

# ANÁLISE COMPARATIVA ENTRE O TRANSCAD E HEURÍSTICAS DE AGRUPAMENTO E DE ROTEAMENTO DE VEÍCULOS

**Maxwell Scardini Kaiser**

**Priscila de Almeida Prata**

**Glaidston Mattos Ribeiro**

Programa de Engenharia de Transporte - PET/COPPE/UFRJ

Programa de Pós-Graduação em Engenharia Urbana - Universidade Federal de São Carlos

Universidade Federal do Espírito Santo - Centro Universitário Norte do Espírito Santo - CEUNES

## RESUMO

O uso de ferramentas computacionais visando à solução dos mais variados problemas tem crescido nas empresas privadas e públicas, principalmente para reduzir os custos e melhorar a qualidade dos serviços prestados. Entre estas ferramentas, está o TransCAD que é um aplicativo especialmente projetado para transportes e que ainda apresenta uma linguagem de programação (GISDK) para inserção de novas funcionalidades no aplicativo. Este trabalho apresenta a implementação em GISDK de dois métodos que podem ser utilizados na roteirização de veículos, baseados na heurística de Gillet e Miller (1974) de Agrupar-Primeiro e Rotar-Depois. Para agrupar, foi implementada a heurística denominada  $k$ -Médias, e para rotear, o método 2-Opt. Os métodos forneceram bons resultados sendo que, em alguns casos, foram melhores do que os apresentados pelo próprio TransCAD.

## ABSTRACT

The use of the computational applications for solving several business problems has been growing in private and public companies, mainly for saving costs and for improving their services. Among these computational applications, we have TransCAD that is an application for transportation that also presents a programming language (GISDK) to insert new functionalities in that software. This paper presents the implementation in GISDK of two methods based in the heuristic of Gillet and Miller (1974). These methods can be used in vehicle routing problems. We implemented the heuristic called  $k$ -Means for clustering and 2-Opt for routing. These methods generated good results and when we compare them with the ones provided by TransCAD, in some cases, our methods provide better results.

## 1. INTRODUÇÃO

As recentes conquistas no mercado internacional e a estabilidade econômica são alguns dos fatores que tem impulsionado grandes avanços nas empresas brasileiras, de tal forma que uma das principais preocupações destas tem sido a Logística associada à questão computacional.

Os avanços computacionais das últimas décadas permitiram o surgimento de novas ferramentas computacionais, principalmente para o setor de Logística. Neste contexto, estão os SIG's (Sistemas de Informações Geográficas) que possuem integrados métodos eficientes de otimização, além de saídas gráficas como mapas (Câmara *et al.*, 1996). Dentre os diversos SIG's existentes, tem se destacado o TransCAD (Caliper, 1996) que é conhecido na literatura como sendo um SIG especialmente voltado para transportes. Alguns autores inclusive o classificam como um SIG-T (Sistema de Informações Geográficas para Transportes) (Silva e Waerden, 1997). O TransCAD também apresenta uma linguagem de programação conhecida como GISDK (*Geographic Information System Development's Kit*), não muito explorada na literatura.

Segundo Novaes (1989), o principal problema logístico associado a transporte é o de roteamento de veículos, conhecido na literatura como Problema de Roteamento de Veículos (PRV). Este problema tem sido bastante estudado por diversos autores os quais buscam melhorias visando uma maximização dos lucros e/ou minimização dos custos. Nas duas últimas décadas os avanços foram significativos, porém os problemas a serem considerados

também cresceram. Fatores como o crescimento urbano sem um planejamento adequado, crescimento econômico e principalmente, o surgimento e crescimento do comércio eletrônico, tem afetado a distribuição de cargas em áreas urbanas (Cunha, 2000).

Em um PRV clássico, veículos deixam um depósito e coletam mercadorias de um conjunto de clientes. Para facilitar a operacionalização, as empresas procuram agrupar os clientes em zonas segundo algum critério de similaridade, por exemplo, distância; e cada uma então é atendida por um veículo. Sob o ponto de vista teórico, essa estratégia é conhecida na literatura como Agrupar-Primeiro e Rotear-Depois (Gillet e Miller, 1974) e vem apresentando bons resultados.

Sendo assim, este trabalho apresenta uma implementação, em GISDK, de uma heurística de agrupamentos, conhecida como  $k$ -Médias, e de uma heurística de roteirização de veículos conhecida como 2-Opt, que parte do pressuposto que os clientes já estão agrupados por algum critério. Resultados computacionais mostram que os agrupamentos obtidos com o  $k$ -Médias são melhores que os definidos pelo TransCAD, assim como, as rotas criadas com a heurística 2-Opt.

O trabalho está assim dividido. Na Seção 2 são apresentados alguns conceitos sobre o PRV. As Seções 3 e 4 descrevem, respectivamente, as implementações do método  $k$ -Médias e do algoritmo 2-Opt, e na Seção 5 são apresentadas às conclusões e sugestões para trabalhos futuros.

