

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE CONSTRUÇÃO CIVIL
CURSO DE ENGENHARIA DE PRODUÇÃO CIVIL

LUIS GUSTAVO PEREIRA

**SIMULAÇÃO ESTOCÁSTICA E MÉTODOS HEURÍSTICOS PARALELIZADOS
PARA A RESOLUÇÃO DO PROBLEMA DE ROTEAMENTO DE VEÍCULOS
CAPACITADOS COM BASE NA ESTRATÉGIA “CLUSTER FIRST ROUTE
SECOND”**

TRABALHO DE CONCLUSÃO DE CURSO

CURITIBA
2012

LUIS GUSTAVO PEREIRA

**SIMULAÇÃO ESTOCÁSTICA E MÉTODOS HEURÍSTICOS PARALELIZADOS
PARA A RESOLUÇÃO DO PROBLEMA DE ROTEAMENTO DE VEÍCULOS
CAPACITADOS COM BASE NA ESTRATÉGIA “CLUSTER FIRST ROUTE
SECOND”**

Trabalho de Conclusão de Curso de graduação, apresentado à disciplina de Projeto Final 2, do Curso Superior de Engenharia de Produção Civil do Departamento Acadêmico de Construção Civil – DACOC – da Universidade Tecnológica Federal do Paraná – UTFPR, como requisito parcial para obtenção do título de Engenheiro.

Orientador: Profa. MSc. Lilian Caroline Xavier Cândido

Co-Orientador: Prof. MSc. Thiago André Guimarães

CURITIBA

2012

FOLHA DE APROVAÇÃO

SIMULAÇÃO ESTOCÁSTICA E MÉTODOS HEURÍSTICOS PARALELIZADOS PARA A RESOLUÇÃO DO PROBLEMA DE ROTEAMENTO DE VEÍCULOS CAPACITADOS COM BASE NA ESTRATÉGIA “CLUSTER FIRST ROUTE SECOND”

Por

LUIS GUSTAVO PEREIRA

Trabalho de Conclusão de Curso apresentado ao Curso de Engenharia de Produção Civil, da Universidade Tecnológica Federal do Paraná, defendido e aprovado em 19 de junho de 2012, pela seguinte banca de avaliação:

Profa. Orientadora – MSc. Lilian Caroline Xavier Cândido, MSc.

UTFPR

Prof. Co-orientador – MSc. Thiago André Guimarães, MSc.

FAE

Prof. Cassius Tadeu Scarpin, Dr.

UFPR

Prof. Moacyr Aristeu Molinari Neto, MSc.

UTFPR

Prof. José Manoel Caron, MSc.

UTFPR

RESUMO

PEREIRA, Luis G. Simulação Estocástica e Métodos Heurísticos paralelizados para a Resolução do Problema de Roteamento de Veículos Capacitados com base na Estratégia “Cluster First Route Second”. Trabalho de Conclusão de Curso, Universidade Tecnológica Federal do Paraná. Curitiba, 2011.

O Problema de Roteamento de Veículos Capacitados (PRVC) busca determinar um conjunto de rotas para uma frota homogênea de veículos, partindo de um depósito central com destino a um conjunto de clientes que demandam determinado produto. Por ser um dos mais importantes e complexos problemas afetos à logística de distribuição, alternativas heurísticas para o problema são constantemente apresentadas. Nesse sentido, este trabalho endereça a uma abordagem heurística baseada na estratégia “*Cluster First Route Second*” que consiste primeiramente em agrupar os clientes conforme a demanda para posteriormente construir rotas para os grupos formados. A abordagem emprega Simulação de Monte Carlo em conjunto com o clássico algoritmo de GILLET e JOHNSON, 1976 para a resolução do problema do agrupamento, enquanto que as rotas para cada grupo são construídas pela heurística do vizinho mais próximo com refinamentos advindos de trocas 2-opt. Empregaram-se ainda técnicas de processamento paralelo para redução do tempo de processamento computacional.

Palavras-chave: Roteamento de veículos capacitados; Procedimentos heurísticos; Agrupamento de pontos de demanda, Simulação de Monte Carlo

ABSTRACT

PEREIRA, Luis G. Simulação Estocástica e Métodos Heurísticos paralelizados para a Resolução do Problema de Roteamento de Veículos Capacitados com base na Estratégia "Cluster First Route Second". 2011. Trabalho de Conclusão de Curso, Universidade Tecnológica Federal do Paraná. Curitiba, 2011.

The Capacitated Vehicle Routing Problem (CVRP) seeks to determine a set of routes for a homogeneous fleet of vehicles, from a central warehouse bound to a set of customers who demand a certain product. As one of the most important and complex problems related to the distribution logistics, heuristic approaches to this problem are constantly presented. In this way, this work addresses the heuristic approach based on a strategy "Cluster First Route Second" which consists primarily of grouping customers according to their demands for later build routes for the formed groups. The approach uses Monte Carlo simulation in conjunction with the classic algorithm Gillett and Johnson, 1976 to solve the clustering problem, while the routes for each group are constructed by the nearest neighbor heuristic with refinements coming from 2-opt exchanges. It is also employed parallel processing techniques to reduce the computational time.

Keywords: Capacitated vehicle routing, heuristic procedures; Clustering of demand points, Monte Carlo Simulation

LISTA DE SIGLAS

- LINQ *Language Integrated Query*
- OVRP *Open Vehicle Routing Problem*
- PCV Problema do Caixeiro Viajante
- PLINQ *Parallel Language Integrated Query*
- PRVC Problema de Roteamento de Veículos Capacitados
- UI *User Interface* (Interface do Usuário)
- VMP Vizinho mais próximo

LISTA DE FIGURAS

Figura 1 – Exemplo de codificação paralela	13
Figura 2 - Exemplo de uso de PLINQ	13
Figura 3 - Instância exemplo do PRVC	16
Figura 4 - Implementação do Monte Carlo	19
Figura 5 - Algoritmo de Gillett e Johnson para agrupamento	20
Figura 6 - Algoritmo VMP	20
Figura 7 - Troca 2-opt entre os trechos 1-2 e 3-4	21
Figura 8 - Heurística proposta para a resolução do PRVC	22
Figura 9 - Pseudocódigo representando encadeamento de algoritmos	22
Figura 10 - Interface do usuário para o software	24
Figura 11 - Gráfico de monitoramento Custo x Iteração	25
Figura 12 - Uso dos núcleos computacionais	26
Figura 13 - Tabela com as rotas encontradas para os veículos	27
Figura 14 - Solução gráfica 2D para as rotas dos veículos	27
Figura 15 - Solução gráfica para a instância E-n22-k4	32
Figura 16 - Solução gráfica para a instância E-n23-k3	32
Figura 17 - Solução gráfica para a instância P-n16-k8	33
Figura 18 - Histograma da distribuição de contagem de soluções e seus respectivos desvios do ótimo	35

LISTA DE TABELAS

Tabela 1 - Instâncias utilizadas para avaliação da heurística proposta	23
Tabela 2 - Resultados para as instâncias da classe A.....	28
Tabela 3 - Resultados para as instâncias da classe B.....	29
Tabela 4 - Resultados para as instâncias da classe E	29
Tabela 5 - Resultado para a instância da classe G	30
Tabela 6 - Resultados para as instâncias da classe M.....	30
Tabela 7 - Resultados para as instâncias da classe P	30
Tabela 8 - Resultado para a instância da classe V	31
Tabela 9 - Desvios médios obtidos na resolução do PRVC	31

SUMÁRIO

1.	INTRODUÇÃO.....	7
1.1.	JUSTIFICATIVA.....	7
1.2.	OBJETIVOS	8
1.3.	METODOLOGIA.....	8
2.	REFERENCIAL TEÓRICO	10
2.1.	O PROBLEMA DE ROTEAMENTO DE VEÍCULOS CAPACITADOS.....	10
2.2.	A ESTRATÉGIA “CLUSTER FIRST ROUTE SECOND”	11
2.3.	PROCESSAMENTO PARALELO.....	11
2.3.1.	Manipulação de conjuntos de forma paralela	13
2.4.	ESTRATÉGIAS PARA A RESOLUÇÃO DO PRVC.....	14
2.5.	SIMULAÇÃO DE MONTE CARLO.....	15
3.	MATERIAIS E MÉTODOS	16
3.1.	DESCRIÇÃO FORMAL DO PRVC.....	16
3.2.	ESTRATÉGIA DE RESOLUÇÃO DO PRVC.....	17
3.2.1.	O Problema das p-medianas	18
3.2.2.	O Problema da designação de grupos (clusterização).....	19
3.2.3.	O Problema da Roteirização: Geração de uma Solução Inicial	20
3.2.4.	O Problema da Roteirização: Melhoria da Solução Inicial	21
3.3.	HEURÍSTICA DESENVOLVIDA	22
4.	RESULTADOS E DISCUSSÕES.....	23
4.1.	INSTÂNCIAS ANALISADAS.....	23
4.2.	O SOFTWARE DESENVOLVIDO.....	24
4.3.	RESULTADOS OBTIDOS PARA AS INSTÂNCIAS.....	27
5.	CONCLUSÕES	34
	REFERÊNCIAS	36
	APÊNDICES.....	38

1. INTRODUÇÃO

Uma das mais importantes abordagens para a melhoria da distribuição de produtos em redes logísticas é a do Problema do Roteamento de Veículos Capacitados (PRVC). Desde que foi proposto por DANTZIG e RAMSER, 1959, o PRVC vem sendo estudado e solucionado sob a ótica de inúmeras e diferentes abordagens. Abordagens estas que contemplam desde os métodos exatos de programação linear aos mais recentes empregos de métodos heurísticos.

Com relação aos métodos heurísticos, os principais avanços estão relacionados às heurísticas que aplicam uma escolha racional de algoritmos para se chegar às respostas mais próximas da solução ótima, mas com um tempo de processamento computacional menor. Esse método busca obter soluções factíveis próximas do resultado ótimo, mas com um tempo de cálculo que atenda as necessidades das empresas.

Ao pesquisar esse método heurístico, o presente trabalho apresenta uma alternativa de resolução do PRVC pautada em técnicas da pesquisa operacional atuais, contribuindo com as discussões sobre o tema e desenvolvendo novas ferramentas tecnológicas para a resolução do problema.

1.1. JUSTIFICATIVA

Devido a crescente pressão por redução de custos de transportes no Brasil, as empresas atuantes no setor logístico buscam automatizar, de forma otimizada, ao máximo suas decisões. Nesse escopo, a necessidade de aprimorar as formas como as empresas designam as rotas para os seus veículos se apresenta como um incentivo à redução de custo e ao desenvolvimento de novas tecnologias.

Contudo, várias abordagens matemáticas podem ser utilizadas para reduzir a distância total dos trechos percorridos pelos veículos. Dentre as mais promissoras, destacam-se as relacionadas aos métodos heurísticos de resolução do Problema de Roteamento de Veículos Capacitados (PRVC).

Nesse sentido, o presente trabalho justifica-se por apresentar, testar e avaliar uma nova abordagem heurística para a resolução de um importante problema de distribuição de produtos em redes logísticas.

Esta pesquisa justifica-se ainda pela importância de propor e desenvolver alternativas computacionais atuais que auxiliem as empresas na resolução do PRVC.

1.2. OBJETIVOS

O objetivo deste trabalho é propor uma abordagem heurística diferente para a resolução do Problema de Roteamento de Veículos Capacitados (PRVC). Esta abordagem baseia-se na estratégia “*Cluster First Route Second*”, empregando simulação de Monte Carlo para a determinação das p-mediana, com a aplicação do algoritmo proposto por GILLET e JOHNSON, 1976 para o agrupamento, na sequência a utilização do algoritmo do vizinho mais próximo para a roteirização dos veículos, ao final a resposta será refinada com o algoritmo 2-Opt e juntamente com processamento paralelo para a redução do tempo computacional.

Para tanto, propõem-se os seguintes objetivos específicos:

- a) Caracterizar os principais conceitos relacionados ao PRVC e as técnicas atualmente utilizadas para a resolução deste problema;
- b) Apresentar uma sequência lógica de execução de algoritmos capazes de resolver o problema, para então implementar computacionalmente esta rotina;
- c) Comparar os resultados obtidos com esta proposta heurística com os resultados exatos ótimos presentes na literatura;
- d) Verificar o impacto da abordagem proposta devido aos tempos de processamento.

1.3. METODOLOGIA

Este trabalho se dará inicialmente pela pesquisa baseada em livros, artigos e dissertações relacionados ao Problema de Roteamento de Veículos Capacitados, Processamento Paralelo e as principais metodologias de resolução aplicáveis ao PRVC.

Com base nos algoritmos clássicos pesquisados, será elaborada uma rotina em que estes algoritmos serão aplicados a fim de se obter um procedimento heurístico para a solução do PRVC.

Em seguida, é viabilizada a implementação computacional da heurística proposta, utilizando técnicas de processamento paralelo, e são testadas as instâncias clássicas existentes na literatura.

Por fim, será determinado o nível percentual médio dos desvios das soluções obtidas em relação às soluções ótimas das instâncias, analisando os resultados e verificando a consistência da abordagem heurística proposta para a resolução do PRVC.

2. REFERENCIAL TEÓRICO

A busca pela melhor rota a ser seguida por um veículo é uma etapa fundamental para a gestão logística das empresas, principalmente quando essas empresas não terceirizam suas operações e entregam seus produtos diretamente aos clientes.

Este capítulo, portanto, apresenta os principais aspectos a serem analisados na construção das rotas, enfatizando o Problema de Roteamento de Veículos Capacitados, a estratégia “*Cluster First Route Second*” e os conceitos relacionados ao Processamento Paralelo.

Cabe ressaltar que os aspectos supracitados servirão de referência para fundamentar a estratégia utilizada, neste trabalho, para a construção da rotina heurística.

2.1. O PROBLEMA DE ROTEAMENTO DE VEÍCULOS CAPACITADOS

A primeira notícia que se tem dos problemas de roteirização tratados pela matemática datam de 1800 que foram inicialmente estudados por Sir. William Rowan Hamilton e o matemático Thomas Penyngton Kirkman sendo já chamados de Problema do Caixeiro Viajante (PCV). Este problema também foi denotado como sendo o Problema do Mensageiro, onde cada viajante era responsável por sua própria rota.

O Problema do Roteamento de Veículos Capacitados (PRVC) foi proposto inicialmente por DANTZIG e RAMSER, 1959 como forma de generalização do PCV. Tal estudo foi decorrente da análise de um problema de roteamento para distribuição de gasolina transportada por caminhões capacitados. Este se diferenciava do PCV por tratar da análise de múltiplos veículos, ante a solução do problema do mensageiro que era solucionado individualmente.

O PRVC busca determinar um conjunto de rotas para uma frota homogênea de veículos, a partir de um depósito central com destino a um conjunto de clientes que demandam determinado produto. Cada cliente deve ser atendido por apenas um veículo (embora haja variações no problema original que permite entregas fracionadas) e, além

disso, a demanda transportada não deve exceder sua capacidade de carregamento. O objetivo do PRVC é minimizar a distância total percorrida pela frota de veículos.

2.2. A ESTRATÉGIA “CLUSTER FIRST ROUTE SECOND”

Segundo GUIMARÃES, SCARPIN e STEINER, 2011 a estratégia “*Cluster First Route Second*” para a resolução do PRVC baseia-se na aplicação de duas fases elementares de cálculo para a obtenção da solução final. A primeira consiste em agrupar os clientes conforme a localização e a quantidade de produtos demandada. A segunda, objetiva construir as rotas necessárias para atender a cada grupo de clientes formado, considerando como principal restrição à capacidade máxima de carregamento do veículo.

Garantindo que a demanda dos clientes pertencentes a um grupo não exceda a capacidade do veículo, o PRVC recai na resolução do Problema do Caixeiro Viajante (PCV) ao ensejar um roteiro para cada grupo que passe por todos os pontos e apresente, ao final, a menor distância total possível.

FISHER e JAIKUMAR, 1981 e GILLET e JOHNSON, 1976 são alguns dos autores que utilizam heurísticas do tipo “*Cluster First Route Second*” para a resolução do PRVC em seus trabalhos. Variações do problema clássico tratadas com essa mesma estratégia são encontradas em SARIKLIS e POWELL, 2000 que trabalham para a resolução do OVRP (*Open Vehicle Routing Problem*). Cabe destacar que o OVRP diferencia-se do PRVC por não obrigar o retorno do veículo ao depósito quando do término da rota.

A estratégia “*Cluster First Route Second*” é utilizada ainda por STEINER, ZAMBONI, *et al.*, 2000 para a resolução de problema de roteamento em transporte escolar. Esses autores sugere o emprego da estratégia tanto para o roteamento capacitado quanto para o roteamento não capacitado de veículos.

2.3. PROCESSAMENTO PARALELO

LEIJEN e HALL, 2007 indicam que os computadores de vários núcleos tornaram-se o padrão industrial aplicado atualmente. Infelizmente, segundo os mesmo autores, a maioria dos aplicativos utiliza um só núcleo e não apresenta aumento de velocidade quando

executada em uma máquina com vários núcleos. Dessa forma, para aprimorar o desempenho de um software torna-se necessário executá-lo em vários processadores em paralelo.

Classicamente os desenvolvedores têm manipulado diretamente os *threads* tentando chegar a aplicativos cliente responsivos, algoritmos paralelizados e servidores escalonáveis (TOUB, 2011). No entanto, tais técnicas têm também sido a maneira pela qual os desenvolvedores, historicamente, chegaram a *deadlocks*, *livelocks*, comboios de bloqueio, danças de dois passos, condições de corrida, assinaturas em excesso e uma quantidade de outras dificuldades indesejáveis em seus aplicativos. Em vista desse problema TOUB, 2011 apresenta uma questão da indústria de software, pois há vertentes que não consideram linguagens de alto nível capazes de lidar com isto:

Há quem possa questionar o valor de um subsistema avançado em linguagem gerenciada para escrever código paralelo. Afinal de contas, paralelismo e simultaneidade dizem respeito a desempenho, e desenvolvedores interessados em desempenho procuram por linguagens nativas que ofereçam acesso extremamente rápido ao hardware e controle total sobre cada manipulação de bit, de linha de cache e cada operação sincronizada... certo? Eu temo pela situação de nossa indústria se esse for realmente o caso. (TOUB, 2011)

De acordo com CALLAHAN, 2008 a programação paralela difere da programação simultânea no fato de introduzir oportunidades para a execução simultânea em uma tarefa logicamente única (o que pode ser expresso usando construções sequenciais familiares com suporte em todas as principais linguagens). Entretanto, quando oportunidades simultâneas são introduzidas com sub-tarefas que compartilham objetos de dados, há de se preocupar com bloqueios e disputas. Por isso, a programação paralela apresenta todos os desafios de correção e segurança de programas sequenciais, além de todas as dificuldades de paralelismo e acesso simultâneo a recursos compartilhados. Em decorrência disto cabe ao desenvolvedor o discernimento de em que situações podem ser aplicadas as técnicas de paralelismo.

