**Netzwerkflussproblem**

Von Alexander M. Westphal, Klaus Riedl, Robert Taglauer

Inhaltsverzeichnis

[Einleitung (Motivation/Aufgabenstellung, Ziele, Aufbau, Umfeld) 3](#_Toc12647689)

[Theoretische Grundlagen 3](#_Toc12647690)

[Was ist die Graphentheorie 3](#_Toc12647691)

[Was ist ein Netzwerk 4](#_Toc12647692)

[Praktischer Teil und das mathematische Modell 5](#_Toc12647693)

[Gegeben 5](#_Toc12647694)

[Gesucht 5](#_Toc12647695)

[Lösungsweg 6](#_Toc12647696)

[Der Netzwerkfluss mit Kapazitätsbeschränkung 6](#_Toc12647697)

[Zusammenfassung + Fazit + Ausblick 11](#_Toc12647698)

[Literaturverzeichnis 11](#_Toc12647699)

[Bücher und Journals 11](#_Toc12647700)

[Internetquellen 11](#_Toc12647701)

# Einleitung (Motivation/Aufgabenstellung, Ziele, Aufbau, Umfeld)

Man stelle sich folgendes Scenario vor: Sie sind als Logistiker in einem Unternehmen angestellt, der Netz AG. Die Netz AG produziert Waren in ihren Fertigungshallen. Nun besteht ihre Aufgabe als Logistiker darin einen geeigneten Weg von A nach B zu finden. Zwischen Punkt A und B befinden sich jedoch noch weitere Punkte an denen sie halt machen müssen, da zum Beispiel das Transportmittel gewechselt wird. Nun ist es so das verschiedene Transportmittel unterschiedlich viel Kosten verursachen.

Womit eine weitere Größe Einzug in unser Problem gefunden hat und es wird deutlicher welchen Schwierigkeiten sich ergeben. So wird weiterhin offensichtlicher das wir hier ein Optimierungsproblem lösen wollen. Da der Transport einer Ware von A nach B entweder zu maximaler Menge oder minimalen Kosten führen sollte. Diese Art von Problem kann auf verschiedenen Art und Weisen beschrieben werden. In letzter Konsequenz reduziert es sich dennoch immer auf die optimale Auslastung beziehungsweise Minimierung der Kosten oder Maximierung der transportieren Menge. Deshalb werden wir im Folgenden näher betrachten wie es möglich ist eine optimale Verteilung beziehungsweise Auslastung unseres Netzwerks zu finden. Hierzu wird im Folgenden eine Vertiefung des zuvor beschriebenen Problems stattfinden sowie eine mathematische Modellierung. Ebenfalls wird ein Beispiel Netzwerk in Python implementiert und näher erläutert.

# Theoretische Grundlagen

## Was ist die Graphentheorie

Um sich dem Problem des Netzwerkflusses anzunähern macht es Sinn zu verdeutlichen in welchem Bereich der Mathematik Netzwerke und Flüsse zu finden sind. Somit sollte zunächst einmal die Definition der Wortes Graphentheorie näher Betrachtet werden. Unter dem Terminus Graphentheorie versteht man ein Teilbereich der Mathematik welcher sich die Eigenschaften sowie die Beziehung von Graphen näher erschließt. Besonders hervorzuheben ist hierbei noch das jenes Themengebiet nicht nur in der Mathematik anzutreffen ist. Besonders in der Informatik findet sich das Thema Graphen beziehungsweise Algorithmen in Verbindung mit Graphen wieder und hat somit einen hohen Bezug zur Praxis.

## Was ist ein Netzwerk

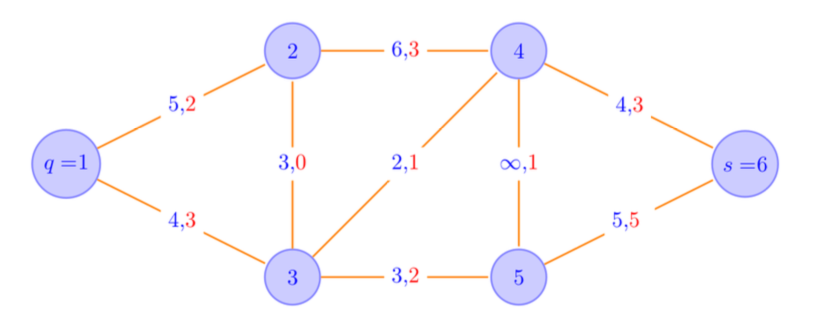
Nun nach dem zuvor verdeutlicht wurde was ein Graph ist sollte ebenfalls