## **2. PROBLEMA DE ROTEAMENTO DE VEÍCULOS**

O problema de roteirização de veículos vem sendo objeto de estudos desde a segunda metade do século passado, no entanto ainda não se conhece uma solução ótima para muitos problemas. Segundo Cunha (2000), alguns dos problemas que dificultam os estudos e conseqüentemente a aplicação das soluções são:

- Falta de conhecimento do problema o que implica em escolhas erradas (métodos, software, etc.);
- Soluções podem variar mesmo para dados e problemas similares e grande número de restrições, dificultando a aplicação de métodos de otimização.

Cunha (2000) apresenta ainda algumas variações do PRV como as propostas por Bodin *et al.* (1983). Basicamente, o PRV pode ser classificado como um problema de roteirização pura, que compõem uma derivação do Problema do Caixeiro Viajante (PCV) bastante difundido na literatura, e um problema de roteirização e programação (combinados) que envolvem outras variáveis como horário de entrega e/ou coleta, preferência de atendimento, etc. Dependendo do número de variáveis, estes problemas podem apresentar um alto grau de complexidade. Atualmente na literatura o maior problema do caixeiro viajante otimamente resolvido possui 24978 cidades, entretanto, foi utilizado um *cluster* com 96 computadores trabalhando aproximadamente 1 ano e 2 meses com um algoritmo de *cutting planes*. Para maiores detalhes, ver <http://www.tsp.gatech.edu/sweden/>.

No PRV clássico, consideram-se  $m$  clientes espacialmente distribuídos, cada um com uma demanda por produtos. Os produtos são entregues a partir de um depósito por uma frota de veículos homogêneos. Cada veículo realiza um percurso que envolve: sair do depósito,

entregar os produtos para um subconjunto de clientes, e retornar ao depósito. A rota de cada veículo deve obedecer a algumas restrições como: a quantidade de produtos entregue não deve exceder a capacidade do veículo e o tempo limite para realizar uma rota não deve ser ultrapassado.

Sendo assim, o PRV tem como objetivo definir as rotas para os veículos otimizando alguma função objetivo (Lorena, 2002). Uma rota compreende um subconjunto de clientes que devem ser atendidos em uma determinada ordem de custo mínimo. Dentre as diversas abordagens na literatura para o PRV, está a de Agrupar-Primeiro e Rotear-Depois (Gillet e Miller, 1974) que apresenta resultados interessantes (Goldbarg e Luna, 2000).

Conforme Goldbarg e Luna (2000), a heurística de Gillet e Miller (1974) procura obter a solução do problema em duas etapas distintas. A primeira serve para agrupar os pontos de demanda (formar o que se chama de *cluster*) segundo algum critério de proximidade, enquanto a segunda etapa define a sequência de atendimento dos clientes agrupados em cada *cluster*.

Como as fases dessa heurística são distintas, estas podem ser implementadas separadamente, gerando assim dois problemas: o de agrupamentos ou *clustering* e o de determinação das rotas. O segundo pode ser visto como um PCV.

## **2.1. Agrupar-Primeiro**

*Clusters* são agrupamentos de objetos com base nas similaridades e/ou diferenças entre os mesmos de tal maneira que os grupos obtidos sejam os mais homogêneos e bem separados possíveis, ou seja, procura-se coesão interna e isolamento externo (Andeberg, 1973). As técnicas de agrupamentos são utilizadas nas mais variadas áreas de pesquisas para finalidades diversas como (Viana e Negreiros, 2003): detecção de enfermidades, classificação de células sanguíneas, tratamento de imagens médicas (ultra-sonografia e tomografia, por exemplo), identificação de padrões de assinaturas, distinção de rostos, classificação de minerais, classificação de componentes químicos, reconhecimento de imagens de satélites, estudo de padrões de consumo, logística, estudo de tráfego, marketing e análises comportamentais.

Os métodos de agrupamentos podem ser divididos em Hierárquico e de Particionamento. Nos algoritmos clássicos de agrupamento hierárquico, os *clusters* são formados gradativamente através de aglomerações ou divisões de *clusters*, gerando uma hierarquia de *clusters*, normalmente representada através de uma estrutura em árvore. Nesta classe de algoritmos, os dados de uma forma geral são formados por um *cluster* que por sua vez é composto de vários *clusters* menores (Oliveira, 2005). Por outro lado, os métodos de particionamento procuram encontrar a melhor partição dos  $m$  objetos (clientes, neste trabalho) em  $k$  grupos. Conforme Neves *et al.* (2001), normalmente os  $k$  *clusters* encontrados neste método são de melhor qualidade do que os gerados pelos métodos hierárquicos. Dentre os algoritmos que realizam agrupamentos está o  $k$ -Média ou *k-Means*.

Os estudos iniciais do algoritmo de  $k$ -Médias foram feitos por Macqueen (1967). Para Huang (1998), é o algoritmo mais conhecido pela sua eficiência em realizar agrupamentos em bases de dados numerosas. É um algoritmo de aprendizado não-supervisionado e vem sendo utilizado em reconhecimento de padrões e mineração de dados (*Data Mining*).

