

# Bachelorarbeit

Andreas Windorfer

17. Juli 2020

# Inhaltsverzeichnis

<b>1</b>	<b>Multisplay Baum</b>	<b>3</b>
1.1	Die <i>access</i> Operation beim Multisplay Baum . . . . .	4

# 1 Multisplay Baum

Der Multisplay Baum ist eine Variation zum Tango Baum. Ein preferred path wird hier durch einen Splaybaum dargestellt. Amortisiert betrachtet, ist er  $\log(\log(n))$ -competitive und garantiert  $O(\log(n))$  im worst case, bei einer einzelnen *access* Operation.  $n$  steht wieder für der Anzahl der Knoten von  $T$ . Da der Splaybaum kein balancierter Baum ist, gibt es zusätzliche mögliche Zustände im Vergleich zu einem Tango Baum mit der gleichen Knotenzahl. Auch der Multisplay Baum verwendet einige Hilfsdaten je Knoten. Zum

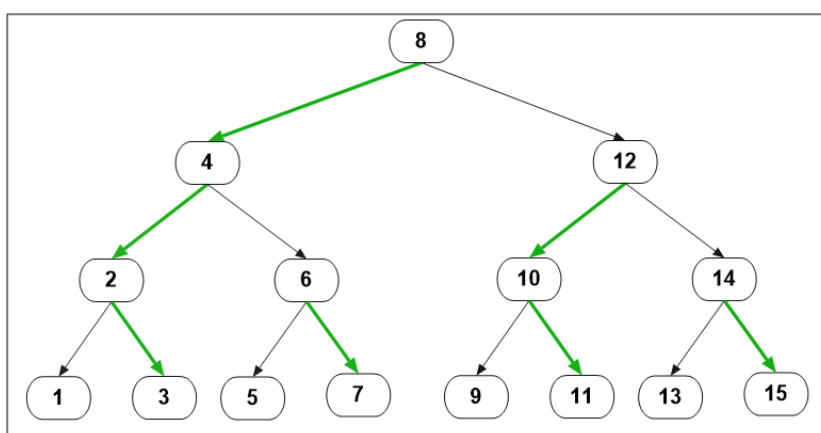


Abbildung 1: Referenzbaum mit grün gezeichneten preferred paths

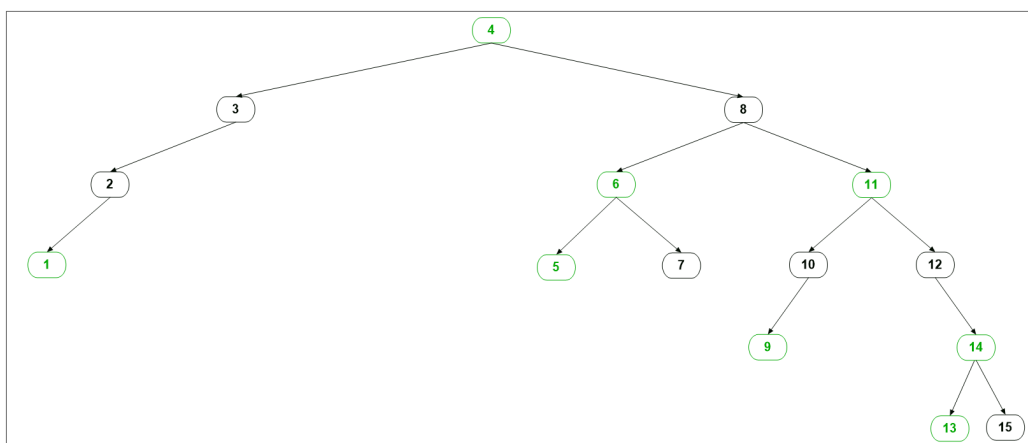


Abbildung 2: Referenzbaum mit grün gezeichneten preferred paths

einen bereits bekannte Variablen bzw. Konstanten *isRoot*, *depth* und *min-Depth*. Aber auch welche die beim Tango Baum nicht verwendet sind. Sei  $v$

ein Knoten in  $T$  und  $u$  der Knoten mit  $key(v) = key(u)$  im Referenzbaum  $P$ . Sei  $H$  der Hilfsbaum der  $v$  enthält. Die Konstante *height* hat den Wert der Höhe von  $u$ . Die Variable *treeSize* enthält die Anzahl der Knoten von  $H$ .

