

# Ergebnisse Dynamischer Bitvektor und Balanced Parentheses

Projekt - Fortgeschrittene Datenstrukturen

Manuel Haag | 25. Juli 2022

# Implementierung

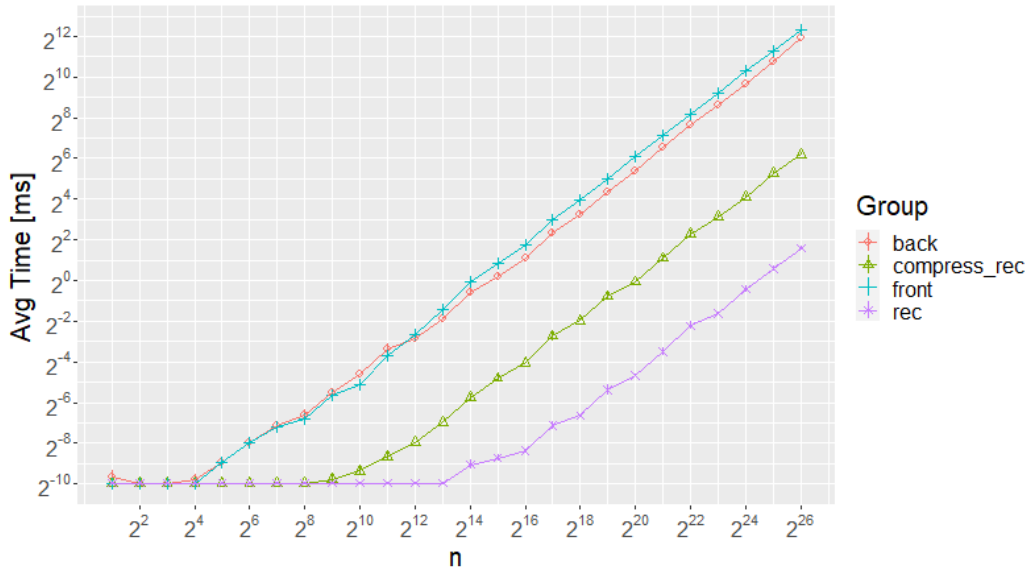
## Code

- Codezeilen: 4118 (src), 1247 (test)
- C++20

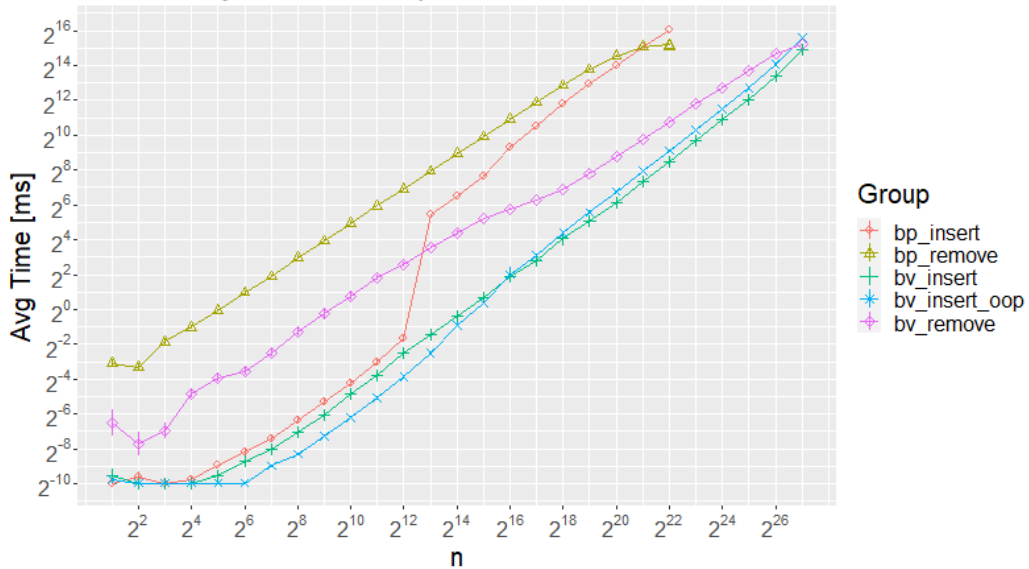
## Besonderheiten

- AVL-Tree mit statischen Blättern
- naives Balancieren
- Fallunterscheidung für Knoten im Baum
- iterativ

