ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

Σχολή Ηλεκτρολόγων Μηχανικών & Μηχανικών Υπολογιστών

Αλγόριθμοι και Πολυπλοκότητα 7^{Ο} ΕΞΑΜΗΝΟ

3η Σειρά Γραπτών Ασκήσεων

Μανιάτης Ανδρέας el18070

maniatis.andreas@gmail.com

Άσκηση 1: Υπολογισιμότητα

α. Από τον ορισμό της αποκρίσιμης γλώσσας εκφράζουμε τον εξής αλγόριθμο που ημιαποφασίζει το Πρόβλημα Διοφαντικών Εξισώσεων:

Για δεδομένη είσοδο ψάξε συστηματικά όλους τους ακέραιους συνδυασμούς μεταβλητών έτσι ώστε:

- αν ο συνδυασμός είναι επιτυχής ο αλγόριθμος να τερματίζει με απάντηση ΝΑΙ και
- αν ο συνδυασμός δεν είναι επιτυχής ο αλγόριθμος είτε τερματίζει με έξοδο ΟΧΙ είτε συνεχίζει.

β. Στη συνέχεια περιγράφουμε αναγωγή από το Πρόβλημα Διοφαντικών Εξισώσεων στο Πρόβλημα Τερματισμού.

Μπορούμε να κατασκευάσουμε μια μηχανή Turing, η οποία προσομοιώνει τη διαδικασία αναζήτησης λύσεων στον εκθετικό χώρο λύσεων. Διαβάζοντας την είσοδο, αν αυτή ικανοποιεί το Πρόβλημα Διοφαντικών εξισώσεων τερματίζει και αν όχι παράγει την επόμενη πιθανή λύση την οποία στη συνέχεια δοκιμάζει κ.ο.κ.

Το Πρόβλημα Τερματισμού θα τερματίσει αν και μόνο αν ο εκάστοτε συνδυασμός που θα δοκιμάσει είναι σωστός, διαφορετικά συνεχίζει να εκτελείται.

Έχουμε τελικά ότι η λύση του Προβλήματος Διοφαντικών εξισώσεων θα δώσει θετική έξοδο στο Πρόβλημα Τερματισμού και επίσης η λύση που κατασκευάζει η μηχανή Turing για το Πρόβλημα Τερματισμού και έτσι ολοκληρώνεται η αναγωγή.

Άσκηση 2: Πολυπλοκότητα – Αναγωγές

α. Αρχικά αφού το Π είναι C-πλήρες έχουμε ότι

$$\Pi \in C \leftrightarrow \overline{\Pi} \in coC$$

Π C-hard άρα

$$\forall \Pi' \in C, \Pi' \leq_R \Pi$$

Θέλουμε να αποδείξουμε το ίδιο για κάθε $\overline{\Pi} \in coC$.

Έχουμε:

(3-5)

$$x \in \Pi' \leftrightarrow R(x) \in \Pi$$

$$x \neg\in \Pi' \leftrightarrow R(x) \neg\in \Pi$$

$$x \in \overline{\Pi'} \leftrightarrow R(x) \in \overline{\Pi}$$

Επομένως αποδείξαμε ότι Π’ είναι co-C-πλήρες ως προς την αναγωγή R.

β. Έστω ντετερμινιστική μηχανή Turing M και πολυώνυμα p και q.
Για κάθε x και y η M τρέχει για p(x) χρόνο στην είσοδο (x,y).

$$\forall x \in \{0,1\}^*,$$

$$x \in L_{tautology} \leftrightarrow \forall y \in \{0,1\}^{q(x)} M(x,y) = 1$$

Επομένως, $Tautology \in coNP$.

Στη συνέχεια θα αποδείξουμε ότι το πρόβλημα είναι $coNP - \pi\lambda\acute{\eta}\rho\varepsilon\varsigma$, δείχνοντας ότι

$$\forall\ \Pi \in coNP \leftrightarrow \Pi \leq_R Tautology.$$

Με την ακολουθία (3-5) του ερωτήματος αποδεικνύουμε ότι για κάθε x που ανήκει σε γλώσσα του co-NP υπάρχει πολυωνυμική συνάρτηση R έτσι ώστε το R(x) να ανήκει στην $L_{tautology}$.

γ. Ισχύει από τον ορισμό της κλάσης co-NP ότι το συμπλήρωμα κάθε προβλήματος που ανήκει στην κλάση NP ανήκει στην κλάση co-NP. Έστω ένα NP-πλήρες πρόβλημα που ανήκει στην κλάση co-NP. Θα δείξουμε ότι τότε προκύπτει ότι NP = co-NP.

