Makroskopische Umlegung 2

„Stochastic User Equilibrium“

Simulation großer Verkehrsnetzwerke

- ✗ Aufwand zur Berechnung der einzelnen Fahrzeuge zu hoch
- ✗ Klar: Ergebnis der Simulation sollte trotzdem realitätsnah sein

Fluss in Netzwerken - Makroskopische Umlegung

- Verkehrsstrom als Fluss in Netzwerken
  - Neu: Abhängigkeit der (Fahr-)Zeit vom Fluss (Auslastung)
- Netzwerk strebt zum Equilibrium
  - pendelt sich auf optimales Gleichgewicht ein
  - Erklärung: Benutzer wählen wenn möglich schnellere/schneller empfundene Routen

Equilibrium – Einpendeln auf ein Gleichgewicht

- Abhängigkeit von Nachfrage, Wartezeit und Durchsatz
  - Tankstelle
    - Ankunftshäufigkeit, Abfertigung, Wartezeit
  - (minimales) Straßennetz
    - Benutzer, Passierdauer, Fahrzeugdurchsatz

Benutzergleichgewicht

- Gilt UE für ein Paar Quelle/Senke ist die Reisezeit auf allen benutzten Pfaden gleich und kleiner/gleich der Reisezeit auf einem ungenutzten Pfad bei minimalem Fluss.

- Folgerung:
  - Kein einzelner Benutzer kann seine Reisezeit durch Wechsel seiner Route verbessern:

    stabile und optimale Lösung

Mathematische Grundlagen

- Das Minimum der „Gesamtpassierdauer": $\min[z(x)]$: $\quad z(x)=\sum_a \int_0^{x_a} t_a(\omega)\,d\omega$
- Klar: $\sum_k f_k^{rs}=q_{rs} \quad \forall r,s \quad ; \quad f_k^{rs}\geq 0 \quad \forall k,r,s$

  $x_a=\sum_r \sum_s \sum_k f_k^{rs}\delta_{a,k}^{rs} \quad \forall a$

N Menge der Knoten
A Menge der Kanten
R Menge der Ursprungsknoten
D Menge der Zielknoten
$K_{r,s}$ Menge der Pfade die r und s verbinden; $r\in R, s\in D$
x_a Fluss über Kante a; $x=(\ldots, x_a, \ldots)$
t_a Reisezeit über Kante a; $t=(\ldots, t_a, \ldots)$
f_k^{rs} Fluss über Pfad k der r und s verbindet; $f^{rs}=(\ldots, f_k^{rs}, \ldots)$; $f=(\ldots, f^{rs}, \ldots)$
c_k^{rs} Reisezeit über Pfad k der r und s verbindet; $c^{rs}=(\ldots, c_k^{rs}, \ldots)$; $c=(\ldots, c^{rs}, \ldots)$
q_{rs} Nachfrage zwischen r und s
$\delta_{a,k}^{rs}$ charakteristische Funktion $\delta_{a,k}^{rs}=\begin{cases}1 & \text{für } a \text{ auf einem Pfad} \in K_{rs}\\ 0 & \text{sonst}\end{cases}$

Beispielnetzwerk

- Zwei Knoten mit zwei Kanten verbunden.

$$t_1 = 3 + 0{,}5\, x_1$$
$$t_2 = 1 + x_2$$
$$q = 1{,}5 = x_1 + x_2$$

$$\rightarrow \textit{offensichtlich wird Kante } 1 \textit{ nicht genutzt , da} \quad t_1(0) > t_2(q)$$
$$\rightarrow \textit{UE bei } x_2 = q \textit{ und } x_1 = 0 \,;\, t_2(x_2) = 2{,}5$$

$$I \;\;:\;\; t_1 = 2 + x_1$$
$$II \;:\;\; t_2 = 1 + 2\mathrm{x}_2$$
$$III :\;\; q = 5 = x_1 + x_2$$

$$\rightarrow t_1 = t_{2,}\, \textit{da offensichtlich beide Kanten genutzt werden}$$
$$\rightarrow 2 + x_1 = 1 + 2\mathrm{x}_2 \quad \rightarrow \quad x_1 = 2\mathrm{x}_2 - 1$$
$$\rightarrow \textit{zusammen mit III ergibt das } x_1 = 3, x_2 = 2 \textit{ und } t_1 = t_2 = 5$$
$$\textit{Vergleich mit Minimalisierungsansatz...}$$

Systemoptimierung

- Idee: Minimierung der Gesamtreisedauer im System

$$min[\tilde{z}(x)]: \quad \tilde{z}(x) = \sum_a x_a t_a(x_a)$$

- Benutzer können nur „gemeinsam" optimalen Zustand erreichen, führt nicht zwangsläufig zu Equilibrium.

