

# APPLY VOLGORDE VAN EEN SDD-COMPILATIE



Auteur:  
Gijs Vliegen

Supervisor:  
Prof. Dr. Luc De Raedt

Begeleiders:  
Dr. Ir. Jessa Bekker  
Dr. Ir. Vincent Derkinderen

## Motivatie

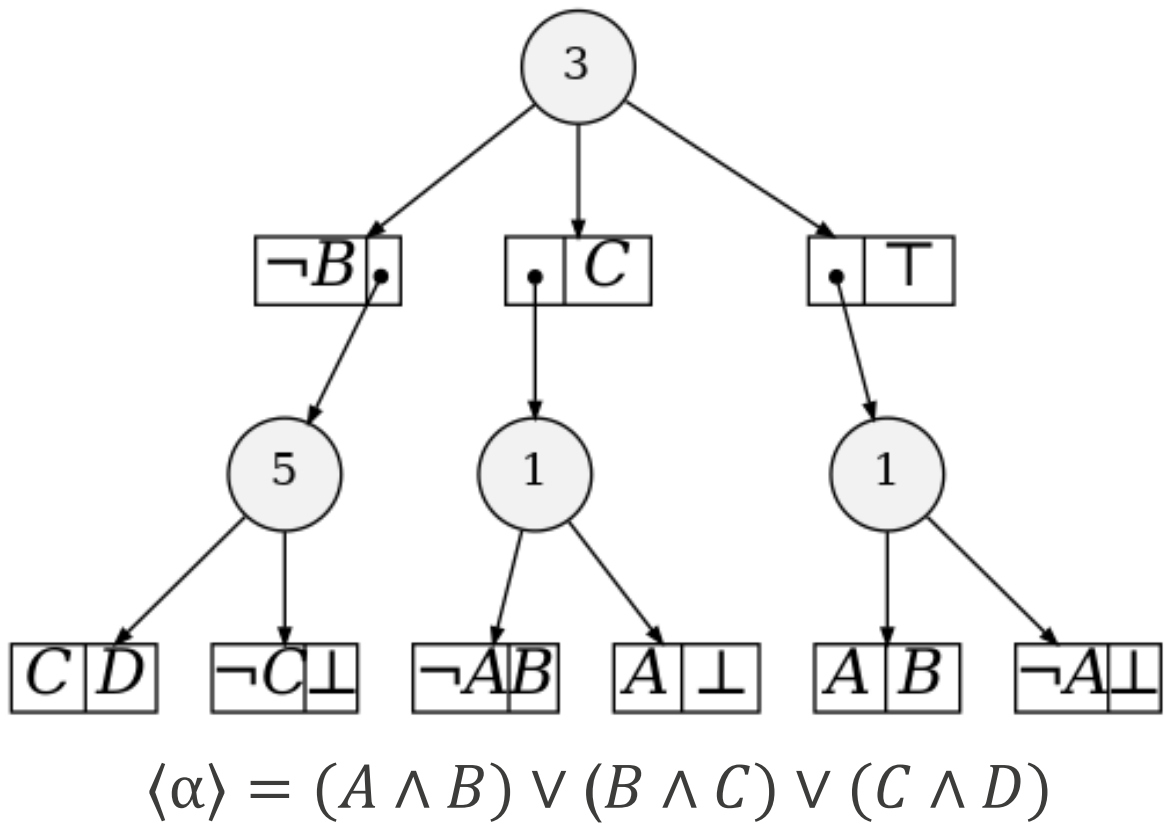
- Logische circuits hebben toepassingen
- Probabilistische inferentie
  - Foutdetectie en foutdiagnose
  - Stochastische beperking-optimalisatie-problemen

SDD's zijn logische circuits met nuttige eigenschappen

- Bewerkingen in polytijd
- Bottom-up compileerbaar
- Theoretisch compact

## Sentential Decision Diagram $\alpha$

- $\alpha = \top$  of  $\alpha = \perp$ ,  $\langle \top \rangle = \text{Waar}$ ,  $\langle \perp \rangle = \text{Onwaar}$
- $\alpha = X$  of  $\alpha = \neg X$ ,  $\langle X \rangle = X$ ,  $\langle \neg X \rangle = \neg X$
- $\alpha = \{(p_1, s_1), \dots (p_n, s_n)\}$ ,  $\langle \alpha \rangle = \bigvee_{i=1}^n \langle p_i \rangle \wedge \langle s_i \rangle$   
Met elke  $p_i, s_i$  een SDD

