ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΠΡΟΗΓΜΕΝΑ ΘΕΜΑΤΑ ΑΛΓΟΡΙΘΜΩΝ

(2020-2021)

2η Σειρά Ασκήσεων

Ονοματεπώνυμο:

- Χρήστος Τσούφης

Αριθμός Μητρώου:

- 03117176

Στοιχεία Επικοινωνίας:

- el17176@mail.ntua.gr

$1^{η}$ Άσκηση

Έστω μια δημοσκόπηση για τη στάση των πολιτών ως προς μια σημαντική πολιτικο-οικονομική μεταβολή. Οι πολίτες απαντούν στη δημοσκόπηση με "ναι" ή "όχι" (υπέρ ή εναντίον της μεταβολής). Αν το (πραγματικό) ποσοστό των πολιτών που τάσσονται υπέρ της μεταβολής είναι p, ζητείται ο υπολογισμός μιας εκτίμησης $\hat{p}$ του p ώστε $Pr[|\hat{p} - p| \leq \varepsilon p] > 1 - \delta$, για δεδομένα ε, δ $\in (0, 1)$. Για τη δημοσκόπηση, ρωτήθηκαν N πολίτες, που επιλέγονται ισοπίθανα και ανεξάρτητα από το σύνολο των πολιτών. Η εκτίμησή $\hat{p}$ θα είναι το ποσοστό των N πολιτών που τάσσονται υπέρ της μεταβολής. Χρησιμοποιώντας Chernoff-Hoeffding bounds, ζητείται ο υπολογισμός (ως συνάρτηση των ε, δ, και p) του ελάχιστου μεγέθους N του δείγματος που απαιτείται. Να βρεθεί η τιμή του N για ε = 0.02 και δ = 0.05, αν είναι γνωστό ότι p $\in [0.1, 0.7]$ (και να παρατηρηθεί ότι αυτή η τιμή είναι ανεξάρτητη του πληθυσμού της χώρας!). Ακόμη, ζητείται να υπολογιστεί το ελάχιστο μέγεθος N′ δείγματος (ως συνάρτηση των ε και δ) ώστε η εκτίμηση $\hat{p}'$ να ικανοποιεί $Pr[|\hat{p}' - p| \leq \varepsilon] > 1 - \delta$. Ποια είναι η τιμή του N′ για ε = 0.02 και δ = 0.05;

<u>$1^{ος}$ Τρόπος Επίλυσης</u>:

i) Για την πρώτη περίπτωση, έστω τυχαίες μεταβλητές X_i, n το πλήθος, που θα είναι ίσες με 1 εάν και μόνο εάν ο i-οστός πολίτης του δείγματος απαντήσει "ναι" στην δημοσκόπηση, αλλιώς θα είναι 0. Ακόμη, οι X_i ακολουθούν Bernoulli Distribution με παράμετρο p, οπότε η εκτίμηση για το p θα είναι:

$$\hat{p} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

Χρησιμοποιώντας τα Chernoff Bounds:

$$\mathbf{Pr}[\textstyle\sum_{i=1}^{n} X_i \geq (1+\varepsilon)np] = \mathbf{Pr}[\hat{p} - p \geq \varepsilon p] \leq e^{-\varepsilon^2 np/3} \qquad (1)$$

$$\mathbf{Pr}[\textstyle\sum_{i=1}^{n} X_i \leq (1-\varepsilon)np] = \mathbf{Pr}[\hat{p} - p \leq -\varepsilon p] \leq e^{-\varepsilon^2 np/2} \qquad (2)$$

Από τις (1), (2) προκύπτει ότι:

$$\mathbf{Pr}[|\hat{p} - p| \geq \varepsilon p] \leq \mathbf{Pr}[\hat{p} - p \geq \varepsilon p] + \mathbf{Pr}[\hat{p} - p \leq -\varepsilon p] \leq e^{-\varepsilon^2 np/3} + e^{-\varepsilon^2 np/2} \leq 2e^{-\varepsilon^2 np/3}$$

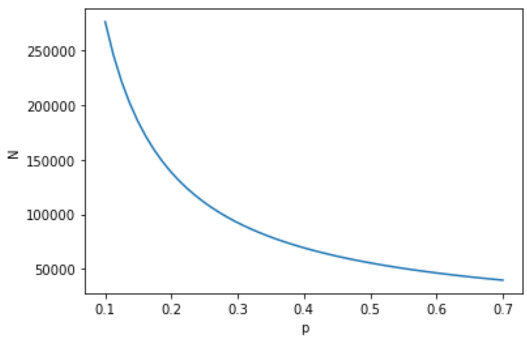
Οπότε, η πιθανότητα λάθους θα είναι: $\mathbf{Pr}[|\hat{p} - p| \leq \varepsilon p] \leq 1 - \mathbf{Pr}[|\hat{p} - p| \geq \varepsilon p] \leq 1 - 2e^{-\varepsilon^2 np/3}$

Όμως, πρέπει $\mathbf{Pr}[|\hat{p} - p| \leq \varepsilon p] \leq 1 - \delta$. Άρα, $1 - 2e^{-\varepsilon^2 np/3} \leq 1 - \delta \Rightarrow n \geq \frac{3\ln(2/\delta)}{\varepsilon^2 p}$

Είναι γνωστό ότι $p \geq 0.1$, οπότε $n \geq 276.666$.

Συγκεκριμένα:

```python
import matplotlib.pyplot as plt
import numpy as np
x = np.linspace(0.1, 0.7)
y = 3*(np.log(40)/(0.0004*x))
plt.plot(x,y)
plt.xlabel("p")
plt.ylabel("N")
plt.show()
```



ii) Για την δεύτερη περίπτωση, χρησιμοποιώντας τα Hoeffding Bounds, προκύπτει:

$$\mathbf{Pr}[|\hat{p} - p| \geq \varepsilon] \leq 2e^{-2n\varepsilon^2}$$

Οπότε, η πιθανότητα λάθους θα είναι: $\mathbf{Pr}[|\hat{p} - p| \leq \varepsilon] \leq 1 - \mathbf{Pr}[|\hat{p} - p| \geq \varepsilon] \leq 1 - 2e^{-2n\varepsilon^2}$

Όμως, πρέπει $\mathbf{Pr}[|\hat{p} - p| \leq \varepsilon p] \leq 1 - \delta$. Άρα, $1 - 2e^{-2n\varepsilon^2} \leq 1 - \delta \Rightarrow n \geq \frac{\ln(2/\delta)}{2\varepsilon^2}$

Οπότε $n \geq 4.612$.

<u>2^{ος} Τρόπος Επίλυσης</u>:

i) Ζητείται ο υπολογισμός της εκτίμησης $\hat{p}$ του p ώστε να ισχύει $\mathbf{Pr}[|\hat{p} - p| \leq \varepsilon p] > 1 - \delta$, για δεδομένα $\varepsilon, \delta \in (0, 1)$.

Έστω N, το μέγεθος του δείγματος. Τότε, θα ισχύει $\mathbf{Pr}[|N\hat{p} - Np| \leq \varepsilon Np] > 1 - \delta$.

Οπότε, αρκεί να δειχθεί ότι: $\mathbf{Pr}[|N\hat{p} - Np| \leq \varepsilon Np] > 1 - \delta \Rightarrow \mathbf{Pr}[|N\hat{p} - Np| \geq \varepsilon Np] \leq \delta$.

