Τμήμα Μηχανικών Η/Υ και Πληροφορικής

Επιστήμη και Τεχνολογία των Υπολογιστών

ΘΕΩΡΙΑ ΑΠΟΤΕΛΕΣΜΑΤΙΚΩΝ ΠΡΟΣΕΓΓΙΣΤΙΚΩΝ ΑΛΓΟΡΙΘΜΩΝ

Ασκήσεις 2005

ΛΙΒΑΘΙΝΟΣ ΝΙΚΟΛΑΟΣ

libathin@ceid.upatras.gr

Α.Μ.: 403

Έστω G ένα πλήρες μη κατευθυνόμενο γράφημα, όπου τα κόστη σε κάθε πλευρά είναι είτε 1 είτε 2. Δώστε έναν 4/3-προσεγγιστικό αλγόριθμο για το πρόβλημα TSP για αυτή την ειδική κατηγορία γραφημάτων.

Αρχικά παρατηρούμε ότι η λύση του TSP προβλήματος αποτελεί ταυτόχρονα λύση του minimum 2-matching προβλήματος M[1], οπότε η προσέγγιση του minimum 2-matching αποτελεί κάτω όριο στην προσέγγιση του TSP. Στη συνέχεια θα δώσουμε έναν αλγόριθμο που θα κοστίζει το πολύ cost(M)+n/3 που είναι το πολύ τα 4/3 του κόστους του βέλτιστου TSP.

1. Βρες ένα min 2-matching M για το γράφημα G.
2. Εντόπισε τους κύκλους C1,C2,…Ck που δημιούργησε το M.
3. Από κάθε κύκλο Ci διάλεξε δύο διαδοχικούς κόμβους ui και ui′. Διέγραψε την ακμή (ui,ui′) και πρόσθεσε την ακμή (ui′,ui+1) όπου η ui+1 αντιστοιχεί στον κύκλο Ci+1. Επανέλαβε την διαδικασία για i=1,…,k κυκλικά (δηλαδή τον k-οστό κύκλο ένωσε τον με τον πρώτο).
4. Δώσε στην έξοδο τον τελικό κύκλο που έχει δημιουργηθεί από την συνένωση των κύκλων στο τρίτο βήμα.

Προφανώς με τον αλγόριθμο αυτό έχουμε πετύχει να δημιουργήσουμε έναν κύκλο. Στη συνέχεια θέλουμε να δούμε πόσο θα είναι το κόστος που θα πετύχουμε με αυτόν τον αλγόριθμο. Αρχικά παρατηρούμε ότι εφόσον κάθε κόμβος στο M έχει βαθμό ακριβώς 2, θα πρέπει οπωσδήποτε να δημιουργούνται κύκλοι. Όπως αναφέρει και το δεύτερο βήμα του αλγορίθμου έστω ότι έχουν δημιουργηθεί k κύκλοι (ξένοι μεταξύ τους). Κάθε κύκλος έχει το ελάχιστο 3 κόμβους, οπότε και παίρνουμε ένα άνω φράγμα για το k, $k \le n/3$.

Στη συνέχεια το τρίτο βήμα του αλγορίθμου αναλαμβάνει απλώς να συνενώσει τους ανεξάρτητους κύκλους επιλέγοντας σε κάθε κύκλο ένα σημείο (την ακμή (u_i,u_i')) που ο κύκλος «σπάει» και ενώνεται με έναν άλλο κύκλο. Είναι ουσιαστικά σαν να ανοίγουμε τους κύκλους μετατρέποντάς τους σε τόξα και κατόπιν ενώνουμε τα τόξα σε έναν ενιαίο κύκλο. Για αυτή τη διαδικασία αφαιρούμε k ακμές και προσθέτουμε k ακμές. Εφόσον η κάθε ακμή έχει βάρος στο {1,2}, αφαιρούμε το ελάχιστο k και προσθέτουμε το μέγιστο 2k. Άρα τελικά στην χειρότερη περίπτωση το βήμα 3 προσθέτει βάρος k στο TSP.

Αθροίζοντας το βάρος του πρώτου και του τρίτου βήματος έχουμε ότι το κόστος του αλγορίθμου μας A είναι cost(A)≤cost(M)+k. Όμως όπως είδαμε ισχύει $k \le n/3$, άρα cost(A)≤cost(M)+n/3. Όπως είπαμε στην αρχή της απόδειξης cost(M) ≤OPT. Επίσης εφόσον το ελάχιστο κόστος των ακμών είναι 1, κατά συνέπεια OPT≥n$\Rightarrow \frac{n}{3} \le \frac{OPT}{3}$. Άρα καταλήγουμε στο ζητούμενο: $\cos t(A) \le \cos t(M) + n/3 \le OPT + \frac{1}{3}OPT = \frac{4}{3}OPT$.

---

[1] Ο σκοπός του minimum 2-matching προβλήματος είναι η εύρεση ενός υποσυνόλου των ακμών ενός γραφήματος με την ιδιότητα κάθε κορυφή του γραφήματος να αποτελεί την άκρη σε ακριβώς δύο ακμές του matching, ελαχιστοποιώντας παράλληλα το συνολικό κόστος των ακμών του matching. Το minimum 2-matching πρόβλημα μπορεί να λυθεί σε πολυωνυμικό χρόνο.