Από την υπόθεση και τον ορισμό της πληρότητας προκύπτει ότι υπάρχει μη-ντετερμινιστική μηχανή Turing, η οποία επαληθεύει σε πολυωνυμικό χρόνο μέσω πολυωνυμικής αναγωγής το συμπλήρωμα κάθε προβλήματος που ανήκει στο NP.

Επομένως:

$$NP \subseteq coNP \tag{0}$$

Παίρνοντας τα συμπληρώματα και στις δύο πλευρές έχουμε:

$$\overline{NP} \subseteq \overline{coNP} \tag{1}$$

$$coNP \subseteq NP \tag{2}$$

Από (0) και (2) προκύπτει ότι:

$$coNP = NP$$

δ. Θα ακολουθήσουμε την εξής σειρά αναγωγών:

$$3SAT \leq_R NAE4SAT \leq_R NAE3SAT$$

Και τα 3 προβλήματα ανήκουν στην κλάση NP αφού μπορούμε να επαληθεύσουμε τη λύση τους σε πολυωνυμικό χρόνο.

Από κάθε clause

$$c = \bigvee l_i$$

κατασκευάζουμε το

$$c' = c \vee s$$

όπου s κοινό για όλα τα clauses.

Θα δείξουμε ότι:

$$x \in 3SAT \leftrightarrow R(x) \in NAE4SAT$$

όπου R η πολυωνυμική αναγωγή που περιγράψαμε παραπάνω.

Για λύση x που ικανοποιεί το 3SAT με έστω ένα clause με 3 true-literals θέτουμε s=0 και άρα R(x) θα ικανοποιεί και το NAE4SAT.

Για λύση που ικανοποιεί το NAE4SAT με s=0 σημαίνει ότι η αντίστοιχη θα ικανοποιεί και το πρόβλημα 3SAT (θα υπάρχει τουλάχιστον ένα true-literal σε κάθε clause).

Για λύση που ικανοποιεί το NAE4SAT με s=1 είναι πιθανό να υπάρχουν clauses, όπου όλα τα άλλα literals είναι false και επομένως το αντίστοιχο στιγμιότυπο στο πρόβλημα 3SAT δεν θα ικανοποιείται. Για αυτόν το λόγο και επειδή στο πρόβλημα NAE4SAT υπάρχει τουλάχιστον ένα true και ένα false literal σε κάθε clause μπορούμε να αντιστρέψουμε την τιμή κάθε μεταβλητής ώστε να έχουμε satisfiability και για το πρόβλημα 3SAT.

Επομένως, το πρόβλημα NAE4SAT είναι NP-πλήρες.

Στη συνέχεια μπορούμε πολύ εύκολα να πάμε με μια επιπλέον αναγωγή στο πρόβλημα NAE3SAT. Συγκεκριμένα, (για την αναγωγή)μετατρέπουμε κάθε

$$c = l1 \lor l2 \lor l3 \lor l4$$

σε

$$c' = (l1 \lor l2 \lor u) \land (\, l3 \lor l4 \lor \neg u) = L1 \land L2$$

Αν μόνο ένα literal του c έχει true value τότε θέτουμε u=0 αν l1=1 ή l2=1 και u=1 αν l3=1 ή l4=1. Έτσι τα c και c' θα έχουν τον ίδιο πίνακα τιμών αληθείας.

Επομένως, και το NAE3SAT είναι NP-πλήρες.

ε. Θα δείξουμε ότι το πρόβλημα αντιπροσώπων είναι NP-πλήρες ανάγοντας το πρόβλημα του Minimum Dominating Set σε αυτό.

Αρχικά το πρόβλημα των αντιπροσώπων ανήκει στο NP, αφού μπορούμε να αποτιμήσουμε σε πολυωνυμικό χρόνο αν μια λύση περιέχει αντιπροσώπους από όλες τις ομάδες.

Κατασκευάζουμε γράφο G με κορυφές V και ακμές E σε πολυωνυμικό χρόνο, έτσι ώστε

$$(u, v) \in E \; ανν \; u, v \in U_i, για \; i = 1, \dots, r$$

Έστω λύση του MDS, δηλαδή το ελάχιστο σύνολο κόμβων που η ένωσή τους με τους γείτονές τους μας δίνει το σύνολο U και επίσης μικρότερο ή ίσο από k. Αυτή θα αποτελεί λύση και για το πρόβλημα των αντιπροσώπων αφού όλα τα άτομα είτε θα είναι αντιπρόσωποι είτε θα έχουν ένα άτομο από την ομάδα τους για αντιπρόσωπο. Επίσης, δεν γίνεται να έχουμε 2 αντιπρόσωπους από την ίδια ομάδα αφού αυτοί θα συνδέονταν και έτσι δεν είχε το MDS. Παρομοίως, η λύση του προβλήματος των αντιπροσώπων θα αποτελούσε και λύση του MDS.

