Algorithmic Game Theory and Mechanism Design

אור דגמי, http://digmi.org

26 בפברואר 2014

סיכום הרצאות של ד"ר שחר דובזינסקי בשנת לימודים 2014.
ספרים רלוונטים:

1. Algorithmic Game Theory.

- נשתמש בייחוד בחלק 2 שנקרא Mechanism Desgin.

תוכן עניינים

1 מבוא 3
1.1 תורת המשחקים 3
1.2 ngiseD msinahceM 3

2 מבוא לתורת המשחקים 4
2.1 דוגמאות 4
2.1.1 דילמת האסיר 4
2.1.1.1 פורמליזם 4
2.1.2 מלחמת המינים 5
2.1.3 הטלת מטבע 5
2.2 משפט נאש 5
2.3 מכרזים 6
2.3.1 המטרות שלנו 7
2.3.2 מודלים 7
2.4 Revelation Principle 9
2.4.1 Revenue Equivalence 9
2.5 משפט מיירסון משנת '81 (Myerson) 10
2.5.1 הערכה וירטואלית 10
2.5.2 משפט Mayerson 11
2.6 Lookahead Auction 12
2.7 שאלות פתוחות 14
2.8 מקרים חריגים 14
2.8.1 Combinatorial Auction 14
2.8.2 Routing 14

3 VCG (Vickrey, Clarke, Groves) 16
3.1 מה אנחנו רוצים להוכיח? 16
3.2 בעיית ה Bridge Building 17
3.3 snoitcuA lairotanibmoC 18
3.4 noitcuA tinu-itluM 19
3.5 melborP snoitcelE 23
3.6 בעיית Scheduling 26
3.7 Walrasian equilibrium/Competitive equilibrium 29
3.7.1 Subadditive valuations 30
3.7.2 Gross subtitutes (GS) 30
3.8 הפרסומות של גוגל... 32

פרק 1

מבוא

24/10/

1.1 תורת המשחקים

הנושא הזה מקיף אותנו ביום יום, איך לחלק משאבים בין מתחרים רבים בצורה טובה ויעילה.
לדוגמה Google Adwards או חלוקת תדרים של רשתות סלולריות, רשתות P2P ועוד.
הדוגמאות שניתנו בכיתה:

1. Ebay, Amazon, Expedia
2. Google Adwords
3. Social Networks
4. Peer to peer Networks
5. Spectrum auctions

1.2 Mechanism Design

כאן הכל יותר הנדסי, לא בעיות תיאורתיות. ה"טריק" שלנו בקורס הוא שאנחנו הולכים להוסיף אלגוריתמים.

פרק 2

מבוא לתורת המשחקים

2.1 דוגמאות

2.1.1 דילמת האסיר

כמו בכל קורס בתורת המשחקים חובה להתחיל בדילמת האסיר.

	C	D
C	$(-1,-1)$	$(-5,0)$
D	$(0,-5)$	$(-3,-3)$

ישנם שני שחקים, שחקן השורות ושחקן העמודות. כל אחד יכול להלשין על השני או להגיד שהוא לא יודע מי עשה את השוד. אם שניהם לא מודים אז שניהם הולכים לכלא לשנה (העמודה של C,C) אם שניהם אומרים שהשני עשה אז שניהם יכלו לכלא ל3 שנים (העמודה של D,D), ואם רק אחד מאשים את השני אז השני ילך לכלא ל5 שנים, עמודות C,D ו D,C תלוי מי האשים את מי.

במשחק במקרה הזה, השחקנים ילשינו אחד על השני, מכיוון שאם שחקן העמודות יגיד שהוא לא יודע מי עשה את זה, אז לשחקן השורות יש 2 אפשרויות: או להגיד שהוא לא יודע וללכת לשנה בכלא, או להגיד שהשני וללכת במקרה הגרוע ל5 שנים.

ברור כי אם הם תיאמו מראש עדיף לשניהם להגיד שהם לא יודעים.

2.1.1.1 פורמליזם

יש לנו קבוצה של N שחקנים. לכל שחקן i יש קבוצה S_i של אסטרטגיות.

את הרווח של השחקן ה i נסמן:

$$\forall i \quad U_i : S_1 \times S_2 \to \mathbb{R}$$

(לעניות דעתי, מדובר במקרה של המשחק הספציפי עם שני שחקנים, במשחק כללי חסרים שם כל ה S_i של יתר השחקנים)

הערה 2.1.1 מדובר במשחק של Complete Information כלומר אנו יודעים את כל הנתונים. בד"כ, זה לא המצב שלנו.

הגדרה 2.1.2 בהינתן S_{-i} (כל האסטרטגיות של השחקנים האחרים פרט לשחקן הi), נגיד ששחקן i משחק best response אם הוא משחק $s \in S_i$ כך שלכל $s' \in S_i$ מתקיים:

$$u_i\left(s, S_{-i}\right) \geq u_i\left(s', s_{-i}\right)$$

הגדרה 2.1.3 $s \in S_i$ נקרא Dominant (עבור שחקן i) אם לכל $s' \in S_i$, ועבור $s_{-i} \in \prod_{j \neq i} S_j$ מתקיים:

$$u_i\left(s, s_{-i}\right) \geq u_i\left(s', s_{-i}\right)$$

הערה 2.1.4 כלומר: $s\in S_i$ דומיננטי (עבור i) אם s הוא best response לכל S_{-i}.

הגדרה 2.1.5 פרופיל אסטרטגי $s=(s_1,\ldots,s_n)$ נקרא dominant strategies equilibrium אם לכל S_i הוא דומיננטי עבור i.

2.1.2 מלחמת המינים

	A	B
A	$(2,1)$	$(0,0)$
B	$(0,0)$	$(1,2)$

נגיד שיש לנו זוג (בן ובת בלי הגבלת הכלליות) שרוצים ללכת לסרט, A או B, האישה רוצה ללכת לסרט B ואילו הבן רוצה ללכת לסרט A (לכן הוא מקבל 2 utility במקרה הזה ואילו הבת מקבלת 1, בניגוד לשני שזה הפוך).

נבחין כי במשחק הזה אין אסטרטגיה דומיננטית, לפעמים עדיף לשחק A ולפעמים B תלוי מה השחקן השני עשה.

הגדרה 2.1.6 פרופיל אסטרטגיה $\vec{s}=(s_1,\ldots,s_n)$ נקרא Nash Eq. (שיווי משקל נאש) אם לכל i, s_i הוא התגובה הטובה ביותר ל s_{-i}.

הערה 2.1.7 נבחין כי מדובר בשיווי משקל אשר חלש מהשיווי משקל הדומיננטי.

באיזשהו מובן לא פורמלי, שיווי משקל נאש הוא יציב, לאף אחד מהשחקנים אין סיבה לסטות כי הוא רק ידפק מזה (אלא אם כמה ישנו).

2.1.3 הטלת מטבע

	T	H
T	$(0,1)$	$(1,0)$
H	$(1,0)$	$(0,1)$

מטילים מטבע, שחקן אחד מהמר על מה יצא (עץ או פאלי), אם הוא צודק הוא מקבל שקל, אם הוא טועה השני מקבל שקל.

נבחין כי אין כאן שיווי משקל נאש.

נסתכל על משחק טיפה שונה, נסתכל על טבלה של ההתפלגויות על $\{0,1\}$ בעמודות ובשורות.

לכל שחקן יש לנו סט של אסטרטגיות. נחשב את תוחלת התועלת עבור כל התפלגות.

השיווי משקל נאש של משחק זה נקרא ה Mixed Nash של המשחק המקורי.

2.2 משפט נאש

משפט 2.2.1

תחת תנאים מתונים, *Mixed Nash* תמיד קיימים.

31/10/

תזכורת 2.2.2 היו לנו n שחקנים, וS_i אסטרטגיות לכל i. $U_i(S_1,\ldots,S_n)\to\mathbb{R}$.
s נקרא best response אם:

$$u_i(s,s_{-i})\geq u_i(s',s_{-i})\quad\forall s'\in S_i$$

כמו כן הגדרנו dominant strategy אם: s היא best response לכל s_{-i} (כלומר, לא משנה מה השחקנים האחרים משחקים, כדאי לנו לשחק s).

בנוסף הגדרנו Dominant strategy equalibrium במקרה של $(s_1,\ldots,s_n)$ וs_i היא Dominant strategy לכל i.

לסיום, הגדרנו Nash Equalibrium במקרה של $(s_1,\ldots,s_n)$ אזי s_i היא best response עבור s_{-i}.

נסמן $s_i\Rightarrow\tilde{S}_i=\Delta S_i$ (מדובר באוסף ההתפלגויות על ה S_i־ים).

$$u'_i\left(\tilde{S}_1,\ldots,\tilde{S}_n\right)=\sum_{(s_1,\ldots,s_n)}\prod\Pr_{\tilde{S}_i}[s_i]\,u_i(s_1,\ldots,s_n)$$

2.3 מכרזים

הגדרה 2.3.1 First price auction: כל שחקן נותן אישהו bid שיסומן b_i, המוצר שלנו הולך לשחקן עם ההצעה הכי גבוהה, והתשלום שהוא ישלם זה מה שהוא אמר, דהיינו b_i.

הגדרה 2.3.2 Second price auction: כל שחקן נותן אישהו bid שיסומן b_i, המוצר שלנו הולך לשחקן עם ההצעה הכי גבוהה, והתשלום שהוא ישלם זו ההצעה השניה בגובהה.

הגדרה 2.3.3 English auction: אנו מתחילים במחיר $p=0$ ומעלים את המחיר בצורה הדרגתית. כל שחקן יכול לפרוש (אבל לא יכול להצטרף, הוא רק יכול להגיד: "אני בחוץ"). המנצח הוא זה שנשאר אחרון ומשלם את המחיר הנוכחי ברגע עזיבת השחקן האחד לפני אחרון.

הגדרה 2.3.4 Dutch auction: מתחילים במחיר מאוד גבוה, $p=\infty$. המחיר יורד עד שיש שחקן אשר קונה את המוצר.

איזה מכרז יותר טוב לנו לאיזה צורך (למוכר\לקונה)?

נניח שלכל bidder i יש ערך v_i (כמה המוצר שווה לשחקן, לדוגמה אם מדובר בתמונה להקונה הוא שוכר אומנות ויש לו קונה שמוכן לשלם עליה 2 מליון דולר, אז v_i יהיה 2 מליון דולר). אנו נגדיר ערך 0 במקרה שלא קיבל את המוצר (כלומר אין נזק אם הוא לא מנצח בauction).

נגדיר את הutility שלנו להיות $v_i - p$ (כאשר p זה המחיר ששילם המנצח). המטרה היא למקסם את הutility.

