Teoria da Intratabilidade

Universidade Federal do Pará
Instituto de Tecnologia
Faculdade de Engenharia da Computação e Telecomunicações
Teoria da Computação II
Prof^o *Jamir*
Danilo Henrique Costa Souza - 201006840008

Belém, 09 de Dezembro de 2014

Resumo

A teoria da intratabilidade trata da classe de problemas que não se consegue encontrar a solução usando um algoritmo de tempo polinomial e o maior desafio dentro desse ramo da computação é sem dúvida o problema P *versus* NP. Este artigo tem como objetivo explicar de forma simplificada este problema, o mais importante da Teoria da Intratabilidade.

Palavras-chaves: P *versus* NP, teoria da intratabilidade, NP-completo, NP-difícil.

Sumário

Lista de ilustrações

1 Introdução

Todas ou quase todas as pessoas que estudaram em algum momento teoria da complexidade já ouviram falar da Teoria da intratabilidade, mais especificamente do problema P *versus* NP, só que normalmente este assunto não é apresentado de forma simplificada, exatamente por não ser um tema fácil de apresentar.

Este trabalho tem por objetivo apresentar o problema P *versus* NP da forma mais simplificada possível sem perder as principais definições que regem o problema, o foco será apresentar a temática na sua forma mais computacional por meio definições e conclusões lógicas do que por meio de definições matemáticas formais.

O problema foi primeiramente mencionado em uma carta de Kurt Godel para John von Neumann perguntando se um dado problema NP-completo poderia ser resolvido em tempo linear ou quadrático, porém foi Stephen Cook em 1971 que definiu formalmente o problema $P = NP$ em seu artigo "*The complexity of theorem proving procedures*"

2 Algumas Definições

Antes de explicar o problema, veremos primeiramente algumas definições que nos ajudarão a entender de forma mais objetiva e concisa a problemática.

- MT: Máquina de Turing Determinística
- MTND: Máquina de Turing não Determinística
- Tempo de execução (HARDESTY, 2009)
  - Polinomial: Algoritmos que possuem tempo de execução proporcional a N operações.
  - Exponencial: Algoritmos que possuem tempo de execução proporcional a 2^N operações.
- A classe P: Problemas que podem ser resolvidos em tempo polinomial usando uma MTD

- A classe NP: Classe de problemas de podem ter sua solução verificada em tempo polinomial usando uma MTD. É possível dizer, dado uma resposta a priori, se esta resposta está correta, porém se ela não for a resposta correcta é impossível comprovar isso. É também a classe de problemas que podem ser resolvidos em tempo polinomial usando uma MTND.
  - NP-completo: É a classe de problema a qual todo e qualquer problema NP pode ser reduzido a ela em tempo polinomial e sua solução também pode ser verificada em tempo polinomial, em outras palavras, qualquer problema NP que posso ser reduzido em tempo polinomial para um problema NP-completo pode ter sua solução verificada em tempo polinomial.
  - NP-difícil: É a classe de problemas que não precisam ter sua solução verificada necessariamente em tempo polinomial.
  - NP-intermediário: São problemas que não estão nem em P nem em NP-completo. Esta classe só existe se $P \neq NP$ confrome demonstrado por Ladner em (LADNER, 1975)

3 Teoria da Intratabilidade

Podemos definir a Teoria da Intratabilidade como o conjunto de técnicas utilizadas para provar que determinados problemas não podem ser resolvidos usando algoritmos de tempo polinomial, ou seja, são intratáveis por MTD.

Existe um número muito grande de problemas do tipo NP, mas para provar que um dado problema não pode ser resolvido em tempo polinomial não é necessariamente necessário provar este problema pertence a classe NP, pode-se apenas provar que o problema A (origem), já conhecido pertencer a classe NP, é redutível ao problema B (destino). Este procedimento é chamado de redução, entretanto para este caso específico a noção clássica de redução deve ser modificada para "*redução de tempo polinomial*", ou seja, a redução do problema de origem (A) (conhecido ser NP-completo) ao de destino (B) (que se deseja provar ser NP-completo) deve ser realizável em tempo polinomial, dizemos assim que A é redutível a B, caso A não seja redutível a B ficará impossível concluir a intratabilidade de A. A Figura 1 mostra problemas NP-completo comumente utilziados em reduções (GIAN LUCA RUGGERO).

