Optymalizacja wyceny przejazdów w *ride-hailingu* za pomocą narzędzi obliczeniowej mikroekonomii

(Optimizing *ride-hailing* fares using computational microeconomy tools)

Maurycy Borkowski

Praca licencjacka

Promotor: dr Marek Adamczyk

Uniwersytet Wrocławski
Wydział Matematyki i Informatyki
Instytut Informatyki

4 lipca 2022

Streszczenie

W swojej pracy Hikima et al. [2] przedstawili model wyceny przejazdów samochodowych w usługach tak zwanego *ride-hailingu* jak Uber czy Bolt. W tym modelu zaproponowali algorytm 3-aproksymacyjny dla problemu maksymalizacji zysku z przejazdów. W niniejszej pracy pokazujemy 2-aproksymacyjny algorytm dla problemu, który zdaje się być znacznie prostszy w potencjalnej implementacji. Ponadto, jego gwarancja aproksymacji odnosi się do znacznie silniejszego ograniczenia górnego w porównaniu do tego, którego użyto w pracy [2].

In their work [2], Hikima et. al presented a pricing model for car travel services also known as *ride-hailing*. In this model, they proposed a 3-approximation algorithm for problem of maximizing the profit from fares. In this paper we show a 2-approximation algorithm for the problem, which seems to be much simpler in potential implementation. Moreover, its approximation guarantee refers to much stronger upper bound compared to the one used in [2].

Spis treści

Rozdział 1.

Wstęp

Rozważmy sposób działania aplikacji oferujących usługę *ride-hailingu* np. Uber albo Bolt. W aplikacji użytkownik zamawiający przejazd zaznacza adres docelowy, do którego chce dojechać oraz adres odbioru, w którym się znajduje. Dostawca wycenia jego przejazd i oferuje mu cenę, którą klient akceptuje lub nie. Jeżeli zaakceptuje cenę to dostawca przydziela mu kierowcę, który obsłuży jego przejazd za ustaloną cenę i przyjedzie go odebrać w niedługim ($\approx$ 5 min.) czasie. Może się jednak zdarzyć, że dostawca nie ma żadnej taksówki w pobliżu klienta i wtedy wysyła informacje z odpowiednim komunikatem i w tym momencie nie obsługuje jego przejazdu. Jest to oczywiście niepożądane i jako dostawca zależy nam na jak najmniejszej liczbie takich sytuacji.

Z punktu widzenia przewoźnika dostajemy w każdej minucie kilkadziesiąt czy nawet kilkaset zapytań w danej lokacji o przejazdy. Mamy dane lokalizacje wolnych taksówek i informacje, które z nich są w stanie dotrzeć w niedługim czasie do pewnego klienta. Nasz serwis ma za zadanie wyznaczenia cen dla klientów. Na zbiorze zamawiających, którzy zaakceptowali cenę liczymy *najlepsze* skojarzenie (dopasowanie klient-taksówka), *najlepsze* tj. takie, które maksymalizuje dochód minus koszty przejazdów. Na etapie (liczenie skojarzenia) może się zdarzyć, że pewnych klientów nie skojarzymy z taksówką mimo tego, że zaakceptowali oni cenę.

W tej pracy przybliżymy rozwiązanie problemu optymalizacji oczekiwanego zysku *ride-hailingu*, przedstawione przez Hikimę et. al w pracy [2], a następnie zaproponujemy prostszy i efektywniejszy algorytm optymalizujący ten problem.

Rozdział 2.

Definicja problemu

W swojej pracy [2] Hikima et. al przedstawili następujące sformułownanie modelu.

2.1. Model

Instancja problemu

Dany jest ważony graf dwudzielny $G = (C, T, E)$. Zbiór wierzchołków C interpretujemy jako zbiór klientów, natomiast T jako zbiór taksówek, które możemy potencjalnie przypisać zamawiającym. Krawędzie zadają relację osiągalności poszczególnych taksówek do konkretnych agentów tj. dla każdej krawędzi $(c, t) \in E$ oznacza, że taksówka t może obsłużyć klienta c. Koszt każdej krawędzi oznaczamy jako $w_{c,t}$.

Model rozwiązania i procedura skojarzenia

Rozwiązanie sprowadza się do znalezienia wektora cen $(\pi_c)_{c \in C}$. Dla każdego klienta c, na podstawie ceny π_c stochastycznie decyduje się, czy bierze on udział w kolejnym etapie procesu tworzenia skojarzenia: klient kontynuuję procedurę (przechodzi do kolejnego etapu) z prawdopodobieństwem $\mathbb{P}\left[v_c \geq \pi_c\right]$ albo odchodzi z prawdopodobieństwem $1 - \mathbb{P}\left[v_c \geq \pi_c\right]$. Jeżeli klient c nie akceptuje zaproponowanej ceny, usuwamy z grafu jego wierzchołek i krawędzie incydentne z nim.