במקרה של First price auction : במקרה זה ברור שלא כדאי לנו להציע סכום שעולה על v_i, אבל מה כדאי לנו כן להציע?

במקרה של שני שחקנים, כאשר השחקנים יודעים את הערך של v_1, v_2 . האטסטרגיות של השחקן היא איזה bid לתת. כלומר אנו מסתכלים על הטבלה:

$$\begin{array}{cccc} & b_2=0 & b_2=1 & b_2=2 & \ldots \\ b_1=0 \\ b_1=1 \\ b_1=2 \\ \vdots \end{array}$$

כל איבר בטבלה הזו יציין את הutility של השחקנים. אם $b_1 > b_2$ אז:

$$\begin{aligned} u_1(b_1,b_2) &= v_1 - b_1 \\ u_2(b_1,b_2) &= 0 \end{aligned}$$

אחרת:

$$\begin{aligned} u_1(b_1,b_2) &= 0 \\ u_2(b_1,b_2) &= v_2 - b_2 \end{aligned}$$

בה"כ נניח כי $v_1 \geq v_2$.

למה 2.3.5

הNE (*Nash Equalibrium*) היחידי במשחק הזה הוא כאשר: $b_1=b_2=v_2$.

נבחין כי במצב זה, שחקן שני יכול לעלות את המחיר, אבל אז הוא יפסיד כסף במקום רווח אפס, כלומר זה לא משתלם לו. הוא יכול גם להוריד את המחיר אבל אז הוא יפסיד את המוצר לכן עדיף לו כבר לתת את המחיר המלא של המוצר.

במקרה של Second price auction: **למה 2.3.6**

$v_i=b_i$ הוא *dominant strategy*.

במקרה של English auction: כל שחקן יצא כאשר $p=v_i$. המנצח יקבל את המוצר במחיר $\max_{j\neq i} v_j$.

במקרה של Dutch auction: יקנו במחיר של $v_i - 1$.

2.3.1 המטרות שלנו

המטרות שלנו הן:

1. יעילות (Efficiency) - השחקן עם הערדך הגבוה ביותר מקבל את המוצר.
2. למקסם את התמורה (Revenue).

2.3.2 מודלים

Private value: לכל שחקן יש את v_i שלו, והוא סודי, הוא לא מוכן לגלות אותו.

Distributional Value: $v_i \sim D_i$. נניח כי כל השחקנים יודעים כל D_i ושהם I.I.D.

במקרים של English Auction ו- Second Price Auction לא אכפת לשחקן מה ההחלטה של השחקנים האחרים (כאשר מדובר ב private value).

<u>מה קורה כאשר אנו מדברים על Distributional Value?</u>

נניח שיש לנו שני סוגי שחקנים, סוחר וחובב. כאשר בהסתברות 80% הוא סוחר ו20% הוא חובב. אם שחקן יודע שהוא סוחר, אז בהסתברות 0.2 הוא ישחק נגד חובב ו0.8 נגד סוחר. הוא מעדיף לשחק אסטרטגיות שונות בכל אחד מהמקרים. כעת הutility הוא utility ממוצע. כנ"ל הbest response ושיווי המשקל.

מה יהיה הRevenue במקרה של Second price auction?

במקרה זה מדובר בתוחלת של המינימום בין שני משתנים מקריים. במילים אחרות:

$$\mathbb{E}\left[v \mid v < u\right]$$

במקרה של שני משתנים שמתפלגים אחיד בין $0, 1$, התוחלת המינימלית תהיה $\frac{1}{3}$.

מה קורה עם First Price Auction?

נניח שיש לנו פונצקיית התפלגות F (c.d.f - cumulative distribution function) כלומר: $F(\alpha)$ היא ההסתברות שהערך קטן מα. את הנגזרת שלה נסמן $F' = f$. אנו מניחים כי אנחנו השחקן עם הערך הכי גבוהה, ואנחנו מנסים לשאול את עצמנו: מה יהיה הערך של השחקן השני? $\beta(v) = \mathbb{E}\left[u \mid u < v\right]$.

<u>טענה 2.3.7</u>

זהו NE.

<u>מסקנה 2.3.8</u>

ה*Revenue* במקרה של *First Price* ו *Second Price* זהה.

נוכיח את הטענה: **הוכחה:** נסמן:

$$\beta(v_1) = \mathbb{E}\left[v_2 \mid v_2 < v_1\right]$$

נניח שלשחקן 1 יש ערך v_1 והוא נותן bid עם הערך b. אנו רוצים לחשב את הפונקציה: $\text{profit}(b, v_1)$.

אנו רוצים לשאול את עצמנו, איזה שחקן יתן הצעה b, כלומר:

$$v_1' = \beta^{-1}(b) \iff b = \beta(v_1')$$

במקרה זה פונקציית הרווח תהא:

$$\text{profit}(b, v_1) = F(v_1')(v_1 - b)$$

כאשר $F(v_1')$ היא הסתברות הזכייה ו $(v_1 - b)$ הרווח במקרה של זכייה. 07/11/

תזכורת 2.3.9 אינטגרציה בחלקים:

$$\int uv' = uv - \int u'v$$

זאת מכיוון ש:

$$(uv)' = u'v + uv'$$

לכן, מאינטגרציה והמשפט היסודי נקבל:

$$uv = \int (uv)' = \int u'v + \int uv'$$

ומהעברת אגפים נקבל את נוסחת האינטגרציה בחלקים.
תוחלת מותנית (Conditional Expectation) נתונה לפי:

$$\mathbb{E}\left[v_2 \mid v_2 < v_1'\right] = \frac{\int\limits_0^{v_1'} yf\left(y\right)dy}{F\left(v_1'\right)}$$

הערה 2.3.10 התוחלת של הערכים עד v_1' חלקי ההסתברות שקיבלנו אחד מהם. האנלוג הדיסקרטי הוא במקרה של קוביה, התוחלת עד שיצא 3 היא:

$$\frac{\frac{1}{6}\left(1+2+3\right)}{\frac{1}{2}} = 2$$

אנו רוצים להראות כי:

$$\text{profit}\left(\beta\left(v_1\right), v_1\right) - \text{profit}\left(\underbrace{\beta\left(v_1'\right)}_{b}, b_1\right) \geq 0$$

נבחין כי:

$$\text{profit}\left(b, v_1\right) = F\left(v_1'\right)v_1 - F\left(v_1'\right)\mathbb{E}\left[v_2 \mid v_2 < v_1'\right] = F\left(v_1'\right)v_1 - \int_0^{v_1'} yf\left(y\right)dy$$

מאינטגרציה בחלקים נקבל:

$$= F\left(v_1'\right)v_1 - F\left(v_1'\right)v_1' + \int\limits_0^{v_1'} F\left(y\right)dy = F\left(v_1'\right)\left(v_1 - v_1'\right) + \int\limits_0^{v_1'} F\left(y\right)dy$$

כעת, נבחין כי:

$$\text{profit}\left(\beta\left(v_1\right), v_1\right) - \text{profit}\left(\underbrace{\beta\left(v_1'\right)}_{b}, b_1\right) = F\left(v_1'\right)\left(v_1' - v_1\right) - \int\limits_{v_1}^{v_1'} F\left(y\right)dy$$

אבל זה ברור מכיוון ש F היא c.d.f, כלומר היא מונוטונית עולה ולכן אם $v_1' > v_1$ אז ברור שהביטוי משמאל חוסם את האינטגרל מלמעלה, ואם הוא קטן יותר ברור שהוא חוסם אותו מלמטה, אבל אנחנו מקבלים סימן מינוס (גם באינטגרל) ולכן בכל מקרה נקבל משהו אי שלילי. כנדרש. ■

2.4 Revelation Principle

משפט 2.4.1

בהינתן מכרז ושיווין משקל, אזי קיים מכרז אחר כך ש:

1. זה שיווי משקל לכל שחקן לשחק את הערך האמיתי שלו.
2. התוצאה היא זהה לתוצאה של המכרז המקורי.

נוטציה:
אנו מסמנים את $x:\mathbb{R}^n\to D$ את פונקציית האלוקציה (כאשר $D\subseteq\mathbb{R}^n$), שמראה את הסתברות של קבלת המוצר לכל שחקן.
ו$p_i:\mathbb{R}^n\to R$ היא פונקציית התשלום, כמה השחקן ה i משלם?

למה 2.4.2 מונוטוניות

נקבע שחקן i. יהי $v\geq v'$ אזי:

$$\mathbb{E}_{v-i}\left[x_i\left(v,v_{-i}\right)\right]\geq\mathbb{E}_{v_{-i}}\left[x_i\left(v',v_{-i}\right)\right]$$

הוכחה: התשלום הממוצע ששחקן i משלם הוא:

$$P\left(V\right)=\mathbb{E}_{v_{-i}}\left[p_i\left(v,v_{-i}\right)\right]$$

■

$$Pr\left[win\right]\cdot\left[value-price\right]$$

$$\begin{aligned}vx\left(v\right)-p\left(v\right)&\geq x\left(v'\right)v-p\left(v'\right)\\ v'x\left(v'\right)-p\left(v'\right)&\geq x\left(v\right)v'-p\left(v\right)\end{aligned}$$

מחיבור המשוואות נקבל:

$$x\left(v\right)\left(v-v'\right)\geq x\left(v'\right)\left(v-v'\right)$$

ולכן נקבל כי:

$$x\left(v\right)\geq x\left(v'\right)$$

2.4.1 Revenue Equivalence

טענה 2.4.3

נקבע מכרז (עם פונקציות p,x), לכל שיווי משקל של המכרז התוחלת של התשלום של שחקן i הינה:

$$p_i\left(v\right)=x_i\left(v,v_{-i}\right)v-\int_0^v x_i\left(t,v_{-i}\right)dt+c\left(v_{-i}\right)$$

הוכחה:

$$\begin{aligned}v\cdot x\left(v\right)-p\left(v\right)&\geq x\left(v+\varepsilon\right)-p\left(v+\varepsilon\right)\\ \left(v+\varepsilon\right)x\left(v+\varepsilon\right)-p\left(v+\varepsilon\right)&\geq\left(v+\varepsilon\right)x\left(v\right)-p\left(v\right)\end{aligned}$$

$$v\left[x\left(v\right)-x\left(v+\varepsilon\right)\right]\geq p\left(v\right)-p\left(v+\varepsilon\right)\geq\left(v+\varepsilon\right)\left[x\left(v\right)-x\left(v+\varepsilon\right)\right]$$

נחלק את הכל ב ε ונקבל:

$$v\frac{\left[x\left(v\right)-x\left(v+\varepsilon\right)\right]}{\varepsilon}\geq\frac{p\left(v\right)-p\left(v+\varepsilon\right)}{\varepsilon}\geq\left(v+\varepsilon\right)\frac{\left[x\left(v\right)-x\left(v+\varepsilon\right)\right]}{\varepsilon}$$

ומהשאפה של אפסילון ל 0 (והכפלה במינוס 1 כי בתכלס זה המינוס של הנגזרת) נקבל:

$$v\cdot x'\left(v\right)\geq p'\left(v\right)\geq vx'\left(v\right)$$

כלומר:

$$p'\left(v\right)=vx'\left(v\right)$$

ולכן:

$$p\left(v\right)=p\left(0\right)+\int_{0}^{v}tx'\left(t\right)dt$$

ומאינטגרציה בחלקים נקבל:

$$p\left(v\right)=p\left(0\right)+vx\left(v\right)-\int_{0}^{v}x\left(t\right)dt$$

■

21/11/

תזכורת 2.4.4 ראינו כי Second price נותן לנו revenue של $\frac{1}{3}$ וכי $\frac{1}{2}$ S.P. reserve נותן של $\frac{5}{12}$.

