Robusztus és kétszíntű optimalizálás

Mályusz Attila Edmund
Témavezető: Kis Tamás
2022.12.18

Bevezetés

A projektmunka zárásaként a kétszintű optimalizálással és annak robusztus optimalizáshoz való kapocslatáról foglalkoztam. Először bevezetem a kétszintű optimalizálás alap definícióit, nehézségeit. Végűl bemutatok egy feladatot melyben mindkét témakör előkerül.

Kétszintes optimalizálás

A kétszintű optimalizálási problémák olyan optimalizálási problémák, amelyeknél az optimális megoldási halmaz (részben) egy második paraméteres feladat megoldási halmazától függ. A második probléma a következő

$$\min\{f(x,y) : g(x,y) \leq 0, y \in T\} \tag{1}$$

, ahol $f : \mathbf{R}^n \times \mathbf{R}^m \to \mathbf{R}$, $g : \mathbf{R}^n \times \mathbf{R}^m \to \mathbf{R}^p$, $T \subset \mathbf{R}^m$ zárt halmaz.
Legyen $Y : \mathbf{R}^n \rightrightarrows \mathbf{R}^m$ a feltételt teljesitő halmaz (,ahol $\rightrightarrows$ pontról halmazra képző függvényt értünk):

$$Y(x) = \{y : g(x,y) \leq 0\} \tag{2}$$

Legyen

$$\varphi(x) = \min_y\{f(x,y) : g(x,y) \leq 0, y \in T\} \tag{3}$$

az optimális érték függvénye. Legyen $\Psi(x) : \mathbf{R}^n \rightrightarrows \mathbf{R}^m$ a megoldás halmaza 1-nek. Legyen

$$\Psi(x) = \{y \in Y(x) \cap T : f(x,y) \leq \varphi(x)\} \tag{4}$$

$$\mathbf{gph}\Psi = \{(x,y) \in \mathbf{R}^n \times \mathbf{R}^m : y \in \Psi(x)\} \tag{5}$$

a gráfja Ψ-nek. Ekkor a kétszintes optimalizálási probléma a következő:

$$\min_x\{F(x,y) : G(x) \leq 0, (x,y) \in \mathbf{gph}\Psi, x \in X\} \tag{6}$$

, ahol $F : \mathbf{R}^n \times \mathbf{R}^m \to \mathbf{R}$, $G : \mathbf{R}^n \to \mathbf{R}^q$, $X \subset \mathbf{R}^n$ zárt halmaz.
1 és 6 egy két játékos feladatként lehet értelmezni, ahol az első játékos (vezető) választ egy x-et és ezt lekommunikálja a második játékosnak (követő). Aki ezek után x-et figyelembe véve választ magának egy optimális y-t amit vissza ad az első játékosnak. Az első játékos pedig x és y segítségével kiszámolja a saját veszteségét. Az első játékos célja a saját veszteségének minimalizálása. 6 felső problémának nevezzük. 1-et pedig alsó problémának.

Nehézségek a kétszíntű optimalizálással kapcsolatban

Vegyük a következő esetet, ahol $n = 1$, $m = 1$. Azaz az alsó és felső problémának egy-egy változója van. Itt $G(x) \equiv 0$ és $\{(x,y) : g(x,y) \leq 0\}$ a lehetséges pontok halmaza pedig a vonalazott halmaz a képen. Ha $x = x_0$ esetet nézzük, akkor a lehetséges megoldásai az alsó problámának 1 az x_0 fölötti szakasz. A mi példánkon az alsó probléma optimalizálandó függvénye $f(x,y) = -y$. Így az optimális megoldás pontosan a vastagított vonal. Másszóval $\mathbf{gph}\Psi$ a vastagított vonal. Ahogy az ábárn látszik a kétszintes optimalizálási probléma egy nem konvex feladat lesz.