Oznaczmy jako $G_{\text{ok}} = (C_{\text{ok}}, T, E_{\text{ok}})$ graf powstały po wyrzuceniu wszystkich klientów, którzy nie zaakceptowali ceny, oraz wszystkich krawędzi z nimi związanych. Z tego grafu wybieramy skojarzenie $M \subseteq E_{\text{ok}}$ (zbiór krawędzi nie posiadających wspólnych wierzchołków). Wartością rozwiązania jest suma cen pomniejszonych o wagę, tj. $\pi_c - w_{c,t}$.

Skojarzenia

Zauważmy, że gdy ceny $(\pi_c)_{c\in C}$ są ustalone i poznaliśmy decyzje klientów na temat ich akceptacji, tj. wektor $(\mathbf{1}[v_c \geq \pi_c])_{c\in C}$ — czy też równoważnie zbiór $C_{\text{ok}} = \{c \in C \,|v_c \geq \pi_c\}$ — , to nasz problem sprowadza się do klasycznego problemu znajdowania maksymalnego skojarzenia w ważonym grafie dwudzielnym, który możemy opisać następujący programem liniowym:

$$\begin{aligned} f(\pi_1, ..., \pi_n; C_{\text{ok}}) = \max_{z_{c,t}\in C\times T} & \sum_{(c,t)\in E} (\pi_c - w_{c,t}) \cdot z_{c,t} \\ \text{s.t.} & \sum_{t\in\delta(c)} z_{c,t} \leq \mathbf{1}[c \in C_{\text{ok}}] \quad \forall_{c\in C} \\ & \sum_{c\in\delta(t)} z_{c,t} \leq 1 \quad \forall_{t\in T} \\ & z_{c,t} \in \{0,1\} \quad \forall_{(c,t)\in E} \end{aligned}$$

gdzie $\delta(c)$ oznacza zbiór taksówek incydentnych z c — i analogicznie $\delta(t)$ oznacza zbiór klientów incydentnych z t — $z_{c,t} \in \{0,1\}$ określa czy $(c,t) \in E$ są ze sobą skojarzone $(z_{c,t} = 1)$ czy nie $(z_{c,t} = 0)$.

Pierwsze ograniczenie mówi, że tylko wierzchołek c, który zaakceptuje zaproponowaną cenę może zostać skojarzony z jedną taksówką t; drugie gwarantuje, że każdy wierzchołek t może zostać skojarzony tylko z jednym klientem c.

Problem ride-hailingu

Funkcja $f\left(\pi_1, ..., \pi_n; (v_c)_{c\in C}\right)$ wyznacza maksymalne skojarzenie dla zadanych przez nas cen $(\pi_c)_{c\in C}$ przy jednoczesnej wiedzy o tym, którzy klienci zaakceptowali zaproponowane im ceny. Jednakże przed wyznaczeniem cen $(\pi_c)_{c\in C}$ nie znamy jeszcze realizacji $(\mathbf{1}[v_c \geq \pi_c])_{c\in C}$. A w związku z tym nasze zadanie to znalezienie takiego wektora $(\pi_c)_{c\in C}$, który zmaksymalizuje nam średnią wartość skojarzenia, gdzie średnia jest wzięta po wszystkich możliwych realizacjach wektora $(\mathbf{1}[v_c \geq \pi_c])_{c\in C}$. Innymi słowy nasze zadanie polega na maksymalizacji następującej funkcji wektora $(\pi_c)_{c\in C}$:

$$\max_{(\pi_c)_{c\in C}\in\mathbb{R}_+^{|C|}} \mathbb{E}_{(v_c)_{c\in C}}\left[f(\pi_1, ..., \pi_n; \{c\,|v_c \geq \pi_c\})\right]. \qquad \text{(RH-PROBLEM)}$$

Powyższą zwięzłą formułę $\mathbb{E}_{(\pi_c)_{c\in C}}\left[f(\pi_1, ..., \pi_n; \{c\,|v_c \geq \pi_c\})\right]$ można zapisać w nieco bardziej zrozumiały sposób:

$$\mathbb{E}_{(v_c)_{c\in C}}\left[f(\pi_1, ..., \pi_n; \{c\,|v_c \geq \pi_c\})\right] = \sum_{C_{\text{ok}}\subseteq C} \left(\prod_{c\in C_{\text{ok}}} \mathbb{P}[v_c \geq \pi_c] \cdot \prod_{c\notin C_{\text{ok}}} (1 - \mathbb{P}[v_c \geq \pi_c]) \cdot f(\pi_1, ..., \pi_n; C_{\text{ok}}) \right)$$

2.2. Podejście Hikimy et al.

Z tego sformułowania widać od razu, że nasz problem jest trudny obliczeniowo, ponieważ występuje w nim sumowanie po wykładniczej ilości składników. Jednocześnie funkcja celu nie jest wklęsła wg [2], co uniemożliwia użycie standardowych narzędzi optymalizacji wypukłej.

