Prova formal do problema da satisfatibilidade booleana (SAT) por Cook-Levin

Allan Cordeiro Rocha de Araújo.

[1]Centro de ciência e tecnologia– Universidade Federal de Roraima (UFRR)
Boa Vista– RR – Brasil

allanps32008@gmail.com

Resumo. *Será mostrado como o Cook-Levin fez a prova do teorema da satisfatibilidade booleana.*

1. Introdução

O objetivo neste artigo é, de forma clara, provar que o SAT é um problema NP-Completo através do que foi mostrado por Cook-Levin.

2. Cook-Levin

Stephen Arthur Cook foi um renomeado cientista da área de computação no Canada, ele, junto de Leonid Levin, foi o responsável por provar o primeiro teorema NP-Completo, o problema da satisfatibilidade booleana (SAT), por isso o teorema se chama Cook-Levin. Além disso ele formalizou o conceito de NP-Completo.

3. Problema da satisfatibilidade booleana (SAT)

Na teoria da complexidade computacional, o SAT foi o primeiro problema NP-Completo provado ser NP-Completo, tanto que para se provar qualquer outro problema NP-Completo usa-se o SAT (reduzindo-se tal problema a ele). O problema do SAT é encontrar se existe uma solução verdade para determinada fórmula booleana. Exemplo:

Tendo-se as variáveis x1, x2, x3, x4

$$(x_1 \lor \neg x_3 \lor x_4) \land (\neg x_2 \lor x_3 \lor \neg x_4)$$

Atribui-se valores à essas variáveis, caso exista uma combinação de valores que resulte em uma solução verdade ela é dita satisfatível. Assim SAT é um problema de decisão.

4. Prova

Agora irá ser mostrado a prova proposta pelo teorema Cook-Levin.

Para provar que o SAT é NP-Completo precisa provar dois tópicos:

1 – Mostrar que o SAT é um problema NP.

2 - Mostrar que cada problema NP pode ser reduzido por um algoritmo de tempo polinomial para um caso de um problema SAT.

Primeiro - Para mostrar o primeiro, basicamente temos que criar um algoritmo não-determinístico que gere uma solução aleatória em tempo polinomial e verificar se ela é válida.

Algoritmo:

1 – Atribuição de valores booleanos às variáveis dadas.

2 – Obter o resultado dessas atribuições na formula/expressão booleana em questão.

3 – Verificar se o resultado é verdadeiro ou falso.

Segundo – Agora para o tópico 2. Aqui é preciso reduzir um problema NP-Completo ao SAT, ou seja, se for feito a redução de apenas um deles ao SAT, significa que todos os outros também o serão.

Para a questão usaremos o problema 3-SAT e reduziremos ele ao SAT.

3-SAT é um caso especial do SAT, em que cada cláusula possui exatos 3 literais.

A notação: x <=p y. Significa que existe uma redução de x a y. Se x <=p y e y está em P, então x está em P.

SAT <=p 3-SAT

Legenda

Quando se diz 3 literais por clausula significa isso:

Ø = (x1 v !x1 v !x2) ^ (x3 v x2 v x4)

Um literal são as variáveis x1,x2.. e sua negação é !x1,!x2...

Uma clausula é (x1 v !x1 v !x2)...

Descreveremos um algoritmo polinomial T que recebe uma fórmula booleana Ø e devolve uma fórmula booleana Ø’ com exatamente 3 literais por cláusulas tais que:

Ø é satisfazível se e somente se Ø’ é satisfazível

A transformação consiste em substituir cada clausula de Ø por uma coleção de claúsulas com exatamente 3 literais cada e equivalente a Ø;

Seja (l1 ∨ l2 ∨ · · · ∨ lk) uma claúsula de Ø .

Caso 1. k = 1

Troque (l1)

por (l1 ∨ y1 ∨ y2) (l1 ∨ ¬ y1 ∨ y2) (l1 ∨ y1 ∨ ¬ y2) (l1 ∨ ¬ y1 ∨ ¬ y2) onde y1 e y2 são variáveis novas.

Caso 2. $k = 2$

Troque $(l_1 \lor l_2)$ por $(l_1 \lor l_2 \lor y)\ (l_1 \lor l_2 \lor \neg y)$. onde y é uma variáveis nova. Caso 3. $k = 3$

Mantenha $(l_1 \lor l_2 \lor l_3)$.

Caso 4. $k > 3$

Troque $(l_1 \lor l_2 \lor \cdots \lor l_k)$ por

$(l_1 \lor l_2 \lor y_1)$

$(\neg y_1 \lor l_3 \lor y_2)\ (\neg y_2 \lor l_4 \lor y_3)\ (\neg y_3 \lor l_5 \lor y_4) \ldots$

$(\neg y_{k-3} \lor l_{k-1} \lor l_k)$

onde $y_1, y_2, \ldots, y_{k-3}$ são variáveis novas

Verifique que Ø é satisfazível se e somente se nova fórmula é satisfazível. O tamanho da nova claúsula é O(m), onde m é o número de literais que ocorrem em Ø (contando-se as repetições).

Assim reduziu-se 3-SAT ao SAT, e como existe um problema NP-Completo que é possível reduzir a ele, significa que todos os outros são redutíveis de acordo com o teorema de Cook-Levin.

Referências

"Problemas NP-Completo"; IME. Disponível em < https://www.ime.usp.br/~weslley/probNP.htm >. Acesso em 11 de julho de 2018.

"Teorema de Cook-Levin";. Disponível em https://pt.wikipedia.org/wiki/Teorema_de_Cook-Levin>. Acesso em 11 de julho de 2018.