ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΡΙΤΗ ΣΕΙΡΑ ΑΣΚΗΣΕΩΝ
ΑΛΓΟΡΙΘΜΟΙ ΚΑΙ ΠΟΛΥΠΛΟΚΟΤΗΤΑ

Αναστασία Χριστίνα Λίβα
03119029

Περιεχόμενα

Άσκηση 1: Υπολογισιμότητα

Ερώτημα Πρώτο:

Στόχος: Θα αποδείξω ότι υπάρχει μηχανή Turing η οποία:

- σταματά με απάντηση «ναι» εάν υπάρχει ακέραια λύση
- τρέχει ατέρμονα εάν δεν υπάρχει ακέραια λύση

Αποδεικνύοντας αυτό έχω ουσιαστικά δείξει ότι το πρόβλημα ανήκει στην τάξη RE.

Δημιουργώ μηχανή Turing η οποία θα δέχεται τη διοφαντική εξίσωση ως input. Δημιουργώ επίσης όλες τις πιθανές πλειάδες ακεραίων που ενδέχεται να λύνουν την εξίσωση, οπότε τις έχω καταγεγραμμένες με δομημένο τρόπο. Για κάθε τούπλα που παράγεται αντικαθιστώ τους αριθμούς της εξίσωσης και υπολογίζω το αποτέλεσμα χρησιμοποιώντας την παρούσα πλειάδα. Εάν δε λάβω μηδενικό αποτέλεσμα επαναλαμβάνω τη διαδικασία χρησιμοποιώντας την επόμενη τούπλα. Σε αντίθετη περίπτωση η εξίσωση ικανοποιείται οπότε η μηχανή τερματίζει με απάντηση «ναι», γεγονός το οποίο υποδηλώνει την ύπαρξη ακέραιας λύσης. Αν δεν υπάρχει ακέραια λύση η μηχανή θα τρέχει ατέρμονα παράγοντας και ελέγχοντας συνεχώς καινούριες τούπλες.

Η μηχανή Turing τερματίζει δίνοντας θετική έξοδο σε περίπτωση εύρεσης ακέραιας λύσης που πληροί τα κριτήρια σύμφωνα με τα οποία το πρόβλημα ταξινομείται στην κλάση RE και ουσιαστικά υποδεικνύει την ύπαρξη αλγορίθμου που έχει τη δυνατότητα να ημιαποφασίζει το πρόβλημα απαριθμώντας όλες τις θετικές περιπτώσεις, δηλαδή όσες τερματίζουν με απάντηση «ναι». Σε περίπτωση μη ύπαρξης λύσης η μηχανή έχει συμπεριφορά όμοια του ορισμού RE, όπου το πρόβλημα είναι ημιαποφασίσιμο.

Ερώτημα Δεύτερο:

Πρόβλημα Τερματισμού: In computability theory, the halting problem is the problem of determining, from a description of an arbitrary computer program and an input, whether the program will finish running, or continue to run forever. The halting problem is undecidable, meaning that no general algorithm exists that solves the halting problem for all possible program–input pairs. [2]

Έστω δεδομένη μηχανή Turing M η οποία δέχεται είσοδο w. Σχεδιάζω δεύτερη μηχανή Turing H η οποία προσομοιώνει τη λειτουργία της μηχανής M, δηλαδή εάν η M τερματίζσει με είσοδο w τερματίζει και η H με output «ναι», ενώ σε περίπτωση μη τερματισμού με είσοδο w η H ακολουθεί την πορεία της M ατέρμονα. Συνεπώς προκύπτει ότι HP $\in$ RE.

