Εθνικό Μετσόβιο Πολυτεχνείο

Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Προηγμένα Θέματα Αλγορίθμων

Αλγόριθμοι Δικτύων και Πολυπλοκότητα

Εαρινό εξάμηνο 2025-2026

(ΕΜΠ – ΑΛΜΑ)

Διδάσκων: Δημήτρης Φωτάκης

1η Σειρά Ασκήσεων

Προθεσμία παράδοσης: 30/4/2026

Άσκηση 1 (Online Learning και PAC Learning, 1 μον.). Θεωρούμε τον Αλγόριθμο Υποδιπλασιασμού (Halving Algorithm, όπως παρουσιάζεται στις διαφάνειες τις αντίστοιχης διάλεξης), που επιδιώκει να ελαχιστοποιήσει το πλήθος των λαθών σε περιβάλλον online learning. Υποθέτουμε ότι η κλάση υποθέσεων $\mathcal{H}$ είναι πεπερασμένη και τα δείγματα κατηγοριοποιούνται με βάση υπόθεση $f \in \mathcal{H}$ (realizability).

Να δείξετε ότι αν τα δείγματα $(x_t, f(x_t))$ έρχονται από μια οποιαδήποτε (άγνωστη, αλλά συγκεκριμένη) κατανομή $\mathcal{D}$ και το σύνολο έγκυρων υποθέσεων S_t που διατηρεί ο αλγόριθμος δεν μεταβάλλεται για $\Omega(\log(1/\delta)/\varepsilon)$ συνεχόμενα δείγματα, τότε με πιθανότητα τουλάχιστον $1-\delta$, κάθε έγκυρη υπόθεση $h \in S_t$ επιτυγχάνει loss $L_{(\mathcal{D},f)}(h) \le \varepsilon$ (δηλαδή έχουμε επιτύχει την εγγύηση του PAC Learning). Ποια είναι η δειγματική πολυπλοκότητα του αλγορίθμου για πεπερασμένη κλάση υποθέσεων $\mathcal{H}$;

Άσκηση 2 (Ανάλυση Regret του Αλγόριθμου Hedge, 1.5 μον.). Θεωρούμε την παρακάτω γενίκευση του Randomized Weighted Majority Algorithm για online learning / online linear optimization και αναλύουμε το regret που επιτυγχάνει.

- **Input**: n actions $\{1,\dots,n\}$, time horizon T, $\boldsymbol{w}_1 = (1,\dots,1)$, $\boldsymbol{x}_1 = (1/n,\dots,1/n)$
- for $t = 1$ to T do:
  - Select action $i_t \in \{1,\dots,n\}$ with probability $\boldsymbol{x}_t(i_t)$
  - Get loss $\boldsymbol{\ell}_t \in [0,1]^n$ for all actions and incur loss $\boldsymbol{\ell}_t(i_t)$
  - Update weights $\boldsymbol{w}_{t+1}(i) = \boldsymbol{w}_t(i) e^{-\eta \boldsymbol{\ell}_t(i)}$, for all $i \in \{1,\dots,n\}$
  - Update probabilities $\boldsymbol{x}_{t+1}(i) = \frac{\boldsymbol{w}_{t+1}(i)}{\sum_{i=1}^n \boldsymbol{w}_{t+1}(i)}$, for all $i \in \{1,\dots,n\}$

(α) Θεωρούμε τη συνάρτηση δυναμικού $\Phi(t) = \sum_{i=1}^n \boldsymbol{w}_t(i)$ (συνολικό βάρος εμπιστοσύνης τη χρονική στιγμή t). Αρχικά είναι $\Phi(1) = n$. Να δείξετε ότι $\Phi(T+1) \ge e^{-\eta \sum_{t=1}^T \boldsymbol{\ell}_t(i^*)}$, όπου $i^* = \arg\min_i \sum_{t=1}^T \boldsymbol{\ell}_t(i)$ η επιλογή (action) με το ελάχιστο συνολικό κόστος (loss).