W związku z tym Hikima et al. w swoim podejściu postawili nowy problem matematyczny:

$$\begin{aligned} \max_{(\pi_c)_{c\in C}\in\mathbb{R}_+^C}\ \max_{z_{c,t}\in C\times T} & \sum_{(c,t)\in E}(\pi_c - w_{c,t})\cdot z_{c,t} \qquad \text{(UP-PROBLEM)} \\ \text{s.t.} & \sum_{t\in\delta(c)} z_{c,t} \le \mathbb{P}\left[v_c \ge \pi_c\right] \quad \forall_{c\in C} \\ & \sum_{c\in\delta(t)} z_{c,t} \le 1 \quad \forall_{t\in T} \\ & z_{c,t}\in(0,1) \quad \forall_{(c,t)\in E} \end{aligned}$$

Następnie pokazali, że optymalne rozwiązanie programu (UP-PROBLEM) jest 1/3-aproksymacją optymalnego rozwiązania (RH-PROBLEM). Ostatecznie ich rozwiązanie sprowadzało się do znalezienia optymalnego rozwiązania (UP-PROBLEM) za pomocą metody szukania wypukłego przepływu o minimalnym koszcie.

Dodatkowo poczynili oni pewne założenia dotyczące rozkładu zmiennych v_c. W jednym wariancie założyli, że klient się zgodzi się na zaproponowaną cenę π_c z prawdopodobieństwem:

- ReLU:

$$\mathbb{P}\left[v_c \ge \pi_c\right] := 1 - F_c^{ReLU}\left(\pi_c\right) = \begin{cases} 1 & (\pi_c < q_c) \\ -\frac{1}{(\alpha-1)\cdot q_c}\cdot\pi_c + \frac{\alpha}{\alpha-1} & (q_c \le \pi_c \le \alpha\cdot q_c) \\ 0 & (\pi_c > \alpha\cdot q_c) \end{cases}$$

- Sigmoid:

$$\mathbb{P}\left[v_c \ge \pi_c\right] := 1 - F_c^{Sigmoid}\left(\pi_c\right) = 1 - \frac{1}{1+e^{\frac{-(\pi_c-\beta\cdot q_c)}{\gamma\cdot|q_c|}}}$$

gdzie $\alpha, \beta, \gamma, q_c$to stałe a q_c w powyższym modelu oznacza minimalną cenę, dla której klient na pewno się zgodzi. Funkcji postaci Sigmoidy możemy użyć gdy zależy nam by była ona różniczkowalna we wszystkich miejscach.

W naszym modelu nie musimy wykonywać takich założeń. Wystarczy nam pewne, znacznie bardziej ogólne założenie, które przybliżymy w poniższej sekcji.

2.3. Rozkłady ewaluacji v_c

Aby można było użyć zaproponowanego przez nas rozwiązania potrzebujemy założenia, że funkcja $R_c\left(p_c\right) = F_c^{-1}\left(1-p_c\right)\cdot p_c$ jest funkcją wklęsłą. W tej sekcji podamy intuicję co ta funkcja oznacza, zilustrujemy to na przykładzie dystrybuanty rozkładu ewaluacji F_c typu ReLU, którą również zaproponowali Hikima et. al. w swojej pracy.

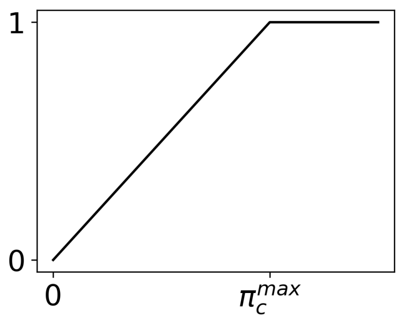
Zdefiniujmy funkcję $R_c\left(p_c\right) = F_c^{-1}\left(1-p_c\right)\cdot p_c$, w naszym rozwiązaniu będziemy proponować każdemu klientowi cenę $F_c^{-1}\left(1-p_c\right)$, czyli tak na prawdę cenę odpowiadającą kwantylowi rzędu

p_c rozkładu empirycznego cen. Przy takiej propozycji cen klient zgodzi się na nią z prawdopodobieństwem p_c (Dowód Lematu 3). Wtedy funkcja R_c określa nam oczekiwany dochód z każdego klienta c.

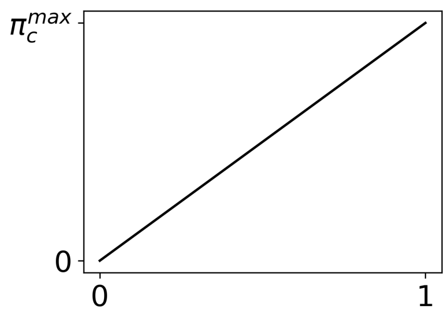
Rozważymy funkcję postaci ReLU z założeniami, że klient zgodzi się na przejazd z prawdopodobieństwem równym 1 tylko dla ceny równej 0 oraz każda $v_c \leq \pi_c^{max}$.