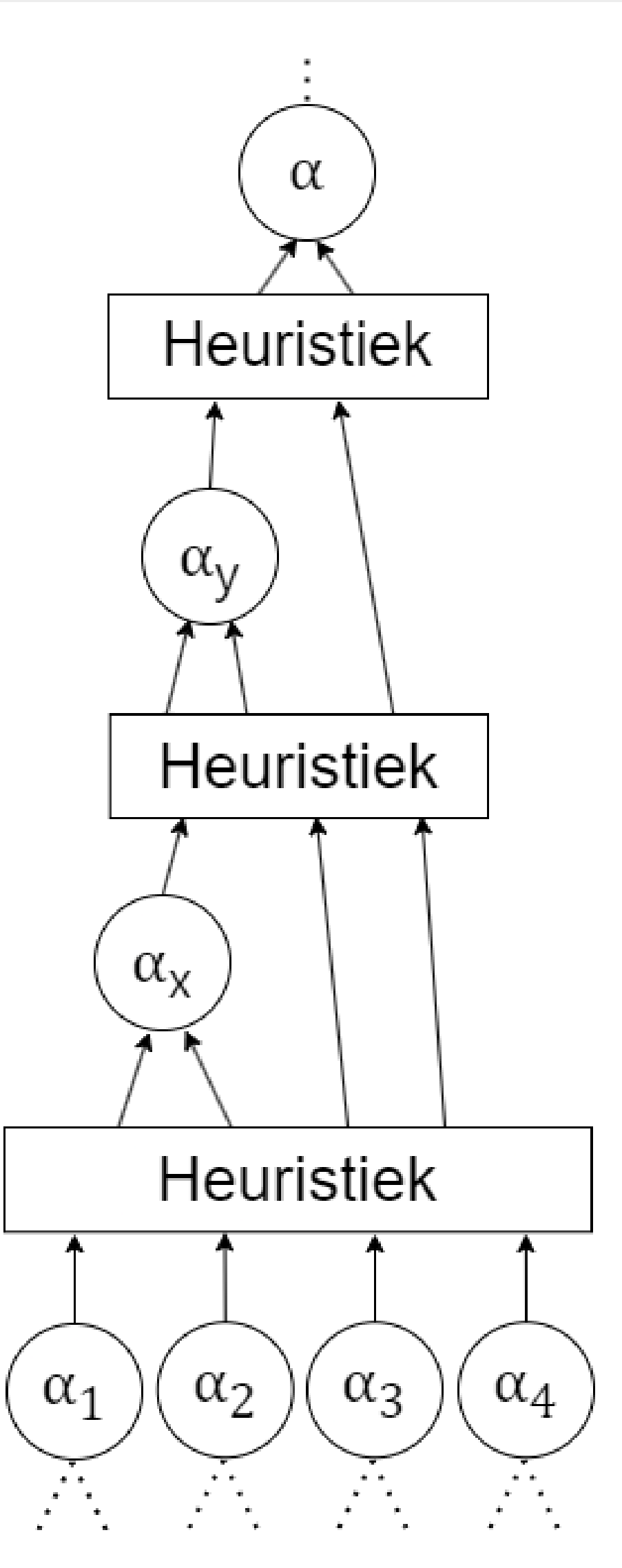