O  $k$ -Médias é, basicamente, representado pelo seguinte modelo:

$$R = \sum_{k=1}^K \sum_{x_i \in C_k} d(x_{0,k}, x_j) \quad (1)$$

Onde:

- $x_{0,k}$  é o centróide do *cluster*  $C_k$ ;
- $d(x_{0,k}, x_j)$  é a distância entre o centro do *cluster* e o cliente  $j$ ;
- $K$  é o número total de *clusters*; e
- $R$  é a soma dos resultados obtidos por cada cluster.

Então, procura-se definir os clusters  $C_k$ 's de tal maneira que a função (1) seja minimizada. Outros algoritmos de agrupamentos são encontrados descritos amplamente nos trabalhos de Viana e Negreiros (2003) e Neves (2001).

## 2.2. Rotear-Depois

Segundo Cunha *et al.* (2002), o PCV pode ser definido como o problema de encontrar o roteiro de menor distância ou custo que passa por um conjunto de cidades, sendo que cada cidade deve ser visitada exatamente uma vez.

Existem vários métodos heurísticos para resolver um PCV. De uma forma geral, esses métodos podem ser agrupados em duas categorias:

- Métodos de construção de roteiros;
- Métodos de melhoria de roteiros.

Segundo Cunha *et al.* (2002), a construção do roteiro pode se dar através do:

- Método do vizinho mais próximo, no qual o caixeiro inicia em uma cidade qualquer e então segue para a cidade mais próxima; a partir desta, busca-se a cidade mais próxima ainda não visitada e assim sucessivamente até que todas as cidades sejam visitadas, retornando então à cidade de origem. Esse método pode ser classificado como um método guloso ou míope; e
- Método de inserção, no qual, partindo-se de um roteiro inicial com apenas duas cidades, e considerando todas as demais cidades ainda não incluídas no roteiro, seleciona-se para inserção no roteiro parcial sendo construído, aquela que atenda um determinado critério. Esse procedimento é repetido sucessivamente, com a análise da inserção entre cada par de cidades do roteiro parcial, até que todas as cidades sejam inseridas no roteiro.

Os métodos de melhoria partem de uma solução obtida com o auxílio de outro método, e procura aperfeiçoar o resultado, utilizando para isso, uma sistemática predefinida. Um dos métodos que realiza tal procedimento é conhecido como  $k$ -Opt (Lin e Kernighan, 1973). O método 2-Opt testa as trocas possíveis entre pares de arcos, refazendo as conexões quando houver uma melhoria no roteiro. O processo termina quando não for possível realizar nenhuma troca que resulte em melhoria. Cunha *et al.* (2002) afirma que, quanto maior o valor de  $k$ , melhor a precisão do método, porém maior também é o esforço computacional. Na prática, são considerados os métodos 2-Opt e 3-Opt.

A seguir são mostradas as etapas para resolução do método 2-Opt conforme Novaes (2004):

**Etapla 1.** Inicia-se com um roteiro qualquer, gerado de forma aleatória, ou por algum método de construção.

**Etapla 2.** Remove-se 2 arcos do roteiro. Tenta-se re-conectá-los de modo que se tenha uma nova rota. Se essa nova rota produzir um resultado melhor, isto é, gerar uma rota com custo menor que a anterior, substitui-se o roteiro inicial pelo novo roteiro e repete-se a Etapa 2. Caso contrário continua-se com o roteiro anterior e tentam-se outros dois arcos, repetindo a Etapa 2.

**Etapla 3.** O processo termina quando não se conseguir nenhuma melhoria.

### 3. IMPLEMENTAÇÃO DOS MÉTODOS *k*-MÉDIAS E 2-OPT

As implementações aqui desenvolvidas foram realizadas em GISDK com o objetivo de comparar os agrupamentos e roteamentos criados com os produzidos pelo TransCAD. Para *clustering*, foram implementados dois métodos: o *k*-Médias com e sem restrição de capacidade. O *k*-Médias com restrição de capacidade tem haver com a questão da capacidade de carga do veículo, ou seja, os *clusters* devem ser formados de tal maneira que a soma das demandas não exceda a capacidade do veículo que deverá servir (coletar ou entregar mercadorias) aquele conjunto de clientes.