(β) Να δείξετε ακόμη ότι για κάθε $t \ge 1$,

$$\Phi(t+1) \le \Phi(t)\, e^{-\eta\, \boldsymbol{x}_t \cdot \boldsymbol{\ell}_t + \eta^2 \boldsymbol{x}_t \cdot \boldsymbol{\ell}_t^2}\,,$$

όπου $\boldsymbol{\ell}_t^2 = (\ell_t^2(1),\dots,\ell_t^2(n))$, για όλα τα $i \in \{1,\dots,n\}$. Χρησιμοποιώντας αυτή τη σχέση, να βρείτε ένα άνω φράγμα για το $\Phi(T+1)$ ως συνάρτηση του $\Phi(1) = n$.

(γ) Χρησιμοποιώντας τα άνω και κάτω φράγματα για το $\Phi(T+1)$ που υπολογίσατε στα (α) και (β) και το γεγονός ότι $\boldsymbol{\ell}_t \in [0,1]^n$, για κάθε $t \in \{1,\dots,T\}$, να δείξετε ότι:

$$\text{Exp-Regret}(T) = \sum_{t=1}^T \boldsymbol{x}_t \cdot \boldsymbol{\ell}_t(i_t) - \sum_{t=1}^T \boldsymbol{\ell}_t(i^*) \le \eta\, T + \ln(n)/\eta$$

Ποια τιμή θα επιλέγατε για το η ώστε να ελαχιστοποιήσετε το άνω φράγμα στο Exp-Regret; Ποιο είναι το Exp-Regret(T) που επιτυγχάνει ο αλγόριθμος για αυτή την τιμή του η;

Άσκηση 3 (Set Cover – Αλγόριθμοι Προσέγγισης, 1.5 μον.). (α) Να γενικεύσετε τον προσεγγιστικό αλγόριθμο για το πρόβλημα Vertex Cover που βασίζεται στο maximal matching σε έναν f–προσεγγιστικό αλγόριθμο για την unweighted εκδοχή του προβλήματος Set Cover (f είναι το μέγιστο πλήθος συνόλων στα οποία μπορεί να ανήκει κάποιο στοιχείο). Να αποδείξετε την ορθότητά του αλγορίθμου σας και τον λόγο προσέγγισης που επιτυγχάνει. Να βρείτε tight example για τον λόγο προσέγγισης του αλγορίθμου σας.

(β) Να εξηγήσετε γιατί ο παραπάνω αλγόριθμος δεν επιτυγχάνει λόγο προσέγγισης f για την weighted εκδοχή του προβλήματος Set Cover. Να δώσετε κατάλληλο αντιπαράδειγμα.

(γ) Να διατυπώσετε f–προσεγγιστικό αλγόριθμο για τη weighted εκδοχή του προβλήματος Set Cover με βάση το αντίστοιχο Linear Programming relaxation.

Άσκηση 4 (**Maximum Coverage – Αλγόριθμοι Προσέγγισης, 1 μον.**). Να αναλύσετε τον λόγο προσέγγισης που επιτυγχάνει ο άπληστος αλγόριθμος για το πρόβλημα Maximum Coverage στις παρακάτω περιπτώσεις:

1. Κάθε στοιχείο e έχει θετικό βάρος $w(e)$ και επιδιώκουμε τη μεγιστοποίηση του συνολικού βάρους των καλυπτόμενων στοιχείων (επιλέγοντας k σύνολα).
2. Κάθε σύνολο S έχει θετικό κόστος $c(S)$ και δίνεται (αντί για το πλήθος συνόλων k) ένα budget B. Επιδιώκουμε να μεγιστοποιήσουμε το πλήθος των καλυπτόμενων στοιχείων (τα στοιχεία δεν έχουν βάρη σε αυτή την περίπτωση), επιλέγοντας σύνολα με συνολικό κόστος το πολύ B.

Άσκηση 5 (**Minimize Makespan on Identical Machines – Αλγόριθμοι Προσέγγισης, 1.5 μον.**). Δίνονται m παράλληλες υπολογιστικές μηχανές και σύνολο J με n υπολογιστικές διεργασίες. Κάθε διεργασία $i \in J$ έχει χρόνο εκτέλεσης $p_i \in \mathbb{N}^*$ και μπορεί να ανατεθεί σε οποιαδήποτε μηχανή. Ζητείται να αναθέσουμε κάθε διεργασία σε μία υπολογιστική μηχανή ώστε ο συνολικός χρόνος εκτέλεσης των εργασιών στη μηχανή που δέχεται το μεγαλύτερο υπολογιστικό φορτίο να είναι ελάχιστος (makespan minimization). Τυπικά, ζητείται μια διαμέριση του J σε m υποσύνολα $J_1, \dots, J_m$, με $\bigcup_{k=1}^m J_k = J$, ώστε να ελαχιστοποιηθεί η ποσότητα (η οποία είναι γνωστή ως makespan):