Proponujemy poniższą, prostszą (dla ułatwienia zrozumienia) postać dystrybuanty dla każdego klienta c:

$$F_c\left(\pi_c\right) = \begin{cases} 0 & (\pi_c = 0) \\ \frac{\pi_c}{\pi_c^{max}} & (0 < \pi_c \leq \pi_c^{max}) \\ 1 & (\pi_c > \pi_c^{max}) \end{cases}$$

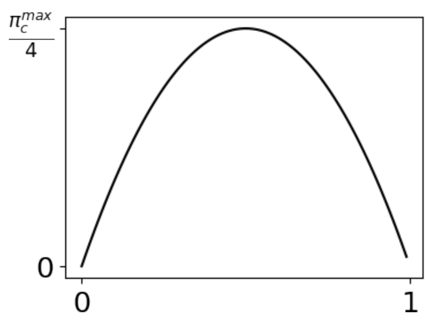


Rysunek 2.1: Wykres $F_c\left(\pi_c\right)$



Rysunek 2.2: Wykres $F_c^{-1}\left(p_c\right)$

Rysunek 2.3: Wykres $R_c\left(p_c\right)$



Jak widać z wykresu $R_c\left(p_c\right)$ lub po prostu wyliczając jej postać, widzimy, że jest ona wklęsła i spełnia nasze założenie.

W ogólności dystrybuanta F_c nie musi być taka, że R_c będzie wklęsła, ale nie będziemy się tym zajmować w tej pracy. Istnieje procedura nazywana *żelazkowaniem*, która pozwala pominąć to założenie i uogólnia model. Użycie tej metody prezentował Qigi Yan w [4].

Rozdział 3.

Nasze podejście

Hikima et al. w swojej pracy ([2]) przedstawili rozwiązanie (UP-PROBLEM) za pomocą algorytmu minimalnych przepływów, osiągając rozwiązanie 3-aproksymacyjne. Zaproponujemy rozwiązanie o lepszej skuteczności (2-aproksymacja), używające drastycznie prostszego algorytmu. Ponadto, co nawet istotniejsze, Hikima et al. porównywali swoje rozwiązanie do optymalnego algorytmu *postowanych cen* (klientowi proponowana jest cena on ją odrzuca lub nie). Nasz algorytm porównywamy potem (Sekcja 4..) do optymalnej aukcji Bayesowskiej (klient proponuję cenę, my ją odrzucamy lub nie), która siłą rzeczy jest mechanizmem mocniejszym, ale nie możliwym do zaimplementowania w warunkach *ride-hailingu*, tym mocniejszy jest jednak nasz wynik.

3.1. Problem

Chcemy rozwiązać problem propozycji cen przejazdów dla klientów tak by zoptymalizować oczekiwany zysk.

Na potrzebę łatwiejszego zrozumienia dodamy i zmienimy niektóre oznaczenia. Każdemu klientowi będziemy proponować cenę π_c. Wagi krawędzi dalej oznaczamy przez $w_{c,t}$. Każdy klient c ma własną ewaluację, na którą wycenia swój przejazd, oznaczymy ją przez v_c. Nie jest nam ona dana oczywiście wprost, ale zakładamy znajomość rozkładów ewaluacji dla każdego klienta, co jest sensownym założeniem, w realnym świecie rozkłady te możemy wyliczać na podstawie danych z poprzednich przejazdów, których przewoźnicy mają wystarczająco dużo.

W naszym modelu klient zgadza się na przejazd wtedy i tylko wtedy gdy $v_c \geq \pi_c$, dzieje się to z prawdopodobieństwem $\mathbb{P}\left[v_c \geq \pi_c\right] = 1 - F_c\left(\pi_c\right)$, które znamy lub możemy szybko policzyć na podstawie dystrybuanty przybliżonego rozkładu ewaluacji F_c. Przez $p_{c,t}\forall_{(c,t)\in E}$ oznaczymy prawdopodobieństwo tego, że rozważymy dla klienta c przejazd taksówką t (jeżeli c zgodzi się na cenę), a przez $p_c = \sum_{t\in\delta(c)} p_{c,t}$, prawdopodobieństwo, że obsłużymy klienta c (jeśli zgodzi się na zaproponowaną cenę). Przez zysk rozumiemy oczywiście sumę $\pi_c - w_{c,t}$ dla tych klientów c, których przejazdy obsługujemy.

Skoro tak na prawdę dalej budujemy skojarzenie na grafie dwudzielnym dobrze myśleć o problemie *ride-hailingu* analogicznie. Przez dodanie krawędzi (c, t) będziemy rozumieć pomyślne skojarzenie klienta c z taksówką t (klient c jedzie do miejsca docelowego taksówką t). Powiemy, że krawędź (c, t) jest *zablokowana* przez t, jeśli taksówka t została pomyślnie skojarzona z innym klientem i obsługuje już jego przejazd.