#### 3.1. Algoritmo *k*-Médias sem restrição de capacidade

Para este caso os clientes são alocados a cada grupo sem considerar nenhuma restrição de capacidade para o cluster. Esta implementação consiste dos seguintes passos básicos do *k*-Médias, como mostra o pseudocódigo a seguir:

```
Algoritmo k-médias sem restrição de capacidade
// k é o número de clusters a serem gerados
Determine k
Escolha aleatória da posição do centro dos k clusters.
Enquanto Atualiza=1 faça
    Para i←1 até i=número de clientes faça
        Para k←1 até número de clusters faça
            Calcule a distância entre i e k;
        Fim para;
        Associe i ao centro k próximo;
    Fim Para;
    Enquanto t <= número de clusters faça
        Recalcula Centro;
        Se (nova posição do centro - posição atual) > Estabiliza
            Atualiza=1;
        Fim Se;
        Senão
            Atualiza=0;
        Fim Senão;
    Fim Enquanto;
Fim Enquanto;
Fim k-médias sem restrição de capacidade
```

#### 3.2 Algoritmo *k*-Médias com restrição de capacidade

Neste caso, foi considerado que a frota de veículos é homogênea (capacidade igual para todos os veículos) e que os clientes possuem demandas variáveis. A seguir são descritos, em pseudocódigo, os passos do *k*-Médias com restrição de capacidade, sendo que, o critério de estabilização pode variar de acordo com o problema.

```
Algoritmo k-médias com restrição de capacidade
// k é o número de clusters a serem gerados
// C é a capacidade máxima de cada cluster
```

```

// Estabiliza é um valor pequeno utilizado para mover ou não o centro de um cluster
Determine k
Determine C
Se (C*k) >= Soma das demandas dos clientes então
    Escolha aleatória da posição de k.
    Enquanto Atualiza-1 faça
        Para i=1 até número de clientes faça
            Para k=1 até número de clusters faça
                Calcule a distância entre i e K;
            Fim para;
            Associe i ao centro k mais próximo;
        Fim Para;
        Para k=1 até número de clusters faça
            Enquanto total demanda do cluster k > capacidade cluster k faça
                Para todo cliente (i) associado ao cluster k faça
                    Para n=1 até n=número de cluster e n ≠ k faça
                        Se (total demanda do cluster n + demanda do cliente i) <=
                            capacidade do cluster n então
                            Calcule a distância entre cliente i e o cluster n;
                            Associe cliente (i) mais próximo do centro n;
                            Desaloque cliente (i) do cluster k;
                        Fim Se.
                    Fim Para.
                Fim Para.
            Fim Enquanto.
        Fim Para.
        Enquanto k <= número de clusters faça
            Recalcular Posição do Centro k
            Se (nova posição - posição atual) > Estabiliza
                Atualiza-1;
            Fim Se;
            Senão
                Atualiza-0;
            Fim Senão;
        Fim Enquanto;
    Fim Enquanto;
Resultado dos clusters formados.
Fim Se
Fim k-médias com restrição de capacidade

```

### 3.3 Heurística 2-Opt

A heurística 2-Opt é um algoritmo de melhoria de roteiros, portanto necessita de um roteiro inicial para, a partir dele, melhorar os resultados. Para a construção da rota inicial, foi escolhido o método do vizinho mais próximo. A heurística 2-Opt foi formulada com base na troca dos vértices (clientes) dos arcos que compõe a rota. A seguir está descrito o pseudocódigo do algoritmo.

```

TROCA 2-Opt
RotaInicial ← Obtém rota inicial
Melhora ← 1
Enquanto Melhora = 1
    Melhor ← null
    CustoInicial ← custo rota inicial
    Para j ← 2 até tamanho da rota - 1 faça
        Para k ← j+1 até tamanho da rota - 1 faça
            Nrota ← Troca(j,k)
            CustoNrota ← custo nova rota
            Se CustoInicial > CustoNrota então
                CustoInicial ← CustoNrota
                Melhor ← Nrota
            Fim Se
        Fim Para
    Fim Para
    Se Melhor <> null então
        RotaInicial ← Melhor
    Senão
        Melhora ← 0
    Fim Se
Fim Enquanto

```

Onde:

- RotaInicial - É a rota obtida pelo método do vizinho mais próximo;
- CustoInicial - É o custo da rota inicial;
- Nrota - É a nova rota gerada a partir da troca de dois clientes na rota;
- Melhor - Guarda a troca que gerou o menor custo entre todas as trocas possíveis para uma rota.

### 3.4. Interface da aplicação

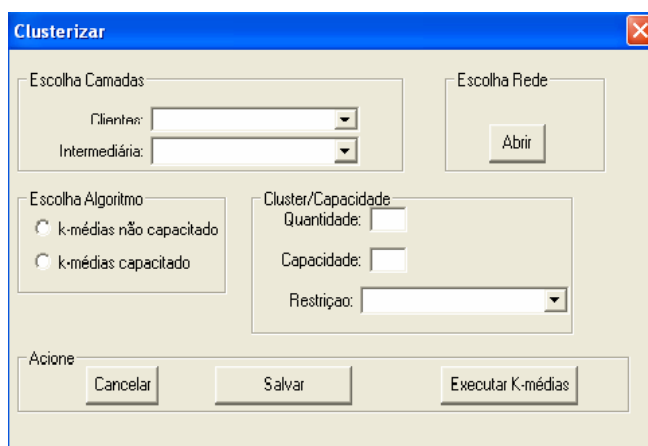
A interface com o usuário é feita através do componente de recurso conhecido por *Dbox* do GISDK, e é responsável por obter as informações necessárias para a execução da aplicação. A seguir são apresentadas as interfaces implementadas.