Άσκηση 3: Προσεγγιστικοί Αλγόριθμοι – Vertex Cover

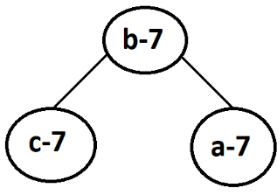
1. Το C καλύπτει όλες τις ακμές αφού το while loop εκτελείται όσο το C δεν είναι vertex cover, δηλαδή από τον ορισμό του δεν καλύπτονται όλες οι ακμές.

2. Παρατηρούμε ότι τη στιγμή που προσθέτουμε έναν κόμβο i στο σύνολο C ισχύει:

$$w_i = \sum_{e=(i,j)} c(e)$$

Επομένως, για όλο το C θα ισχύει:

$$w(C) = \sum_{i \in C} \sum_{e=(i,j)} c(e)$$



δηλαδή ένα άθροισμα ακμών πάνω σε κάθε κόμβο του vertex cover.

Επειδή κάθε ακμή μπορεί να συμπεριληφθεί στο άθροισμα το πολύ δύο φορές (μία για κάθε κόμβο) έχουμε:

$$w(C) \leq 2 \sum_{e \in E} c(e)$$

3.

Έστω το παρακάτω γράφημα, στο οποίο μέσα σε κάθε κόμβο αναγράφεται το όνομά του και το βάρος του και για το οποίο το Optimal Weighted Cover είναι το {b} με w(Opt) = 7.

Εφαρμόζουμε τώρα τον αλγόριθμο της εκφώνησης:

Αρχικοποίηση

$$C \leftarrow \{\,\}$$

κτλ.

Επιλέγουμε την ακάλυπτη ακμή {a,b}

$$\delta \leftarrow 7$$

$$t(a) \leftarrow 0, t(b) \leftarrow 0$$

$$c(\{a, b\}) \leftarrow 7$$

και

$$C \leftarrow C \cup \{\, z \in \{a, b\} \mid t(z) = 0 \,\}$$

$$C \leftarrow \{a, b\}$$

όπου ο αλγόριθμος τερματίζει.

Το Vertex Cover που επιστρέφει ο αλγόριθμος όμως είναι:

$$w(Sol) = 7 + 7 = 14 = 2 * w(Opt)$$

δηλαδή με συνολικό βάρος διπλάσιο από αυτό της βέλτιστης λύσης.

Άσκηση 4: Πιθανοτικοί Αλγόριθμοι – Έλεγχος Ταξινόμησης

α. Έστω ο εξής <u>όχι σχεδόν ταξινομημένος</u> πίνακας:

1 2 4 3 13 12 11 10 9 8 7 6 5 1/11 < 1/10 πιθανότητα λάθους

ο οποίος μπορεί να συνεχίζει με περίοδο 11 για n συνολικά στοιχεία. Επομένως κάτι λιγότερο από 10% των n στοιχείων θα είναι θετικά στον έλεγχο και τα υπόλοιπα αρνητικά. Λόγω της περιοδικής φύσης των τιμών θα πρέπει να ελέγξουμε πάνω από n/10 στοιχεία, άρα Ω(n).

β. Για binary search αν $x \neq y$ μετά από ένα ή περισσότερα βήματα θα υπάρξει τιμή mid για την οποία ισχύει:

$$y < mid < x \text{ ή } x < mid < y$$

[1]

και η οποία θα κατευθύνει την αναζήτηση του μικρότερου στοιχείου στις μικρότερες τιμές μειώνοντας την τιμή up της αναδρομικής κλήσης και το άλλο στις μεγαλύτερες αυξάνοντας την τιμή του up.

Στα επόμενα βήματα ακόμα και αν ο πίνακας δεν είναι ταξινομημένος, το binary search του μικρότερου στοιχείου δεν μπορεί να πάει σε τιμές μεγαλύτερες από το mid και του μεγαλύτερου σε μικρότερες λόγω της ιδιότητας της δυαδικής αναζήτησης.

Οπότε αν $k < l \leftrightarrow x < y$.

[1] Κάποια ανισότητα θα μπορούσε να είναι και ίση χωρίς βλάβη της γενικότητας