A programação simultânea é sabidamente difícil como demonstra PATIL e GEORGE, 2008, com isto novas ferramentas de programação foram desenvolvidas para empregar os recursos de paralelismo em computadores multinucleados. Dentre elas, LEIJEN e HALL, 2007 destaca a classe *Parallel* presente no *.NET Framework 4* e exemplificada na Figura 1 abaixo:

```
System.Threading.Tasks.Parallel.For(0, N, i =>
{
    // Processa i aqui
});
```

Figura 1 – Exemplo de codificação paralela
Fonte: (LEIJEN e HALL, 2007)

2.3.1. Manipulação de conjuntos de forma paralela

O presente estudo emprega a linguagem de programação C# 4.0 em conjunto com a extensão de linguagem denominada *Language Integrated Query* (LINQ). O LINQ possibilita uma manipulação de conjuntos de dados. Com esta possibilidade há uma modificação no paradigma de programação exposto por Luca Bolognese em PIALORSI e RUSSO, 2008 mudando o foco da orientação a objeto para a relação entre objetos.

Com este avanço está um grande passo à frente, mas ainda trata os dados de forma serializada, elemento a elemento, TOUB, 2011 indica a existência da classe *Parallel* antes da aplicação do LINQ e esta chega a abordar superficialmente as funcionalidades disponíveis para paralelismo. Conforme o mesmo autor, um dos mais formidáveis avanços em paralelização foi o realizado pelo *.NET Framework 4* através da introdução da extensão *Parallel LINQ* (PLINQ). Essa extensão permite o processamento paralelo dos dados internos de um conjunto, desde que os mesmo possuam alto grau de independência linear.

A Figura 2 abaixo exemplifica a utilização da extensão PLINQ – *AsParallel()* – dentro do software criado neste trabalho:

```
var medianaProxima = (from m in mMedianas.AsParallel()
    orderby m % designando.cliente.Coordenada
    where m.CapacidadeDisponivel >= designando.cliente.Demanda
    select m).First();
```

Figura 2 - Exemplo de uso de PLINQ
Fonte: Elaborado pelo autor

Para reduzir o tempo de processamento e aumentar a capacidade de análise do software, a heurística proposta nesse trabalho bem como os algoritmos utilizados na sua implementação foram programados utilizando os conceitos de processamento paralelo supracitados.

2.4. ESTRATÉGIAS PARA A RESOLUÇÃO DO PRVC

Existem diversas abordagens para a resolução do PRVC. Em geral, essas abordagens podem ser agrupadas em duas classes distintas: métodos exatos e métodos heurísticos.

Referente à primeira classe, alguns dos trabalhos são baseados em algoritmos *branch-and-cut* ou relaxação lagrangeana/geração de colunas. Um dos marcos na abordagem exata para o PRVC foi apresentado em CHRISTOFIDES, MINGOZZI e TOTH, 1981, que trabalharam com limitantes lagrangeanos para a geração de subrotas. O algoritmo *branch-and-bound* foi capaz de resolver instâncias com até 25 clientes. Já, FUKASAWA, LYSGAARD, *et al.*, 2000 desenvolveram um algoritmo *branch-and-cut-and-price* em instâncias com mais de 135 clientes, obtendo para todas elas resultados ótimos.

Pesquisas recentes intensificam o emprego de planos de corte, como se verifica em ACHUTHAN, CACCETTA e HILL, 2003, que propuseram novos planos de corte baseados em um algoritmo *branch-and-cut* e testaram em instâncias que contêm uma faixa de 15 até 100 clientes, além de 24 problemas clássicos da literatura. O trabalho produziu benefícios em termos de esforço computacional.

Já em relação à segunda classe (métodos heurísticos), CAMPOS e MOTA, 2000 apresentaram duas heurísticas: uma baseada em *scratch*, que gera uma solução inicial sem qualquer informação obtida *a priori*, e outra que emprega informações advindas de relaxações lineares fortes a partir do problema original. Os autores utilizaram técnicas de busca tabu para refinar as soluções iniciais. As heurísticas foram testadas em instâncias da literatura com a quantidade de clientes variando de 22 a 135 clientes, obtendo soluções ótimas para a maioria delas.

BERGER e BARKAOUI, 2003 apresentaram um algoritmo genético híbrido, trabalhando com a evolução de duas populações ao mesmo tempo, utilizando operadores que combinam variações de técnicas de roteamento e técnicas de busca. Por sua vez, CHEN,

YANG e WU, 2006 propuseram uma abordagem híbrida baseada em nuvem de partículas e testaram em instâncias clássicas com uma faixa de clientes variando de 33 a 155. Os resultados obtidos se aproximaram dos valores ótimos.

Também empregando técnicas meta-heurísticas, ZHISHUO e YUETING, 2005 desenvolveram um novo método baseado em colônia de formigas. O trabalho apresenta algumas inovações em relação às abordagens tradicionais da meta-heurística para a resolução do PRVC, dentre elas o procedimento para a construção de sub-rotas, que são iniciadas randomicamente a partir dos vértices mais distantes ao invés de iniciarem nos depósitos. Na etapa de refinamento da solução atual, um algoritmo de varredura é aplicado sobre as sub-rotas, trocando pares de vértices entre elas. O trabalho apresentou resultados apenas 0,28% inferiores aos valores ótimos para as instâncias testadas, o que representa um desempenho superior entre outras abordagens baseadas em colônia de formigas para o PRVC.

2.5. SIMULAÇÃO DE MONTE CARLO

Proposta pela primeira vez por VON NEUMANN, ULAM e METROPOLIS, 1928 a simulação de Monte Carlo se dá pela teoria dos jogos, e seu nome tem origem no casino homônimo situado no Principado de Mônaco. Este método foi primeiramente utilizado em cunho científico no projeto Manhattan, no Laboratório Nacional de Los Alamos, para o desenvolvimento de armas nucleares.

O método de forma geral impõe 4 padrões a serem seguidos:

1. Definição do domínio de entradas;
2. Geração de entradas aleatórias a partir de uma distribuição de probabilidades sobre este domínio;
3. Execução de um método determinístico sobre os valores de entrada;
4. Agregação do resultado.

Não há consenso na designação do nome Monte Carlo conforme exposto por RIPLEY, 1987, pois podem ser consideradas como simulações estocásticas. Havendo somente a designação quando da possibilidade de restrição do domínio de entrada, com os demais padrões similares ao modelo estocástico.

3. MATERIAIS E MÉTODOS

Este capítulo busca apresentar formalmente a problemática estudada, os algoritmos clássicos pesquisados e a estratégia de resolução utilizada neste trabalho. Em seu último tópico, será apresentada a heurística desenvolvida para a resolução do PRVC.

3.1. DESCRIÇÃO FORMAL DO PRVC

Segundo DANTZIG e RAMSER, 1959 o PRVC pode ser formalmente definido sob o seguinte aspecto: seja $G(V, E)$ um grafo não direcionado contendo o conjunto de vértices $V = \{1, 2, \dots, n\}$, onde o vértice “1” representa o depósito, enquanto que todos os outros vértices representam os clientes, sendo que cada cliente i possui uma demanda d_i . Cada arco $e \in E$ possui um comprimento não negativo $l(e)$. Dado G e dois números positivos e inteiros (K e C), o PRVC consiste em encontrar um conjunto de rotas para os K veículos, que atenda as restrições:

- Cada rota inicia e termina no depósito;
- Cada cliente é visitado por um único veículo;
- A demanda total de todos os clientes de uma sub-rota não excede a capacidade C do veículo.

Uma instância exemplo é apresentada na Figura 3 a seguir:

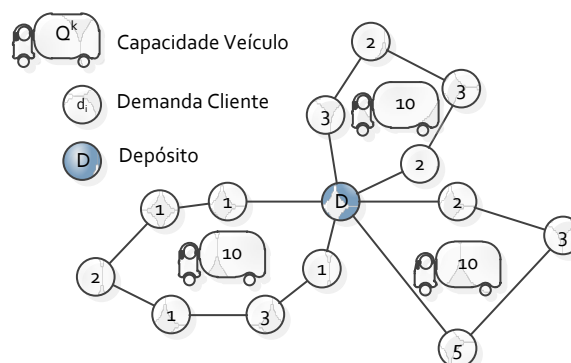


Figura 3 - Instância exemplo do PRVC
Fonte:Elaborado pelo autor

O objetivo do PRVC é minimizar o somatório das distâncias de todas as sub-rotas. O problema é fortemente *NP-hard*, dado que é uma generalização do Problema do Caixeiro

Viajante (PCV) demandando, portanto, elevado esforço computacional para obtenção da solução ótima através de busca exaustiva. Baseado em LIN, LEE, *et al.*, 2009, o PRVC pode ser modelado como um problema de programação inteira mista como segue:

$$\text{Minimizar: } \sum_{i=1}^N \sum_{j=1}^N \sum_{k=2}^K C_{ij} X_{ij}^k \quad (1)$$

$$\text{Sujeito à: } \sum_{i=1}^N \sum_{j=1}^N X_{ij}^k d_i \leq Q^k, \quad 2 \leq k \leq K, \quad (2)$$

$$\sum_{j=2}^N X_{ij}^k = \sum_{j=2}^N X_{ij} \leq 1, \quad \text{para } i = \{1, \dots, N\} \text{ e } k \in \{2, \dots, K\}, \quad (3)$$

$$\sum_{k=2}^K \sum_{j=2}^N X_{ijk} \leq K, \quad \text{para } i = 1, \quad (4)$$

onde C_{ij} é o custo¹ para se ir do cliente i para o cliente j , K é o número de veículos disponível, N é o número de clientes, Q^k é a capacidade de carregamento do veículo k , d_i é a demanda do cliente i e finalmente $X_{ij}^k \in \{0, 1 (i \neq j; i, j \in \{1, 2, \dots, N\})\}$.

A equação (1) é a função objetivo do problema que, conforme já comentado, minimiza o somatório das distâncias de todas as sub-rotas. As restrições (2) referem-se à capacidade de carregamento do veículo e atendimento das demandas de cada cliente, onde $X_{ij}^k = 1$, se o veículo k viaja do cliente i para o cliente j diretamente e $X_{ij}^k = 0$, caso contrário.

As restrições (3) garantem que toda rota tem início e fim no depósito, enquanto que a restrição (4) especifica que há no máximo K rotas partindo do depósito.

3.2. ESTRATÉGIA DE RESOLUÇÃO DO PRVC

A construção da heurística proposta neste artigo requer a resolução dos problemas da p -medianas, da designação ou agrupamento (clusterização), da roteirização de veículos e da melhoria de rotas. Assim, as subseções a seguir discutem esses problemas de forma pormenorizada para que a nova abordagem heurística seja devidamente apresentada ao final.

¹ Sempre que este trabalho trata de custo entre clientes ou custo de rotas está se referindo a UD (unidades de distância).

3.2.1. O Problema das p -medianas

O problema das p -medianas tem por objetivo determinar, em um grafo formado por um conjunto de pontos (clientes), p -pontos principais que representam sementes ou depósitos fictícios. Os demais pontos (clientes) serão designados a cada uma destas medianas de modo a formar p -conjuntos iniciais (p -grupos) de clientes onde cada conjunto contenha a “capacidade limite” igual à capacidade de carregamento dos veículos. Ressalta-se que neste trabalho a capacidade do grupo é a mesma para todos eles (frota homogênea de veículos).

A estratégia de resolução reside na teoria dos grafos que tem por objetivo localizar facilidades ao longo de uma rede viária. Estas facilidades (p -medianas) devem ser escolhidas de forma a minimizar a soma da distância de cada um dos pontos à facilidade mais próxima, ponderada por um fator de demanda.

O problema pode ser resolvido de forma exata, como sugere HAKIMI, 1965, usando enumeração exaustiva ou programação inteira. Porém, segundo JUAN, FAULIN, *et al.*, 2010 recentes avanços no desenvolvimento de geradores números pseudoaleatórios vêm abrindo novas perspectivas para o uso de simulação de Monte Carlo em problemas de otimização.

TAKES, 2010 também opina no mesmo sentido, enfatizando a aplicabilidade do método estocástico de Monte Carlo para a resolução das variantes do *Vehicle Routing Problem* (PRV) em virtude do PRVC ser um problema *np-hard* de otimização combinatória.

Nessa visão, o presente trabalho optou por empregar o método estocástico de Monte Carlo para resolução do problema das p -medianas, no claro intuito de gerar um grupo de pontos a serem empregados como pontos iniciais (depósitos fictícios) para o agrupamento dos clientes.

Para a determinação destes p -pontos iniciais, foi utilizado o gerador de números pseudoaleatórios existente no *.NET Framework 4*, devidamente limitado pelas coordenadas máximas e mínimas nos eixos das ordenadas e abscissas, para determinar esses p -pontos iniciais dentro do grafo inicial de clientes existentes nas instâncias do PRVC. Desta forma garantindo que a escolha dos pontos iniciais será feita de maneira estocástica e baseada em uma série estatística de distribuição normal. Na Figura 4 a seguir pseudocódigo desta implementação:

	Procedimento MonteCarlo (<i>destinos, quantidadeVeiculos, capacidadeVeiculo</i>)
1	Início
2	$xMin \leftarrow \text{Min } destinos.X$
3	$xMax \leftarrow \text{Max } destinos.X$
4	$yMin \leftarrow \text{Min } destinos.Y$
5	$yMax \leftarrow \text{Max } destinos.Y$
6	Para <i>iteração</i> $\leftarrow 1$ até <i>quantidadeVeiculos</i> faça
7	Retorno iterado nova mediana($X \leftarrow \text{aleatório} * (xMax - xMin) + xMin,$
8	$Y \leftarrow \text{aleatório} * (yMax - yMin) + yMin,$
9	<i>capacidade</i> $\leftarrow \text{capacidadeVeiculo}$)
10	FimPara
11	Fim

Figura 4 - Implementação do Monte Carlo

Fonte: Elaborado pelo autor

A orientação citada anteriormente baseia-se na delimitação de uma a área para a atuação da simulação de Monte Carlo. Tal delimitação visa otimizar o processo de geração de pontos pelo método de Monte Carlo, uma vez que elimina a possibilidade de serem escolhidos pontos demasiadamente distantes e externos ao grafo do PRVC.

Cabe ressaltar que essa etapa contribui demasiadamente para o funcionamento e a precisão da heurística proposta neste estudo. Principalmente porque a determinação dos pontos iniciais será pré-requisito para que o algoritmo de GILLET e JOHNSON, 1976 consiga designar os demais pontos e formar os grupos (clusters) de clientes.

3.2.2. O Problema da designação de grupos (clusterização)

Definido o conjunto das medianas é necessário designar os pontos de maneira a se compor o agrupamento para roteirização posterior. Esta designação objetiva minimizar a distância entre os pontos do conjunto e a respectiva mediana, respeitando a capacidade da própria mediana.

Neste trabalho, a designação dos pontos é feita através do algoritmo de GILLET e JOHNSON, 1976. O princípio do algoritmo é construir uma lista ordenada dos pontos a serem designados considerando distâncias entre esses pontos e as duas medianas mais próximas. O algoritmo é descrito pela Figura 5 a seguir:

Seja: t_i^1 e t_i^2 : Primeira e segunda mediana mais próxima ao ponto i ;
 c_i^1 e c_i^2 : Distância à primeira e segunda mediana mais próxima ao ponto i ;
Passo 1. Calcular a distância entre cada nó ainda não designado até cada um dos depósitos (medianas) correspondentes que ainda possuem capacidade;
Passo 2. Para cada ponto i , obter t_i^1 e t_i^2 e as respectivas distâncias c_i^1 e c_i^2 .
Passo 3. Para cada ponto i , calcular a razão $r_i = c_i^1/c_i^2$.
Passo 4. Ordenar os pontos de acordo com r_i em ordem decrescente.
Passo 5. Designar todos os pontos i da lista ordenada para t_i^1 até sua capacidade ser atingida. Se houver algum ponto ainda não designado volte ao passo 1.

Figura 5 - Algoritmo de Gillett e Johnson para agrupamento
 Fonte: (GILLETT e JOHNSON, 1976)

3.2.3. O Problema da Roteirização: Geração de uma Solução Inicial

Inúmeros são os algoritmos existentes para a roteirização de veículos tendo um conjunto de clientes já definido. A resolução do problema do roteamento enseja a obtenção de uma solução inicial para posterior refinamento. Para a abordagem proposta neste trabalho foi utilizado o algoritmo do Vizinho Mais Próximo (VMP) pela sua ampla divulgação na literatura existente e fácil implementação computacional.

O algoritmo de roteirização VMP foi originalmente proposto por COVER e HART, 1967 e consiste composição da rota com base na inserção sequencial de pontos através de um ponto inicial conforme a menor distância entre este e os seus demais pontos (vizinhos). Após determinar todas as distâncias entre o ponto inicial e os demais pontos do cluster, os demais pontos são ordenados de maneira decrescente para possibilitar a escolha e determinação do nó mais próximo que será designado à rota. O algoritmo é exposto na Figura 6 a seguir:

Passo 1. Iniciar a roteirização a partir do depósito e designar este ponto como ponto i visitado ($i \in S$), onde S é o grupo de clientes visitados;
Passo 2. Para cada ponto $j \notin S$, obter d_{ij} sendo este a respectiva distância entre o ponto i e j .
Passo 3. Escolher o menor d_{ij} e marcar o ponto j como visitado;
Passo 4. Faça o ponto j ser o ponto i ;
Passo 5. Se todos os pontos já foram visitados, encerrar o algoritmo;
Passo 6. Caso contrário, retorne ao Passo 2.

Figura 6 - Algoritmo VMP
 Fonte: (COVER e HART, 1967)

3.2.4. O Problema da Roteirização: Melhoria da Solução Inicial

O problema da melhoria de rotas busca aumentar a eficiência do trajeto, ou seja, refinar uma solução inicial obtida para uma rota a ser seguida, *a priori*, por um veículo. Uma das melhores abordagens heurísticas para a resolução desse problema é a proposta por LIN e KERNIGHAN, 1973 denominada troca de arcos k-opt, onde as trocas 2-opt (2 arcos) e 3-opt (3 arcos) são as mais utilizadas.