Como dito anteriormente, problemas NP não possuem um algoritmo eficiente capaz de resolvê-los em tempo polinomial, como os problemas do Caixeiro viajante, da fatoração de um número inteiro entre vários outros. Esses problemas, porém podem ter uma dada solução facilmente verificada, por exemplo, tomemos o número 55544445, é impossível saber quais números foram multiplicados para gerar este número e se eles são inteiros, mas se tiver-

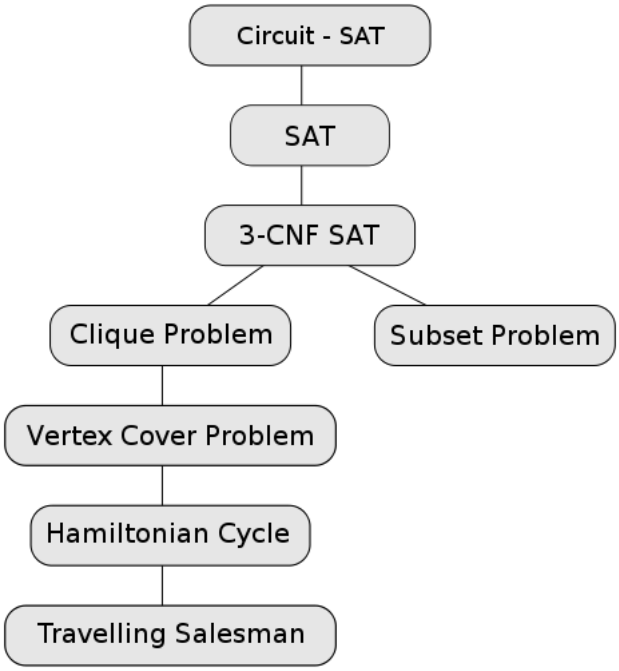

Figura 1 – Problemas NP-completos usados em reduções

mos a informação de que os números utilizados foram 5555 e 9999 podemos facilmente verificar que o resultado é 55544445.

4 O problema P *versus* NP e sua importância

Para melhor compreensão da importância deste problema para o mundo da computação vamos primeiro dar um definição simplificada do mesmo.

> O problema P *vs.* NP consiste em descobrir se um problema cujo a solução é facilmente verificada por um computador pode também ser facilmente resolvido por um computador (MIESSLER, 2014).

Provar que $P = NP$ ou que $P \neq NP$ implicaria em grandes mudanças no mundo da computação. Caso o primeiro seja provado, toda a variedade problemas considerados intratáveis por um computador passariam a ser triviais e poderiam ser resolvidos em tempo polinomial, o que representaria o fim da criptografia RSA, por exemplo, uma vez que um número poderia ser facilmente fatorado até encontrar os inteiros que foram multiplicados para gerá-lo, como veremos em um exemplo na seção 4.

Caso o segundo seja provado, o que acreditam a maioria dos especialistas, será impossível obter um algoritmo eficiente para para encontrar a solução

de um problema NP em tempo polinomial, isso garantiria que a criptografia RSA nunca seria quebrada pelos computadores utilizados na atualidade.

Como dito anteriormente o problema P *versus* NP levanta a hipótese se a classe NP de problemas é igual ou não a classe P e a solução deste problema teria implicações muito significativas no mundo de hoje, entretanto realizar essa prova não é uma tarefa fácil, pois este é sem dúvida um dos problemas que mais desafia os pesquisadores da área de complexidade computacional desde que foi formalmente definido. Por ser um problema extremamente complexo suas tentativas de provas devem ser exaustivamente analisadas pela comunidade internacional.

5 Conclusão

Após esta análise do problema P *versus* NP é possível ter uma melhor compreensão do problema em si e de suas consequências. A Figura 2 mostra o diagrama de Euler para ambos os casos ($P = NP$ ou $P \neq NP$).