$$\max_k \sum_{i \in J_k} p_i$$

Αναλύουμε τον λόγο προσέγγισης του άπληστου αλγόριθμου για το αυτό το πρόβλημα. Ο άπληστος αλγόριθμος ταξινομεί τις διεργασίες σε φθίνουσα σειρά μεγέθους $p_1 \geq p_2 \geq \cdots \geq p_n$ και αναθέτει την επόμενη διεργασία στη μηχανή με το ελάχιστο υπολογιστικό φορτίο στην τρέχουσα ανάθεση. Έστω q η μηχανή με το μέγιστο υπολογιστικό φορτίο στο τέλος του αλγόριθμου και έστω ℓ η διεργασία που ανατέθηκε τελευταία στην q.

Στην αντίστοιχη διάλεξη, αποδείξαμε πως στο τέλος του άπληστου αλγόριθμου, το υπολογιστικό φορτίο της q είναι το πολύ $P_{\text{tot}}/m + p_\ell$, όπου $P_{\text{tot}} = \sum_{i=1}^n p_i$. Με βάση αυτό (και χρησιμοποιώντας μόνο ότι $p_\ell \leq \max_i\{p_i\} \leq \text{OPT}$, όπου OPT το makespan που επιτυγχάνει η βέλτιστη λύση), αποδείξαμε ότι ο λόγος προσέγγισης του άπληστου αλγόριθμου είναι το πολύ $2 - \frac{1}{m}$. Στη συνέχεια, προσδιορίζουμε τον ακριβή λόγο προσέγγισης χρησιμοποιώντας ότι ο άπληστος αλγόριθμος δρομολογεί τις διεργασίες σε φθίνουσα σειρά μεγέθους.

1. Να δείξετε πως αν $\ell \leq m$, η λύση που υπολογίζει ο άπληστος αλγόριθμος είναι βέλτιστη. Με βάση αυτό, να δείξετε πως αν $\ell \geq m+1$, τότε (i) $p_\ell \leq p_{m+1} \leq \text{OPT}/2$, και (ii) ο λόγος προσέγγισης του άπληστου αλγόριθμου είναι το πολύ $\frac{3}{2} - \frac{1}{2m}$.
2. Αποδεικνύεται (μπορείτε να το θεωρήσετε δεδομένο) πως αν $p_\ell > \text{OPT}/3$, η λύση που υπολογίζει ο άπληστος αλγόριθμος είναι βέλτιστη. Με βάση αυτό, να δείξετε πως ο λόγος προσέγγισης του άπληστου αλγόριθμου είναι το πολύ $\frac{4}{3} - \frac{1}{3m}$.
3. Να περιγράψετε μια οικογένεια συνόλων διεργασιών J_m, όπου για κάθε $m \geq 2$, $|J_m| = 2m+1$ και ο άπληστος αλγόριθμος έχει λόγο προσέγγισης ίσο με $\frac{4m-1}{3m}$.

Άσκηση 6 (**Linear Integer Programming Formulations, 1.5 μον.**). (α) Να διατυπώσετε τα παρακάτω προβλήματα βελτιστοποίησης ως προβλήματα ακέραιου γραμμικού προγραμματισμού:

1. Δίνονται m παράλληλες υπολογιστικές μηχανές και n υπολογιστικές διεργασίες. Κάθε διεργασία i έχει χρόνο εκτέλεσης $p_{ij} \in \mathbb{N}^*$ στην μηχανή j. Ζητείται να αναθέσουμε κάθε διεργασία σε μία υπολογιστική μηχανή ώστε ο συνολικός χρόνος εκτέλεσης των εργασιών στη μηχανή που δέχεται το μεγαλύτερο υπολογιστικό φορτίο να είναι ελάχιστος.