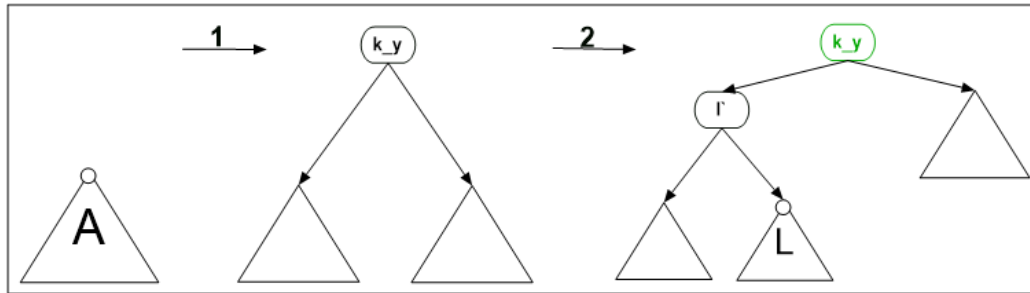
## 1.1 Die *access* Operation beim Multisplay Baum

Zu beachten ist, dass jede BST Darstellung auch eine Splaybaum Darstellung ist. Anders als beim Tango oder Zipper Baum, muss ein neu erzeugter Hilfsbaum also nicht so angepasst werden, dass er weitere Invarianten einhält. Nach einer *access(k)* Operation ist der Knoten  $v_k$  mit dem Schlüssel  $k$  die Wurzel von  $T$ . Zunächst wird eine gewöhnliche Suche in  $T$  durchgeführt, bis der Zeiger  $p$  der Operation auf  $v_k$  zeigt. Eine Abweichung zu den preferred path des Tango Baum ist, dass das preferred child des Knoten mit Schlüssel  $k$  zunächst unverändert bleibt. Ist  $v_k$  gefunden werden die Pfadrepräsentationen aktualisiert. Dabei wird bottom up vorgegangen.

**cut Operation beim Multisplay Baum** Sei  $u$  ein Knoten in  $P$ , mit Schlüssel  $k_u$  an dem das preferred child vom linken zum rechten Kind gewechselt ist, und sei  $u_r$ , das preferred child von  $u$  nach der *access* Operation. Seien  $v$  bzw.  $v_r$  die Knoten in  $T$  mit  $key(v) = k_u$  bzw.  $key(v_r) = key(u_r)$ . Sei  $B$  der Hilfsbaum der  $v_r$  enthält und  $A$  der Hilfsbaum der den Knoten mit  $key(u)$  enthält.  $S$  ist die Menge der Knoten aus  $A$ , bei denen der Wert von *depth* größer ist, als die Riefe von  $u$ . Zunächst wird die *isRoot* Variable von der Wurzel von  $A$  auf *false* gesetzt und *splay(k<sub>u</sub>)* auf  $A$  ausgeführt. Dadurch entsteht ein Hilfsbaum  $C$  mit Wurzel  $v$ , bei der *isRoot* auf *true* gesetzt wird. Sei  $C_L$  der linke Teilbaum von  $C$ .

Es wird der Vorgänger  $v_l$  des Knotens aus  $S$  mit dem kleinsten Schlüssel benötigt. Sei  $l'$  der Schlüssel von  $v_l$ . Der Knoten in  $P$  mit Schlüssel  $key(l')$  muss eine kleinere Tiefe als  $u$  haben. Deshalb kann  $l'$  gefunden werden indem wie folgt vorgegangen wird.  $p$  muss auf die Wurzel von  $C_L$  gesetzt werden. In einer Schleife wird  $p$  so oft auf das linke Kind  $p_l$  von  $p$  gesetzt, bis der Wert der *minDepth* Variable von  $p_l$  größer als die Tiefe von  $u$  ist. Nun wird *splay(l')* auf  $C_l$  ausgeführt. Bei der Wurzel des rechten Teilbaumes von  $v_l$  muss nun *isRoot* noch auf *true* gesetzt werden. Das Vorgehen ist sehr Ähnlich zu dem aus ???. Existiert  $v_l$  nicht, entfällt die zweite *splay* Operation und es wird *isRoot* der Wurzel von  $C_L$  auf *true* gesetzt. Ein Wechsel eines preferred child vom rechten zum linken Kind, kann direkt abgeleitet werden.

**join Operation beim Multisplay Baum** Seien  $A$  und  $B$  die Parameter der Operation, wobei  $A$  die Knoten enthält bei denen die *depth* Variable kleinere Werte annimmt.



**Abbildung 3:** Ablauf zum Erzeugen einer neuen Pfadrepräsentation, nach einem preferred child Wechsel vom linken Kind zum Rechten.

## Literatur