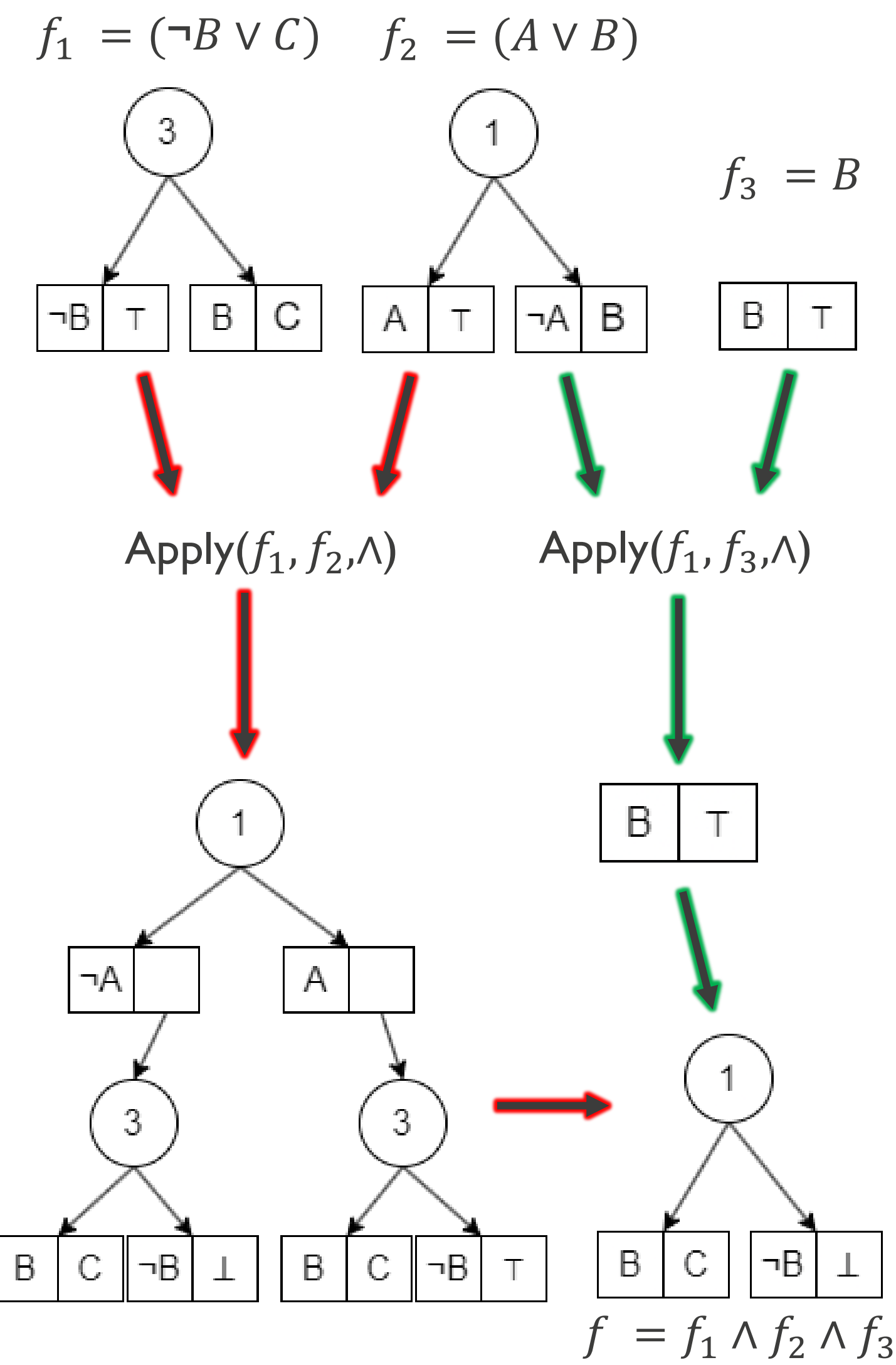