הערה 2.4.5 קיימת שקילות בין truthful למונוטוניות. כלומר, על מנת להראות שMechanism מסווים הוא Truthful, מספיק להראות שהוא מונוטוני, דהיינו:

$$v>v'\Rightarrow x_{i}\left(v\right)\geq x_{i}\left(v'\right)$$

2.5 משפט מיירסון משנת 81' (Myerson)

2.5.1 הערכה וירטואלית

המקרה הראשון: מוצר אחד ושחקן אחד. הערך של המוצר עבור השחקן מגיע מהתפלגות F (עם צפיפות f). אנו מחפשים את המחיר p אשר ממקסם את:

$$\max\left(p\left(1-F\left(p\right)\right)\right)$$

ההסתברות שהערך של השחקן גדול מp היא $1-F\left(p\right)$, ואם הערך של השחקן גדול מ p הוא ישלם p.

נסמן:

$$p\left(q\right)=\text{in what price the bidder accepts with probability } q.=F^{-1}\left(1-q\right)$$

הערה 2.5.1 נסמן ב p כך ש $q = 1 - F(p)$ ואז נקבל $p = F^{-1}(1-q)$.

הRevenue שלנו הוא: $p(q)\,q$. על מנת למקסם את הערך, אנו רוצים לגזור לפי q דהיינו:

$$\frac{d}{dq}\left(p(q)\,q\right) = F^{-1}(1-q) - \frac{q}{F'\left(F^{-1}(1-q)\right)}$$

הערה 2.5.2 כאשר אנחנו משתמשים בכלל הגזריה עבור פונקציה הופכית, דהיינו אם $x = g(f(x)) = f(g(x))$ אז:

$$g'(x) = \frac{1}{f'(g(x))}$$

ותחת ההגדרה של p נקבל:

$$= p - \frac{1 - F(p)}{f(p)}$$

נסמן את זה בתור:

$$= \phi(p)$$

ונדרוש ש $\phi(p) = 0$, וזה נקרא הערכה וירטואלית (Virtual Valuation).

דוגמה 2.5.3 : במקרה של התפלגות האחידה בקטע $[0,1]$:

$$0 \stackrel{!}{=} \phi(p) = p - \frac{1-F(p)}{f(p)} = p - \frac{1-p}{1} \Rightarrow 2p = 1 \Rightarrow p = \frac{1}{2}$$

2.5.2 משפט Mayerson

משפט 2.5.4 משפט Mayerson

The expected revenue of a mechanism is its expected virtual surplus:

$$\mathbb{E}\left[\sum_i \phi(v_i)\, x_i(\vec{v})\right]$$

נניח שלכל שחקן יש ערך v_i מציבים אותם בנוסחא של Virtual Valuation, ונותנים את זה לשחקן עם הערך הכי גבוה. האם זה optimal? האם זה truthful?

למעשה משפט mayerson אומר שזה הערך האופטימלי.

נניח כעת שv_i הם משתנים מקריים i.i.d, אם כולם באים מההתפלגות האחידה, אז ה ערך הויראוטלי שלהם הולך להיות חיובי אם"ם זה גדול מ$\frac{1}{2}$. בעצם, עשינו Second price auction עם Reserve price של חצי.

אנו נדבר על "regular distributions" כלומר על מקרים שהvirtual valuation הוא מונוטוני לא יורד.

תחת הנחה זו, מדוע זה Thruthful, אנו מקבלים את זה מהמונוטוניות, מכיוון שאז ככל ש v_i עולה כך ה virtual valuation עולה, כך שמשתלם להציע כמה שיותר על מנת לנצח בauction.

למה 2.5.5

The expected payment of player i is: $\mathbb{E}\left[\phi(x_i)\, x_i(v)\right]$

הערה 2.5.6 נבחין כי זה נותן לנו את משפט מיירסון באופן טריוויאלי מהלינאריות של התוחלת.

הוכחה: לשם הנוחות, נניח כי ההתפלגות של השחקן היא בין 0 ל h. למעשה, אנו רוצים לחשב את הביטוי:

$$\int_{0}^{h} p(b) f(b)\, db = \int_{0}^{h} bx(b) f(b)\, db - \int_{b=0}^{h} \int_{t=0}^{b} x(t) f(b)\, dtdb$$

כאשר אנחנו משתמשים כאן ב Revenue Equivalence. נשנה סדר סכימה:

$$= \int_{0}^{h} bx(b) f(b)\, db - \int_{t=0}^{h} x(t) \int_{b=t}^{h} f(b)\, dbdt$$

אבל נבחין כי: $\int_t^h f(b)\, db = 1 - F(t)$. נבצע שינוי שם משתנים ($b = t$) ונקבל:

$$= \int_{0}^{h} bx(b) f(b)\, db - \int_{b=0}^{h} x(b)(1 - F(b))\, db$$

כעת נחבר את האינטגרלים לאינטגרל אחד ונקבל:

$$= \int_{0}^{h} \underbrace{\left[b - \frac{1 - F(b)}{f(b)}\right]}_{\phi(b)} x(b) f(b)\, db = \mathbb{E}_{\vec{v}}\left[\phi(v)\, x(v)\right]$$

כנדרש. ■

חסרונות:

1. אנחנו חייבים לדעת את ההתפלגויות, חייבים לדעת לחשב את ה virtual values.
2. לאורך כל הדרך, הנחנו אי תלות בהתפלגויות. 28/11/

אפליקציה אפשרית: לתת את הitem לשחקן עם הvirtual value הכי גבוה.
נשאלת השאלה האם זה Thruthful? "רגולרי" שקול למונוטוניות של virtual value.

2.6 Lookahead Auction

$\frac{1}{2}$ – approx.
נגיד שיש לנו שחקן אחד כאשר v_1 מתפלג אחיד בקטע $[2,3]$. ונניח כי $v_2 = 2^{100v_1}$. מה הוא ה Optimal Auction?
לשאול את v_1 מה הערך שלו, לא נותנים לו כלום (ואומרים לו מראש), ואז משתמשים בערך הזה על מנת למכור לv_2.
מה המכרז הזה אומר?

1. For every instnace, find the bidder with the highest value (wlog, $v_1 \geq v_2 \geq \ldots \geq v_n$)
2. Find the marginal distribution of v_1 given $v_2, \ldots, v_n$.
3. Take it or leave it offer to player 1 at the best price p given D_{v_1} (The best price is the price that maximaize the seller revenue).

טענה 2.6.1

$p \geq v_2$ (בכל *instance*).

טענה 2.6.2

The mechanism is truthful

הוכחה: נסתכל על השחקן עם הערך הגבוה ביותר.

1. כל bid מעל v_2 זהה.
2. כל bid מתחת ל v_2 הוא הולך להפסיד.

נסתכל על שחקן אחר:

1. כל bid מתחת לv_1 יגרום לו להפסיד.
2. כל bid מעל v_1 יגרום ל take it or leave it עם הצעה גדולה מ v_1.

■

למה 2.6.3

The lookahead auction is the revenue-max auction that may only sell the item to the bidder with the highest value

הוכחה:

1. קח את המוצר במחיר 1.
2. קח חצי מהמוצרה במחיר $\frac{1}{4}$.

אנו שואלים מתי:

$$v - 1 \geq \frac{1}{2}v - \frac{1}{4}$$

וזה כמובן:

$$v \geq 1.5$$

טענה 2.6.4

המכרז האופטימלי(*Revenue-max*) לשחקן יחיד הוא *Take-it-or-leave-it*.

הוכחה: מיירסון. ■

■

טענה 2.6.5

The expected revenue of the lookahead auction is at least 1/2 of the expected optimal revenue

הוכחה: נסמן: OPT - משתנה מקרי שמקבל את הRevenue של ה Optimal Auction.
A - המאורע שהoptimal auction מוכר את המוצר לhighest bidder.
OPT_1 - הRevenue של הOptimal Auction אם A ואחרת 0.
OPT_2 - הRevenue של הOptimal Auction אם A לא קרה ואחרת 0.
הבחנה פשוטה:

$$\text{OPT} = \text{OPT}_1 + \text{OPT}_2$$

נסמן ב ALG את הrevenue של lookahead. אנו רוצים להוכיח כי:

$$\mathbb{E}[\text{ALG}] \geq \frac{\mathbb{E}[\text{OPT}]}{2}$$

ספציפית נוכיח כי:

$$\mathbb{E}\left[\mathrm{ALG}\right] \geq \mathbb{E}\left[\mathrm{OPT}_1\right]$$

וגם כי:

$$\mathbb{E}\left[\mathrm{ALG}\right] \geq \mathbb{E}\left[\mathrm{OPT}_2\right]$$

ומשניהם יחדיו נקבל כי:

$$2\mathbb{E}\left[\mathrm{ALG}\right] \geq \mathbb{E}\left[\mathrm{OPT}\right]$$

וזה הנדרש.

נוכיח כי $\mathbb{E}\left[\mathrm{ALG}\right] \geq \mathbb{E}\left[\mathrm{OPT}_2\right]$. נבחין כי:

$$\mathbb{E}\left[\mathrm{OPT}_2\right] \leq \mathbb{E}\left[v_2\right]$$

אם אנו משתמשים ב Take-it-or-leave-it של v_2 אז התוחלת של הrevenue הינה: $\mathbb{E}\left[v_2\right]$. אז זהו חסם תחתון עבור ה revenue הכי טוב בהצעת TIOLI.

נראה כי $\mathbb{E}\left[\mathrm{ALG}\right] \geq \mathbb{E}\left[\mathrm{OPT}_1\right]$.

נסתכל על הoptimal auction, ונעשה בדיוק מה שהיא עושה אם בשחקן עם הערך הגבוה ביותר מקבל את המוצר, אחרת אף שחקן לא מקבל את המוצר.