# Praktischer Teil und das mathematische Modell

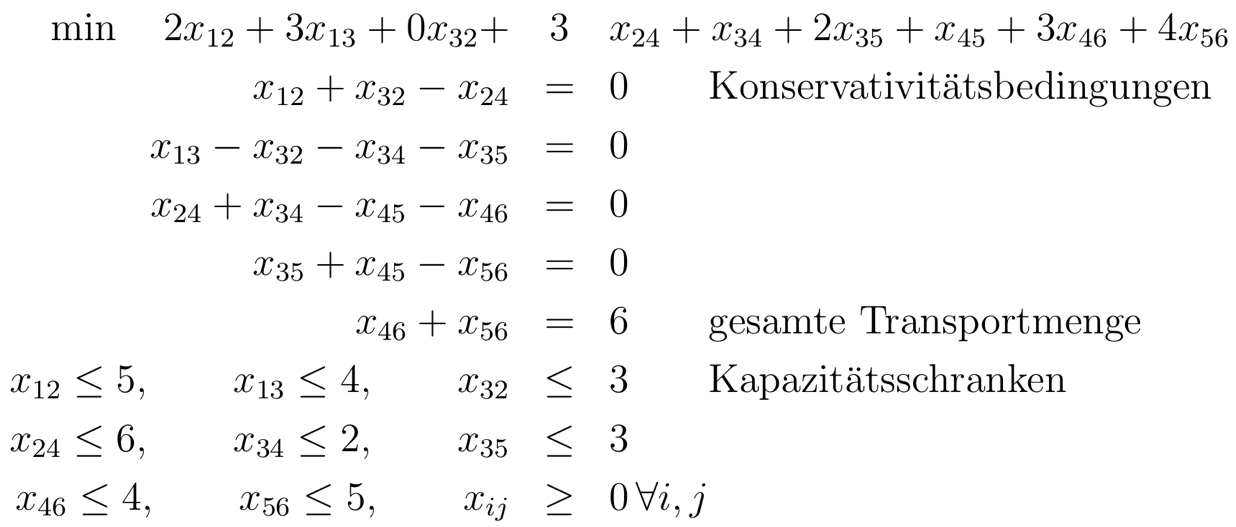
## Gegeben

* Zu optimieren:
  + Maximale Menge die Transportiert werden kann
  + Eine Bestimmte Menge soll zu minimalen Kosten transportiert werden
* Ein Beispiel mit folgenden Informationen
  + An den Kanten des Graphen sind Kapazität und Kosten gegeben (siehe Abbildung 1)
  + Ebenfalls wurde ein Lineares Gleichungssystem mit Nebenbedingungen gegeben (siehe Abbildung 2)

*Anmerkung: Im Folgenden sind alle Berechnungen und Beispiele wie auch konkreten Angaben auf Basis dieses Beispiels.*



1. Abbildung



1. Abbildung

## Gesucht

Gesucht war nun die maximale Flussmenge beziehungsweise die minimalen Kosten zu einer Bestimmten Menge an Gütern. Somit sollten also zum einen, ein Minimierungsproblem und ein Maximierungsproblem gelöst werden.

## Lösungsweg

Mathematische Bedingungen:

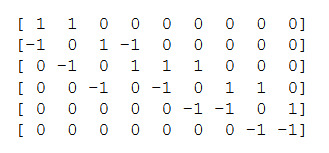
Nun mehr haben wir für unser Problem verschiedene Randbedingungen aufgestellt, die in jedem Fall erfüllt sein müssen, damit unser Problem lösbar ist.

1. Die Menge, die durch Rohre geschickt wird, kann nicht kleiner als 0 sein.
2. Die Menge, kann nicht größer sein als die maximale Transport Kapazität.
3. Die Menge, die ein Punkt bekommt, muss gleich sein der Menge, die er verschickt (Ausnahme Quelle und Ziel)

**Also wie folgt:**

## Der Netzwerkfluss mit Kapazitätsbeschränkung

Zunächst brauchen wir für unser Mathematisches Modell eine Inzidenzmatirx. Unter einer Inzidenzmatrix versteht sich eine besondere Form einer Matrix, die in er Graphen Theorie üblich ist. Auch Knoten-Kanten-Matrix genannt, besteht ihr Verwendungszweck darin die Beziehung zwischen Knoten und Kanten in mathematischer Form darzustellen. So ergibt sich folgende Bild der Matrix in unserem Netzwerkflussproblem.