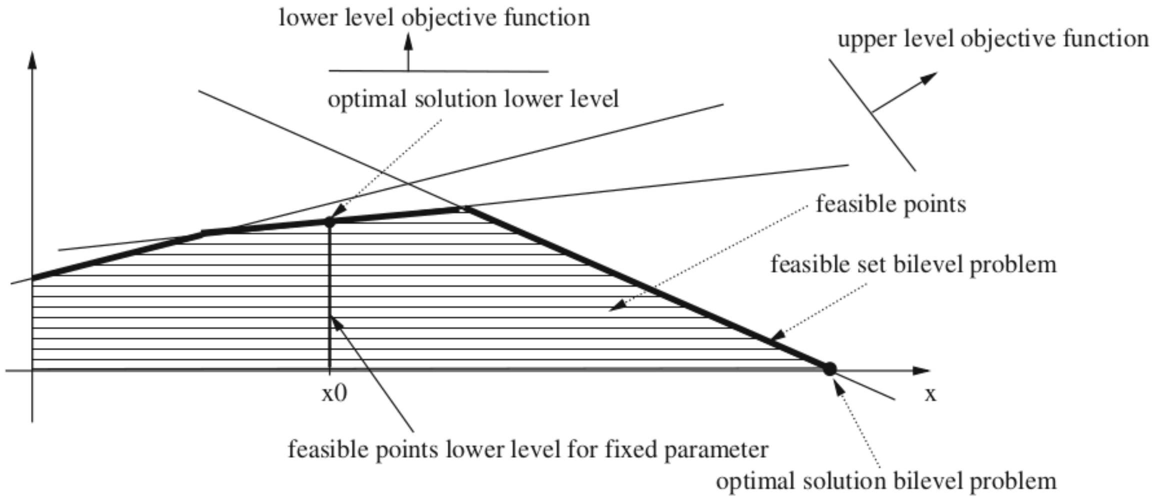


Figure 1: 1

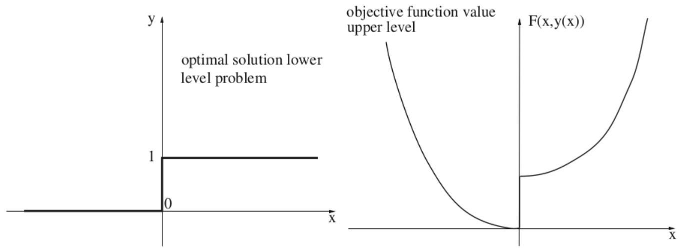
A kövekező példához legyen a következő optimalizálási feladatunk.

$$\min_x \{x^2 + y : y \in \Psi(x)\} \tag{7}$$

, ahol

$$\Psi(x) = argmin_y\{-xy : 0 \leq x \leq 1\} \tag{8}$$

A Ψ gráfja a jobb oldalon látható a képen. A felső probléma $x \to F(x, \Psi(x))$ pedig a jobb oldalon látható. Az gráf nem egy függvényt ad ugyanis az érték függ az $y \in \Psi(x)$ értéktől $x = 0$-ban. Ha $y = 0$-t mond a követő, akkor optimista kétszíntű optimalizálási problémáról beszélünk. Ha $y = 1$-et, akkor pesszimista kétszintű optimalizálási problémáról. Ha pedig $y \neq 0, 1$, akkor nincs minimuma a felső problémának csak infínuma.



Optimista és pesszimista kétszintű optimalizálás

Ha a vezető feltételezheti, hogy a követő együttműködő. Azaz a $\Psi(x)$ halmazból azt fogja választani amelyik a legjobb a felső feladat célfüggvényére. Ekkor ez a következő függvényt kell minimalizálni

$$\varphi_0(x) = \min\{F(x, y) : y \in \Psi(x)\} \tag{9}$$

a $\{x : G(x) \leq 0, x \in X\}$ halmazon. Ez az optimista kétszintű optimalizálás.
Abban az eseteben, ha a vezető nem számíthat kooperációra. Akkor a következő függvényt kapjuk.

$$\varphi_p(x) = \min\{F(x, y) : y \in \Psi(x)\} \tag{10}$$

a $\{x : G(x) \leq 0, x \in X\}$ halmazon. Ez a pesszimista kétszintű optimalizálás.

