3η Σειρά Γραπτών Ασκήσεων

Σωτήρου Θεόδωρος 03118209

March 12, 2023

Άσκηση 1: Υπολογισιμότητα

Σε αυτην την άσκηση μελετάμε το Πρόβλημα Διοφαντικών Εξισώσεων. Στο πρώτο ερώτημα περιγράφουμε εναν αλγόριθμο που ημιαποφασίζει το πρόβλημα αυτό και στο δεύτερο ερώτημα περιγράφουμε μια αναγωγή απο το πρόβλημα αυτο στο Πρόβλημα Τερματισμού.

α) Σε αυτό το ερώτημα περιγράφουμε έναν αλγόριθμο που ημιαποφασίζει το Πρόβλημα Διοφαντικών Εξισώσεων.

1. Το στιγμιότυπο (είσοδος) του αλγορίθμου είναι μια πολυωνυμική εξίσωση $\alpha_1 X_1 + \alpha_2 X_2 + ... + \alpha_n X_n = d$ με ακαίρεους συντελετές α_i και d και πεπερασμένο αριθμό αγνώστων x_i , $\forall i \in 1, ..., n$.

2. Μετέτρεψε την πολυωνυμική εξίσωση στην μοργή $Ax^T = d$ οπου A πίνακας συντελεστών της μορφής $[\alpha_1\alpha_2\alpha_3...\alpha_n]$ και x πίνακας μεταβλητών της μορφής $[x_1x_2...x_n]$.

3. Για κάθε άγνωστη μεταβλητή διάλεξε ενα εύρος τιμών.

4. Για κάθε συνδιασμό τιμών μεσα στο εύρος, αντικατέστησε τις τιμές στον πίνακα και δες αν ικανοποιούνται οι περιορισμοί.

5. Αν κάποιος συνδιασμός βρίσκει λύση σταμάτα και επέστρεψε την λύση.

6. Επανέλαβε τα 4 και 5 έως ότου βρείς λύση

7. Αν δεν βρείς λύση επέστρεψε οτι η εξίσωση δεν έχει λύση.

β) Στο ερώτημα αυτό ανάγουμε το Πρόβλημα Διοφαντικών Εξισώσεων στο Πρόβλημα Τερματισμού. Μπορούμε να μετατρέψουμε μια μηχανή Turing σε μια διοφαντική εξίσωση. Αυτό είναι δυνατόν μεσα απο το θεώρημα του Matiyasevich. Η μηχανή τερματίζει ανν βρεθεί λύση στην εξίσωση.

Άσκηση 2: Πολυπλοκότητα - Αναγωγές

Οι αποδείξεις των ατνίστοιχων προτάσεων φαίνονται παρακάτω:

(α) Για να δείξουμε οτ το πρόβλημα Π' είναι co C-πλήρες ως προς την R πρέπει να δείξουμε δύο πράγματα. 1. $\Pi' \in$coC και 1.$\forall \pi' \in$coC, $\pi' \leq_R \Pi'$.Προφανώς το πρώτο ισχύει αφού το Π' είναι το συμπληρωματικό του Π άρα ανήκει και στην κλάση coC. Έστω $\pi \in$NP, το π μπορεί να αναχθεί στο Π μέσω της R, δηλαδή $x \in \pi$ ανν $R(x) \in L$. Το συμπληρωματικό του π μπορεί να αναχθεί μέσω της R στο Π' αφού $x \notin \pi$ ανν $R(x) \in \Pi'$.

(β) Το πρόβλημα Tautology είναι το συμπληρωματικό του SAT που είναι NP-πλήρες άρα σύμφωνα με το παραπάνω θεώρημα και για C=NP το Tautology είναι coNP-Πλήρες.