Δείξτε ότι ο παρακάτω αλγόριθμος είναι ένας 2-προσεγγιστικός αλγόριθμος για το πρόβλημα MAX-SAT: Έστω τ μια αυθαίρετη ανάθεση αλήθειας και τ΄ η συμπληρωματική της, δηλαδή μια μεταβλητή είναι αληθής στην τ αν και μόνο αν είναι ψευδής στην τ΄. Υπολογίστε το βάρος των προτάσεων που ικανοποιούνται στην τ και στην τ΄ και δώσε ως έξοδο την καλύτερη ανάθεση από τις δύο.

Το αντικείμενο του MAX-SAT προβλήματος είναι δεδομένης μιας Boolean έκφρασης που έχει αναπαρασταθεί ως γινόμενο αθροισμάτων, προσπάθησε να βρεις την ανάθεση των μεταβλητών που μεγιστοποιεί τα αθροίσματα που αληθεύουν. Εφόσον οι προτάσεις αποτελούνται από αθροίσματα των μεταβλητών, δηλαδή συνδέονται μεταξύ τους με την λογική πράξη OR, αρκεί να είναι αληθής έστω μια από τις μεταβλητές ώστε να αληθεύει όλη η πρόταση.

Αν λοιπόν έχουμε μια τυχαία ανάθεση τ και την αντίστοιχη συμπληρωματική της, τότε σίγουρα μια οποιαδήποτε παράσταση C_i θα ικανοποιείται είτε από την τ είτε από την τ' είτε και από τις δύο. Εφόσον κάθε φορά επιλέγουμε την καλύτερη από τις δύο, τότε η ιδανικότερη περίπτωση είναι η μια ανάθεση να ικανοποιεί όλες τις παραστάσεις και η άλλη καμία. Αντίστοιχα η χειρότερη περίπτωση είναι να ικανοποιούν και οι δύο ακριβώς τις μισές παραστάσεις. Επομένως ακόμα και στη χειρότερη περίπτωση θα έχουμε έναν 2 προσεγγιστικό αλγόριθμο.

Θεωρήστε τον παρακάτω αλγόριθμο για το πρόβλημα SET COVER.

1. Βρες μια βέλτιστη λύση στη χαλαρωμένη έκδοση του προβλήματος ως γραμμικό πρόβλημα.

2. Διάλεξε όλα τα σύνολα $s \in S$ για τα οποία $x_s > 0$ στην κλασματική λύση.

Δείξτε ότι ο αλγόριθμος αυτός πετυχαίνει προσέγγιση f, όπου f η συχνότητα του πιο πολυεμφανιζόμενου στοιχείου.

Προκειμένου να λύσουμε το πρόβλημα χρησιμοποιούμε τις primary complementary slackness conditions για τη βέλτιστη κλασματική λύση του γραμμικού προγράμματος, όπως ορίζονται στην ακόλουθη σχέση:

$$\forall S \in \$: \text{ή } x_S = 0 \text{ ή } \sum_{e:e\in S} y_e = c(S) \qquad (1)$$

Οπότε άμεσα προκύπτει πως:

$$\forall x_S \neq 0 \Rightarrow \sum_{e:e\in S} y_e = c(S) \qquad (2)$$

Εφόσον ο αλγόριθμος της εκφώνησης επιλέγει στην ακέραια λύση όλα τα x_s που είναι μεγαλύτερα από μηδέν, θα ισχύει η σχέση (2).

Αν ονομάσουμε I το σύνολο που αποτελείται από τα S_i που επιλέγει ο αλγόριθμος, τότε το συνολικό του κόστος προκύπτει από την ακόλουθη σχέση:

$$C(A) = \sum_{s\in I} c(S) \qquad (3)$$

$$(2), (3) \Rightarrow C(A) = \sum_{s\in I}\sum_{e:e\in S} y_e = \sum_{e:e\in S} y_e \left|\{S \in I : e \in S\}\right| \qquad (4)$$

Όμως εξ' ορισμού ισχύει ότι $\left|\{S \in I : e \in S\}\right| \leq$f, άρα καταλήγουμε στην ακόλουθη σχέση:

C(A)$\leq f \sum_{e:e\in S} y_e \leq f \cdot OPT$ (5)

Η τελευταία σχέση αποδεικνύει ότι ο αλγόριθμος είναι όντως ένας f-προσεγγιστικός αλγόριθμος.

Δείξτε ότι η dual fitting ανάλυση του άπληστου αλγορίθμου για το πρόβλημα SET COVER δίνει προσέγγιση H_k, όπου k είναι το μέγεθος του μεγαλύτερου συνόλου και $H_k = \sum_{i=1}^{k} \frac{1}{i}$.

