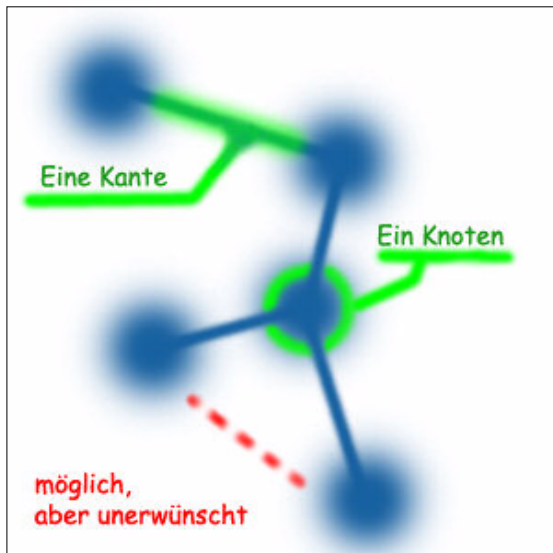


MINIMUM SPANNING TREES MIT KRUSKALS ALGORITHMUS



Um einen Minimum Spanning Tree (MST – zu Deutsch „minimaler aufspannender Baum“) wachsen zu lassen, benötigt man zuerst das Gewicht der einzelnen Kanten. Dies ist keine vordefinierte Einheit wie das reguläre Gewicht, sondern kann nach Bedarf zurechtgelegt werden; So kann man beispielsweise die Länge der Kante, Materialverbrauch beim Bau der Kante oder Dauer der Zeit, die man braucht, um sie zu überqueren, darauf abbilden. Die Aufgabe des MST ist es, *alle* Knoten eines Graphen miteinander zu verbinden, und zwar mit einem möglichst geringen Gesamtgewicht der verwendeten Kanten. Ein Informatiker namens Joseph B. Kruskal hat einen geeigneten Algorithmus für diese Aufgabe entwickelt, der hier vorgestellt werden soll. Ein Algorithmus ist eine Abfolge von Anweisungen, die mit einem bestimmten Eingabewert – in diesem Fall Knoten sowie nutzbare Kanten – eine Folge von Berechnungen anstellt, um einen Ausgabewert zu produzieren, der für weitere Anwendung benutzt werden kann; hier ist dies ein

Graph in Form eines MST. Der Algorithmus wird in Form von Pseudocode dargestellt, d.h. mit Anweisungen die so auch an den Computer gestellt werden, hier allerdings geändert in normale Schriftsprache.

MST-KRUSKAL

- 1 $A \leftarrow \emptyset$
- 2 Erstelle für jeden im Graphen vorkommenden Knoten ein eigenes Set
- 3 Sortiere alle im Graphen vorkommenden Kanten nach Gewicht, in aufsteigender Reihenfolge
- 4 Führe für jede dieser Kanten, in aufsteigender Reihenfolge, diese Anweisungen aus:
 - 5 Wenn $\text{CHECK-SET}(\text{Anfangsknoten}) \neq \text{CHECK-SET}(\text{Endknoten})$:
 - 6 Dann $A \leftarrow A \cup \{(\text{Anfangsknoten}, \text{Endknoten})\}$
 - 7 $\text{VERBINDE}(\text{Anfangsknoten}, \text{Endknoten})$
- 8 Gib A als Ausgabewert zurück

Der erwünschte MST wird hier in A gespeichert (A kann als Variable betrachtet werden, so wie z.B. ein x). Er ist ein Verbund von Kanten, die zusammen den MST bilden. Zeile 1 weist A die leere Menge zu, d.h. es gibt noch keinen MST. In Zeile 2 wird für jeden Knoten ein eigenes Set erstellt; Sets sind praktisch nichts anderes als Mengen, sie werden hier benutzt um zu zeigen welche Knoten in irgendeiner Weise bereits miteinander verbunden sind. Das Set $\{u; x; v\}$ würde zum Beispiel bedeuten, daß zwischen den Knoten u, x und v irgendwelche Kanten verlaufen. Es würde aber auch bedeuten, daß diese Knoten *nicht* mit irgendwelchen anderen Knoten verbunden sind. Diese Schreibweise ist also ausschließlich, ein Knoten kann immer nur in *genau einem* Set enthalten sein. In Zeile 3 werden alle Kanten nach Gewicht sortiert, damit man im Endeffekt einen *minimalen* Spannungsbaum erhält; wenn sie unsortiert wären, würde der nun folgende Part nichts bringen. Und zwar werden die Kanten jetzt (in den Zeilen 4-7) der Reihe nach zu A hinzugefügt, von klein nach groß. Dies geschieht in Zeile 6. Eine Zeile davor muß die Kante aber noch einen Sicherheitscheck durchlaufen, der zeigt, ob sie wirklich hinzugefügt werden darf; gefordert ist ja, daß nur *wirklich nötige* Kanten zu A, also dem MST hinzugefügt werden. Dazu wird nun die Funktion $\text{CHECK-SET}(\text{Irgendein Knoten})$ benutzt; sie gibt für einen beliebigen Knoten die Nummer des Sets zurück zu dem er gehört. Wenn der Algorithmus also eine beliebige Kante mit einem Anfangs- und einem Endknoten untersucht, so vergleicht Zeile 5 einfach die Nummern der Sets, in denen sich die beiden Knoten befinden. Ist die Nummer gleich, so sind beide Knoten im selben Set; das bedeutet, daß sie bereits verbunden sind und eine weitere Kante zwischen ihnen beiden sinnlos wäre. In diesem Fall passiert nichts, und der Algorithmus geht zur Kante mit dem nächsthöheren Gewicht. Ansonsten werden aber die Anweisungen in 6 und 7 ausgeführt, wobei 6 den MST um die untersuchte Kante erweitert und 7 beide Knoten in einem neuen Set zusammenfaßt; werden sie später untersucht, erscheinen sie dann als bereits verbunden. Die letzte Zeile gibt A als Ausgabewert wieder, das ist der nun fertige MST.

Aufgaben:

- 1) Zeichne einen eigenen MST mit insgesamt 5 Knoten und 8 möglichen Kanten, sowie dem Gewicht der einzelnen Kanten (frei wählbar). Trage die vom Algorithmus gewählten Kanten ein.
- 2) Wann werden mehr Kanten zu A hinzugefügt: Gegen Anfang oder Ende der Ausführung? Warum?
- 3) Erkläre die elementare Bedeutung von Zeile 3 für den Algorithmus in ein paar Sätzen.
- 4) Bei Interesse: Besuch <http://www.ics.uci.edu/~eppstein/gina/mst.html> (praktische Anwendungen für MSTs in Wissenschaft und Technik) oder lies „Introduction to Algorithms“ (Thomas H. Cormen u.a.).

Dieses Referat online und in Farbe: http://www.geocities.com/cooler_name/Referat.pdf