אז: $\mathbb{E}\left[\mathrm{ALG}\right] \geq \mathbb{E}\left[\mathrm{OPT}_1\right]$ מכיוון שהLookahead הוא האופטימלי שמוכר את המוצר במחיר הגבוה ביותר. ■

2.7 שאלות פתוחות

1. Better approx ratio? Compute optimal auction is NP-hard. Better approx ratio in poly-time.
2. Determinism vs radomness.

05/

2.8 מקרים חריגים

נתאר כמה מצבים שהפורמולציה עד עכשיו של המכרזים פשוט לא תפסה אותם.

2.8.1 Combinatorial Auction

מה קורה כאשר יש לנו מוצרים עם תלות? לדוגמה מוכרים שני נעליים, נעל ימין ונעל שמאל. לכל אחת לחוד אין ערך, אבל ביחד יש להן.

או לחילופין מקרים שבהם השלם לא שווה לסכום מרכיביו (נגיד סט של עט ועפרון יחד יכול להיות שווה לקונה יותר מאשר כל אחד לחוד).

2.8.2 Routing

נתבונן בגרף, כאשר יש לנו Nodes שהם Source, נסמנם ב S ו Nodes שהם T וכאלה ריקים, נגיד ש S הוא מפעל ו T הוא לקוח.

נגיד גם שהצלעות בגרף הם כבישים, ונניח שהם עמוסים מידי, כך שכל כביש יכול לשמש מפעל אחד בלבד. אפשר לחשוב על זה כמכרז של מכירת כבישים. נבחין כי זה סוג של מקרה פרטי של המקרה הקודם.

Elections יש לנו n שחקנים ו m אלטרנטיביות, נאמר ששחקן i יש ערך $v_i(j)$ לאלטרנטיבה j.

דוגמה 2.8.1 : במקרה של Single item auction, יש לנו $n+1$ אלטרנטיבות (candidates) מכיוון שהמוצר יכול להגיע לכל אחד מהשחקנים או לא לחולק כלל. ולשחקן i יש ערך:

$$v_i(j)=\begin{cases}0 & i\neq j\\ v_i & i=j\end{cases}$$

Chapter 3

VCG (Vickrey, Clarke, Groves)

3.1 מה אנחנו רוצים להוכיח?

אנו רוצים להראות את המשפט הבא:

משפט 3.1.1

קיים *Mechanism* שהו *Efficient* ל*Election Problem* שהוא *Truthful*.

ההגדרה כאן ל efficient הינה:

$$\max_j \sum v_i(j)$$

בVCG מתקיים:

1. לכל שחקן יש bid שהוא b_i.
2. מוצאים את ה efficient outcome שיוסמן ב o (according to the b_i).
3. לשלם לכל שחקן i את הסכום: $\sum\limits_{j\neq i} v_j(o)$.

אנו נראה שזה Mechanism שהוא Truthful:

למה 3.1.2

ה*dominant strategy* של כל שחקן היא: $b_i(\cdot) = v_i(\cdot)$.

הוכחה: הרווח של כל שחקן i מbidding של $b_i(\cdot)$ ותוצאה o הוא הwelfate(value) של התוצאה:

$$v_i(o) + \sum_{j\neq i} b_j(o)$$

אז אם i בוחר $b_i(\cdot) \neq v_i(\cdot)$ כל מה שיכול לקרות זהש o' יבחר אבל אז:

$$\text{welfare}(o') \leq \text{welfare}(o)$$

■

Individual rational: If no player ever loses from participating.

פונקציית התשלום: נסמן o_{-1} בתור הefficient outcome כאשר $v_i(j) = 0$ לכל j. אז מכל שחקן i לוקחים:

$$\sum_{j\neq i} v_j(o_{-1}) - \sum_{j\neq i} v_j(o)$$

נבחין כי:

$$v_i(o) - \left(\sum_{j\neq i} v_j(o_{-i}) - \sum_{j\neq i} v_j(o)\right) \geq 0$$

דהיינו:

$$\sum_j v_i(o) \geq \sum_{j\neq i} v_j(o_{-1})$$

כל שחקן i משלם את ה"נזק" שהוא גרם לחברה.

נתבונן בSecond price auction.

המנצח משלם את הsecond highest bid. והמפסידים משלמים 0. קיבלנו את מה שרצינו.

בתרגיל הראשון היה לנו מכרז של m מוצרים, כל שחקן רצה רק מוצר אחד. המנצחים משלמים את הביד של השחקן ה $m+1$. בה"כ נניח כי:

$$v_1 \geq v_2 \geq v_3 \geq v_4$$

נאמר ש $m = 2$ ונאמר ש $o = \{1, 2\}$, אז אם אנו רוצים את $o_{-2} = \{1, 3\}$. באופן זהה: $o_{-1} = \{2, 3\}$. אז ההבדל בין o_{-i} ל o הוא בדיוק שחקן אחד.

3.2 בעיית ה Bridge Building

דוגמה קצת יותר מסובכת: בעיית ה Bridge Building (או public project).

יש לנו n תושבים שגרים באי מבודד, והממשלה שוקלת לבנות גשר מהאי ליבשה, הממשלה מוכנה רק אם התועלת מהגשר תיהיה גדולה מהעלות שלו. לאנשים שונים יש ערכים שונים.

או באופן "פורמלי": יש לנו n שחקנים עם ערך v_i, הממשלה מוכנה לבנות את הגשר אס"ם $\sum v_i \geq 1$.

יש לנו כאן שתי אלטרנטיבות Y ו N (לבנות או לא לבנות). יש לנו n שחקנים עם v_i. נוסיף עוד dummy player עם ערך $v(y) = 0$ ו $v(n) = 1$ (אפשר להגיד, הממשלה).

טענה 3.2.1

הEfficient outcom של לבנות את הגשר אס"ם $\sum v_i \geq 1$.

הוכחה:

$$\begin{aligned} \text{welfare}(y) &= \sum v_i(Y) = v_i \\ \text{welfare}(n) &= 1 + \underbrace{\sum v_i(N)}_{=0} \end{aligned}$$

■

במקרה של שחקן נורמלי:

1. אם N אז: 0.
2. אם Y אז: אם $\sum_{i\neq j} v_i \geq 1$ שלם 0, אחרת שלם $1 - \sum_{i\neq j} v_j$.

3.3 Combinatorial Auctions

יש לנו n שחקנים ו m מוצרים. נסמן את הקבוצה של השחקנים ב N ושל המוצרים ב M גדול. אנחנו רוצים לבטא את העובדה שיש כאן עסקאות חבילה. לכן למעשה:

$$v_i : 2^M \to \mathbb{R}$$

כלומר לכל קומבינציה של מוצרים אנו מקבלים את ערך כלשהו. בדרך כלל, נניח כי v_i הוא נתון פרטי.

נניח כי v_i מנורמל, דהיינו: $v_i(\varnothing) = 0$. ונניח free disposal (או monotonicity) דהיינו לכל $S \subseteq T$ אז $v_i(T) \geq v_i(S)$.

המטרה שלנו היא למקסם את הwelfare, או במילים אחרות, למצוא חלוקה של המוצרים $(s_1, \ldots, s_n)$ כך ש $\max \sum_{i \in N} v_i(s_i)$.

נאמר ששחקן i הוא unit demand אם לכל s מתקיים $v_i(s) = \max_{j \in s} v_i(\{j\})$.

למה 3.3.1

אם כל השחקנים הם *unit demand* אז קיים אלגוריתם שרץ בזמן פולינומיאלי שמוצא את ה *efficient outcome*.

12/12/

נתבונן במקרה הפרטי של single minded bidders.

מה זה אומר?

הגדרה 3.3.2 שחקן יקרא single minded אם קיים s_i כך ש:

$$v_i(s) = \begin{cases} v_i & s_i \subset s \\ 0 & \text{otherwise} \end{cases}$$

כלומר, אם קיים מוצר מסויים ברשימה, אז הערך שלו הוא v_i, אחרת הוא 0.

משפט 3.3.3

למצוא את ה*optimal allocation* היא בעיה *NP-Hard* עם *SM-bidders*.

יתרה מזאת, זה *NP-Hard* לקרב את הפתרון עם פאקטור $M^{\frac{1}{2}-\varepsilon}$ לכל $\varepsilon > 0$ קבוע.

הגדרה 3.3.4 α-approx. $\text{ALG}(I) \geq \frac{\text{OPT}}{\alpha}$ לכל I,

הוכחה: $G = (V, E)$, $|V| = n$ ו $|E| < M$

בעיית ה Independent set גוררת CA.

משפט 3.3.5 משפט Hastad

זה *NP-Hard* להפריד בין המקרים הבאים:

1. קיים *IS* (*Independent Set*) בגודל $n^{1-\varepsilon}$.
2. כל *IS* הוא לכל היותר n^{ε} לכל קבוע $\varepsilon > 0$.

לכן זה NP-Hard לתת קירוב $n^{1-2\varepsilon}$ מכיוון ש:

$$\text{ALG} \geq \frac{\text{OPT}}{n^{1-2\varepsilon}} \geq \frac{n^{1-\varepsilon}}{n^{1-2\varepsilon}} = n^{\varepsilon}$$

n קודקודים אומר n שחקנים, m צלעות אומר m מוצרים. s_i לכל שחקן i היא הקבוצה של ה adjacent edjes. $v_i(s_i) = 1$.

אבל נבחין כי: $n^2 \geq M$, מכיוון שלכל היותר כל הקודקודים מחוברים לכולם, ולכן אי אפשר לעשות קירוב של שורש M. ■

נראה שזה truthful.

$\sqrt{M}$ קירוב, בזמן פולינומיאלי עבור CA's עם SM bidders.

נניח שv_i של השחקן הi הוא פרטי, אבל ה s_i הוא פומבי. נבחר את הטוב ביותר מהאלוקציות הבאות:

1. לתת את כל המוצרים לשחקן עם הערך v_i הגבוה ביותר ו $|s_i| \geq \sqrt{M}$.
2. אלגוריתם גרידי לכל שחקן עם $|S_i| < \sqrt{M}$.

למה 3.3.6

האלגוריתם הוא *truthful*

הגדרה 3.3.7 אלגוריתם (עבור SM Bidders) הוא מונוטוני אם שחקן מנצח i שמנצח עם ערך v מנצח גם עם $v' \geq v$ לכל v'.

למה 3.3.8

מונוטוניות $\iff$ *truthful*.
יתרה מזאת, התשלום לכל *bidder* מנצח יהיה ה*threshold value* (הערך המינימלי שגורם לו לנצח).

הוכחה: נראה ראשית כי truthful גורר מונוטוניות.

