

# Vergleich der Komplexität von Tree-Search-Algorithmen

13. April 2023

## **Abstract**

In diesem Projekt wird die Komplexität von verschiedenen einfachen Tree-Search-Algorithmen untersucht. Es werden verschiedene Algorithmen wie die Tiefensuche, Breitensuche und andere heuristische Suchalgorithmen auf binären Suchbäumen getestet. Dabei wird die Anzahl der durchsuchten Knoten und die benötigte Zeit gemessen, um die Effizienz der Algorithmen zu vergleichen. Das Ziel ist es, die Komplexität der Suchalgorithmen zu berechnen, um diese dadurch zu vergleichen.

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>4</b>
<b>2</b>	<b>Theorie</b>	<b>4</b>
2.1	Suchbaum . . . . .	4
2.1.1	binärer Suchbaum . . . . .	4
2.2	Tree-Search Algorithmus . . . . .	4
2.3	Komplexität von Tree-Search-Algorithmen . . . . .	4
<b>3</b>	<b>Vergleich der Algorithmen</b>	<b>5</b>
<b>4</b>	<b>Resultate</b>	<b>5</b>
<b>5</b>	<b>Schlussfolgerung</b>	<b>5</b>
<b>6</b>	<b>Reflexion</b>	<b>5</b>
	<b>Literatur</b>	<b>6</b>

# 1 Einleitung

(Goodrich, T., Tamassia & Mount, 2011)

## 2 Theorie

### 2.1 Suchbaum

Suchbäume sind Datenstrukturen aus der Graphentheorie. Der Begriff "Suchbaum" bezieht sich auf die Tatsache, dass die Datenstruktur als Baum organisiert ist. Ein Baum besteht aus einer Wurzel, die den obersten Knoten (engl. nodes) darstellt, und einer Anzahl von Zweigen oder Kanten, die von der Wurzel zu den Blättern führen, die die Endpunkte des Baums darstellen. Jeder Knoten im Baum repräsentiert eine Entscheidung oder einen Zustand, und jeder Zweig repräsentiert eine Aktion, die von diesem Zustand ausgeführt werden kann.

Ein Baum ist also ein zusammenhängender Graph, der keine geschlossenen Pfade enthält (Diestel, 1996 - 1996), d.h. sich nur in eine Richtung von der Wurzel ausbreitet.

#### 2.1.1 binärer Suchbaum

Ein binärer Suchbaum ist eine spezielle Art von Suchbaum, bei der jeder Knoten höchstens zwei Kindknoten (engl. child nodes) haben kann. Ausserdem erfordert ein binärer Suchbaum, im Gegensatz zum normalen Suchbaum, eine bestimmte Anordnung der Knoten: Diese sind so angeordnet, dass jeder Knoten kleiner als alle Knoten im rechten Teilbaum ist, aber grösser als alle Knoten im linken Teilbaum.

Der Vorteil dieser speziellen Anordnung ist, dass man schnell einen Knoten im Baum finden kann. Auch ist das Einfügen und Löschen von Knoten aus dem Baum einfacher.

### 2.2 Tree-Search Algorithmus

Tree-Search-Algorithmen sind eine Art von Algorithmen, die zur Lösung von Suchproblemen verwendet werden. Sie sind nützlich, wenn man aus einer Vielzahl von möglichen Optionen den passenden finden muss.

Tree-Search-Algorithmen arbeiten mit Suchbäumen, bei denen die verschiedenen Optionen als Knoten, die über Kanten dargestellt sind. Die Suche nach der Lösung des Problems besteht darin, durch die Baumstruktur zu navigieren und den Pfad finden, der von der Wurzel (engl. root) bis zum Zielzustand führt.

### 2.3 Komplexität von Tree-Search-Algorithmen

Das Durchsuchen eines Baumes nach einem gewünschten Zustand dauert nicht immer gleich lange. Es hängt vielmehr von einer Vielzahl verschiedener Faktoren ab, wie zum Beispiel der Größe und Tiefe des Suchbaums oder der Anzahl der Knoten. Eine Schätzung für die obere Schranke der Suchdauer und damit der Effizienz eines Tree-Search-Algorithmus gibt die Komplexität.

Die Komplexität von Tree-Search-Algorithmen wird in der sogenannten  $O$ -Notation ausgedrückt, die angibt, wie schnell die Laufzeit eines Suchalgorithmus mit der Grösse eines Baumes wächst. Eine Komplexität von  $O(n)$  zum Beispiel bedeutet, dass die Laufzeit eines Algorithmus linear von der Knotenzahl  $n$  abhängt.

- 3 Vergleich der Algorithmen
- 4 Resultate
- 5 Schlussfolgerung
- 6 Reflexion

## Literatur

Diestel, R. (1996 - 1996). *Graphentheorie*. Berlin: Springer.

Goodrich, T., M., Tamassia, R. & Mount, D. M. (2011). *Data structures and algorithms in c++*. 2nd ed. Hoboken: Wiley. print.