

Bipartiter Global-Curveball

Marius Hagemann

1 Einleitung

Bei der Analyse komplexer Netzwerke, wie beispielsweise soziale Netzwerke, werden die zugrundeliegenden Graphen häufig mit zufälligen Graphen verglichen, um deren Struktur zu untersuchen [?].

Zum Erzeugen von zufälligen Graphen existieren diverse Modelle wie beispielsweise der Erdős-Rényi-Graph [?] oder der Gilbert-Graph [?]. Diese Graphen weisen jedoch in der Regel kaum eine Ähnlichkeit zu dem zu analysierenden Netzwerk auf. Deshalb verwendet man Zufallsgraphen, die zu einem gegebenen Graphen eine identische Gradsequenz besitzen. Im Zufallsgraph soll also jeder Knoten denselben Grad haben wie im originalen Graph.

Algorithmen, welche diese Eigenschaft erfüllen, sind beispielsweise Curveball [?] und Global-Curveball.

Ziel dieser Arbeit ist die Anpassung von Global-Curveball an bipartite Graphen zur Reduzierung der Laufzeit.

2 Grundlagen

Curveball ist ein Prozess, bei dem Kanten zufällig getauscht werden. Bei einem Curveball-Tausch werden zwei verschiedene Knoten u und v ($u \neq v$) zufällig uniform verteilt ausgewählt und deren disjunkte Nachbarschaft zufällig durchmischt. Dabei muss darauf geachtet werden, dass durch das Tauschen keine Eigenschleifen oder Multikanten entstehen und dass die Bipartitheit nicht verletzt wird. Ein Beispiel ist in Abbildung 1 gegeben.

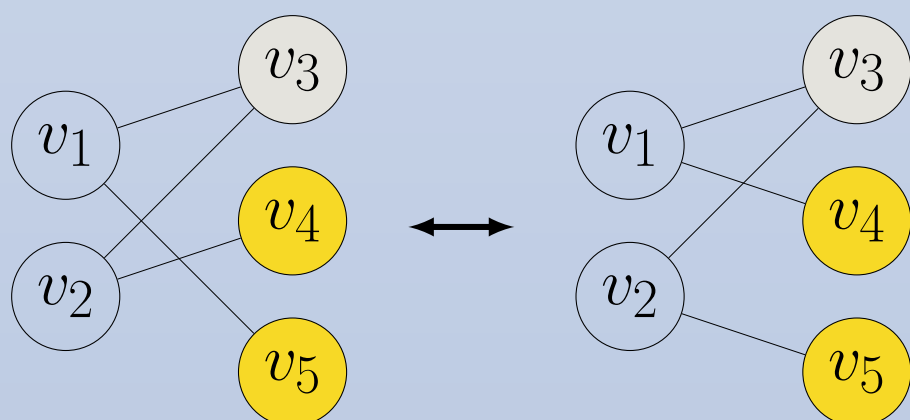


Abbildung 1: Auf einem der beiden Graphen wird ein Curveball-Tausch auf den Knoten v_1 und v_2 ausgeführt. In der grau markierten gemeinsamen Nachbarschaft liegt v_3 , die disjunkte Nachbarschaft $\{v_4, v_5\}$ ist in gelber Farbe gekennzeichnet. In diesem Beispiel gibt es nur die zwei dargestellten Graphen, die durch Tauschen der disjunkten Nachbarschaft entstehen können. Ein Curveball-Tausch würde jeweils mit Wahrscheinlichkeit 0.5 einen der beiden Graphen zurückgeben.

Ein **Global-Curveball Tausch** besteht aus mehreren Curveball-Tauschen, wobei möglichst jeder Knoten Teil eines solchen Curveball-Tausches sein soll.

Im Fall von bipartiten Graphen werden die Curveball-Tausche lediglich auf den Knoten aus einer der beiden Partitionsklassen ausgeführt. Diese wird als **aktive** Partition bezeichnet. Somit wird verhindert, dass durch einen Global-Curveball Tausch Eigenschleifen entstehen oder die Bipartitheit verletzt wird. Auch werden dadurch unnötige Curveball-Tausche verhindert.