(γ) Υποθέτουμε το πρόβλημα Π της κλάσης coNP είναι και NP-πλήρες. Τότε $\forall \Pi' \in$NP, $\Pi' \leq_R \Pi$ και αφου $\Pi \in coNP$ τοτε και $\Pi' \in$coNP. Επιπλέον για κάθε πρόβλημα $\Pi'' \in$conNP το να βρούμε μια λύση YES μπορεί να αναχθεί στο Π που είναι NP-complete και επομένως το ερώτημα αν υπάρχει NO μπορεί να αναχθεί στο συμπληρωματικό του Π που είναι NP.

(δ) Το NAE3SAT είναι NP πρόβλημα επειδή μπορούμε να επιβεβαιώσουμε οτι μια ανάθεη τιμών αληθείας ικανοποιεί την συνθήκη μας. Για να δείξουμε οτι το NAE3SAT είναι NP-complete θα κάνουμε 2 αναγωγές. Πρώτα θα ανάγουμε το 3SAT στο NAE4SAT και έπειτα το NAE4SAT στο NAE3SAT δηλαδή $3SAT \leq_R NAE4SAT \leq_R NAE3SAT$. Για κάθε clause του προβλήματος 3SAT δημιουργούμε το αντίστοιχο clause για το NAE4SAT με ένα παραπάνω literal v. Θ.δ.ο. η πρόταση φ σε 3CNF είναι ικανοποιήσιμη ανν αντίστοιχη πρόταση ψ σε 4CNF μετά την προσθήκη του literal ικανοποιεί το NAE4SAT.

Έστω μια ανάθεση που ικανοποιεί την φ τότε θέτοντας v = False ικανοποιείται και η ψ αφού έχει τουλάχιστον ένα και το πολύ τρία Literals True για κάθε clause. Έστω μια ανάθεση που ικανοποιεί την ψ τότε αν η μεταβλητή s είναι False σε κάθε clause, η φ προκύπτει απο τις αναθέσεις των υπόλοιπων μεταβλητών ενώ αν η μεταβλητή s είναι True τότε η φ προκύπτει απο τις συμπληρωματικές αναθέσεις των υπόλοιπων μεταβλητών.

Μπορούμε να ανάγουμε το NAE4SAT στο NAE3SAT με όμοιο τρόπο της αναγωγής του SAT στο 3SAT. Συγκεκριμένα αντικαθιστούμε κάθε όρο $c_j = (l_{j1}, l_{j2}, l_{j3}, l_{j4})$ με 2 όρους $c'_{1j} = (l_{j1}, l_{j2}, w_i)$ και $c'_{2j} = (\bar{w_i}, l_{j3}, l_{j4})$. Εύκολα προκύπτει οτι c_j σε NAE ανν c'_{1j} και c'_{2j} σε NAE.

(ε) Το πρόβλημα της Επιλογής Αντιπροσώπων είναι NP αφού αν μας δώσουν το σύνολο $R \subseteq U$ μπορούμε να επιβεβαιώσουμε σε πολυωνυμικό χρόνο οτι εκπροσωπούνται οι διάφορες κοινωνικές ομάδες. Για να δείξουμε οτι είναι NP-πλήρες θα ανάγουμε το Set Cover σε αυτό δηλαδή Set Cover$\leq_R$ Επιλογή Αντιπροσώπων. Έχουμε το σύνολο $S = 1, 2, ..., r$, τα υποσύνολα $X_1, ..., X_m$ του S και τον φυσικό αριθμό κ, $1 < k < m$. Η Επιλογή Αντιπροσώπων προκύπτει οταν κάθε στοιχείο $e \in S$ υποδικνύει μια κοινωνική ομάδα (το index της i) και τα υποσύνολα $X_j, j < m$ αναπαριστούν το υποσύνολο των κοινωνικών ομάδων που ανήκει το άτομο j. Η αναγωγή μπορεί να γίνει σε πολυωνυμικό χρόνο αφου μπορούμε να βρούμε σε πολυωνυμικό χρόνο τις κοινωνικές ομάδες που ανήκει κάθε άτομο j. Είναι εύκολο να δείξουμε οτι υπάρχουν $\leq$ k υποσύνολα των οποίων η ένωσή είναι το S ανν υπάρχει $R \subseteq U$ με $|R| \leq$ k, τέτοιο ώστε $\forall i \leq r, U_i \cap R \neq \emptyset$.