Os métodos k-opt buscam a melhoria de um trajeto pela substituição de “k” arcos no roteiro estabelecido anteriormente, isto é, “k” arcos são removidos do roteiro e substituídos por outros “k” arcos. Caso alguma melhoria seja detectada, a troca é aceita e o novo arco passa a compor a solução incumbente. Neste presente trabalho esta dinâmica se repete até que nenhuma troca resulte em melhoria.

Conforme apontado por LAPORTE, GENDREAU, *et al.*, 2000, o processo de melhoria k-opt termina em um mínimo local e possui ordem de complexidade $O(nk)$. Quanto maior for o valor de k, melhor será a solução, entretanto o esforço computacional requerido também será maior. Isto leva a um *trade-off* entre qualidade e tempo computacional. Dessa forma, trocas 4-opt e superiores ensejam um custo computacional muitas vezes superior à melhoria da solução obtida.

Neste trabalho optou-se por utilizar a troca 2-opt pela sua simplicidade de implementação, uma vez que esta estratégia de melhoria realiza a comparação somente entre dois trechos para então determinar se a inversão de sentido entre eles é mais econômica em relação à rota inicial. Caso seja, a nova rota passa a ser a rota preferencial.

Na Figura 7 abaixo, é ilustrada a sistemática de troca de arcos para a abordagem 2-opt que será utilizada na heurística proposta no trabalho:

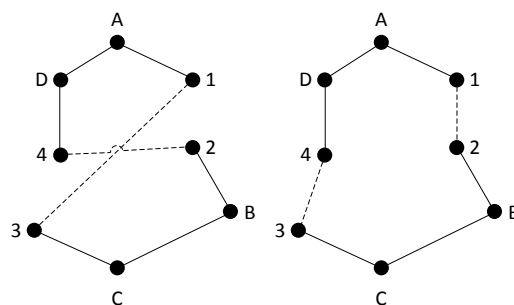


Figura 7 - Troca 2-opt entre os trechos 1-2 e 3-4
Fonte: Adptado de (COSTA, 1997)

Cabe destacar que a geração de boas soluções iniciais é fundamental para que se obtenham boas soluções finais com os métodos k-opt para a melhoria de rotas (COSTA, 1997) (CAMPOS, 2008).

3.3. HEURÍSTICA DESENVOLVIDA

A heurísticas proposta neste trabalho foi desenvolvida e implementada computacionalmente seguindo a concatenação de algoritmos representada através da Figura 8 a seguir:

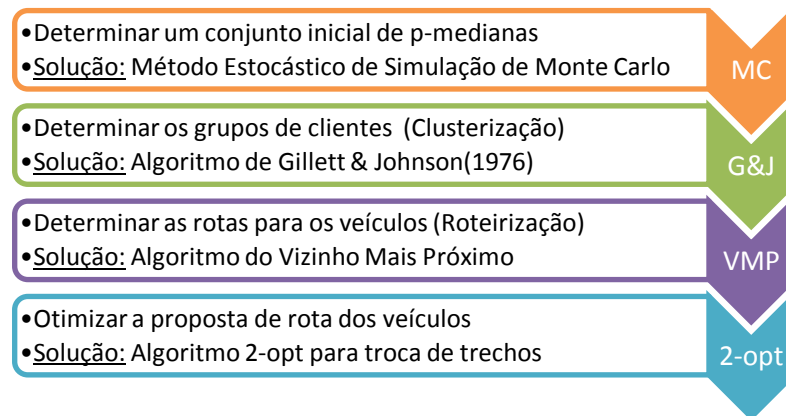


Figura 8 - Heurística proposta para a resolução do PRVC
Fonte: Elaborado pelo autor

A partir desta heurística proposta acima foi criado o seguinte trecho de pseudocódigo que representa o encadeamento procedural. Como pode ser observado na Figura 9 abaixo:

Procedimento Rotina (<i>instância</i>)	
1	Início
2	<i>iteração</i> ← 0
3	Enquanto <i>iteração</i> < 1000 faz
4	<i>SoluçãoInicial</i> ← ProcessarMonteCarloComGillettJohnson(<i>instância</i>)
5	<i>Roteirização</i> ← ProcessarVizinhoMaisProximo(<i>SoluçãoInicial</i> , <i>instância</i>)
6	<i>Resposta</i> ← 2-Opt(<i>Roteirização</i> , <i>instância</i>)
7	<i>iteração</i> ← <i>iteração</i> + 1
8	Se <i>Resposta.CustoTotal</i> < <i>Solução.CustoTotal</i> então
9	<i>Solução</i> ← <i>Resposta</i>
10	FimSe
11	FimEnquanto
12	Retorna <i>Solução</i>
13	Fim

Figura 9 - Pseudocódigo representando encadeamento de algoritmos
Fonte: Elaborado pelo autor

4. RESULTADOS E DISCUSSÕES

4.1. INSTÂNCIAS ANALISADAS

Foram analisados os resultados obtidos para a resolução das instâncias clássicas da literatura disponíveis no *website* <http://www.branchandcut.org>. Dentre as instâncias disponibilizadas, foram avaliadas apenas aquelas cujos dados principais estavam compreendidos no espaço vetorial de domínio Euclidiano (2D).

Dessa forma, foram avaliadas as 90 instâncias euclidianas pertencentes às classes A, B, E, G, M, P, V e evidenciadas na Tabela 1 abaixo:

Tabela 1 - Instâncias utilizadas para avaliação da heurística proposta

Instância	Instância	Instância	Instância	Instância	Instância
A-n32-k5	A-n54-k7	B-n41-k6	B-n68-k9	M-n151-k12	P-n55-k8
A-n33-k5	A-n55-k9	B-n43-k6	B-n78-k10	M-n200-k17	P-n55-k10
A-n33-k6	A-n60-k9	B-n44-k7	E-n101-k8	P-n16-k8	P-n60-k10
A-n34-k5	A-n61-k9	B-n45-k5	E-n101-k14	P-n19-k2	P-n60-k15
A-n36-k5	A-n62-k8	B-n45-k6	E-n22-k4	P-n20-k2	P-n65-k10
A-n37-k5	A-n63-k9	B-n50-k7	E-n23-k3	P-n21-k2	P-n70-k10
A-n37-k6	A-n63-k10	B-n50-k8	E-n30-k3	P-n22-k2	P-n76-k4
A-n38-k5	A-n64-k9	B-n51-k7	E-n33-k4	P-n22-k8	P-n76-k5
A-n39-k5	A-n65-k9	B-n52-k7	E-n51-k5	P-n23-k8	P-n101-k4
A-n39-k6	A-n69-k9	B-n56-k7	E-n76-k10	P-n40-k5	att-n48-k4
A-n44-k6	A-n80-k10	B-n57-k7	E-n76-k14	P-n45-k5	
A-n45-k6	B-n31-k5	B-n57-k9	E-n76-k7	P-n50-k7	
A-n45-k7	B-n34-k5	B-n63-k10	E-n76-k8	P-n50-k8	
A-n46-k7	B-n35-k5	B-n64-k9	G-n262-k25	P-n50-k10	
A-n48-k7	B-n38-k6	B-n66-k9	M-n101-k10	P-n51-k10	
A-n53-k7	B-n39-k5	B-n67-k10	M-n121-k7	P-n55-k7	

Fonte: Elaborado pelo autor

Cabe ressaltar que a diferença existente entre as classes de instâncias pode variar desde a região onde os clientes estão distribuídos até a densidade dos mesmos em relação à área do plano 2D estudado.

4.2. O SOFTWARE DESENVOLVIDO

A elaboração do código fonte do *software* foi realizada no ambiente colaborativo disponibilizado pelo *website* <http://www.codeplex.com>. Nesse sentido, todo o código fonte encontra-se disponível em <http://prvc.codeplex.com>, enfatizando a natureza do *software* (*open source*) e possibilitando a proposição de melhorias por pessoas interessadas no tema.

Quanto ao *hardware* utilizado, o *software* foi executado em um processador Intel® Core™2 Quad Q6600, 2.4 GHz com 4 GB de RAM e sistema operacional Windows 7™, 64-Bits.

Na sequência, as principais características do *software* desenvolvido serão apresentadas, iniciando-se pela interface do usuário (UI) indicada na Figura 10 abaixo:

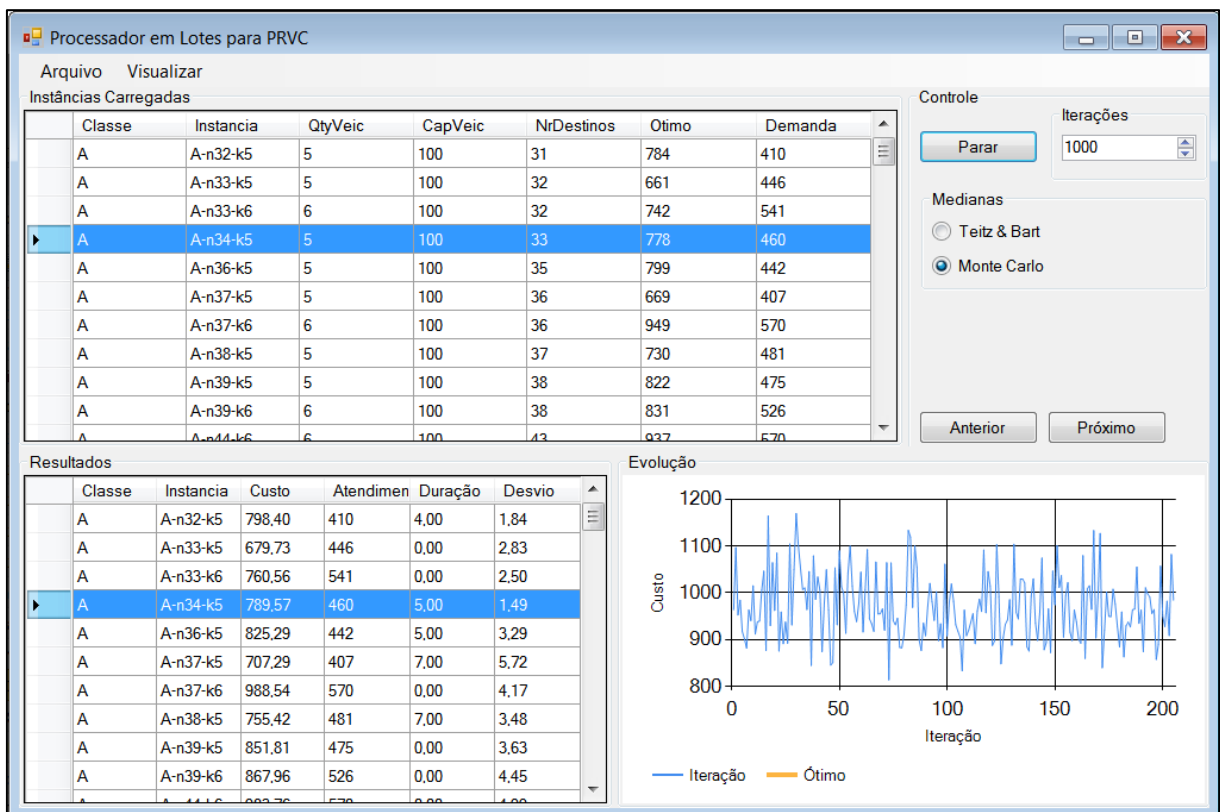


Figura 10 - Interface do usuário para o *software*
Fonte: Elaborado pelo autor

Nessa interface são apresentados os seguintes núcleos essenciais de informações: Instâncias Carregadas, Controle, Resultados e Evolução.

Na parte superior esquerda da UI (Instâncias Carregadas) localizam-se as instâncias a serem testadas e as suas principais informações. Observa-se, da esquerda para a direita, os

seguintes dados: a classe e o nome da instância, a quantidade de veículos, o número de destinos atendidos, a solução ótima e a sua demanda total.

No canto inferior esquerdo da UI (Resultados), apresentam-se as seguintes informações: a classe e o nome da instância, o caminho total da solução ótima encontrada (Custo), a demanda atendida (Atendimento), o tempo de processamento (Duração) e o desvio da solução obtida em relação à solução ótima da instância. Cabe destacar que o tempo de processamento está exposto em milissegundos (ms) e o desvio em porcentagem (%).

Na parte superior direita da UI (Controle) é possível especificar a quantidade de iterações a serem executadas e o método de resolução do problema das p -medianas. Após escolher na UI (Instâncias Carregadas) a instância que será resolvida, basta clicar no botão “iniciar” e o *software* calcula o resultado.

Para acompanhar o andamento da resolução da instância, foi colocado um gráfico Custo x Iterações (Evolução) no canto inferior direito da UI. Nesse gráfico é possível identificar a melhoria dos resultados obtidos à medida que as iterações são executadas.

A evolução dos resultados das iterações é indicada por uma linha na cor azul e a evolução das melhores soluções (possível ótimo) obtidas é indicada por uma linha na cor laranja. Essa diagramação é exemplificada pela Figura 11 a seguir:

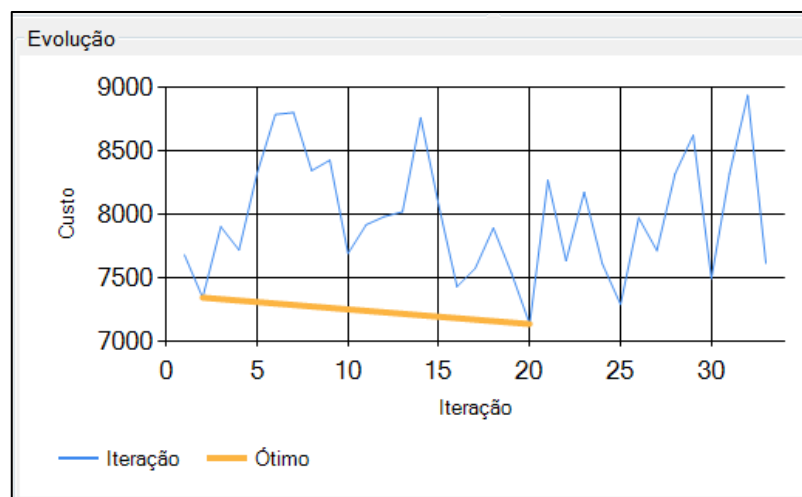


Figura 11 - Gráfico de monitoramento Custo x Iteração

Fonte: Elaborado pelo autor

Como mencionado no Capítulo 2, técnicas de processamento paralelo foram inseridas na arquitetura do software para otimizar os cálculos e possibilitar a utilização de toda a infraestrutura computacional existente. Uma evidência do uso dessas técnicas de

programação, além dos baixos tempos de processamento, pode ser identificada quando se analisa a taxa de ocupação dos núcleos de processadores existentes.

Durante a execução do software, foram obtidos taxas de utilização da capacidade de processamento dos núcleos computacionais de 90%, conforme pode ser verificado pelo “Histórico do uso de CPU” apontado na Figura 12 a seguir:

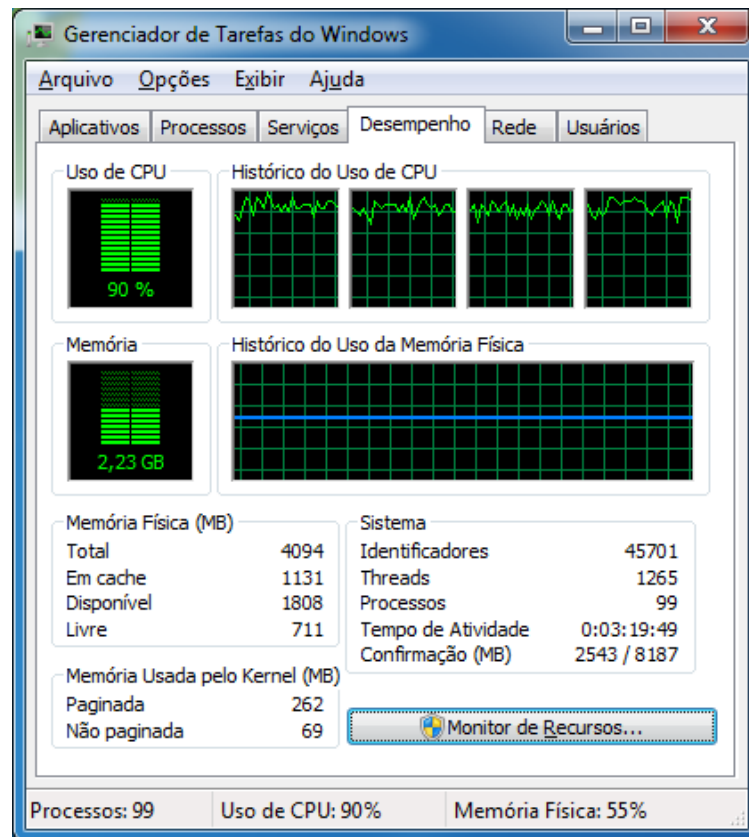


Figura 12 - Uso dos núcleos computacionais

Fonte: Elaborado pelo autor

O software desenvolvido apresenta duas possibilidades de visualização para as rotas obtidas. Na primeira, as rotas dos veículos são apresentadas em formato de tabela através da opção “Visualizar” (canto superior direito) e depois escolhendo a opção “Rotas em forma de tabela”.

Um exemplo da tabela gerada através dessa opção é exposto pela Figura 13 a seguir, onde o ponto 1 representa o depósito central da instância:

Classe	Instancia	Caminhao	Rota
B	B-n50-k8	1	1, 35, 49, 41, 8, 34, 13, 28, 38, 7, 1
B	B-n50-k8	2	1, 3, 10, 26, 50, 44, 12, 37, 21, 31, 1
B	B-n50-k8	3	1, 39, 42, 46, 40, 16, 1
B	B-n50-k8	4	1, 2, 22, 43, 18, 9, 1
B	B-n50-k8	5	1, 33, 29, 30, 20, 1
B	B-n50-k8	6	1, 15, 6, 47, 1
B	B-n50-k8	7	1, 45, 32, 23, 24, 36, 11, 14, 4, 1
B	B-n50-k8	8	1, 17, 25, 48, 27, 19, 5, 1

Figura 13 - Tabela com as rotas encontradas para os veículos

Fonte: Elaborado pelo autor

A segunda opção para a visualização das rotas é através de um gráfico 2D gerado pelo *software*. Para tanto, basta clicar duas vezes sobre o nome da instância resolvida no canto inferior esquerdo da UI(Resultados) para que o gráfico seja apresentado ao usuário.

