



Zauberwürfelloseroboter

# Hilfsmittel

# Hilfsmittel

Der Server nutzt verschiedene Werkzeuge, wie den Move-Pruner oder den Farbkorrigierer.

## Der Move-Pruner

Der Move-Pruner vereinfacht Züge. Er basiert darauf, dass Züge in sogenannte *Gruppen* zerlegt werden, die dann wieder zu Zügen zusammengesetzt werden.

Eine Gruppe besteht dabei aus den Bewegungen zweier gegenüberliegenden Seiten. Nach einer Gruppe muss immer eine Gruppe anderen Typs kommen, also nach einer Gruppe `{R: 1, L: 2}` muss eine Gruppe kommen, die keine `L` und `R` Züge mehr hat.

Dann werden die Gruppen wieder in Züge umgewandelt, indem für jede Gruppe nacheinander die Anzahl der Rotationen Modulo 4 gerechnet wird. Dann wird die Seite plus ein Modifikator ( `2` und `'` ) für die Rotationen 2 und 3 hinzugefügt.

Dadurch verringert sich die Länge der Zugfolgen durchschnittlich um ein Viertel (nachher nur noch  $\frac{76}{100}$  der Originallänge). Dieser Wert stammt aus einem Test mit 1 000 verschiedenen zufälligen Zugfolgen der Länge 100.

### Beispiel

Am Anfang haben wir die Züge `R L' F U D2 U`. Dann werden die Züge nacheinander in Gruppen eingeteilt:

1. `{R: 1, L: -1}`
2. `{F: 1}`
3. `{U: 2, D: 2}`

Dann werden die Gruppen wieder zusammengefügt:

Die erste Gruppe hat ein R mit `Anzahl Rotationen mod 4 = 0`, daher wird kein Modifikator angehängen. Beim L ist der Modulo 4 aber 3, daher wird ein `'` angehängen. Für die anderen Gruppen wird genauso verfahren.

Wenn man diese Züge aneinanderfügt, erhält man `R L' F U2 D2`, das ist einen Zug kürzer als am Anfang. Dieser ist aber nicht verschwunden, die zwei Us wurden zu einem `U2`, weil kein Zug dazwischen war, der dies verhindert hätte.

## Farbkorrigierer

Da die Farben von der [Farberkennung](#) selten zu 100 % stimmen, werden sie nach dem Auswerten von einem Algorithmus korrigiert, wenn der erkannte Würfel nicht existieren kann. Im Laufe der Zeit sind drei verschiedene Korrektur-Algorithmen entstanden, von denen aber nur einer wirklich zuverlässig arbeitet. Der Vollständigkeit halber sind alle hier dokumentiert. Ich möchte keine Beschwerden über die sehr kreativen Namen hören!

Aktuell nutze ich den Neu-Algorithmus, da er am besten funktioniert und ich kaum Probleme habe. Die Anderen liefern zu oft falsche Würfel.

## Neu

Dieser Algorithmus korrigiert eine Farbe, in dem er prüft, welche Zu-Viel-Farbe am ehesten zu der Defizitfarbe passt. Das Wichtige daran ist, dass nur Möglichkeiten beachtet werden, bei denen ein möglicher Würfel herauskommt. Das führt dazu, dass es immer ein Ergebnis gibt, wenn nur eine Farbe falsch ist, aber auch, dass es möglich ist, dass ein Würfel herauskommt, der zwar möglich ist, aber nicht der gewollte.

## Komplex

Dieser Algorithmus korrigiert die Farben, indem er je eine der überschüssigen Farben einer Defizitfarbe zuordnet.

Als Erstes werden die Positionen der Farben, die zu häufig vorkommen, gespeichert, ebenso wie die Farben, die es zu wenig gibt (Defizitfarben), und die Anzahl, die sie mehr vorkommen müsste.

Dann wird für jede Position der Farben, die es zu viel gibt, die Wahrscheinlichkeit der Zugehörigkeit zu den Defizitfarben berechnet.

Als Erstes wird die Zu-Viel-Farbe genommen, die die größte Wahrscheinlichkeit hat, zu einer der Defizitfarben zu passen, und, sortiert nach Wahrscheinlichkeit der Zugehörigkeit zur Defizitfarbe, alle Wahrscheinlichkeiten durchgegangen. Wenn die Farbe zu einer Defizitfarbe passt und die Defizitfarbe immer noch eine ist (nicht schon vorher aufgefüllt wurde), wird das Feld zu der Defizitfarbe geändert und die Bestände aktualisiert.

Am Ende sollte es keine Defizitfarbe mehr geben. Da es immer noch die Wahrscheinlichkeit gibt, dass die Farben nicht stimmen, wird in einem solchen Fall der Würfel noch einmal eingescannt.

## Einfach

Hier wird wie in *Neu* eine Farbe korrigiert, nur, dass dabei nicht nur valide Würfel als Ergebnis möglich sind.