Αν θεωρηθεί ότι η απάντηση κάθε ατόμου του δείγματος είναι μια ανεξάρτητη δοκιμή Poisson (ή Bernoulli) και θέτοντας για $X = N\hat{p}$ τον αριθμό των θετικών απαντήσεων στις N δοκιμές, τότε, με εφαρμογή Chernoff Bound της μορφής: $\mathbf{Pr}[|X - \mu| \geq \varepsilon\mu] \leq 2e^{-\mu\varepsilon^2/3}$, όπου $\mu = Np$ η μέση τιμή της διωνυμικής κατανομής $X \sim Bin(N, p)$ θα προκύψει ότι:

$$\mathbf{Pr}[|N\hat{p} - Np| \geq \varepsilon Np] \leq 2e^{-Np\varepsilon^2/3} \leq \delta \Rightarrow -\frac{Np\varepsilon^2}{3} \leq \ln(\frac{\delta}{2}) \Rightarrow Np\varepsilon^2 \geq 3\ln(\frac{2}{\delta}) \Rightarrow N \geq \frac{3\ln(\frac{2}{\delta})}{p\varepsilon^2}$$

Συνεπώς, για $\varepsilon = 0.02$ και $\delta = 0.05$, προκύπτει $N = \frac{3\ln(\frac{2}{\delta})}{p\varepsilon^2} = \frac{27666.6}{p}$. Οπότε, για το δοσμένο διάστημα του p, προκύπτει: $0.1 \leq p \leq 0.7 \Rightarrow 39524 \leq \frac{27666.6}{p} \leq 276666 \Rightarrow 39524 \leq N \leq 276666$

ii) Ακόμη, ζητείται η εύρεση μιας εκτίμησης $\hat{p}$ ώστε να ισχύει:

$$\mathbf{Pr}[|\hat{p}' - p| \leq \varepsilon] > 1 - \delta \Rightarrow \mathbf{Pr}[|\hat{p}' - p| > \varepsilon] \leq \delta \Rightarrow \mathbf{Pr}[|N'\hat{p}' - N'p| \geq \frac{\varepsilon}{p}N'p] \leq \delta$$

Οπότε, με εφαρμογή του Chernoff Bound, προκύπτει:

$$\mathbf{Pr}[|N'\hat{p}' - N'p| \geq \frac{\varepsilon}{p}N'p] \leq 2e^{-\frac{N'\varepsilon^2}{3p}} \leq 2e^{-\frac{N'\varepsilon^2}{3}} \leq \delta \Rightarrow -\frac{N'\varepsilon^2}{3} \leq \ln(\frac{\delta}{2}) \Rightarrow N \geq \frac{3\ln(\frac{2}{\delta})}{\varepsilon^2}$$

Συνεπώς, για $\varepsilon = 0.02$ και $\delta = 0.05$, προκύπτει $N = \frac{3\ln(\frac{2}{\delta})}{\varepsilon^2} = 27666$ πολίτες.

$2^η$ Άσκηση (Sparsification)

(a) *Έστω α, x ∈ [0, 1]^n, με $\sum_i x_i$ = 1 (το x είναι ένα διάνυσμα πιθανοτήτων στο [n] ≡ {1, ..., n}). Έστω ακόμη k(ε) = $\lceil ln(2)/(2\varepsilon^2)\rceil$ + 1. Να δειχθεί ότι για κάθε ε > 0, υπάρχει ένα k(ε)-ομοιόμορφο διάνυσμα πιθανοτήτων y στο [n] τέτοιο ώστε |α·x – α·y| ≤ ε. Ένα διάνυσμα πιθανοτήτων y είναι k-ομοιόμορφο (k-uniform) αν κάθε y_i είναι ακέραιο πολλαπλάσιο του 1/k.*

(b) *Έστω A πίνακας m×n με όλα τα στοιχεία του στο [0, 1] και έστω x ένα διάνυσμα πιθανοτήτων στο [n]. Έστω ακόμη k(m, ε) = $\lceil ln(2m)/(2\varepsilon^2)\rceil$ + 1. Να δειχθεί ότι για κάθε ε > 0, υπάρχει ένα k(m, ε)-ομοιόμορφο διάνυσμα πιθανοτήτων y στο [n] τέτοιο ώστε ||A·x – A·y||$_\infty$ ≤ ε.*

*Hint: Να χρησιμοποιηθεί το ακόλουθο Hoeffding bound: Έστω X_1, ..., X_n ανεξάρτητες τυχαίες μεταβλητές στο [0, 1], και έστω X = $\frac{\sum_{i=1}^{n} X_i}{n}$. Για κάθε ε > 0, **Pr**[|X – Exp[X]| > ε] ≤ $2e^{-2n\varepsilon^2}$.*

(a) Έστω τα διανύσματα α, x ∈ [0, 1]^n, με $\sum_{i=1}^{n} x_i = 1$ και k(ε) = $\lceil \ln(2)/(2\varepsilon^2)\rceil$ + 1. Θα δειχθεί ότι για κάθε ε > 0, υπάρχει ένα k(ε)-ομοιόμορφο διάνυσμα πιθανοτήτων y στο [n] τέτοιο ώστε |α·x – α·y| ≤ ε. Η παραγωγή των k-ομοιόμορφων διανυσμάτων στο [n] είναι η εξής:

Αρχικά, επιλέγεται ένας αριθμός n_0∈[n] και το x χρησιμοποιείται ως κατανομή πιθανότητας, οπότε:

$$\mathbf{Pr}[n_0 = i] = x_i$$

Μετά, κατασκευάζεται το διάνυσμα y που είναι ίσο με μηδέν εκτός από την θέση n_0:

$$y_i = \delta(i - n_0)$$

Έπειτα, παράγονται k(ε) διανύσματα ^1y, ^2y, …, $^{k(\varepsilon)}y$ ώστε να δημιουργηθεί το διάνυσμα:

$$y = \frac{1}{k(\varepsilon)} \sum_{i=1}^{k(\varepsilon)} {}^{i}y$$

Επομένως, το y είναι k(ε)-ομοιόμορφο, αφού κάθε συντεταγμένη είναι ακέραιο πολλαπλάσιο του 1/k(ε).

Άρα, η αναμενόμενη τιμή για κάθε συντεταγμένη του y είναι:

$$\mathbf{E}[y_i] = \frac{1}{k(\varepsilon)} \sum_{i=1}^{k(\varepsilon)} \mathbf{E}[{}^{i}y] \underset{\mathbf{E}[{}^{i}y]=x_i}{\Longleftrightarrow} \mathbf{E}[y_i] = \frac{1}{k(\varepsilon)} \sum_{i=1}^{k(\varepsilon)} x_i = x_i \qquad (1)$$

Οπότε, για το εσωτερικό γινόμενο α·y θα ισχύει:

$$\mathbf{E}[\alpha\cdot y] = \mathbf{E}[\textstyle\sum_{i\in[n]} a_i y_i] = \sum_{i\in[n]} \mathbf{E}[a_i y_i] = \sum_{i\in[n]} a_i \mathbf{E}[y_i] \underset{(1)}{\Leftrightarrow} \mathbf{E}[\alpha\cdot y] = \sum_{i\in[n]} a_i x_i = \alpha\cdot x$$

Ομοίως, προκύπτει ότι: $\mathbf{E}[\alpha\cdot {}^{i}y] = \alpha\cdot x$.

Ύστερα, θέτοντας $Y_i = \alpha\cdot {}^{i}y$ και Y = α·y, θα ισχύει λόγω της γραμμικότητας, ότι:

$$Y = \frac{1}{k(\varepsilon)} \sum_{i=1}^{k(\varepsilon)} Y_i$$

Εφαρμόζοντας τώρα την Hoeffding Inequality, για τις ανεξάρτητες μεταβλητές Y_i, προκύπτει:

$$\mathbf{Pr}[|Y - \mathbf{E}[Y]| > \varepsilon] \le 2e^{-2k(\varepsilon)\varepsilon^2} \le 2e^{-\ln(2)} = 1, \text{ αφού } k(\varepsilon) > \ln(2)/(2\varepsilon^2)$$

Επιπλέον,

$$\mathbf{Pr}[|Y - \mathbf{E}[Y]| > \varepsilon] = \mathbf{Pr}[|\alpha \cdot y - \alpha \cdot x| > \varepsilon]$$

Συνεπώς, η πιθανότητα το y να ικανοποιεί την ιδιότητα, θα είναι:

$$\mathbf{Pr}[|\alpha \cdot y - \alpha \cdot x| \leq \varepsilon] = 1 - \mathbf{Pr}[|\alpha \cdot y - \alpha \cdot x| > \varepsilon] > 0$$

Άρα, από την πιθανοτική μέθοδο, υπάρχει κάποιο y τέτοιο ώστε $|\alpha \cdot y - \alpha \cdot x| \leq \varepsilon$.

(b) Έστω A πίνακας m×n με όλα τα στοιχεία του στο [0, 1], x ένα διάνυσμα πιθανοτήτων στο [n] και $k(m, \varepsilon) = \lceil \ln(2m)/(2\varepsilon^2) \rceil + 1$. Θα δειχθεί ότι για κάθε $\varepsilon > 0$, υπάρχει ένα k(m, ε)-ομοιόμορφο διάνυσμα πιθανοτήτων y στο [n] τέτοιο ώστε $||A \cdot x - A \cdot y||_\infty \leq \varepsilon$.