תחילה, לא ייתכן ש i מנצח עם v ו v' עם תשלום שונה. באופן ספציפי, לכל $v > p$, השחקן i אמור לנצח. ולכן זה מונוטוני כנדרש.

כיוון שני, נראה שמונוטוניות גורר truthful.

כיצד? נעשה בדיוק את אותו טריק, המחיר יהיה הערך המינימל יהיה הערך שהוא משלם. השחקן ה i נצח עם $v' \geq p$ ואז השחקן משלם p. נניח שהשחקן הi לא מנצח עם $v'' > p$, אבל אם הוא מצהיר v'' הרווח שלו עבור v'' יהיה 0 אבל עבור v' הוא מקבל: $v' - p > 0$. לכן עדיף לו להגיד v'. ■

למה 3.3.9

האלגוריתם מספק $\sqrt{M}$ קירוב.

הוכחה: $(o_1, \ldots, o_n)$ זה האופטימלי $\text{OPT} = \sum_{i \in \mathbb{N}} v_i(o_i)$.

אז יש לנו שני מקרים:

1. יהי B תת קבוצה של big שחקנים. אם:

$$\sum_{i \in B} v_i(o_i) \geq \frac{\text{OPT}}{2}$$

המספר של big players שמנצח ב OPT הוא $\sqrt{M}$. אז קיים $i \in B$ כך ש $v_i(o_i) \geq \frac{\text{OPT}}{2\sqrt{m}}$.

2. $S = N - B$ (נניח כי $v_1(o_1) \geq v_2(o_2) \geq \ldots$) אז:

$$\sum_{i \in S} V_i(o_i) \geq \frac{\text{OPT}}{2}$$

■

19.12

בשיעור שעבר ראינו כי עבור Combinatorial auctions אין קירוב יותר טוב מאשר $\sqrt{m}$-קירוב, כלומר שזה NP-Hard. יתרה מכך, ראינו שיש אלגוריתם שנותן קירוב $\sqrt{m}$ בזמן פולינומיאלי והוא truthful.

השיעור נסתכל על מקרה פרטי של Combinatorial Auction שהוא Multi-unit auctions.

3.4 Multi-unit Auction

יש לנו n שחקנים וm מוצרים זהים. ויש לנו Valuation $v_i : [m] \to \mathbb{R}$. וכמו כן אנחנו נשתמש בהנחות הרגילות: $v_i(0) = 0$ וכמו כן אנו מניחים free disposal ומונוטוניות (כלומר אם קיבלנו את המוצר ואנחנו לא צריכים אותו אנחנו יכולים להפטר ממנו ללא עלות).

גישה לVals, יש לנו כמה שיטות:

1. נתנו לנו מעגל של input/output, נותנים לו את כמות המוצרים והוא נותן לנו את הערך. "Succinct representation"

- למשל Single minded bidders. $\langle k, v \rangle$ (כאשר k זו הקבוצה של המוצרים) , נבחין כי על מנת לכתוב את המספר k צריך $\log m$ ביטים, ואת המספר v צריך $\log v$, אם $v = O(m)$, הזכרון הוא פולינומיאלי ב $\log m$.

2. Oracles

   (a) Value oracles

   - בהינתן S אנו רוצים לדעת מה זה $V(S)$?

   (b) Demand Oracle

   - בהינתן מחירים $p_1,\ldots,p_m$ מה זו החבילה הכי רווחית: $\arg\max_S V(S) - \sum_{j\in S} p_j$.

   (c) communication complexity

   - אכפת לנו רק מכמה ביטים עוברים בשאלה, לכל שאלה תהיה לנו תשובה.

הגדרה 3.4.1 בעיית ה knapsack: הבעיה הזאת היא בעיה NP-Hard. הבעיה מתואר כדלקמן:
יש לנו n מוצרים, כל אחד מהם יש גודל s_i עם ערך v_i . המטרה שלנו היא למצוא את התת־קובצה עם הערך מקסימלי כאשר הגודל הסופי הוא לכל היותר m.

איך זה קשור? נעשה רדוקציה לבעיה שלנו: במקרה של single minded bidder שרוצה S_i מוצרים עם ערך v_i .
סה"כ המוצרים שיש לנו הוא m .

טענה 3.4.2

יהי $(a_1,\ldots,a_n)$ תוצאה של MUA. בה"כ אפשר להניח כי שחקן i קיבל a_i מוצרים, והוא גם "שמח" אז אפשר להניח ש $a_i = s_i$ אחרת $a_i = 0$.

טענה 3.4.3

בהינתן $(a_1,\ldots,a_n)$ פתרון ל MUA. אז קיים פתרון לבעיית ה$knapsack$ עם ערך $\sum v_i(a_i)$.

הוכחה: אם שחקן i "מנצח" ב MUA מוצר i בקבוצה הנבחרת של הknapsack. ואז אנו מקבלים:

$$\sum a_i \le m$$

מכיוון שיש לנו לכל היותר m מוצרים. ■

הקלט של knapsack הוא פולינומיאלי ב n וב $\log m$. כל עוד $P \neq NP$ אין אלגוריתם שפותר את MUA בזמן פולינומיאלי ב n ו$\log m$.

משפט 3.4.4

יש אלגוריתם ל MUA שרץ בזמן פולינומיאלי (n,m).

הוכחה: אלגוריתם דינמי. נגדיר $M(i,k)$ בתור best value of instance restricted to player $1,\ldots,i$ and k items.
במילים אחרון אנו בונים טבלה של כמות המוצרים וכמות השחקנים, ואז:

$$M(i,k) = \max_{0\le t\le k} \left[v_i(t) + M(i-1,k-t)\right]$$

■

משפט 3.4.5

כל אלגוריתם שפותר את MUA בדיוק, עושה לפחות M שאילתות לקופסאות השחורות (אפילו במקרה של שני שחקנים).

הוכחה:

$$\begin{aligned} v_1(s) &= 2\cdot s \\ v_2^t(s) &= \begin{cases} 2s & s\neq t \\ 2s+1 & s = t \end{cases} \end{aligned}$$

על opt הוא מקבל $2m+1$ ועל כל אלוקציה אחרת הוא מקבל $2m$.
ואז השאלה היא מה זה הt, על מנת לאתרו צריך לבצע $M-1$ שאילתות. ■

משפט 3.4.6

יש אלגוריתם שהוא $(1+\varepsilon)-\mathrm{approx}$ ל MUA שרץ ב: $\mathrm{poly}\left(n,\log m,\frac{1}{\varepsilon}\right)$.

הוכחה:

$$v_{\max}=\max v_i\left(m\right)$$

$$\delta = \frac{\varepsilon v_{\max}}{n}$$
$$W = \frac{n^2}{\varepsilon}$$

$$v_{\max}\leq \mathrm{opt}\leq n\cdot v_{\max}$$

$K\left(i,w\right)$ הוא המספר המינימלי של מוצרים שאם מוקצים אופטימלית ל $\{1,\ldots,i\}$ אז יש את הערך של לפחות $w\cdot\delta$. ■

דוגמה 3.4.7 : 2 מוצרים, 2 שחקנים עם אותו: $v\left(2\right)=3.1$, $v\left(1\right)=1.9$, $v\left(0\right)=0$.
הכי כדאי זה להביא מוצר אחד לכל אחד מהשחקנים.
מה ההאלגוריתם שלנו עושה? הוא זורק את החלקים השבריים לזבל ומוצא את הפתרון האופטימיטאלי בעזרת VCG payments .

טענה 3.4.8

לא $truthful$!

הוכחה: $V'\left(0\right)=0$, $V'\left(1\right)=1$, $V'\left(2\right)=3$, האלגוריתם יתן את שני המוצרים לשחקן 1 . אבל התשלום יהיה הנזק שהשחקן עושה לחברה, בלי שחקן אחד, הפתרון האופטימלי הוא לתת לשחקן 2 את כל המוצרים, שחקן 2 יענה לצורך העניין שהערך שלו על שני המוצרים הוא 3.1 , ולכן שחקן 1 נאלץ לשלם 3.1 כדי לפצות על הנזק לחברה, ולכן profit שלו יהיה 0.

נניח שהשחקן מצהיר: $\tilde{v}_1\left(1\right)=\tilde{v}_1\left(2\right)=3$. הפתרון האופטימלי במקרה הזה הוא לחלק את המוצרים בין השחקנים, והתשלום יהיה 1.2 , ולכן הרווח של השחקן יהיה 0.7! כלומר, אם לקחנו איזשהו אלגוריתם קירוב והשתמשנו ב VCG אז קיבלנו משהו שהוא לא truthful. ■

26/12/

הגדרה 3.4.9 אלגוריתם הוא maximal in range אם הוא תמיד מוצא את הפתרון הטוב ביותר בתחום.

הערה 3.4.10 הRange זה תחום האלוקציות שהאלגוריתם יכול להחזיר.

דוגמה 3.4.11 : קצת מטופש, אבל אלגוריתם שמחזיר אלוקציה קבועה לדוגמה הוא Maximal in range.

אלגוריתם:

1. צור n^2 חבילות, כל אחת בגודל $\frac{M}{n^2}$.
2. תקצה את החבילות בצורה אופטימלית (דהיינו כל שחק מקבל כפולה של $\frac{M}{n^2}$ מוצרים).

למה 3.4.12

האלגוריתם הזה הוא $Truthful$.

הוכחה: אין שום דבר שהשחקן יכול לשקר כדי לצאת מהRange. אין משמעות אמיתית למוצרים המקוררים מאחרי שהם אוגדו לחבילות. ■

למה 3.4.13

זמן הראיצה הוא $\mathrm{poly}\left(\log n,n\right)$.

למה 3.4.14

האלגוריתם מספר קירוב־2.

הוכחה: יש לנו שני מקרים, שניהם קלים. נקבע פתרון אופטימלי $(o_1, \ldots, o_n)$ (שחקן אחד מקבל o_1 מוצרים, שחקן 2 o_2 וכו').

נניח כי $\sum o_i = M$. לפחות שחקן אחד מקבל M/n מוצרים. ללא הגבלת הכלליות נניח כי $o_1 \geq M/n$.

שחקן אחד הוא "big" אם $v_1(o_1) \geq \frac{\sum v_i(o_i)}{2}$. במקרה ששחקן 1 הוא "big", ניתן את כל המוצרים לשחקן הזה.

במקרה השני, בו השחקן הראשון הוא לא big, אז $a_1 = 0$ ולכל יתר השחקים, נגדי את a_i להיות o_i מעוגל למעלה לכפולה הקרובה ביותר של $\frac{M}{n^2}$. נבחין כי זה בתחום. וזה נותן קירוב.