Um exemplo desse gráfico é exemplificado na Figura 14 a seguir:

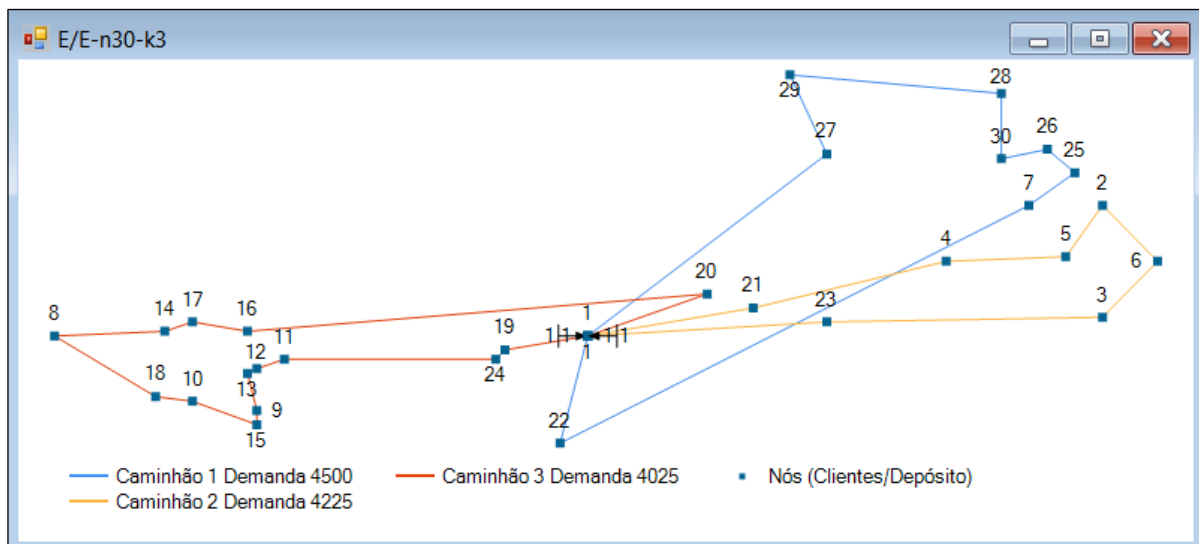


Figura 14 - Solução gráfica 2D para as rotas dos veículos

Fonte: Elaborado pelo autor

4.3. RESULTADOS OBTIDOS PARA AS INSTÂNCIAS

Todas as 90 instâncias contidas na Tabela 1 do item 4.1 foram devidamente resolvidas pelo *software*. Devido a grande quantidade de dados, as soluções encontradas para as rotas a serem seguidas pelos veículos estão identificadas e representadas no Apêndice A.

Nesta parte do estudo pretende-se apenas apontar os resultados globais obtidos para as instâncias, os seus desvios percentuais em relação à solução ótima e os respectivos tempos de processamento obtidos com o recurso do paralelismo computacional (*Parallel*). Para tanto, os resultados foram divididos em tabelas e apresentados na ordem alfabética das instâncias (A, B, E, G, M, P e V).

Ressalta-se que o número de pontos (clientes e depósito) está indicado no nome da própria instância após a letra “n” e o número de veículos utilizados está indicado após a letra “k”. Cabe ainda destacar que cada instância evidenciada na Tabela 1 foi resolvida pelo *software* estipulando-se um limite de 1000 iterações.

Para as instâncias da classe A, foram obtidos os resultados constantes da Tabela 2 abaixo:

Tabela 2 - Resultados para as instâncias da classe A

Instância	Número de Pontos	Solução Ótima	Melhor Solução Obtida	Desvio da Solução (%)	Tempo (ms)
A-n32-k5	32	784	798,4	1,84	4
A-n33-k5	33	661	679,7	2,83	0
A-n33-k6	33	742	760,6	2,50	0
A-n34-k5	34	778	789,6	1,49	5
A-n36-k5	36	799	825,3	3,29	5
A-n37-k5	37	669	707,3	5,72	7
A-n37-k6	37	949	988,5	4,17	0
A-n38-k5	38	730	755,4	3,48	7
A-n39-k5	39	822	851,8	3,63	0
A-n39-k6	39	831	868,0	4,45	0
A-n44-k6	44	937	983,8	4,99	0
A-n45-k6	45	944	1088,1	15,27	0
A-n45-k7	45	1146	1212,5	5,81	0
A-n46-k7	46	914	1001,7	9,60	10
A-n48-k7	48	1073	1133,2	5,61	0
A-n53-k7	53	1010	1088,3	7,75	0
A-n54-k7	54	1167	1260,8	8,04	0
A-n55-k9	55	1073	1132,6	5,55	0
A-n60-k9	60	1354	1438,6	6,25	0
A-n61-k9	61	1034	1214,7	17,48	0
A-n62-k8	62	1288	1402,9	8,92	0
A-n63-k9	63	1616	1752,8	8,46	0
A-n63-k10	63	1314	1434,4	9,17	0
A-n64-k9	64	1401	1480,7	5,69	0
A-n65-k9	65	1174	1253,4	6,76	10
A-n69-k9	69	1159	1278,0	10,27	0
A-n80-k10	80	1763	1919,9	8,90	15,6
Média				6,59	2,35

Para as instâncias da classe B, os resultados obtidos constam da Tabela 3 a seguir:

Tabela 3 - Resultados para as instâncias da classe B

Instância	Número de Pontos	Solução Ótima	Melhor Solução Obtida	Desvio da Solução (%)	Tempo (ms)
B-n31-k5	31	672	683,2	1,66	0
B-n34-k5	34	788	800,6	1,60	0
B-n35-k5	35	955	961,8	0,71	0
B-n38-k6	38	805	827,6	2,81	0
B-n39-k5	39	549	563,3	2,60	0
B-n41-k6	41	829	849,6	2,49	0
B-n43-k6	43	742	765,4	3,16	0
B-n44-k7	44	909	941,8	3,61	0
B-n45-k5	45	751	778,9	3,71	0
B-n45-k6	45	678	695,1	2,52	0
B-n50-k7	50	741	753,7	1,71	0
B-n50-k8	50	1312	1361,1	3,74	0
B-n51-k7	51	1032	1042,2	0,98	0
B-n52-k7	52	747	780,2	4,45	0
B-n56-k7	56	707	754,9	6,77	15,6
B-n57-k7	57	1153	1301,5	12,88	0
B-n57-k9	57	1598	1636,1	2,38	0
B-n63-k10	63	1496	1609,1	7,56	15,6
B-n64-k9	64	861	952,3	10,60	0
B-n66-k9	66	1316	1396,6	6,13	0
B-n67-k10	67	1032	1205,5	16,81	0
B-n68-k9	68	1272	1319,4	3,73	0
B-n78-k10	78	1221	1350,5	10,60	0
Média				4,92	1,35

Fonte: Elaborado pelo autor

Para as instâncias da classe E, a Tabela 4 abaixo traz os resultados:

Tabela 4 - Resultados para as instâncias da classe E

Instância	Número de Pontos	Solução Ótima	Melhor Solução Obtida	Desvio da Solução (%)	Tempo (ms)
E-n22-k4	22	375	375,3	0,07	0
E-n23-k3	23	569	568,6	-0,08 ²	0
E-n30-k3	30	534	535,8	0,34	0
E-n33-k4	33	835	852,1	2,04	0
E-n51-k5	51	521	548,5	5,27	0
E-n76-k10	76	830	948,2	14,24	15,6
E-n76-k14	76	1021	1216,5	19,15	0
E-n76-k7	76	682	756,5	10,92	15,6
E-n76-k8	76	735	814,4	10,81	15,6
E-n101-k8	101	817	906,2	10,92	31,2
E-n101-k14	101	1071	1251,7	16,87	15,6
Média				8,23	8,51

Fonte: Elaborado pelo autor

² Esta divergência se dá além do ótimo devido a arredondamento, pois todos os valores são apresentados inteiros na fonte.

Para a única instância da classe G, o resultado obtido consta na Tabela 5 a seguir:

Tabela 5 - Resultado para a instância da classe G

Instância	Número de Pontos	Solução Ótima	Melhor Solução Obtida	Desvio da Solução (%)	Tempo (ms)
G-n262-k25	262	6119	6676,3	9,11	62,4
Média				9,11	62,4

Fonte: Elaborado pelo autor

Para as instâncias da classe M, os resultados obtidos constam na Tabela 6 abaixo:

Tabela 6 - Resultados para as instâncias da classe M

Instância	Número de Pontos	Solução Ótima	Melhor Solução Obtida	Desvio da Solução (%)	Tempo com <i>Parallel</i> (ms)
M-n101-k10	101	820	909,4	10,91	15,6
M-n121-k7	121	1034	1089,2	5,34	109,2
M-n151-k12	151	1053	1206,7	14,59	46,8
M-n200-k17	200	1373	1555,0	13,26	62,4
Média				11,02	58,50

Fonte: Elaborado pelo autor

Para as instâncias da classe P, os resultados obtidos constam na Tabela 7 a seguir:

Tabela 7 - Resultados para as instâncias da classe P

Instância	Número de Pontos	Solução Ótima	Melhor Solução Obtida	Desvio da Solução (%)	Tempo (ms)
P-n16-k8	16	450	451,3	0,30	0
P-n19-k2	19	212	212,7	0,31	0
P-n20-k2	20	216	219,0	1,37	0
P-n21-k2	21	211	214,7	1,77	0
P-n22-k2	22	216	220,2	1,93	0
P-n22-k8	22	603	620,7	2,93	0
P-n23-k8	23	529	571,6	8,05	0
P-n40-k5	40	458	472,5	3,17	0
P-n45-k5	45	510	534,6	4,82	0
P-n50-k7	50	554	594,2	7,25	0
P-n50-k8	50	631	825,2	30,77	0
P-n50-k10	50	696	748,5	7,54	0
P-n51-k10	51	741	810,2	9,34	0
P-n55-k7	55	568	613,7	8,05	15,6
P-n55-k8	55	588	636,5	8,25	0
P-n55-k10	55	694	730,2	5,22	0
P-n60-k10	60	744	812,8	9,25	0
P-n60-k15	60	968	1096,8	13,31	0
P-n65-k10	65	792	874,6	10,43	0
P-n70-k10	70	827	946,0	14,39	10
P-n76-k4	76	593	626,9	5,72	62,4
P-n76-k5	76	627	672,4	7,25	31,2
P-n101-k4	101	681	737,4	8,27	171,6
Média				7,38	12,64

Fonte: Elaborado pelo autor

Para a instância euclidiana da classe V, foi obtido o resultado constante da Tabela 8 a seguir:

Tabela 8 - Resultado para a instância da classe V

Instância	Número de Pontos	Solução Ótima	Solução Obtida	Desvio da Solução (%)	Tempo (ms)
att-n48-k4	48	40002	41359,0	3,39	15,6
Média				3,39	15,6

Fonte: Elaborado pelo autor

A Tabela 9 apresenta a síntese dos desvios globais obtidos em relação às soluções ótimas da instâncias, determinando um desvio médio geral de 6,25% para o presente trabalho.

Tabela 9 - Desvios médios obtidos na resolução do PRVC

Classe da Instância	Quantidade de Instâncias Analisadas	Desvio Médio em Relação à Solução Ótima (%)	Tempo médio de Processamento (ms)
A	27	6,59	2,35
B	23	4,92	1,35
E	11	8,23	8,51
G	1	9,11	62,4
M	5	11,02	58,50
P	23	7,38	12,64
V	1	3,39	15,6
Média Global das 90 Instâncias:		6,25 %	8,79 (ms)

Fonte: Elaborado pelo autor

Todos os tempos obtidos foram inferiores a 01 segundo, corroborando a decisão tomada de empregar técnicas de processamento paralelo na implementação computacional da heurística utilizada.

Quanto ao desvio médio geral obtido, o valor de 6,25% representa um nível de precisão significativo para a heurística desenvolvida, principalmente em função da grande quantidade e variedade de instâncias testadas no estudo. Várias instâncias foram resolvidas na sua otimalidade como apontam, por exemplo, os resultados obtidos para as instâncias E-n22-k4(0,07%), E-n23-k3 (-0,08%³) e P-n16-k8 (0,30%).

Todas as tabelas aqui expostas tiveram as linhas com desvio inferior a 2% destacadas.

³ Esta divergência se dá além do ótimo devido a arredondamento, pois todos os valores são apresentados inteiros na fonte.

Para a instância P-n16-k8, o software obteve a solução gráfica apresentada na figura 17 a seguir:

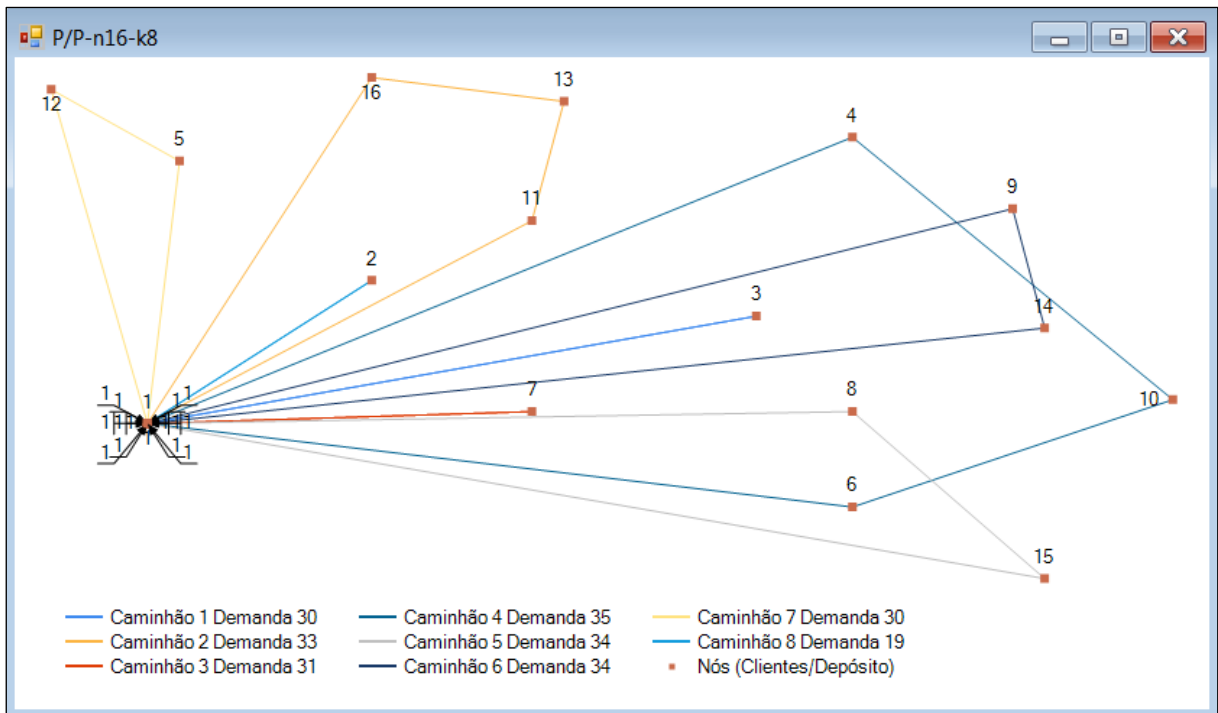


Figura 17 - Solução gráfica para a instância P-n16-k8

Fonte: Elaborado pelo autor

Cabe ressaltar que as soluções gráficas aqui expostas se aproximam do ótimo em menos de 0,5% de desvio da solução. Podendo representar apenas a manipulação de números inteiros para ponto flutuante, com isto, pode-se considerar como solução ótima.

Pelos resultados obtidos, pode-se verificar plenamente a qualidade da estratégia empregada tanto para o desenvolvimento do *software* quanto para a construção da heurística proposta neste trabalho. Cabe destacar, por fim, que o desvio médio obtido (6,25%) pode garantir uma boa precisão para a aplicabilidade real desta ferramenta na área de roteamento de veículos capacitados.

5. CONCLUSÕES

O PRVC tem se tornado um dos mais estudados problemas de otimização em redes, principalmente pela sua grande aplicabilidade na área de transporte. Contudo, mesmo possuindo uma formulação exata relativamente simples, a sua resolução analítica é de grande complexidade.

Por esse motivo, torna-se inviável o emprego da sua resolução analítica em problemas com um grande número de clientes a serem atendidos e veículos a serem roteirizados. Nesse contexto, a utilização de heurísticas como instrumentos de resolução do PRVC torna-se propícia, conforme demonstrado pelos resultados demonstrados neste trabalho.

O presente estudo apresentou uma heurística baseada na estratégia “*Cluster First Route Second*” para resolver o PRVC. A estratégia utilizada diferencia-se por duas características: a utilização de simulações de Monte Carlo para determinação de grupos de clientes a serem atendidos (cluster) e o uso do algoritmo VMP para a roteirização do veículo dentro de cada grupo.

Outro diferencial deste trabalho foi o uso do processamento paralelo (*Parallel Computing*) na implantação computacional da heurística. Essa implantação obteve um tempo médio de processamento inferior a 1 segundo, fortalecendo a vantagem de sua utilização em problemas do tipo *Np-Hard*.

Para consolidar a precisão da heurística, foi determinado o desvio médio geral de 6,25% em relação aos resultados ótimos das instâncias testadas. Esse índice de precisão possibilita uma excelente aplicabilidade da heurística em problemas práticos das empresas, sem requerer das mesmas um tempo de espera elevado para a tomada de decisão.