Ακολουθώντας παρόμοια διαδικασία με το (a) ερώτημα, προκύπτει το διάνυσμα:

$$y = \frac{1}{k(m,\varepsilon)} \sum_{i=1}^{k(m,\varepsilon)} {}^{i}y$$

Η πιθανότητα για το y να μην ισχύει η ιδιότητα $||A \cdot x - A \cdot y||_\infty \leq \varepsilon$ φράσσεται ως εξής:

$$\mathbf{Pr}[||A \cdot x - A \cdot y||_\infty > \varepsilon] \leq \mathbf{Pr}[\cup_{j=1}^{m}(|A_j x - A_j y| > \varepsilon)] \leq \sum_{j=1}^{m} \mathbf{Pr}[|A_j x - A_j y| > \varepsilon]$$, από Union-Bound

Εφαρμόζοντας Hoeffding Bound, προκύπτει:

$$\sum_{j=1}^{m} \mathbf{Pr}[|A_j x - A_j y| > \varepsilon] \leq \sum_{j=1}^{m} 2e^{-2k(m,\varepsilon)\varepsilon^2} = 2me^{-2k(m,\varepsilon)\varepsilon^2}$$

Χρησιμοποιώντας $k(m, \varepsilon) = \lceil \ln(2m)/(2\varepsilon^2) \rceil + 1$ διανύσματα ${}^{i}y$ για την κατασκευή του y, προκύπτει:

$$\mathbf{Pr}[||A \cdot x - A \cdot y||_\infty > \varepsilon] < 2me^{-\ln(2m)} = 1$$

Συνεπώς, η πιθανότητα το y να ικανοποιεί την ιδιότητα, θα είναι:

$$\mathbf{Pr}[||A \cdot x - A \cdot y||_\infty \leq \varepsilon] = 1 - \mathbf{Pr}[||A \cdot x - A \cdot y||_\infty > \varepsilon] > 0$$

Άρα, από την πιθανοτική μέθοδο, υπάρχει κάποιο y τέτοιο ώστε $||A \cdot x - A \cdot y||_\infty \leq \varepsilon$

Εναλλακτικά, για την απόδειξη, θέτοντας ως a_i τα m διανύσματα γραμμής του A, x το διάνυσμα πιθανοτήτων στο [n] και $x' = [a_1 x \ldots a_m x]$ προκύπτει ότι:

$||A \cdot x - A \cdot y||_\infty = ||x' - y'||_\infty = ||a_1(x - y), a_2(x - y), \ldots, a_m(x - y)||_\infty \leq |a_1 x - a_1 y| + \ldots + |a_m x - a_m y|$

Όπως και στο (a) ερώτημα, αποδεικνύεται $|a_1 x - a_1 y| \leq \frac{\varepsilon}{m}$

Συνεπώς, $||A \cdot x - A \cdot y||_\infty \leq |a_1 x - a_1 y| + \ldots + |a_m x - a_m y| \leq m \cdot \frac{\varepsilon}{m}$

<u>$3^η$ Άσκηση</u>

Crossing number cr(G) ενός γραφήματος G είναι ο ελάχιστος αριθμός συναντήσεων ακμών σε μια επίπεδη αποτύπωση του G. Το G είναι επίπεδο αν και μόνο αν cr(G)=0, ενώ π.χ., cr(K_5)=cr($K_{3,3}$)=1.

***(a)** Να δειχθεί ότι για κάθε απλό γράφημα G με n κορυφές και m ακμές, cr(G) ≥ m – 3n + 6.*

***(b)** Να δειχθεί ότι για κάθε απλό γράφημα G με n κορυφές και m ≥ 4n ακμές, cr(G) ≥ $m^3/(64n^2)$.*

Hint: Για το (a), να χρησιμοποιηθεί ότι για κάθε απλό επίπεδο γράφημα με n κορυφές και με m ακμές, m ≤ 3n – 6. Για το (b), να θεωρηθεί ένα τυχαίο επαγόμενο υπογράφημα G' του G και να υπολογιστεί το αναμενόμενο πλήθος κορυφών και ακμών στο G', καθώς και το αναμενόμενο cr(G'). Στην συνέχεια, να χρησιμοποιηθεί το (a).

(a) Θα δειχθεί ότι για κάθε απλό γράφημα G με n κορυφές και m ακμές, cr(G) ≥ m – 3n + 6.

Είναι γνωστό από τον τύπο του Euler, ότι κάθε επίπεδο γράφημα έχει m ≤ 3n – 6 ακμές.

Έστω, μια επίπεδη αποτύπωση του G. Τότε, εξ ορισμού θα υπάρχουν cr(G) σημεία όπου ακμές θα διασταυρώνονται. Για κάθε σημείο διασταύρωσης ακμών γίνεται μετατροπή σε μια κορυφή και το νέο γράφημα που δημιουργείται θα είναι το G". Από την κατασκευή του, θα ισχύει ότι cr(G") = 0, δηλαδή θα είναι επίπεδο. Το G" έχει n + cr(G) κορυφές και m + 2cr(G) ακμές αφού κάθε σημείο διασταύρωσης 2 ακμών έχει βαθμό 4, εφόσον προσθέτει 2 νέες ακμές. Συνεπώς, αφού το G" είναι επίπεδο, θα ισχύει ο τύπος του Euler: m' ≤ 3n' – 6 ⇒ m + 2cr(G) ≤ 3n + 3cr(G) – 6 ⇒ cr(G) ≥ m – 3n + 6.

(b) Θα δειχθεί ότι κάθε απλό γράφημα G με n κορυφές και m ≥ 4n ακμές, cr(G) ≥ $\mathrm{m}^3/(64\mathrm{n}^2)$.

Αρχικά, στρίβοντας ένα νόμισμα για κάθε κορυφή του G και με πιθανότητα p, επιλέγεται η κορυφή.

Μετά, σχηματίζεται το επαγόμενο υπογράφημα G' που θα περιέχει μόνο τις κορυφές που επιλέχθηκαν.

Έστω, τυχαίες μεταβλητές $\tilde{m}$, $\tilde{n}$, $\tilde{x}$ που αντιστοιχούν στον αριθμό των ακμών, στον αριθμό των κορυφών και στον crossing number του G' αντιστοίχως.

Κάθε κορυφή επιλέγεται με πιθανότητα p, οπότε το αναμενόμενο πλήθος κορυφών είναι: $\mathbf{E}[\tilde{n}] = p\cdot n$.

Στο G', για να υπάρχει μια ακμή πρέπει να επιλεχθούν και τα 2 άκρα της. Οπότε: $\mathbf{E}[\tilde{m}] = p^2\cdot m$.

Για την ύπαρξη ενός σημείου συνάντησης του G στο G' πρέπει να επιλεχθούν και οι δύο ακμές που διασταυρώνονται, δηλαδή και τα 4 άκρα των 2 ακμών. Οπότε: $\mathbf{E}[\tilde{x}] = p^4\cdot cr(G)$.

Έτσι, από το (a) ερώτημα θα ισχύει ότι:

$\tilde{x} - \tilde{m} + \tilde{n} \geq 0 \Rightarrow E[\tilde{x} - \tilde{m} + \tilde{n}] \geq 0 \Rightarrow p^4\cdot cr(G) - p^2\cdot m - p\cdot n \geq 0 \Rightarrow cr(G) \geq \frac{m}{p^2} - \frac{n}{p^3}$

Επίσης, ισχύει ότι m ≥ 4n, οπότε θέτοντας $p = \frac{4n}{m}$ προκύπτει: $cr(G) \geq \frac{1}{64}\frac{m^3}{n^2}$.