## Tussenresultaten

- Bij bottom-up compilatie
- Samennemen van twee SDD's mbv. Apply
- Welke SDD's eerst? Volgorde bepalen!

SDD is canoniek  
↓  
Volgorde te kiezen, eindresultaat gelijk

## Voorbeeld



## Heuristieken

Op basis van de SDD's in subC.

- $|\alpha \circ \beta| \leq |\alpha| * |\beta|$  ⇒ Kleinste Eerst
- $|\alpha \circ \beta| = \sum el(\alpha \circ \beta, v)$  ⇒ Element bovengrens

Op basis van de **vtree** variabelen:

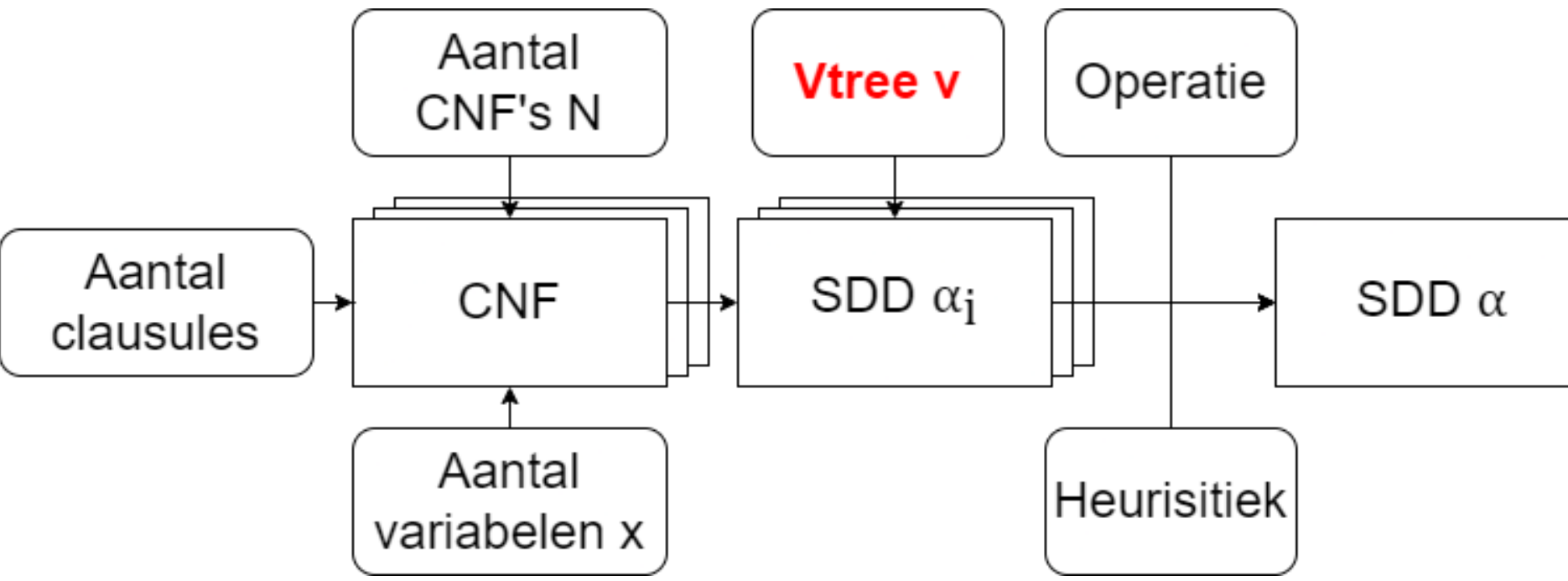
