Mathematische Lösung

Bei der Lösung eines linearen Assignment Problems werden mit den OR-Tools immer dieselben Schritte vollführt:

1. Definieren der Parameter der Zielfunktion
2. Definieren der Constrains
3. Definieren einer Zielfunktion

Mit diesen Vorgaben können zwei Lösungen generiert werden. Entweder werden Parameter für eine minimale Lösung der Zielfunktion unter Berücksichtigung der Constrains gefunden oder nicht (siehe Kapitel {nummer} ungeeignete constrains).

Ausgangssituation

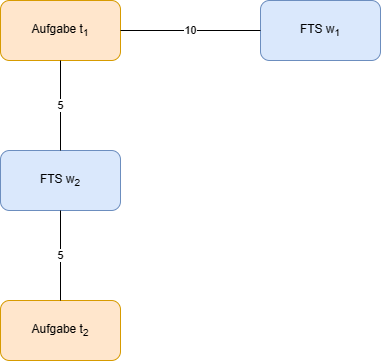
Wie schon in Kapitel {nummer} erläutert, sind auf einem Firmengelände mehrere FTS an verschiedenen Standpunkten und müssen Tasks zugewiesen werden, zu den sie fahren müssen. Dabei sind die Kosten die Zeit, die die FTS benötigen um zu einer Task zu fahren (siehe ).

Abbildung 1: Darstellung eines Graphen von verteilten FTS und Tasks auf einem Firmengelände mit Fahrzeit in Sekunden

Aus diesem Graphen (siehe ) lässt sich eine Kostentabelle erstellen, die beschreibt, welches FTS wie lange zu welcher Task braucht (siehe ).

|  |  |  |
| --- | --- | --- |
|  | Task t1 | Task t2 |
| FTS w1 | 10 | 20 |
| FTS w2 | 5 | 5 |

Tabelle 1: Kosten Tabelle des Graphen in Abbildung 1

Definieren der Parameter der Zielfunktion

Die Anzahl der Parameter ist sowohl abhängig von der Anzahl der FTS als auch der Anzahl der Tasks. Für jedes FTS / Task Paar wird ein Parameter definiert (siehe Formel 1).

1

Zusätzlich stellt man die Nebenbedingung für jeden Parameter, dass er nur zwischen null und eins liegen darf, als auch dass es nur ein Natürliche Zahl, inklusive null, sein darf (siehe Formel 2). Wenn OR-Tools als Lösung einen Parameter auf eins setzt wurde das FTS der Task zugewiesen und bei null nicht.

2

Für müssen folgende Parameter erstellt werden (zur Vereinfachung wird für jedes FTS (w) / Task (t) Paar der Parametername witj verwendet):

Beispiel 1: Parameter zu Abbildung 1

Definieren der Nebenbedingungen

Es wird für jedes FTS und für jede Task eine Nebenbedingung definiert (siehe Formel 3). Dabei wird jeder Parameter summiert, der durch das FTS bzw. der Task entstanden ist. Die obere Grenze wird immer auf eins gesetzt, während die untere Grenze abhängig von der Anzahl der FTS und Task ist und entweder auf null oder eins gesetzt wird. Die untere Grenze der Nebenbedingungen der FTS wird auf eins gesetzt, wenn es weniger FTS als Tasks gibt, weil jedem FTS eine Task zugewiesen werden kann, aber nicht jeder Task ein FTS (siehe Formel 4). Umgekehrt gilt dasselbe. Die untere Grenze der Nebenbedingungen der Tasks wird auf eins gesetzt, wenn es weniger Tasks als FTS gibt und auf null, wenn es mehr Tasks als FTS gibt (siehe Formel 5). Bei gleicher Anzahl an FTS und Tasks werden die unteren Grenzen aller Ungleichungen auf eins gesetzt, weil jedem FTS eine Task und jeder Task ein FTS zugewiesen werden kann.   
Die Grenzen bestimmen bei den Ungleichungen für die Tasks, wie viele FTS zugewiesen werden können und bei den Ungleichungen für die FTS, wie viele Tasks dieses gleichzeitig bekommen kann. In den Vorgaben (siehe Kapitel x) wurde bereits definiert, dass jedes FTS nur eine Task erledigen kann und jede Task nur ein FTS benötigt, weshalb die obere Grenze immer eins ist.

3

Für jedes FTS (wi) mit der Anzahl von m FTS und n Tasks (tj) gilt:

Wenn :

Wenn :

4

Für jede Task (tj) mit der Anzahl von n Tasks und m FTS (wi) gilt:

Wenn :

Wenn :

5

Für müssen insgesamt 4 Nebenbedingungen erstellt werden, da es zwei FTS und zwei Tasks gibt (siehe ) und lassen sich vereinfacht tabellarisch darstellen (siehe ). Die untere Grenze ist dabei immer eins, weil es die gleiche Anzahl an FTS und Tasks sind.

FTS w1:

FTS w2:

Task t1:

Task t2:

Beispiel 2: Nebenbedingungen zu

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | Task t1 |  | Task t2 |  |  |
|  |  |  | 1 |  | 1 |  |  |
|  |  |  |  |  |  |  |  |
| FTS w1 | 1 |  |  | + |  |  | 1 |
|  |  |  | + |  | + |  |  |
| FTS w2 | 1 |  |  | + |  |  | 1 |
|  |  |  |  |  |  |  |  |
|  |  |  | 1 |  | 1 |  |  |

Tabelle 2: Nebenbedingungen als Tabelle von

Definieren einer Zielfunktion

Die Zielfunktion wird definiert als die Summe aller Parameter mit den jeweiligen Kosten, also die Dauer die das FTS zur Task braucht, als Koeffizient (siehe Formel 6).

6

Um eine optimale Lösung zu finden muss die Zielfunktion minimiert werden (siehe Formel 7). Es werden Parameter gesucht für die die Funktion den geringsten Wert zurück gibt.

7

Für die Situation in muss eine Zielfunktion mit 4 Parametern erstellt werden (siehe ). Die Kosten lassen sich aus der Kostentabelle (siehe ) entnehmen.

Beispiel 3: Zielfunktion zu Abbildung 1

Lösen der Zielfunktion

Um die minimale Lösung einer Zielfunktion zu ermitteln müssen alle gültigen Parameterwerte, unter Berücksichtigung der Nebenbedingungen, ausprobiert werden. Die Anzahl aller Möglichkeiten beträgt bei m FTS und n Tasks unter Berücksichtigung der Nebenbedingungen (Barot & Hromkovic, 2017, adaptiert):

8

In gibt es dementsprechend 2 Möglichkeiten (siehe ). Entweder bekommt FTS w1 Task t1 und FTS w2 Task t2 oder FTS w1 Task t2 und FTS w2 Task t1. Die tatsächliche Lösung, welcher Wert für welchen Parameter eingesetzt wird damit ein minimales Ergebnis erzielt wird, wird von dem Framework OR-Tools berechnet. Die Dauer und Komplexität der Berechnung steigt mit der Anzahl der Möglichkeiten.

Beispiel 4: Anzahl der möglichen Aufteilung zu

Dieser Lösungsansatz hat den Vorteil das die Nebenbedingungen immer so definiert sind, dass es nie keine Lösung geben kann, weil das Problem dann der Variation ohne Wiederholung aus der Kombinatorik entspricht und nur noch die optimale Teilmenge ermittelt werden muss.

# Literaturverzeichnis

Barot, M. & Hromkovic, J., 2017. *Stochastik: diskrete Wahrscheinlichkeit und Kombinatorik.* Cham, Deutschland: Birkhäuser.