$4^η$ Άσκηση (Linear Integer Programming Formulations)

Να διατυπωθούν τα παρακάτω προβλήματα βελτιστοποίησης ως προβλήματα ακέραιου γραμμικού προγραμματισμού:

(a) *Δίνονται m παράλληλες υπολογιστικές μηχανές και n υπολογιστικές διεργασίες. Κάθε διεργασία i έχει χρόνο εκτέλεσης $p_{ij} \in \mathbb{N}^*$ στην μηχανή j. Ζητείται η ανάθεση κάθε διεργασίας σε μια υπολογιστική μηχανή ώστε ο συνολικός χρόνος εκτέλεσης των εργασιών στη μηχανή που έχει δεχθεί το μεγαλύτερο υπολογιστικό φορτίο να ελαχιστοποιηθεί.*

(b) *Δίνονται ένα σύνολο C με τις θέσεις n πελατών και ένα σύνολο F με m πιθανές θέσεις καταστημάτων. Δίνονται ακόμη, το κόστος $f_j \in \mathbb{N}$ για το άνοιγμα καταστήματος σε κάθε θέση $j \in F$ και οι αποστάσεις d: $C \times F \rightarrow \mathbb{N}$ για κάθε ζευγάρι θέσεων $(i, j) \in C \times F$. Ζητείται ένα υποσύνολο $F' \subseteq F$ θέσεων όπου θα ανοίξουν καταστήματα, ώστε να ελαχιστοποιηθεί το συνολικό κόστος για το άνοιγμα καταστήματος στις θέσεις του F' και τη συνολική απόσταση κάθε πελάτη $i \in C$ από το κοντινότερό του ανοικτό κατάστημα.*

(a) Δίνονται m υπολογιστικές μηχανές και n διεργασίες. Είναι γνωστό ότι κάθε διεργασία i έχει χρόνο εκτέλεσης $p_{ij} \in \mathbb{N}^*$ στην μηχανή j και αναζητείται μια ανάθεση των διεργασιών σε μηχανές έτσι ώστε να ελαχιστοποιηθεί το μεγαλύτερο υπολογιστικό φορτίο των μηχανών.

Έστω οι μεταβλητές $x_{ij} \in \{0, 1\}$ για τις οποίες ισχύει ότι η x_{ij} θα είναι ίση με 1 αν και μόνο αν ανατεθεί η διεργασία i στην μηχανή j. Εφόσον είναι θεμιτό κάθε διεργασία να ανατεθεί σε μια μηχανή, θα πρέπει να ισχύει: $\sum_{j=1}^{m} x_{ij} = 1$.

Έστω επίσης, μια μεταβλητή z η οποία θα αντιπροσωπεύει το μεγαλύτερο υπολογιστικό φορτίο που έχει δεχθεί κάποια μηχανή. Προφανώς η z θα είναι μεγαλύτερη ή ίση με κάθε υπολογιστικό φορτίο, οπότε θα ισχύει ότι: $\sum_{i=1}^{n} x_{ij} \cdot p_{ij} \leq z$.

Στόχος πλέον είναι η ελαχιστοποίηση του z.

Εύκολα παρατηρεί κανείς πως για κάθε fixed σύνολο feasible τιμών των x_{ij}, το ελάχιστο z θα είναι ακριβώς το ίσο με το μέγιστο υπολογιστικό φορτίο. Συνεπώς, αναζητείται το ελάχιστο z για όλα τα πιθανά feasible x_{ij}.

min: z

s.t.: $\sum_{j=1}^{m} x_{ij} = 1 \qquad \forall i \in [m]$

$\sum_{i=1}^{n} x_{ij} \cdot p_{ij} \leq z \qquad \forall j \in [n]$

$x_{ij} \in \{0, 1\}$

(b) Δίνονται ένα σύνολο C με τις θέσεις n πελατών και ένα σύνολο F με m πιθανές θέσεις καταστημάτων. Δίνονται ακόμη, το κόστος $f_j \in \mathbb{N}$ για το άνοιγμα καταστήματος σε κάθε θέση $j \in F$ και οι αποστάσεις $d: C \times F \rightarrow \mathbb{N}$ για κάθε ζευγάρι θέσεων $(i, j) \in C \times F$.

Έστω, μεταβλητές $x_j \in \{0, 1\}$, $\forall j \in F$ όπου το x_j θα είναι ίσο με 1 αν και μόνο αν στην λύση, επιλεχθεί το facility j.

Έστω, ακόμη, μια σταθερά $M = \max_{\substack{i \in C \\ j \in F}} d_{ij} + 1$, η οποία θα χρησιμοποιηθεί για την κωδικοποίηση κάποιων if-constraints.

Έστω, επίσης, μεταβλητές $p_i \geq 0$, $\forall i \in C$ που δείχνουν την απόσταση του πελάτη I από το κοντινότερο επιλεγμένο μαγαζί.

Τέλος, έστω μεταβλητές $y_{ij} \in \{0, 1\}$, $\forall (i, j) \in C \times F$, όπου το y_{ij} θα είναι ίσο με 1 αν και μόνο αν το κοντινότερο επιλεγμένο μαγαζί στον πελάτη i είναι το j.

Έτσι, η αντικειμενική συνάρτηση θα είναι το άθροισμα των αποστάσεων των πελατών από τα κοντινότερα σε αυτούς μαγαζιά, καθώς και τα κόστη ανοίγματος αυτών των μαγαζιών. Οπότε:

$$\sum_{i \in C} p_i + \sum_{j \in F} x_j \cdot f_j$$

Για τις y_{ij} πρέπει να ισχύει ότι αν το κατάστημα j δεν έχει επιλεχθεί στην λύση, τότε $y_{ij} = 0$. Αυτό εξασφαλίζεται με τα constraints: $x_j \geq y_{ij}$, $\forall (i, j) \in C \times F$

Επιπροσθέτως, κάθε πελάτης θα πρέπει να έχει ένα μαγαζί σε ελάχιστη απόσταση (αν υπάρχουν 2 στην ίδια απόσταση με αυτό το formulation, το πρόγραμμα μπορεί να διαλέξει όποιο από τα 2 θέλει):

$$\sum_{j \in F} y_{ij} = 1, \forall i \in C$$

Τέλος, επειδή θα πρέπει κάθε p_i να είναι ίσο με την απόσταση του κοντινότερου στο i επιλεγμένου μαγαζιού: $p_j \geq d_{ij} - M \cdot (1 - y_{ij})$, $\forall (i, j) \in C \times F$

Στο constraint αυτό, από την μια, αν το κατάστημα j δεν είναι το κοντινότερο επιλεγμένο στο i, τότε το $y_{ij} = 0$ και ο περιορισμός γίνεται $p_i \geq d_{ij} - M$, αφού $M > d_{ij}$ και $p_i > 0$ εξ ορισμού και από την άλλη, αν το j είναι το κοντινότερο στο i επιλεγμένο κατάστημα, τότε το constraint είναι μη-τετριμμένο, δηλ. $p_i \geq d_{ij}$. Όμως, ισχύει μόνο ένα τέτοιο constraint ανά πελάτη και ζητείται να ελαχιστοποιηθεί το άθροισμα p_i, θα πρέπει κάθε p_i να πάρει την τιμή της ελάχιστης απόστασης. Επομένως, το ακέραιο πρόγραμμα θα είναι το εξής:

min: $\sum_{i \in C} p_i + \sum_{j \in F} x_j \cdot f_j$

s.t.: $x_j \geq y_{ij}$, $\forall (i, j) \in C \times F$

$\sum_{j \in F} y_{ij} = 1$, $\forall i \in C$

$p_j \geq d_{ij} - M \cdot (1 - y_{ij})$, $\forall (i, j) \in C \times F$

$x_j \in \{0, 1\}$, $y_{ij} \in \{0, 1\}$, $p_i \geq 0$

<u>5η Άσκηση (Partial Vertex Cover)</u>

Δίνονται ένα γράφημα G(V, E) και μια παράμετρος β > 0. Ζητείται ένα υποσύνολο κορυφών C ⊆ V που ελαχιστοποιεί το β|C| + |U(C)|, όπου U(C) = {{u, v}∈ E: u, v ∉ C} είναι οι ακμές που δεν καλύπτονται από το C.