#### 3.4.1. K-Médias

Para executar a aplicação, o usuário deve escolher (Veja Figura 1):

- Camada *Clientes*: que é a camada dos dados com as características a serem agrupados;
- Camada *Intermediária*: que representa o Sistema Viário;
- A opção “Escolher a Rede” permite o usuário escolher a rede que foi gerada para o mapa aberto. Esse é um dado necessário no TransCAD;
- Qual algoritmo será executado: k-médias capacitado ou não capacitado;
- Se escolher o caso capacitado, o usuário deverá inserir, além da quantidade de *clusters*, a capacidade a ser considerada e escolher por qual característica deseja limitar o *cluster* (demanda, tempo de viagem, etc) que deve ser selecionado através do campo *Restrição*;
- No botão “Salvar” é escolhido um local para salvar um arquivo com as informações obtidas na geração dos clusters.

Com os dados citados acima, o botão “Executar K-médias” permite aplicar os algoritmos descritos neste trabalho. O resultado é apresentado no mapa, ou seja, cada cluster definido recebe uma cor diferenciada, assim como um símbolo.



**Figura 1:** Interface da Aplicação.

#### 3.4.2. 2-Opt

A tela inicial (ver Figura 2) é a responsável por obter as informações necessárias para a execução da aplicação. O usuário deve informar:

- Camada *Clientes*: a camada que representa os clientes já agrupados por seleções, estas seleções devem iniciar com a palavra “cluster”;
- Camada *Depósito*: camada que possui o depósito de onde partem as rotas, com a seleção do ponto que corresponde ao depósito, que deve iniciar com a palavra “deposito”;
- Camada *Linha*: a camada que representa o sistema viário.

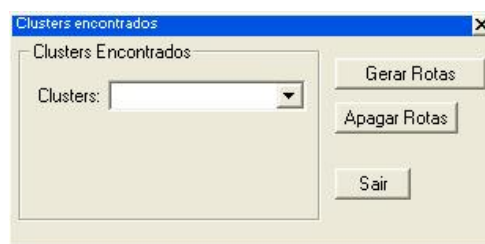
Após o preenchimento das informações acima, o botão “Escolher Rede” é habilitado para que o usuário escolha a rede que foi gerada para o mapa a ser utilizado na aplicação. A rede é necessária, pois o GISDK calcula o menor caminho a partir deste arquivo matemático. Ela é criada com base na camada de linhas e representa todas as ruas e avenidas do mapa, com os respectivos sentidos de tráfego (mão e contramão).

Com essas informações preenchidas o usuário deve solicitar que elas sejam confirmadas, e se tudo estiver correto, é mostrada uma janela como a da Figura 3.

A aplicação gera como saída o desenho da rota no mapa e um arquivo TXT, que é aberto no Bloco de Notas do *Windows*, mostra qual a ordem em que os clientes deverão ser visitados e o custo final do roteiro em quilômetros.



**Figura 2:** Configurações necessárias para aplicação.



**Figura 3:** Caixa de Ferramentas para o cálculo das rotas.

#### 4. RESULTADOS COMPUTACIONAIS

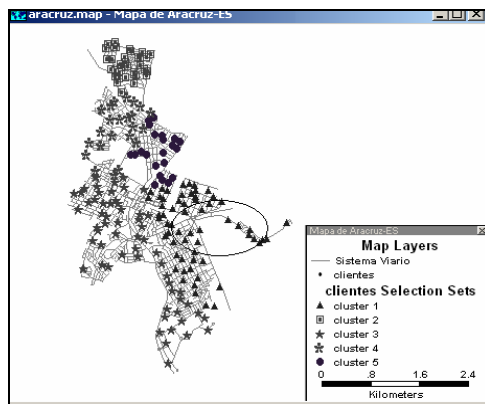
Os testes foram feitos utilizando o mapa da cidade de Aracruz-ES que é composto de vias, clientes e um depósito. Os algoritmos utilizados para criar os *clusters* geralmente utilizam distância Euclidiana como medida de similaridade entre os clientes, entretanto, neste trabalho, considerou-se a distância real da rede viária obtida através de comandos em GISDK.

Com o *k*-Médias sem restrição de capacidade observou-se que os *clusters* criados não possuem “uniformidade”, existe certa “dispersão”. A falta de equilíbrio entre os *clusters* é proporcional ao número de *clusters*, ou seja, quanto maior o número de *clusters*, mais dispersos ficam os *clusters*, alguns maiores que outros, como pode ser observado na Figura 4 que mostra o resultado para o caso de cinco *clusters* e 223 clientes. A Figura 5 mostra o resultado apresentado pelo TransCAD. Ao comparar os resultados, visualmente percebe-se que o *k*-Médias fornece resultados mais interessantes que o TransCAD (veja círculos nas Figuras 3 e 4).