Lináris kétszintes optimalizálás

A lineáris kétszintes optimalizálásra már láthattunk példát 1.
A következőt feladatot nevezzük lináris kétszintes optimalizálás problámának.

$$\min_{x,y}\{a^T x + b^T y : Ax + By \leq c, (x,y) \in \mathbf{gph}\Psi\} \tag{11}$$

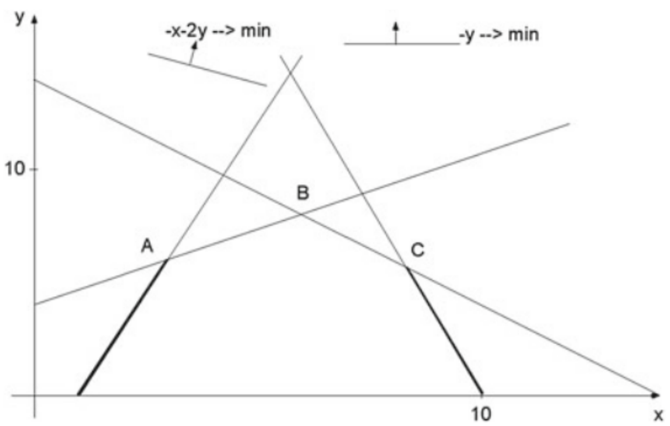
, ahol

$$\Psi(x) = argmin_y\{d^T y : Cy \leq x\} \tag{12}$$

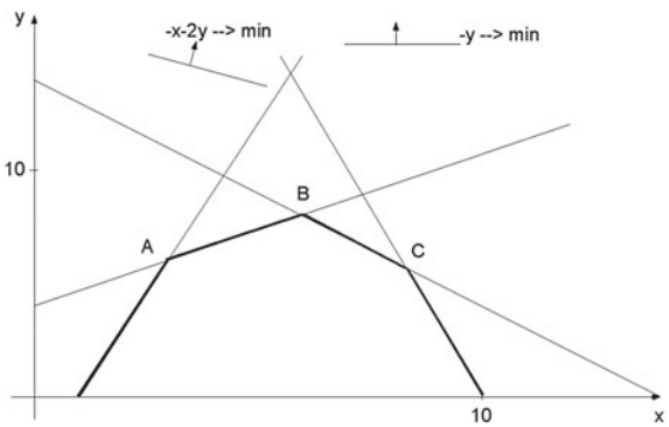
Az a probléma ezzel az általános probléma megfogalmazással, hogy az $Ax+By \leq c$ feltétel teljesülését csak y megválasztása után lehet leellenőrizni. Ezeket összefüggő feltételeknek hívjuk. Erre a problémára a következő példa (megtalálható [6]) hívja fel a figyelmet.

$$\begin{aligned} \min_{x,y} & -x - 2y \\ \text{subject to } 2x - 3y \geq & -12 \\ x + y \geq & 14 \\ y \in argmin_y\{-y : -3x + y \leq & -3, 3x + y \leq 30\} \end{aligned}$$

A $\mathbf{gph}\Psi(x)$ halmaz a vastagított vonal. Ebben az esetben az optimális megoldása a feladatnak a C pontban van.



Ha a két feltételt a felső problémából levisszük az alsóba, akkor a feladat optimum most már a B pont.



A kövekezőkben elkerüljük azt a problémát, hogy a felső feladat feltételei függenek az alsó változóitól. Így a következő lineáris kétszintes programozási feladat tekintjük

$$\min_{x,y}\{a^T x + b^T y : Ax \leq c, (x,y) \in \mathbf{gph}\Psi\} \tag{13}$$

, ahol

$$\Psi(x) = argmin_y\{d^T y : Cy \leq x\} \tag{14}$$

Meg lehet mutatni [5], hogy a $\Psi()$ leképzés gráfja előáll az $\{(x,y)^T : Cy \leq x\}$ halmaz oldalainak az összefüggő uniójaként.
Itt egy M halmaz összefüggő ha, nincsen benne két diszjunkt nyílt halmaz úniójában. Ebből kapjuk, hogy 13 feladat lineáris optimalizálás feladat és az optimum megtalálható

$$\{(x,y)^T : Cy \leq x, Ax \leq c\} \tag{15}$$

valamelyik csúcsán. Ezen tétel bizonyítása megtalálható [1]-ban és sok algorimus alapját képzi.
A lineáris kétszintes optimalizálási problémák általánosan NP-nehezek. Ezt a következő tétel mutatja meg.