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

(β) Στο ακέραιο γραμμικό πρόγραμμα του (α.1), αντικαθιστούμε τους περιορισμούς ακεραιότητας των μεταβλητών με περιορισμούς μη αρνητικότητας. Θεωρούμε μια βέλτιστη λύση του γραμμικού προγράμματος που προκύπτει η οποία αντιστοιχεί σε basic feasible solution. Πόσες (το πολύ) μπορεί να είναι οι μεταβλητές με θετική τιμή σε αυτή τη λύση;

Άσκηση 7 (Δυϊκότητα και Γραμμικά Συστήματα, 0.75 μον.). Έστω $\boldsymbol{A}$ πίνακας $m \times n$, $\boldsymbol{x}$ ένα n-διάνυσμα μεταβλητών και $\boldsymbol{b}$ ένα m-διάνυσμα. Το γραμμικό σύστημα $\boldsymbol{Ax} \leq \boldsymbol{b}$ καλείται *μη-συμβιβαστό* αν υπάρχει m-διάνυσμα $\boldsymbol{y}$ τέτοιο ώστε $\boldsymbol{A}^T\boldsymbol{y} = 0$, $\boldsymbol{b}^T\boldsymbol{y} < 0$ και $\boldsymbol{y} \geq 0$. Να αποδείξετε ότι το σύστημα $\boldsymbol{Ax} \leq \boldsymbol{b}$ είναι *μη-επιλύσιμο* αν και μόνο αν είναι *μη-συμβιβαστό*.

Άσκηση 8 (Κυρτότητα Συνόλου Βέλτιστων Λύσεων Γραμμικού Προγράμματος, 0.75 μον.). Να δείξετε ότι το σύνολο των βέλτιστων λύσεων ενός γραμμικού προγράμματος είναι κυρτό. Ειδικότερα, θεωρούμε το γραμμικό πρόγραμμα

$$\mathrm{LP} = \min\{\, \boldsymbol{c}^T\boldsymbol{x} : \boldsymbol{Ax} \geq \boldsymbol{b}, \boldsymbol{x} \geq \boldsymbol{0} \,\},$$

όπου $\boldsymbol{A}$ πίνακας $m \times n$, $\boldsymbol{x}$ ένα n-διάνυσμα μεταβλητών, $\boldsymbol{b}$ ένα m-διάνυσμα και $\boldsymbol{c}$ ένα n-διάνυσμα. Έστω $\boldsymbol{x}_1$ και $\boldsymbol{x}_2$ δύο βέλτιστες λύσεις του LP. Να δείξετε πως για κάθε $\lambda \in [0, 1]$, η $\lambda\boldsymbol{x}_1 + (1 - \lambda)\boldsymbol{x}_2$ είναι επίσης βέλτιστη λύση του LP.

Άσκηση 9 (Walrasian equilibrium, 2.5 μον.). Θεωρούμε σύνολο m ετερογενών αντικειμένων $U = \{1, \dots, m\}$ και σύνολο n παικτών $N = \{1, \dots, n\}$ που ανταγωνίζονται για αυτά. Κάθε παίκτης i έχει *συνάρτηση αποτίμησης* (valuation function) $v_i : 2^U \to \mathbb{N}$ που καθορίζει την αξία $v_i(S)$ κάθε συνόλου αντικειμένων $S \subseteq U$ για τον i. Θεωρούμε ότι οι συναρτήσεις v_i είναι κανονικοποιημένες ώστε $v_i(\emptyset) = 0$ και μονότονες, δηλ. για κάθε $S \subseteq T$, $v_i(S) \leq v_i(T)$. Μια ανάθεση αγαθών στους παίκτες είναι μια συλλογή $\mathcal{S} = (S_1, \dots S_n)$ υποσυνόλων του U που ανά δύο είναι ξένα μεταξύ τους. Θέλουμε να υπολογίσουμε μια ανάθεση $\mathcal{S} = (S_1, \dots S_n)$ που μεγιστοποιεί την κοινωνική ωφέλεια (social welfare) $\mathrm{sw}(\mathcal{S}) = \sum_{i=1}^{n} v_i(S_i)$.