Ressalta-se ainda que 77,5% das respostas obtidas estão com desvio inferior a 10% da solução ótima. Como pode se observar no histograma a seguir:

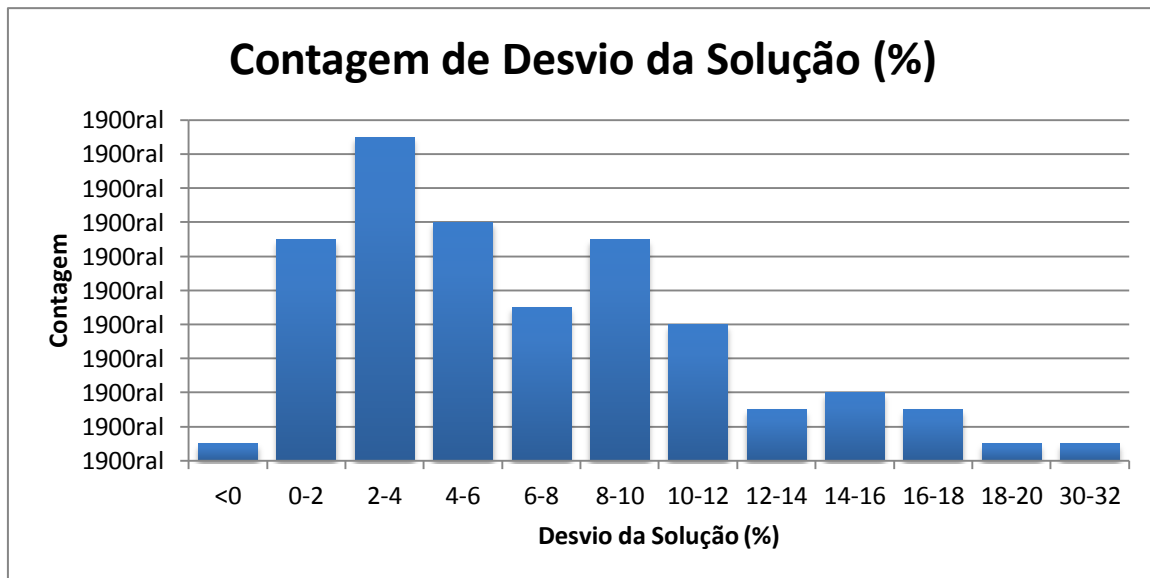


Figura 18 - Histograma da distribuição de contagem de soluções e seus respectivos desvios do ótimo
Fonte: Elaborado pelo autor

Por fim, para os trabalhos futuros, sugere-se o desenvolvimento de estratégias de aperfeiçoamento da designação dos grupos de clientes, ou seja, melhores alternativas ao algoritmo de Gillet & Johnson para a resolução do problema das p -medianas. É possível que ocorram oportunidades de melhoria nessa área ao se sofisticar a designação dos clientes a seus respectivos *clusters*. Outras possibilidades é o refino no critério de parada. Além da possibilidade de estudo em formas de otimização da solução fazendo trocas de clientes entre *clusteres*, analogamente a troca de arestas realizadas no 2-Opt.

REFERÊNCIAS

ACHUTHAN, N. R.; CACCETTA, L.; HILL, S. P. **An Improved Branch-and-Cut Algorithm for the Capacitated Vehicle Routing Problem**. *Transportation Science*, v. 37, n. 2, p. 153-169, Maio 2003.

BERGER, Jean; BARKAOUI, Mohamed. **A hybrid genetic algorithm for the capacitated vehicle routing problem**. *Proceedings of the 2003 international conference on Genetic and evolutionary computation: Part I, Chicago*, p. 646-656, 2003. ISSN 3-540-40602-6.

CALLAHAN, David. **Considerações sobre design de programação paralela**. *MSDN Magazine*, p. 86-88, Outubro 2008. Disponível em: <<http://msdn.microsoft.com/pt-br/magazine/cc872852.aspx>>. Acesso em: 29 ago. 2011.

CAMPOS, D. S. **Integração de problemas de carregamento e roteamento de veículos, com janela de tempo e frota heterogênea**. Universidade de São Paulo. São Paulo. 2008.

CAMPOS, V.; MOTA, E. **Heuristic Procedures for the Capacitated Vehicle Routing Problem**. *Computational Optimization and Applications*, v. 16, n. 3, p. 265-277, 2000.

CHEN, Ai-ling; YANG, Gen-ke; WU, Zhi-ming. **Hybrid discrete particle swarm optimization algorithm for capacitated vehicle routing problem**. *Journal of Zhejiang University - Science A*, v. 7, n. 4, p. 607-614, 2006.

CHRISTOFIDES, N.; MINGOZZI, A.; TOTH, P. **Exact algorithms for the vehicle routing problem, based on spanning tree and shortest path relaxations**. *Mathematical Programming*, v. 20, n. 1, p. 255-282, 1981. ISSN 00255610.

COSTA, Deise Maria B. **Aplicação de algumas técnicas da pesquisa operacional na otimização de serviços postais**. Universidade Federal do Paraná. Curitiba. 1997.

COVER, T.; HART, P. **Nearest neighbor pattern classification**. *IEEE Transactions on Information Theory*, v. 13, n. 1, p. 21-27, Janeiro 1967. ISSN 0018-9448.

DANTZIG, G.; RAMSER, R. **The truck dispatching problem**. *Management Science*, v. 6, p. 80-91, 1959.

FISHER, M.; JAIKUMAR, R. **A generalized assignment heuristic for vehicle routing**. *Networks*, v. 11, p. 109-124, 1981.

FUKASAWA, Ricardo et al. **Robust Branch-and-Cut-and-Price for the Robust Branch-and-Cut-and-Price for the**. *Mathematical Programming*, v. 106, n. 3, p. 491-511, 2000.

GILLETT, Billy E.; JOHNSON, Jerry G. **Multi-terminal vehicle-dispatch algorithm**. *Omega*, v. 4, p. 711-717, 1976.

GUIMARÃES, Thiago A.; SCARPIN, Cassius T.; STEINER, Maria Teresinha A. **Uma nova abordagem heurística para a resolução do problema do roteamento de veículos capacitados**. *Revista Gestão Industrial*, v. 7, n. 3, p. 103-119, 2011. ISSN 1808-0448.

HAKIMI, S. L. **Optimum distribution of switching centers in a communication network and some related graph theoretic problems.** Operations Research, v. 13, n. 3, p. 462-475, Junho 1965.

JUAN, Angel A. et al. **The SR-GCWS hybrid algorithm for solving the capacitated vehicle routing problem.** Applied Soft Computing, v. 10, n. 1, p. 215-224, Janeiro 2010.

LAPORTE, G. et al. **Classical and modern heuristics for the vehicle routing problem.** International Transactions in Operational Research, v. 7, n. 4-5, p. 285-300, 2000.

LEIJEN, Daan; HALL, Judd. **Otimize o código gerenciado para máquinas de vários núcleos.** MSDN Magazine, Outubro 2007.

LIN, S.; KERNIGHAN, B. W. **An Effective Heuristic Algorithm for the Traveling-Salesman Problem.** Operations Research, v. 21, n. 2, p. 498-516, Abril 1973.

LIN, Shih-Wei et al. **Applying hybrid meta-heuristics for capacitated vehicle routing problem.** Expert Systems with Applications, v. 36, n. 2, p. 1505-1512, Março 2009.

PATIL, Rahul V.; GEORGE, Bobby. **Ferramentas e técnicas para identificar problemas de simultaneidade.** MSDN Magazine, Junho 2008. Disponível em: <<http://msdn.microsoft.com/pt-br/magazine/cc546569.aspx>>. Acesso em: 06 Novembro 2011.

PIALORSI, Paolo; RUSSO, Marco. **Programming Microsoft LIQN.** Redmond: Microsoft Press, 2008.

RIPLEY, Brian D. **Frontmatter, in Stochastic Simulation.** Hoboken: John Wiley & Sons, Inc., 1987.

SARIKLIS, D.; POWELL, S. **A heuristic method for the open vehicle routing problem.** Journal of the Operational, v. 51, p. 564-573, 2000.

STEINER, Maria Teresinha A. et al. **O problema de roteamento no transporte escolar.** Pesquisa Operacional, Rio de Janeiro, v. 20, n. 1, p. 83-99, Junho 2000. ISSN 0101-7438.

TAKES, Frank. **Applying Monte Carlo Techniques to the Capacitated Vehicle Routing Problem,** 2010. Disponível em: <<http://www.liacs.nl/assets/Masterscripties/2010-01FrankTakes.pdf>>. Acesso em: 5 Novembro 2011.

TOUB, Stephen. **O passado, o presente e o futuro da paralelização de aplicativos.NET.** MSDN Magazine, Agosto 2011.

VON NEUMANN, John; ULAM, Stanislaw; METROPOLIS, Nicholas. **Zur theorie der gesellschaftsspiele.** Mathematische Annalen, v. 100, n. 6, p. 295-320, 1928.

ZHISHUO, Liu; YUETING, Cai. **Sweep based multiple ant colonies algorithm for capacitated vehicle routing problem.** IEEE International Conference on e-Business Engineering, 2005. ICEBE 2005, Beijing, p. 387-394, Outubro 2005. ISSN 0-7695-2430-3.

APÊNDICES

APÊNDICE A – RESOLUÇÃO DAS INSTÂNCIAS COM AS ROTAS DOS VEÍCULOS

Obs.: A numeração de identificação dos pontos (clientes) permanece como indicado no corpo das instâncias, onde o depósito é representado sempre como o ponto 1.

Rotas Obtidas para os Veículos das 90 Instâncias Clássica Resolvidas			
Classe	Instância	Caminhão	Rota
A	A-n32-k5	1	1, 15, 25, 28, 1
A	A-n32-k5	2	1, 27, 8, 14, 18, 20, 32, 22, 1
A	A-n32-k5	3	1, 12, 5, 29, 24, 3, 4, 7, 1
A	A-n32-k5	4	1, 19, 9, 10, 23, 16, 11, 26, 30, 6, 21, 1
A	A-n32-k5	5	1, 31, 17, 2, 13, 1
A	A-n33-k5	1	1, 12, 20, 15, 22, 2, 32, 19, 29, 1
A	A-n33-k5	2	1, 16, 17, 18, 10, 4, 30, 1
A	A-n33-k5	3	1, 3, 25, 7, 24, 1
A	A-n33-k5	4	1, 13, 6, 28, 26, 31, 11, 1
A	A-n33-k5	5	1, 21, 33, 14, 9, 8, 27, 5, 23, 1
A	A-n33-k6	1	1, 15, 18, 30, 12, 11, 33, 1
A	A-n33-k6	2	1, 21, 10, 16, 3, 4, 9, 5, 1
A	A-n33-k6	3	1, 2, 8, 20, 7, 19, 1
A	A-n33-k6	4	1, 13, 1
A	A-n33-k6	5	1, 29, 28, 31, 17, 26, 22, 1
A	A-n33-k6	6	1, 32, 24, 25, 27, 23, 6, 14, 1
A	A-n34-k5	1	1, 11, 14, 26, 32, 29, 33, 22, 1
A	A-n34-k5	2	1, 15, 30, 9, 16, 7, 8, 21, 1
A	A-n34-k5	3	1, 28, 2, 24, 12, 20, 18, 1
A	A-n34-k5	4	1, 5, 27, 6, 31, 25, 1
A	A-n34-k5	5	1, 34, 17, 23, 4, 13, 10, 3, 19, 1
A	A-n36-k5	1	1, 10, 13, 4, 7, 5, 20, 32, 1
A	A-n36-k5	2	1, 17, 2, 12, 25, 28, 26, 1
A	A-n36-k5	3	1, 11, 8, 27, 21, 6, 1
A	A-n36-k5	4	1, 29, 15, 35, 24, 3, 36, 9, 16, 1
A	A-n36-k5	5	1, 22, 19, 34, 30, 31, 18, 14, 33, 23, 1
A	A-n37-k5	1	1, 4, 26, 9, 28, 12, 10, 25, 24, 18, 2, 1
A	A-n37-k5	2	1, 35, 33, 29, 32, 27, 19, 36, 31, 1
A	A-n37-k5	3	1, 17, 23, 14, 6, 34, 5, 8, 1
A	A-n37-k5	4	1, 22, 30, 37, 16, 1
A	A-n37-k5	5	1, 13, 11, 7, 3, 20, 21, 15, 1
A	A-n37-k6	1	1, 27, 11, 12, 31, 16, 1
A	A-n37-k6	2	1, 19, 5, 18, 35, 20, 32, 7, 1
A	A-n37-k6	3	1, 22, 3, 29, 24, 23, 13, 14, 1
A	A-n37-k6	4	1, 26, 36, 33, 28, 1
A	A-n37-k6	5	1, 25, 17, 8, 15, 37, 30, 1

A	A-n37-k6	6	1, 21, 9, 2, 4, 6, 34, 10, 1
A	A-n38-k5	1	1, 35, 7, 26, 17, 5, 2, 4, 13, 11, 1
A	A-n38-k5	2	1, 33, 23, 28, 12, 6, 8, 1
A	A-n38-k5	3	1, 22, 30, 19, 29, 32, 38, 21, 1
A	A-n38-k5	4	1, 25, 15, 3, 16, 14, 37, 18, 10, 1
A	A-n38-k5	5	1, 20, 31, 27, 24, 36, 34, 9, 1
A	A-n39-k5	1	1, 10, 32, 2, 24, 21, 30, 16, 39, 18, 12, 9, 1
A	A-n39-k5	2	1, 15, 20, 26, 34, 13, 19, 5, 1
A	A-n39-k5	3	1, 11, 33, 23, 4, 8, 17, 6, 1
A	A-n39-k5	4	1, 22, 31, 14, 29, 28, 37, 7, 1
A	A-n39-k5	5	1, 27, 25, 35, 38, 36, 3, 1
A	A-n39-k6	1	1, 16, 14, 1
A	A-n39-k6	2	1, 25, 4, 39, 13, 10, 29, 30, 6, 1
A	A-n39-k6	3	1, 3, 20, 34, 26, 36, 15, 32, 38, 1
A	A-n39-k6	4	1, 27, 19, 33, 21, 31, 1
A	A-n39-k6	5	1, 12, 7, 2, 37, 18, 24, 22, 1
A	A-n39-k6	6	1, 23, 35, 28, 11, 17, 5, 9, 8, 1
A	A-n44-k6	1	1, 24, 31, 41, 30, 44, 14, 1
A	A-n44-k6	2	1, 8, 29, 28, 20, 16, 9, 1
A	A-n44-k6	3	1, 34, 11, 12, 38, 43, 33, 22, 6, 25, 1
A	A-n44-k6	4	1, 18, 35, 40, 13, 4, 26, 7, 1
A	A-n44-k6	5	1, 32, 27, 17, 21, 19, 36, 2, 1
A	A-n44-k6	6	1, 5, 23, 37, 10, 39, 15, 42, 3, 1
A	A-n45-k6	1	1, 23, 43, 17, 5, 37, 40, 13, 1
A	A-n45-k6	2	1, 2, 45, 7, 27, 30, 29, 1
A	A-n45-k6	3	1, 38, 28, 44, 6, 22, 34, 42, 9, 1
A	A-n45-k6	4	1, 24, 8, 33, 14, 11, 4, 21, 35, 1
A	A-n45-k6	5	1, 10, 16, 26, 3, 39, 32, 36, 15, 1
A	A-n45-k6	6	1, 25, 12, 20, 31, 41, 18, 19, 1
A	A-n45-k7	1	1, 4, 12, 44, 42, 28, 1
A	A-n45-k7	2	1, 22, 25, 45, 15, 11, 10, 1
A	A-n45-k7	3	1, 41, 8, 19, 14, 33, 40, 36, 1
A	A-n45-k7	4	1, 13, 29, 7, 5, 27, 35, 3, 1
A	A-n45-k7	5	1, 43, 2, 38, 31, 6, 23, 1
A	A-n45-k7	6	1, 34, 30, 20, 37, 32, 1
A	A-n45-k7	7	1, 9, 39, 16, 26, 24, 18, 21, 17, 1
A	A-n46-k7	1	1, 13, 36, 25, 17, 43, 1
A	A-n46-k7	2	1, 7, 15, 3, 38, 33, 1
A	A-n46-k7	3	1, 24, 46, 18, 10, 1
A	A-n46-k7	4	1, 21, 28, 2, 41, 35, 32, 30, 42, 22, 1
A	A-n46-k7	5	1, 4, 8, 11, 45, 26, 19, 1
A	A-n46-k7	6	1, 37, 6, 23, 40, 12, 9, 34, 16, 20, 29, 1
A	A-n46-k7	7	1, 39, 5, 31, 14, 27, 44, 1
A	A-n48-k7	1	1, 28, 16, 9, 40, 27, 21, 1
A	A-n48-k7	2	1, 19, 45, 36, 33, 37, 8, 46, 41, 1
A	A-n48-k7	3	1, 15, 18, 48, 42, 1
A	A-n48-k7	4	1, 30, 22, 31, 47, 14, 25, 5, 12, 43, 10, 35, 1

