

# Tópicos em Fundamentos da Computação

## Aula 10

### SAT – Modificações DPLL

Karina Girardi Roggia  
karina.roggia@udesc.br

Departamento de Ciência da Computação  
Centro de Ciências Tecnológicas  
Universidade do Estado de Santa Catarina

2024



# Aprendizado de Novas Cláusulas

- Aprender = **adição** de novas cláusulas, sem aumentar o número de átomos.
- Encolher o espaço de busca! As novas cláusulas são de tal forma que proíbem a repetição de más escolhas.
- Novas cláusulas são adicionadas quando encontradas contradições no processo de simplificação.



# Aprendendo das Más Escolhas

- DPLL: construção de valoração que satisfaça o conjunto de cláusulas, a partir de uma valoração parcial, inicialmente vazia.
- Expansão da valoração:
  - pela escolha de literais;
  - pela propagação unitária.
- Quando uma contradição é encontrada, faz-se *backtracking*, contraindo a valoração parcial construída.



# Aprendendo das Más Escolhas

Ao iniciar o DPLL

$$\mathcal{V} = \emptyset$$

Supondo que não há cláusulas unitárias no conjunto. Então deve-se escolher algum literal para a valoração. Seja **a<sub>1</sub>** o literal escolhido.

$$\mathcal{V} = \{\mathbf{a_1}\}$$

Procede-se à propagação unitária, que pode obter a valoração de outros literais  $a_2, \dots, a_{k_a}$

$$\mathcal{V} = \{\mathbf{a_1}, a_2, \dots, a_{k_a}\}$$



# Aprendendo das Más Escolhas

Se não houver contradição e o conjunto não estiver totalmente valorado, então deve-se proceder a nova escolha de literal e executar a propagação unitária

$$\mathcal{V} = \{\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_{k_a}, \mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_{k_b}\}$$

Caso seja encontrada uma contradição, o algoritmo DPLL desfaz a propagação unitária de  $\mathbf{b}_1$  e inicia tentativa com

$$\mathcal{V} = \{\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_{k_a}, \sim \mathbf{b}_1\}$$

A derivação da contradição indica que  $\mathbf{a}_1$  e  $\mathbf{b}_1$  não são possíveis simultaneamente, ou seja,  $\neg(\mathbf{a}_1 \wedge \mathbf{b}_1)$ , o que é equivalente à cláusula

$$\neg \mathbf{a}_1 \vee \neg \mathbf{b}_1$$

que pode ser adicionada ao conjunto de cláusulas a serem satisfeitas.



# Aprendendo das Más Escolhas

- Tal aprendizagem, utilizada isoladamente, *não* melhora muito a eficiência.
- Tendo-se escolhido  $k$  literais no ramo em que a cláusula foi aprendida, tal aprendizado somente será usado se  $k - 1$  literais aparecerem em outro ramo do DPLL.
- Deve-se empregar conjuntamente com outras técnicas, como a de Reinício Aleatório.



# Aprendendo por Inferência

- Aprendizado a partir das cláusulas envolvidas na obtenção da contradição encontrada
- *Resolução* das cláusulas que resultaram na contradição
- Necessária adição de informação junto à valoração parcial
  - Literal obtido por propagação unitária: qual a cláusula que deu origem ao literal
  - Literal valorado por escolha: associado ao termo  $\top$



# Aprendendo por Inferência – Exemplo

Seja o conjunto de cláusulas

$$\{p \vee q, p \vee \neg q, \neg p \vee t \vee s, \neg p \vee \neg t \vee s, \neg p \vee \neg s\}$$

Tome-se as escolhas  $\neg t$  seguida por  $p$ , representadas pela valoração

$$\{(\neg t, \top), (p, \top)\}$$

Após a propagação linear, tem-se

$$\{(\neg t, \top), (p, \top), (\neg s, \neg p \vee \neg s), (s, \neg p \vee t \vee s)\}$$





# Aprendendo por Inferência – Exemplo

Aplicando-se a regra de resolução sobre as cláusulas que geraram a contradição

$$\frac{\frac{\neg p \vee \neg s \quad s \vee t \vee \neg p}{\neg p \vee t \vee \neg p}}{\neg p \vee t}$$

O que resulta no aprendizado da cláusula  $\neg p \vee t$ .

Na prática, este tipo de aprendizado traz efeitos mais positivos sobre a eficiência do DPLL.