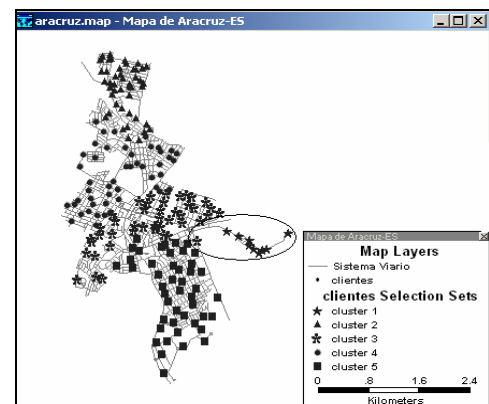


Testes aleatórios foram realizados para comparar os resultados gerados pelo  $k$ -Médias com os do TransCAD. A Tabela 1 apresenta os principais resultados. As colunas da tabela referem-se a:

- Clientes: o número de clientes inseridos no mapa;
- *Clusters*: o número de clusters definido pelo usuário;
- Custo TransCAD: representa o resultado obtido pelo TransCAD. Este custo é a soma da distância em quilômetros de cada cliente ao centro do *cluster*;
- $k$ -Médias NC: Apresenta um custo médio de 10 soluções geradas utilizando o método  $k$ -Médias não capacitado. Isso porque os centros iniciais dos  $k$  *clusters* são definidos aleatoriamente. Este custo é a soma da distância em quilômetros de cada cliente ao centro do *cluster*;
- Diferença (%): É a diferença percentual em módulo do resultado do  $k$ -médias em relação ao do TransCAD, ou seja:  $|(Custo K-Médias - Custo TransCAD)| / (Custo TransCAD)$ .



**Figura 4:** *Clusters* gerados pelo  $k$ -Médias sem restrição de capacidade.



**Figura 5:** *Clusters* gerados pelo TransCAD sem restrição de capacidade.

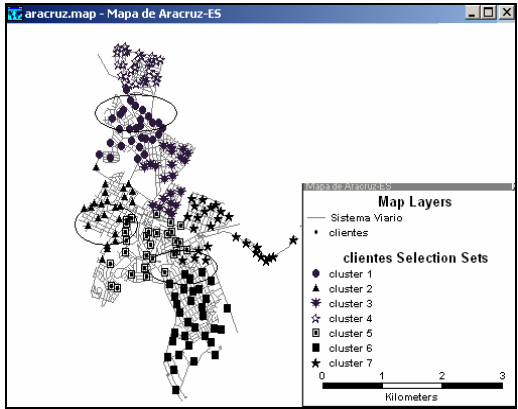
**Tabela 1:** Resultados sem restrição de capacidade.

INSTÂNCIA	CLIENTES	CLUSTERS	TRANSCAD	K-MÉDIAS NC	DIFERENÇA (%)
I-1	82	5	38,97	38,63	0,88
I-2	82	7	32,47	31,72	2,36
I-3	223	5	163,81	157,00	4,34
I-4	223	7	130,48	126,60	3,06
I-5	223	10	98,90	106,81	7,41
I-6	223	15	79,10	80,30	1,49

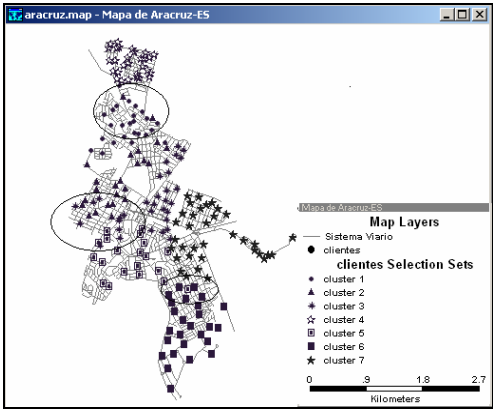
Percebe-se pela Tabela 1, que os resultados são promissores. O  $k$ -Médias apresenta resultados melhores para as instâncias I-1, I-2, I-3 e I-4. Para a instância I-5 o resultado foi 7,41% superior e para I-6 os resultados são bem similares.

Com o  $K$ -Médias com restrição de capacidade observou-se que os *clusters* estão mais coesos (uniformes). Isto pode ser observado na Figura 6 que mostra o resultado para o caso de 223 clientes agrupados em sete *clusters*. A demanda desses clientes foi criada aleatoriamente entre 1 e 10. A Figura 7 mostra os *clusters* gerados pelo TransCAD. Nestas figuras foram ressaltadas duas áreas (ver círculos). Note que o TransCAD apresenta alguns *clusters* “misturados”, gerando agrupamento não bem delimitados.

A Tabela 2 mostra os resultados obtidos utilizando métodos com restrição de capacidade (coluna *k*-Médias C) para as instâncias da Tabela 1. Foi considerado que, a capacidade somada de todos os *clusters* é igual à demanda total de todos os clientes acrescida em 10%.