A	A-n48-k7	5	1, 38, 4, 26, 20, 39, 23, 7, 1
A	A-n48-k7	6	1, 13, 6, 2, 32, 44, 24, 1
A	A-n48-k7	7	1, 17, 29, 34, 11, 3, 1
A	A-n53-k7	1	1, 47, 38, 3, 37, 51, 44, 46, 1
A	A-n53-k7	2	1, 13, 17, 33, 16, 20, 24, 43, 49, 1
A	A-n53-k7	3	1, 26, 15, 22, 18, 8, 10, 48, 1
A	A-n53-k7	4	1, 42, 12, 25, 53, 14, 35, 1
A	A-n53-k7	5	1, 31, 45, 30, 50, 11, 27, 41, 19, 39, 1
A	A-n53-k7	6	1, 2, 5, 29, 23, 52, 28, 9, 36, 32, 1
A	A-n53-k7	7	1, 21, 7, 34, 4, 6, 40, 1
A	A-n54-k7	1	1, 15, 28, 3, 32, 23, 4, 12, 1
A	A-n54-k7	2	1, 35, 42, 26, 48, 52, 25, 43, 47, 33, 7, 1
A	A-n54-k7	3	1, 27, 22, 34, 10, 45, 39, 1
A	A-n54-k7	4	1, 14, 20, 9, 41, 49, 38, 13, 1
A	A-n54-k7	5	1, 24, 21, 8, 51, 40, 50, 30, 11, 16, 36, 1
A	A-n54-k7	6	1, 37, 2, 46, 18, 54, 1
A	A-n54-k7	7	1, 44, 5, 19, 6, 29, 17, 53, 31, 1
A	A-n55-k9	1	1, 13, 11, 6, 54, 41, 17, 39, 33, 1
A	A-n55-k9	2	1, 5, 8, 43, 32, 21, 38, 1
A	A-n55-k9	3	1, 48, 40, 50, 10, 36, 1
A	A-n55-k9	4	1, 31, 23, 20, 28, 14, 29, 1
A	A-n55-k9	5	1, 55, 24, 53, 25, 45, 22, 1
A	A-n55-k9	6	1, 37, 9, 30, 19, 49, 51, 42, 26, 1
A	A-n55-k9	7	1, 27, 3, 52, 12, 16, 1
A	A-n55-k9	8	1, 47, 15, 4, 18, 35, 34, 1
A	A-n55-k9	9	1, 7, 2, 46, 44, 1
A	A-n60-k9	1	1, 39, 8, 14, 30, 38, 9, 1
A	A-n60-k9	2	1, 37, 55, 6, 46, 43, 1
A	A-n60-k9	3	1, 40, 11, 23, 49, 59, 1
A	A-n60-k9	4	1, 33, 10, 52, 13, 57, 44, 1
A	A-n60-k9	5	1, 25, 3, 2, 45, 50, 31, 32, 29, 1
A	A-n60-k9	6	1, 19, 34, 53, 20, 36, 56, 51, 24, 48, 15, 1
A	A-n60-k9	7	1, 60, 58, 28, 18, 27, 16, 1
A	A-n60-k9	8	1, 42, 17, 21, 4, 26, 1
A	A-n60-k9	9	1, 47, 41, 12, 5, 22, 54, 7, 35, 1
A	A-n61-k9	1	1, 11, 60, 50, 3, 9, 54, 1
A	A-n61-k9	2	1, 18, 58, 53, 32, 12, 61, 29, 1
A	A-n61-k9	3	1, 8, 24, 20, 25, 4, 41, 1
A	A-n61-k9	4	1, 51, 14, 2, 17, 19, 36, 49, 35, 1
A	A-n61-k9	5	1, 56, 31, 15, 52, 10, 45, 40, 1
A	A-n61-k9	6	1, 39, 38, 16, 1
A	A-n61-k9	7	1, 59, 26, 7, 42, 6, 55, 47, 1
A	A-n61-k9	8	1, 13, 23, 5, 27, 44, 33, 43, 34, 1
A	A-n61-k9	9	1, 48, 57, 28, 22, 37, 30, 21, 46, 1
A	A-n62-k8	1	1, 60, 56, 27, 34, 59, 18, 51, 1
A	A-n62-k8	2	1, 20, 45, 23, 38, 17, 37, 53, 1
A	A-n62-k8	3	1, 12, 5, 22, 42, 24, 28, 40, 4, 43, 55, 1

A	A-n62-k8	4	1, 7, 58, 41, 57, 49, 1
A	A-n62-k8	5	1, 29, 16, 3, 54, 21, 8, 1
A	A-n62-k8	6	1, 13, 14, 15, 61, 6, 2, 10, 50, 36, 26, 32, 35, 1
A	A-n62-k8	7	1, 52, 11, 9, 44, 30, 33, 62, 1
A	A-n62-k8	8	1, 39, 46, 48, 25, 47, 31, 19, 1
A	A-n63-k10	1	1, 8, 47, 61, 22, 1
A	A-n63-k10	2	1, 60, 11, 20, 43, 4, 40, 46, 6, 18, 1
A	A-n63-k10	3	1, 9, 36, 28, 7, 13, 1
A	A-n63-k10	4	1, 31, 39, 5, 27, 14, 51, 63, 1
A	A-n63-k10	5	1, 54, 15, 3, 10, 42, 2, 1
A	A-n63-k10	6	1, 57, 30, 41, 26, 16, 50, 19, 1
A	A-n63-k10	7	1, 25, 34, 38, 17, 55, 1
A	A-n63-k10	8	1, 37, 53, 44, 29, 21, 49, 1
A	A-n63-k10	9	1, 62, 24, 56, 23, 59, 1
A	A-n63-k10	10	1, 32, 52, 45, 48, 12, 35, 33, 58, 1
A	A-n63-k9	1	1, 57, 62, 37, 50, 61, 14, 1
A	A-n63-k9	2	1, 10, 39, 32, 44, 47, 2, 19, 48, 1
A	A-n63-k9	3	1, 49, 4, 42, 43, 5, 24, 1
A	A-n63-k9	4	1, 45, 29, 23, 54, 34, 1
A	A-n63-k9	5	1, 16, 7, 38, 18, 36, 52, 63, 53, 40, 1
A	A-n63-k9	6	1, 22, 56, 33, 59, 41, 15, 58, 51, 1
A	A-n63-k9	7	1, 12, 13, 11, 17, 26, 8, 27, 1
A	A-n63-k9	8	1, 6, 46, 28, 60, 25, 9, 1
A	A-n63-k9	9	1, 20, 55, 30, 31, 3, 35, 21, 1
A	A-n64-k9	1	1, 7, 26, 30, 34, 11, 1
A	A-n64-k9	2	1, 4, 31, 45, 40, 41, 37, 8, 22, 27, 36, 15, 28, 1
A	A-n64-k9	3	1, 25, 61, 17, 9, 47, 48, 64, 3, 59, 1
A	A-n64-k9	4	1, 39, 18, 58, 12, 16, 46, 1
A	A-n64-k9	5	1, 19, 23, 13, 44, 29, 2, 1
A	A-n64-k9	6	1, 42, 57, 53, 14, 62, 33, 43, 49, 1
A	A-n64-k9	7	1, 21, 20, 52, 24, 38, 35, 10, 1
A	A-n64-k9	8	1, 5, 55, 6, 51, 1
A	A-n64-k9	9	1, 54, 56, 50, 32, 60, 63, 1
A	A-n65-k9	1	1, 48, 35, 32, 27, 7, 65, 47, 1
A	A-n65-k9	2	1, 21, 62, 59, 41, 11, 60, 1
A	A-n65-k9	3	1, 30, 54, 45, 57, 33, 46, 6, 1
A	A-n65-k9	4	1, 63, 29, 24, 2, 34, 56, 1
A	A-n65-k9	5	1, 31, 38, 36, 37, 4, 5, 50, 1
A	A-n65-k9	6	1, 61, 51, 17, 42, 3, 39, 43, 1
A	A-n65-k9	7	1, 18, 52, 40, 8, 12, 64, 55, 49, 58, 1
A	A-n65-k9	8	1, 16, 23, 10, 15, 28, 44, 1
A	A-n65-k9	9	1, 19, 13, 14, 25, 53, 9, 20, 26, 22, 1
A	A-n69-k9	1	1, 67, 14, 36, 46, 21, 42, 60, 9, 40, 1
A	A-n69-k9	2	1, 38, 18, 33, 11, 51, 17, 1
A	A-n69-k9	3	1, 20, 24, 31, 52, 7, 4, 16, 45, 1
A	A-n69-k9	4	1, 26, 12, 64, 47, 6, 55, 27, 53, 1
A	A-n69-k9	5	1, 68, 22, 61, 5, 56, 28, 1

A	A-n69-k9	6	1, 32, 19, 54, 3, 65, 62, 34, 50, 10, 59, 1
A	A-n69-k9	7	1, 49, 2, 37, 57, 63, 1
A	A-n69-k9	8	1, 29, 8, 35, 66, 48, 15, 43, 58, 25, 1
A	A-n69-k9	9	1, 23, 13, 69, 39, 41, 30, 44, 1
A	A-n80-k10	1	1, 65, 34, 16, 56, 10, 70, 57, 42, 26, 47, 1
A	A-n80-k10	2	1, 53, 29, 80, 19, 49, 15, 72, 1
A	A-n80-k10	3	1, 63, 24, 31, 60, 6, 45, 13, 1
A	A-n80-k10	4	1, 7, 25, 79, 69, 9, 38, 3, 35, 1
A	A-n80-k10	5	1, 52, 78, 4, 54, 67, 68, 71, 59, 39, 1
A	A-n80-k10	6	1, 50, 74, 37, 43, 14, 22, 41, 1
A	A-n80-k10	7	1, 2, 64, 12, 11, 8, 1
A	A-n80-k10	8	1, 77, 73, 55, 23, 5, 33, 46, 51, 1
A	A-n80-k10	9	1, 75, 30, 28, 18, 32, 61, 40, 1
A	A-n80-k10	10	1, 21, 76, 62, 17, 44, 58, 20, 27, 36, 66, 48, 1
B	B-n31-k5	1	1, 24, 8, 14, 18, 10, 7, 1
B	B-n31-k5	2	1, 3, 11, 28, 21, 1
B	B-n31-k5	3	1, 5, 31, 9, 13, 29, 27, 1
B	B-n31-k5	4	1, 4, 2, 20, 25, 12, 16, 15, 1
B	B-n31-k5	5	1, 22, 17, 19, 26, 6, 23, 30, 1
B	B-n34-k5	1	1, 13, 27, 9, 24, 11, 34, 1
B	B-n34-k5	2	1, 8, 22, 3, 23, 33, 21, 5, 1
B	B-n34-k5	3	1, 4, 28, 30, 25, 15, 17, 19, 20, 2, 1
B	B-n34-k5	4	1, 12, 31, 10, 1
B	B-n34-k5	5	1, 6, 7, 26, 18, 32, 29, 14, 16, 1
B	B-n35-k5	1	1, 18, 30, 9, 1
B	B-n35-k5	2	1, 22, 26, 32, 7, 34, 4, 27, 19, 1
B	B-n35-k5	3	1, 23, 15, 5, 29, 25, 28, 35, 1
B	B-n35-k5	4	1, 16, 31, 6, 12, 20, 3, 33, 14, 1
B	B-n35-k5	5	1, 13, 11, 17, 24, 2, 10, 8, 21, 1
B	B-n38-k6	1	1, 8, 6, 13, 30, 25, 35, 9, 18, 37, 1
B	B-n38-k6	2	1, 4, 38, 3, 36, 14, 1
B	B-n38-k6	3	1, 32, 10, 20, 7, 26, 28, 1
B	B-n38-k6	4	1, 33, 29, 16, 2, 11, 21, 1
B	B-n38-k6	5	1, 24, 19, 34, 5, 31, 1
B	B-n38-k6	6	1, 27, 17, 23, 12, 22, 15, 1
B	B-n39-k5	1	1, 9, 15, 8, 30, 1
B	B-n39-k5	2	1, 33, 3, 34, 22, 25, 38, 24, 31, 17, 23, 27, 1
B	B-n39-k5	3	1, 37, 18, 28, 39, 36, 5, 29, 1
B	B-n39-k5	4	1, 4, 32, 2, 26, 13, 12, 16, 6, 1
B	B-n39-k5	5	1, 10, 35, 20, 21, 7, 19, 14, 11, 1
B	B-n41-k6	1	1, 6, 16, 23, 31, 8, 37, 9, 14, 1
B	B-n41-k6	2	1, 26, 7, 12, 34, 35, 29, 1
B	B-n41-k6	3	1, 2, 30, 15, 18, 24, 11, 32, 1
B	B-n41-k6	4	1, 33, 13, 41, 21, 17, 1
B	B-n41-k6	5	1, 27, 20, 28, 39, 40, 5, 1
B	B-n41-k6	6	1, 25, 36, 3, 10, 38, 4, 19, 22, 1
B	B-n43-k6	1	1, 17, 37, 3, 9, 39, 32, 11, 1

B	B-n43-k6	2	1, 5, 36, 28, 30, 33, 26, 41, 42, 1
B	B-n43-k6	3	1, 13, 24, 2, 38, 29, 15, 20, 1
B	B-n43-k6	4	1, 34, 21, 22, 6, 19, 16, 8, 10, 1
B	B-n43-k6	5	1, 31, 14, 12, 4, 1
B	B-n43-k6	6	1, 27, 43, 18, 40, 35, 7, 25, 23, 1
B	B-n44-k7	1	1, 19, 20, 33, 14, 42, 38, 35, 22, 7, 1
B	B-n44-k7	2	1, 3, 37, 15, 40, 25, 27, 11, 1
B	B-n44-k7	3	1, 10, 24, 12, 1
B	B-n44-k7	4	1, 36, 28, 32, 44, 34, 16, 21, 1
B	B-n44-k7	5	1, 41, 2, 13, 4, 29, 26, 1
B	B-n44-k7	6	1, 8, 17, 30, 18, 6, 1
B	B-n44-k7	7	1, 5, 9, 31, 43, 23, 39, 1
B	B-n45-k5	1	1, 38, 33, 8, 35, 28, 15, 45, 19, 1
B	B-n45-k5	2	1, 24, 31, 42, 44, 22, 13, 7, 6, 41, 43, 1
B	B-n45-k5	3	1, 11, 4, 32, 39, 21, 36, 25, 37, 1
B	B-n45-k5	4	1, 23, 9, 5, 17, 30, 12, 20, 34, 27, 26, 1
B	B-n45-k5	5	1, 29, 3, 16, 14, 10, 2, 18, 40, 1
B	B-n45-k6	1	1, 19, 16, 4, 10, 32, 22, 1
B	B-n45-k6	2	1, 5, 35, 7, 27, 30, 44, 1
B	B-n45-k6	3	1, 36, 45, 42, 39, 41, 34, 8, 21, 1
B	B-n45-k6	4	1, 40, 9, 23, 29, 13, 33, 24, 38, 20, 28, 1
B	B-n45-k6	5	1, 26, 43, 2, 6, 31, 1
B	B-n45-k6	6	1, 25, 18, 15, 3, 37, 11, 12, 17, 14, 1
B	B-n50-k7	1	1, 48, 34, 40, 27, 14, 2, 1
B	B-n50-k7	2	1, 35, 50, 45, 20, 18, 49, 3, 33, 1
B	B-n50-k7	3	1, 32, 42, 8, 1
B	B-n50-k7	4	1, 36, 5, 30, 24, 47, 10, 39, 13, 23, 1
B	B-n50-k7	5	1, 41, 29, 16, 7, 1
B	B-n50-k7	6	1, 31, 26, 19, 15, 44, 25, 28, 17, 11, 43, 4, 1
B	B-n50-k7	7	1, 22, 46, 37, 6, 9, 21, 12, 38, 1
B	B-n50-k8	1	1, 35, 49, 41, 8, 34, 13, 28, 38, 7, 1
B	B-n50-k8	2	1, 3, 10, 26, 50, 44, 12, 37, 21, 31, 1
B	B-n50-k8	3	1, 39, 42, 46, 40, 16, 1
B	B-n50-k8	4	1, 2, 22, 43, 18, 9, 1
B	B-n50-k8	5	1, 33, 29, 30, 20, 1
B	B-n50-k8	6	1, 15, 6, 47, 1
B	B-n50-k8	7	1, 45, 32, 23, 24, 36, 11, 14, 4, 1
B	B-n50-k8	8	1, 17, 25, 48, 27, 19, 5, 1
B	B-n51-k7	1	1, 18, 11, 3, 39, 35, 19, 1
B	B-n51-k7	2	1, 9, 13, 38, 33, 32, 28, 42, 1
B	B-n51-k7	3	1, 48, 26, 47, 23, 25, 31, 50, 2, 1
B	B-n51-k7	4	1, 12, 5, 44, 27, 29, 8, 7, 1
B	B-n51-k7	5	1, 22, 45, 21, 10, 46, 51, 24, 16, 1
B	B-n51-k7	6	1, 6, 40, 30, 14, 43, 15, 1
B	B-n51-k7	7	1, 37, 41, 17, 49, 34, 20, 4, 36, 1
B	B-n52-k7	1	1, 3, 49, 39, 10, 17, 33, 35, 20, 14, 1
B	B-n52-k7	2	1, 38, 30, 28, 9, 6, 11, 45, 1

B	B-n52-k7	3	1, 27, 23, 18, 47, 50, 16, 51, 13, 24, 1
B	B-n52-k7	4	1, 34, 36, 44, 8, 52, 1
B	B-n52-k7	5	1, 43, 21, 19, 31, 2, 1
B	B-n52-k7	6	1, 41, 37, 26, 7, 48, 42, 1
B	B-n52-k7	7	1, 5, 46, 22, 12, 29, 15, 40, 25, 32, 4, 1
B	B-n56-k7	1	1, 54, 7, 25, 42, 5, 16, 8, 20, 1
B	B-n56-k7	2	1, 37, 33, 50, 34, 14, 1
B	B-n56-k7	3	1, 3, 31, 35, 55, 32, 21, 56, 12, 1
B	B-n56-k7	4	1, 22, 27, 40, 48, 43, 38, 30, 46, 1
B	B-n56-k7	5	1, 41, 6, 13, 10, 39, 51, 23, 24, 19, 1
B	B-n56-k7	6	1, 4, 49, 36, 17, 28, 53, 2, 26, 1
B	B-n56-k7	7	1, 47, 9, 29, 52, 45, 15, 18, 11, 44, 1
B	B-n57-k7	1	1, 6, 45, 22, 17, 55, 36, 30, 41, 1
B	B-n57-k7	2	1, 26, 29, 13, 54, 9, 10, 39, 37, 19, 25, 49, 1
B	B-n57-k7	3	1, 16, 12, 32, 11, 28, 1
B	B-n57-k7	4	1, 15, 8, 44, 21, 48, 33, 5, 1
B	B-n57-k7	5	1, 56, 7, 46, 50, 34, 14, 47, 24, 1
B	B-n57-k7	6	1, 2, 3, 31, 51, 53, 23, 43, 40, 38, 1
B	B-n57-k7	7	1, 4, 52, 35, 18, 42, 20, 27, 57, 1
B	B-n57-k9	1	1, 33, 41, 4, 51, 47, 32, 18, 11, 1
B	B-n57-k9	2	1, 56, 10, 29, 19, 28, 2, 34, 1
B	B-n57-k9	3	1, 15, 35, 20, 14, 17, 24, 1
B	B-n57-k9	4	1, 7, 23, 57, 1
B	B-n57-k9	5	1, 8, 21, 38, 3, 45, 12, 43, 1
B	B-n57-k9	6	1, 44, 55, 53, 25, 46, 1
B	B-n57-k9	7	1, 6, 48, 37, 22, 42, 30, 54, 1
B	B-n57-k9	8	1, 9, 16, 26, 52, 27, 40, 1
B	B-n57-k9	9	1, 5, 31, 50, 36, 49, 39, 13, 1
B	B-n63-k10	1	1, 61, 45, 17, 41, 4, 5, 32, 1
B	B-n63-k10	2	1, 19, 51, 2, 1
B	B-n63-k10	3	1, 23, 40, 27, 22, 3, 43, 1
B	B-n63-k10	4	1, 16, 9, 47, 60, 57, 50, 1
B	B-n63-k10	5	1, 39, 12, 18, 25, 44, 15, 59, 13, 55, 42, 1
B	B-n63-k10	6	1, 38, 11, 24, 54, 58, 34, 62, 1
B	B-n63-k10	7	1, 35, 26, 36, 7, 52, 20, 53, 63, 1
B	B-n63-k10	8	1, 48, 10, 31, 33, 49, 1
B	B-n63-k10	9	1, 6, 56, 14, 30, 29, 1
B	B-n63-k10	10	1, 46, 21, 8, 37, 28, 1
B	B-n64-k9	1	1, 63, 53, 24, 26, 40, 43, 64, 58, 1
B	B-n64-k9	2	1, 50, 44, 21, 3, 1
B	B-n64-k9	3	1, 8, 32, 55, 42, 27, 2, 1
B	B-n64-k9	4	1, 4, 33, 6, 60, 52, 56, 10, 38, 1
B	B-n64-k9	5	1, 18, 39, 41, 35, 28, 22, 9, 47, 31, 1
B	B-n64-k9	6	1, 17, 14, 57, 62, 19, 11, 13, 1
B	B-n64-k9	7	1, 59, 54, 7, 46, 45, 34, 1
B	B-n64-k9	8	1, 20, 16, 25, 5, 15, 36, 37, 49, 61, 23, 1
B	B-n64-k9	9	1, 51, 48, 30, 12, 29, 1