3.2. Program liniowy

Pokażemy, że rozwiązanie poniższego programu liniowego jest górnym ograniczeniem problemu *ride-hailingu.* Co więcej, nie jest to nawet ograniczenie optymalnego problemu ustalania cen, ale jest to ograniczenie górne na wynik optymalnej aukcji Bayesowskiej (Sekcja 4..). Program jest inspirowany wynikiem Qiqi Yana [4]:

$$\max \sum_{c} \left(F_c^{-1}(1-p_c) \cdot p_c \right) - \sum_{(c,t)\in E} p_{c,t} \cdot w_{c,t} \qquad \text{(PRICE-MATCH-LP)}$$

$$\text{s.t.} \quad \sum_{t\in\delta(c)} p_{c,t} \le 1 \qquad \forall_c \tag{3.1}$$

$$\sum_{c\in\delta(t)} p_{c,t} \le 1 \qquad \forall_t \tag{3.2}$$

$$p_c = \sum_{t\in\delta(c)} p_{c,t} \qquad \forall_c \tag{3.3}$$

Twierdzenie 1. *Rozwiązanie* (PRICE-MATCH-LP) *jest górnym ograniczeniem OPT (aukcji postowanych cen).*

Dowód. Rozważmy optymalny mechanizm (w modelu aukcji postowanych cen), pokażemy, że $p^*_{c,t} = \mathbb{P}\left[OPT \text{ przewozi } c \text{ taksówką } t\right]$ jest dopuszczalnym rozwiązaniem. W następnym kroku pokażemy, że optymalny mechanizm, średnio na kliencie c nie zarobi, więcej jak $R_c(p^*_c) = F_c^{-1}(1-p^*_c) \cdot p^*_c$

Rozwiązanie OPT musi spełniać (3.1) i (3.2), ponieważ OPT odpowiada pewnemu średniemu skojarzeniu a spełnienie (3.1) i (3.2) to warunek konieczny i wystarczający średniego skojarzenia, stąd $p^*_{c,t}$ jest dopuszczalnym rozwiązaniem.

Kiedy OPT ma ustalone ceny π_c dla każdego klienta c (na przejazd t) to zgadza się on na nią z prawdopodobieństwem $\mathbb{P}[v_c \ge \pi_c] = 1 - F_c(\pi_c)$. Pokażemy, że $\mathbb{E}[OPT(c)]$ czyli średni uzysk OPT-a na kliencie c jest nie większy niż $F_c^{-1}(1-p^*_c) \cdot p^*_c$. Wiemy, że

$$\begin{aligned}\mathbb{E}[OPT(c)] =& \pi_c \cdot \mathbb{P}[c \text{ zgadza się na } \pi_c \text{ ORAZ } OPT \text{ przewozi } c] \\ =& \pi_c \cdot \mathbb{P}[v_c \ge \pi_c] \cdot \mathbb{P}[OPT \text{ przewozi } c | \, v_c \ge \pi_c].\end{aligned}$$

Ale jednocześnie $\pi_c = F_c^{-1}(1 - \mathbb{P}[v_c \ge \pi_c])$, a więc:

$$\begin{aligned}\mathbb{E}[OPT(c)] &= \pi_c \cdot \mathbb{P}[v_c \ge \pi_c] \cdot \mathbb{P}[OPT \text{ przewozi } c | \, v_c \ge \pi_c] \\ &= F_c^{-1}(1 - \mathbb{P}[v_c \ge \pi_c]) \cdot \mathbb{P}[v_c \ge \pi_c] \cdot \mathbb{P}[OPT \text{ przewozi } c | \, v_c \ge \pi_c] \\ &\le F_c^{-1}(1 - \mathbb{P}[v_c \ge \pi_c] \cdot \mathbb{P}[OPT \text{ przewozi } c | \, v_c \ge \pi_c]) \cdot \mathbb{P}[v_c \ge \pi_c] \cdot \mathbb{P}[OPT \text{ przewozi } c | \, v_c \ge \pi_c] \\ &= R^*_c(\mathbb{P}[v_c \ge \pi_c] \cdot \mathbb{P}[OPT \text{ przewozi } c | \, v_c \ge \pi_c]) \\ &= R^*_c(p^*_c),\end{aligned}$$

co wynika z tego, że funkcja $F_c^{-1}(\cdot)$ jest monotoniczna, a $\mathbb{P}[OPT \text{ przewozi } c | \, v_c \ge \pi_c] \le 1$. □

Twierdzenie 2. *Program* (PRICE-MATCH-LP) *możemy rozwiązać efektywnie o ile R_c jest wklęsła.*

Dowód. Rozważmy funkcję $R_c(p_c) = F_c^{-1}(1-p_c) \cdot p_c$ (dla $p_c \in [0,1]$) . Z definicji przyjętej przez nas postaci F_c otrzymujemy $R_c(p_c) = F_c^{-1}(1-p_c) \cdot p_c = q_c^{max} \cdot (1-p_c) \cdot p_c$.

Ta funkcja jest oczywiście wklęsła, stąd funkcja celu (PRICE-MATCH-LP) jest wklęsła (odejmowane liniowe składniki nie zaburzają tej własności).