(α) Να δείξετε ότι το (LP) αποτελεί μια χαλάρωση (relaxation) του προβλήματος μεγιστοποίησης της κοινωνικής ωφέλειας και ότι το (DP) αποτελεί το δυϊκό του (LP).

$$(\mathrm{LP}) \quad \begin{aligned} \max \; & \sum_{i=1}^{n} \sum_{S \subseteq U} x_{i,S} v_i(S) \\ \text{s.t.} \; & \sum_{S \subseteq U} x_{i,S} \leq 1 && \forall i \in N \\ & \sum_{S : j \in S} \sum_{i=1}^{n} x_{i,S} \leq 1 && \forall j \in U \\ & x_{i,S} \geq 0 \end{aligned}$$

$$\begin{aligned} \min \; & \sum_{i \in N} u_i + \sum_{j \in U} p_j \\ \text{s.t.} \; & u_i \geq v_i(S) - \sum_{j \in S} p_j && \forall i \in N,\ \forall S \subseteq U \\ & u_i \geq 0,\ p_j \geq 0 \end{aligned} \quad (\mathrm{DP})$$

(β) Να διατυπώσετε τα complementary slackness conditions για βέλτιστη λύση $(x_{i,S})_{i \in N, S \subseteq U}$ του (LP) και βέλτιστη λύση $((u_i)_{i \in N}, (p_j)_{j \in U}))$ του (DP). Να δείξετε ακόμη ότι σε κάθε βέλτιστη λύση του (DP), $u_i = \max_{S \subseteq U}\{v_i(S) - p(S)\}$, όπου χρησιμοποιούμε τη συντομογραφία $p(S) = \sum_{j \in S} p_j$.

(γ) Ερμηνεύουμε τις δυϊκές μεταβλητές p_j ως τιμές για τα αγαθά $j \in U$ και τις δυϊκές μεταβλητές u_i ως την ωφέλεια (utility, αξία $v_i(S)$ μείον κόστος $p(S)$ για σύνολο αγαθών S) των παικτών $i \in N$ από μια βέλτιστη επιλογή $\arg\max_{S \subseteq U}\{v_i(S) - p(S)\}$ αγαθών για τον παίκτη i στις τιμές $(p_1, \dots, p_n)$.

Μια ανάθεση αγαθών $\mathcal{S}^* = (S_1^*, \ldots, S_n^*)$ στους παίκτες και ένα σύνολο τιμών $(p_1, \ldots, p_n)$ για τα αγαθά αποτελούν μια *Walrasian ισορροπία* (Walrasian ή competitive equilibrium) αν

1. το σύνολο αγαθών S_i^* που ανατίθεται σε κάθε παίκτη i αποτελεί βέλτιστη επιλογή για τον i στις τιμές $(p_1, \ldots, p_n)$, δηλ. έχουμε ότι $v_i(S_i^*) - p(S_i^*) = \max_{S \subseteq U}\{v_i(S) - p(S)\}$, και
2. κάθε αγαθό j με θετική τιμή ανατίθεται σε κάποιον παίκτη, δηλ. $\forall\, j$ με $p_j > 0$, $j \in \cup_{i \in N} S_i^*$.

Να δείξετε ότι η ανάθεση αγαθών $\mathcal{S}^* = (S_1^*, \ldots, S_n^*)$ μιας Walrasian ισορροπίας αντιστοιχεί σε μια βέλτιστη λύση του (LP) (για τη λύση του (LP), θέτουμε $x_{i,S} = 1$ για $S = S_i^*$, και $x_{i,S} = 0$ διαφορετικά). Άρα η ανάθεση αγαθών $\mathcal{S}^*$ μιας Walrasian ισορροπίας μεγιστοποιεί την κοινωνική ωφέλεια.

(δ) Με χρήση των complementary slackness conditions (και του (γ)), να δείξετε ότι ένα στιγμιότυπο του προβλήματος μεγιστοποίησης της κοινωνικής ωφέλειας έχει Walrasian ισορροπία αν και μόνο αν το (LP) έχει ακέραια βέλτιστη λύση (όπου $x_{i,S} \in \{0, 1\}$ για κάθε $i \in N$ και $S \subseteq U$).

(ε) Θεωρούμε στιγμιότυπα με 2 αγαθά και 2 παίκτες. Να δώσετε παράδειγμα συναρτήσεων αποτίμησης v_1 και v_2 για τις οποίες υπάρχει Walrasian ισορροπία και παράδειγμα συναρτήσεων αποτίμησης v_1' και v_2' για τις οποίες δεν υπάρχει Walrasian ισορροπία.