Tétel 1. ([4]) $\forall\ \epsilon > 1$-ra NP-nehéz megtalálni egy olyan megoldását a (11)lineáris kétszintes optimalizálás feladatnak, amely legfeljebb ϵ szorosa az optimális megoldásnak.

A következő példa arról szól, hogy egy globál optimális megoldás elvesztheti optimalitását, ha hozzá adunk feltételt a feladatothoz, amely nem aktív az optimum értéknél . Ez egy fontos észrevétel ugyanis folytonos esetben nem így van. A példa [3]-ből van. Legyen a következő feladatunk

$$(x-1)^2 + (y-1)^2 \to \min_{x,y}, \tag{16}$$

$$\text{, ahol } y \text{ a következőt} \tag{17}$$

$$0.5y^2 + 500y - 50xy \to \min_y \tag{18}$$

Mivel az alsó probléma egy konvex optimalizálás probléma az egész téren, így kicserálhetjük az optimalitás feltételekre. Így a feladat a következő lesz

$$\min_{x,y}\{(x-1)^2 + (y-1)^2 : y - 50x + 500 = 0\} \tag{19}$$

Ennek a feladatnak az egyetlen optimális megoldása a $(x^*, y^*) = (\frac{50102}{5002}, \frac{4100}{5002})$. Az optimális érték pedig $z^* = 81,33$.
Adjuk most a feladathoz az $y \geq 0$ feltételt.

$$(x-1)^2 + (y-1)^2 \to \min_{x,y},$$

$$\text{, ahol } y \text{ a következőt}$$

$$y \in argmin_y\{0.5y^2 + 500y - 50xy : y \geq 0\}$$

Ekkor az optimális megoldás $(\bar{x}, \bar{y}) = (1, 0)$. Ez az érték viszont nem optimum 18-nak. Ez mutatja, hogy bár (x^*, y^*) lokálisan optimum, de nem globális optimum.

Globális algoritmus

Vegyük a következő két lineáris kétszintes optimalizálás problémát

$$\min_{x,y}\{a^T x + b^T y : Ax \leq c, (x,y) \in \mathbf{gph}\Psi^1\} \tag{20}$$

, ahol

$$\Psi^1(x) = argmin_y\{x^T y : By \leq d\} \tag{21}$$

és

$$\min_{x,y}\{a^T x + b^T y : Ax \leq c, By \leq d, x^T y \leq \varphi^1(x)\} \tag{22}$$

, ahol

$$\varphi^1(x) = \min_y\{x^T y : By \leq d\} \tag{23}$$

A két probléma ekvivalens és a következő algoritmust lehet adni a megoldás megtalálására.

1. Válasszunk egy x^0-at mely kielégíti a $Ax^0 \leq c$ feltételt és válasszunk egy $y^0 \in \Psi(x^0)$-t. Állítsuk $k = 1$-re és $\mathbf{Y} = \{y^0\}$.
2. Oldjuk meg a következő problémát

$$\min_{x,y}\{a^T x + b^T y : Ax \leq c, By \leq d, x^T y \leq \bar{y}^T x, \forall \bar{y} \in \mathbf{Y}\} \tag{24}$$

   legyen a globál optimum megoldás (x^k, y^k).
3. Ha $y^k \in \Psi(x^k)$, akkor megállunk és (x^k, y^k) globál optimális megoldás. Ellentétes esetben legyen

$$\mathbf{Y} = \mathbf{Y} \cup \{\bar{y}^k\} \tag{25}$$

   majd menjünk a 2. lépésre.