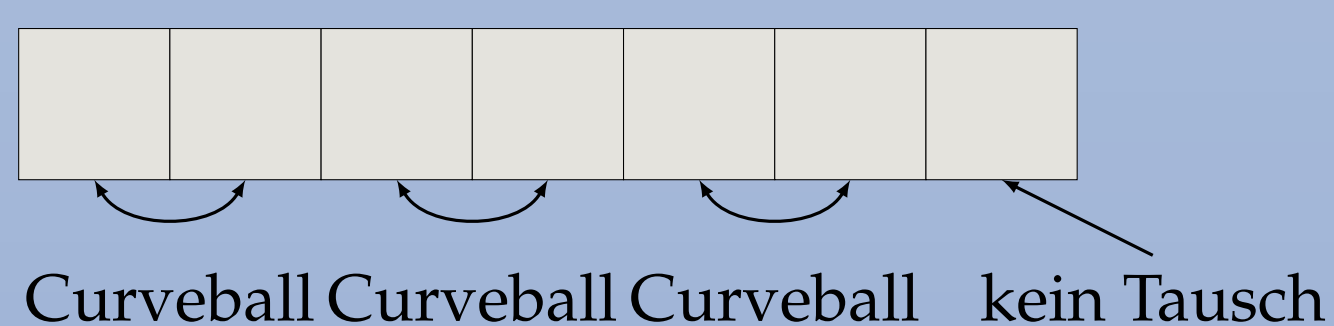


Abbildung 2: Global-Curveball auf dem zufällig permutierten Array der aktiven Partition. Da die Anzahl der Elemente in der aktiven Permutation ungerade ist, wird auf einem Knoten kein Curveball-Tausch ausgeführt.

3 Beschreibung

Für einen Curveball-Tausch muss also die gemeinsame und disjunkte Nachbarschaft bestimmt und die disjunkte Nachbarschaft zufällig durchmischt werden. Die Nachbarschaften

eines jeden Knotens sind jeweils in einem Array gespeichert.

