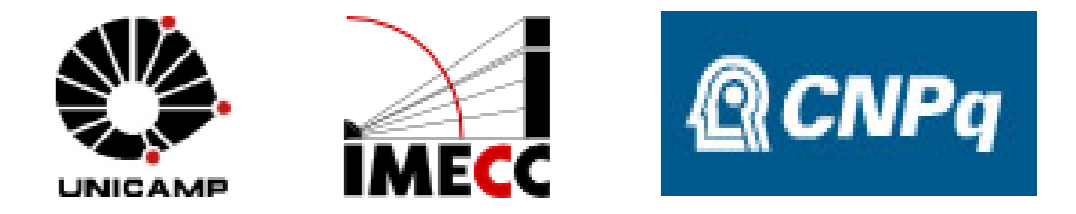


# Meta-heurísticas para roteamento de veículos com janelas de tempo



Autora: Kamila de Medeiros Galvani - kamegani@gmail.com  
Orientador: Francisco Magalhães Gomes – chico@ime.unicamp.br

IMECC/Unicamp – Instituto de Matemática, Estatística e Computação Científica / Universidade Estadual de Campinas

Este projeto solucionou problemas de roteamento de veículos com janela de tempo através de meta-heurísticas populares: Busca Tabu, Colônia de Formigas, GRASP e Busca em Vizinhança Variável. Essas meta-heurísticas, aplicadas a problemas propostos por Solomon, foram comparadas quanto à sua eficiência, assim como comparadas ao algoritmo genético desenvolvido por Vieira em VIEIRA, H. P. Meta-Heurística para a Solução de Problemas de Roteamento de Veículos com Janelas de Tempo. Dissertação de mestrado, UNICAMP, Departamento de Matemática Aplicada, 2008.

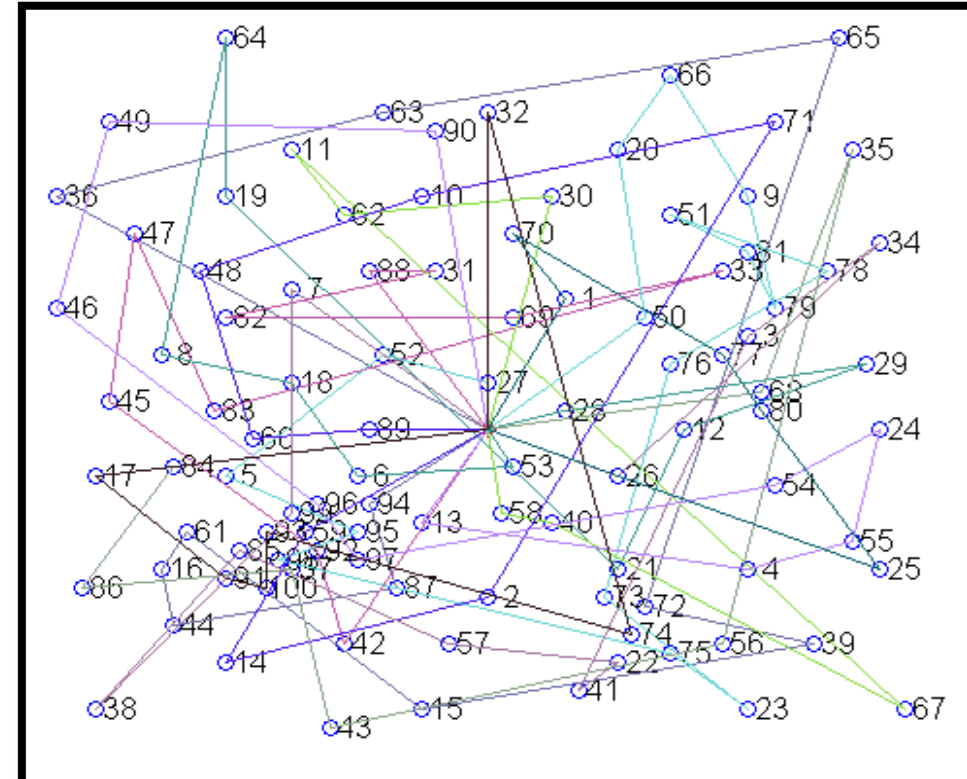
## O Problema

O problema consiste em determinar as rotas de uma frota de caminhões que deve entregar um conjunto de produtos, armazenados em um depósito, a seus consumidores, ou clientes. Cada caminhão sai do depósito e volta a ele depois de fazer suas entregas. Além disso, cada caminhão possui uma capacidade máxima de produtos.

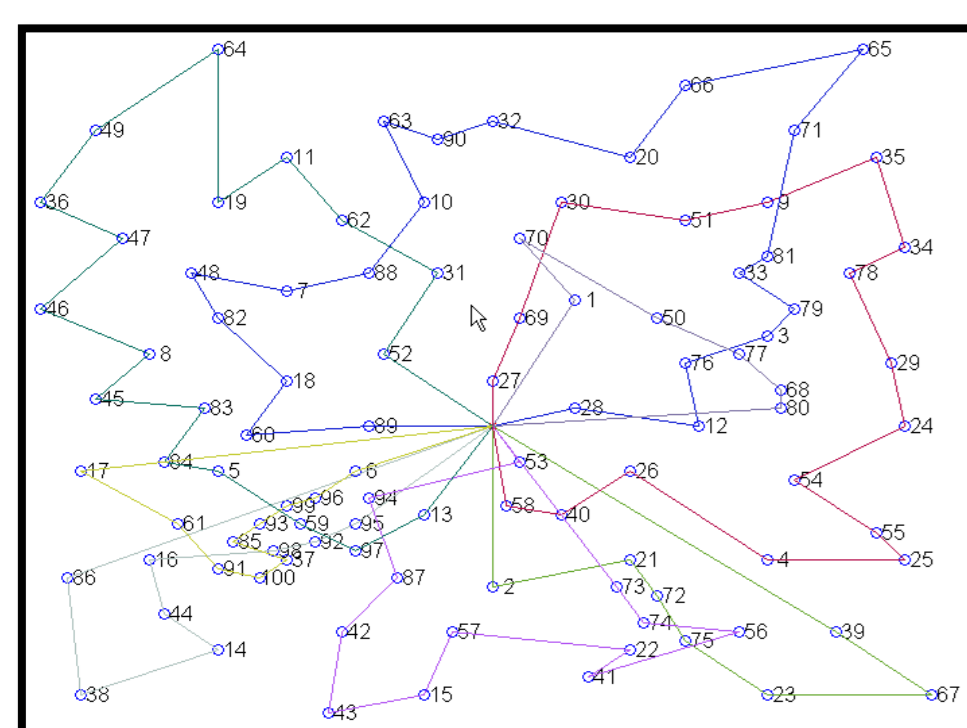
Cada cliente, além de uma cota de produtos que lhe deve ser entregue, possui um tempo de serviço e um período dentro do qual um caminhão deve começar a atendê-lo, ou seja, uma janela de tempo.