Tétel 2. ([2]) Legyen $\{(x, y) : Ax \leq c, By \leq d\}$ korlátos. Ekkor a fenti algoritmus megtalálj a globális optimumát a lineáris kétszintes feladatnak.

A kétszintűároptimalizálási probléma

Adott $G = (V, E)$ gráf, egy szolgáltató, és S utas, akik s_j-ből t_j-be akarnak eljutni, $j = 1, \ldots, S$. A szolgáltatónak $p : E \to \mathbb{R}$ élköltsége merül fel. Minden utasnak van egy u_j költség vektora az éleken, ami előre nem ismert, de azt tudjuk, hogy $u \in U$, ahol U egy előre ismert poliéder.

A szolgáltató minden élhez egy q_e árat akar rendelni. A q vektornak egy Q poliéderből kell kikerülni. A szolgáltató célja a hasznának a maximalizálásása, tudva azt, hogy minden utas olyan útvonalat választ, ami a költségét minimalizálja. Tehát a szolgáltató optimalizálási problémája

$$\max_{q \in Q} z \tag{26}$$

$$\min_{u \in U}\left\{\sum_{j=1}^{S}(q-p)x_j \mid x \in \Omega(q, u)\right\} \geq z, \tag{27}$$

ahol $x = (x_1, \ldots, x_S)$, és $\Omega(q, u)$ a következő feladat optimális megldásainak a halmaza:

$$\begin{aligned}
\text{minimize } & \sum_{j=1}^{S}\sum_{e\in E}(q_e+u_{j,e})x_j(e)\\
\text{subject to } & x_j(\rho(t_j))\geq 1, \quad \forall j\\
& x_j(\delta(s_j))\leq 1, \quad \forall j\\
& x_j(\delta(v_j))\leq x_j(\rho(v_j)), \quad \forall j, v\notin\{s_j,t_j\}\\
& x_j(e)\in\{0,1\}, \quad \forall j,e
\end{aligned}$$

Itt $x_j(\delta(v))=\sum_{e\in\delta(v)}x_j(e)$, and $x_j(\rho(v))=\sum_{e\in\rho(v)}x_j(e)$, ahol $\delta(v)$ és $\rho(v)$ a v csúcs kimenő és bemenő éleinek halmaza. A következőkben $x_j(e)$-re csak a $0\leq x_j(e)\leq 1$ megkötést tesszük a $x_(e)\in\{0,1\}$ helyett.

Rögzített tarifához globálisan robusztus útvonal választás

Rögzítsünk egy $q\in Q$ tarifát. Találnunk kell x és u vektorokat úgy, hogy azok a legkedezőtlenebbek megoldást adják a szolgáltató számára, de optiamális az utasoknak, azaz

$$\begin{aligned}
&\text{minimize } \sum_{j=1}^{S}\sum_{e\in E}(q_e-p_e)x_j(e)\\
&\text{subject to } u\in U,\\
&0\leq \alpha_j^{+}\perp x_j(\rho(t_j))-1\geq 0, && \forall j\\
&0\leq \alpha_j^{-}\perp -x_j(\delta(s_j))+1\geq 0, && \forall j\\
&0\leq \beta_{j,v}\perp x_j(\rho(v_j))-x_j(\delta(v_j))\geq 0, && \forall j, v\notin\{s_j,t_j\}\\
&\;0\leq x_j(e)\perp \quad q_e+u_{j,e}-\alpha^{+}_{j,e\in\rho(t_j)}+\alpha^{-}_{j,e\in\delta(s_j)}-\\
&\qquad\qquad \sum_{e=\rho(v),v\notin\{s_j,t_j\}}\beta_{j,v}+\sum_{e=\delta(v),v\notin\{s_j,t_j\}}\beta_{j,v}\geq 0 \quad \forall j,e
\end{aligned}$$

Egy relaxált feladat véges, diszkrét költség halmazzal

Adott az u^k költségvektorok egy véges halmaza, ami mellett a szolgltató problémája keresünk egy q árazást, feltételezve, hogy az utasok a legkedvezőbb válaszaikat adják minden egyes u^k költésgvektor esetében.