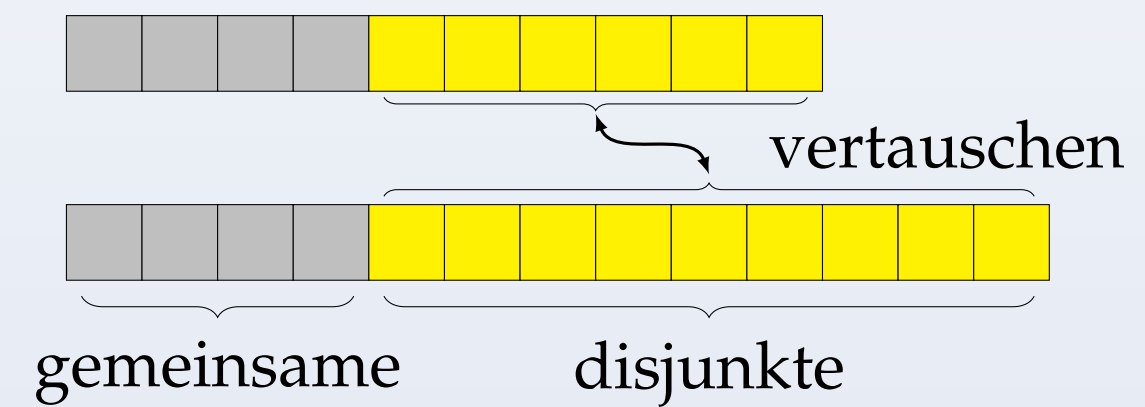


Abbildung 3: Skizze eines Curveball-Tausches auf den Arrays

4 Grundlagen

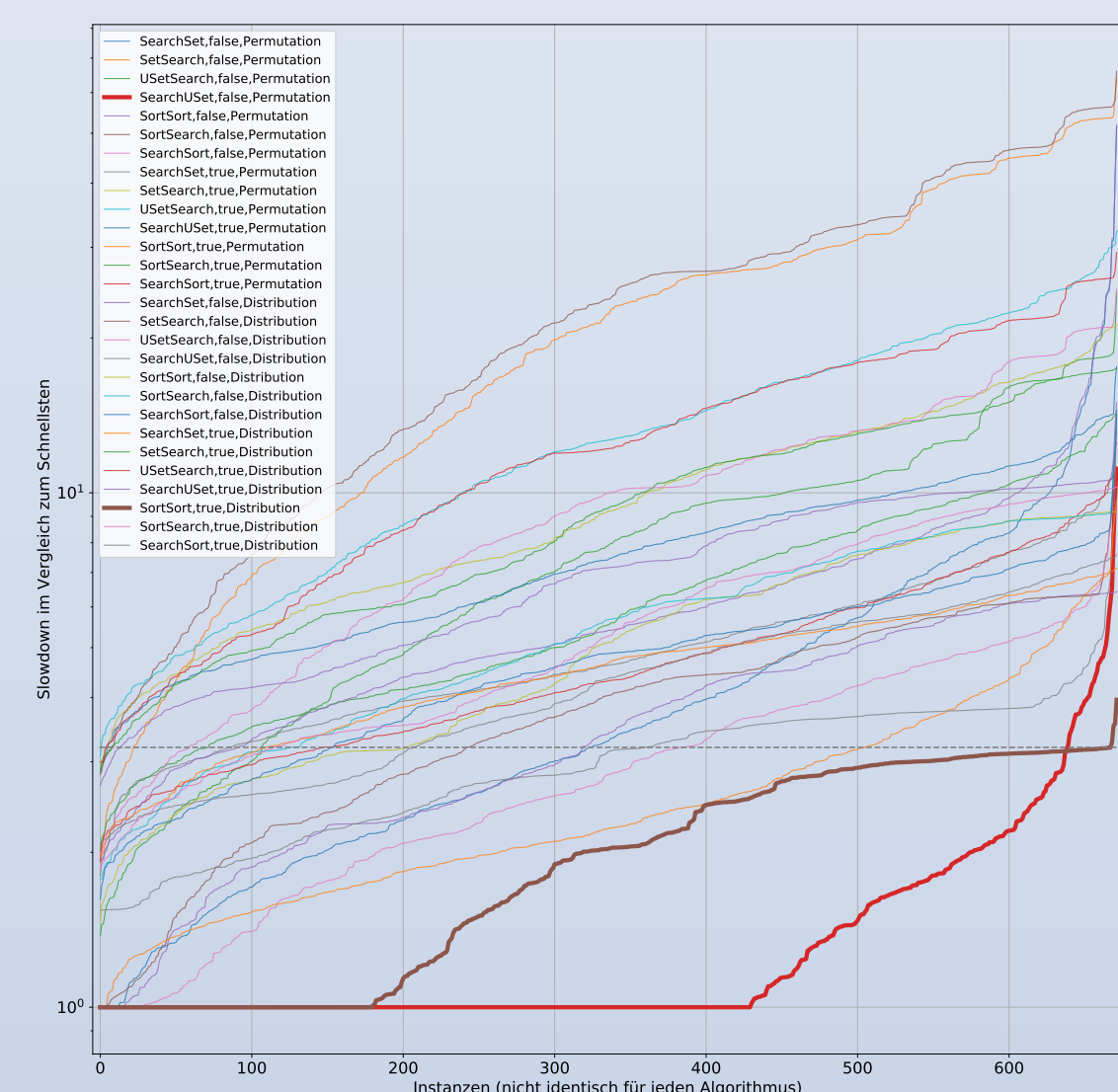


Abbildung 4: Slowdown der einzelnen Varianten im jeweiligen Vergleich zur Variante mit der geringsten Laufzeit. Dabei ist der Wert 3.5 als gestrichelte Linie eingezeichnet

