



ROTEIRIZAÇÃO ÓTIMA USANDO LINHAS DE ÔNIBUS

Italo Messias Felix Santos¹ – italofcontato@gmail.com

Mariane dos Santos Bispo² – maribispo_12@hotmail.com

Hassan Sherafat³ – sherafat.hassan@gmail.com

¹ Universidade Federal de Sergipe, Campus São Cristovão – São Cristovão, SE, Brazil

² Universidade Federal de Sergipe, Campus São Cristovão – São Cristovão, SE, Brazil

³ Universidade Federal de Sergipe, Campus São Cristovão – São Cristovão, SE, Brazil

Resumo. *O ônibus é o principal meio de locomoção nas grandes cidades. Em algumas cidades existem terminais de integração que facilitam a troca de linhas, sem custo adicional para o passageiro. Para se locomover de um ponto da cidade para outro em geral existem diversas combinações de linhas para fazer o percurso, entretanto, não existe clareza para o passageiro que sequência de linhas minimizaria o tempo da viagem. Um aplicativo que pudesse fornecer esta informação seria de grande utilidade para todo usuário do sistema de transporte público.*

Neste trabalho é desenvolvido uma metodologia que permitiu o desenvolvimento de um aplicativo denominado RoloApp (Roteirização Ótima usando Linhas de Ônibus), a qual identifica a melhor sequência de linhas de ônibus, com troca em terminais de integração, de modo a minimizar a duração total do percurso que inclui os tempos de espera nos terminais, em tempo real.

Palavras-chave: *Ônibus, Caminhos Mais Curtos, Roteirização, Otimização.*

1. INTRODUÇÃO

O ônibus é o principal meio de locomoção nas grandes cidades. Em algumas cidades existem terminais de integração que facilitam a troca de linhas, sem custo adicional para o passageiro. Para se locomover de um ponto da cidade para outro em geral existem diversas combinações de linhas para fazer o percurso, entretanto, não existe clareza para o passageiro que sequência de linhas minimizaria o tempo da viagem. Um aplicativo que pudesse fornecer esta informação seria de grande utilidade para todo usuário do sistema de transporte público.

O problema de Roteirização Ótima usando Linhas de Ônibus (ROLO) pode ser modelado em uma estrutura de grafo, entretanto, possui características que divergem do problema padrão. A grande diferença do problema em estudo com o problema clássico de caminhos mínimos é a estrutura de custos associada às arestas, uma vez que o tempo de viagem entre dois pontos, por exemplo dois terminais de integração, não é uma constante, mas depende do

horário (trânsito) e de que linha de ônibus percorrem o trecho (normalmente existem várias linhas que passam por um mesmo trecho). Por outras palavras, a matriz de custo não é dada no problema de ROLO. É algo a ser construído no decorrer do algoritmo.

Nenhum dos algoritmos da Teoria dos Grafo resolveria diretamente o problema posto acima. Entretanto, a estrutura do algoritmo de Dijkstra é usada como a lógica geral para resolver o problema, e para aperfeiçoar a teoria a ser chamada de GrafoRolo e utiliza-la como base para o desenvolvimento de um algoritmo a ser denominado RoloCalc.

Neste trabalho é desenvolvido uma metodologia que permitiu o desenvolvimento de um aplicativo denominado RoloApp (Roteirização Ótima usando Linhas de Ônibus), a qual identifica a melhor sequência de linhas de ônibus, com troca em terminais de integração, de modo a minimizar a duração total do percurso que inclui os tempos de espera nos terminais, em tempo real.

2. MODELAGEM DO PROBLEMA – GRAFO-ROLO

Seja $G=(V, A)$ um grafo onde V é um conjunto de vértices, A um conjunto de arcos onde para cada arco $\alpha \in A$ será atribuído um período T de 0 à 1440 minutos (ou 0 à 24 horas). O período T será particionado em n intervalos. Seja $L_\alpha = \{F_k^\alpha: V_X V \rightarrow Z\}; k=(1, \dots, n)$ onde o índice α de L_α indica que este conjunto de funções é de um arco α , como conjuntos de funções de incidência para cada α em A (funções tempo) que associa a cada par de vértices ordenado i e j distintos um peso P em minutos que representa o tempo necessário para ir de i à j num horário t , tal que, t está entre as extremidades de um intervalo qualquer do período T .

No grafo G existem caminhos ligando alguns pares de vértices (os quais constituem linhas de ônibus no problema real). Seja $G_k[i, j]$ um subgrafo de G , formado por caminhos que ligam dois vértices i e j em G , e S_k um vetor de horários no período T que representa a partida do ônibus no início da linha, através da rota G_k , onde $G_k[p_i]$ é o tempo necessário para ir do início do trajeto G_k (o vértice i) até o ponto p_i , sendo este o ponto onde o usuário se encontra. Por fim, seja $v^t \in V$ um vértice de G por onde é permitido mudar de uma rota G_k para uma outra G_u (o que representa um terminal de integração no problema real), e seja $V^t = \{v^t\}$ o conjunto de todos estes vértices.