Hierbei ist gut zuerkennen das lediglich drei Werte in der Matrix gespeichert werden können.

*So gilt folgendes für einen Schleifenfreien und gerichteten Graphen:*

Aus dieser Feststellung folgt nun die Erkenntnis das zwischen der Strecke A zu B oder B zu A keine Verbindung besteht, wenn der Wert in der Matrix 0 beträgt. Hingegen sollte der Wert 1 sein so besteht eine Verbindung von A nach B und zwar gerichtet in Richtung B. Hierbei stellt -1 lediglich das Kompliment da, also von B nach A. Ebenfalls

Nun müssen als nächstes unsere Nebenbedingungen aufgestellt werden beziehungsweise wir übernehmen einen Teil aus dem zu Beginn geschilderten Abschnitt mathematische Bedingungen.

B sei hier unser Kontrollvektor. Mit jenem ist es uns möglich zu bestimmen in Abhängigkeit von A - also dem Wert unserer Quelle – was B als Ausgang haben muss beziehungsweise sicherzustellen, dass diese gleich sind.

Nun sind wir in der Lage unser Problem zu minimieren beziehungsweise die niedrigsten Kosten zu berechnen. Hierzu nutzen wir folgende Summenformel:

Hierbei repräsentiert ji ein Element so das wir alle möglichen Kombinationen erfassen. Beziehungsweise alle möglichen Kanten. So ergibt sich im Folgenden ein Netzwerkflussproblem der Form:

Lineare Optimierung:

Um nun zu einer Lösung zu kommen müssen wir di von uns zuvor Aufgestellten Gleichungen, unter Berücksichtigung der Nebenbedingungen Lösen. Hierzu haben wir uns des Models SciPy (Scientific Computing Tools for Python) bedient. In jenem Modul ist die Funktion *„LinProg“* enthalten. Diese können wir nutzen um unsere Gleichungen zu lösen.

**“ scipy.optimize.linprog(**c, A\_ub=None, b\_ub=None, A\_eq=None, b\_eq=None, bounds=None**) “**

Hierzu geben wir der Funktion lediglich unsere Werte mit uns bekommen ein optimales Ergebnis unter Berücksichtigung der Nebenbedingungen und aller Gleichungen. **A\_ub** steht für unsere Inzidenzmatrix, **b\_ub** für unseren Kontrollvektor, **c** für unseren Kostenvektor der Optimiert werden soll und **Bounds** für **Lowerbound** (Nullvektor) und **Upperbound** (Vektor mit unseren Kapazitäten).

# 

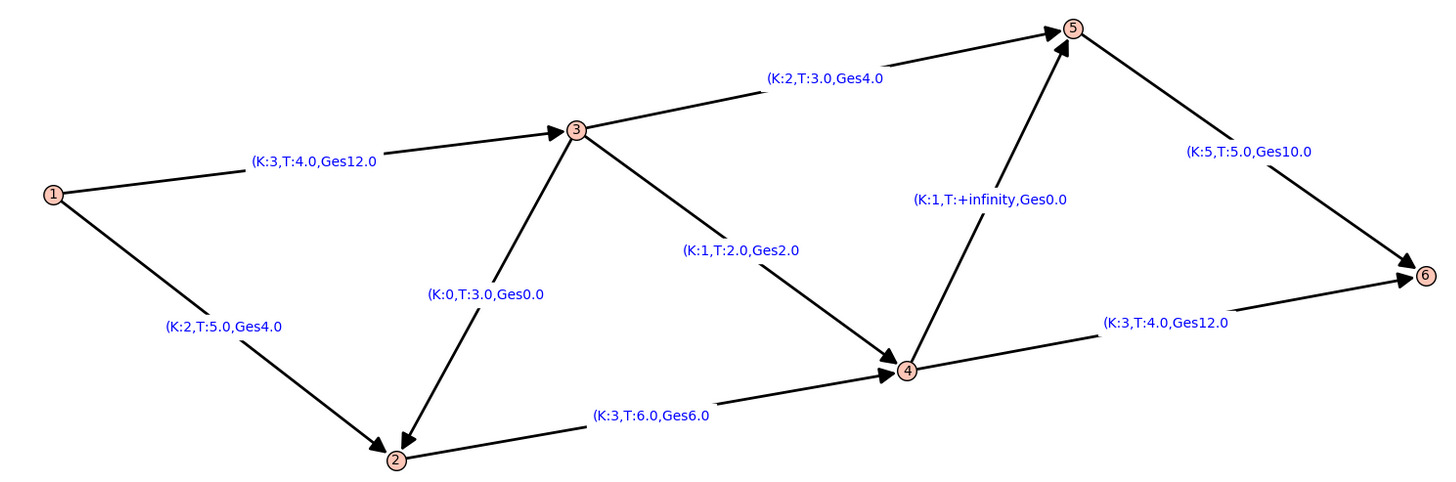
Ergebnis:

Somit ergibt sich als Lösung für unser Beispiel bei einer Transportmenge von 6 und einer Empfangsmenge von 6

Lösung:

Wenn wir eine Menge von 6 durch unseren Netzwerkfluss schicken wollen und auch wieder 6 an der Sekante ankommen sollen haben, ist der günstigste Weg insgesamt 50 Kosten teuer mit der folgenden Lösung.

x1,2=2|x1,3=4|x2,4=2|x3,2=0|x3,4=2|x3,5=2|x4,5=0|x4,6=4|x5,6=2



**2.1 Transportproblem: Ohne Kapazitätsbeschrenkung**

Ohne Kapazitätsbeschrenkung suchen wir den Günstigsten Pfad. Hierzu haben wir uns überlegt das wir diesen Pfad bekommen indem wir einfach davon ausgehen das wir in Unseren gegebenen Netzwerkfluss nur die Menge 1 von der quelle zur Sekante transportieren und dies wie in 2.0 beschrieben auflösen mit der Änderung das man alle Einträge des Kapazitätsvektors mit einer 1 ersetzt und somit sicherstellt das die Menge 1 theoretisch über jede Kante kann.

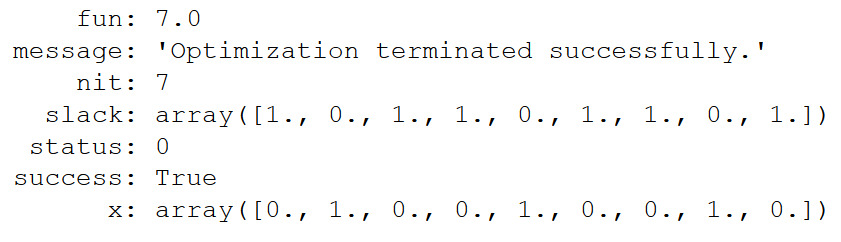
scipy.optimize.linprog(c, A\_ub=None, b\_ub=None, A\_eq=None, b\_eq=None, bounds=None)