A representação do problema é feita através de um grafo  $G = (V, A)$ ; em que os vértices representam o depósito e os clientes; e as arestas, os caminhos entre os vértices. As arestas possuem, também, um custo  $c_{ij}$  associado ao caminho que estas representam, assim como um tempo  $t_{ij}$ , que os caminhões levam para percorrer tal caminho. Por simplicidade, adotou-se  $c_{ij} = t_{ij}$ , o que corresponde a atribuir o valor 1 à velocidade dos veículos. Além disso, supõe-se, sempre, que a desigualdade triangular é respeitada.

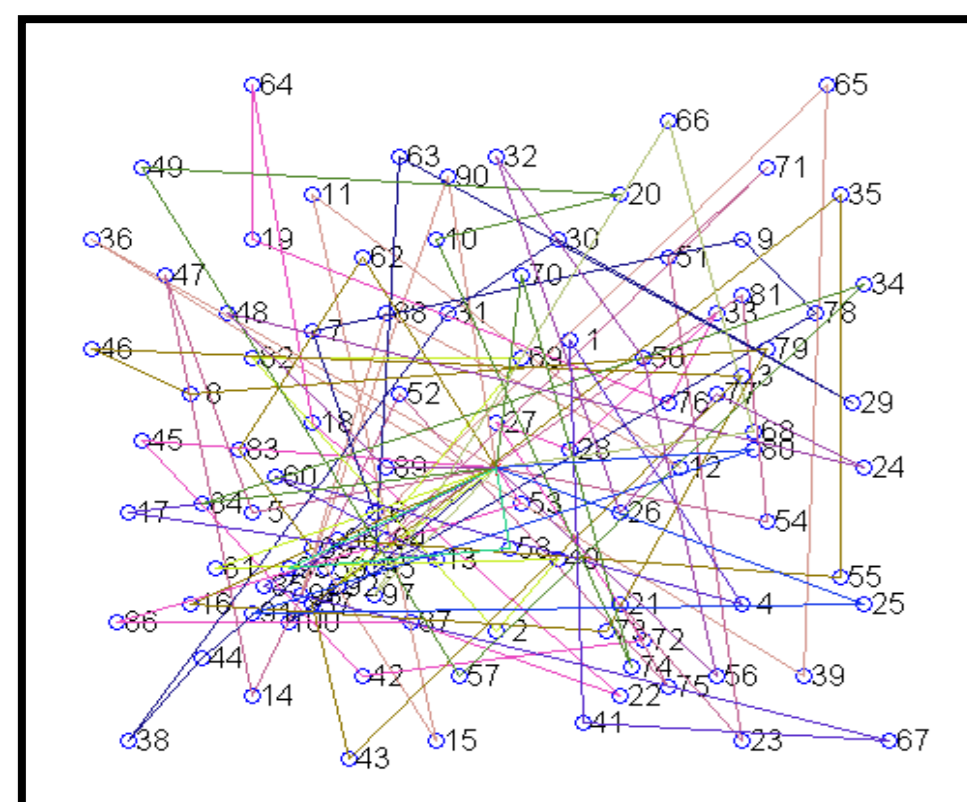
Assim, o problema exige que cada veículo percorra uma única rota, que começa e termina no depósito; que os produtos entregues não ultrapassem (em volume ou peso) a capacidade do veículo; que o início do atendimento de um cliente seja feito dentro de sua janela de tempo; e que o custo total de transporte seja minimizado.



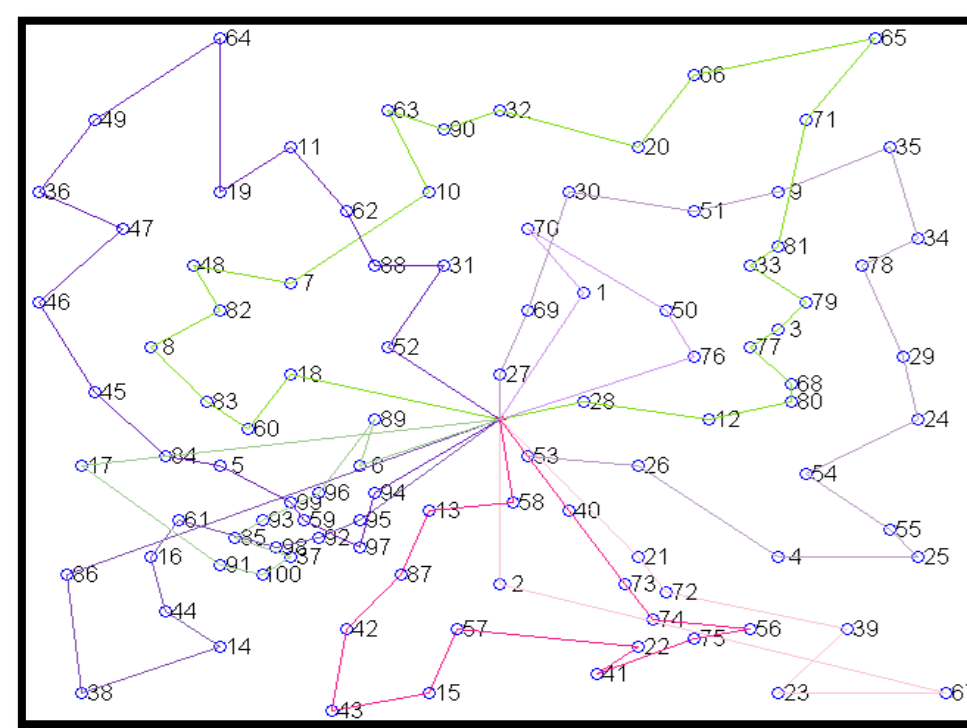
Problema R201 – Solução inicial gerada a partir de uma heurística simples