Αρχικά θα παραθέσουμε τον άπληστο αλγόριθμο που χρησιμοποιείται για το SET COVER.

```
C = 0
while C ≠ U do
    Βρες το πιο αποδοτικό σύνολο σε αυτή την αναδρομή, έστω S.
    Θέσε α= c(S)/|S − C| ως την αποδοτικότητα του συνόλου S.
    Επέλεξε το S, και για κάθε e ∈ S − C , θέσε price(e)=α².
    C = C ∪ S
Δώσε στην έξοδο τα σύνολα που επιλέχθηκαν.
```

Επίσης παραθέτουμε την primal και dual μορφή του SET COVER ως LP.

Minimize: $\sum_{S\in\$} c(S)x_S$

Δεδομένου ότι: $\sum_{S:e\in S} x_S \geq 1, \quad e \in U$

$x_S \geq 0, S \in \$$

Και η αντίστοιχη dual μορφή:

Maximize: $\sum_{e\in U} y_e$

Δεδομένου ότι: $\sum_{e:e\in S} y_e \leq c(S), \quad S \in \$$

$y_e \geq 0, \quad e \in U$

[2] Στην πραγματικότητα αυτό το βήμα δεν είναι απαραίτητο για τη λειτουργία του αλγορίθμου αλλά προστίθεται για να μας διευκολύνει στην ανάλυσή του.

Αρχικά θέλουμε να δείξουμε ότι αν θέσουμε $y_e = \frac{price(e)}{H_k}$, τότε προκύπτει μια εφικτή λύση για το dual.

Έστω ότι ένα σύνολο S αποτελείται από k στοιχεία και έστω ότι αριθμούμε τα στοιχεία του με τη σειρά που τα επιλέγει ο άπληστος αλγόριθμος $e_1, e_2, ... e_k$. Ας υποθέσουμε ότι βρισκόμαστε στην επανάληψη που ο αλγόριθμος επιλέγει το στοιχείο e_i. Αυτό σημαίνει ότι το S περιέχει τουλάχιστον k-i+1 μη επιλεγμένα στοιχεία (δεν γνωρίζουμε αν θα καλυφθούν όλα όσα έχει). Επομένως σε αυτή την επανάληψη η αποδοτικότητα του S είναι το πολύ $\frac{c(S)}{k-i+1}$ και κατά συνέπεια θα ισχύει ότι

$$y_{e_i} \leq \frac{1}{H_k} \frac{c(S)}{k-i+1} \qquad (1)$$

Στη συνέχεια για να αποδείξουμε ότι τα y_e αποτελούν όντως λύση στο dual αρκεί να δείξουμε ότι:

$$\sum_{e:e \in S} y_e \leq c(S), \quad S \in \$ \qquad (2)$$

$$\sum_{e:e \in S} y_e \leq \frac{c(S)}{H_k} \left(\frac{1}{k} + \frac{1}{k-1} + \frac{1}{k-2} + ... \frac{1}{1} \right) \leq c(S) \frac{H_k}{H_k} = c(S) \qquad (3)$$

Οπότε όντως αποτελεί λύση για το dual.

Τέλος θα δείξουμε ότι ο άπληστος αλγόριθμος επιτυγχάνει συντελεστή προσέγγισης H_k. Έτσι υπολογίζουμε το κόστος για όλα τα στοιχεία του σύμπαντος: $\sum_{e \in U} price(e) = H_k \left(\sum_{e \in U} y_e \right) \leq H_k$ OPT.

Οπότε και προκύπτει το ζητούμενο.

Έστω G=(V,E) ένα διμερές (bipartite) γράφημα.

(α). Δείξτε ότι η ακόλουθη LP relaxation είναι μια ακριβής LP relaxation για το πρόβλημα Maximum Matching στο G (δείξτε δηλαδή ότι έχει πάντα μια βέλτιστη ακέραια λύση):

Maximize : $\sum_{e \in E} x_e$

Subject to: $\sum_{e: e incident to u} x_e \leq 1, \; u \in V$

$$x_e \geq 0, \; e \in E$$

(β) Δώστε το δυικό (dual) του παραπάνω γραμμικού προγράμματος και δείξτε ότι είναι μια ακριβής LP relaxation για το πρόβλημα εύρεσης ελαχίστου vertex cover στο διμερές γράφημα G.

(γ) Χρησιμοποιείστε τα παραπάνω για να δείξετε ότι για κάθε διμερές γράφημα,

$$\max_{matching M} |M| = \min_{vertex cover U} |U|$$

(α). Αρχικά θέλουμε να δείξουμε ότι η λύση στο LP relaxed πρόβλημα είναι ακέραια, οπότε και βέλτιστη. Προκειμένου να καταλάβουμε καλύτερα τι ακριβώς σημαίνει ότι η χαλαρωμένη LP λύση είναι ακριβής, ας φανταστούμε το αντίστοιχο γεωμετρικό πρόβλημα. Το ζητούμενο είναι η ταυτόχρονη ικανοποίηση ενός συνόλου από γραμμικές ανισότητες, δηλαδή ο προσδιορισμός μιας κλειστής επιφάνειας ή ενός πολυγώνου (δηλαδή ενός υπόχωρου) μέσα στον χώρο που ορίζονται οι ανισότητες. Αφού βρούμε αυτή τις λύσεις (πολύγωνο) θέλουμε κατόπιν να προσδιορίσουμε τη μέγιστη λύση του συστήματος των ανισώσεων, δηλαδή το πιο ψηλό σημείο του πολυγώνου. Το ότι αναμένουμε η λύση μας να είναι ακριβής σημαίνει ότι οι κορυφές του πολυγώνου έχουν ακέραιες συντεταγμένες. Επομένως αρκεί να δείξουμε ότι το συγκεκριμένο LP πρόβλημα περιγράφεται από ένα πολύγωνο με ακέραιες κορυφές.