(a) *Να διατυπωθεί το παραπάνω πρόβλημα ως πρόβλημα (γραμμικού) ακέραιου προγραμματισμού, να δοθεί το αντίστοιχο LP relaxation, και να βρεθεί το αντίστοιχο δυϊκό γραμμικό πρόγραμμα. Να διατυπωθεί το πρόβλημα βελτιστοποίησης που περιγράφεται από το δυϊκό πρόγραμμα σε φυσική γλώσσα.*

(b) *Με βάση το LP Relaxation του (a), να διατυπωθεί ένας προσεγγιστικός αλγόριθμος που βασίζεται σε deterministic rounding και να αναλυθεί ο λόγος προσέγγισης που επιτυγχάνει ο αλγόριθμός.*

(a) Δίνεται ένα γράφημα G(V, E) και μια παράμετρος β > 0, ζητείται ένα υποσύνολο κορυφών C ⊆ V που ελαχιστοποιεί το β|C| + |U(C)|, όπου U(C) = {{u, v}∈ E: u, v ∉ C} είναι οι ακμές που δεν καλύπτονται από το C. Θα γραφτεί το πρόβλημα ως ακέραιο πρόγραμμα χρησιμοποιώντας ως μεταβλητές, τις εξής:

- $\forall u \in V$, $x_u \in \{0, 1\}$, όπου x_u θα είναι ίση με 1 αν και μόνο αν $u \in C$
- $\forall (u, v) \in E$, $y_{uv} \in \{0, 1\}$, όπου y_{uv} θα είναι ίση με 1 αν και μόνο αν $(u, v) \in U(C)$

Ο μοναδικός περιορισμός (constraint) που υπάρχει είναι ότι κάθε ακμή (u, v), είτε καλύπτεται από ένα κόμβο, είτε είναι ακάλυπτη. Δηλ.

$$x_u + x_v + y_{uv} \geq 1, \ \forall (u, v) \in E$$

Είναι προφανές ότι στην βέλτιστη λύση, δεν θα υπάρχει καμία ακμή που να καλύπτεται και να έχει μη-μηδενικό y_{uv} αφού στόχος είναι η ελαχιστοποίηση του αθροίσματος των y_{uv}.

Οπότε, το <u>ακέραιο πρόγραμμα</u> θα είναι:

min.: $\beta \sum_{u \in V} x_u + \sum_{(u,v) \in E} y_{uv}$

s.t.: $x_u + x_v + y_{uv} \geq 1, \quad \forall (u, v) \in E$

$x_u \in \{0, 1\}, y_{uv} \in \{0, 1\}$

Το <u>relaxed LP</u> θα είναι:

min.: $\beta \sum_{u \in V} x_u + \sum_{(u,v) \in E} y_{uv}$

s.t.: $x_u + x_v + y_{uv} \geq 1, \quad \forall (u, v) \in E$

$x_u \geq 0, y_{uv} \geq 0$

Ο πίνακας A, του παραπάνω προγράμματος, έχει n + m στήλες και m γραμμές αφού υπάρχει ένα constraint για κάθε ακμή του γραφήματος. Το δυϊκό πρόγραμμα θα έχει m μεταβλητές και n + m περιορισμούς.

Σε κάθε στήλη του A, αν η στήλη αντιστοιχεί σε μεταβλητή κορυφής τότε υπάρχουν άσσοι μόνο στις γραμμές που αντιστοιχούν σε προσπίπτουσες στην κορυφή ακμές ενώ αν η στήλη αντιστοιχεί σε μεταβλητή ακμής τότε υπάρχει άσσος μόνο στην γραμμή της αντίστοιχης ακμής.

Έτσι, το <u>δυϊκό πρόγραμμα</u> θα είναι:

max.: $\sum_{e \in E} z_e$

s.t.: $\sum_{e \in E: u \in e} z_e \leq \beta, \quad \forall u \in V$

$z_e \leq 1, \quad \forall e \in E$

Το δυϊκό πρόγραμμα έχει μια μεταβλητή για κάθε ακμή και μπορεί να διατυπωθεί ως εξής: Δίνεται ένας γράφος G(V, E), κάθε ακμή κοστολογείται με μια τιμή η οποία είναι το πολύ 1. Στόχος είναι η αύξηση των τιμών των ακμών έτσι ώστε να μεγιστοποιηθεί το άθροισμα όλων των τιμών, διατηρώντας τον περιορισμό πως οι προσκείμενες σε έναν κόμβο ακμές κοστίζουν αθροιστικά το πολύ β.

(b) Η προσέγγιση της λύσης του αρχικού προβλήματος θα γίνει λύνοντας το LP Relaxation και φτιάχνοντας μια εφικτή λύση μέσω deterministic rounding των μεταβλητών. Έστω OPT η βέλτιστη ακέραια λύση, SOL* η βέλτιστη λύση του relaxed LP και SOL η λύση που θα βρει ο αλγόριθμος για το αρχικό πρόβλημα. Έστω επίσης, x*, y* οι μεταβλητές της SOL* και $\tilde{x}$, $\tilde{y}$ οι μεταβλητές της SOL.

Αρχικά, θα κατασκευαστούν οι x, y από τις x*, y* κάνοντας round στο 1/3, δηλαδή μια μεταβλητή της SOL θα είναι 1 αν και μόνο αν η αντίστοιχη μεταβλητή του SOL* είναι μεγαλύτερη ή ίση του 1/3. Τυπικά, $x = \lfloor 3x^* \rfloor$ και $y = \lfloor 3y^* \rfloor$.

Για κάθε constraint $x_u + x_v + y_{uv} \geq 1$ του relaxed LP παρατηρείται ότι τουλάχιστον μια μεταβλητή από τις x_u^*, x_v^*, y_{uv}^* θα είναι μεγαλύτερη ή ίση του 1/3, αφού αν ήταν όλες μικρότερες από 1/3 τότε δεν θα ικανοποιούνταν ο περιορισμός. Επομένως, τουλάχιστον μια μεταβλητή από τις $\widetilde{x_u}$, $\widetilde{x_v}$, $\widetilde{y_{uv}}$ θα είναι 1, οπότε η SOL είναι feasible λύση.

Επιπλέον, ισχύει ότι OPT ≥ SOL*. Επίσης, από τον ορισμό του SOL, ισχύει ότι:

$\forall u \in V, \widetilde{x_u} = \lfloor 3x_u^* \rfloor \leq 3x_u^* \quad \& \quad \forall e \in E, y_e^* = \lfloor 3y_e^* \rfloor \leq 3y_e^* \quad \Rightarrow$

$\beta \sum_{u \in V} \widetilde{x_u} + \sum_{(u,v) \in E} \widetilde{y_{uv}} \leq 3(\beta \sum_{u \in V} x_u^* + \sum_{(u,v) \in E} y_{uv}^*) \quad \Rightarrow$

Τελικά ισχύει: SOL ≤ 3SOL* & SOL* ≤ OPT ⇒ SOL ≤ 3OPT

Οπότε, ο αλγόριθμος θα είναι 3-προσεγγιστικός για το αρχικό πρόβλημα.

<u>6η Άσκηση</u>

Να λυθεί η [1, Άσκηση 5.3] και η [1, Άσκηση 5.6].

<u>Άσκηση 5.3 από [1]</u>

In the maximum directed cut problem (sometimes called MAX DICUT) we are given as input a directed graph G = (V, A). Each directed arc (i, j)∈A has nonnegative weight w_{ij} ≥ 0. The goal is to partition V into two sets U and W = V – U so as to maximize the total weight of the arcs going from U to W (that is, arcs(i, j) with i∈U and j∈W). Give a randomized ¼-approximation algorithm for this problem.

Δίνεται ως είσοδος ένα κατευθυνόμενο γράφημα G = (V, A) με μη-αρνητικά βάρη w_{ij}. Στόχος είναι ο διαχωρισμό των κορυφών σε δύο σύνολα U, W = V\U, έτσι ώστε να μεγιστοποιηθεί το συνολικό βάρος των ακμών (u, v) με u∈U και v∈W.

Πιθανοκρατικός αλγόριθμος: με το στρίψιμο ενός νομίσματος για κάθε κορυφή αποφασίζεται με πιθανότητα ½ αν θα τοποθετηθεί στο σύνολο U ή στο σύνολο W. Τα ενδεχόμενα είναι ανεξάρτητα.

Θα δειχθεί ότι αυτός ο πιθανοτικός αλγόριθμος επιτυγχάνει λόγω της προσέγγισης ¼.

