##### Pontifícia Universidade Católica de Goiás

Programa de Mestrado em Engenharia de Produção e Sistemas

**Análise nodal de circuito elétrico empregando computação paralela GPU CUDA.**

Bierley Souza Machado

10 de fevereiro de 2020.

**Análise nodal de circuito elétrico empregando computação paralela GPU CUDA.**

Bierley Souza Machado

Dissertação de Mestrado apresentada ao Pro- grama de Pós-Graduação em Engenharia de Produção e Sistemas da Pontifícia Universidade Católica de Goiás, como parte dos requisitos para obtenção do título de Mestre em Engenha- ria de Produção e Sistemas.

Orientador: Clarimar José Coelho, Dr.

Coorientador: Arlindo Rodrigues Galvão Filho, Dr.

Goiânia

10 de fevereiro de 2020

**Análise nodal de circuito elétrico empregando computação paralela GPU CUDA.**

Bierley Souza Machado

Esta Dissertação julgada adequada para obtenção do título de Mestre em Engenharia de Produção e Sistemas, e em sua forma final pelo Programa de Pós-Graduação em Engenharia de Produção e Sistemas da Pontifícia Universidade Católica de Goiás em 10 de fevereiro de 2020.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. Marcos Lajóvic Carneiro, Dr.

Coordenador do Programa de Pós-Graduação em engenharia de Produção e Sistemas

Banca Examinadora:

Prof. Clarimar José Coelho, Dr. Orientador

Prof. Rafael Viana Carvalho Dr. Avaliador Externo

Prof.ª Solange da Silva. Dr.ª Avaliadora Interna - PUC Goiás

GOIÂNIA - GO

10 de fevereiro de 2020.

Dados Internacionais de Catalogação da Publicação (CIP) (Sistema de Biblioteca PUC Goiás)

Bierley Souza Machado

Análise nodal de circuito elétrico empregando computação paralela

GPU CUDA – Goiânia – Goiás: PUC Goiás/MEPROS, 10 de fevereiro de 2020.

Orientador: Clarimar José Coelho

Dissertação (mestrado) – Pontifícia Universidade Católica de Goiás Programa de Mestrado em Engenharia de Produção e Sistemas, 10 de fevereiro de 2020.

Referências Bibliográficas: p.40-44.

Programa de Mestrado em Engenharia de Produção e Sistemas. Análise nodal de circuito elétrico empregando computação paralela GPU CUDA.

A Deus, que nos capacita até para os sonhos que parecem impossíveis; à minha esposa, Elaine Rodrigues Nogueira, e a minha mãe Solange Nilce de Souza, que sempre me apoiaram, acreditaram no meu esforço e abriram mão de passar um tempo comigo para a concretização deste trabalho.

**Agradecimentos**

Ao meu orientador, Prof. Dr. **Clarimar José Coelho**, que me acompanhou em todo tempo da minha jornada.

Aos meus colegas e companheiros do MEPROS que sempre me incentivaram.

Ao Programa de Mestrado em Engenharia de Produção e Sistemas que me possibilitou realizar este trabalho.

A colaboração nesse projeto do Laboratório de Computação Científica da ECEC-PUC Goiás e a Energia Sustentável do Brasil S.A. (ESBR) e como recurso humano nesse projeto a coorientação do Professor Dr. **Arlindo Rodrigues Galvão Filho**.

À Fundação Aroeira e Usina Jirau, pelo estímulo a pesquisa e pelo financiamento do Projeto de Pesquisa & Desenvolvimento **PD-06631- 0007/2018** ao qual este projeto faz parte, intitulado como **”Desenvolvimento Experimental de Gêmeo Digital para Usinas Hidrelétricas”**.

**Resumo**

No contexto da abordagem de processamento paralelo para a solução de sistemas lineares que representam nós de circuito elétrico para o cálculo de suas tensões e demais grandezas elétricas: corrente e potência. A análise nodal é baseada na primeira Lei dos nós de Kirchhoff e é usada para a resolução do circuito elétrico através de equações que formam um sistema linear. A solução deste sistema linear é obtida com o processamento paralelo empregando o hardware da placa de vídeo e a arquitetura de dispositivo de computação unificada (*CUDA*). Porém para esta modelagem necessitamos de um esforço computacional que pode inviabilizar seu uso no contexto de Gêmeo Digital. Neste contexto, este trabalho propõe o uso da programação paralela *CUDA* para a redução do tempo computacional gasto nos modelos. Com o uso dos modelos paralelizados, obtém-se um ganho de performance temporal do Gêmeo Digital equiparando o tempo de resposta ao da planta real. Os resultados obtidos serão incorporados em um modelo de Gêmeo Digital como previsto no projeto de Pesquisa Desenvolvimento Experimental de Gêmeo Digital da UHE-Jirau em Escopo Completo com Tecnologia de Processamento de Eventos Complexos Distribuídos para Investigações Sistêmicas (PD-06631-0007/2018).

**Abstract**

In the context of the parallel processing approach to the solution of linear systems that represent electrical circuit nodes for the calculation of their voltages and other electrical quantities: current and power. Nodal analysis is based on Kirchhoff's first Node Law and is used for the resolution of the electrical circuit through equations that form a linear system. The solution of this linear system is achieved through parallel processing employing the graphics card hardware and unified computing device architecture (CUDA). However for this modeling we need a computational effort that can make its use in the context of Digital Twin unfeasible. In this context, this work proposes the use of CUDA parallel programming to reduce the computational time spent on the models. Using parallelized models, the Digital Twin's temporal performance gain is expected to match the response time to that of the actual plant. The results obtained will be incorporated into a Digital Twin model as foreseen in the UHE-Jirau Digital Twin Experimental Development Research Project in Full Scope with Distributed Complex Event Processing Technology for Systemic Investigations (PD-06631-0007 / 2018).