Προκειμένου να αποδείξουμε το ζητούμενο βασιζόμαστε σε μια αλγεβρική μοντελοποίηση του προβλήματος. Αρχικά ορίζουμε έναν πίνακα A ως totally unimodular (TUM) αν η ορίζουσα κάθε τετραγωνικού υποπίνακά του έχει τιμές στο {-1,0,1}. Στο maximal matching πρόβλημα ο πίνακας των περιορισμών A του χαλαρωμένου LP προβλήματος είναι ο πίνακας γειτνίασης του γραφήματος. Εξ' ορισμού ο πίνακας γειτνίασης ενός μη κατευθυνόμενου γραφήματος με n κορυφές και m ακμές είναι ο nxm πίνακας όπου το στοιχείο (i,j) είναι 1 αν η κορυφή i πρόσκειται στην ακμή j και 0 διαφορετικά. Αν το γράφημα είναι κατευθυνόμενο τότε αν η ακμή εξέρχεται από τον κόμβο θέτουμε την προσημασμένη τιμή +1 και αν εισέρχεται την τιμή -1. Όλα αυτά είναι σημαντικά διότι για τους TUM πίνακες ισχύει το ακόλουθο θεώρημα που πρόκειται να αποδείξουμε:

> Όλες οι κορυφές του πολύεδρου $P_1 = \{x \mid Ax \le b\}$ είναι ακέραια διανύσματα αν ο A είναι TUM και το b είναι ένα ακέραιο διάνυσμα.

Προκειμένου να αποδείξουμε τον παραπάνω ισχυρισμό παρατηρούμε ότι οι κορυφές του P_1 θα είναι επίσης λύσεις των εξισώσεων $A'x = b'$, όπου A' είναι ένας υποπίνακας του A και b' το αντίστοιχο υποδιάνυσμα του b. Σύμφωνα με τον κανόνα του Cramer γνωρίζουμε πως οι λύσεις x_i του $A'x = b'$ δίνονται από την ακόλουθη σχέση:

$$x_i = \frac{\det(M_i)}{\det(A')}$$

Όπου M_i είναι ο πίνακας που προκύπτει με αντικατάσταση της i-οστής στήλης του A' από το διάνυσμα b. Εφόσον ο πίνακας A αποτελείται από τις τιμές {-1,0,1} και εφόσον το διάνυσμα b έχει επίσης ακέραιες τιμές, άρα και ο πίνακας M_i θα έχει ακέραιες τιμές, άρα και η ορίζουσά του θα είναι ακέραια. Επίσης εφόσον ο A είναι TUM ο A' ως υποπίνακάς του θα έχει ορίζουσα που θα λαβαίνει τιμές στο {-1,1} (η περίπτωση να είναι 0 αποκλείεται καθώς ο A είναι μη ιδιόμορφος). Επομένως κάθε x_i θα είναι ακέραιο.

Έχοντας αποδείξει όλα αυτά αρκεί να δείξουμε ότι αν ο πίνακας γειτνίασης του διμερούς γράφου είναι TUM τότε η LP relaxed λύση θα είναι ακριβής. Πρώτα θα αποδείξουμε ότι ο πίνακας γειτνίασης ενός οποιουδήποτε κατευθυνόμενου γράφου είναι TUM. Για να το δείξουμε αυτό θα χρησιμοποιήσουμε επαγωγή αρχίζουντας με τον 1x1 υποπίνακα και γενικεύοντας.

Αρχικά παρατηρούμε ότι κάθε στήλη του πίνακα θα έχει το πολύ δύο μη μηδενικά στοιχεία και αυτά θα είναι τα -1, +1. Αυτό είναι προφανές εφόσον κάθε στήλη αντιστοιχεί σε μια ακμή και κάθε ακμή θα αντιστοιχεί σε μια εισερχόμενη και μια εξερχόμενη κορυφή (δηλαδή γραμμή). Άρα ο 1x1 υποπίνακας θα είναι κάποιο από τα στοιχεία {-1,0,1} και κατ' επέκταση η ορίζουσά του (δηλαδή το ίδιο) θα είναι {-1,0,1}. Αν θεωρήσουμε ότι το ίδιο θα ισχύει και για τον kxk υποπίνακα, θα αποδείξουμε ότι θα ισχύει και για τον (k+1)x(k+1) υποπίνακα C. Συγκεκριμένα υπάρχουν οι ακόλουθες 3 περιπτώσεις:

1. Ο C έχει μια μηδενική στήλη, άρα det(C)=0.