Άσκηση 3: Προσεγγιστικοί Αλγόριθμοι - Vertex Cover

Σε αυτήν την άσκηση μελετάμε εναν προσεγγιστικό αλγόριθμο για το Πρόβλημα του καλύματος κόμβων. Το παράδειγμα που ζητείται βρίσκεται στην εικόνα 2.

1. Το σύνολο κόμβων C που κατασκευάζει ο αλγόριθμος αποτελεί κάλυμμα κόμβων (και επομένως καλύπτει όλες τις ακμές) διότι ο βρόχος συνεχίζει να εκτελείται μέχρις ότου κάθε ακμή του συνόλου E καλυφθεί απο κάποιον κόμβο του C.

2. Το άθροισμα $\sum_{e \in E} c(e)$ αποτελείται απο μηδενικούς και μη μηδενικούς όρους. Οι όροι αυτοί είναι είτε το μικρότερο απο τα βάρη των μόμβων μιας ακμής που επιλέχθηκε είτε το βάρους κάποιου κόμβου μείον το βάρος κάποιου μικρότερου που επιλέχθηκε ήδη. Το άθροισμα αυτών των όρων αποτελεί υποσύνολο των όρων που βρίσκονται στο άθροισμα των βαρών (π.χ. $w(C) = w_1 + w_2 + w_3 + w_4$ και $c(C) = w_1 + w_3$).Το υποσύνολο αυτό έχει μέγεθος το πολύ $|C|$ και τουλάχιστον $\frac{|C|}{2}$. Στις περιπτώσεις που το υποσύνολο αυτό έχει μήκος $< |C|$, τα βάρη που απαλοίφονται είναι μικρότερα απο κάποιο άλλο μέσα στο υποσύνολο. Άρα το άθροισμα των όρων του υποσυνόλου πολλαπλασιασμένο επι δύο θα είναι πάντα μεγαλύτερο απο σο συνολιγό βάρος της λύσης. Ενα παράδειγμα δίνεται στην εικόνα 1.

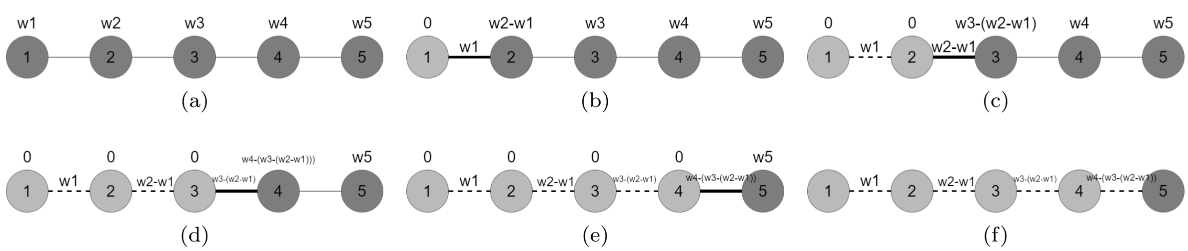

Figure 1: Η λειτουργία του WeightedVertexCover. (a) Το γράφημα εισόδου G, με 5 κόμβους με βάρη $w_1 < w_2 < ... < w_5$. (b) Ο αλγόριθμος επιλέγει (τυχαία) την ακάλυπτη ακμή 1-2 και θέτει τιμή c(1-2) = w_1 ενώ ανανεώνει την τιμή t(2) σε $w_2 - w_1$ και βάζει τον κόμβο 1 στο κάλυμα. Όμοια στα (c), (d), (e) και (f) ο αλγόριθμος επιλέγει τυχαία και ανανεώνει τις τιμές των C, t καθώς και το κάλυμα. Τελικά $\sum_{e \in E} c(e) = w_2 + w_4$ ενώ $w(C) = w_1 + w_2 + w_3 + w_4$ δηλαδή εμφανίζονται μονο δύο βάρη απο τα 4. Ισχύει όμως οτι $w_1 < w_2$ και $w_3 < w_4$ γιαυτό $w_1 + w_2 + w_3 + w_4 < 2w_2 + 2w_4$.