**Sumário**

[Capítulo 1 - Introdução](#_Toc4401603) 1

1[.1 - Considerações Iniciais](#_Toc4401603) 1

1[.2 - Análise Nodal](#_Toc4401603) 1

1[.3 - Análise de Malhas](#_Toc4401603) 2

1[.4 - A arquitetura CUDA](#_Toc4401603) 3

1[.5 - Visão Geral](#_Toc4401603) 4

1[.6 - Organização do trabalho](#_Toc4401603) 5

[Capítulo 2 – Materiais e Métodos](#_Toc4401603) 6

[2.1 - O uso dos materiais e métodos](#_Toc4401603) 6

[2.2 - Resolução de circuitos pela análise nodal](#_Toc4401603) 7

[2.3 - A solução do sistema linear da equação paralelizada em GPU CUDA](#_Toc4401603) 9

[2.4 - O algoritmo do processo adotado](#_Toc4401603) 10

[Capítulo 3 - Programação paralela em CUDA](#_Toc4401603) 13

[3.1 - Introdução](#_Toc4401603) 13

[3.2 - Speed-up](#_Toc4401603) 14

[3.3 - Lei de Amdahl](#_Toc4401603) 14

[3.4 - Lei de Gustafson](#_Toc4401603) 16

[3.5 - Vantagens da programação paralela](#_Toc4401603) 17

[3.6 - Principais dificuldades](#_Toc4401603) 17

[3.7 - Sistema operacional multitarefa](#_Toc4401603) 18

[3.8 - Flops](#_Toc4401603) 19

[3.9 - As principais diferenças entre o acesso as memorias da GPU e CPU](#_Toc4401603) 20

[Capítulo 4 - Metodologia proposta para resolução de circuito elétrico](#_Toc4401603) 21

[4.1 - Introdução](#_Toc4401603) 21

[4.2 - OpenMP](#_Toc4401603) 21

[4.3 - OpenACC](#_Toc4401603) 22

[4.4 - OpenCL](#_Toc4401603) 22

[4.5 - Ganho de eficiência no envio de dados](#_Toc4401603) 22

[4.6 – Onde Aplicar a metodologia na usina Hidrelétrica de Jirau](#_Toc4401603) 23

[4.7 – Experimento](#_Toc4401603) 23

4.7.1 - Ladrilhamento .......................................................................... 24

4.7.2 - Multiplicação de matrizes na GPU ........................................... 24

4.7.3 - Multiplicação de matrizes na GPU com OpenMP ................ .... 25

4.7.4 – Multiplicação para precisão simples ......................................... 25

4.7.5 – CPU × CPU + OpenMP ............................................................. 26

4.7.6 – CPU + OpenMP × GPU ........................................................... 27

4.7.7 – Multiplicação para precisão dupla ............................................ 28

4.7.8 – CPU × CPU + OpenMP ........................................................... 28

4.7.9 – GPU × CPU + OpenMP ........................................................... 28

4.7.10 – Comparação entre precisão simples e dupla .......................... 29

[Capítulo 5 - Comentários Finais](#_Toc4401603) 54

Referências [Bibliográfias](#_Toc4401603) 55

**Índice de Figuras**

[Figura 1. Circuito para análise nodal, adaptado de Alexander . 6](#_Toc4401600)

[Figura 2. Elementos da arquitetura CUDA . 9](#_Toc4401600)

Figura 3. Argumento de Amdahl. 15

Figura 4. O tempo de execução e *speedup*, adaptado de Amdahl. 16

Figura 5. Sistema paralelo, adaptado de Correa. 18

Figura 6. Memória compartilhada, adaptado de Chapman. 20

Figura 7. Multiplicação de matrizes com e sem ladrinho (GPU). 25

Figura 8. Multiplicação de matrizes com e sem ladrinho (CPU). 26

Figura 9. Processamento em Blocos de tamanho 8, 16 e 32. 27

Figura 10. Custo computacional do algoritmo. 30

**Índice de tabelas**

Tabela 1 - Tempo processamento CPU x GPU em segundos...................................29

**Lista de siglas**

ALU *Arithmetic Logic Unit*

AFIPS Conferência Conjunta de Informática unificada

API *Application programming interface*

BLOCK Conjunto de *threads* que serão processadas simultaneamente

CACHE Arquivo temporário

CPS *Cyber Physical System*

CUDA *Compute Unified Device Architecture*

CPU *Central Processing Unit*

CLOCK Referencial de potência

DIGITAL TWIN Gêmeo Digital

DT Gêmeo Digital

GRID Configuração da placa para a execução de um *kernel*

GPU *Graphics Processor Unit*

INTRANET Uma [rede de computadores](https://pt.wikipedia.org/wiki/Rede_de_computadores) privada

INTERNET Sistema global de [redes de computadores](https://pt.wikipedia.org/wiki/Redes_de_computadores) interligadas

KCL *Kirchhoff Current Law*

KERNEL Núcleo do sistema operacional

KVL *Kirchhoff Voltage Law*

PIPELINE Busca de uma ou mais instruções além da próxima a ser executada

OPENACC For *open accelerators*) is a programming standard for [parallel computing](https://en.wikipedia.org/wiki/Parallel_computing)

OPENCL Uma arquitetura para escrever programas que funcionam em plataformas heterogêneas

OPENMP *Open Multi-Processing*

SM *Stream Multiprocessors*

SPEEDUP A[umento da velocidade de algo](https://www.linguee.com.br/portugues-ingles/traducao/aumentar+a+velocidade+de.html)

THREADS São fluxos de processamento independente

THROUGHPUT Taxa de transferência

UHE Usina Hidrelétrica