Έστω, η τυχαία μεταβλητή y_{uv}, για κάθε ακμή η οποία θα είναι ίση με 1 αν και μόνο u∈U ∩ v∈W:

$$\mathbf{Pr}[y_{uv} = 1] = \mathbf{Pr}[u \in U \cap v \in W] = \mathbf{Pr}[u \in U] \cdot \mathbf{Pr}[v \in W] = \frac{1}{4}$$

Η αναμενόμενη τιμή της αντικειμενικής συνάρτησης της λύσης, είναι:

$$\mathbf{E}[\mathrm{SOL}] = \mathbf{E}[\textstyle\sum_{(u,v)\in A} y_{uv} w_{uv}] = \sum_{(u,v)\in A} \mathbf{E}[y_{uv}] w_{uv} = \frac{1}{4} \sum_{(u,v)\in A} w_{uv}$$

Η βέλτιστη λύση φράσσεται άνω από το άθροισμα των βαρών όλων των ακμών, αφού μπορεί να περιέχει το πολύ όλες τις ακμές (π.χ. σε διμερή γραφήματα). Συνεπώς,

$$\mathbf{E}[\mathrm{SOL}] = \frac{1}{4} \sum_{(u,v)\in A} w_{uv} \qquad \& \qquad \mathrm{OPT} \le \sum_{(u,v)\in A} w_{uv} \quad \Rightarrow \qquad \mathbf{E}[\mathrm{SOL}] \le \frac{1}{4}\,\mathrm{OPT}$$

Προκύπτει λοιπόν, ότι ο παραπάνω αλγόριθμος είναι ¼-προσεγγιστικός για το πρόβλημα MAX DICUT.

<u>Άσκηση 5.6 από [1]</u>

Consider again the maximum directed cut problem from Exercise 5.3.

***(a)** Show that the following integer program models the maximum directed cut problem:*

max.: $\sum_{(i,j)\in A} w_{ij} z_{ij}$

s.t.: $z_{ij} \le x_i$, $\quad \forall (i, j) \in A$,

$z_{ij} \le 1 - x_j$, $\quad \forall (i, j) \in A$,

$x_i \in \{0, 1\}$, $\quad \forall i \in V$,

$0 \le z_{ij} \le 1$, $\quad \forall (i, j) \in A$

(b) *Consider a randomized rounding algorithm for the maximum directed cut problem that solves a linear programming relaxation of the integer program and puts vertex* $i \in U$ *with probability* $1/4 + x_i/2$*. Show that this gives a randomized ½-approximation algorithm for the maximum directed cut problem.*

(a) Θα δειχθεί ότι το παραπάνω integer program λύνει το maximum directed cut problem:

Έστω ότι η κορυφή i θα μπει στο σύνολο U αν και μόνο αν $x_i = 1$ αλλιώς θα είναι 0 όταν επιλέγεται το σύνολο W. Παρατηρείται επίσης, πως $z_{ij} = 1$ αν και μόνο αν $i \in U \cap j \in W$. Έτσι:

$$i \notin U \xrightarrow[z_{ij} \leq x_i]{x_i = 0} z_{ij} \leq 0 \qquad \& \qquad j \notin W \xrightarrow[z_{ij} \leq 1 - x_j]{x_j = 1} z_{ij} \leq 0$$

Συνεπώς, η αντικειμενική συνάρτηση μεγιστοποιεί το άθροισμα των βαρών των ακμών που έχουν κατεύθυνση από το U στο W.

Με άλλα λόγια, υπάρχουν οι εξής περιπτώσεις:

- $i \in U$ και $j \in U$: Από τις συνθήκες του LP, θα ισχύει $x_i = 1$ και $x_j = 1$ και προκύπτει $z_{ij} \leq 0 \equiv 0$. Συνεπώς, η ακμή και το βάρος της ορθώς δεν συνυπολογίζονται στο αποτέλεσμα.
- $i \in U$ και $j \in W$: Από τις συνθήκες του LP, θα ισχύει $x_i = 1$ και $x_j = 0$ και προκύπτει $z_{ij} \leq 1$. Συνεπώς, η ακμή και το βάρος της ορθώς συνυπολογίζονται στο αποτέλεσμα.
- $i \in W$ και $j \in U$: Από τις συνθήκες του LP, θα ισχύει $x_i = 0$ και $x_j = 1$ και προκύπτει $z_{ij} \leq 0 \equiv 0$. Συνεπώς, η ακμή και το βάρος της ορθώς δεν συνυπολογίζονται στο αποτέλεσμα.
- $i \in W$ και $j \in W$: Από τις συνθήκες του LP, θα ισχύει $x_i = 0$ και $x_j = 0$ και προκύπτει $z_{ij} \leq 0 \equiv 0$. Συνεπώς, η ακμή και το βάρος της ορθώς δεν συνυπολογίζονται στο αποτέλεσμα.

Οπότε, το δοθέν ακέραιο πρόγραμμα μοντελοποιεί ακριβώς το πρόβλημα MAC DICUT.

(b) Έστω μια τυχαία μεταβλητή y_{uv} για κάθε ακμή $(u, v) \in A$, η οποία η οποία θα είναι ίση με 1 αν και μόνο $u \in U \cap v \in W$.

Η λύση του αλγορίθμου θα έχει αντικειμενική τιμή:

$$\text{SOL} = \sum_{(u,v) \in E} y_{uv} w_{uv}$$

Η μέση τιμή μιας y_{uv} είναι:

$$\text{E}[y_{uv}] = 1 \cdot \mathbf{Pr}[y_{uv} = 1] = \mathbf{Pr}[u \in U \cap v \in W] = \mathbf{Pr}[u \in U] \cdot \mathbf{Pr}[v \in W] = (\tfrac{1}{4} + \tfrac{x_u}{2}) \cdot (1 - (\tfrac{1}{4} + \tfrac{x_v}{2})) \Rightarrow$$

$$\text{E}[y_{uv}] = (\tfrac{1}{4} + \tfrac{x_u}{2}) \cdot (\tfrac{3}{4} - \tfrac{x_v}{2}) \geq (\tfrac{1}{4} + \tfrac{z_{uv}}{2}) \cdot (\tfrac{1}{4} + \tfrac{z_{uv}}{2}) = (\tfrac{1}{4} + \tfrac{z_{uv}}{2})^2 \geq \tfrac{z_{uv}}{2}$$

Η αναμενόμενη τιμή της αντικειμενικής συνάρτησης της λύσης, είναι:

$$\mathbf{E}[\text{SOL}] = \mathbf{E}[\textstyle\sum_{(u,v) \in A} y_{uv} w_{uv}] = \sum_{(u,v) \in A} \mathbf{E}[y_{uv}] w_{uv} \geq \frac{1}{2} \sum_{(u,v) \in A} z_{uv} w_{uv}$$

Η OPT θα είναι μεγαλύτερη από την λύση του relaxed LP, οπότε:

$$\mathbf{E}[\text{SOL}] \geq \frac{1}{2} \sum_{(u,v) \in A} z_{uv} w_{uv} \qquad \& \qquad \text{OPT} \geq \sum_{(u,v) \in A} z_{uv} w_{uv} \qquad \Rightarrow \qquad \mathbf{E}[\text{SOL}] \geq \frac{1}{2} \text{OPT}$$

<u>Άσκηση 7 (Public Project)</u>

Η δημοτική αρχή μελετά την κατασκευή ενός νέου πάρκου, το οποίο αναμένεται να κοστίσει C ευρώ. Κάθε δημότης i, i = 1, ..., n, εκτιμά την ωφέλειά του από το νέο πάρκο σε $v_i \geq 0$ ευρώ, τιμή που είναι γνωστή μόνο στον ίδιο. Η δημοτική αρχή ζητάει από τους δημότες να δηλώσουν τις εκτιμήσεις τους v_1, ..., v_n, και θα προχωρήσει στην κατασκευή του πάρκου μόνο αν $\sum_{i=1}^{n} v_i \geq C$. Σε αυτή την περίπτωση, κάθε δημότης θα πληρώσει μια εισφορά $p_i \geq 0$. Η εισφορά έχει στόχο οι δημότες να δηλώσουν τις πραγματικές τους εκτιμήσεις, και όχι να καλύψει το συνολικό κόστος του πάρκου. Να σχεδιαστεί ένας φιλαλήθης (truthful) μηχανισμός για αυτό το πρόβλημα.

Έστω n δημότες με προσωπικές εκτιμήσεις u_1, …, u_n (που είναι γνωστές μόνο για αυτούς) για το όφελος που θα είχαν από την δημιουργία ενός πάρκου το οποίο κοστολογείται C ευρώ. Η δημοτική αρχή ζητάει από τους πολίτες να δηλώσουν τις εκτιμήσεις τους ε_1, …, ε_n και θα κατασκευαστεί το πάρκο αν και μόνο αν $\sum_{i=1}^{n} \varepsilon_i \geq C$. Είναι λογικό, όλοι οι πολίτες απλά να πουν C ανεξαρτήτως από το πόσο θέλουν να χτιστεί το πάρκο. Οπότε, για να αναγκαστούν να πουν την πραγματική τους εκτίμηση, κάθε δημότης καλείται να πληρώσει και μια εισφορά $p_i \geq 0$.

