

# Lösungsstrategien für NP-schwere Probleme

## Blatt 8

Jakob Rieck  
6423721

Konstantin Kobs  
6414943

Thomas Maier  
6319878

Tom Petersen  
6359640

Abgabe zum 13.06.16

### Aufgabe 1

### Aufgabe 2

Für die Lösung des Problems nutzen wir einen einfachen Greedy-Algorithmus. Dieser nimmt iterativ ein Tripel aus  $T$  hinzu, und entfernt alle dazu in Konflikt stehenden Tripel aus dem Pool der möglichen Tripel. „In Konflikt stehen“ heißt hier, dass die erste, zweite oder dritte Stelle eines Tripels mit dem benutzten Tripel übereinstimmt. Das Ziehen geschieht iterativ, bis  $T$  leer ist.

Dieser Algorithmus funktioniert wegen der Tatsache, dass ein Algorithmus, der perfekt Kanten des maximalen Matchings auswählt, im Vergleich zu unserem Algorithmus maximal drei Tripel pro Iteration auswählen kann. Wird vom Greedy-Algorithmus ein Tripel ausgewählt, so fallen alle damit „verbundenen“ Kanten aus der nächsten Auswahl weg. Ein Algorithmus für das maximale Matching könnte aus diesen wegfallenden und dem ausgewählten Tripel maximal drei Tripel für das maximale Matching auswählen, die nicht in mindestens einer Stelle übereinstimmen. Danach fallen die gegebenen Kanten wiederum weg. Es kann also maximal so viele Iterationen geben wie mit dem Greedy-Algorithmus. Somit gilt, dass die Anzahl der Tripel im Greedy-Ansatz mindestens ein Drittel der Anzahl der Tripel im Maximal-Ansatz ist. Der vorgeschlagene Algorithmus erfüllt also die Anforderungen.