2. Ο C έχει μια στήλη με ακριβώς ένα μη μηδενικό στοιχείο. Αναπτύσσοντας την ορίζουσα του C με βάση αυτή την στήλη τότε αυτή θα είναι $\pm 1 \det(C_k)$, όπου C_k είναι ο kxk υποπίνακας. Ωστόσο από το δεύτερο βήμα της αναδρομής γνωρίζουμε ότι $\det(C_k) \in \{-1,0,1\}$. Άρα και $\det(C\ ) \in \{-1,0,1\}$.

3. Ο C έχει σε όλες του τις στήλες τα δύο μη μηδενικά στοιχεία -1, +1. Σε αυτή την περιπτωση μπορούμε εκτελώντας γραμμοπράξεις στην ορίζουσα του C να καταλήξουμε στην μηδενική ορίζουσα (πολύ εύκολα φαίνεται ότι το άθροισμα των γραμμών του C σε αυτή την περίπτωση είναι το μηδενικό διάνσυμα). Επομένως det(C)=0.

Άρα και στις 3 περιπτώσεις ισχύει ότι $\det(C\ ) \in \{-1,0,1\}$ και έτσι αποδείξαμε ότι σε κάθε κατευθυνόμενο γράφημα ο πίνακας γειτνίασης θα είναι TUM.

Στη συνέχεια θα ολοκληρώσουμε το πρώτο μέρος της άσκησης δείχνοντας πως μπορούμε να μετασχηματίσουμε ένα διμερές γράφημα σε κατευθυνόμενο, οπότε και ο αντίστοιχος πίνακας γειτνίασης θα είναι TUM. Για να το επιτύχουμε αυτό αρκεί να θέσουμε κατευθύνσεις στις ακμές του διγραφήματος από την μια κατεύθυνση προς την άλλη (π.χ. από τα αριστερά προς τα δεξιά). Με αυτή την αλλαγή θα αλλάξουν τα πρόσημα κάποιων άσσων στον πίνακα γειτνίασης του (μη κατευθυνόμενου) διγραφήματος και από 1 θα γίνουν -1, κάτι που προφανώς επηρεάζει μόνο το πρόσημο της ορίζουσας του πίνακα του διγραφήματος ή οποιουδήποτε υποπίνακά του (ουσιαστικά δεν θα υπάρχει καν η τρίτη περίπτωση στην απόδειξη της αναδρομής που είδαμε παραπάνω). Επομένως ο πίνακας γειτνίασης ενός διγραφήματος θα είναι και αυτός TUM.

Τελικά συγκεντρώνοντας όλα τα παραπάνω αποτελέσματα καταλήγουμε ότι εφόσον το διγράφημα έχει TUM πίνακα γειτνίασης και εφόσον το διάνυσμα των σταθερών b στους περιορισμούς του LP προβλήματος είναι ακέραιο θα έχουμε και ακέραιες λύσεις x_i, άρα η χαλαρωμένη λύση του LP θα είναι ακριβής.

(β). Αρχικά παραθέτουμε το dual του γραμμικού προγράμματος της εκφώνησης:

$$\text{Minimize: } \sum_{u \in V} y_u$$

$$\text{Subject to: } \sum_{u:u incident to e} y_u \geq 1,\ \ e \in E$$

$$y_v \geq 0\ ,\ u \in V$$

Το ζητούμενο στο set cover πρόβλημα είναι η εύρεση του μικρότερου υποσυνόλου των κορυφών ενός μη κατευθυνόμενου γραφήματος με την ιδιότητα κάθε ακμή του γραφήματος να έχει τουλάχιστον την μια της άκρη σε κόμβο του συνόλου αυτού. Αν χρησιμοποιήσουμε πάλι μια μεταβλητή $x_i \in \{0,1\}$ για να διαλέξουμε τους κόμβους που θα ανήκουν στο cover τότε έχουμε το ακόλουθο γραμμικό πρόγραμμα:

$$\text{Minimize: } \sum_{u \in V} x_u$$

$$\text{Subject to: } \sum_{u:u incident to e} x_u \geq 1,\ \ e \in E$$

$$y_v \geq 0\ ,\ u \in V$$

Παρατηρούμε πως το dual του maximum matching και το set cover είναι ακριβώς ίδια!

(γ). Παρατηρούμε ότι μπορούμε (εξ' ορισμού) να διατυπώσουμε την ακόλουθη ανισοτική σχέση.

Max matching = IP(max. Matching) $\leq$ Relaxed_LP(max. matching) $\leq$ Dual_LP(max. matching) $\leq$ IP_of_Dual_LP(max. matching)=Vertex Cover.

Η πρώτη ανισότητα αποδείξαμε στο πρώτο υποερώτημα ότι ισχύει με ισότητα, επίσης από το θεώρημα του δυισμού γνωρίζουμε ότι για τη βέλτιστη λύση του LP και η δεύτερη ανισότητα είναι ισότητα. Τέλος, μπορούμε να εφαρμόσουμε πάλι την ιδέα του πρώτου υποερωτήματος για να αποδείξουμε ότι και η τρίτη ανισότητα είναι ισότητα. Για αυτό αρκεί να σκεφτούμε ότι ο αντίστοιχος πίνακας 'γειτνίασης' που υπάρχει στο dual πρόβλημα είναι στην πραγματικότητα ο ανάστροφος του primal που προφανώς θα είναι και αυτός TUM άρα θα ισχύουν όλα όσα έχουμε πει.