A változók a q árvektor, és egy $x^k=(x_1^k,\ldots,x_S^k)$ útvektor minden u^k költségvektorhoz.

$$\begin{aligned}
&\text{maximize } Z\\
&\text{subject to } q\in Q\\
&\qquad Z\leq (\sum_{j=1}^{S}\alpha^{+,k}_{j,e\in\rho(t_j)}-\alpha^{-,k}_{j,e\in\delta(s_j)}+\sum_{e\in E}(-u^k_{j,e}-p(e))x_j^k(e)) \quad \forall k\\
&\qquad 0\leq \alpha_j^{+,k}\perp x_j^k(\rho(t_j))-1\geq 0, \quad \forall j,k\\
&\qquad 0\leq \alpha_j^{-,k}\perp -x_j^k(\delta(s_j))+1\geq 0, \quad \forall j,k\\
&\qquad 0\leq \beta_{k,j,v}\perp x_j^k(\rho(v_j))-x_j^k(\delta(v_j))\geq 0, \quad \forall j,k,v\notin\{s_j,t_j\}\\
&\qquad 0\leq x_j^k(e)\perp q_e+u^k_{j,e}-\alpha^{+,k}_{j,e\in\rho(t_j)}+\alpha^{-,k}_{j,e\in\delta(s_j)}-\sum_{e=\rho(v),v\notin\{s_j,t_j\}}\beta_{k,j,v}+
\end{aligned}$$

$$\sum_{e=\delta(v), v \notin \{s_j, t_j\}} \beta_{k,j,v} \geq 0 \quad \forall k, j, e$$

Legnagyobb elmozdulás megkeresése a hasznossági térben

Adott q tarifa, u' hasznosság, $u_0 \in U^*$, a követőnek x útja és a vezetőnek egy maximum profitja F_{max}. $\nu = (u' - u_0)$.
Cél: δ megtalálása.

maximize δ

subject to $q \in Q$,

$(u_0 + \nu\delta) \in U$

$\sum_{j=1}^{S} \sum_{e \in E} (q_e - p_e) x_j(e) \leq F_m ax$

$0 = \alpha_j^+ \qquad \forall j\colon\ x_j(\rho(t_j)) - 1 > 0$

$0 = \alpha_j^- \qquad \forall j\colon\ - x_j(\delta(s_j)) + 1 > 0$

$0 = \beta_{j,v} \qquad \forall j, v \notin \{s_j, t_j\}\colon x_j(\rho(v_j)) - x_j(\delta(v_j)) > 0$

$q_e + u_{j,e} - \alpha^+_{j,e \in \rho(t_j)} + \alpha^-_{j,e \in \delta(s_j)} - \sum_{e=\rho(v), v \notin \{s_j, t_j\}} \beta_{j,v} + \sum_{e=\delta(v), v \notin \{s_j, t_j\}} \beta_{j,v} \geq 0 \quad \forall j, e\colon\ x(e) = 0$

$\alpha_j^+, \alpha_j^-, \beta_{j,v} \geq 0 \qquad \forall j, v$

Perturbáció

A következő perturbációt végezzük a kapott q megoldáson, hogy a követő megodlása erre a q-ra egyedi legyen.
Legyen π egy permutációja $(1, \ldots, |E|)$-nek, hogy teljesüljön $q_{\pi(t)} - p_{\pi(t)} \leq q_{\pi(t+1)} - p_{\pi(t+1)}$ $\forall t = 1, \ldots, |E| - 1$. Legyen T^+ a legkisebb t index, amelyre $q_{\pi(t)} - p_{\pi(t)} > 0$. Legyen q^δ a perturbált vektor a következő módon definálva.