  Ergebnis trotzdem interessant: optimale Lösung für das gesamte Verkehrsnetzwerk

Stochastic Network Loading

- ✗ Reisezeit eines Pfades ist nie genau bekannt

  deswegen:
  - ✗ Hinzunahme einer Zufallskomponente bei Routenwahl
  - ✗ Benutzer wählt Pfad mit **seiner Meinung nach** kürzester Reisezeit

Auswahl Modelle

Nützlichkeits Funktion

- Gibt Attraktivität einer Auswahl für einen Benutzer an
- Besteht aus deterministischer und zufälliger Komponente
- a enthält charakteristische Attribute

$$U_k(a) = \underbrace{V_k(a)}_{deterministisch} + \underbrace{\xi_k(a)}_{zufällig} \quad \forall k \in K$$

Auswahl Funktion

- Gibt Wahrscheinlichkeit an mit der Alternative k gewählt wird
- Kann abhängig von a sein
- Steht Verteilung von ξ fest, lässt sich P berechnen

$$P_k(a) = Pr\left[U_k(a) \geq U_l(a) \,\forall\, l \in K\right] \quad \forall\, k \in K$$

wobei gilt, dass

$$0 \leq P_k(a) \leq 1 \quad und$$

$$\sum_k P_k(a) = 1 \quad \forall\, k \in K$$

Das Multinomial Logit Modell

- Am meisten verwendetes Auswahl Modell
- Zufallskomponenten sind Gumbel Zufallsvariablen
- Die Auswahl Funktion ist dann:

$$P_k = \frac{e^{V_k}}{\sum_{l=1}^{K} e^{V_l}}$$

bzw

$$P_k = \frac{1}{1 + \sum_{l \neq k} e^{V_l - V_k}}$$

Die Routenwahl (1)

- Jeder Benutzer nimmt die Reisezeit der Pfade anders wahr
- Somit wählt ein Benutzer den Pfad, den er für den schnellsten hält
- Die wahrgenommene Reisezeit ist:

$$C_k^{rs} = \underbrace{c_k^{rs}}_{gemessen} + \underbrace{\varepsilon_k^{rs}}_{Fehlerterm} \quad \forall k, r, s$$

wobei gilt, dass

$$E[\varepsilon_k^{rs}] = 0 \quad bzw. \quad E[C_k^{rs}] = c_k^{rs}$$

Die Routenwahl (2)

- Für größere Benutzermengen ergibt sich somit:

$$P_k^{rs} = Pr\left(C_k^{rs} \leq C_l^{rs}, \forall l \in \boldsymbol{K}_{\boldsymbol{rs}}\right) \quad \forall k, r, s$$

- Zum Vergleich nochmal die allgemeine Auswahlfunktion abhängig von U

$$P_k(a) = Pr\left[U_k(a) \geq U_l(a) \forall l \in K\right] \quad \forall k \in K$$

- Somit ist $U_k^{rs} = -C_k^{rs}$

Die Routenwahl (3)

- Es gilt wieder: sobald Verteilung der wahrgenommenen Reisezeit feststeht, lassen sich die Flüsse über die Pfade berechnen

$$f_k^{rs} = q_{rs} P_k^{rs} \quad \forall k, r, s$$

- Der Fluss über eine Kante a ist dann:

$$x_a = \sum_{rs} \sum_k f_k^{rs} \delta_{a,k}^{rs} \quad \forall a$$

Logit basierte Belastungsmodelle (1)

- Von folgender Formel wird ausgegangen, wobei θ ein positiver Korrekturfaktor ist:

$$U_k^{rs} = -\theta\, c_k^{rs} + \xi^{rs} \quad \forall k, r, s$$

- Anmerkung: $\xi^{rs} = -\theta\, \varepsilon^{rs}$

$$C_k^{rs} = c_k^{rs} + \varepsilon_k^{rs} \qquad \forall k, r, s$$

$$C_k^{rs} = c_k^{rs} - \frac{1}{\theta}\xi^{rs} \quad \forall k, r, s$$

Logit basierte Belastungsmodelle (2)

