P vs NP vs NP-Completos

Diogo Araujo Miranda[1]

[1]ICEI - Pontifícia Universidade Católica de Minas Gerais (PUC-MG)

1. Contextualização

Durante o estudo de algoritmos, é realizado a análise de complexidade do mesmo, em que em vários casos são descritos como algoritmos de tempo polinomial (Polynominal Algorithm) em que, sobre entradas de tamanho n, seu tempo de execução do pior caso é $O(n^k)$ para qualquer constante k. Porém nem todos os problemas podem ser resolvidos de forma polinomial. Alguns problemas não podem ser resolvidos por um computador, não importa quanto tempo seja fornecido. Outros algoritmos podem até ser resolvidos, mas não no tempo $O(n^k)$ para qualquer constante k. No geral, dizemos que algoritmos que podem ser resolvidos em tempo polinomial são algoritmos tratáveis, e problemas que exigem um tempo superpolinomial são intratáveis ou difíceis.
Os problemas NP-completos são problemas que encaixam nesse conceito de superpolinomial, em que seu status é desconhecido e não foi encontrado nem provado nenhum algoritmo em tempo polinomial para resolver esse problema. O mais interessante é que problemas NPC se parecem muito com problema polinominal, como por exemplo o problema de se encontrar o caminho mais curto e mais longo: é possível encontrar com pesos de arestas negativas o caminho mais curto a partir de uma única origem em um grafo orientado em um determinado tempo. Porém, encontrar o caminho simples mais longo entre dois vértices é NP-completo. Problemas NPC são tão difíceis de ser resolvidos que prêmios foram atribuídos á sua ”resolução”, para quem for capaz de provar um algoritmo eficiente.

2. Definição

Durante o estudo dessas classes, podemos descrever a classe P como problemas resolvidos no tempo $O(n^k)$ para alguma constante k, onde n é o tamanho da entrada para o problema.
A classe NP consiste em problemas verificáveis em tempo polinomial. Ou seja, se existir um certificado para uma solução, pode-se verificar se o certificado é correto em tempo polinomial no tamanho da entrada do problema. Todo problema P está em NP, sabendo que podemos resolver todo P em tempo polinomial, ou seja $P \subseteq NP$.
Já um problema NP-completo, está na classe NPC se ele está em NP e é tão difícil quanto qualquer problema em NP. Então se qualquer problema que está na classe NP-completo pode ser resolvido em tempo polinomial, então qualquer problema NPC tem um algoritmo em tempo polinomial. Em contrapartida muitos teóricos da computação dizem que esses problemas NPC são irresolvíveis e intratáveis, dado a quantidade de estudos que foram realizadas e n ão levaram a quase nada.

3. Problemas de otimização

Na computação surgem muitos problemas de otimização, existindo uma solução válida com um valor associado, e desejamos encontrar o melhor valor possível. Ao realizar a análise de complexidade de um algoritmo, podemos perceber que o seu problema de otimização pode ser de fácil resolução, ou não mais difícil. Porém quando esse algoritmo é provado verdadeiramente difícil, irá implicar diretamente na sua otimização, dada a teoria de problemas NPC.

4. Problemas de tempo polinomial

Frequentemente na computação problemas de deparam com tempos polinomiais, sendo esse tratáveis e razoáveis do ponto de vista matemático e filosófico. Porém, embora um problema que exige um tempo $\Theta(n^{100})$ pode ser considerado intratável, pois gera um problema de alto grau, embora que na prática não são encontrados muitos problemas polinomiais desse alto grau. Levando esse pensamento em conta deve-se entender que, um problema polinomial muitas vezes é melhorado com o passar do tempo. Não necessariamente um problema que exige $\Theta(n^{100})$ irá exigir esse tempo, pois existem diferentes conjuntos e diferentes modos de execução desse conjunto. Além disso, algoritmos e codificações melhores surgem com o tempo. Execuções de forma paralela em computadores paralelos podem resolver problemas desse tipo, quando o número de processadores cresce polinomialmente com o tamanho da entrada. Por fim, os algoritmos polinomiais tem propriedades de fechamento, fechados sob adição, multiplicação divisão e composição.