Zatem jej maksymalizacja po wypukłym obszarze (tj. obszarze skojarzeń zdefiniowanych przez (3.1) i (3.2)) jest możliwa w czasie wielomianowym, na przykład *metodą elipsoid* przedsawioną w [1] □

3.3. Algorytm

Na podstawie (PRICE-MATCH-LP) proponujemy poniższy algorytm, który będzie dawał 2-aproksymacyjne rozwiązanie problemu *ride-hailingu*:

Algorytm 1

For each client c in random order:
  propose c price $\pi_c = F_c^{-1}(1-p_c)$
  if c accepts price π_c:
    pick taxi t with propability $\frac{p_{c,t}}{p_c}$
    if (c,t) is not blocked:
      serve c with taxi t for price π_c

3.4. Uzasadnienie 2-aproksymacji

Lemat 3. *Krawędź (c,t) jest zablokowana przez t z prawdopodobieństwem co najwyżej $\frac{1}{2}$.*

Dowód. Krawędź (c,t) będzie zablokowana przez t wtedy i tylko wtedy, gdy taksówka t została *skojarzona* już z innym klientem c', wtedy powiemy, że (c,t) jest zablokowana przez (c',t). Klient c' został skojarzony z taksówką t tylko wtedy gdy algorytm rozważył klienta c' przed c, c' zgodził się na zaproponowaną mu cenę i algorytm zaproponował taksówkę t klientowi c'.

Z konstrukcji algorytmu, klientów rozważamy w losowej kolejności stąd prawdopodobieństwo, że c' wystąpi przed c w losowej permutacji dla ustalonego c wynosi $\frac{1}{2}$. Proponujemy cenę $\pi_c = F_c^{-1}(1-p_c)$ natomiast c' zgadza się na nią z prawdopodobieństwem:

$$\mathbb{P}[v_{c'} \geq \pi_{c'}] = 1 - \mathbb{P}[v_{c'} \leq \pi_{c'}] = 1 - F_{c'}(\pi_{c'}) = 1 - F_{c'}\left(F_{c'}^{-1}(1-p_{c'})\right) = p_{c'}$$

Jeżeli c' zgodzi się na cenę to przydzielimy mu taksówkę t z prawdopodobieństwem $\frac{p_{c',t}}{p_{c'}}$. Wobec powyższego krawędź (c,t) zostanie zablokowana przez (c',t) z prawdopodobieństwem:

$$\mathbb{P}\left[(c,t)\,\text{is blocked by}\,\left(c',t\right)\right] \leq \frac{1}{2} \cdot p_{c'} \cdot \frac{p_{c',t}}{p_{c'}} = \frac{1}{2} \cdot p_{c',t}$$

Dla ustalonego klienta c i taksówki t, krawędź (c, t) będzie zablokowana gdy będzie zablokowana przez jakąkolwiek z krawędzi $(c', t)\,\forall_{c' \in \delta(t) \setminus \{c\}}$, z dowolności wyboru c' w powyższym rozważaniu otrzymujemy ostatecznie:

$$\mathbb{P}\left[(c,t)\,\text{is blocked by}\ t\right] \leq \sum_{c' \in \delta(t)} \mathbb{P}\left[(c,t)\,\text{is blocked by}\ \left(c', t\right)\right] \leq \frac{1}{2} \cdot \sum_{c' \in \delta(t)} p_{c',t} \leq \frac{1}{2}$$

Przy czym ostatnia nierówność wynika z (3.2) w (PRICE-MATCH-LP), a przedostatnia to po prostu wyciągnięcie stałej przed sumę.

□

Twierdzenie 4. Algorytm (1) oblicza 2-aproksymację OPT.

Dowód. Z Twierdzenia (1) wiemy, że OPT (PRICE-MATCH-LP) jest górnym ograniczeniem problemu *ride-hailingu* w modelu aukcji Bayesowskiej. Pokażemy, że $\frac{OPT}{2}$ jest dolnym ograniczeniem ALG, (wynik Algorytmu (1)).

Przez $p_{c,t}$ oznaczmy prawdopodobieństwo z jaką OPT rozważy (c, t). Obliczmy oczekiwany zysk ALG określonego przez wartości $p_{c,t}$. Z rozważań w dowodzie Lematu (3) wiemy, że prawdopodobieństwo, tego że klient c zgodzi się na cenę $F_{c'}^{-1}\left(1 - p_{c'}\right)$ wynosi p_c, stąd i z konstrukcji algorytmu otrzymujemy:

$$ALG = \sum_{c} \left(\sum_{t \in \delta(c)} p_c \cdot \frac{p_{c,t}}{p_c} \cdot \mathbb{P}\left[(c,t)\ \text{ is not blocked by}\ t\right] \cdot \left(F_{c'}^{-1}\left(1 - p_{c'}\right) - w_{c,t}\right) \right) \geq \tag{3.1}$$

$$\geq \sum_{c} \left(\sum_{t \in \delta(c)} p_{c,t} \cdot \frac{1}{2} \cdot \left(F_{c'}^{-1}\left(1 - p_{c'}\right) - w_{c,t}\right) \right) = \tag{3.2}$$

$$= \frac{1}{2} \sum_{c} \left(\sum_{t \in \delta(c)} p_{c,t} \cdot F_{c'}^{-1}\left(1 - p_{c'}\right) - \sum_{t \in \delta(c)} p_{c,t} \cdot w_{c,t} \right) = \tag{3.3}$$

$$= \frac{1}{2} \left[\sum_{c} \left(F_{c'}^{-1}\left(1 - p_{c'}\right) \cdot \sum_{t \in \delta(c)} p_{c,t} \right) - \sum_{(c,t) \in E} p_{c,t} \cdot w_{c,t} \right] = \tag{3.4}$$

$$= \frac{1}{2} \left[\sum_{c} \left(F_{c'}^{-1}\left(1 - p_{c'}\right) \cdot p_c \right) - \sum_{(c,t) \in E} p_{c,t} \cdot w_{c,t} \right] = \frac{OPT}{2} \tag{3.5}$$

przy czym przejście między (3.1) a (3.2) wynika z Lematu (3). □

Rozdział 4.