לכל שחקן נתנו לכל היותר עוד $\frac{M}{n^2}$, ויש לנו n שחקנים אז קיבלנו $\frac{M}{n}$ מוצרים. ■

משפט 3.4.15

יהי A אלגוריתם *Maximal in range* שנותן קירוב $(2 - \varepsilon)$ עבור $\varepsilon > 0$. אז A מבצע M שאילתות ל*values*.

הוכחה: נניח שיש רק שני שחקנים.

טענה 3.4.16

הrange של A מכיל את כל האלוקציות.

הוכחה: נניח בשלילה כי זה לא נכון, נניח שיש אלוקציה $(k, m - k)$ שהיא לא בrange.

שני השחקנים הם single minded, השחק הראשון רוצה k מוצרים (או יותר) במחיר 1 ו 0 אם פחות. השחקן השני אותו דבר עם $M - k$.

הפתרון האופטימלי הוא לתת k מוצרים לשחקן הראשון ו $M - k$ לשחקן השני, הvalue במקרה זה הוא 2 .

כל אלוקציה אחרת תשאיר שחקן אחד לא מרוצה עם value≤ 1. ■

טענה 3.4.17

A מבצע לכל הפחות M שאילתות.

■

Conjectore: Every truthful algorithm for MUA that fives an approx ratio better than 2 makes at least M value queries.

ישנן שלוש דרישות:

1. Truthful
2. Polynomial time
3. Approx

לכל זוג כבר ראינו פתרון, הבעיה זה לדרוש את כל השלושה.

טענה 3.4.18

כל אלגוריתם שהוא *MUA* הוא *MIR*.

כמו שרשמנו אותו, ברור שהוא לא נכון...

טענה 3.4.19

כל *MIR* נותן לנו קירוב $2 - \varepsilon$ בזמן פולינומיאלי.

הטענה הראשונה לא נכונה גם אם נשנה את הטענות ל:

טענה 3.4.20

Every thruthful alg. for MUA is affine maximizer.

טענה 3.4.21

No affine maximizer can give $2 - \varepsilon$ approx in poly time.

נסתכל על הדיקטטור האדיש, כלומר, הוא קובע את החלוקה לגמרי, אבל לא אכפת לא באמת מה יצא מכיוון שהוא לא מקבל כלום.
לאף אחד אין סיבה לשקר מכיוון שהשחקן הראשון קובע הכל, והוא לא מקבל כלום, והאחרים לא משפיעים על התוצאה.
זה מוכיח שהטענה הראשונה כאן אינה נכונה.
עוד דוגמה היא Fixed price auction, כלומר כל מוצר הוא במחיר p.
לכן אנחנו צריכים לתקן את הטענה הראשונה:

טענה 3.4.22

Every thruthful alg. for MUA that gives an approx ratio of $2-\varepsilon$ is affine maximizer.

גם זה לא נכון, יש דוגמה יותר מסובכת שלא נציין, אבל זה כמעט נכון. אנו יודעים להוכיח את זה תחת כל מיני תנאים, למשל אם יש לנו MUA עם שני שחקנים שתמיד מחלק את כל המוצרים. תחת התנאי הזה, הטענה הזו כן נכונה.

3.5 Elections Problem

יש לנו n שחקנים ו m אלטרנטיבות, לכל שחקן i יש ערך $v_i\left(j\right)$ לכל אלטרנטיבה j. אנחנו לא מניחים שום דבר על היחסים בינהם.
רק נניח כי $v_i\left(j\right)\geq 0$ לכל i,j .

משפט 3.5.1

Roberts Theorem
Let A be a truthful alg. for the elections probllem. If the size of the range of A is at least 3, then A is affine maximzer.

אנו נוכיח את המשפט בשבוע הבא, למעשה נוכיח גרסה שונה שלו:

משפט 3.5.2

Let A be a two player mechanism for the elections problem.
Suppose that:

1. *The range of A is at least 3.*
2. *Scalable.*

מה זה Scalable? זה אומר ש:

$$A\left(v,u\right)=A\left(\alpha v,\alpha u\right)$$

לכל v,u,α.

משהו קצת שונה באופן זמני

איך יודעים אם אלגוריתם הוא Truthful?
אנו יודעים כי

$$\text{Truthful}\Rightarrow\text{Weakly monotone } v$$

ובד"כ גם הכיוון ההפוך. נבתונן ב $A\left(v,u\right)=x$ ונניח כי: $A\left(v',u\right)=y\neq x$. אז:

$$v'\left(y\right)-v'\left(x\right)\geq v\left(y\right)-v\left(x\right)$$

02/01/

יש לנו n שחקנים, וm מועמדים\אלטרנטיבות. כל $v_i\left(j\right)$ הוא הערך של שחקן i לאלטרנטיבה j.

השאלה שאנו שואלים את עצמנו: מה היא קבוצת ה Mechanisms שהיא truthful?

מה תהיה התשובה?

משפט 3.5.3

יהי f *truthful mechanism*. אז אם range של f הוא ≥ 3 אז אנו רוצים להוכיח כי f הוא *affine maximizer*.

הגדרה 3.5.4 affine maximizer

זה פונקציה מהצורה:

$$\max_j \sum w_i v_i (j) + c_j$$

עבור $w_i \geq 0$.

כאן $n = 2$, סקלאביליות: $f(v,u) = f(\alpha v, \alpha u)$.

בשיעור האחרון דיברנו על מונוטוניות חלשה. אם כלומר:

$$\begin{aligned} f(v,u) &= x \\ f(v',u) &= y \end{aligned}$$

אז:

$$v'(y) - v'(x) \geq v(y) - v(x)$$

משפט 3.5.5

Truthfulness⇒Weak monotone.

הוכחה: נסמן $p(v,u)$ בתור התשלום של v עבור $f(v,u)$.

$$v(x) - p(v,u) \geq v(y) - p(v',u)$$

אבל אם הוא אומר v' במקום v:

$$v'(y) - p(v',u) \geq v'(x) - p(v,u)$$

■

נסמן:

$$P(x,y) = \{\alpha \mid \exists v,u \text{ s.t. } f(v,u) = x,\ v(x) - v(y) = \alpha_1,\ u(x) - u(y) = \alpha_2\}$$

אנו מסתכלים על המישור של α, ועל הקו $y = -x$. כל מה שמעל הקו הזה נמצא ב $P(x,y)$ וכל מה שמתחתיו לא נמצא. כלומר, לכל מה שמעל הקו:

$$\begin{aligned} v(x) - v(y) &\geq -[u(x) - u(y)] \\ v(x) + u(x) &\geq v(y) + u(y) \end{aligned}$$

למה 3.5.6

Consistency Lemma: If $\alpha \in P(x,y)$ then $-\alpha \notin P(y,x)$.

הוכחה: נקח v,u כך ש $f(v,u)=x$ ו: $v(x)-v(y)=\alpha_1$ ו $u(x)-u(y)=\alpha_2$.
נקח v',u' כך ש $f(v',u')=y$ אבל עדיין $v(x)-v(y)=\alpha_1$ ו $u(x)-u(y)=\alpha_2$.

$$v''(a) = \begin{cases} c & a=x \\ c-\alpha_1 & a=y \\ -\infty & a=z \end{cases}$$

$$u''(a) = \begin{cases} c & a=x \\ c-\alpha_2 & a=y \\ -\infty & a=z \end{cases}$$

$$f(v'',u'') = ?$$

$$\begin{aligned} x &= f(v,u)=f(v'',u)=f(v'',u'') \\ y &= f(v',u')=f(v',u'')=f(v'',u'') \end{aligned}$$

■

למה 3.5.7

Closure Lemma: Let $\varepsilon=(\varepsilon_1,\varepsilon_2)>0$, if $\alpha\in P(x,y)$ tehn $\alpha+\varepsilon\in P(x,y)$.

הוכחה:

$$v'(a) = \begin{cases} v(a)+\varepsilon_1 & x=a \\ v(a) & \text{otherwise} \end{cases}$$

$$u'(a) = \begin{cases} u(a)+\varepsilon_2 & x=a \\ u(a) & \text{otherwise} \end{cases}$$

$$\begin{aligned} x &= f(v,u)=f(v',u)=f(v',u') \\ v'(x)-v'(y) &= \varepsilon_1+\alpha_1 \\ u'(x)-u'(y) &= \varepsilon_2+\alpha_2 \end{aligned}$$

לכן:

$$\alpha+\varepsilon\in P(x,y)$$

■

למה 3.5.8

אדטיביות, יהי $\alpha\in P(x,y)$ ו $\beta\in P(x,y)$ אזי: $\alpha+\beta\in P(x,y)$.

נניח שהוכחנו את הלמה הזאת, נוכיח אותה בהמשך.

למה 3.5.9

אם α הוא בפנים של $P(x,y)$ אז α הוא בפנים של $P(x',y')$.

הוכחה:

תחילה, יהי $\delta\to 0$ כך ש $\delta\in P(y,z)$, $\alpha+\delta\in P(x,z)$.

יהי $\delta_1\to 0$ כך ש: $\alpha+\delta_1\in P(z,y)$.

היה פה בלאגן שלא עקבתי... סורי... ∎

נוכיח את האדטיביטי למה: **הוכחה:** יהי $\alpha\in P(x,y)$ ו $\beta\in P(y,z)$.

$$V(a) = \begin{cases} c & a=x \\ c-\alpha_1 & a=y \\ c-\alpha_1-\beta_i & a=z \end{cases}$$

$$U(a) = \begin{cases} c & a=x \\ c-\alpha_2 & a=y \\ c-\alpha_2-\beta_2 & a=z \end{cases}$$

$$f(v',u') \neq y$$
$$f(v',u') \neq z \Rightarrow \beta\in P(y,z)$$

ולכן:

$$f(v',u') = x \Rightarrow \alpha+\beta\in P(x,z)$$

∎

09/01/

3.6 בעיות Scheduling

הולכים להיות לנו n שחקנים, ונקרא להם גם מכונות.

יהיה לנו m משימות, jobs. מה שאנחנו רוצים לעשות היא, שלכל שחקן תהיה לנו עלות $c_i(j)\geq 0$, שהיא העלות של השחקן i לבצע את המשימה j.

אנו מניחים שמתקיים:

$$c_i(S)=\sum_{j\in S} c_i(\{j\})$$

נגדיר את makespan להיות $\max_i c_i(s_i)$ כך ש: $(s_1,\ldots,s_n)$ ומתקיים $\bigcup s_i = M$ (כאשר M זה כל הג'ובים).

נגיד, אם שחקן ראשון מבצע את משימה a במחיר 2, ושחקן שני את משימה b במחיר 1. אז ה makespan שלנו הוא 2. ואילו אם שחקן אחד מבצע גם את a במחיר 2 וגם את b במחיר 0.5 אז הוא יהיה 2.5 .