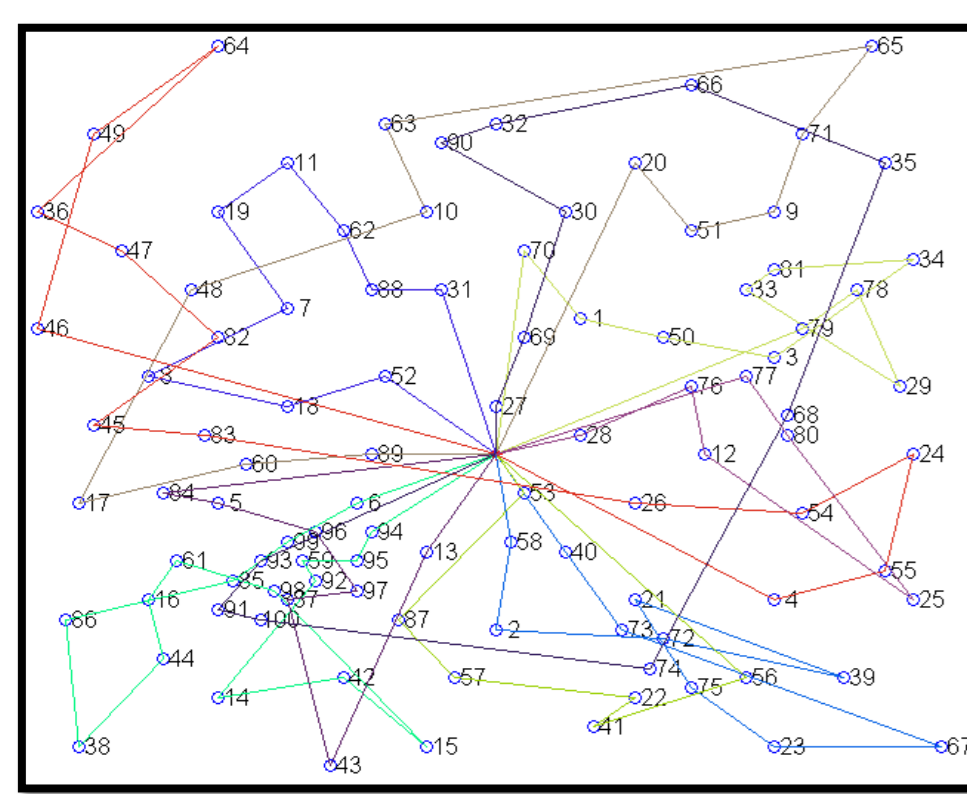
Problema R201 – Solução gerada a partir da Busca Tabu



Problema R201 – Solução gerada a partir da Colônia de Formigas



Problema R201 – Solução gerada a partir da heurística GRASP



Problema R201 – Solução gerada a partir da heurística VNS

## Busca Tabu

A busca tabu é um método de busca que procura encontrar o mínimo global de um espaço de busca, ou vizinhança, escapando de mínimos locais. Quando um mínimo local é encontrado, a busca tabu move para o melhor vizinho possível, mesmo que este seja pior que a solução atual. Para evitar que, a partir deste vizinho, o método retorne à posição anterior, a busca tabu tem uma memória dos últimos movimentos realizados, que são impedidos de se repetir.

A estrutura que contém os últimos movimentos feitos chama-se lista tabu. Essa lista também armazena a iteração em que tais movimentos voltam a ser permitidos.

## Colônia de Formigas

O algoritmo de colônia de formigas baseia-se na maneira que formigas reais buscam o menor caminho entre o formigueiro e uma fonte de comida. Tais formigas, inicialmente, andam aleatoriamente e, ao encontrar comida, voltam ao formigueiro, criando uma trilha de feromônio neste caminho. Outras formigas, ao encontrar esta trilha, tendem a segui-la até a comida, de onde voltam ao formigueiro, reforçando o depósito de feromônio.

Como, a princípio, as formigas andam aleatoriamente, várias trilhas de feromônio podem ser criadas. O feromônio, porém, evapora ao longo do tempo, de modo que as trilhas menos usadas (e não reforçadas) tendem a desaparecer. Assim, quanto mais tempo demorar para uma formiga passar pelo caminho e voltar, mais tempo o feromônio tem para evaporar. Deste modo, caminhos pequenos, por onde muitas formigas passam e voltam em um intervalo de tempo, tendem a ficar mais fortes que os longos, já que, nestes, menos formigas conseguem ir e voltar no mesmo intervalo. Quando há mais de uma trilha, as formigas possuem uma probabilidade maior de escolher o caminho com mais feromônio, o que aumenta a vantagem de caminhos menores.

## GRASP

O algoritmo GRASP (*Greedy Randomized Adaptive Search Procedure*) consiste de duas fases, uma em que são construídas soluções factíveis para o problema, e outra em que uma busca local tenta melhorar estas soluções. Para criar uma solução na primeira fase, usa-se um algoritmo aleatório e guloso. Nesta fase, cada cliente é avaliado através de uma função gulosa. A partir desta avaliação, escolhe-se um cliente para ser adicionado na solução em construção, que deve ser sempre factível. A segunda fase utiliza uma busca local para tentar encontrar um mínimo local a partir da solução gerada na primeira fase. Se o mínimo local encontrado for a melhor solução encontrada até então, esta é atualizada. As duas fases são repetidas a cada iteração.