Extensão do DPLL que utiliza aprendizado e uso dos seguintes elementos:

- Literais Vigados
- Retrossaltos (*backjumping*)
- Reinícios Aleatórios
- Heurística de Seleção de Literais



- 80% do tempo de execução de algoritmo DPLL é destinado à propagação unitária
- É interessante, portanto, investir em otimização desta parte da execução
- Propriedades da Técnica dos Literais Vigados:
  - Aceleração da Propagação Unitária
  - Sem necessidade de apagar literais ou cláusulas
  - Sem necessidade de vigilância de *todos* os literais de uma cláusula
  - Retrocesso em tempo constante (muito rápido)



Estrutura de Dados tal que:

- A cada instante, toda cláusula  $c$  tem exatamente **dois** literais selecionados:  $\lambda_{c_1}$  e  $\lambda_{c_2}$ ;
- $\lambda_{c_1}$  e  $\lambda_{c_2}$  são escolhidos dinamicamente, e mudam durante a execução;
- $\lambda_{c_1}$  e  $\lambda_{c_2}$  são *propriamente vigiaados*, ou seja,
  - ambos são indefinidos, ou
  - ao menos um deles é valorado com 1.



Início:

- $\mathcal{V} = \emptyset$
- cada cláusula terá um par de literais vigiados (independente da escolha, todo par será propriamente vigiado)

Execução do DPLL:

- Expansão de  $\mathcal{V}$ : escolha de literais ou propagação unitária
- Algum literal vigiado pode ser falsificado, tornando o par impróprio
- Necessário acionar algoritmo de manutenção dos literais vigiados



# Literais Vigiadados – Manutenção

Caso um par de literais vigiadados se torne impróprio:

- troca-se os literais vigiadados, retirando os literais valorados em 0 por outros da cláusula que tornem o par próprio
- se não há par próprio, duas situações podem ocorrer
  - há um único literal indefinido na cláusula
  - todos os literais foram falsificados



# Literais Vigiados – Manutenção

Há um único Literal indefinido na cláusula

- O literal será valorado como 1
- Expansão da valoração  $\mathcal{V}$
- Propagação da informação



# Literais Vigiaados – Manutenção

Todos os literais foram falsificados

- Necessário fazer o retrocesso da valoração
- Apagar as propagações unitárias até achar um literal que foi escolhido
- Inverte-se a escolha do literal
- A propagação é feita pelos literais vigiaados





## Literais Vigados – Exemplo

Cláusula	$\lambda_{c_1}$	$\lambda_{c_2}$
$p \vee q \vee r$	$p = *$	$q = *$
$p \vee \neg q \vee s$	$p = *$	$\neg q = *$
$p \vee r \vee \neg s$	$p = *$	$r = *$

Supondo escolha do literal  $\neg p$  para a valoração, ou seja,  $\mathcal{V} = \{\neg p\}$ .



## Literais Vigados – Exemplo

Cláusula	$\lambda_{c_1}$	$\lambda_{c_2}$
$p \vee q \vee r$	$p = *$	$q = *$
$p \vee \neg q \vee s$	$p = *$	$\neg q = *$
$p \vee r \vee \neg s$	$p = *$	$r = *$

Supondo escolha do literal  $\neg p$  para a valoração, ou seja,  $\mathcal{V} = \{\neg p\}$ .  
Então todos os pares de literais ficam não próprios, portanto cada cláusula terá um novo par de literais vigiados:

Cláusula	$\lambda_{c_1}$	$\lambda_{c_2}$
$p \vee q \vee r$	$r = *$	$q = *$
$p \vee \neg q \vee s$	$s = *$	$\neg q = *$
$p \vee r \vee \neg s$	$\neg s = *$	$r = *$



# Literais Vigados – Exemplo

Nova escolha deve ser feita. Seja então  $\mathcal{V} = \{\neg \mathbf{p}, \neg \mathbf{r}\}$ .

Cláusula	$\lambda_{c_1}$	$\lambda_{c_2}$
$p \vee q \vee r$	$r = 0$	$q = *$
$p \vee \neg q \vee s$	$s = *$	$\neg q = *$
$p \vee r \vee \neg s$	$\neg s = *$	$r = 0$



## Literais Vigiadados – Exemplo

Nova escolha deve ser feita. Seja então  $\mathcal{V} = \{\neg \mathbf{p}, \neg \mathbf{r}\}$ .

Cláusula	$\lambda_{c_1}$	$\lambda_{c_2}$
$p \vee q \vee r$	$r = 0$	$q = *$
$p \vee \neg q \vee s$	$s = *$	$\neg q = *$
$p \vee r \vee \neg s$	$\neg s = *$	$r = 0$

A primeira e terceira cláusulas tornam-se impróprias. Como não é possível escolher novos literais não valorados para substituir os literais com valor 0, então os literais vigiados não valorados passarão a valer 1. Logo  $\mathcal{V} = \{\neg \mathbf{p}, \neg \mathbf{r}, q, \neg s\}$ .