$$q^\delta_{\pi(t)} = (q_{\pi(t)} - (t - T^+ + \frac{1}{2})\delta) \tag{28}$$

Az algoritmus

Legyen $U^* = \{u^k\} \subset U$ olyan hasznosságok dikszrét halmaza, amelyek a lehető legrosszabbak a vezetőnek. Minden egyes lépésben megoldunk egy relaxált problémát, ahol megkeresünk egy optimális q-t a diszkrét U^*-ot használva. Ezt a q értéket megperturbáljuk. Ezek után ezzel a q értékkel keresünk egy új megoldást a követőknek x, u. Ha a megoldás érték túl kicsi változáson ment át, akkor megpróbálunk a hasznosság térben elmozdulni. Ellenkező esetben hozzáadjuk U^*-hoz.

1. Legyen $U^* = \emptyset$ és $F = \infty$ (az optimuma a vezető problémájának). Válasszunk heurisztikusan egy q értéket. Legyen az a q, amely maximalizálja a $\sum q_i$ $q \in Q$ függvényt. Menjünk a 3. lépésre

2. Oldjuk meg a relaxált problémát a jelenlegi U^*-al. Ebből kapunk q tarifát és F cél fügvény értéket.

3. A kapott q értéket perturbáljuk δ-val. Így megkapjuk a q^δ-t. Ezek után oldjuk meg a feladatot q^δ-val, hogy megkapjunk egy u' értéket és F' cél függvény értéket.

4. Ha F és F' megfelelően közel van, akkor megtaláltunk egy jó q tarifát és visszaadjuk q^δ-t.

5. Ellenkező esetben találtunk egy u' vezetőnek kedvezőtlen hasznosságot. Ekkor két lehetőséget különböztetünk meg.

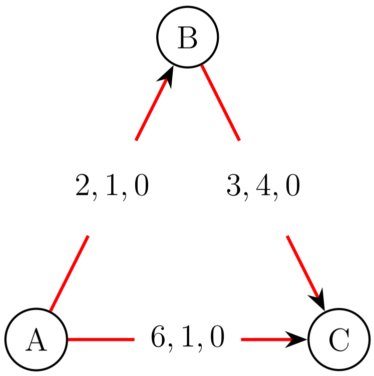
   (a) Ha u' jelentősen különbözik minden U^*-ban lévő hasznosságtól, akkor u'-t hozzáadjuk U^*-hoz.

   (b) Ha létezik u'-höz $u_0 \in U^*$, amelyre minden koordinátában legfeljebb $|E| * \delta$ mértékben különbözik, akkor keresünk egy lényegesen eltérő hasznosságot a segítségével.

6. Ha iterációs limitet elértük, akkor megállunk és visszadjuk a legjobb megoldást.

Példa az algoritmus futására

Egy utasunk lesz aki A-ból B-be akar utazni. Vegyük a következő gráfot.



Az élekre írt számok a q, u és végűl p változókhoz tartoznak. Mégpedig q és u esetén a max értékeket jelölik, p esetén pedig a jelenlegi értéket. Az élekre a következő sorrendben fogunk hivatkozni a továbbiakban. $e_1 = (A, B)$, $e_2 = (A, C)$, $e_3 = (B, C)$.

1. Először veszünk egy q-t amely maximalizálja a $\sum_{e \in E} q_e$. Mivel a példa nagyon egyszerű, így a megoldás a következő lesz $q = [2, 6, 3]$.

2. A perturbálás a következő értéket adja ki q-ra $\delta = 0.05$-el $q = [1.975, 5.925, 2.875]$

3. Ezek után a fix q tarifára való optimális megoldás keresése feladat a következőképpen

néz ki.