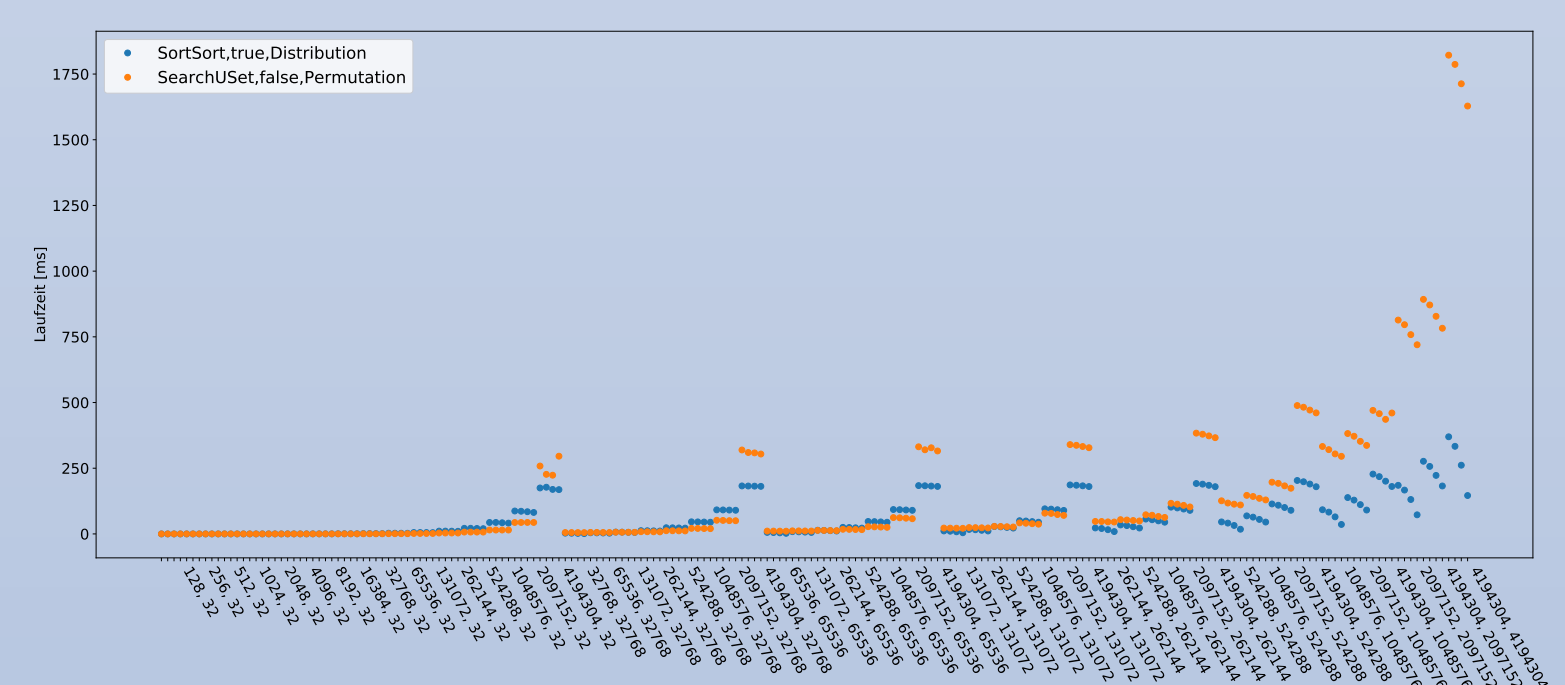


Abbildung 5: Vergleich der Laufzeiten der zwei besten Varianten auf ausgewählten Instanzen. Die Instanzen sind nach aufsteigenden Werten für small sortiert. Aus Gründen der Übersichtlichkeit wird bei der Beschriftung der Instanzen der Teil fraction weggelassen, die Instanzen werden nur mit large, small bezeichnet. Weiterhin wurden der Übersichtlichkeit wegen die Instanzen aus dem Bereich $32 < \text{small} < 32768$ ausgelassen, da sie das gleiche Bild wie die üblichen Instanzen zeigen.