- Hoog en links eerst: ⇒ Vtree variabelen Volgorde
- Laag en links eerst: ⇒ Inverse Volgorde RL
- Laag en rechts eerst: ⇒ Inverse Volgorde LR

Opsplitsen in deelproblemen

- Volgens de vtree: ⇒ Vtree Partitionering

## Experimenten

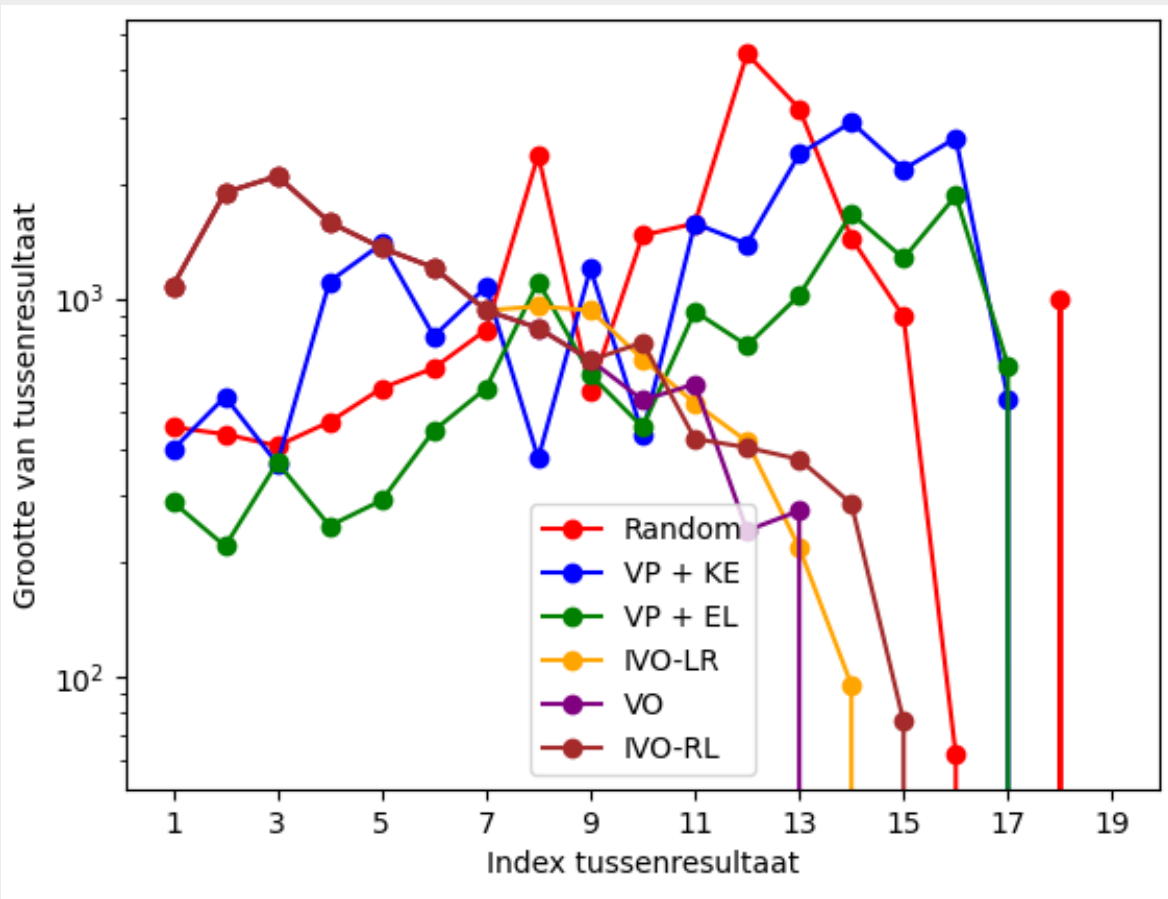
- Generatie mbv. CNF's
- Ratio  $r = \text{clausules}/\text{vars}$  als moeilijkheidsgraad
- Heuristiek of willekeurig
- Metrieken: Compilatie tijd & Groottes van tussenresultaten