$$\begin{aligned}\text{minimize } & 1.975x_1 + 5.925x_2 + 2.875x_3 \\ \text{subject to } & u \in U \\ & 0 \leq \alpha_1^+ \perp x(e_2) + x(e_3) \geq 1 \\ & 0 \leq \alpha_1^- \perp x(e_2) + x(e_1) \leq 1 \\ & 0 \leq \beta_{1,B} \perp x(e_3) - x(e_1) \geq 0 \\ & 0 \leq x_{e_3} \perp 2.875 + u_3 - \alpha_1^+ + \beta_{1,B} \geq 0 \\ & 0 \leq x_{e_2} \perp 5.925 + u_2 - \alpha_1^+ + \alpha_1^- \geq 0 \\ & 0 \leq x_{e_1} \perp 1.975 + u_1 + \alpha_1^- - \beta_{1,B} \geq 0\end{aligned}$$

Erre az optimális megoldás a következő.
$u = [1, 0, 0.075]$, $x = [1, 0, 1]$, $\alpha^+ = [5.925]$, $\alpha^- = [0]$, $\beta = [0, 2.975, 0]$. Az optimum érték pedig 4.85.

4. A fix költség vektorokhoz (jelenleg csak $u = [1, 0, 0.075]$) keresünk egy legjobb q tarifát. A következő lineáris programozás feladatot kapjuk.

$$\begin{aligned}\text{minimize } & Z \\ \text{subject to } & q \in Q \\ & Z \leq (\sum_{j=1}^{S} \alpha_{j,e\in\rho(t_j)}^{+,k} - \alpha_{j,e\in\delta(s_j)}^{-,k} + \sum_{e\in E}(-u_{j,e}^k - p(e))x_j^k(e)) \quad \forall k \\ & 0 \leq \alpha_1^+ \perp x(e_2) + x(e_3) \geq 1 \\ & 0 \leq \alpha_1^- \perp x(e_2) + x(e_1) \leq 1 \\ & 0 \leq \beta_{1,B} \perp x(e_3) - x(e_1) \geq 0 \\ & 0 \leq x_{e_3} \perp 2.875 + u_3 - \alpha_1^+ + \beta_{1,B} \geq 0 \\ & 0 \leq x_{e_2} \perp 5.925 + u_2 - \alpha_1^+ + \alpha_1^- \geq 0 \\ & 0 \leq x_{e_1} \perp 1.975 + u_1 + \alpha_1^- - \beta_{1,B} \geq 0\end{aligned}$$

Az optimális megoldás pedig $x = [0, 1, 0]$, $q = [2, 6, 2.925]$, $\alpha^+ = [6]$, $\alpha^- = [0]$, $\beta = [0, 3, 0]$. Az optimum érték 6.

A diszkrét költségek halmaza végűl a kövekező lesz $u = [[1, 0, 0.075], [1, 0, 0.15]]$. Az algoritmus szerint a szolgáltató legrosszabb esetben is 6 egyésget nyerhet.

Abban az esetben, ha az élekre bevezetünk $p = [1, 1, 1]$ költségeket a szolgáltatónak, az algoritmus a következőket fogja találni. A diszkrét utilitytik a következők lesznek $u = [[0.075, 0, 0], [0, 0.85, 0], [1, 0, 4]]$. Az optmimum értéke a szolgáltatónak 3.15.

Implementálás

Az algoritmust C++-ban lett implementálva használja az ELTE lemon és IBM Cplex C++ API-t. A következő linken érhető el `https://github.com/atimaly/Robust_Path_Tariff`

Bibliography

[1] W. F. Bialas and M. H. Karwan. Two-level linear programming. *Management Science*, 30(8):1004–1020, 1984.

[2] S. Dempe and S. Franke. Solution algorithm for an optimistic linear stackelberg problem. *Computers and Operations Research*, 41:277–281, 01 2014.

[3] S. Dempe and S. Lohse. Dependence of bilevel programming on irrelevant data. 2011.

[4] X. Deng. Complexity issues in bilevel linear programming. In *Multilevel optimization: Algorithms and applications*, pages 149–164. Springer, 1998.

[5] H. H. B. B. F. Nožička, J. Guddat. Theorie der linearen parametrischen optimierung. 1974.

[6] A. G. Mersha and S. Dempe. Linear bilevel programming with upper level constraints depending on the lower level solution. *Applied Mathematics and Computation*, 180(1):247–254, 2006.