המחיר שמכונה i תעבד את S הינו $c_i(S)$.

קיים אלגוריתם פולינומית אשר נותן קירוב 2 (לא truthful). לעומת זאת עם truthfulness המצב לא טוב כל־כך, אנו יודעים לבצע קירוב n. וניסן רונן הראו ב99 שאין קירוב יותר טוב מ2. כלומר, אין מנגנון שהוא truthful שהוא יותר טוב מ2.

כמו כן, אנו יודעים כרגע כי אין קירוב 2.41, ואפילו אנו לא יודעים כי אין קירוב 2.618. אבל לא יודעים כרגע מה יש בין זה לבין n. למרות שמאמינים שזה n.

נראה שקיים קירוב n.

כל הקורס אמרנו VCG, אז למה שלא נשתמש בזה גם כאן?

משפט 3.6.1

אלגוריתם VCG נותן קירוב n.

הוכחה: נבחין כי:

$$\text{OPT} \geq \frac{\sum_i \min c_i(j)}{n}$$

אם אנו נותנים כל עבודה למי שמבצע אותה הכי מהיר, ברור כי אין אף חלוקה שתעשה את זה יותר טוב. כלומר אנו מחלקים כל עבודה לn המכונות ונותנים את הזמן המינימלי לכל עבודה.

ולכן, נבחר אלגוריתם שנותן את המשימה למי שמבצע אותה הכי מהר ונקבל כי:

$$\text{ALG} \leq \sum_j \min c_i(j)$$

כלומר קיבלנו קירוב n. ■

דוגמה 3.6.2 :

$$\begin{aligned} c_i(j) &= 1 \\ c_1(j) &= 1-\varepsilon \end{aligned}$$

כאשר $M = n$.
נבחין כי:

$$\text{OPT} = 1$$

ואילו VCG יתן לנו:

$$n(1-\varepsilon)$$

אנו כעת רוצים להוכיח את המשפט הבא:

משפט 3.6.3

אין אלגוריתם *truthful* שנותן קירוב שהוא יותר טוב מ2.

לשם כך תחילה נוכיח את הלמה הבאה:

למה 3.6.4

נקבע הערכה של השקנים האחרים. נניח כי עם *valuation* c, i מקבל את S.
יהי C כך ש $c'_i(j) < c_i(j)$ לכל $j \in S$ וגם: $c_i(j) < c'_i(j)$ לכל $j \notin S$.
אז: i מקבל את S עם *valuation* c'_i.

זה נובע ממש מweak monotonicity. אם שיפרנו את המצב, אם הורדנו את הערך, של האלטרנטיבה שקיבלנו, אנו עדיין מקבלים את המוצר.

נתבונן בטבלה:

ג'ובים\מכונות	a	b	c
I	1	1	1
II	1	1	1

נבתונן במקרה הראשון, בו האלגוריתם נתן את כל המשימות למכונה הראשונה.

כעת, נסתכל במקרה של:

ג'ובים\מכונות	a	b	c
I	$1-\varepsilon$	$1-\varepsilon$	0
II	1	1	1

נבחין כי זה עונה בדיוק לתנאים של הלמה, לכן, האלוקציה תהיה שוב לתת את כל המשימות למכונה הראשונה. והאלגוריתם נותן לנו $2-\varepsilon$. אבל הmakespan האופטימלי שלנו הוא 1.

וזה קירוב 2.

נתבונן במקרה שני, נניח שהמשימה האחרונה קיבלה מכונה II ואת הקודמות המכונה הראשונה. כעת נסתכל על המופע:

ג'ובים\מכונות	a	b	c
I	1	1	1
II	$1+\varepsilon$	$1+\varepsilon$	0

האופטימלי כאן זה $1+\varepsilon$, אבל האלגוריתם יתן לנו 2.

כלומר, בכל מקרה לא קיבלנו קירוב טוב יותר מ2.

אנו כעת רוצים להראות שגם לא ייתכן קירוב 2.41. כיצד?

נסתכל על המקרים הבאים, המכילים 3 Dummy Jobs:

ג'ובים\מכונות	משימה טיפשית 1	משימה טיפשית 2	משימה טיפשית 3		
I	d_1	∞	∞	$\sqrt{2}$	$\sqrt{2}$
II	∞	d_2	∞	$\sqrt{2}$	$\sqrt{2}$
III	∞	∞	d_3	$\sqrt{2}$	$\sqrt{2}$

כלומר, מכונה 1 מקבלת את הראשון, מכונה 2 מקבל את השני ומכונה 3 מקבל את השלישי. כל העסק בעצם מתרחש בשתיים האחרונות.

נסמן ב $T\subseteq S$ את המשימות ה"אמיתיות" (כלומר לא שלושת ה Dummy Jobs).

נתבונן כרגע במקרה:

ג'ובים\מכונות	משימה טיפשית 1	משימה טיפשית 2	משימה טיפשית 3		
I	0	∞	∞	$\sqrt{2}$	$\sqrt{2}$
II	∞	0	∞	$\sqrt{2}$	$\sqrt{2}$
III	∞	∞	0	$\sqrt{2}$	$\sqrt{2}$

נתבונן באלגוריתם אשר מחלק את המשימות כך:

ג'ובים\מכונות	משימה טיפשית 1	משימה טיפשית 2	משימה טיפשית 3		
I	$\boxed{0}$	∞	∞	$\boxed{\sqrt{2}}$	$\boxed{\sqrt{2}}$
II	∞	$\boxed{0}$	∞	$\sqrt{2}$	$\sqrt{2}$
III	∞	∞	$\boxed{0}$	$\sqrt{2}$	$\sqrt{2}$

ונעשה את הדבר הבא:

ג'ובים\מכונות	משימה טיפשית 1	משימה טיפשית 2	משימה טיפשית 3		
I	$\boxed{\sqrt{2}}$	∞	∞	$\boxed{\sqrt{2}-\varepsilon}$	$\boxed{\sqrt{2}-\varepsilon}$
II	∞	$\boxed{0}$	∞	$\sqrt{2}$	$\sqrt{2}$
III	∞	∞	$\boxed{0}$	$\sqrt{2}$	$\sqrt{2}$

לפי הלמה האלוקציה נשארה אותו דבר (היות ואין ברירה עם Dummy Job #1).

כעת נבחין כי makespan יהיה:

$$
\begin{aligned}
\text{ALG} &= 3\sqrt{2}-2\varepsilon \\
\text{OPT} &= \sqrt{2}
\end{aligned}
$$

כלומר, קירוב 3.

כעת נסתכל במקרה השני:

ג'ובים\מכונות	משימה טיפשית 1	משימה טיפשית 2	משימה טיפשית 3		
I	$\boxed{0}$	∞	∞	$\boxed{\sqrt{2}}$	$\sqrt{2}$
II	∞	$\boxed{0}$	∞	$\sqrt{2}$	$\boxed{\sqrt{2}}$
III	∞	∞	$\boxed{0}$	$\sqrt{2}$	$\sqrt{2}$

כלומר, הפעם שני אנשים שונים מקבלים את הג'ובים האמיתיים.

כעת נעבור למופע הזה:

ג'ובים\מכונות	משימה טיפשית 1	משימה טיפשית 2	משימה טיפשית 3		
I	$\boxed{0}$	∞	∞	$\boxed{1}$	1
II	∞	$\boxed{0}$	∞	$\sqrt{2}$	$\sqrt{2}$
III	∞	∞	$\boxed{0}$	$\sqrt{2}$	$\sqrt{2}$

כרגע יכול לקרות שני דברים, או שהמכונה הראשונה תקבל את הג'וב האחרון, או שעדיין השניה. (את הג'וב האמיתי הראשון היא בטוח מקבלת כי היא קיבלה אותו ב$\sqrt{2}$ ולכן בהכרח תקבל אותו ב 1).

לכן ייתכנו שני מצבים:

ג'ובים\מכונות	משימה טיפשית 1	משימה טיפשית 2	משימה טיפשית 3		
I	$\boxed{0}$	∞	∞	$\boxed{1}$	$\boxed{1}$
II	∞	$\boxed{0}$	∞	$\sqrt{2}$	$\sqrt{2}$
III	∞	∞	$\boxed{0}$	$\sqrt{2}$	$\sqrt{2}$

ו:

ג'ובים\מכונות	משימה טיפשית 1	משימה טיפשית 2	משימה טיפשית 3		
I	$\boxed{0}$	∞	∞	$\boxed{1}$	1
II	∞	$\boxed{0}$	∞	$\sqrt{2}$	$\boxed{\sqrt{2}}$
III	∞	∞	$\boxed{0}$	$\sqrt{2}$	$\sqrt{2}$

במקרה הראשון נקבל כי נעבור ל:

ג'ובים\מכונות	משימה טיפשית 1	משימה טיפשית 2	משימה טיפשית 3		
I	$\boxed{0}$	∞	∞	$\boxed{1-\varepsilon}$	$\boxed{1-\varepsilon}$
II	∞	$\boxed{0}$	∞	$\sqrt{2}$	$\sqrt{2}$
III	∞	∞	$\boxed{0}$	$\sqrt{2}$	$\sqrt{2}$

ונקבל כי:

$$\begin{aligned} \text{OPT} &= \sqrt{2} \\ \text{ALG} &= \sqrt{2}+2-2\varepsilon \end{aligned}$$

ובמקרה השני נעבור ל:

ג'ובים\מכונות	משימה טיפשית 1	משימה טיפשית 2	משימה טיפשית 3		
I	$\boxed{0}$	∞	∞	$\boxed{0}$	$1+\varepsilon$
II	∞	$\boxed{0}$	∞	$\sqrt{2}$	$\sqrt{2}$
III	∞	∞	$\boxed{0}$	$\sqrt{2}$	$\boxed{\sqrt{2}}$

המעבר של הג'וב האחרון הוא סתם בלי הגבלת הכלליות, יכול להיות שאנו עוברים ויכול להיות שלא. זה לא משנה מסימטריות. כעת נעבור ל:

ג'ובים\מכונות	משימה טיפשית 1	משימה טיפשית 2	משימה טיפשית 3		
I	$\boxed{0}$	∞	∞	$\boxed{0}$	$1+\varepsilon$
II	∞	$\boxed{0}$	∞	$\sqrt{2}$	$\sqrt{2}$
III	∞	∞	$\boxed{1}$	$\sqrt{2}+\varepsilon$	$\boxed{\sqrt{2}-\varepsilon}$

ונקבל כי:

$$\begin{aligned} \text{OPT} &= 1+\varepsilon \\ \text{ALG} &= 1+\sqrt{2}-\varepsilon \end{aligned}$$

16/01/

3.7 Walrasian equilibrium/Competitive equilibrium

בהינתן חלוקת מוצרים $(S_1,\ldots,S_n)$, ומחירים $p_1,\ldots,p_m$ של כל אחד מהמוצרים, ושמתקיים:

1. לכל i:

$$v_i\left(s_i\right)-\sum_{j\in S_i}p_j\geq v_i\left(T\right)-\sum_{j\in T}p_j$$

לכל $T\subset M$ (כלומר כל שחקן לוקח את הבילה שהוא הכי רוצה).