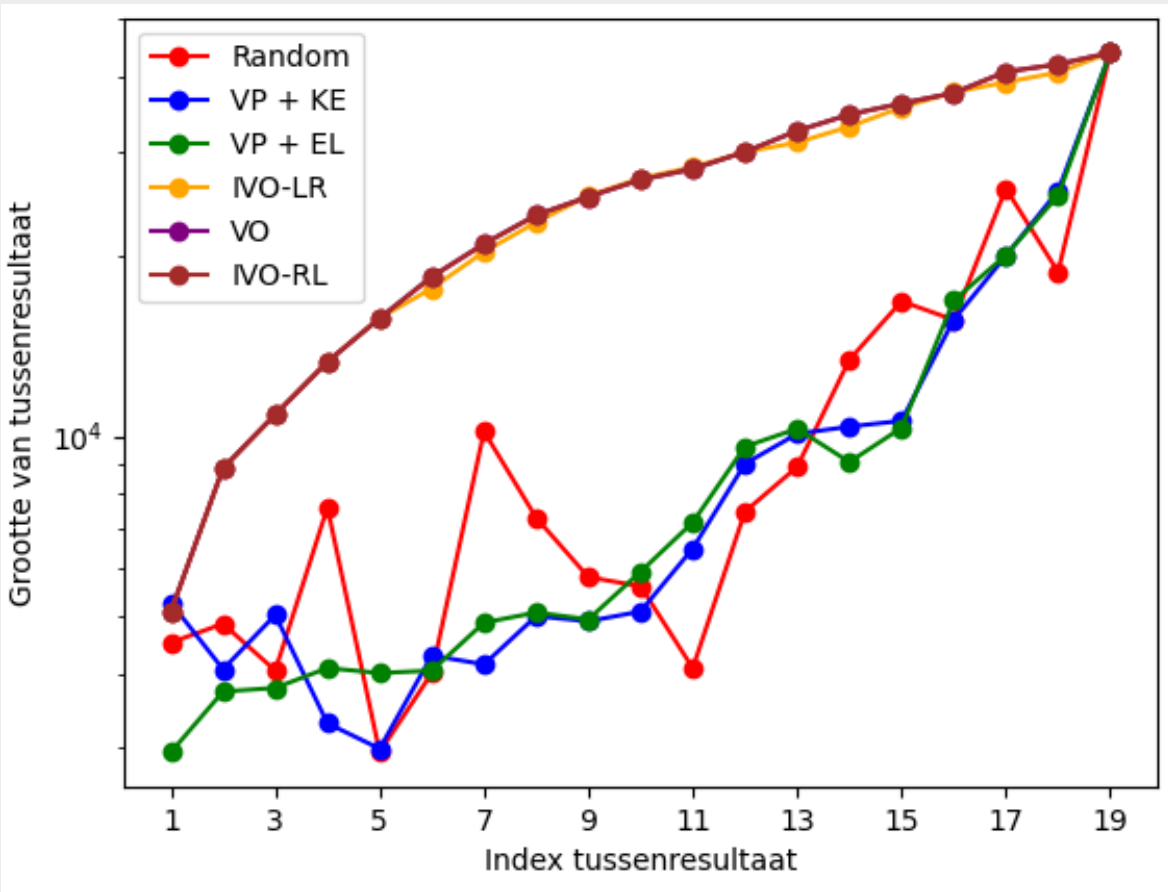
## Voorlopige resultaten

### Groottes

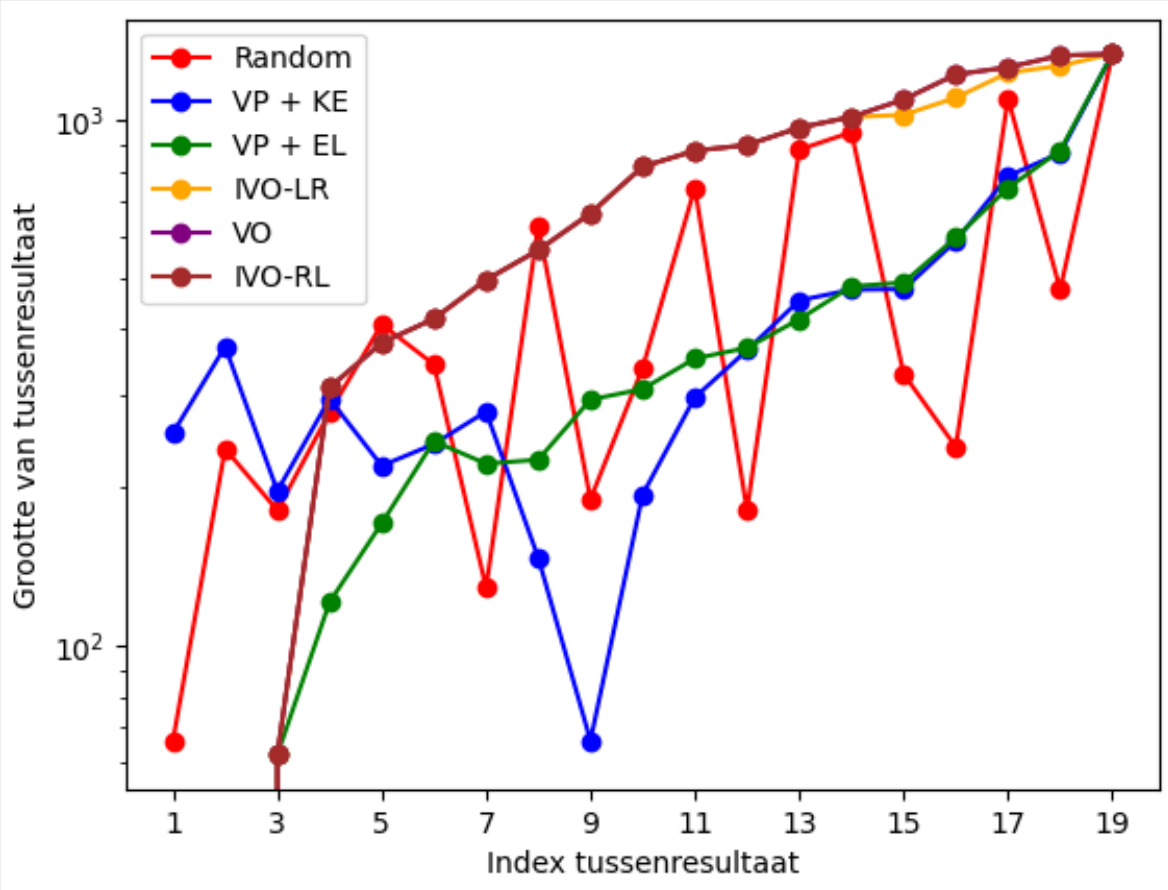
$R = 0,25$



$R = 2$



$R = 4$



### Compilatie tijd