Porównanie do optymalnego mechanizmu aukcji *Bayesowskiej*

Istnieje bardzo bogata teoria dotycząca tworzenia efektywnych mechanizmów sprzedaży. Wśród niej jednym z głównych wyników jest teoria Myersona [3], za którą Myerson otrzymał nagrodę Nobla. Teoria Myersona pozwala na stworzenie mechanizmu przydziału klientów do taksówek, który w *optymalny* sposób maksymalizuje zysk przewoźnika. Poniżej krótko przedstawiamy jak wygląda mechanizm Myersona w przypadku problemu *ride-hailingu*.

4.1. Mechanizm Myersona

Formalnie, rozważamy mechanizm aukcji *Bayesowskiej* z jednym sprzedawcą (systemem ride-hailingowym) i n uczestnikami z *jednym parametrem*, którzy licytują usługę (w tym przypadku przejazd). *Jeden parametr* oznacza, że z każdym klientem c jest związana jego prywatna informacja reprezentowana przez v_c, jedna liczba, która jest jego ewaluacją ceny. Ceny są losowe i niezależne między sobą i zakładamy, że pochodzą ze znanych rozkładów $\mathcal{D}_c$ dla każdego klienta c. Ewaluacje nie są jawne, ale rozkłady $\mathcal{D}_c$ już tak.

Każdy klient c składa *ofertę* b_c reprezentującą jego ewaluacje. Sprzedawca ma ograniczenie na obsługę klientów $\mathcal{I} \subseteq 2^{[n]}$ i może obsłużyć dowolny zbiór agentów z $\mathcal{I}$ — naszym przypadku $\mathcal{I}$ reprezentuje warunek, że każdy klient musi być skojarzony z co najwyżej jedną taksówką, a taksówka może przewieźć co najwyżej jednego klienta, tj. reprezentuje skojarzenia w grafie.

Mechanizm jest funkcją, która odwzorowuje wektor *ofert* $\boldsymbol{b}$ na *alokację* $A(\boldsymbol{b}) \in \mathcal{I}$, z *cenami* $\pi_c(\boldsymbol{b})$ płaconymi przez każdego klienta c. Dla ułatwienia notacji definiujemy przez $X_c(\boldsymbol{b}) := \mathbf{1}_{c \in A(\boldsymbol{b})}$ funkcję, która określa czy mechanizm obsługuje agenta c. *Zysk* każdego klienta c pod warunkiem wektora *ofert* $\boldsymbol{b}$, określamy przez $v_c \cdot X_c(\boldsymbol{b}) - \pi_c(\boldsymbol{b})$. Mechanizmy rozważane przez Myersona spełniają następujące założenia:

- *Dobrowolny udział*: klient płaci tylko wtedy gdy go obsługujemy i robimy to za cenę równą co najwyżej złożonej przez niego *ofercie*. Dla każdego wektora składanych *ofert* $\boldsymbol{b}$ mamy, $\pi_c(\boldsymbol{b}) \leq X_c(\boldsymbol{b}) \cdot b_c \quad \forall_c$.

- *Transfer w jedną stronę*: mechanizm nie płaci klientom, $\pi_c(\boldsymbol{b}) \geq 0 \quad \forall_c$.

- *Oczekiwana prawdomówność*: dla każdego klienta c, jeżeli v_c jest jego prawdziwą ewaluacją, wtedy złożenie *oferty* v_c daje oczekiwany *zysk* nie mniejszy niż przy złożeniu jakiejkolwiek innej *oferty* b_c:

$$\mathbb{E}_{b_{c'} \leftarrow \mathcal{D}_{c'} : c' \neq c} \left[v_c \cdot X_i \left(\boldsymbol{b}_{-c}, v_c \right) - \pi_c \left(\boldsymbol{b}_{-c}, v_c \right) \right] \geq \mathbb{E}_{b_{c'} \leftarrow \mathcal{D}_{c'} : c' \neq c} \left[v_c \cdot X_i \left(\boldsymbol{b}_{-c}, b_c \right) - \pi_c \left(\boldsymbol{b}_{-c}, b_c \right) \right]$$

4.2. Myerson a ride-hailing

Jednakże gdyby system ride-hailingowy chciał skorzystać z tego mechanizmu to zupełnie zmieniłby on sposób działania aplikacji — użytkownik zamiast akceptawać cenę, musiałby najpierw zaproponować własną cenę za przejazd, a następnie czekać na potwierdzenie systemu, czy ta cena jest akceptowalna czy nie, a nawet jeśli jest akceptowalna, to czy przejazd się odbędzie.

