diferentes funções de Kernel, tendo as mesmas sido feitas apenas para o Kernel RBF.

Publicações e trabalhos elaborados no âmbito da bolsa

Código do método dos mínimos quadrados:

http://algos.inesc.pt/wikicuda/index.php/C%C3%B3digo_Lan%C3%A7ado:M%C3%A9todo_dos_M%C3%ADnimos_Quadrados

Código da decomposição de Cholesky (adaptado de Steven Gratton [1]):

http://algos.inesc.pt/wikicuda/index.php/C%C3%B3digo_Lan%C3%A7ado:Decomposi%C3%A7%C3%A3o_de_Cholesky

Bolseiro

Manuel Luís Henriques de Araújo

Assinatura: Manuel Luís Henriques de Araújo Data: 2009 - 12 - 4

Orientador Científico

Luís Miguel Silveira

Assinatura: Luís Miguel Silveira Data: 2009 - 12 - 4