Um grafo G com esta estrutura será definido como *Grafo Rolo*.

3. CAMINHOS MÍNIMOS EM UM GRAFO ROLO - RoloCalc

O *RoloCALC* é um algoritmo desenvolvido com o intuito de calcular problemas de caminhos mínimos em um *Grafo Rolo*. É uma ferramenta essencial para o *AppROLO* que calcula a rota ótima entre dois pontos da cidade através de conexões entre linhas públicas de transportes em terminais de integração, considerando o grafo $G=(V, A)$ como a cidade, e cada caminho G_k de G como as linhas de transporte público, e V^T os terminais de integração em que é possível a troca de transporte (não necessariamente de um mesmo tipo). A partir daqui qualquer referência a um grafo G será levando em conta que este é um Grafo Rolo.

Para o problema em questão é necessário tomar um caso particular do Grafo Rolo. Para isso será definido o que seria uma *sequência de caminhos* e um *grafo rolo conexo por caminhos*.

Definição 1: Uma *sequência de caminhos* em um grafo rolo G é uma lista ordenada de caminhos G_k tomados para ir de um ponto i a um ponto j .

Definição 2: Um Grafo Rolo G é dito *conexo por caminhos* se dado qualquer par de vértices distintos i e j , então existe uma sequência de caminhos de i para j .

Por questões de conveniência cada caminho G_k e o conjunto V^T serão referidos como sendo linha e terminais, respectivamente.

Com todas as definições em mãos pode-se enunciar o algoritmo *RoloCALC*.

RoloCalc (Algoritmo)

Variáveis

- p_i := Ponto de partida;
- p_f := ponto de chegada (destino);
- V^T := Terminais de integração;
- V^R := Inícios (Terminais rotulados);
- $\text{Min}\{V^R\}$:= Início de menor rótulo (Terminal de menor rótulo);
- CV^T := Tempo de deslocamento calculado de um $i \in V^R$ (um ponto de início) até um $j \in V^T$;
- RV^R := Valor rotulado em $i \in V^R$ (horário em que chegou no terminal);
- L_i := Linhas de transporte de um ponto i ;
- E_{is} := Tempo de espera para uma linha L_i em um ponto inicial (Terminal rotulado);
- S_{L_i} := Saídas de L_i (horários de partida da linha);

Início

Todo $T \in V^T$ e p_f recebem rótulo de valor “infinito” ;

$V^R = \{p_i\}$;

$\text{pass}(G_k[p_i, p_f]) =$ Conjunto de rotas para ir de p_i à p_f ;

$R_{p_f} =$ Tempo para se deslocar de p_i à p_f ou rótulo de p_f ;

Enquanto $R_{p_f} > R(\{V^R\} - \{p_i\})$ faça (enquanto R_{p_f} for menor que todos os rótulos do conjunto V^R)

$\text{Min}\{V^R\}$ = início de menor rótulo ;

$i = \text{Min}\{V^R\}$;

Para cada linha que passa por i faça

Identifique a saída $s \in S_{L_i}$ apropriada e ótima para ir do início de L_i até i ;

Calcule E_{is} com respeito a linha L_i ;

$C_j := C_i + E_{is}$; (C_j = caminho calculado do começo de L_i até i), (j = ponto de para de L_i)

Caminhe de i até um $j \in V^T$ ou até $j = p_f$ usando L_i ;

Se parou em um $j \in V^T$ então

Se $R_j > C_j$ então

$R_j := C_j$;

Remova a passagem de j em $pass(G_k)$ e $\{pass(L_i[i, j])\} \rightarrow pass(G_k[p_i, p_f])$;
Se j não pertence à V^R então
 $V^R = V^R + \{j\}$;
Se Parou em $j=p_f$ então
Se $R_{p_f} > C_j$ então
 $R_{p_f} := C_j$;
 Remova a passagem de p_f em $pass(G_k)$ e $\{pass(L_i[i, j])\} \rightarrow pass(G_k[p_i, p_f])$
 $V^R = V^R - \{j\}$;
Se $R_{p_f} < R\{V^R\}$ então (se R_{p_f} for menor que o rótulos de todos os elementos de V^R então)
 Quebrar;
 Emita R_{p_f} e $pass(G_k[p_i, p_f])$
 Fim Algoritmo

Condição da Rota Ótima

Teorema:

Se i e j são vértices de um grafo de rolo conexo por circuito onde todas o conjunto das funções de incidência de cada aresta são todas positivas, a sequência de circuitos encontrada pelo RoloCALC de i até j é ótima.

(Demonstração)

A cada interação o algoritmo seleciona o terminal com menor rótulo e calcula todos os possíveis caminhos a partir deste, daí não sendo mais necessário para os cálculos visto que se qualquer outro terminal retorna a este vértice em um tempo posterior o rotulo dele será sempre maior, assim nunca voltando a rotular este. No momento em que o vértice final é o menor de todos os rótulos a mesma condição de aplicará a este e nenhuma outra rota será melhor.