Siłą rzeczy więc jesteśmy (tzn. przewoźnik jest) skazani na mechanizm postowanych cen, czyli taki jak rozważaliśmy wcześniej. Powstaje jednak naturalne pytanie:

Jak efektywny jest nasz mechanizm w porównaniu z optymalnym mechanizmem Myersona?

Główne twierdzenie niniejszej pracy jest więc następujące:

Twierdzenie 5. *Algorytm 1 jest mechanizmem dla problemu ride-hailingu, który jest 2-aproksymacją optymalnego mechanizmu Bayesowskiego wg. teorii Myersona.*

Dowód tego twierdzenia sprowadza się jednak dość prosto do następującego lematu.

Lemat 6. *Rozwiązanie* (PRICE-MATCH-LP) *jest górnym ograniczeniem wartości optymalnej aukcji Bayesowskiej.*

Przypomnijmy więc (PRICE-MATCH-LP):

$$\max \sum_c \left(F_c^{-1} \left(1 - p_c \right) \cdot p_c \right) - \sum_{(c,t) \in E} p_{c,t} \cdot w_{c,t} \qquad \text{(PRICE-MATCH-LP)}$$

$$\text{s.t.} \quad \sum_{t \in \delta(c)} p_{c,t} \leq 1 \qquad \forall_c \tag{4.1}$$

$$\sum_{c \in \delta(t)} p_{c,t} \leq 1 \qquad \forall_t \tag{4.2}$$

$$p_c = \sum_{t \in \delta(c)} p_{c,t} \qquad \forall_c \tag{4.3}$$

Dowód. Rozważmy optymalny mechanizm Bayesowski OPT. Niech $OPT(c)$ to będzie zysk mechanizmu z klienta c. Załóżmy, że p_c to prawdopodobieństwo, że OPT przewozi klienta c, a $p_{c,t}$ to prawdopodobieństwo, że mechanizm przewozi klienta c taksówką t. Oczywiście te prawdopodobieństwa spełniają nierówności powyższego programu liniowego. Koszt jaki optymalny mechanizm płaci za przewóz klienta c taksówką t to $p_{c,t} \cdot w_{c,t}$. Pozostaje więc pokazać jedynie, że

$$\sum_c \mathbb{E} \left[OPT \left(c \right) \right] \leq \sum_c \left(F_c^{-1} \left(1 - p_c \right) \cdot p_c \right).$$

Myerson w swojej pracy [3] pokazał, że jeśli $\mathbb{P}\left[OPT \text{ przewozi } c\right] = p_c$, to $\mathbb{E}\left[OPT\left(c\right)\right] \leq \left(F_c^{-1}\left(1 - p_c\right) \cdot p_c\right)$ dla każdego klienta c.

To oznacza ostatecznie, że wektor $\left(p_c\right)_{c \in C} = \left(\mathbb{P}\left[OPT \text{ przewozi } c\right]\right)_{c \in C}$ jest rozwiązaniem dopuszczalnym programu (PRICE-MATCH-LP) o wartości nie mniejszej niż wartość $\mathbb{E}\left[OPT\right]$, co kończy dowód. □

Wobec powyższego lematu oraz z Twierdzenia 4, Algorytm 1 oblicza 2-aproksymację porównując do optymalnego mechanizmu aukcji *Bayesowskiej.*

Rozdział 5.

Podsumowanie i otwarty problem

W powyższej pracy zaprezentowaliśmy rozwiązanie problemu *ride-hailingu*. Przedstawiliśmy podejście i rozwiązanie tego problemu autorstwa Hikimy et. al, którzy osiągnęli 3-aproksymację porównując do mechanizmu aukcji postowanych cen. Następnie pokazaliśmy Algorytm 1, osiągający 2-aproksymację (PRICE-MATCH-LP) porównując do mechanizmu aukcji bayesowskiej co zostało pokazane w Sekcji 4..

W bogatej literaturze dotyczącej optymalizacji stochastycznej pojawia się wiele wyników, które wskazują na to, że możliwym jest, aby uzyskać mechanizm $\left(1 - \frac{1}{e}\right)$-aproksymacyjny korzystając z faktu, że struktura kombinatoryczna, na której pracujemy to skojarzenia. Pozostawiamy więc otwartym pytanie, czy da się poprawić 2-aproksymację na $\left(1 - \frac{1}{e}\right)$-aproksymację.

Bibliografia

[1] Martin Grötschel, Lovász László, and Alexander Schrijver. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica*, 1:169–197, 06 1981.

[2] Yuya Hikima, Yasunori Akagi, Hideaki Kim, Masahiro Kohjima, Takeshi Kurashima, and Hiroyuki Toda. Integrated optimization of bipartite matching and its stochastic behavior: New formulation and approximation algorithm via min-cost flow optimization. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(5):3796–3805, May 2021.

[3] Roger B. Myerson. Optimal auction design. *Mathematics of Operations Research*, 6(1):58–73, 1981.

[4] Qiqi Yan. Mechanism design via correlation gap. *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms*, 08 2010.