

Rainhas e ROBDD

Nathan Benedetto Proença e Victor Sena Molero

October 26, 2015

Abstract

Com o objetivo de aplicar os conhecimentos obtidos em sala de aula com o estudo de *Reduced Ordered Binary Decision Diagrams*, e de melhor compreender o funcionamento e a aplicabilidade de uma ferramenta tão ampla, modelamos e resolvemos o clássico problema das n rainhas.

1 Abordagem

Para cada célula do tabuleiro criamos uma variável booleana que indica se há ou não uma rainha posicionada. A própria formulação do problema torna obrigatória a presença de uma e somente uma rainha por coluna ou linha. Além disso é necessário garantir que não haja ataque nas diagonais, limitando a quantidade de peças de acordo.

Devido à simetria do problema, a descrição da solução para as linhas é suficiente para o caso das linhas e das colunas. Conseguimos facilmente descrever a existência de pelo menos uma rainha com a equação 1 apenas variando $\forall i \in 0 \dots n - 1$

$$\bigvee_{j \in (0 \dots n-1)} x_{i,j} \quad (1)$$

Limitar exigiu um pouco mais de sutileza. A solução por nós encontrada foi para todo par de variáveis na mesma linha negar sua conjunção. Note que

$$\neg(a \wedge b) \equiv \neg a \vee \neg b$$

Assim resolvemos os ataques com

$$\neg x_{i,j} \vee \neg x_{i,k} \quad \forall (j, k) \in (0..n-1) \times (0..n-1) \quad (2)$$

Felizmente essa segunda equação também descreve a mesma abordagem utilizada para evitar conflitos nas diagonais, apenas adaptando as posições no tabuleiro para serem coerentes. Após pré-computar as diagonais do tabuleiro, temos então todas as cláusulas que precisam ser respeitadas para se encontrar uma solução do problema das n -rainhas.

2 Conclusões e performance

Conseguimos resolver com grids de até tamanho 8. Pensando no problema como um sat de n^2 variáveis, se trata de uma solução muito eficiente. No entanto, é

possível encarar como uma busca entre as $n!$ permutações. Com essa segunda abordagem, podemos trivialmente buscar todas as soluções com complexidade $O(n^2n!)$, que nos permitira resolver esse problema com um tabuleiro de 9×9 ou 10×10 .