Το πρόβλημα των διοφαντικών εξισώσεων είναι γνωστό πρόβλημα RE, συνεπώς θα το ανάξω στο HP με σκοπό να δείξω ότι για να είναι ένα πρόβλημα RE-hard θα πρέπει κάθε πρόβλημα $\in$ RE να μπορεί να αναχθεί στο HP. Φτιάχνω μηχανή Turing D που απαριθμεί όλες τις τούπλες ακεραίων και ελέγχει αν μια δεδομένη διοφαντική εξίσωση ικανοποιείται από αυτές. Το πρόβλημα ανάγεται στο HP όμως μένει να δω αν η μηχανή D τερματίζει σε ειδική είσοδο αντιπροσωπευτική της διαδικασίας απαρίθμησης. Φτιάχνω λοιπόν μια νέα μηχανή Turing D' η οποία έχει σχεδιαστεί έτσι ώστε για δεδομένη διοφαντική εξίσωση να σταματά σε πρίπτωση που βρει τούπλα που την ικανοποιεί ή να τρέχει ατέρμονα σε περίπτωση που δε βρει κατάλληλη τούπλα, συνεπώς το αν η D' τερματίζει είναι ισοδύναμο ερώτημα με αυτό της ύπαρξης λύσης της διοφαντικής εξίσωσης σε πρόβλημα τερματισμού της D'.

Βάσει των παραπάνω προκύπτει ότι το HP είναι RE-complete.

Ερώτημα Τρίτο:

Πρόβλημα Καθολικού Τερματισμού: Σταματάει συγκεκριμένη μηχανή Turing M για κάθε πιθανή είσοδο;

Για να θεωρείται ένα πρόβλημα RE-hard, πρέπει να είναι τουλάχιστον τόσο δύσκολο όσο τα πιο δύσκολα προβλήματα του RE. Αυτό σημαίνει ότι κάθε πρόβλημα του RE μπορεί να μετασχηματιστεί σε αυτό.

Λαμβάνω το πρόβλημα του Περιορισμένου Προβλήματος Τερματισμού, το οποίο περιλαμβάνει μια μηχανή Turing M και μια είσοδο w. Δημιουργώ μια νέα μηχανή Turing M' που προσομοιώνει την M στο w για οποιαδήποτε είσοδο x. Αν η M σταματήσει στο w, τότε η M' σταματάει σε κάθε είσοδο x. Σε διαφορετική περίπτωση, η M' δεν σταματάει σε καμία είσοδο x.

Η επίλυση του καθολικού προβλήματος τερματισμού για την M' δίνει την απάντηση στο εάν η M θα σταματήσει στο w. Εάν η M' έχει καθοριστεί να σταματάει για όλες τις εισόδους, προκύπτει ότι η M θα σταματήσει στο w. Έτσι, εάν μπορέσω να επιλύσω το καρολικό πρόβλημα τερματισμού, θα μπορέσω να επιλύσω και το HP, καθιστώντας το καθολικό πρόβλημα τερματισμού RE-hard.

Άσκηση 2: Πολυπλοκότητα - Αναγωγές

Ερώτημα Πρώτο:

UnSat: The decision problem of determining if a given Boolean formula does not have any satisfying assignment. [1]

NoLargeClique: δίνεται γράφος G και ακέραιος k, και ζητείται αν ισχύει ότι κάθε κλίκα του G έχει μέγεθος το πολύ k

Κατασκευάζω γράφο βάσει του UnSat στον οποίο κάθε node αντιστοιχεί σε μια μεταβλητή και συνδέω κάθε node με κάποιο άλλο εάν και τα δυο nodes μπορούν να έχουν συγχρόνως αληθείς μεταβλητές. Το k ισούται με τον αριθμό των ρητρών πλην ένα διότι κάθε κλίκα που έχει μέγεθος μεγαλύτερο από αυτό θα αντιστοιχούσε σε σύνολο τιμών που ενδεχομένως ικανοποιεί τον τύπο, γεγονός το οποίο αντιτίθεται στη συνθήκη UnSat.

Εφόσον η UnSat είναι αληθής δεν υφίσταται σύνολο τιμών τέτοιο ώστε να ικανοποιείται το σύνολο των προτάσεων και συνεπώς προκύπτει ότι δε μπορεί να υπάρξει κλίκα μεγαλύτερη απο k αφού αυτο θα συνεπαγόταν καλή ανάθεση των λογικών.