c = Vektor von unseren Kosten die wir Optimieren wollen

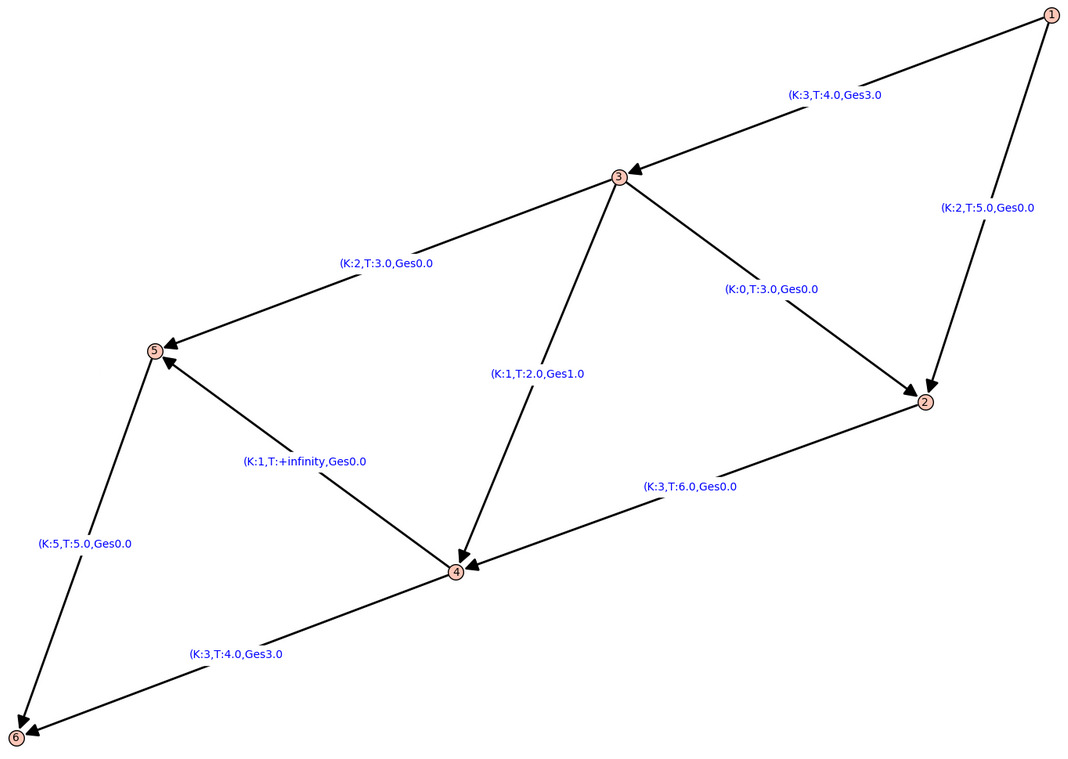
A\_eq=Inzidenzmatrix

b\_eq=Control Vektor (Hier immer ersten Eintrag 1 und letzten Eintrag -1)

bounds = lowerbound -> nullvektor, upperbound->Vektor voll mit einsern



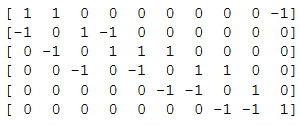
Der günstige Weg verursacht pro Menge Kosten von 7 dank der folgenden Lösung.

x1,2=0|x1,3=1|x2,4=0|x3,2=0|x3,4=1|x3,5=0|x4,5=0|x4,6=1|x5,6=0

**3.0 Maximalerfluss:**

Überlegung ob man ein Maxfluss Problem auch als Kostenminimaler Fluss Problem umschreiben kann.

Als erstes haben wir und überlegt einen Kante von Senke zu Quelle zu machen damit man einen unendlichen Input von der Quelle hat. Hierbei muss man nun auch die Inzidenzmatrix anpassen. Die Kosten werden auf null gesetzt.



Normal hätte man nun als Bedingung das wir den x6,1 maximieren, da wir dann jedoch versuchen wollen unser Problem umzuschreiben dachten wir das wir jetzt pro Mengen Übertragung nicht ins positive sondern ins negative gehen. Somit können wir den Wert der von der Sekante zum Quellen Input kommt und zurück zur Sekante damit Maximieren wenn wir –(x6,1) Minimieren.

3.1 Mathematische Bedingungen:

* Transportproblem:
  + Menge, die durch Rohre geschickt wird, kann nicht kleiner als 0 sein.
  + Menge, kann nicht größer sein als die maximale Transport Kapazität.
  + Menge, die von der Sekante zur Quelle kommt muss gleich sein wie die von der Quelle zur Sekante
* Bildung Nebenbedingungen:

Unsere Nebenbedingungen ergeben Sich aus 3.1

* + 0 ≤ x
  + x ≥ u |u= maximale Transport Kapazität
  + Ax=b |b= nullvektor um sicher zu stellen das auch das was von der Sekante zur

Quelle fließt auch wieder bei der Sekante ankommt

Nun haben wir wieder die gleiche Form wie beim Transportkostenproblem.



Nun können wir wieder die linprog Funktion verwenden.

scipy.optimize.linprog(c, A\_ub=None, b\_ub=None, A\_eq=None, b\_eq=None, bounds=None)

c = nullvektor bei dem der letzte Eintrag -1 ist (x6,1)

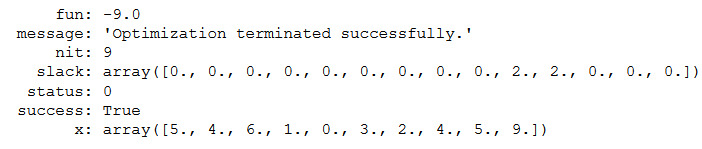
A\_eq=Inzidenzmatrix

b\_eq=Control Vektor (hier jetzt ein nullvektor-> da wir keinen festen input haben )

bounds =

lowerbound -> nullvektor mit letzten Eintrag von -np.inf (wg. Sekante zu Quelle )

upperbound->Vektor den Transportkapazitäten



Hierbei ist unser Ergebnis negativ da wir nicht nach x6,1 Maximiert haben sondern nach –(x6,1) Minimiert um nun das Korrekte Vorzeichen zu erhalten müssen wir fun\*-1 rechnen.

Den Maximalen Fluss von 9 erhalten wir mit der folgenden Lösung.

x1,2=5|x1,3=4|x2,4=6|x3,2=1|x3,4=0|x3,5=3|x4,5=2|x4,6=4|x5,6=5|

# Zusammenfassung + Fazit + Ausblick

# Literaturverzeichnis

## Bücher und Journals

## Internetquellen