Επομένως δείξαμε πως Max bipartite matching = Vertex Cover.

Δώστε έναν strict quadratic program για το πρόβλημα MAX k-CUT, ένα vector program relaxation και ένα ισοδύναμο semidefinite program.

Το ζητούμενο στο max k-cut πρόβλημα είναι η διαμοίραση των κόμβων ενός γραφήματος σε k σύνολα S_i με τέτοιο τρόπο ώστε να μεγιστοποιούνται οι τομές μεταξύ των συνόλων αυτών. Αν λοιπόν εισαγάγουμε για κάθε κορυφή i μια μεταβλητή x_i, τότε πρέπει με κάποιον τρόπο να περιορίσουμε το σύνολο τιμών της x_i σε k διαφορετικές τιμές, που η κάθε μια αντιστοιχεί στο να εισαγάγουμε την κορυφή στο αντίστοιχο σύνολο S_i.

Γνωρίζουμε πώς στο max k-cut δεν έχει σημασία σε ποια δύο διαφορετικά σύνολα θα τοποθετηθούν οι κορυφές μιας ακμής, αρκεί που δεν είναι στο ίδιο. Ας ονομάσουμε αυτή την ιδιότητα *ανεξαρτησία ζεύγους.* Αυτή η ιδιότητα θα πρέπει να μεταφερθεί και στις μεταβλητές των κορυφών. Στην περίπτωση του max-cut μπορούμε εύκολα να ορίσουμε ότι $x_i \in \{-1,1\}$ διότι έχουμε μόνο δύο σύνολα. Ωστόσο όταν προσπαθήσουμε να γενικεύσουμε το πρόβλημα στα k σύνολα, τότε αργά ή γρήγορα θα διαπιστώσουμε ότι δεν είναι δυνατόν να ορίσουμε ένα σύνολο τιμών πάνω στον άξονα των ακεραίων για τις μεταβλητές x_i που θα διατηρεί την ιδιότητα της ανεξαρτησίας ζεύγους. Ο περιοριστικός παράγοντας είναι ότι οι x_i παίρνουν τιμές πάνω σε έναν άξονα όπου συμμετρία μπορεί να υπάρξει μόνο για δύο τιμές (βλέπε max-cut). Επομένως η γενίκευση του max-cut στα k σύνολα μας οδηγεί στην γενίκευση της *διάστασης* του χώρου από τον οποίο θα παίρνουν τιμές οι μεταβλητές x_i.

Έστω λοιπόν ένα simplex[3] Σ_κ πολύγωνο στον $\Re^{k-1}$ χώρο και έστω ότι τα διανύσματα $b_1,b_2, \ldots,b_k$ αντιστοιχούν στις κορυφές του πολυγώνου. Επίσης έστω $c_k = (b_1 + b_2 + \ldots + b_k)/k$ το κεντροειδές[4] του Σ_κ και έστω τα διανύσματα $a_i=b_i-c_k$ που ενώνουν το κεντροειδές με κάθε κορυφή ενώ παράλληλα φροντίζουμε να κλιμακώσουμε το κάθε a_i ώστε $|a_i| = 1$.

---

[3] Για κάθε διάσταση d μπορούμε στο χώρο $\Re^d$ να φτιάξουμε ένα simplex πολύγωνο με την ιδιότητα όλες οι κορυφές του να ισαπέχουν. Έτσι στον $\Re^1$ χώρο υπάρχει το διάστημα [-1,1] (βλέπε max-cut), στον $\Re^2$ το ισόπλευρο τρίγωνο, στον $\Re^3$ το τετράεδρο κ.ό.κ.

[4] Το κεντροειδές ή αλλίως κέντρο βάρους ενός πολυγώνου έχει την ιδιότητα ότι το άθροισμα των διανυσμάτων που συνδέουν το κεντροειδές με κάθε μια από τις κορυφές του πολυγώνου είναι μηδέν. Για τα simplex πολύγωνα το κεντροειδές θα ισαπέχει από κάθε κορυφή.