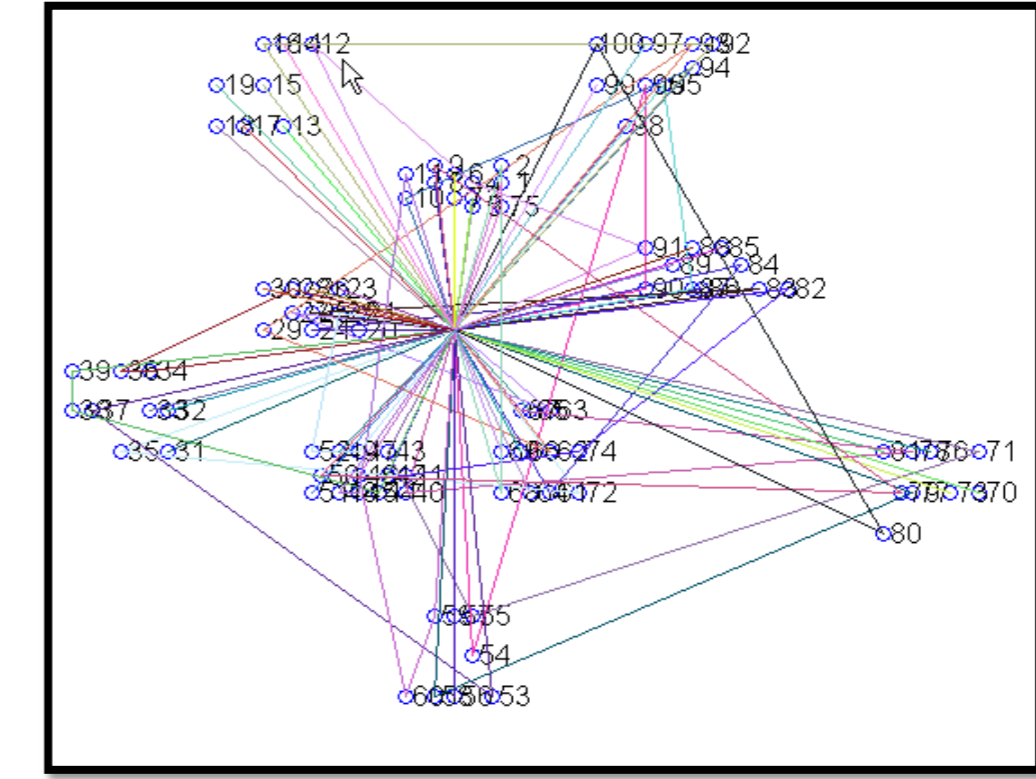
## Busca em Vizinhança Variável

O VNS (*Variable neighborhood search*), ou busca em vizinhança variável, é uma meta-heurística que busca encontrar o mínimo global de um problema combinatorial ou de otimização através de sucessivas mudanças de vizinhança, combinadas com uma busca local. Uma vizinhança de uma solução  $x$  é definida pelo conjunto de soluções possíveis de serem encontradas a partir de  $x$  utilizando-se certo procedimento ou heurística de refinamento. A vizinhança de uma solução varia de acordo com o procedimento que a gerou.

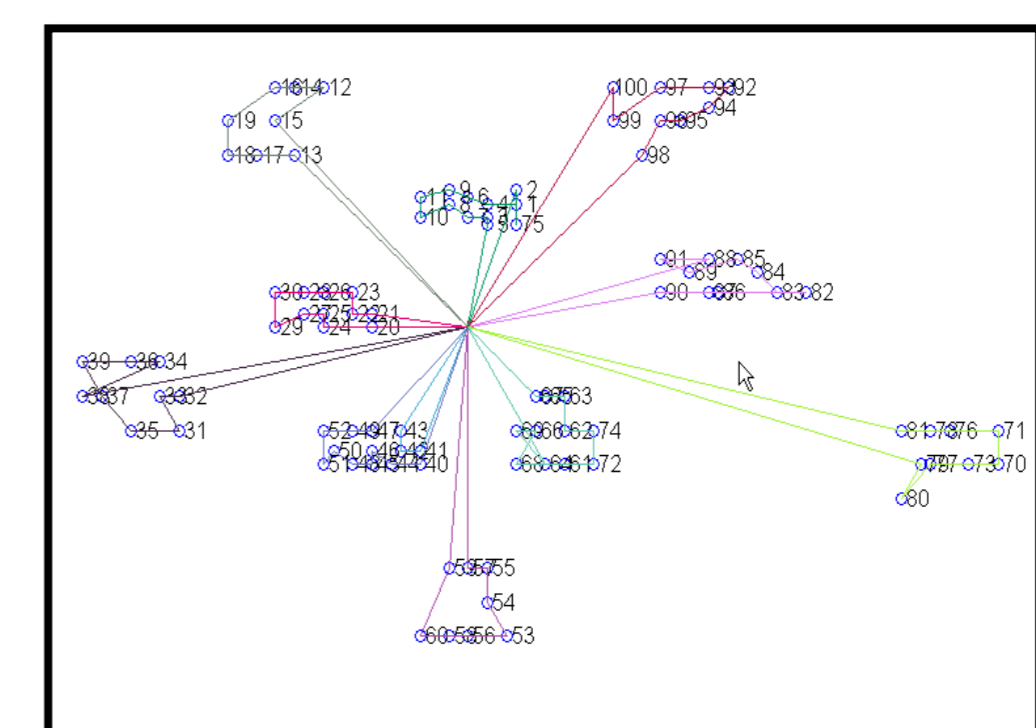
O VNS utiliza essas diferenças entre as vizinhanças tanto para procurar mínimos locais, quanto para escapar dos vales que os contêm. O método se baseia em três fatos: (a) um mínimo local de uma dada vizinhança não é, necessariamente, um mínimo local para outra vizinhança da mesma solução; (b) o mínimo global é um mínimo local de todas as vizinhanças; (c) em muitos problemas, os mínimos locais em relação a duas heurísticas diferentes estão relativamente próximos. Esta última observação é empírica, mas mostra como um mínimo local pode prover informações sobre o mínimo global; como partes de rotas que permanecem inalteradas. Frequentemente, porém, não se sabe quais partes da solução fazem parte de ambos, e deve-se continuar com a busca em torno do mínimo local (através de outra vizinhança) até que se encontre um valor menor para a solução.

## Conclusão

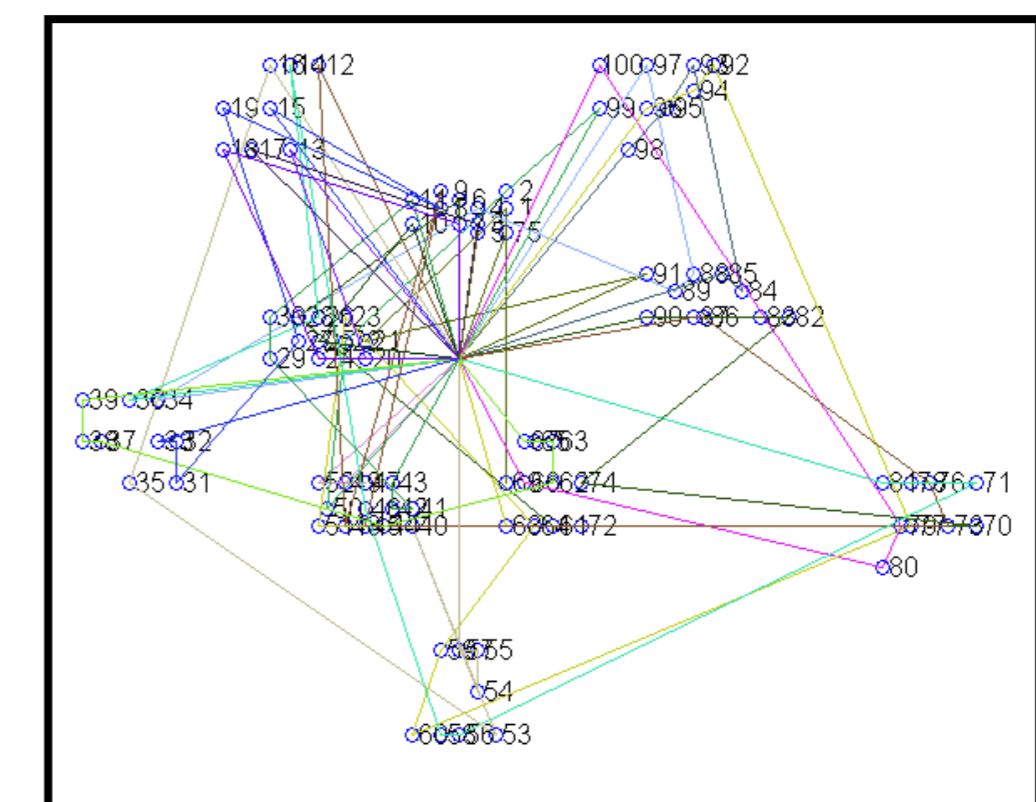
Dos algoritmos comparados, percebeu-se que, para o problema do tipo C1, o VNS utilizando apenas duas heurísticas apresentou melhores resultados, enquanto o algoritmo genético foi o melhor para o problema C2. Já para os problemas dos tipos R e RC, o GRASP encontrou as melhores soluções, embora o algoritmo genético tenha encontrado um valor melhor para o R201. Em todos os casos, não apenas as distâncias totais das rotas geradas por estas heurísticas foram menores, como estas também utilizaram menor número de caminhões para as tarefas. Além disso, com exceção da Colônia de Formigas, todos os métodos apresentaram melhores significativas em relação à solução inicial, gerada pela PFIH. Curiosamente, as soluções apresentadas pelo VNS foram melhores quando este utilizava apenas duas heurísticas para gerar vizinhanças, e não três, que foi o máximo estudado.



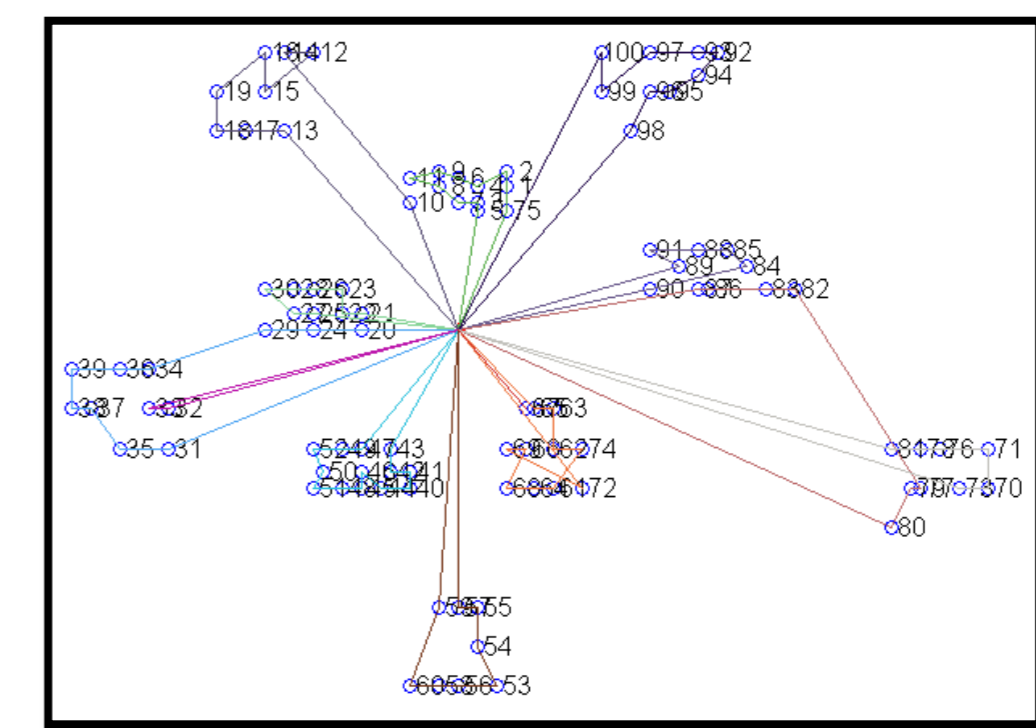
Problema C101 – Solução inicial gerada a partir de uma heurística simples



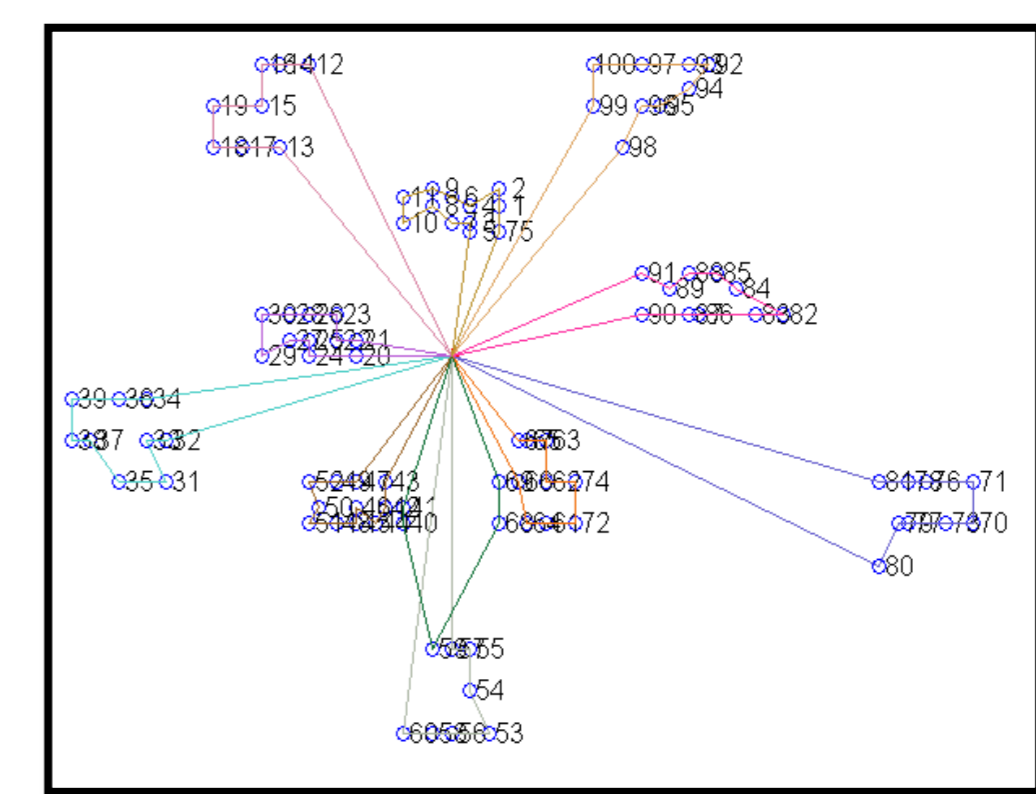
Problema C101 – Solução gerada a partir da Busca Tabu



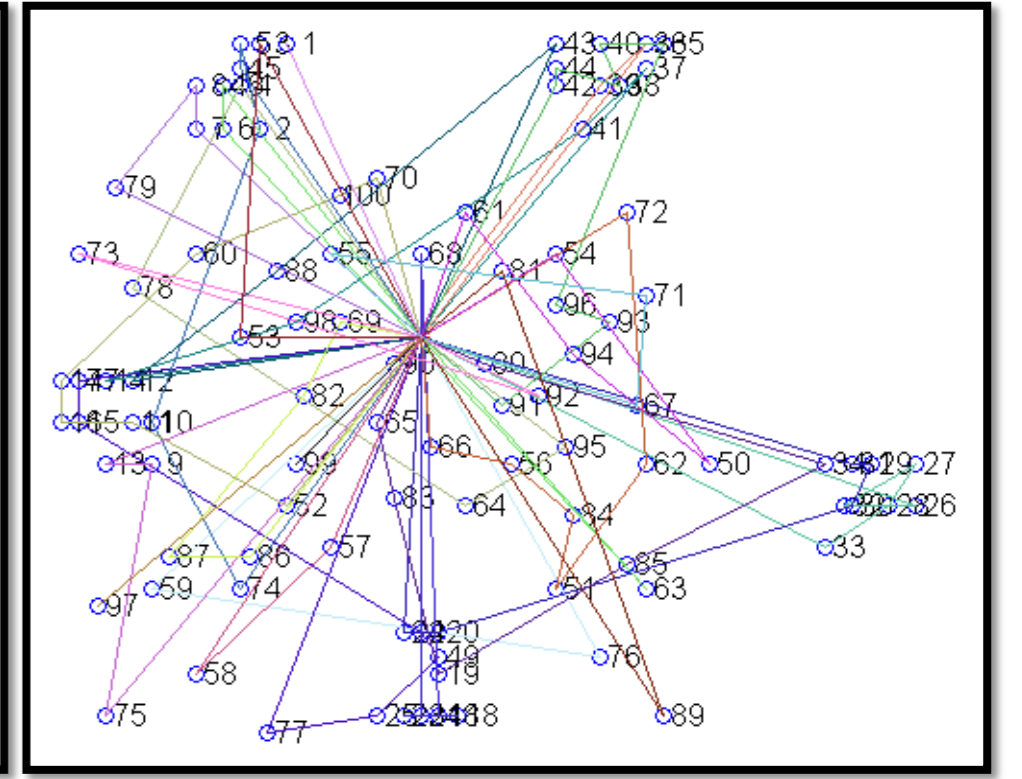
Problema C101 – Solução gerada a partir da Colônia de Formigas



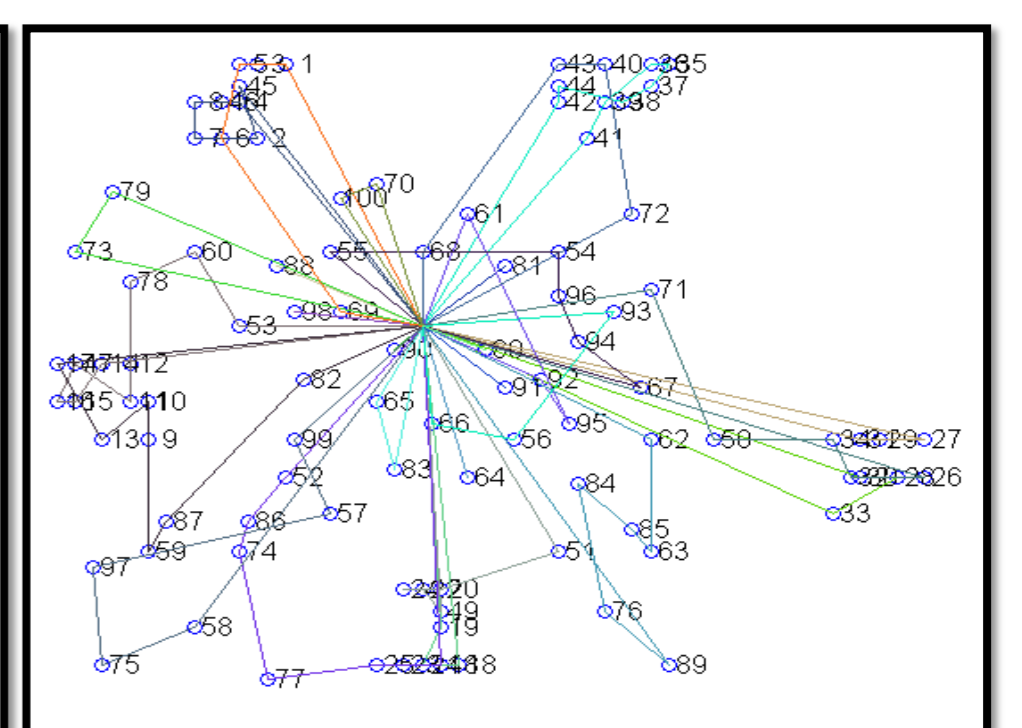
Problema C101 – Solução gerada a partir da heurística GRASP



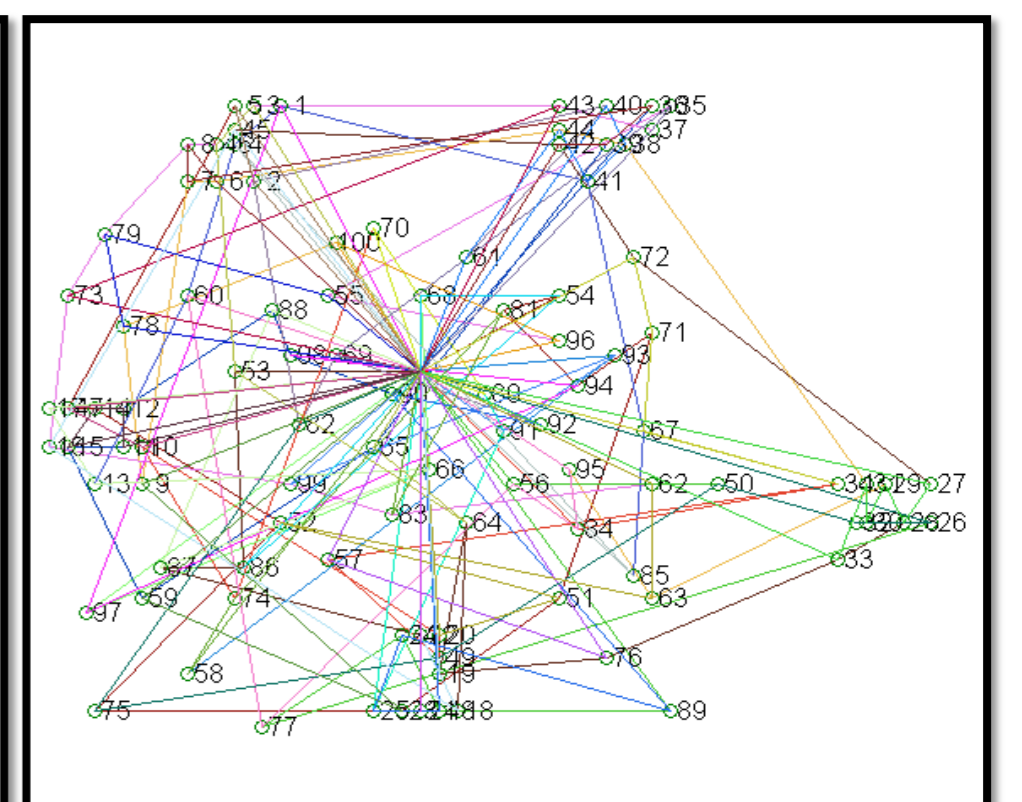
Problema C101 – Solução gerada a partir da heurística VNS



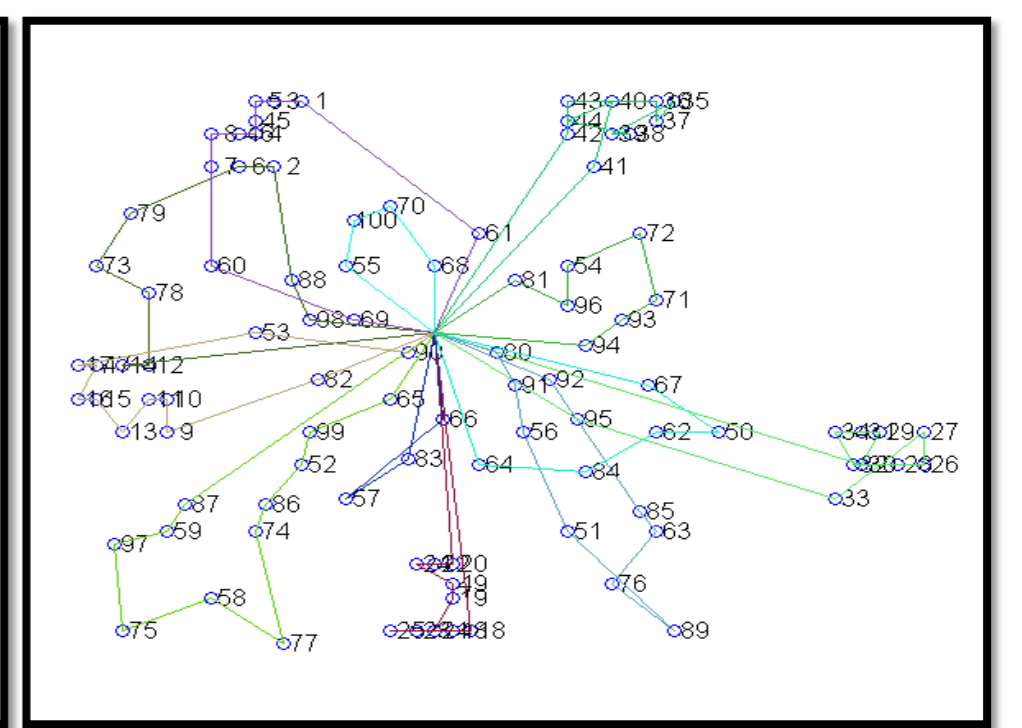
Problema RC11 – Solução inicial gerada a partir de uma heurística simples



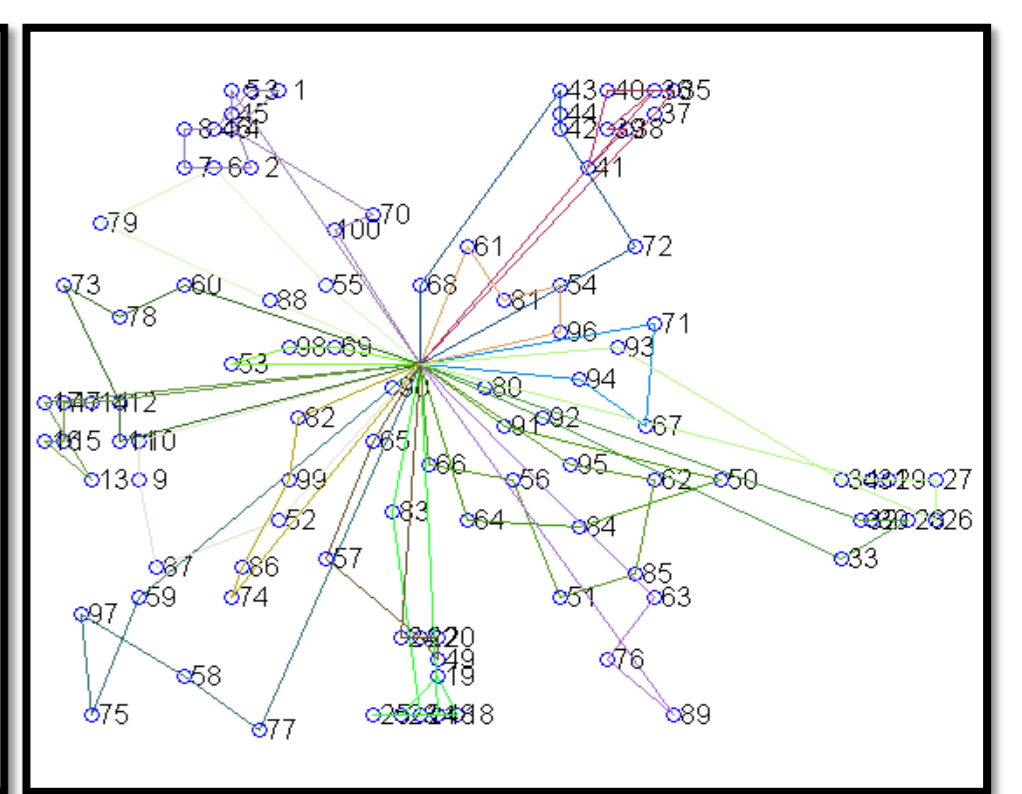
Problema RC11 – Solução gerada a partir da Busca Tabu



Problema RC11 – Solução gerada a partir da Colônia de Formigas



Problema RC11 – Solução gerada a partir da heurística GRASP



Problema RC11 – Solução gerada a partir da heurística VNS

| Problema            | C101      |    | R101      |    | RC101     |    |
|---------------------|-----------|----|-----------|----|-----------|----|
|                     | Distância | NC | Distância | NC | Distância | NC |
| PFIH                | 4845,6    | 63 | 3250,4    | 30 | 3765,1    | 51 |
| Busca Tabu          | 991,0     | 13 | 884,6     | 8  | 1216,5    | 14 |
| Colônia de Formigas | 2779,6    | 19 | 2661,1    | 19 | 3289,5    | 35 |
| VNS – 2 heurísticas | 892,7     | 11 | 650,9     | 5  | 1744,5    | 24 |
| VNS – 3 heurísticas | 1035,3    | 13 | 903,6     | 7  | 1786,7    | 24 |
| GRASP               | 1045,5    | 13 | 1094,6    | 10 | 1060,7    | 12 |
| Algoritmo Genético  |           |    | 1642,88   | 20 | 1650,6    | 16 |

| Problema            | C201      |    | R201      |    | RC201     |    |
|---------------------|-----------|----|-----------|----|-----------|----|
|                     | Distância | NC | Distância | NC | Distância | NC |
| PFIH                | 2239,5    | 11 | 4013,7    | 39 | 2650,4    | 12 |
| Busca Tabu          | 969,2     | 8  | 2082,2    | 27 | 1165,0    | 8  |
| Colônia de Formigas | 2239,5    | 11 | 3454,4    | 27 | 2650,4    | 12 |
| VNS – 2 heurísticas | 1245,2    | 8  | 1821,9    | 19 | 1349,7    | 10 |
| VNS – 3 heurísticas | 1383,9    | 10 | 1940,5    | 21 | 1462,9    | 11 |
| GRASP               | 962,9     | 8  | 1224,5    | 12 | 920,5     | 6  |
| Algoritmo Genético  | 591,56    | 3  | 1153,65   | 9  | 1269,01   | 9  |

Resultados obtidos e comparação com o Algoritmo Genético desenvolvido por Vieira.