

Implementação Paralela do Algoritmo de Roteamento de Veículos com Janelas de Tempo sob a Plataforma CUDA

Aluno: Thiago William Machado

RA: 107577

thiagowilliamm@yahoo.com.br

Orientador: Prof. Dr. Ricardo Ribeiro dos Santos

RF: 59978

ricr.santos@gmail.com

17 de junho de 2009

1 Apresentação e Justificativa

Os dias atuais vêm pontuando cada vez mais a necessidade por pesquisas em processamento de alto desempenho. Problemas novos surgem em todas as áreas com grande necessidade de processamento e enormes massas de dados que necessitam ser manipulados.

Além disso, existem vários problemas importantes e clássicos para os quais já existem algoritmos construídos, mas que, muitas vezes, tais algoritmos não são utilizados pois demandam muito tempo de processamento para a quantidade real de dados da aplicação. Esse é o caso comum dos algoritmos que tratam dos problemas de caminhos e roteamento. Existem, ainda, algumas classes de aplicação para as quais qualquer pequena melhora no tempo de processamento de seus dados já significa algum tipo de sucesso, como por exemplo, aplicações que manipulam problemas que devem ser resolvidos em tempo real e que manipulam uma grande quantidade de informação.

Dessa forma, o processamento de alto desempenho encontra aplicação em quase todas as áreas de conhecimento e pesquisas nessa área são interdisciplinares por natureza. Dentro do Grupo de Pesquisas em Engenharia e Computação (GPEC) da UCDB a área de processamento de alto desempenho faz parte da linha de pesquisa **Computação de Alto Desempenho e Sistemas Distribuídos** com uma interface bastante forte com diversas outras áreas da computação como: arquitetura de computadores, compiladores, sistemas microprocessados, projeto de hardware, bioinformática, teoria dos grafos e otimização. Dentro do GPEC, há também trabalhos envolvendo processamento de alto desempenho desenvolvidos e em desenvolvimento com aplicações em outras áreas do conhecimento como: fluxo de potência ótimo (engenharia elétrica), problema da árvore geradora mínima (teoria dos grafos), análise e detecção do couro bovino (visão computacional, medicina veterinária, zootecnia), entre outros.

Diante do exposto, este plano de trabalho foca o desenvolvimento e avaliação de desempenho de um algoritmo paralelo para o problema de roteamento de veículos com restrições de janela de tempo [17] tendo como plataforma base a arquitetura CUDA (*Computer Unified Device Architecture*) [1,15]. O desenvolvimento e avaliação de desempenho do algoritmo proposta neste plano de trabalho acontecerá em uma plataforma GP-GPU formada por uma placa de vídeo NVIDIA com suporte para a arquitetura CUDA. Esse hardware está sendo adquirido e será instalado no Laboratório de Pesquisa e Desenvolvimento do Grupo de Pesquisas em Engenharia e Computação da UCDB. É importante ressaltar que essa proposta está relacionada com outro plano de trabalho também focado na implementação paralela do PRVJT mas que utilizará

uma plataforma de hardware formada por clusters de computadores utilizando a biblioteca MPI (*Message Passing Interface*) [13]. Além de ter a possibilidade de comparar os resultados obtidos com a implementação na arquitetura CUDA com outras implementações já existentes para o PRVJT, uma outra motivação para o desenvolvimento deste plano de trabalho é que não foram encontrados, na literatura da área, outros trabalhos que tratam do referido problema nessa plataforma de hardware.

2 Objetivos

Implementar algoritmo paralelo para o problema de roteamento de veículos com janela de tempo e avaliar o desempenho dessa implementação com outros trabalhos encontrados na literatura da área.

Os objetivos específicos são:

1. Estudar o algoritmo sequencial para o PRVJT assim como as características da arquitetura CUDA.
2. Pesquisar a literatura da área a fim de realizar levantamento bibliográfico sobre outros algoritmos ou implementações paralelas para o PRVJT.
3. Desenvolver e implementar algoritmo para o PRVJT baseando na arquitetura CUDA.
4. Avaliar o desempenho da versão paralela implementada com a versão sequencial e com outros algoritmos apresentados na literatura da área.
5. Organização do material estudado e desenvolvido, preparação, escrita e submissão de artigos.

3 Revisão da Literatura

Esta Seção apresenta os principais conceitos que são a base para o desenvolvimento deste trabalho. Nas subseções a seguir apresenta-se os conceitos envolvendo roteamento de veículos, roteamento de veículos com janelas de tempo, uma estratégia para resolução do problema PRVJT e algumas informações básicas sobre a arquitetura CUDA.

3.1 Problema de Roteamento de Veículos

Problemas de roteamento de veículos desempenham um papel central no gerenciamento da distribuição de bens e serviços. Sua importância é refletida na grande variedade de aplicações, as quais incluem o roteamento de ônibus urbano, sistemas de coleta de lixo, entrega de periódicos a assinantes, entre outros [16]. Dependendo da aplicação, há necessidade de alterar as restrições do problema. Algumas restrições bem conhecidas para o problema de roteamento de veículos são [6,16]: restrições de horário de atendimento (conhecidas na literatura como janelas de tempo ou janelas horárias), capacidades dos veículos, frota de veículos de diferentes tamanhos, duração máxima dos roteiros dos veículos (tempo ou distância) e restrições de veículos que podem atender determinados clientes. Problemas de roteirização de veículos são muitas vezes definidos como problemas de múltiplos caixeiros viajantes com restrições adicionais de capacidade [5].