Μπορούμε να αποδείξουμε ότι τα διανύσματα a_i έχουν την ακόλουθη ιδιότητα:

$$a_i a_j = -\frac{1}{k-1} \text{ για } 1 \le i \ne j \le k$$

$$a_i^T a_i = 1$$

Στη συνέχεια θα αποδείξουμε την παραπάνω ιδιότητα. Έστω ότι έχουμε ένα simplex πολύγωνο τοποθετημένο στο χώρο με τέτοιον τρόπο ώστε όλες του οι κορυφές εκτός από μια να βρίσκονται στο ίδιο επίπεδο. Άρα τα διανύσματα $b_1, b_2, \ldots, b_{k-1}$ βρίσκονται στο ίδιο επίπεδο, και ορίζουν κάτι σαν «βάση» στο πολύγωνο. Είναι προφανές ότι το πολύγωνο που ορίζουν τα $b_1, b_2, \ldots, b_{k-1}$ είναι επίσης simplex για τον χώρο διάστασης $\Re^{k-2}$. Έστω τώρα ότι το κεντροειδές της «βάσης» του πολυγώνου έχει τοποθετηθεί στην αρχή των αξόνων. Με αυτό τον τρόπο θα ισχύει ότι $b_1+b_2+\ldots+b_{k-1}=0$, ενώ το διάνυσμα b_k θα βρίσκεται πάνω στον άξονα z σε κάποιο ύψος από το κεντροειδές της βάσης. Ομοίως και το κεντροειδές του πολυγώνου c_k θα βρίσκεται πάνω στον άξονα z λίγο πιο κάτω από την κορυφή b_k. Έστω λοιπόν ότι για τα διανύσματα c_k και b_k ισχύουν τα ακόλουθα: $c_k=(0,0,\ldots,x)$ και $b_k=(0,0,\ldots,kx)$ για κάποιο x>0. Εξ' ορισμού ισχύει ότι $|a_i| = 1 \Rightarrow |b_k - c_k| = 1$ άρα προκύπτει ότι $x = \frac{1}{k-1}$. Οπότε μπορούμε να γράψουμε: $a_k a_1 = (b_k - c_k)(b_1 - c_k) = b_k b_1 - c_k b_1 - b_k c_k + c_k^2 = -b_k c_k + c_k^2$ διότι $b_k, c_k \perp b_1$, κάνοντας ακόμα την αντικατάσταση $x = \frac{1}{k-1}$, έχουμε $a_k a_1 = -\frac{1}{k-1}$. Είναι προφανές ότι στη θέση του α_1 μπορεί να τοποθετηθεί οποιοδήποτε από τα $\alpha_1, \ldots, \alpha_{\kappa-1}$. Επίσης λόγω συμμετρίας μπορούμε να αναποδογυρίσουμε το σχήμα ώστε το α_κ να πάρει τη θέση κάποιου από τα $\alpha_1, \ldots, \alpha_{\kappa-1}$ και πάλι θα ισχύουν τα ίδια. Άρα τελικά η ζητούμενη σχέση θα ισχύει για οποιοδήποτε ζεύγος $\alpha_i\alpha_j$.

Με βάση τα όσα είπαμε παραπάνω μπορούμε άνετα να ορίσουμε το ακόλουθο vector πρόγραμμα για το max k-cut πρόβλημα:

Maximize: $$\frac{k-1}{k} \sum_{i,j \in E} w_{ij} (1 - x_i x_j)$$

Και $$a_i a_i = 1$$

$$a_i a_j = -\frac{1}{k-1}$$

$$x_1, \ldots, x_n \in \{a_1, \ldots, a_k\}$$

Άμεσα από το παραπάνω πρόγραμμα μπορούμε να πάρουμε το ακόλουθο semidefinite πρόγραμμα:

Maximize: $$\frac{k-1}{k} \sum_{i,j \in E} w_{ij} (1 - X_{ij})$$

Και $$X_{ii} = 1 \quad i \in V$$

$$X_{ij} \ge -\frac{1}{k-1} \quad i \ne j \in V$$

$$X \succ 0$$

Το πρόβλημα MAX-2SAT ορίζεται όπως και το MAX-SAT, με τον περιορισμό ότι κάθε πρόταση έχει το πολύ δύο literals. Δώστε ένα quadratic program για το πρόβλημα MAX-2SAT.

Αρχικά εισάγουμε για κάθε Boolean μεταβλητή x_i του MAX-2SAT προβλήματος μια αντίστοιχη y_i μεταβλητή που παίρνει τιμές $\{-1,1\}$. Επίσης εισάγουμε και μια μεταβλητή y_0 που επίσης ανήκει στο $\{-1,1\}$. Θεωρούμε ότι η Boolean μεταβλητή x_i είναι true όταν $y_i=y_0$ και false διαφορετικά. Επίσης ορίζουμε ως u(C) τη συνάρτηση που με 1 ορίζει ότι η C ικανοποιείται και με 0 ότι δεν ικανοποιείται. Αν υποθέσουμε ότι κάθε C έχει μόνο μια Boolean μεταβλητή τότε μπορούμε να δώσουμε τον ακόλουθο ορισμό για την συνάρτηση u(C):

$u(x_i)=\dfrac{1+y_0y_i}{2}$ και αντίστοιχα $u(\overline{x_i})=\dfrac{1-y_0y_i}{2}$.