Ο γράφος και το σύνολο k υπάγονται στο NoLargeClique Problem. Αν έχω αληθή απάντηση στο UnSat θα έχω και στο NoLargeClique.

Ερώτημα Δεύτερο:

Έστω $P \in C$ και $P' \in coC$. Αφού Π είναι C-πλήρες τότε $P \leq_R \Pi$ και αν Σ το αλφάβητο $\forall x \in \Sigma^* x \in P \Leftrightarrow R(x) \in \Pi$ από όπου και προκύπτει ότι

$$\forall x \in \Sigma^* x \notin P \Leftrightarrow R(x) \in \Pi$$

Αλλά: $x \notin P \Leftrightarrow x \in P'$ και $R(x) \notin \Pi \Leftrightarrow R(x) \in \Pi', \ \forall x \in \Sigma^*$

Δηλαδή έχω $\forall x \in \Sigma^* x \in P' \Leftrightarrow R(x) \in \Pi'$ που συνεπάγεται ότι:

$$P' \leq_R \Pi'$$

Αφού από πριν το P επιλεχθεί αυθαίρετα αυτό ισχύει $\forall P' \in coC$, και επομένως το Π' είναι coC-δύσκολο. Επιπλέον αφού το Π είναι C-πλήρες $\pi in C$ και $\Pi' \in coC$, οπότε και το Π' είναι coC-πλήρες ως προς την $\leq_R$

Ερώτημα Τρίτο:

Η ύπαρξη ενός προβλήματος NP-πλήρους προβλήματος που ανήκει στο NP και το coNP υποδηλώνει ότι υπάρχει πρόβλημα που είναι εξίσου δύσκολο με τα πιο δύσκολα προβλήματα στο NP, καθώς είναι NP-πλήρες.

Συνεπώς, για αυτό το πρόβλημα, και ως εκ τούτου για όλα τα προβλήματα που μπορούν να αναχθούν σε αυτό, οι περιπτώσεις «όχι» μπορούν να επαληθευθούν όπως και οι περιπτώσεις «ναι» σε πολυωνυμικό χρόνο. Εάν είναι δυνατό να επαληθευτούν τόσο οι θετικές όσο και οι αρνητικές περιπτώσεις σε πολυωνυμικό χρόνο για ένα NP-πλήρες πρόβλημα, τότε η διάκριση μεταξύ NP και coNP παύει να υφίσταται.

Ως αποτέλεσμα, η ύπαρξη ενός τέτοιου προβλήματος συνεπάγεται την ισοδυναμία των δύο κλάσεων.

Ερώτημα Τέταρτο:

3SAT problem: the problem of determining if there exists an interpretation that satisfies a given Boolean formula. In other words, it asks whether the variables of a given Boolean formula can be consistently replaced by the values TRUE or FALSE in such a way that the formula evaluates to TRUE. If this is the case, the formula is called satisfiable. [3]

NAE3SAT: An NP-complete variant of the Boolean satisfiability problem, often used in proofs of NP-completeness. [4]

Θα δείξω ότι το 3SAT αποτελεί ειδική περίπτωση του NAE4SAT και θα το ανάξω στο NAE3SAT.

Έστω c ενα clause του 3SAT: $c = l_1 \vee l_2 \vee l_3$ και c' ενα clause του NAE4SAT: $c' = l_1 \vee l_2 \vee l_3 \vee l_4$. Θέτοντας $l_4 = 0$ στο c' έχω $c' = l_1 \vee l_2 \vee l_3 \vee 0 = l_1 \vee l_2 \vee l_3$ και αφού l_4 δεν ικανοποιείται η ικανοποιησιμότητα του c' είναι ίδια με του c, αφού δεν έχω περιορισμό να μην είναι όλα ίσα. Άρα το NAE4SAT είναι γενίκευση του 3SAT.

Για να το ανάξω στο NAE3SAT:

$$c_4 = (l_1 \vee l_2 \vee l_3 \vee l_4) \rightarrow (l_1 \vee l_2 \vee z) \wedge (\neg z \vee l_3 \vee l_4 = c_3$$

c_4 ικανοποιήσιμο ανν c_3 ικανοποιήσιμο.