Uma aplicação típica para o problema de roteamento de veículos consiste em determinar as rotas a serem operadas por uma frota de veículos que, partindo de uma garagem ou depósito central, deve suprir demandas conhecidas de um conjunto de clientes. Cada cliente é servido por exatamente um veículo e todas as rotas começam e terminam na garagem, tendo como objetivo, minimizar a distância total percorrida

Sob a ótica da otimização, os problemas de roteirização de veículos, incluindo o caso particular do caixeiro viajante, pertencem à classe de complexidade computacional denominada NP-completo [14]. Dessa forma, diversas soluções para o problema de roteamento de veículos são baseadas em heurísticas de tempo polinomial [18]. Nesse sentido, uma grande quantidade de trabalhos focados no problema de roteamento de veículos com janelas de tempo têm sido desenvolvidos aplicando diferentes técnicas [2,3,6,7,10–12]. No entanto, as dificuldades inerentes à definição de algoritmos eficientes para o problema têm limitado a aplicabilidade dessas soluções a partir de determinado tamanho do conjunto de instâncias de entrada.

3.2 Problemas de Roteamento de Veículos com Janelas de Tempo

O problema de Roteamento de Veículos com Janelas de Tempo (PRVJT) apresentado inicialmente na Seção 1 e foco deste plano de trabalho é baseado [9]. O algoritmo Simplex com Geração de Colunas é utilizado em conjunto com o algoritmo de Caminho Mínimo com Janelas de Tempo (CMJT) baseado na generalização do algoritmo clássico de Ford-Bellman-Moore [4, 19]. A utilização do algoritmo CMJT permite resolver o problema de partição de conjuntos, utilizado na modelagem do problema de roteamento com janelas de tempo. A

adoção da técnica de Geração de Colunas possibilita obter subproblemas que permanecem com características combinatórias semelhantes ao problema original.

Uma modelagem de programação linear inteira para o PRVJT pode ser da seguinte forma. Considere um conjunto de viagens onde cada viagem i é especificada por um intervalo de tempo $[a_i, b_i]$, no qual a viagem deve ser iniciada. Uma viagem é chamada uma “jornada produtiva”. Define-se inter-viagem ou viagem de conexão como sendo uma “jornada não-produtiva” conduzida por um veículo. Assim, uma inter-viagem é representada pelo arco (i, j) que vai do final da viagem i para o início da viagem j . A cada inter-viagem (i, j) associamos um tempo de duração dado por t_{ij} e um custo dado por c_{ij} . Uma rota é uma seqüência de viagens e inter-viagens conduzida pelo mesmo veículo. Considere P como sendo o conjunto das viagens, I o conjunto das inter-viagens, A o conjunto de arcos (arestas), $A = I \cup (\{s\} \times P) \cup (P \times \{t\})$, e N o conjunto de nós, $N = P \cup \{s, t\}$. Uma inter-viagem é considerada se e somente se for possível realizar a viagem j depois da viagem i , onde a restrição de intervalo de tempo $(a_i + t_{ij} \leq b_j)$ é respeitada [8, 9, 16]. A formulação para a definição de rotas ótimas para viagens pode ser definida a partir das seguintes variáveis:

- $x_{ij} = \begin{cases} 1 & \text{se o arco } (i, j) \text{ for usado por um veículo} \\ 0 & \text{caso contrário} \end{cases}$, onde $(i, j) \in A$
- t_i : variável que representa o tempo associado ao início de cada viagem i , com $i \in P$.

As rotas ótimas que respeitam as restrições de horários são as soluções para o seguinte problema:

$$\begin{aligned}
 & \text{Minimizar} && \sum_{(i,j) \in A} c_{ij} x_{ij} \\
 & \text{sujeito a} && \\
 & && \sum_{j \in N} x_{ij} = 1, && i \in P && (1) \\
 & && \sum_{k \in N} x_{ik} = 1, && i \in P && (2) \\
 & && x_{ij} \geq 0, && (i, j) \in A && \\
 & && x_{ij} > 0 \Rightarrow t_i + t_{ij} \leq t_j && (i, j) \in N && (3) \\
 & && a_i \leq t_i \leq b_i, && i \in P && (4) \\
 & && x_{ij} = \{0, 1\}, && (i, j) \in A &&
 \end{aligned}$$

A restrição (1) informa que de algum nó j o veículo chegará ao nó i e a restrição (2) indica que o veículo que chegou ao nó i terá que partir para algum nó j . A restrição (3) descreve a compatibilidade requerida entre as rotas e os horários e a restrição (4) estabelece o intervalo de tempo no qual a viagem deve ser iniciada.