**Figura 6:** Clusters gerados pelo método *k*-Médias com restrição de capacidade.



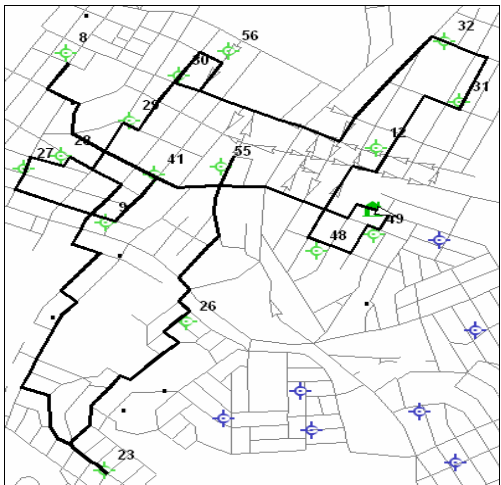
**Figura 7:** Clusters capacitados gerados pelo TransCAD.

Olhando a Tabela 2, percebe-se novamente que o *k*-médias apresenta um bom desempenho. Nesse caso, em todas as instâncias o *k*-Médias apresentou resultados melhores que o TransCAD.

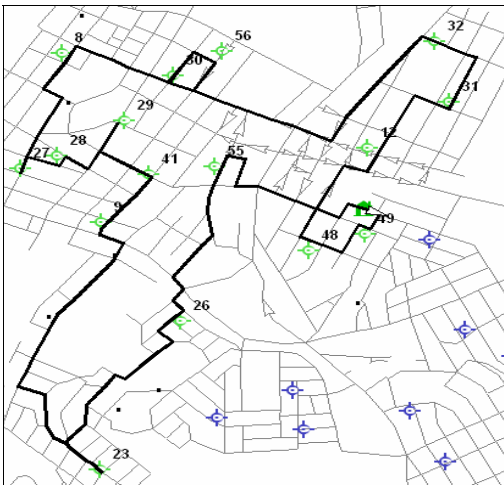
**Tabela 2:** Resultados com restrição de capacidade.

INSTÂNCIA	CLIENTES	CLUSTERS	CUSTO TRANSCAD	CUSTO K- MÉDIAS C	DIFERENÇA (%)
I-1	82	5	40,84	39,11	4,42
I-2	82	7	36,84	36,36	1,32
I-3	223	5	148,94	141,56	5,21
I-4	223	7	140,47	127,83	9,89
I-5	223	10	136,76	120,35	13,64
I-6	223	15	107,97	94,90	13,77

A partir dos clientes agrupados, foram realizados experimentos com a heurística 2-Opt implementada neste trabalho. Foram realizadas comparações entre: a rota inicial, a rota obtida após a execução da troca 2-Opt, e a rota criada pelo próprio TransCAD.



**Figura 8:** Rota inicial para um *cluster* com 16 clientes.



**Figura 9:** Rota final para um *cluster* com 16 clientes.

Na Figura 8 é mostrada a rota inicial com a heurística do vizinho mais próximo e na Figura 9 a rota final após a troca 2-Opt para um *cluster* com 16 clientes. Repare no comportamento da rota próximo dos clientes 28 e 41. Além disso, pode ser observado que a distância da total da rota foi reduzida. A rota inicial apresentou uma distância percorrida de 8,97 km e a rota final 7,72 km, uma redução de aproximadamente 13,94%.

A Tabela 3 mostra os resultados encontrados para o caso capacitado. Pode-se observar que a heurística melhora os resultados iniciais e é comparável com os resultados do TransCAD.

**Tabela 3:** Resultados encontrados para alguns *clusters* gerados para o caso capacitado.

Nº DE CLIENTES	VIZINHO MAIS PRÓXIMO	TROCA 2- OPT	TRANSCAD	DIFERENÇA 2-OPT E VIZINHO MAIS PRÓXIMO (%)	DIFERENÇA TRANCAD E 2- OPT (%)
34	15,58	14,68	13,94	5,79	5,04
48	25,47	24,95	23,34	2,06	6,45
20	11,57	10,72	9,57	7,38	10,69
15	7,42	7,35	5,82	0,94	20,83
6	7,93	7,47	7,47	5,81	0,00
13	5,22	5,08	5,00	2,69	1,55
9	6,27	5,88	5,80	6,23	1,35
18	11,14	11,14	10,73	0,00	3,63
14	5,88	5,85	5,32	0,61	9,00
50	27,96	24,79	22,13	11,34	10,72
60	21,88	20,12	17,29	8,04	14,05

## 5. CONCLUSÕES

Este trabalho apresentou técnicas para agrupamentos capacitados e não capacitados que, juntamente com um algoritmo para o PCV, permite resolver o problema de roteamento de veículos usando a estratégia de “Agrupar-Primeiro e Rotear-Depois”. Uma interface gráfica foi desenvolvida utilizando o software TransCAD e a linguagem de programação GISDK.

Observou-se que os *clusters* gerados pelo método *k*-Médias foram, na maioria das vezes, melhores que os gerados pelo TransCAD. Foi possível observar também que apesar da heurística 2-Opt se tratar de uma heurística simples, apresentou bons resultados se comparados com os resultados do próprio TransCAD com os *clusters* gerados. Isto deixa claro que com heurísticas mais elaboradas ou com o uso de alguma metaheurística, resultados melhores podem ser conseguidos frente ao TransCAD.