Για τον μηχανισμό λοιπόν, θα ισχύει ότι τα πιθανά allocation vectors είναι 2:

- $x_i = 1$, $\forall i \in [n]$, αν $\sum_{i=1}^{n} \varepsilon_i \geq C$
- $x_i = 0$, $\forall i \in [n]$, αν $\sum_{i=1}^{n} \varepsilon_i < C$

Το παραπάνω allocation rule είναι μονότονο οπότε, από το Myerson’s Lemma, υπάρχει payment rule p για το οποίο ο μηχανισμός (x, p) είναι truthful. Συγκεκριμένα, αν οριστεί ως S_{-i} το άθροισμα των εκτιμήσεων ε_{-i} εκτός του πολίτη i, τότε το payment rule θα είναι:

$$p(\varepsilon_i, \varepsilon_{-i}) = \begin{cases} 0, & \varepsilon_i \leq C - S_{-i} \\ C - S_{-i}, & \text{otherwise} \end{cases}$$

Αυτό το payment rule κάνει τον μηχανισμό truthful αφού προέρχεται από το Myerson’s Lemma.

Μπορεί να διατυπωθεί και με μια μικρή ανάλυση περιπτώσεων. Το utility του πολίτη i είναι $u_i(\varepsilon) = v_i \cdot x_i(\varepsilon) - p(\varepsilon)$. Οπότε, οι περιπτώσεις θα είναι οι εξής:

- Αν $v_i < C - S_{-i}$, τότε δηλώνοντας $\varepsilon_i = v_i$, ο δημότης έχει μηδενικό utility επειδή το πάρκο δεν θα χτιστεί. Επίσης, αν δήλωνε $\varepsilon_i < v_i$ θα είχε 0, ενώ αν δήλωνε $\varepsilon_i > v_i$ θα είχε αρνητικό utility.
- Αν $v_i = C - S_{-i}$, τότε θα είχε επίσης utility 0, όπως και για $\varepsilon_i < v_i$.
- Αν $v_i > C - S_{-i}$, τότε θα ισχύει $u(v_i, \varepsilon_{-i}) = v_i - C + S_{-i} = 0$
- Τέλος, για κάθε άλλο $\varepsilon_i < v_i$, θα ισχύει: $u(\varepsilon_i, \varepsilon_{-i}) = 0$ αν $\varepsilon_i < C - S_{-i}$ αφού τότε το πάρκο δεν θα χτιστεί και $u(\varepsilon_i, \varepsilon_{-i}) = v_i - C + S_{-i} = 0$ αν $\varepsilon_i > C - S_{-i}$.

Συνεπώς, σε όλα τα ενδεχόμενα, η δήλωση της πραγματικής εκτίμησης του πολίτη είναι η καλύτερη επιλογή.

Άσκηση 8 (Δημοπρασίες και Multiplicative Price Updates)

Έστω μια δημοπρασία για m διαφορετικά αντικείμενα (για λίγους απλότητας, υποτίθεται ότι κάθε αντικείμενο υπάρχει σε απεριόριστα αντίγραφα και ότι κάθε παίκτης μπορεί να πάρει το πολύ ένα αντίγραφο από κάθε αντικείμενο). Στην δημοπρασία συμμετέχουν n παίκτες. Κάθε παίκτης i έχει μια αύξουσα συνάρτηση αποτίμησης (valuation function) v_i*:* $2^{[m]} \rightarrow \mathbb{R}_{\geq 0}$ *που ορίζει πόσο αξίζει κάθε υποσύνολο αντικειμένων για αυτόν. Ο μηχανισμός αυτός ορίζει την αρχική τιμή κάθε αντικειμένου* $p_1^j = p_0 = L/(4m)$*, για κάποιο κατάλληλα επιλεγμένο L > 0. Στην συνέχεια, ζητά με τη σειρά από κάθε παίκτη* $i \in [n]$ *να επιλέξει το σύνολο* S_i *των αντικειμένων που μεγιστοποιεί τη συνάρτηση* $u_i(S_i) - \sum_{j \in S_i} p_i^j$ *και του το δίνει (ο παίκτης i μπορεί να επιλέξει το* $\emptyset$*, έτσι ισχύει πάντα ότι* $u_i(S_i) - \sum_{j \in S_i} p_i^j \geq 0$*). Πριν προχωρήσει στον επόμενο παίκτη, ο μηχανισμός διπλασιάζει τις τιμές όλων των αντικειμένων* $j \in S_i$ *πολλαπλασιαστικά, θέτοντας* $p_{i+1}^j = 2p_i^j$ *(ακόμη, τίθεται* $p_{i+1}^j = p_i^j$*,* $\forall j \notin S_i$*).*

(a) *Να διερευνηθεί αν ο παραπάνω μηχανισμός είναι φιλαλήθης (truthful) και να αιτιολογηθεί η απάντηση.*

(b) *Έστω ότι* $L = max_i\{u_i([m])\}$*. Να δειχθεί ότι ο μηχανισμός διαθέτει το πολύ* $1 + log_2(4m)$ *αντίγραφα από κάθε αντικείμενο.*

(c) *Να δειχθεί ότι αν* $L = max_i\{u_i([m])\}$*, τότε ο μηχανισμός επιτυγχάνει συνολική αξία* $\sum_i u_i(S_i) \geq 3V^*/8$*, όπου V* η συνολική αξία της βέλτιστης λύσης που όμως διαθέτει το πολύ ένα αντίγραφο από κάθε αντικείμενο. Για να αποδειχθεί αυτός ο ισχυρισμός, πρέπει πρώτα να δειχθεί (i) ότι λόγω της πολλαπλασιαστικής αναπροσαρμογής των τιμών, στο τέλος του μηχανισμού ισχύει ότι*

$$\sum_{i\in[n]} u_i(S_i) \geq \sum_{j\in[m]} p_{n+1}^j - mp_0 \Rightarrow \sum_i u_i(S_i) \geq \sum_{j\in[m]} p_{n+1}^j - \frac{L}{4},$$

και (ii) ότι επειδή η βέλτιστη λύση δίνει το πολύ ένα αντίγραφο από κάθε αντικείμενο,

$$\sum_{i\in[n]} u_i(S_i) \geq V^* - \sum_{j\in[m]} p_{n+1}^j$$

(a) Ο μηχανισμός που περιγράφεται στην εκφώνηση είναι truthful, αφού ο κάθε παίκτης έχει την επιλογή να αγοράσει με κάποιες αυξανόμενες ανά γύρο τιμές (οι οποίες επηρεάζονται από αυτόν) διάφορα υποσύνολα αντικειμένων. Δεν έχει κίνητρο να αφήσει τα αντικείμενα για επόμενο γύρο, αφού τότε θα του προσφέρουν σίγουρα μικρότερο utility. Αντίθετα, αν οι τιμές στον πρώτο γύρο ξεκινούσαν από κάποια μεγάλη σταθερά και υποδιπλασιάζονταν, τότε οι παίχτες θα είχαν κίνητρο να περιμένουν αρκετούς γύρους ώστε να πάρουν φθηνά τα αντικείμενα (μάλιστα αν υπάρχουν και απεριόριστα αντίγραφα, θα τα έπαιρναν σχεδόν δωρεάν).

Με άλλα λόγια, κάθε παίκτης διατηρεί μια συνάρτηση προσωπικής αξίας $u_i(S_i)$ για κάθε υποσύνολο των αντικειμένων (S_i) και καλείται να επιλέξει το σύνολο εκείνο που μεγιστοποιεί την

$$f = u_i(S_i) - \sum_{j\in S_i} p_i^j$$

Η f αποτελεί μια συνάρτηση ωφέλειας και μεγιστοποιώντας την προκύπτει το βέλτιστο σύνολο αντικειμένων (αυτό που έχει τη μεγαλύτερη συνολική αξία συμπεριλαμβανομένης της τιμής των αντικειμένων του) ώστε να το επιλέξει ο παίκτης. Επομένως, ο μηχανισμός είναι truthful αφού ωθεί τους παίκτες να εκφράζουν την πραγματική τους ωφέλεια ως dominant strategy.