3.3 Arquitetura CUDA

A *Computer Unified Device Architecture* (CUDA) [1, 15] é uma arquitetura de computação paralela de propósito geral que tira proveito dos recursos de hardware provenientes das unidades de processamento gráfico (GPUs) NVIDIA[®] para resolver problemas computacionais complexos com desempenho semelhante ou, em alguns casos, superior a um computador de propósito geral com vários núcleos de processamento.

Para programar para a arquitetura CUDA, desenvolvedores podem usar a linguagem C.CUDA é composta de uma hierarquia de grupo de *threads*, memórias compartilhadas e sincronização via barreiras. Todas essas abstrações são disponibilizadas para o desenvolver por meio de um conjunto de extensões para a linguagem C que possibilitam o paralelismo de dados e *threads*.

4 Metodologia

A metodologia para desenvolvimento deste plano de trabalho toma como base os objetivos estabelecidos na Seção 2.

1. Estudo sobre o algoritmo sequencial para o PRVJT e arquitetura CUDA.
2. Levantamento bibliográfico sobre outros algoritmos e soluções paralelas para o PRVJT.
3. Projeto e implementação do algoritmo para PRVJT utilizando a arquitetura CUDA.
4. Avaliação de desempenho considerando a implementação paralela, o algoritmo sequencial e outros algoritmos paralelos para o PRVJT encontrados na literatura da área.
5. Preparação e escrita de artigos e outros materiais de divulgação do trabalho.

5 Cronograma

A Tabela 1 detalha os prazos para execução de cada uma das atividades apresentada na Seção Metodologia.

Atividade	2009					2010						
	Ago	Set	Out	Nov	Dez	Jan	Fev	Mar	Abr	Mai	Jun	Jul
1	x	x	x	x								
2	x	x	x									
3			x	x	x	x	x	x	x			
4								x	x	x	x	x
5								x	x	x	x	x

Tabela 1: Cronograma para execução do plano de trabalho

Referências

- [1] R. Azevedo and B. C. Lopes. Computação de alto desempenho utilizando cuda. In *IX Anais do Simpósio em Sistemas Computacionais (WSCAD-SSC)*, Outubro 2008.
- [2] L. D. Boldin and B. L. Golden. Classification in Vehicle Routing and Scheduling. *Networks*, 11:97–108, 1981.
- [3] M. P. F. Caramuru. Uma Metodologia Híbrida Colônia de Formigas-Busca Tabu-Reconexão por Caminhos para Resolução do Problema de Roteamento de Veículos com Janelas de Tempo. Master’s thesis, CEFET-MG, Belo Horizonte, 2006.
- [4] T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms*. MIT Press, 1990.
- [5] C. B. Cunha. *Uma Contribuição para o Problema de Roteirização de Veículos com Restrições Operacionais*. PhD thesis, Departamento de Engenharia de Transportes - EP/USP, São Paulo, 1997.
- [6] C. B. Cunha. Aspectos Práticos da Aplicação de Modelos de Roteirização de Veículos a Problemas Reais. *Transportes*, 8(2), 2000.
- [7] M. Desrochers, J. Desrosiers, and M. Solomon. A New Optimization Algorithm for the Vehicle Routing Problem with Time Windows. *Operation Research*, 40:342–354, 1992.
- [8] M. Desrochers and F. Soumis. A Generalized Permanent Labelling Algorithm for the Shortest Path Problem with Time Windows. *INFOR*, 26, 1988.
- [9] J. Desrosiers, P. Pelletier, and F. Soumis. Plus Court Chemin Avec Contraintes d’Horaires. *RAIRO*, 17:357–377, 1983.

- [10] J. Desrosiers, F. Soumis, M. Desrochers, and M. Sauvé. Methods for Routing with Time Windows. *European Journal of Operational Research*, 23:236–245, 1986.
- [11] M. Fisher and R. Jaikumar. A Generalized Assignment Heuristics for Vehicle Routing. *Networks*, 23:109–124, 1981.
- [12] M. L. Fisher, K. O. Jörnsten, and O. B. G. Madsen. Vehicle Routing with Time Windows: Two Optimization Algorithms. *Operations Research*, 45:488–492, 1987.
- [13] The MPI Forum. The mpi message passing interface standard. Technical report, 1995.
- [14] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, 1979.
- [15] J. Nickolls, I. Buck, M. Garland, and K. Skadron. *Scalable Parallel Programming with CUDA*. ACM, 2008.
- [16] R. M. Oliveira. A Técnica de Geração de Colunas Aplicada a Problemas de Roteamento. Master’s thesis, ICMC-USP, São Carlos, 2001.
- [17] R. M. Oliveira and R. Santos. Problema de Roteamento com Janelas de Tempo: Uma Abordagem via Geração de Colunas. In *XL Anais do Simpósio Brasileiro de Pesquisa Operacional*, Setembro 2008.
- [18] M. M. Solomon. On the Worst-Case Performance of Some Heuristics for the Vehicle Routing and Scheduling with Time Windows Constraints. *Networks*, 16:161–174, 1986.
- [19] R. E. Tarjan. *Data Structures and Network Algorithms*. Conference Series in Applied Mathematics, 1983.