Segundo (MIESSLER, 2014) em uma pesquisa realizada em 2002, 61 matemáticos e cientistas da computação disseram acreditar que $P \neq NP$, enquanto que apenas 9 disseram acreditar que $P = NP$, sendo que a maioria desses 9 disseram isso apenas para contrariar os outros. Mas o fato é que a resposta para esse problema permanece um mistério até os dias de hoje e não há nenhum sinal de que será resolvido num futuro próximo.

Seja qual for o resultado, qualquer solução encontrada seja $P = NP$ ou $P \neq NP$ certamente irá revolucionar a computação como a vemos hoje em dia, uma vez que problemas considerados complexos e intratáveis atualmente ou serão considerados triviais, gerando então um avanço tecnológico exponencial ou serão considerado de fato intratáveis dando a possibilidade de se eliminar determinados caminhos e de tratar problemas com uma maior especificidade.

Referências

HARDESTY, L. *Explained: P vs. NP*. 2009. `http://newsoffice.mit.edu/2009/explainer-pnp`. Citado na página 2.

LADNER, R. E. On the structure of polynomial time reducibility. *Journal of the ACM (JACM)*, ACM, v. 22, n. 1, p. 155–171, 1975. Citado na página 3.

MIESSLER, D. *P vs. NP Explained*. 2014. `http://danielmiessler.com/study/pvsnp/`. Citado 2 vezes nas páginas 4 e 5.

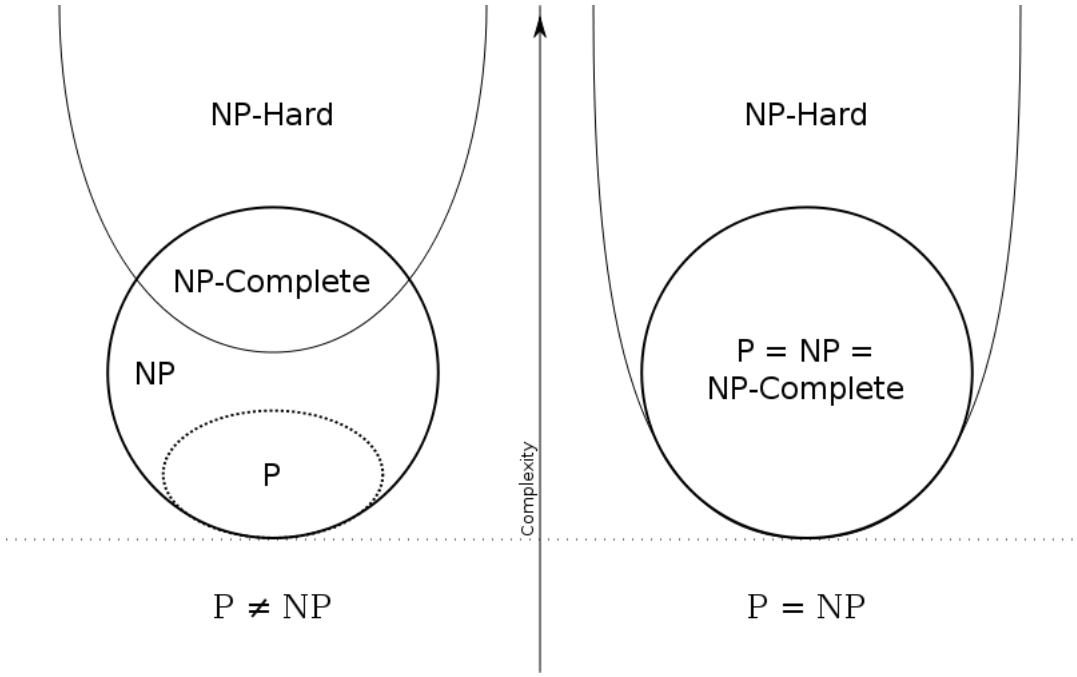

Figura 2 – Diagrama de Euler de P *versus* NP