(b) Έστω ότι L = $\max_i\{u_i([m])\}$. Οι αρχικές τιμές των αντικειμένων είναι $p_0 = L/(4m)$. Αν το παιχνίδι διαρκέσει περισσότερους από $k = 1 + \log_2(4m)$ γύρους, τότε οι τιμές θα έχουν φτάσει:

$$p_k = 2^k p_0 > 4m \cdot \frac{L}{4m} = \max_{i \in [n]} u_i([m])$$

Αυτό σημαίνει ότι κάθε αντικείμενο ξεχωριστά θα κοστίζει περισσότερο από το μέγιστο valuation για όλα τα αντικείμενα. Συνεπώς, από αυτό το σημείο και μετά κανένας παίκτης δεν θα αγοράσει τα αντικείμενα, δηλαδή διατίθενται για το πολύ $1 + \log_2(4m)$ γύρους.

Με άλλα λόγια, έστω ότι σε κάποιο στάδιο του μηχανισμού, ένα αντικείμενο j θα έχει δοθεί σε k αντίγραφα. Τότε, στο επόμενο στάδιο θα επιλεγεί από τον παίκτη i αν η προστιθέμενη αξία που προσφέρει στο σύνολο της επιλογής του είναι μεγαλύτερη της τιμής του, όπως αυτή έχει πολλαπλασιαστικά αυξηθεί ανά τους γύρους:

$$u_i(S_i) - u_i(S_i\{j\}) \geq p_i^j = \frac{L}{m} r^k$$

Οπότε, φαίνεται πως η τιμή του αντικειμένου θα γίνει ίση με L μετά από $\log_r m$ πωλήσεις. Έτσι, μόνο ο παίκτης με αξία μικρότερη από L δεν θα μπορεί να το αγοράσει. Αν προστεθεί στον λογαριασμό και το αρχικό, προκύπτει ότι ο μηχανισμός θα πρέπει να διαθέτει το πολύ $2 + \log_r m$ αντίγραφα από κάθε αντικείμενο.

(c) 1ος Τρόπος Επίλυσης:

Έστω p_e^* η τελική τιμή του αντικειμένου e. Σε κάθε αγορά συνόλου αντικειμένων από κάποιον παίκτη i στον γύρο j ισχύει ότι $u_i(S_i) \geq \sum_{e \in S_i} p_e^j$. Επομένως, στο τέλος του αλγορίθμου, θα ισχύει αθροιστικά για όλους τους παίκτες ότι:

$$\sum_{i \in [n]} u_i(S_i) \geq \sum_{i \in [n]} \sum_{e \in S_i} p_e^j$$

Το δεύτερο μέλος αποτελεί την συνολική τιμή που πουλήθηκαν όλα τα αντίγραφα των αντικειμένων. Αν κάθε αντικείμενο πουλήθηκε l_e^* φορές, τότε η παραπάνω σχέση, γράφεται:

$$\sum_{i \in [n]} \sum_{e \in S_i} p_e^j = p_0 \cdot \sum_{e \in U} \sum_{i=1}^{l_e^*-1} 2^i = p_0 \cdot \sum_{e \in U} (2^{l_e^*} - 1)$$

Όμως, $p_0 \cdot l_e^*$ είναι η τελική τιμή του αντικειμένου e. Οπότε:

$$p_0 \cdot \sum_{e \in U} (2^{l_e^*} - 1) = \sum_{e \in U} p_e^* - m p_0 = \sum_{e \in U} p_e^* - \frac{L}{4}$$

Δηλαδή,

$$\sum_{i \in [n]} u_i(S_i) \geq \sum_{e \in U} p_e^* - \frac{L}{4} \qquad (1)$$

Επιπλέον, έστω S_i^* το σύνολο των αγαθών που παίρνει ο i στην OPT λύση. Όμως, επειδή στον k γύρο, ο παίκτης i διάλεξε το S_i, αυτό είχε την μεγαλύτερη διαφορά $u_i(S_i) - \sum_{j \in S_i} p_j^k$, ισχύει ότι:

$$u_i(S_i) - \sum_{j \in S_i} p_j^k \geq u_i(S_i^*) - \sum_{j \in S_i^*} p_j^k \Rightarrow u_i(S_i) \geq u_i(S_i^*) - \sum_{j \in S_i^*} p_j^k - \sum_{j \in S_i} p_j^k \geq u_i(S_i^*) - \sum_{j \in S_i^*} p_j^k$$

Αθροίζοντας για κάθε παίκτη:

$$\sum_{i\in[n]} u_i(S_i) \geq \sum_{i\in[n]} u_i(S_i^*) - \sum_{i\in[n]} \sum_{j\in S_i^*} p_j^k \geq \sum_{i\in[n]} u_i(S_i^*) - \sum_{e\in U} p_e^* \Rightarrow$$

$$\sum_{i\in[n]} u_i(S_i) \geq \sum_{i\in[n]} u_i(S_i^*) - \sum_{e\in U} p_e^* \qquad (2)$$

Από την άθροιση των (1), (2) κατά μέλη, προκύπτει:

$$2\sum_{i\in[n]} u_i(S_i) \geq \sum_{i\in[n]} u_i(S_i^*) - \frac{\max_i\{u_i([m])\}}{4} \geq \sum_{i\in[n]} u_i(S_i^*) - \frac{\sum_{i\in[n]} u_i(S_i^*)}{4} \Rightarrow$$

$$2\mathrm{SOL} \geq \mathrm{OPT} - \frac{\mathrm{OPT}}{4} \Rightarrow \mathrm{SOL} \geq \frac{3}{8}\,\mathrm{OPT}$$

<u>2ος Τρόπος Επίλυσης</u>:

Εναλλακτικά, έστω το αντικείμενο j επιλέγεται k_j φορές. Τότε, από τη σχέση αθροίσματος γεωμετρικής προόδου, θα προκύψει η ακόλουθη σχέση για κάθε αντικείμενο (παίκτη) i όταν ολοκληρωθεί ο μηχανισμός (n γύροι):

$$\sum_i u_i(S_i) \geq \frac{L}{m}\left(\frac{r^{k_1}-1}{r-1} + \frac{r^{k_2}-1}{r-1} + \ldots + \frac{r^{k_m}-1}{r-1}\right) = \frac{L}{m(r-1)}\left(r^{k_1} + r^{k_2} + \ldots + r^{k_m} - m\right) \rightarrow$$

$$(r-1)\sum_i u_i(S_i) \geq \sum_j \frac{L}{m} r^{k_j} - L = \sum_j p_{n+1}^j - L$$

Έστω ότι, για την βέλτιστη λύση V*, ο παίκτης i έχει επιλέξει το σύνολο S_i, ανεξάρτητο από άλλες επιλογές με βάση την εκφώνηση. Αν τώρα ο παίκτης επιλέξει το σύνολο S_i', θα ισχύει ότι:

$$u_i(S_i') - \sum_{j\in S_i'} p_i^j \geq u_i(S_i) - \sum_{j\in S_i} p_i^j \rightarrow$$

$$u_i(S_i') - u_i(S_i) \geq \sum_{j\in S_i'\setminus S_i} p_i^j - \sum_{j\in S_i\setminus S_i'} p_i^j \geq -\sum_{j\in S_i\setminus S_i'} p_{n+1}^j \geq -\sum_{j\in S_i} p_{n+1}^j$$

Εφόσον πρόκειται για ξένα σύνολα, αθροίζοντας για κάθε παίκτη i, προκύπτει:

$$\sum_i u_i(S_i) \geq V^* - \sum_j p_{n+1}^j$$

Συνεπώς, συνδυάζοντας τα παραπάνω αποτελέσματα, προκύπτει ότι:

$$r\sum_i u_i(S_i) \geq \sum_i u_i(S_i) + \sum_j p_{n+1}^j - L \geq V^* - L \geq \frac{V^*}{2} \rightarrow \sum_i u_i(S_i) \geq \frac{V^*}{2r}$$

<u>*Resources*</u>:

1. *D.P. Williamson and D.B. Shmoys. The Design of Approximation Algorithms. Cambridge University Press, 2010.*
2. *M. Mitzenmacher and E. Upfal. Probability and Computing: Randomized Algorithms and Probabilistic Analysis. Cambridge University Press, 2005*
3. *Prahladh Harsha, Limits of Approximation Algorithms (MAXCUT and Introduction to Inapproximability)*
4. *Christos Papadimitriou, CS294 P29: Algorithmic Game Theory (Mechanism Design)*
5. *Avrim Blum, Algorithms, Games and Networks (Lecture 14), February 2013*
6. *Διαφάνειες Μαθήματος*