4. MODELO COMPUTACIONAL E TESTES

O RoloApp se baseia no uso de estruturas para os pontos/terminais, linhas de ônibus e os caminhos (arestas) onde estas estão interligadas por meio de suas informações facilitando o acesso e o armazenamento de informações. No início de seu desenvolvimento percebeu-se que haveria inúmeros problemas ligados a sua estruturação da cidade como um conjunto de múltiplos grafos variantes (já que para cada linha haveria um grafo diferente e o fato de que várias linhas podem passar pelo mesmo ponto) que causaria um acúmulo muito grande de informação e dados armazenados na memória que tornaria o programa muito pesado e lento, quando fosse transferido para sua versão mobile. Para que o aplicativo voltasse a ser viável foi visto que deveria ser investido no uso de banco de dados e a transformação do aplicativo feito para o PC em um servidor, além de tomar as linhas como um subconjunto de grafos da própria cidade, daí surgiu o *Grafo Rolo*.

No sistema de banco de dados relacional, os dados são armazenados em tabelas, como a forma abaixo:

Pontos			
Nome	Código	Coordenadas	Ponto/Terminal (pont_term)

Linhas	
Nome	Código

Controle de caminhos (Contr_caminhos)		
Linha	Ponto inicial (PI)	Ponto Final (PF)

O mapa da cidade é posto em um plano cartesiano onde são conhecidas as coordenadas x e y de cada ponto e as linhas em sua estruturação só possuem o código de certos pontos de forma lógica e ordenada, ou seja, o controle de caminhos.

O modelo lógico do banco de dados relacional utilizado deve definir quais as tabelas e os nomes das colunas que as compõe. Logo o modelo pode ser definido da seguinte forma:

- Pontos (nome, código, coordenadas, pont_term);
- Linhas (nome, código);
- Contr_caminhos (linhas, pi, pf).

Foram realizados testes computacionais em uma cidade fictícia com 52 pontos de ônibus e 9 terminais de integração simulando trânsito em horários distintos para cada rua (aresta), onde cada linha de ônibus possuía uma velocidade fixa de 40 km/h que alterava conforme o local e o horário onde estava a percorrer. A priori houve falhas que foram úteis para o aperfeiçoamento do algoritmo e da estrutura computacional, após os reajustes todos os testes realizados foram bem-sucedidos.

5. CONCLUSÕES

Nesse trabalho foi visto uma nova teoria baseada no conceito de caminhos mais curtos dinâmicos, onde os custos associados aos arcos do grafo não são definidos em forma de constante, mas sim, são valores dinâmicos, o que é o caso real do problema de roteirização usando Linhas de ônibus. A metodologia possibilitou a criação do aplicativo RoloCalc que embora utiliza uma lógica similar ao algoritmo de Dijkstra, permite que os custos sejam definidos em função do tempo.

Após o término total do programa, isto é, sua versão para dispositivos móveis juntamente com um servidor, planeja-se integrá-lo a uma interface de mapeamento, como o Google API (freeware), tornando-o de fácil manuseio, deixando assim o software mais universal.

6. REFERÊNCIAS

- KARIMI, H. Finding Optimal Bus Service Routes: Internet-Based Methodology to Serve Transit Patrons. JOURNAL OF COMPUTING IN CIVIL ENGINEERING, Vol. 18, No. 2, p. 83-91, 1 Abr. 2004.
- REZENDE, R. Conceitos Fundamentais de Bancos de Dados. Disponível em: <<http://www.devmedia.com.br/conceitos-fundamentais-de-banco-de-dados/1649>> Data de Acesso: 12 de jun 2016.
- SHERAFAT, H.. Algoritmos Heurísticos de Cobertura de Arcos, Tese de Doutorado, PPGEP, Universidade Federal de Santa Catarina, 2004.

SHERAFAT, H.. Sistema Construtor de Rotas e sua Aplicação na Roteirização de Coleta de Lixo Domiciliar.
Revista GEINTEC: gestão, inovação e tecnologias, v. 3, p. 329-347, 2014.

OPTIMAL BUS ROUTING

Abstract. *The bus is the main transportation mean in most cities. In some cities there are integration terminals which facilitate the over switch from one line to an other at no additional cost for the passenger. To get around from one point to an other in general there are several combinations of bus lines to cover the trip, however, there is no clarity for the passenger that which combination minimizes the travel time. An application that could provide this information would be useful to every user of the public transportation system.*

In this paper we describe a methodology that permitted the development of a so-called RoloApp (Optimal Bus Routing Application), which identifies in real time the best sequence of bus lines, with over switches on integration terminals, in order to minimize the overall trip time, what includes the waiting time at terminals.

Keywords: *Bus, Shortest Path, Bus Routing, Optimization.*