Επιπλέον $l_1 = l_2 = l_3 = l_4 = 1$ τότε $z = 1$ ή $z = 0$ ένα από τα δύο θα έχει all 3 equal άρα απορρίπτεται.

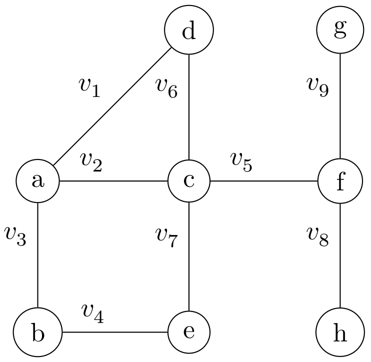
Αλλιώς αν $l_1 = l_2 = 1$ τότε δεν γίνεται το z να ισούται με τη μονάδα διότι δε θα είναι δεκτό στο NAE3SAT. Η ανάθεση $z = 0$ είναι δεκτή.

Οπότε η απάντηση με αυτή την αναγωγή είναι ακριβώς ίδια.

Άρα $3SAT \leq_p NAE4SAT \leq_p NAE3SAT$ οπότε NAE3SAT NP-complete.

Ερώτημα Πέμπτο:

Θα δείξω ότι το πρόβλημα επιλογής αντιπροσώπων είναι γενίκευση του vertex cover. Συγκεκριμένα, αν κάθε ομάδα έχει δύο ακριβώς άτομα τότε θεωρώ κάθε άτομο node και κάθε ομάδα ακμή.



Έχοντας επιλογή κόμβων πλήθους $\leq k$ που καλύπτουν όλες τις ακμές οδηγούμαι στο υπαρξιακό του vertex cover, το οποίο γίνεται σε πολυωνυμικό χρόνο καθώς αποτελεί ειδική περίπτωση.

Συνεπώς $VC \leq_P$ Επιλογή αντιπροσώπων, οπότε η επιλογή αντιπροσώπων είναι NP-complete.

Άσκηση 3: Προσεγγιστικοί Αλγόριθμοι - Vertex Cover -TSP

Ερώτημα Πρώτο:

1. Λόγω του while loop ο μοναδικός τρόπος να τερματίσει ο αλγόριθμος είναι το c να αποτελεί vertex cover, δηλαδή να καλύπτει όλες τις ακμές. Τερματίζει σίγουρα διότι σε κάθε επανάληψη στο c προστίθεται τουλάχιστον ένα node, οπότε, εφόσον δεν τερματίσει νωρίτερα, είναι βέβαιο πως κάποτε θα συμπεριλαμβάνονται όλοι οι κόμβοι του γράφου, οπότε θα αποτελεί αναγκαστικά vertex cover.

2. Θα χρησιμοποιήσω επαγωγή για την απόδειξη του ζητούμενου. Από την αρχικοποίηση έχω:

$$C \leftarrow \text{ άρα } w(C) = 0$$

$$\forall e \in E, c(e) \leftarrow 0 \text{ άρα } 2\sum_{e \in E} c(e) = 0$$

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

Η επαγωγική υπόθεση είναι ότι πριν εισάγει νέο node και αντίστοιχο c(e) $w(c) \leq 2\sum_{e \in E} c(e)$

Είνια η πρώτη φορά που εξετάζονται οι κόμβοι $u, v(t(u) = w(u), t(v) = w(v)$ και έχω τρεις περιπτώσεις:

1. Στο c μπαίνουν και οι δυο κόμβοι της e διότι $t(u) = t(v) = w(u) = w(v)$ οπότε έχω

$$w(c') = w(c) + w(u) + w(v) \leq \sum_{e \in E} c(e) + 2w(v) \leq \sum_{e \in E} c(e) + 2c(e') \leq \sum_{e \in E'} c(e)$$

   Επίσης συνεχίζει να ισχύει ότι οι επόμενοι κόμβοι θα συναντηθούν για πρώτη φορά και γι' αυτούς θα ισχύει $w(u) = t(w)$.

2. Στο C μπαίνει ένας από τους δύο κόμβους, ενώ ο άλλος δε μπαίνει σε καμία επανάληψη. Έστω οτι $\delta = min\{t(u), t(v)\} = t(u)$.

$$t(u) = w(u) \text{διότι δεν έχω ξανασυναντήσει τον κόμβο } u$$

   Τότε $w(c') = w(c) + w(u) \leq \sum_{e \in E} c(e) + \delta \leq \sum_{e \in E} c(e) + 2\delta \leq \sum_{e \in E} c(e) + c(e') \leq \sum_{e \in E'} c(e)$

   Επίσης για κάθε επόμενο κόμβο που μπαίνει στο C $t(u) = w(u)$ αφού μόνο ο r άλλαξε και $t(v) \neq w(v)$ ο οποίος δε θα μπει στο C οπότε διατηρείται η ιδιότητα.

3. στο C μπαίνει ένας από τους δύο κόμβους και ο επόμενος θα μπει σε επόμενη επανάληψη. Θα χρειαστεί να πραγματοποιήσω πολλά βήματα για να διατηρήσω την επαγωγική υπόθεση.

   Έστω χβτγ ότι πρώτα εισάγεται ο u στο C κι έπειτα ο v συγκρίνει κάποια ακολουθία nodes $u_1, u_2, ...u_n$ όπου λόγω ΕΥ $\forall i t(u_i) = w(u_i)$ όπου $\forall i < n t(u_i) < t(v)$, ενώ $t(v) \leq t(u_n)$, δηλαδή στο εσωτερικό της ακολουθίας παρεμβάλονται άσχετοι κόμβοι που συγκρίνονται μεταξύ τους και θα τους αντιμετωπίσω στη συνέχεια με επαγωγή. Τώρα θα βάλω όλη την ακολουθία στο C και μετά αφού έχω διατηρήσει την ΕΥ αυτοί θα αντιστοιχιστούν σε μια από τις τρεις περιπτώσεις και θα πράξουν ανάλογα.

   Θέλω νδο:

$$(I) \quad w(u) + w(u_1) + w(u_2) + ... + w(u_{n-1}) \leq 2t(u) + 2t(u_1)_2 t(u_2) + ...+$$

$$2t(u_{n-1}) + 2(t(v) - t(u_m) - t(u_1) - t(u_2) - ... - t(u_{n-1}) \leq 2t(v) \leq 2w(v)$$

Το οποίο ισχύει γιατί έχω

$$\forall i < nt(u_i) \leq t_i(v) - = w(v) - t(u) - t(u_1) - t(u_2) - ... - t(u_{i-1} =$$
$$w(v) - w(u) - w(u_1) - w(u_2) - .. - w(u_{i-1})$$

Άρα

$$w(u_i) \leq w(v) - w(u) - w(u_1) - w(u_2) - ... - w(u_{i-1} \Leftrightarrow w(u) + w(u_1) + w(u_2) + ... + w(u_i) \leq w(v)$$

Έστω $\delta = n - 1$ και $w(u) + w(u_1) + .. + w(u_{n-1} \leq w(v) \leq 2w(v)$

Άρα η (I) ισχύει και αφού είναι όλα θετικά προσθέτω κατά μέλη την ΕΥ οπότε $W(c') \leq \sum_{e \in E'} c(e)$

Για το u_n αν $t(v) = t(u_m)$ τροποποιώ το παραπάνω επιχείρημα ώστε αντί για u_{n-1} να έχω u_n. Αλλιώς $t(v) < t(u_m)$ οπότε είτε το u_m δε θα μπει σε κάποια επόμενη επανάληψη στο C οπότε διατηρείται ότι $w(u') = t(u')$ για τα επόμενα οπότε έχω και το ζητούμενο είτε το u_n θα μπει οπότε εφαρμόζω αναδρομικά την 3η περίπτωση

3. Η βέλτιστη λύση βάρους w^* προκύπτει εάν στον αλγόριθμο επιλεχθούν μη ντετερμινιστικά οι ακμές που δίνουν τους κόμους που ανήκουν στο min weight vertex cover. Έτσ το αποτέλεσμα του αλγορίθμου θα είνια $c(e) = 0$ για τα nodes που δεν επιλέχθηκαν και $c(e) = \delta = t(u^*) \leq w(u^*)$ γι αυτές που θα επιλεχθούν αφού προκύπτει ότι $\sum_{e \in E} c(e) \leq \sum_{u^* \in V^*} w(u^*) = w^*$
$t(u) \leq w(u)$ από τον αλγόριθμο κι επιπλέον υπάρχει περίπτωση να μπουν ο u και ο v στο c ενώ θα χρεωεθεί μια ακμή e το βάρος του min εκ των δυο οπότε το $\sum c(e)$ θα είναι ακόμη μιρκότερο.

Από τα παραπάνω προκύπτει ότι ο αλγόριθμος είναι 2-προσεγγιστικός καθώς για το c που υπολογίζει $w(c) \leq 2 \sum_{e \in E} c(e) \leq 2w^*$

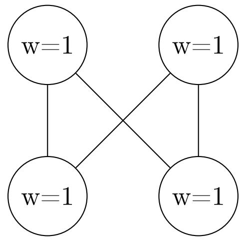
Επιπλεόν το 2 είναι tight αφού δεν υπάρχει οικογένεια γράφων όπου βρίσκεται c με $w(c) = 2w^*$

Συγκεκριμένα στα διμερή γραφήματα K_n, n όπου $w(u) = 1 \forall u$ που είναι κόμβος του $K_{n,n}$

Ο αλγόριθμος σε κάθε επανάληψη επιλέγει μια ακμή της οποίας οι δύο κόμβοι είναι ακόμη στο C. Η ακμή μεταξύ των 2 κόμβων δε μπορεί να καλυφθεί από κάποιο κόμβο εκτός αν έίναι ένας από τους δυο και ο αλγόριθμος θα βάλει και τους δυο στο C αφού έχουν ίδιο βάρος. Ο γρφος είναι διμερής οπότε σε κάθε βήμα θα επιλέγει και θα τοποθετεί στο C έναν κόμβο από τη μια και έναν από την άλλη εώς ότου καλυφθεί όλος ο γράφος χρησιμοποιώντας όλους τους κόμβους, οπότε $w(c) = 2n$

Όμως κάθε $K_{n,n}$ με $w(u) = 1 \forall u$ κόμβο μπορεί να καλυφθεί με $w^* = n$ επιλέγοντας ,όνο τη μια πλευρά που έχει n nodes και καλύπτει όλες τις ακμές.

Ο πιο απλός γράφος όπου $w(c) = 2w^*$ είναι στο $K_{2,2}$ με $w(w) = 1 \forall u$



Οποιαδήποτε από τις u ακμές και να επιλέξει ο αλγόριθμος στο πρώτο βήμα θα καλύψει 3 ακμές αφου κάθε κόμβος καλύπτει δυο και εδω η επιλεγμένη είναι κοινή βάζοντας και τους δυο κόμβους στο C αφού έχουν ίδιο βάρος. Αναγκαστικα στη συνέχεια θα επιλέγει την τέταρτη ακμή και βάζει τους άλλους 2 κόμβους οπότε $w(c) = 4$

Σημείωση: Η βέλτιστη λύση επιλέγει τους 2 αριστερά κόμβους και καλύπτοντας τους καλύπτει και τις 4 e με $w^* = 2$.

Ερώτημα Δεύτερο:

κύκλος Hamilton: Έχοντας ένα γράφημα G(V, E), το ερώτημα είναι να καθορίσω εάν υπάρχει κύκλος που επισκέπτεται κάθε κορυφή ακριβώς μία φορά και επιστρέφει στην αρχική κορυφή.

Περιοδεύων Πωλητής: Σε έναν πλήρη γράφο G'(V', E'), όπου κάθε ακμή έχει ένα συγκεκριμένο βάρος, η πρόκληση είναι να εντοπίσω τον συντομότερο δυνατό δρόμο που επισκέπτεται κάθε κορυφή ακριβώς μία φορά και επιστρέφει στην αρχική κορυφή.

Η συνήθης αναγωγή από το πρόβλημα του κύκλου Hamilton στο πρόβλημα του περιοδεύοντος πωλητή περιλαμβάνει τη δημιουργία ενός προβλήματος περιοδεύοντος πωλητή από έναν δεδομένο γράφημα G για το πρόβλημα του κύκλου Hamilton. Για κάθε ακμή του G που περιλαμβάνεται στο σύνολο E, ανατίθεται ένα μοναδικό βάρος, ενώ σε κάθε ακμή που δεν περιλαμβάνεται στο σύνολο E (ώστε το G' να γίνει πλήρες), ανατίθεται ένα απαγορευτικά υψηλό βάρος.

Για να αποδείξω ότι δεν υπάρχει αλγόριθμος προσέγγισης του προβλήματος περιοδεύοντος πωλητή με πολυωνυμικό χρόνο εκτέλεσης, εκτός αν ισχύει P=NP, προσαρμόζω τα βάρη των ακμών στην αναγωγή ως εξής:

- Εάν υπάρχει ακμή στο G, ορίζω το βάρος της ως 1.
- Διαφορετικά, αντί να τη θέσω σε απαγορευτικά υψηλή τιμή, την ορίζω με βάρος ελαφρώς υψηλότερο από k φορές τον αριθμό των κορυφών στο G, δηλαδή $k \cdot |V| + 1$.

Αν το G περιέχει έναν κύκλο Hamilton, το βάρος της βέλτιστης διαδρομής στο πρόβλημα περιοδεύοντος πωλητή στο G' είναι ακριβώς |V|, διότι ο κύκλος θα περιλαμβάνει μόνο ακμές με βάρος 1. Αντίθετα, εάν δεν υπάρχει κύκλος Hamilton στο G, οποιαδήποτε διαδρομή στο G' πρέπει να περιλαμβάνει τουλάχιστον μία ακμή που δεν υπάρχει στο G, καθιστώντας το βάρος της τουλάχιστον $k \cdot |V| + 1$.

Ωστόσο, μια λύση που προσεγγίζει το βέλτιστο βάρος με παράγοντα όχι μεγαλύτερο από $k \cdot |V|$ προκύπτει μόνον εάν υπάρχει ένας κύκλος Hamilton στο G. Σε περίπτωση που δεν υπάρχει κύκλος Hamilton, το βάρος οποιασδήποτε λύσης που προσεγγίζει το βέλτιστο με παράγοντα k πρέπει να είναι σημαντικά υψηλότερο από το προϋπολογισμένο$k \cdot |V|$

Εάν υπάρχει ένας αλγόριθμος προσέγγισης πολυωνυμικού χρόνου για το πρόβλημα του περιοδεύοντος πωλητή, θα μπορούσούσα να τον χρησιμοποιήσω για να καθορίσω εάν ένας δεδομένος γράφος G διαθέτει έναν κύκλο Hamilton εξετάζοντας το βάρος της προσεγγιστικής λύσης του προβλήματος περιοδεύοντος πωλητή. Αυτό, ωστόσο, συνεπάγεται ότι P=NP, το οποίο είναι άτοπο.

Αναφορές

[1] Ning Luo κ.ά. *Proving UNSAT in Zero Knowledge*. Accessed on February 12, 2024. 2022. URL: https://eprint.iacr.org/2022/206.pdf.

[2] Wikipedia. *Halting Problem*. Accessed on February 12, 2024. URL: https://en.wikipedia.org/wiki/Halting_problem.

[3] wikipedia. *Boolean satisfiability problem*. Accessed on February 12, 2024. URL: https://en.wikipedia.org/wiki/Boolean_satisfiability_problem.

[4] wikipedia. *Boolean satisfiability problem*. Accessed on February 12, 2024. URL: https://en.wikipedia.org/wiki/Not-all-equal_3-satisfiability.