- Auswahlfunktion lässt sich nun mit Hilfe der gemessenen Reisezeit berechnen

$$P_k = \frac{e^{V_k}}{\sum_{l=1}^{K} e^{V_l}} \quad \rightarrow \quad P_k^{rs} = \frac{e^{-\theta c_k^{rs}}}{\sum_l e^{-\theta c_l^{rs}}} \quad \forall k, r, s$$

- 2 Knoten A,B mit 2 Kanten 1,2 verbunden

$$t_1 = 1$$
$$t_2 = 3$$
$$\theta = 0{,}5$$
$$q_{AB} = 100$$

$$P_1^{AB} = 0{,}73 \qquad P_2^{AB} = 0{,}27$$
$$f_1^{AB} = 73 \qquad f_2^{AB} = 27$$

Stochastic User Equilibrium

- Bei SNL war Reisezeit über eine Kante unabhängig vom Fluss
- Im UE ist die wahrgenommene Reisezeit gleich der gemessenen
- SUE ist Verbindung von UE und SNL
- Wahrgenommene Reisezeit über eine Kante ist abhängig vom Fluss und enthält einen zufälligen Term

$$t_a = t_a(x_a) \quad und \quad t_a = E[T_a]$$

- Im SUE kann kein einzelner Benutzer durch Wechseln seiner Route seine **wahrgenommene** Reisezeit verbessern.

Gleichungen des SUE

1. $f_k^{rs} = q_{rs} P_k^{rs}(t_k) \qquad \forall k, r, s$
2. $t_a = t_a(x_a) \qquad \forall a$
3. $\sum_k f_k^{rs} = q_{rs} \qquad \forall r, s$

- Anmerkung:
  Ist t nicht abhängig von x, so lassen sich die Gleichungen 1 und 3 mit Hilfe von SNL bestimmen

Method of Successive Averages (1)

- Schritt 0: *Initialisierung*

- SNL wird basierend auf Fahrzeiten der unbelasteten Kanten ausgeführt. Somit entsteht eine Menge von Flüssen $\{x_a^1\}$ über die Kanten.

Method of Successive Averages (2)

- Schritt 1: *Updaten*

$$t_a^n = t_a(x_a^n), \forall a$$

- Schritt 2: *Richtung bestimmen*

- SNL wird auf Basis der aktuellen Fahrzeiten $\{t_a^n\}$ ausgeführt. Das Ergebnis ist eine Menge von Hilfsflüssen $\{y_a^n\}$

$$y_a^n = \sum_{rs} \sum_k q_{rs} P_k^{rs}(c^{rs^n}) \delta_{a,k}^{rs}$$

Method of Successive Averages (1)

- Schritt 3: *Schritt*

- Der neue Fluss wird berechnet mit:

$$x_a^{n+1} = x_a^n + \underbrace{\alpha_n}_{Schrittgröße} (y_a^n - x_a^n)$$

$$wobei \quad \alpha_n = \left(\frac{k_1}{k_2 + n}\right) \quad mit\ k_1 > 0, k_2 \geq 0$$

Method of Successive Averages (1)

- Schritt 4: *Konvergenz*

  Das Konvergenzkriterium wird überprüft. Wenn es nicht erfüllt ist, dann ist n := n+1 und es wird zu Schritt 1 gesprungen.

- Konvergenz kann auf mehrere Arten bestimmt werden
  1. Nach fester Anzahl von Durchläufen
  2. Mit Hilfe des durchschnittlichen Flusses der letzten m Durchläufe

$$\overline{x}_a^n = \frac{1}{m} \sum_{l=0}^{m-1} x_a^{n-l}$$

wobei m = 3 genügt

Method of Successive Averages (1)

- Schritt 4: *Konvergenz*

$$\overline{x}_a^n = \frac{1}{m} \sum_{l=0}^{m-1} x_a^{n-l}$$

Da $\overline{x}_a^n$ monoton fallend ist, kann nun ein spezielles Konvergenzkriterium verwendet werden

$$\frac{\sqrt{\sum_a \left(\overline{x}_a^{n+1} - \overline{x}_a^n\right)^2}}{\sum_a \overline{x}_a^n} \leq \kappa$$

- Beispielnetzwerk mit 2 Knoten A, B und 2 Kanten 1, 2 welche diese beiden Knoten verbinden

$$t_1 = 1 + x_1$$

$$t_2 = 3 + \frac{x_2}{2}$$

$$q = 10$$

$$a_n = \frac{1}{n}$$