Cláusula	$\lambda_{c_1}$	$\lambda_{c_2}$
$p \vee q \vee r$	$r = 0$	$q = 1$
$p \vee \neg q \vee s$	$s = *$	$\neg q = *$
$p \vee r \vee \neg s$	$\neg s = 1$	$r = 0$



# Literais Vigiadados – Exemplo

Sucedese à propagação nos literais vigiadados.

Cláusula	$\lambda_{c_1}$	$\lambda_{c_2}$
$p \vee q \vee r$	$r = 0$	$q = 1$
$p \vee \neg q \vee s$	$s = 0$	$\neg q = 0$
$p \vee r \vee \neg s$	$\neg s = 1$	$r = 0$



## Literais Vigiadados – Exemplo

Suced-se à propagação nos literais vigiadados.

Cláusula	$\lambda_{c_1}$	$\lambda_{c_2}$
$p \vee q \vee r$	$r = 0$	$q = 1$
$p \vee \neg q \vee s$	$s = 0$	$\neg q = 0$
$p \vee r \vee \neg s$	$\neg s = 1$	$r = 0$

Note que não há forma de conseguir um par de literais próprios para a segunda cláusula. Portanto deve ser feito um retrocesso da valoração até a última escolha, invertendo-a.

$$\mathcal{V} = \{\neg \mathbf{p}, \neg \mathbf{r}, q, \neg s\} \Rightarrow \mathcal{V} = \{\neg \mathbf{p}, r\}$$

E atualizam-se os valores dos literais vigiadados:

Cláusula	$\lambda_{c_1}$	$\lambda_{c_2}$
$p \vee q \vee r$	$r = 1$	$q = *$
$p \vee \neg q \vee s$	$s = *$	$\neg q = *$
$p \vee r \vee \neg s$	$\neg s = *$	$r = 1$



Técnica para evitar duplicação de esforços causados por má escolha de literal que não influencia na obtenção de contradições posteriores.

Exemplo:

$$\mathcal{V} = \{(\neg \mathbf{p}, \top), (\neg \mathbf{r}, \top), (\mathbf{a}, \top), (q, p \vee q \vee r), \\ (\neg s, p \vee r \vee \neg s), (s, p \vee \neg q \vee s)\}$$

A escolha do literal  $\mathbf{a}$  não influenciou na obtenção da contradição, portanto o retrocesso invertendo a polaridade de  $\mathbf{a}$  provocará nova contradição.

Logo, faz-se o *backtracking* diretamente sobre a escolha de  $\neg \mathbf{r}$ .



Cenário:

- fórmula com 1000 átomos, satisfazível
- todas as valorações que a satisfazem são tais que  $p_1 = 0$
- primeira escolha feita no DPLL:  $p_1 = 1$
- possibilidade de encontro de valoração que satisfaça a fórmula: depois da exploração de  $2^{999}$  ramos (pior caso)





Possível solução:

- aliar aprendizado de cláusulas com possibilidade de reinício do processo de escolha de literais
- o aprendizado de novas cláusulas pode influenciar na escolha do literal e qual polaridade utilizar (heurísticas)
- utilizar parâmetro  $\varepsilon$  com probabilidade de reinício da valoração (exemplo:  $\varepsilon = 0,5\%$ )
- quando encontrada contradição, após o aprendizado de cláusulas, o retrocesso ocorre com probabilidade  $1 - \varepsilon$
- com probabilidade  $\varepsilon$ , toma-se  $\mathcal{V} = \emptyset$
- Deve-se manter as fórmulas aprendidas ao reiniciar!!!



## VSIDS – *Variable State Independent Decaying Sum*

- Heurística para escolha de literais
- Privilegia a satisfação de cláusulas recém aprendidas
- Utilização de contadores simples



- Cada polaridade de cada átomo terá um contador que inicia zerado
- Para cada cláusula, é incrementado o contador de cada um dos literais presentes
- Quando uma cláusula é aprendida, os contadores dos literais presentes devem ser atualizados
- Os literais com maior contador são os candidatos à seleção
- Em caso de empate, utiliza-se escolha aleatória entre os habilitados
- Periodicamente, todos os contadores são divididos por uma constante
- Deste modo, literais presentes em cláusulas recém aprendidas tem maior probabilidade de serem escolhidos