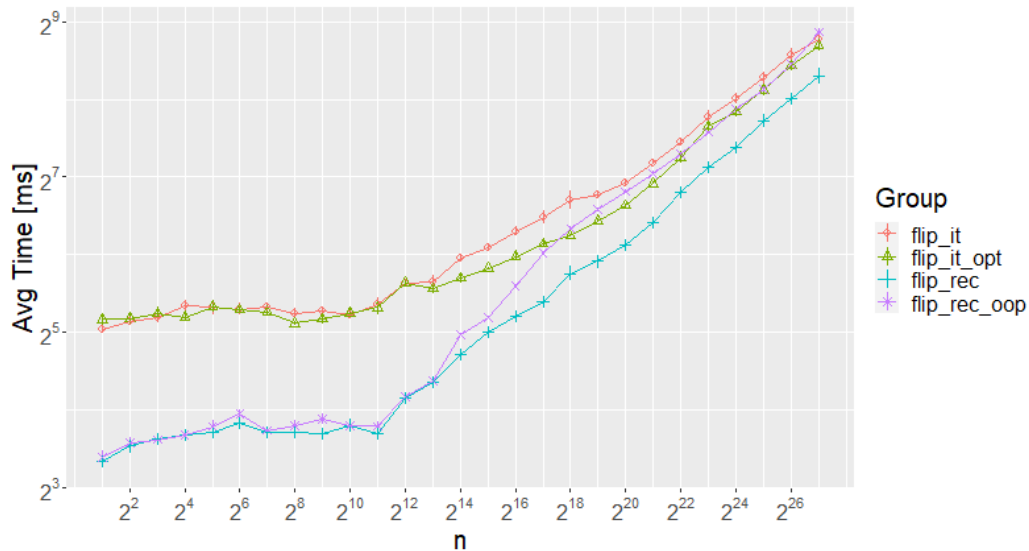
## construction time, 64 leaf blocks



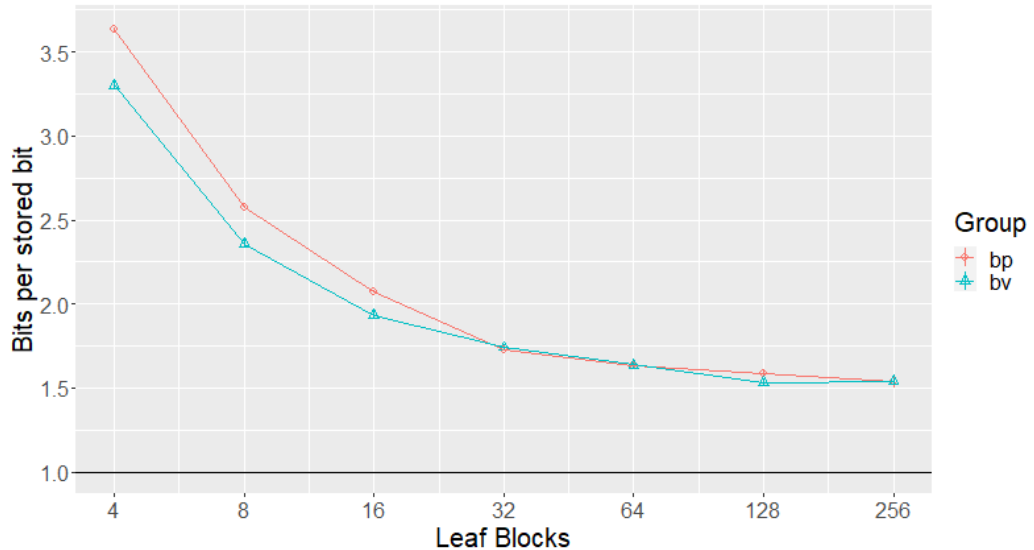
run time by number of operations n



run time of  $m = 10^6$  flip operations



## space usage by leaf sizes, $n = 2^{18}$



# Lektion gelernt?

## Takeaway

- Baseline Implementierung + viele Teste sind hilfreich
- "premature optimization is the root of all evil"
- Benchmarken!