2. $\bigcup_i S_i=M$ (כלומר השוק מתנקה).

מצב זה נקרא Walrasian equilibrium/Competitive equilibrium.

משפט 3.7.1

אם $S=\left(S_1,\ldots,S_n\right)$ ו $p_1,\ldots,p_m$ זה *Walrasian equilibrium* אז S ממקסמת את ה*Welfare*.

הוכחה: נקבע $\left(T_1,\ldots,T_n\right)$ אלוקציה כלשהי, אנו רוצים להראות כי הwelfare של S גדול יותר. נבחין כי:

$$v_i\left(S_i\right)-\sum_{j\in S_i}p_j\geq v_i\left(T_i\right)-\sum_{j\in T_i}p_j$$

לכל i. נסכום את כל השחקנים:

$$\sum_i\left[v_i\left(S_i\right)-\sum_{j\in S_i}p_j\right]\geq\sum_i\left[v_i\left(T_i\right)-\sum_{j\in T_i}p_j\right]$$

כעת נבחין כי אפשר לצמצם את הסכימה על p_j מכיוון שאנחנו מניחים שאנו מנקים את השוק, לכן אנו סוכמים את כל הסכומים של כל המוצרים, כל אחד פעם אחת. ולכן נקבל:

$$\sum v_i\left(S_i\right)\geq\sum v_i\left(T_i\right)$$

כנדרש. ■

3.7.1 Subadditive valuations

ולואציה נקראת סאב־אדטיבית אם:

$$v\left(S\right)+v\left(T\right)\geq v\left(S\cup T\right)$$

ולואציה נקראת סאב־מודולרית (submodular) אם:

$$\forall j,S\subset T\quad v\left(j\mid S\right)\geq v\left(j\mid T\right)$$

כאשר הסימון אומר:

$$v\left(j\mid S\right)=v\left(S\cup\left\{j\right\}\right)-v\left(S\right)$$

3.7.2 Gross subtitutes (GS)

v היא GS אם לכל וקטור מחיר $\vec{p}$ ווקטור מחיר $\vec{q}\geq\vec{p}$ (המחירים גדולים שווים לכל קואורדינטה) אם A_i הוא הביקוש עם מחיר $\vec{p}$ אז קיימת חבילה D_i שהיא בביקוש עבור מחירים $\vec{q}$ אשר כוללת את כל המוצרים עם אותם מחירים.

דוגמה 3.7.2 : נסתכל על המקרה של סינגל מינדד עבור $\{a,b\}$ עם ערך 10.
נסתכל על הוקטור המחירים הבא:

$$\begin{aligned} p_a &= 3 \\ p_b &= 3 \end{aligned}$$

במקרה זה הוא ירצה הכל. אבל אם נסתכל על:

$$\begin{aligned} q_a &= 3 \\ q_b &= 8 \end{aligned}$$

הוא לא רוצה כלום, לכן דוגמה זו היא לא GS.

איזה מקרים יש לנו שהם כן GS?
לשם כך נגדיר ולואציה אדטיבית, כלומר:

$$v(s) = \sum_{j \in S} v(\{j\})$$

ברור כי מקרה כזה הוא GS.
בנוסף, multi unit with decreasing marginals גם הוא GS. כתרגיל לחשיבה בבית (רמז: לחשוב על הגרידי אלגוריתם).

למה 3.7.3

אם v היא GS אז v היא *submodular*.

הוכחה: אנו רוצים להראות כי $v(j \mid s) \geq v(j \mid T)$ לכל $S \subseteq T$.
בהינתן $\vec{p}$ כך ש $p_t = 0$ לכל $t \in T$ ו $p_{j'} = \infty$ לכל $j' \notin T + \{j\}$ (ואותו j מלמעלה).
כעת, נגדיר: $p_j = v(j \mid T)$. הביקוש הוא $T + \{j\}$.
כעת, נתבונן ב $\vec{q}$ שהיא זהה פרט ל $q_{j'} = \infty$ עבור $j' \in T - S$.
כעת, הdemand יהיה בדיוק $S + \{j\}$ (מכיוון שכל היתר עולים אינסוף). ומתקיים:

$$v(j \mid S) \geq p_j \geq v(j \mid T)$$

■

משפט 3.7.4

יש *Walrasian equilibrium* כאשר אנו דנים ב GS.

הוכחה: נאתחל $p_i = 0$ ו $S_i = \varnothing$.
נבצע את האיטרציות הבאות:

1. נגדיר D_i להיות הביקוש של i תחת המחירים p_j לכל $j \in S_i$ ו: $p_j + \varepsilon$ אחרת.
2. אם לכל i, $S_i = D_i$ אז תפסיק, אחרת עבור i כלשהו עם $S_i \neq D'_i$:
   (א) תגדיל את הערך של p_j ב ε לכל $j \in D_i - S_i$.
   (ב) $S_i \leftarrow D_i$ קח מוצרים S_i-ים של שחקנים אחרים.

■

משפט 3.7.5

אם הולואציות הן GS, אז המחירים הסופיים $p_1, \ldots, p_m$ והאלוקציה $(S_1, \ldots, S_n)$ הן *Walrasian equilibrium* (עד כדי ε).

משפט 3.7.6

אם כל הולואציות הן *unit demand* אז המכרז מסתיים במחירי *VCG*.

הוכחה: בכל שלב $S_i \subseteq D_i$.
■

23/01/

3.8 הפרסומות של גוגל...

כאשר אנחנו מחפשים בגוגל דברים, בצד אנו רואים פרסומות אשר מתאימות את עצמן למה שאנחנו מחפשים. למעשה עם כל המידע שאנחנו מחפשים, היא נגשת למפרסמים ושואלת אותם מי רוצה לפרסם ובאיזה מחיר.

נניח כי למפרסמים יש ערך $v_1 \geq v_2 \geq \ldots \geq v_n$ ויש לנו בגוגל 3 מיקומים לפרסומות, נכליל את זה ונגיד שיש m מיקומים: $r_1 \geq r_2 \geq \ldots \geq r_m$.

אז הערך של שחקן i ממיקום j הינו: $v_i \cdot r_j$.

כל שחקן, הוא unit demand, כלומר:

$$v(S) = \max_{j \in S} v(\{j\})$$

המטרה של גוגל היא: למקסם את הsocial welfare.

היינו מצפים שגוגל תריץ VCG, אבל זה לא מה שהיא עושה. כי המחירים של VCG קצת מוזרים, משלמים נזק לחברה, זה משהו שלא כל־כך ברור ללקוח, זה משהו שקשה להבין.

המודל של גוגל הוא "Generalized second price auction".

מה זה המודל הזה?

כרגיל כל bidder i נותן את הbid b_i. הallocation בהינתן ה bids.

התשלום הוא: ה־ bidder i שמקבל את המקום ה j משלם $r_j \cdot b_{j'}$ כאשר j' הוא הbid עבור $(j'+1)$.

דוגמה 3.8.1 : נניח שיש 3 bidders, ו3 מיקומים.
נניח כי:

$$\begin{aligned} v_1 &= 7 \quad v_2 = 6 \quad v_3 = 1 \\ r_1 &= 10 \quad r_2 = 4 \quad r_3 = 0 \end{aligned}$$

נניח שכולם אומרים את האמת, אז bidder 1 מקבל את מיקום 1 עם רווח: $70 - 60 = 10$.
אם $b_1 = 5$ אז הרווח הוא $20 - 4 = 16$.
כלומר, עדיף לו להמר פחות במצב הזה.
נבחין כי בדוגמה זו ישנם שני שיוויי משקל, האחד הוא כאשר $b_1 = 5,\ v_2 = 4,\ b_3 = 2$
השני הוא כאשר: $b_1 = 3,\ b_2 = 5,\ b_3 = 1$. נבחין כי השיווי משקל הזה אפילו לא ממקסם welfare.

לפעמים הrevenue של VCG ה וא גבוה יותר ולפעמים לא.

מה אנחנו יכולים להגיד על המכרז המוזר הזה?

אנחנו יכולים להגיד לפחות דבר אחד:

משפט 3.8.2

יש תמיד שיווי משקל שממקסם את ה*welfare*.

הוכחה: נניח כי $m = n$ רק לשם נוחות (למעשה $n+1$, והמקום האחרון שווה 0, לצורך התשלום).

נניח כי המחירים של ה Walrasian equlibrium הינם: $p_1, \ldots, p_m$.

כמו כן, נניח כי $p_m = 0$ (זה לא תמיד נכון, קיימים שיוויי משקל כך שזה לא המצב, אבל אפשר לנרמל את זה בדרך זו או אחרת, לא מדוייק אבל...).

נסמן: $p_j^* = \frac{p_j}{r_j}$. ונבחר: $b_j = p_{j-1}^*$ עבור $j \geq 2$ ו $b_1 = v_1$.

אנחנו רוצים להראות מיקסום welfare: $b_{j-1} \geq b_j$.

איך נראה זאת? אנו יודעים שהמחירים $p_1, \ldots, p_m$ מנקים את השוק, לכן מתקיים:

$$r_j \cdot v_j - p_j \geq r_{j-1} v_j - p_{j-1}$$

נשנה קצת את הצורה ונקבל:

$$r_j \left(v_j - \frac{p_j}{r_j}\right) \geq r_{j-1}\left(v_j - \frac{p_{j-1}}{r_{j-1}}\right) \geq r_j \left(v_j - \frac{p_{j-1}}{r_{j-1}}\right)$$

כלומר נצמצם את ה r_j ואת ה v_j ונקבל מהחלק הראשון והאחרון:

$$b_{j-1} = \frac{p_{j-1}}{r_{j-1}} \geq \frac{p_j}{r_j} = b_j$$

זה גם שיווי משקל נאש, למה?

מכיוון שהשחקן שמקבל את מקום j משלם את p_j ובגל שזה מחירים של Walrasian equalibrium הם מנקים את השוק, כלומר כל שחקן ממקסם את רווח שלו וכל המקומות מתמלאים. בהנחה שכל הvaluations של האחרים, המהלך הטוב יבותר של שחן i יהיה לתת את ה bid שנתן לו את מקום j במחיר p_j. ■

Price of anarchy:

$$\frac{\text{OPT welfare}}{\text{Welfare of worst nash eq}}$$