Στη συνέχεια μπορούμε να υπολογίσουμε τη συνάρτηση u(C) για την C των δύο μεταβλητών χρησιμοποιώντας το γνωστό θεώρημα De Morgan:

$$u(x_i \vee x_j)=1-u(\overline{x_i})u(\overline{x_j})=1-\frac{1-y_0y_i}{2}\frac{1-y_0y_j}{2}=\frac{1}{4}\left(3+y_0y_i+y_0y_j-y_0^2y_iy_j\right)\Rightarrow$$

$$u(x_i \vee x_j)=\frac{1+y_0y_i}{4}+\frac{1+y_0y_j}{4}+\frac{1-y_iy_j}{4}$$

Τέλος θεωρώντας ότι για κάθε πρόταση C_i που αληθεύει έχουμε ένα αντίστοιχο κέρδος w_i, μπορούμε με την βοήθεια των συναρτήσεων u να διατυπώσουμε το ακόλουθο quadratic πρόβλημα για n Boolean μεταβλητές:

Maximize $\displaystyle\sum_{0\le i<j\le n}\left(\frac{1+y_0y_i}{4}+\frac{1+y_0y_j}{4}+\frac{1+y_iy_j}{4}\right)$

Δεδομένου ότι: $y_i^2=1,\quad 0\le i\le n$

Και με $y_i \in Z$

Έστω G ένα μη κατευθυνόμενο γράφημα. Θεωρείστε ένα διανυσματικό πρόγραμμα (vector program) με n-διάστατα διανύσματα $\vec{u}_1,\ldots,\vec{u}_n$, ένα για κάθε κορυφή του G. Τα διανύσματα υπόκεινται στον περιορισμό ότι ανήκουν στην n-διάστατη σφαίρα ακτίνας 1 και κέντρου 0. Επίσης, για κάθε πλευρά {i,j} του G έχουμε:

$$\vec{u}_i\cdot\vec{u}_j\le-\frac{1}{k-1}\ \textbf{όπου k>1 και}\ k\in Z^+$$

Δείξτε ότι εάν ο G είναι k-χρωματίσιμος (δηλαδή έχει χρωματισμό κορυφών με k χρώματα) τότε το ως άνω πρόγραμμα έχει μια τουλάχιστον λύση που ικανοποιεί τους περιορισμούς.

Αρχικά ας ονομάσουμε ως k vector coloring το πρόβλημα που θέλουμε να λύσουμε όπως αυτό ορίζεται στην εκφώνηση της άσκησης. Επομένως επαναδιατυπώνοντας την άσκηση αυτό που ψάχνουμε είναι να αποδείξουμε ότι ένας k-χρωματίσιμος γράφος G είναι και k vector χρωματίσιμος.

Προκειμένου να καταλήξουμε στο ζητούμενο θα αποδείξουμε την ακόλουθη πρόταση:

Για όλους τους θετικούς αριθμούς k και n όπου $k \le n+1$, υπάρχουν k μοναδιαία διανύσματα (μέτρο 1 με κέντρο 0) στο R^n, τέτοια ώστε το εσωτερικό γινόμενο ενός οποιουδήποτε διακριτού ζεύγους είναι $-\frac{1}{k-1}$.

Θα αποδείξουμε τον παραπάνω ισχυρισμό με επαγωγή στο k. Αν αρχίσουμε από k=2 τότε ο ισχυρισμός ισχύει για τα μονοδιάστατα διανύσματα [1] και [-1]. Θεωρούμε ότι ο ισχυρισμός ισχύει για k, δηλαδή μπορούμε να βρούμε k διανύσματα $v_1,....,v_k$ τέτοια ώστε $v_i \cdot v_j \le \frac{-1}{k-1}$ με $i \ne j$. Χρησιμοποιώντας τα διανύσματα v_i θα δημιουργήσουμε τα διανύσματα $u_1,....,u_{k+1}$. Για $i \le k$ θέτουμε:

$$u_i = \left[\frac{\sqrt{(k-1)(k+1)}}{k} v_i^1,..., \frac{\sqrt{(k-1)(k+1)}}{k} v_i^k, \frac{-1}{k} \right]$$

Όπου ο συμβολισμός v_i^j σημαίνει το j-οστό στοιχείο του διανύσματος v_i. Και ορίζουμε το τελευταίο διάνυσμα $u_{k+1} = [0,...,0,1]$. Στη συνέχεια πρέπει να αποδείξουμε ότι οποιοδήποτε ζευγάρι των διανυσμάτων u_i θα έχει εσωτερικό γινόμενο $-\frac{1}{(k+1)-1} = -\frac{1}{k}$.

Αρχικά παρατηρούμε ότι το εσωτερικό γινόμενο του u_{k+1} με οποιοδήποτε από τα υπόλοιπα δίνει $-\frac{1}{k}$.

Επίσης για τα άλλα θα ισχύει ότι:

$$u_i \cdot u_j = \frac{(k-1)(k+1)}{k^2} v_i \cdot v_j + \frac{1}{k^2} = \frac{-(k-1)(k+1)}{k^2(k-1)} + \frac{1}{k^2} = -\frac{1}{k}$$

Επομένως η πρόταση αποδείχθηκε.

Αν λοιπόν θεωρήσουμε ότι έχουμε έναν k-χρωματίσιμο γράφο, τότε βάσει της προηγούμενης πρότασης μπορούμε να βρούμε k διανύσματα που να ικανοποιούν τις συνθήκες του k-vector coloring προβλήματος. