1
Introdução

1.1
Motivação

Diversas situações encontradas no dia-a-dia são de interesse da indústria e da academia, por representarem um contexto em que se encontra um *Problema de Otimização*. Áreas como pesquisa operacional, projeto de sistemas mecânicos, processamento de imagem e eletrônica levam constantemente seus profissionais a se confrontarem com um problema desse tipo [16].

Um tal problema envolve uma ou mais variáveis e pode possuir várias soluções possíveis, cada qual com um determinado valor, desejando-se obter uma cujo valor seja ótimo, isto é, mínimo ou máximo [13].

Quando um Problema de Otimização envolve variáveis discretas, diz-se que corresponde a um *Problema de Otimização Combinatória*. Mais especificamente, um problema desse tipo é definido sobre um conjunto discreto de elementos e uma solução corresponde a uma combinação de todos ou de parte desses. Tais problemas envolvem conjuntos, arranjos, permutações, grafos e outros, ou seja, entidades discretas por natureza [39].

A crescente importância obtida por Problemas de Otimização Combinatória deu-se em meados do século XX, a partir da mudança de ênfase sobre questões como "Será que um arranjo com uma determinada característica existe?" ou "Quantos arranjos deste ou daquele tipo existem?", para questões da forma "Qual o *melhor* arranjo possível com uma característica específica?" [37]. Assim sendo, aspectos como a existência ou o número de tais estruturas não estão em foco nesse tipo de questionamento, mas sim a obtenção de uma estrutura cujo valor seja ótimo dentre o número de possibilidades que se têm, além da inerente preocupação com o tempo tomado nesse processo.

Essa mudança de perspectiva se deveu, essencialmente, ao advento do computador digital moderno, haja vista que possibilitou e estimulou o desenvolvimento de métodos de resolução envolvendo grande esforço de computação em sua realização, os quais antes se mostravam inexequíveis [37]. Evidentemente, a própria existência do computador deu origem a diversos problemas de natureza combinatória, através da aplicação a um vasto espectro de problemas numéricos e não-numéricos.

Dentre os campos da Otimização Combinatória, destaca-se o da *Programação Inteira* (*Integer Programming* - IP), que corresponde a um Programa Linear com a restrição adicional de que todas as suas variáveis apenas assumam valores inteiros. Área esta que é bastante fértil, na medida em que diversos problemas da maior relevância prática podem ser representados dessa forma [9]. Ademais, um problema de *Programação Inteira Mista* (*Mixed-Integer Programming* - MIP) possui, além de variáveis discretas, outras que podem assumir valores contínuos.

O problema geral de Programação Inteira está contido na classe $\mathcal{NP}$-*Difícil*, ou seja, a linguagem representativa da sua versão de decisão pertence à classe $\mathcal{NP}$-*Completo* [48]. Tem-se, assim, por generalização, que o problema de Programação Inteira Mista também está nessa classe. Existem, pois, fortes evidências quanto à sua intratabilidade intrínseca, isto é, quanto à impossibilidade de se obter um algoritmo polinomial, definido como o critério de "bom" por Edmonds [17].

Tipicamente, tais problemas de interesse possuem a característica de que, para instâncias de pequeno porte, algoritmos meramente enumerativos se portam relativamente bem. Entretanto, há um limiar a partir do qual o simples acréscimo de uma unidade ao tamanho do problema ocasiona grande variação no tempo de computação, denominada de *Explosão Exponencial* (*Exponential Blow-Up*) [9]. Faz-se, portanto, imperioso estudar os aspectos envolvidos e as formas de se lidar com tal limitação.

Nesse contexto, encaixam-se duas correntes: *Métodos Exatos* e os *Aproximados*. Os primeiros visam à obtenção da solução ótima, mas sem fazer qualquer garantia quanto ao tempo de computação exigido. Por outro lado, os do segundo tipo priorizam realizar tempos de computação factíveis, em detrimento da qualidade da solução fornecida, isto é, contentando-se com uma solução cujo valor seja relativamente de "boa" qualidade.

Na classe de algoritmos exatos, a abordagem central reside na de enumeração e em *Branch-and-Bound*, que pode ser definida como uma tentativa de construção de uma prova de que uma solução é ótima, baseada em sucessivos particionamentos do espaço de solução [39]. Outras técnicas de destaque nesta classe, tais como *Branch-and-Cut*, *Branch-and-Price* e *Branch-and-Cut-and-Price* correspondem à introdução de um ou mais algoritmos na estrutura daquele.

Uma das formas de se tratar um problema de maneira exata implica em representá-lo na forma de um MIP. Assim, ferramentas utilizadas para abordálo dessa maneira são denominadas de *Resolvedores MIP*, contendo algoritmos eficientes para abordar um problema de forma genérica e automatizada. Entretanto, mesmo com a rápida evolução dessas ferramentas nos últimos anos, algoritmos aproximados ainda são fundamentais para resolver instâncias grandes ou especialmente intrincadas de problemas $\mathcal{NP}$–Difíceis [1].

Na classe de métodos aproximados, estão os *Algoritmos Aproximativos* e as *Heurísticas*. Um algoritmo aproximativo fornece uma garantia de distância da solução fornecida em relação ao ótimo. Formalmente, é definido como necessariamente polinomial e avaliado pelo pior erro relativo possível em relação a todas as instâncias do problema [28].

Por sua vez, uma heurística corresponde a um algoritmo norteado por argumentos plausíveis, em vez de uma prova formal matemática [37]. Essa procura boas soluções a um custo computacional razoável, sem ser capaz de garantir tanto viabilidade quanto otimalidade, ou até, em muitos casos, afirmar o quão perto do ótimo uma dada solução viável está [45]. Na prática, heurísticas acabam não fornecendo qualquer tipo de informação quanto à proximidade do ótimo.

De fato, o uso de técnicas heurísticas se popularizou a partir da década de 1980. Isso possuiu duas principais motivações: por um lado, o desenvolvimento da teoria de *Complexidade Computacional* proveu uma base racional para a exploração de heurísticas, em vez da contínua perseguição ao Santo Graal da otimalidade; por outro, ao mesmo tempo, houve um aumento significativo no poder e na eficiência das abordagens mais modernas [45].

Entretanto, muitas heurísticas eram especificamente idealizadas para os problemas sendo abordados, havendo grande dificuldade de serem aplicadas a outros. Havia, portanto, enorme interesse no estudo e desenvolvimento de técnicas mais gerais, capazes de serem aplicadas a uma gama maior de problemas.

Assim, surgiu o conceito de *Metaeurística*, cunhado pela primeira vez por Glover [25]. Essa combina técnicas heurísticas básicas em um *framework* de alto nível, com o objetivo de explorar o espaço de buscas de maneira eficiente e efetiva [1]. Destacaram-se por obter bons resultados, tendo como trabalhos pioneiros os de Kirkpatrick [32] sobre *Simulated Annealing* e Holland [29] sobre *Algoritmos Genéticos*.

Recentemente, contudo, alguns trabalhos propuseram alternativas que mesclam as duas correntes anteriormente citadas, aproveitando-se tanto do poderio computacional fornecido por resolvedores MIP, quanto das estratégias de busca trazidas por metaeurísticas. Os resultados obtidos mostraram-se promissores e abriram uma nova frente de pesquisa.

1.2 Objetivos do Trabalho

O objetivo deste trabalho consiste em estudar e experimentar uma abordagem que integre as ideias fornecidas por metaeurísticas, juntamente com resolvedores MIP, para o Problema de Roteamento de Veículos com Restrição de Capacidade (*Capacitated Vehicle Routing Problem* - CVRP).

Mais especificamente, deseja-se examinar a aplicação das vizinhanças de bola e elipsoidal, definidas originalmente por Fischetti e Lodi [19] e Pigatti *et al.* [40], sobre a implementação do algoritmo de *Branch-and-Cut-and-Price Robusto* para o CVRP de Fukasawa *et al.* [22]. Por fim, deseja-se testar a implementação obtida sobre instâncias consagradas na literatura e analisar esses resultados.

1.3 Organização do Trabalho

No capítulo 2, apresenta-se o CVRP, em que é abordado seu aspecto histórico, justifica-se sua relevância e faz-se uma revisão dos trabalhos constantes na literatura. As formulações presentes na implementação são apresentadas e descritas.

O capítulo 3 é dedicado à descrição dos tipos de abordagens que combinam metaeurísticas e resolvedores MIP. Nesse, apresenta-se também a definição das vizinhanças de bola e elipsoidal.

O capítulo 4, por seu turno, discorre sobre a abordagem proposta, descrevendo os experimentos computacionais realizados, além de apresentar os resultados alcançados e fazer uma análise dos dados gerados. Finalmente, no capítulo 5, são expostas as considerações finais e a conclusão deste trabalho.