B	B-n66-k9	1	1, 2, 56, 19, 18, 54, 21, 42, 1
B	B-n66-k9	2	1, 10, 29, 17, 62, 48, 1
B	B-n66-k9	3	1, 51, 7, 37, 22, 13, 1
B	B-n66-k9	4	1, 43, 34, 66, 9, 41, 59, 33, 1
B	B-n66-k9	5	1, 52, 35, 44, 14, 23, 57, 36, 20, 46, 1
B	B-n66-k9	6	1, 58, 31, 26, 64, 47, 27, 11, 3, 1
B	B-n66-k9	7	1, 53, 55, 24, 32, 12, 39, 8, 1
B	B-n66-k9	8	1, 63, 49, 60, 45, 65, 38, 1
B	B-n66-k9	9	1, 28, 6, 5, 61, 16, 4, 30, 40, 15, 50, 25, 1
B	B-n67-k10	1	1, 36, 37, 7, 35, 65, 41, 43, 40, 1
B	B-n67-k10	2	1, 38, 24, 9, 21, 51, 15, 1
B	B-n67-k10	3	1, 63, 62, 31, 58, 4, 61, 1
B	B-n67-k10	4	1, 48, 45, 54, 1
B	B-n67-k10	5	1, 12, 50, 2, 20, 42, 17, 1
B	B-n67-k10	6	1, 27, 3, 66, 33, 60, 25, 30, 52, 67, 1
B	B-n67-k10	7	1, 10, 49, 22, 26, 64, 6, 1
B	B-n67-k10	8	1, 14, 13, 46, 32, 19, 34, 1
B	B-n67-k10	9	1, 18, 39, 44, 28, 11, 8, 5, 47, 1
B	B-n67-k10	10	1, 29, 16, 57, 56, 55, 23, 53, 59, 1
B	B-n68-k9	1	1, 64, 26, 24, 41, 51, 39, 54, 20, 1
B	B-n68-k9	2	1, 7, 59, 11, 25, 19, 29, 1
B	B-n68-k9	3	1, 38, 42, 9, 3, 23, 50, 1
B	B-n68-k9	4	1, 48, 8, 44, 66, 28, 65, 1
B	B-n68-k9	5	1, 46, 60, 17, 35, 45, 62, 52, 58, 1
B	B-n68-k9	6	1, 56, 40, 12, 4, 30, 68, 33, 37, 43, 1
B	B-n68-k9	7	1, 13, 49, 10, 27, 21, 53, 18, 1
B	B-n68-k9	8	1, 67, 47, 14, 6, 15, 63, 55, 31, 22, 1
B	B-n68-k9	9	1, 32, 36, 61, 5, 2, 57, 34, 16, 1
B	B-n78-k10	1	1, 28, 58, 71, 43, 1
B	B-n78-k10	2	1, 26, 19, 69, 67, 61, 7, 63, 1
B	B-n78-k10	3	1, 18, 73, 59, 39, 57, 68, 70, 44, 22, 25, 1
B	B-n78-k10	4	1, 3, 6, 31, 76, 66, 48, 53, 2, 1
B	B-n78-k10	5	1, 62, 49, 14, 34, 56, 36, 60, 5, 1
B	B-n78-k10	6	1, 46, 47, 21, 55, 4, 30, 1
B	B-n78-k10	7	1, 64, 38, 8, 33, 74, 32, 77, 1
B	B-n78-k10	8	1, 78, 24, 45, 27, 12, 65, 41, 54, 11, 17, 1
B	B-n78-k10	9	1, 51, 40, 75, 29, 20, 10, 37, 15, 42, 1
B	B-n78-k10	10	1, 50, 52, 35, 13, 16, 72, 23, 9, 1
E	E-n101-k14	1	1, 54, 59, 41, 27, 13, 77, 29, 1
E	E-n101-k14	2	1, 26, 40, 68, 24, 1
E	E-n101-k14	3	1, 63, 50, 65, 12, 64, 91, 33, 1
E	E-n101-k14	4	1, 53, 8, 89, 32, 11, 31, 71, 2, 70, 1
E	E-n101-k14	5	1, 20, 37, 48, 49, 83, 1
E	E-n101-k14	6	1, 4, 80, 79, 34, 82, 10, 52, 21, 1
E	E-n101-k14	7	1, 22, 74, 73, 75, 57, 76, 23, 42, 1
E	E-n101-k14	8	1, 43, 44, 15, 92, 86, 101, 99, 38, 93, 1
E	E-n101-k14	9	1, 78, 69, 81, 30, 25, 56, 5, 55, 1

E	E-n101-k14	10	1, 97, 17, 87, 39, 45, 1
E	E-n101-k14	11	1, 51, 35, 36, 72, 66, 67, 1
E	E-n101-k14	12	1, 95, 98, 88, 16, 58, 3, 14, 1
E	E-n101-k14	13	1, 28, 90, 46, 18, 62, 60, 100, 7, 1
E	E-n101-k14	14	1, 19, 9, 47, 84, 61, 85, 6, 94, 96, 1
E	E-n101-k8	1	1, 7, 97, 100, 60, 93, 38, 99, 94, 86, 92, 17, 62, 85, 6, 1
E	E-n101-k8	2	1, 59, 41, 22, 74, 73, 75, 76, 23, 42, 16, 58, 3, 88, 14, 1
E	E-n101-k8	3	1, 5, 40, 57, 24, 68, 26, 56, 1
E	E-n101-k8	4	1, 90, 19, 61, 84, 18, 46, 9, 83, 47, 37, 50, 48, 49, 8, 53, 1
E	E-n101-k8	5	1, 70, 2, 51, 34, 82, 10, 52, 21, 72, 66, 36, 35, 79, 80, 4, 1
E	E-n101-k8	6	1, 32, 89, 11, 63, 20, 12, 65, 64, 91, 33, 67, 31, 71, 1
E	E-n101-k8	7	1, 54, 27, 13, 69, 81, 55, 25, 30, 78, 77, 29, 28, 1
E	E-n101-k8	8	1, 95, 96, 101, 15, 45, 87, 39, 44, 43, 98, 1
E	E-n22-k4	1	1, 14, 12, 5, 4, 9, 11, 1
E	E-n22-k4	2	1, 15, 22, 20, 17, 1
E	E-n22-k4	3	1, 18, 21, 19, 16, 13, 1
E	E-n22-k4	4	1, 10, 8, 6, 3, 2, 7, 1
E	E-n23-k3	1	1, 8, 10, 9, 6, 5, 22, 1
E	E-n23-k3	2	1, 13, 12, 7, 2, 3, 4, 17, 16, 15, 18, 23, 21, 20, 19, 1
E	E-n23-k3	3	1, 14, 11, 1
E	E-n30-k3	1	1, 22, 7, 25, 26, 30, 28, 29, 27, 1
E	E-n30-k3	2	1, 21, 4, 5, 2, 6, 3, 23, 1
E	E-n30-k3	3	1, 19, 24, 11, 12, 13, 9, 15, 10, 18, 8, 14, 17, 16, 20, 1
E	E-n33-k4	1	1, 31, 32, 15, 16, 2, 14, 3, 4, 1
E	E-n33-k4	2	1, 12, 19, 20, 22, 21, 23, 24, 25, 26, 18, 1
E	E-n33-k4	3	1, 27, 28, 17, 29, 30, 1
E	E-n33-k4	4	1, 5, 6, 7, 8, 9, 10, 11, 33, 13, 1
E	E-n51-k5	1	1, 13, 6, 34, 46, 16, 45, 38, 18, 48, 1
E	E-n51-k5	2	1, 19, 26, 14, 42, 41, 20, 43, 5, 1
E	E-n51-k5	3	1, 12, 3, 21, 36, 37, 4, 29, 32, 23, 2, 1
E	E-n51-k5	4	1, 50, 11, 40, 31, 35, 22, 30, 17, 51, 10, 39, 33, 1
E	E-n51-k5	5	1, 47, 28, 49, 9, 27, 8, 24, 44, 25, 15, 7, 1
E	E-n76-k10	1	1, 40, 73, 59, 11, 32, 39, 54, 1
E	E-n76-k10	2	1, 17, 50, 25, 57, 24, 64, 48, 49, 1
E	E-n76-k10	3	1, 76, 5, 68, 27, 13, 18, 1
E	E-n76-k10	4	1, 31, 75, 22, 29, 63, 74, 3, 1
E	E-n76-k10	5	1, 2, 44, 42, 43, 65, 23, 62, 70, 37, 1
E	E-n76-k10	6	1, 15, 60, 12, 67, 66, 1
E	E-n76-k10	7	1, 7, 34, 52, 4, 41, 8, 35, 1
E	E-n76-k10	8	1, 47, 9, 36, 20, 55, 14, 28, 53, 1
E	E-n76-k10	9	1, 46, 30, 6, 38, 72, 61, 71, 21, 16, 58, 1
E	E-n76-k10	10	1, 69, 45, 33, 51, 19, 56, 26, 10, 1
E	E-n76-k14	1	1, 30, 6, 48, 37, 16, 47, 1
E	E-n76-k14	2	1, 45, 51, 26, 10, 40, 1
E	E-n76-k14	3	1, 76, 65, 25, 32, 1
E	E-n76-k14	4	1, 11, 39, 66, 67, 1
E	E-n76-k14	5	1, 68, 59, 73, 56, 19, 33, 1

E	E-n76-k14	6	1, 69, 74, 63, 23, 29, 62, 70, 1
E	E-n76-k14	7	1, 27, 13, 41, 18, 75, 1
E	E-n76-k14	8	1, 2, 44, 43, 42, 57, 24, 50, 1
E	E-n76-k14	9	1, 5, 28, 14, 53, 35, 1
E	E-n76-k14	10	1, 54, 12, 15, 1
E	E-n76-k14	11	1, 7, 34, 64, 17, 4, 52, 1
E	E-n76-k14	12	1, 31, 49, 22, 3, 1
E	E-n76-k14	13	1, 46, 72, 61, 71, 21, 38, 58, 1
E	E-n76-k14	14	1, 8, 36, 60, 20, 55, 9, 1
E	E-n76-k7	1	1, 5, 35, 53, 55, 14, 58, 16, 30, 28, 46, 31, 69, 1
E	E-n76-k7	2	1, 76, 52, 17, 64, 24, 57, 50, 25, 19, 51, 33, 45, 4, 1
E	E-n76-k7	3	1, 40, 32, 56, 26, 10, 1
E	E-n76-k7	4	1, 3, 75, 22, 70, 37, 72, 61, 71, 21, 38, 6, 48, 49, 1
E	E-n76-k7	5	1, 68, 47, 27, 39, 11, 59, 73, 13, 41, 18, 1
E	E-n76-k7	6	1, 8, 36, 54, 12, 66, 67, 60, 15, 20, 9, 1
E	E-n76-k7	7	1, 7, 34, 74, 2, 44, 42, 43, 65, 23, 62, 29, 63, 1
E	E-n76-k8	1	1, 76, 13, 73, 40, 32, 11, 39, 59, 27, 1
E	E-n76-k8	2	1, 54, 12, 66, 67, 60, 15, 1
E	E-n76-k8	3	1, 18, 41, 10, 33, 45, 4, 52, 34, 1
E	E-n76-k8	4	1, 68, 35, 47, 9, 8, 36, 20, 55, 53, 69, 1
E	E-n76-k8	5	1, 74, 2, 44, 42, 43, 65, 23, 62, 29, 63, 75, 1
E	E-n76-k8	6	1, 7, 17, 64, 24, 57, 50, 25, 19, 51, 56, 26, 1
E	E-n76-k8	7	1, 6, 38, 21, 71, 61, 72, 70, 37, 48, 22, 49, 1
E	E-n76-k8	8	1, 5, 46, 28, 14, 58, 16, 30, 31, 3, 1
G	G-n262-k25	1	1, 63, 205, 74, 199, 55, 138, 66, 207, 174, 261, 28, 1
G	G-n262-k25	2	1, 223, 38, 201, 64, 217, 47, 219, 145, 181, 153, 251, 1
G	G-n262-k25	3	1, 112, 121, 237, 197, 93, 213, 185, 1
G	G-n262-k25	4	1, 123, 243, 255, 57, 67, 48, 56, 104, 114, 81, 195, 198, 43, 11, 239, 242, 1
G	G-n262-k25	5	1, 65, 126, 252, 115, 149, 27, 231, 171, 3, 196, 1
G	G-n262-k25	6	1, 258, 254, 179, 118, 241, 99, 108, 15, 144, 175, 24, 23, 1
G	G-n262-k25	7	1, 204, 58, 132, 226, 44, 224, 39, 147, 1
G	G-n262-k25	8	1, 238, 244, 26, 247, 8, 165, 141, 142, 75, 113, 1
G	G-n262-k25	9	1, 208, 167, 73, 186, 20, 80, 262, 6, 192, 10, 1
G	G-n262-k25	10	1, 225, 128, 246, 148, 135, 70, 1
G	G-n262-k25	11	1, 31, 19, 256, 125, 50, 13, 134, 91, 62, 222, 130, 131, 54, 229, 1
G	G-n262-k25	12	1, 218, 84, 37, 221, 151, 184, 59, 1
G	G-n262-k25	13	1, 209, 200, 180, 162, 188, 94, 92, 78, 46, 129, 49, 45, 1
G	G-n262-k25	14	1, 22, 101, 146, 116, 5, 82, 187, 173, 1
G	G-n262-k25	15	1, 194, 102, 215, 236, 83, 158, 103, 164, 1
G	G-n262-k25	16	1, 228, 21, 110, 211, 42, 111, 61, 159, 160, 1
G	G-n262-k25	17	1, 154, 190, 107, 122, 249, 90, 168, 14, 12, 250, 178, 1
G	G-n262-k25	18	1, 193, 120, 95, 152, 87, 150, 248, 89, 51, 1
G	G-n262-k25	19	1, 16, 220, 137, 41, 52, 68, 163, 53, 7, 1
G	G-n262-k25	20	1, 124, 119, 4, 136, 85, 17, 166, 117, 127, 105, 216, 214, 9, 191, 1

G	G-n262-k25	21	1, 100, 71, 109, 155, 77, 156, 233, 232, 203, 140, 2, 230, 72, 1
G	G-n262-k25	22	1, 40, 234, 177, 227, 33, 79, 202, 25, 18, 259, 176, 169, 1
G	G-n262-k25	23	1, 97, 139, 210, 245, 253, 34, 235, 143, 69, 32, 206, 212, 1
G	G-n262-k25	24	1, 161, 29, 157, 260, 240, 36, 183, 96, 35, 88, 189, 257, 1
G	G-n262-k25	25	1, 76, 60, 172, 182, 170, 133, 98, 30, 106, 86, 1
M	M-n101-k10	1	1, 21, 48, 44, 42, 70, 67, 63, 75, 64, 66, 68, 1
M	M-n101-k10	2	1, 82, 79, 77, 72, 71, 74, 78, 80, 81, 1
M	M-n101-k10	3	1, 33, 34, 32, 36, 38, 39, 40, 37, 35, 1
M	M-n101-k10	4	1, 99, 97, 96, 95, 93, 94, 98, 101, 100, 1
M	M-n101-k10	5	1, 53, 51, 52, 49, 60, 61, 59, 57, 55, 1
M	M-n101-k10	6	1, 24, 14, 18, 19, 20, 16, 17, 15, 13, 1
M	M-n101-k10	7	1, 91, 90, 88, 87, 84, 83, 85, 86, 89, 92, 1
M	M-n101-k10	8	1, 23, 27, 29, 31, 30, 28, 26, 25, 1
M	M-n101-k10	9	1, 50, 47, 46, 45, 43, 41, 58, 54, 56, 69, 65, 62, 73, 1
M	M-n101-k10	10	1, 22, 11, 12, 9, 10, 7, 8, 6, 4, 5, 3, 2, 76, 1
M	M-n121-k7	1	1, 7, 8, 10, 9, 13, 14, 15, 16, 12, 11, 6, 5, 4, 2, 3, 1
M	M-n121-k7	2	1, 105, 41, 44, 46, 49, 52, 51, 50, 48, 47, 45, 42, 38, 39, 43, 40, 1
M	M-n121-k7	3	1, 101, 69, 80, 81, 79, 78, 77, 74, 75, 73, 76, 72, 71, 70, 68, 1
M	M-n121-k7	4	1, 100, 22, 21, 24, 27, 29, 33, 36, 30, 37, 35, 32, 31, 34, 28, 25, 23, 26, 20, 17, 18, 1
M	M-n121-k7	5	1, 53, 55, 58, 60, 66, 62, 63, 65, 67, 64, 61, 57, 59, 56, 54, 1
M	M-n121-k7	6	1, 83, 112, 82, 120, 121, 106, 107, 103, 102, 108, 104, 117, 99, 111, 116, 110, 98, 97, 96, 1
M	M-n121-k7	7	1, 89, 87, 86, 113, 85, 118, 114, 84, 109, 119, 19, 115, 91, 92, 95, 94, 93, 90, 88, 1
M	M-n151-k12	1	1, 147, 90, 61, 84, 19, 107, 115, 9, 126, 46, 85, 60, 95, 1
M	M-n151-k12	2	1, 7, 97, 100, 105, 94, 99, 92, 86, 6, 119, 148, 1
M	M-n151-k12	3	1, 53, 89, 149, 63, 12, 65, 64, 127, 91, 33, 109, 11, 1
M	M-n151-k12	4	1, 8, 124, 20, 108, 50, 144, 37, 47, 125, 48, 49, 83, 1
M	M-n151-k12	5	1, 81, 151, 25, 135, 55, 131, 56, 26, 68, 40, 140, 57, 5, 111, 1
M	M-n151-k12	6	1, 74, 73, 75, 76, 24, 134, 23, 42, 146, 16, 44, 58, 1
M	M-n151-k12	7	1, 14, 118, 96, 98, 93, 38, 101, 120, 15, 143, 43, 88, 145, 116, 3, 138, 1
M	M-n151-k12	8	1, 28, 133, 70, 128, 32, 102, 2, 51, 103, 78, 117, 77, 112, 1
M	M-n151-k12	9	1, 71, 123, 31, 132, 129, 21, 67, 66, 72, 104, 69, 1
M	M-n151-k12	10	1, 4, 80, 130, 122, 30, 79, 35, 36, 137, 136, 121, 10, 52, 82, 34, 1
M	M-n151-k12	11	1, 62, 17, 142, 45, 39, 141, 87, 114, 18, 1
M	M-n151-k12	12	1, 54, 106, 27, 29, 139, 13, 110, 150, 22, 41, 59, 113, 1
M	M-n200-k17	1	1, 157, 54, 106, 27, 139, 155, 29, 185, 77, 112, 28, 1
M	M-n200-k17	2	1, 20, 108, 176, 65, 50, 144, 37, 48, 169, 1
M	M-n200-k17	3	1, 147, 168, 191, 128, 53, 19, 167, 90, 148, 113, 1
M	M-n200-k17	4	1, 41, 153, 59, 138, 14, 118, 98, 96, 97, 95, 184, 7, 1
M	M-n200-k17	5	1, 181, 3, 179, 116, 146, 58, 16, 44, 141, 114, 45, 173, 145, 1
M	M-n200-k17	6	1, 101, 192, 142, 87, 39, 120, 193, 15, 143, 43, 88, 1
M	M-n200-k17	7	1, 150, 196, 13, 110, 178, 55, 131, 180, 5, 111, 199, 22, 1