3. Το ερώτημα αυτό προκύπτει εύκολα αν σκεφτούμε οτι το πλήθος των ακμών με μη μηδενική τιμή c(e) ισούται με το πλήθος των κόμβων στο κάλυμα. Έπειτα απο την επιλογή μιας ακμής απο τον αλγόριθμο απο τους δύο κόμβους που εννώνει η ακμή αυτός με την μικρότερη τιμή t(u) επιλέγεται για

το κάλυμα ενώ ο άλλος ανανεώνει την τιμή του t(v) = t(v) - t(u). Αρχικά οι τιμές t όλων των κόμβων ισούνται με τα βάρη τους. Άρα το άθροισμα $\sum_{e \in E} c(e)$ γίνεται μέγιστο και ίσο με το συνολικό βάρος όταν επιλέγονται ακμές που δεν έχουν κοινά άκρα. Με άλλα λόγια μετά απο κάθε επανάληψη η συνάρτηση στον πίνακα c(e) εισάγεται μια καινούρια τιμή x το πολύ ίση με το βάρος του κόμβου που επιλέχθηκε για το κάλυμα δηλαδή $x \leq w(v)$.

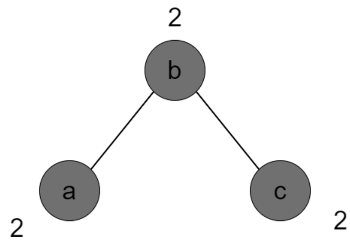

Figure 2: Στο παράδειγμα αυτό ο αλγόριθμος WeightedVertexCover υπολογίζει Vertex Cover με συνολικό βάρος διπλάσιο απο αυτό της βέλτιστης λύσης. Διαλέγει 2 απο τις 3 κορυφές ενώ η βέλτιστη λύση περιέχει μόνο την κορυφή b.

Άσκηση 4: Πιθανοτικοί Άλγόριθμοι - Έλεγχος Ταξινόμησης

Ένας πίνακας είναι σχεδόν ταξινομημένος αν η διαγραφή το πολύ n/4 απο τα στοιχεία του μας δίνει εναν ταξινομημένο πίνακα.

(α) Ο αλγόριθμος αυτός χρειάζεται να ελέγξει Ω(n) θέσεις ώστε η πιθανότητα λάθους να γίνει μικρότερη του 0.1 στην περίπτωση που ο πίνακας είναι της μορφής $A = [1, 3, 5, 7, ..., n-1, 2, 4, ..., n]$ ο οποίος δεν είναι σχεδόν ταξινομημένος. Ο αλγόριθμος δίνει λανθασμένη απάντηση αν διαλέξει αποιοδήποτε εκτός απο τα δύο μεσαία στοιχεία δηλαδή P(λάθους) = $(\frac{n-2}{n})^{\kappa}$.

(β) Εύκολα μπορούμε να δούμε οτι αν $k < l$ τότε υπάρχει κάποιο στοιχείο w του πίνακα για το οποίο να ισχύει $x < A[w] < y$, δηλαδή το w θα έιναι κάποιο μέσο mid στην διάρκεια της αναζήτησης ώστε το x να είναι μικρότερο και το y μεγαλύτερο ώστε να ισχύει $k < l$.

(γ) Δεδομένου οτι έχω n/4 + 1 στοιείά λάθος ταξινομημένα αναζητώ την πιθανότητα να απαντήσει Ναι ο αλγόριθμος στο αν είναι σχεδόν ταξινομημένος ο πίνακας. Απο το (β) προκύπτει οτι P(Λάθος) $\leq (78)^k$ και για κάποια τιμή του κ αυτή η πιθανότητα είναι πάντα μικρότερη του 0.1.