5 blabla

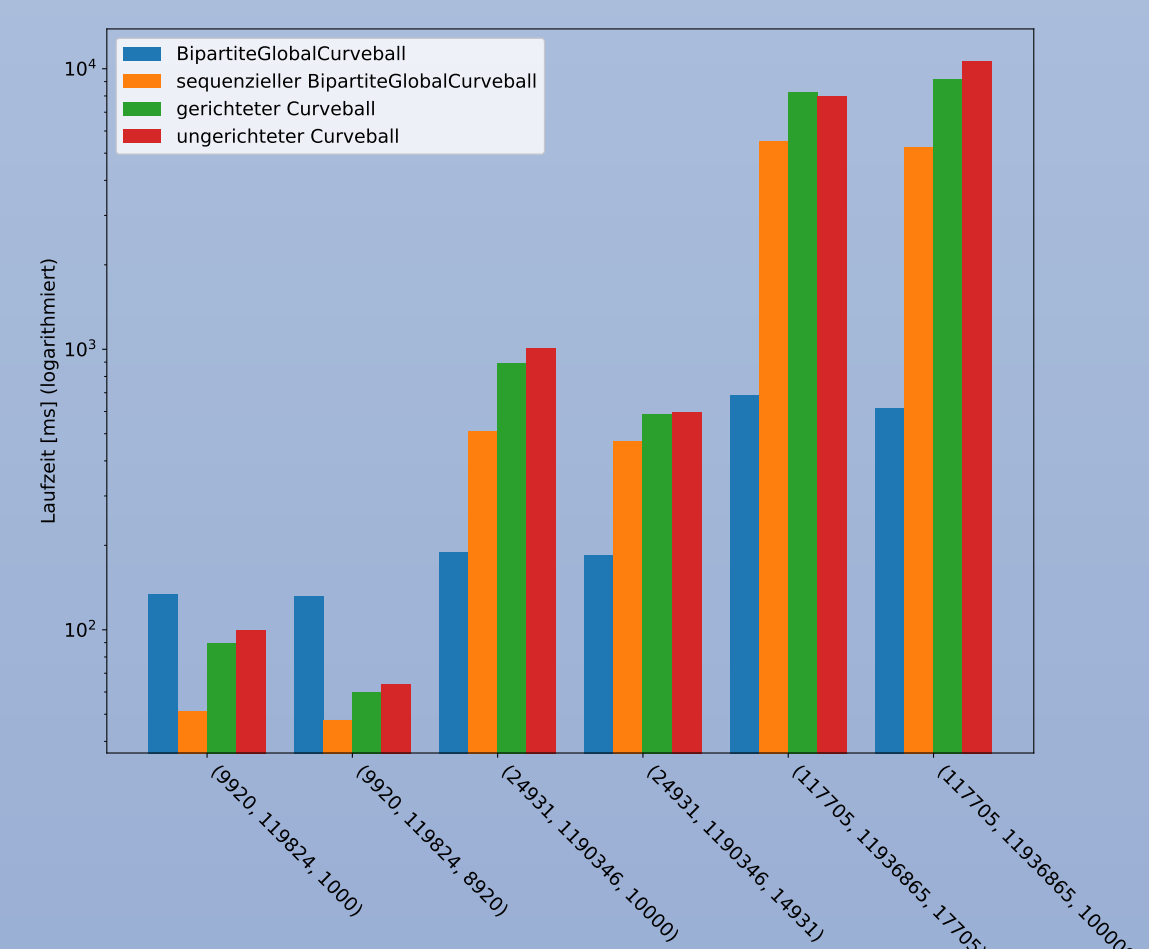


Abbildung 6: Laufzeitvergleich von bipartitem Global-Curveball und der abgeänderten Version des Curveballs. Auf der horizontalen Achse sind die einzelnen Instanzen als Tripel aufgetragen. Die erste Stelle steht dabei für die Knotenanzahl des Graphen, die zweite Stelle für die Anzahl an Kanten und die dritte für die Anzahl der Knoten aus der aktiven Partition.