Como o TransCAD vem sendo utilizado constantemente nos órgãos públicos e privados, acredita-se que este trabalho vem contribuir no sentido de incentivar o desenvolvimento de novas funcionalidades no software, conforme necessidades dos usuários. Novos algoritmos mais eficientes que os embutidos no TransCAD podem ser implementados em GISDK, promovendo um ganho maior para o órgão.

## REFERÊNCIAS

- Andeberg, M.R. (1973) *Cluster analysis for applications*. New York: Academic Press.
- Bodin, L. D.; Golden, B. L.; Assad, A.; Ball, M. (1983) Routing and Scheduling of Vehicles and Crews. *The state of the Art. Comp. Opns. Res.* 10, 69-211.
- Caliper (1996) *TransCAD Transportation GIS Software*. User's Guide Version 3.0 for Windows. Caliper Corporation, Newton, EUA.

- Câmara, G.; Casanova, M. A.; Hemerly, A. S.; Magalhães, G. C.; Medeiros, C. M. B. (1996) *Anatomia de Sistemas de Informações Geográficas*. Campinas: Instituto de Computação, UNICAMP.
- Cunha, C.B. (2000) Aspectos práticos da aplicação de modelos de roteirização de veículos a problemas reais. *Revista Transportes da ANPET – Associação Nacional de Pesquisa e Ensino em Transportes*, v.8, n.2, p.51-74, novembro 2000.
- Cunha, C.B.; Bonasser, U.O.; Abrahão, F.T.M. (2002) Experimentos Computacionais com Heurísticas de Melhorias para o Problema do Caixeiro Viajante. *XVI Congresso da Anpet – Associação Nacional de Pesquisa e Ensino em Transportes*. Natal.
- Gillet, B.E; Miller, L.R. (1974) A heuristic algorithm for the vehicle dispatch problem. *Operations Research* 22, 240-349.
- Goldbarg, M.C.; Luna, L.; Pacca H. (2000) *Otimização Combinatória e Programação Linear - Modelos e Algoritmos*. 6ª edição. Rio de Janeiro. Campos.
- Huang, Z. (1998) Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data Mining and Knowledge Discovery*, Kluwer Academic Publishers, Hingham, MA, USA, Vol. 2, Issue 3. pp. 283-304.
- Lin, S.; Kerningham, B.W. (1973) An Effective Heuristic Algorithm for the Traveling- Salesman Problem. *Operation Research*, vol. 21, pp. 498-516.
- Lorena, L.A.N. (2002) Análise Espacial de Redes em Sistemas de Informações Geográficas. *LAC – Laboratório Associado de Computação e Matemática Aplicada. INPE – Instituto Nacional de Pesquisas Espaciais*. São José dos Campos. Disponível em: <http://www.lac.inpe.br/~lorena/producao/Analiseredes.pdf>. Acessado em 28/06/2007.
- Macqueen, J. (1967) Some methods for classification and analysis of multivariate observations. In *5th Berkeley Symposium on Mathematical Statistics*, eds L. Le Cam and J. Neymann, p. 281-297, Berkeley, California.
- Neves, M. C.; Freitas, C. C.; Câmara, G. (2001) Mineração de Dados em Grandes Bancos de Dados Geográficos. INPE. *Relatório Técnico-CTBRASIL*.
- Novaes, A.G. (1989) *Sistemas Logísticos: Transporte, Armazenagem e Distribuição de Produtos*, Edgard Blucher, São Paulo.
- Novaes, A.G. (2004) *Logística e Gerenciamento da Cadeia de Distribuição: Estratégia, Operação e Avaliação*. Campus e Elsevier, 2ª edição.
- Oliveira, A. L. R. (2005) *Otimização de Recebimento e Distribuição em Unidades Armazenadoras de Soja*. Dissertação de Mestrado ACPM-CET Universidade Federal do Paraná (UFP).
- Silva, A.N. R. e Waerden, P. (1997) First Steps with a Geographic Information System for Transportation. *Eindhoven University of Technology*, Netherland.
- Viana, J.F.R.; Negreiros, M.J.G. (2003): *Scluster: Um sistema para análise multivariada de agrupamentos de dados em larga escala*. Monografia Ciência da Computação-UECE. Disponível em: <http://www.lcc.uece.br/~negreiro/artigos/artdownload/monografias/MonografiaFred.pdf>. Acessado em 28/06/2007.

---

Maxwell Scardini Kaiser (scardinikaiser@gmail.com)

Programa de Engenharia de Transporte - PET/COPPE/UFRJ

Priscila de Almeida Prata (pricilaprata@gmail.com)

Programa de Pós-Graduação em Engenharia Urbana - Universidade Federal de São Carlos

Glaydston Mattos Ribeiro (glaydston@terra.com.br)

Universidade Federal do Espírito Santo - Centro Universitário Norte do Espírito Santo - CEUNES