M	M-n200-k17	8	1, 11, 190, 109, 91, 33, 132, 182, 64, 127, 12, 63, 160, 1
M	M-n200-k17	9	1, 51, 21, 129, 161, 31, 123, 71, 32, 163, 102, 2, 177, 133, 70, 1
M	M-n200-k17	10	1, 103, 158, 34, 82, 186, 80, 79, 121, 10, 189, 52, 1
M	M-n200-k17	11	1, 100, 105, 60, 94, 86, 62, 17, 92, 194, 99, 38, 152, 93, 1
M	M-n200-k17	12	1, 74, 75, 172, 42, 23, 134, 76, 24, 187, 57, 198, 73, 1
M	M-n200-k17	13	1, 154, 83, 9, 47, 125, 49, 124, 149, 89, 183, 8, 195, 107, 1
M	M-n200-k17	14	1, 197, 117, 159, 4, 130, 170, 122, 30, 25, 164, 135, 81, 69, 1
M	M-n200-k17	15	1, 61, 119, 6, 85, 174, 18, 46, 126, 175, 115, 200, 84, 1
M	M-n200-k17	16	1, 104, 162, 72, 136, 165, 35, 36, 137, 66, 67, 1
M	M-n200-k17	17	1, 78, 151, 166, 56, 26, 171, 68, 40, 188, 140, 156, 1
P	P-n101-k4	1	1, 14, 98, 43, 44, 16, 88, 3, 58, 42, 23, 76, 57, 24, 68, 40, 5, 73, 75, 74, 22, 41, 59, 54, 1
P	P-n101-k4	2	1, 29, 27, 13, 69, 81, 55, 56, 26, 25, 30, 79, 35, 36, 72, 66, 67, 21, 52, 10, 82, 34, 80, 4, 78, 77, 51, 2, 70, 28, 1
P	P-n101-k4	3	1, 95, 96, 93, 60, 97, 100, 94, 86, 99, 38, 101, 92, 62, 17, 45, 15, 39, 87, 18, 85, 6, 61, 7, 90, 1
P	P-n101-k4	4	1, 84, 46, 9, 47, 48, 37, 50, 20, 63, 12, 65, 64, 91, 33, 11, 31, 71, 32, 89, 8, 49, 83, 19, 53, 1
P	P-n16-k8	1	1, 3, 1
P	P-n16-k8	2	1, 11, 13, 16, 1
P	P-n16-k8	3	1, 7, 1
P	P-n16-k8	4	1, 6, 10, 4, 1
P	P-n16-k8	5	1, 8, 15, 1
P	P-n16-k8	6	1, 14, 9, 1
P	P-n16-k8	7	1, 5, 12, 1
P	P-n16-k8	8	1, 2, 1
P	P-n19-k2	1	1, 7, 9, 17, 18, 4, 13, 15, 12, 5, 1
P	P-n19-k2	2	1, 19, 6, 14, 16, 10, 8, 3, 11, 2, 1
P	P-n20-k2	1	1, 2, 11, 3, 8, 14, 10, 17, 15, 6, 20, 1
P	P-n20-k2	2	1, 7, 9, 18, 19, 4, 13, 16, 12, 5, 1
P	P-n21-k2	1	1, 17, 2, 11, 9, 19, 20, 4, 13, 16, 12, 5, 1
P	P-n21-k2	2	1, 7, 3, 14, 10, 18, 15, 6, 8, 21, 1
P	P-n22-k2	1	1, 17, 2, 11, 9, 19, 20, 4, 13, 16, 12, 5, 1
P	P-n22-k2	2	1, 7, 3, 14, 10, 18, 15, 22, 8, 6, 21, 1
P	P-n22-k8	1	1, 19, 21, 1
P	P-n22-k8	2	1, 13, 18, 22, 1
P	P-n22-k8	3	1, 14, 12, 1
P	P-n22-k8	4	1, 15, 20, 1
P	P-n22-k8	5	1, 16, 17, 1
P	P-n22-k8	6	1, 6, 10, 1
P	P-n22-k8	7	1, 11, 3, 2, 7, 9, 1
P	P-n22-k8	8	1, 8, 4, 5, 1
P	P-n23-k8	1	1, 4, 20, 19, 1
P	P-n23-k8	2	1, 11, 13, 16, 12, 1
P	P-n23-k8	3	1, 22, 21, 7, 1
P	P-n23-k8	4	1, 6, 9, 1
P	P-n23-k8	5	1, 3, 23, 1

P	P-n23-k8	6	1, 5, 8, 1
P	P-n23-k8	7	1, 14, 10, 18, 1
P	P-n23-k8	8	1, 2, 17, 15, 1
P	P-n40-k5	1	1, 39, 10, 31, 35, 22, 17, 12, 33, 1
P	P-n40-k5	2	1, 13, 18, 38, 16, 34, 40, 11, 6, 1
P	P-n40-k5	3	1, 19, 26, 14, 20, 5, 1
P	P-n40-k5	4	1, 28, 2, 9, 27, 8, 24, 25, 15, 7, 1
P	P-n40-k5	5	1, 23, 32, 29, 4, 37, 36, 21, 30, 3, 1
P	P-n45-k5	1	1, 19, 14, 42, 41, 20, 43, 5, 1
P	P-n45-k5	2	1, 28, 7, 24, 8, 44, 25, 26, 15, 1
P	P-n45-k5	3	1, 3, 30, 22, 35, 31, 11, 40, 34, 1
P	P-n45-k5	4	1, 13, 18, 45, 16, 38, 6, 39, 10, 17, 12, 1
P	P-n45-k5	5	1, 33, 2, 23, 21, 36, 37, 4, 29, 32, 27, 9, 1
P	P-n50-k10	1	1, 6, 48, 37, 38, 21, 16, 1
P	P-n50-k10	2	1, 7, 34, 2, 24, 17, 1
P	P-n50-k10	3	1, 41, 33, 45, 4, 50, 1
P	P-n50-k10	4	1, 9, 14, 28, 46, 5, 1
P	P-n50-k10	5	1, 36, 20, 15, 12, 1
P	P-n50-k10	6	1, 18, 27, 8, 47, 35, 1
P	P-n50-k10	7	1, 30, 22, 23, 43, 42, 44, 1
P	P-n50-k10	8	1, 13, 10, 26, 19, 25, 1
P	P-n50-k10	9	1, 31, 49, 29, 3, 1
P	P-n50-k10	10	1, 40, 32, 11, 39, 1
P	P-n50-k7	1	1, 27, 8, 12, 39, 11, 13, 1
P	P-n50-k7	2	1, 41, 10, 26, 32, 40, 1
P	P-n50-k7	3	1, 5, 46, 31, 49, 30, 28, 35, 1
P	P-n50-k7	4	1, 7, 34, 2, 44, 42, 43, 23, 29, 1
P	P-n50-k7	5	1, 47, 9, 36, 15, 20, 14, 1
P	P-n50-k7	6	1, 3, 22, 48, 37, 38, 21, 16, 6, 1
P	P-n50-k7	7	1, 18, 4, 45, 33, 19, 25, 50, 24, 17, 1
P	P-n50-k8	1	1, 8, 32, 26, 45, 3, 31, 1
P	P-n50-k8	2	1, 13, 11, 40, 10, 41, 1
P	P-n50-k8	3	1, 4, 30, 6, 21, 16, 14, 28, 9, 1
P	P-n50-k8	4	1, 49, 48, 37, 22, 29, 23, 1
P	P-n50-k8	5	1, 46, 39, 33, 19, 25, 1
P	P-n50-k8	6	1, 12, 15, 20, 36, 47, 1
P	P-n50-k8	7	1, 27, 18, 7, 5, 38, 35, 1
P	P-n50-k8	8	1, 34, 2, 44, 43, 42, 24, 50, 17, 1
P	P-n51-k10	1	1, 28, 35, 31, 40, 11, 1
P	P-n51-k10	2	1, 24, 44, 8, 27, 9, 1
P	P-n51-k10	3	1, 33, 2, 49, 29, 23, 12, 1
P	P-n51-k10	4	1, 14, 42, 41, 20, 43, 1
P	P-n51-k10	5	1, 50, 34, 46, 16, 45, 1
P	P-n51-k10	6	1, 7, 25, 26, 15, 1
P	P-n51-k10	7	1, 3, 17, 30, 22, 51, 10, 1
P	P-n51-k10	8	1, 47, 48, 5, 19, 1
P	P-n51-k10	9	1, 32, 4, 37, 36, 21, 1

P	P-n51-k10	10	1, 13, 18, 38, 6, 39, 1
P	P-n55-k10	1	1, 34, 44, 43, 42, 24, 17, 52, 1
P	P-n55-k10	2	1, 22, 48, 37, 38, 21, 1
P	P-n55-k10	3	1, 8, 36, 9, 20, 55, 14, 16, 1
P	P-n55-k10	4	1, 11, 32, 26, 10, 40, 1
P	P-n55-k10	5	1, 4, 45, 51, 19, 25, 50, 1
P	P-n55-k10	6	1, 7, 2, 23, 29, 3, 1
P	P-n55-k10	7	1, 5, 28, 53, 47, 35, 1
P	P-n55-k10	8	1, 27, 13, 41, 33, 18, 1
P	P-n55-k10	9	1, 54, 15, 12, 39, 1
P	P-n55-k10	10	1, 46, 30, 6, 49, 31, 1
P	P-n55-k7	1	1, 13, 40, 10, 32, 26, 51, 33, 1
P	P-n55-k7	2	1, 41, 4, 45, 19, 25, 50, 24, 17, 1
P	P-n55-k7	3	1, 35, 47, 9, 53, 28, 14, 55, 20, 36, 8, 1
P	P-n55-k7	4	1, 46, 30, 6, 16, 21, 38, 37, 48, 49, 1
P	P-n55-k7	5	1, 22, 29, 23, 43, 42, 44, 2, 34, 1
P	P-n55-k7	6	1, 27, 18, 52, 7, 3, 31, 5, 1
P	P-n55-k7	7	1, 54, 15, 12, 39, 11, 1
P	P-n55-k8	1	1, 35, 53, 55, 20, 15, 36, 9, 47, 1
P	P-n55-k8	2	1, 18, 52, 4, 45, 25, 50, 17, 34, 7, 1
P	P-n55-k8	3	1, 24, 44, 42, 43, 23, 2, 1
P	P-n55-k8	4	1, 27, 13, 1
P	P-n55-k8	5	1, 33, 51, 19, 26, 32, 40, 10, 1
P	P-n55-k8	6	1, 46, 30, 6, 48, 37, 38, 21, 16, 14, 28, 1
P	P-n55-k8	7	1, 8, 54, 12, 39, 11, 41, 1
P	P-n55-k8	8	1, 5, 31, 49, 22, 29, 3, 1
P	P-n60-k10	1	1, 42, 57, 24, 25, 50, 17, 1
P	P-n60-k10	2	1, 47, 9, 8, 36, 20, 55, 53, 1
P	P-n60-k10	3	1, 34, 2, 44, 43, 23, 29, 1
P	P-n60-k10	4	1, 52, 4, 45, 33, 41, 13, 1
P	P-n60-k10	5	1, 27, 10, 54, 15, 35, 1
P	P-n60-k10	6	1, 28, 14, 58, 16, 21, 6, 30, 1
P	P-n60-k10	7	1, 7, 3, 31, 46, 5, 1
P	P-n60-k10	8	1, 18, 51, 19, 56, 26, 32, 40, 1
P	P-n60-k10	9	1, 22, 60, 37, 38, 48, 49, 1
P	P-n60-k10	10	1, 59, 11, 39, 12, 1
P	P-n60-k15	1	1, 22, 47, 32, 1
P	P-n60-k15	2	1, 41, 45, 4, 17, 1
P	P-n60-k15	3	1, 30, 6, 38, 21, 16, 1
P	P-n60-k15	4	1, 52, 50, 25, 57, 24, 1
P	P-n60-k15	5	1, 34, 37, 48, 49, 1
P	P-n60-k15	6	1, 35, 12, 39, 1
P	P-n60-k15	7	1, 59, 11, 40, 13, 1
P	P-n60-k15	8	1, 51, 19, 56, 26, 1
P	P-n60-k15	9	1, 53, 55, 14, 58, 28, 1
P	P-n60-k15	10	1, 5, 3, 7, 1
P	P-n60-k15	11	1, 46, 31, 29, 1

P	P-n60-k15	12	1, 27, 8, 36, 20, 9, 1
P	P-n60-k15	13	1, 54, 60, 15, 1
P	P-n60-k15	14	1, 18, 33, 10, 1
P	P-n60-k15	15	1, 2, 44, 42, 43, 23, 1
P	P-n65-k10	1	1, 4, 51, 19, 56, 26, 32, 10, 1
P	P-n65-k10	2	1, 18, 41, 45, 33, 40, 13, 1
P	P-n65-k10	3	1, 7, 6, 38, 61, 21, 16, 58, 1
P	P-n65-k10	4	1, 49, 48, 37, 22, 62, 29, 1
P	P-n65-k10	5	1, 5, 28, 53, 9, 47, 35, 1
P	P-n65-k10	6	1, 27, 12, 60, 15, 1
P	P-n65-k10	7	1, 31, 3, 46, 30, 14, 55, 20, 1
P	P-n65-k10	8	1, 34, 64, 24, 57, 25, 50, 17, 1
P	P-n65-k10	9	1, 63, 23, 65, 43, 42, 44, 2, 1
P	P-n65-k10	10	1, 52, 59, 11, 39, 54, 36, 8, 1
P	P-n70-k10	1	1, 53, 28, 14, 55, 20, 60, 15, 1
P	P-n70-k10	2	1, 59, 11, 39, 66, 32, 26, 40, 1
P	P-n70-k10	3	1, 46, 30, 6, 38, 61, 21, 16, 58, 1
P	P-n70-k10	4	1, 27, 57, 23, 29, 63, 3, 1
P	P-n70-k10	5	1, 34, 2, 44, 43, 65, 42, 24, 64, 1
P	P-n70-k10	6	1, 5, 68, 12, 67, 1
P	P-n70-k10	7	1, 18, 41, 45, 4, 17, 52, 7, 1
P	P-n70-k10	8	1, 31, 22, 62, 70, 37, 48, 49, 1
P	P-n70-k10	9	1, 50, 25, 19, 56, 51, 33, 10, 1
P	P-n70-k10	10	1, 69, 35, 47, 9, 36, 54, 8, 13, 1
P	P-n76-k4	1	1, 18, 41, 13, 73, 40, 10, 33, 45, 4, 50, 25, 19, 51, 26, 56, 32, 11, 59, 1
P	P-n76-k4	2	1, 35, 53, 28, 14, 55, 20, 15, 60, 12, 67, 66, 39, 54, 8, 36, 9, 47, 1
P	P-n76-k4	3	1, 27, 68, 5, 46, 30, 6, 16, 58, 38, 21, 71, 61, 72, 70, 37, 48, 22, 75, 49, 31, 69, 1
P	P-n76-k4	4	1, 76, 7, 3, 29, 62, 23, 65, 43, 44, 42, 57, 24, 2, 63, 74, 34, 64, 17, 52, 1
P	P-n76-k5	1	1, 7, 34, 74, 63, 23, 2, 44, 43, 65, 42, 57, 24, 25, 50, 64, 17, 52, 1
P	P-n76-k5	2	1, 13, 73, 40, 10, 32, 26, 56, 19, 51, 33, 45, 4, 41, 18, 76, 1
P	P-n76-k5	3	1, 27, 59, 11, 39, 66, 67, 12, 60, 15, 54, 36, 8, 1
P	P-n76-k5	4	1, 68, 49, 22, 48, 6, 16, 38, 21, 71, 61, 72, 37, 70, 62, 29, 75, 1
P	P-n76-k5	5	1, 69, 3, 31, 30, 58, 14, 55, 20, 9, 47, 35, 53, 28, 46, 5, 1
V	att-n48-k4	1	1, 24, 4, 40, 16, 38, 10, 9, 2, 23, 17, 41, 34, 15, 1
V	att-n48-k4	2	1, 39, 32, 25, 11, 45, 35, 5, 26, 42, 3, 29, 6, 48, 1
V	att-n48-k4	3	1, 14, 12, 13, 21, 47, 22, 1
V	att-n48-k4	4	1, 33, 36, 30, 43, 18, 27, 20, 37, 7, 28, 8, 19, 44, 31, 46, 1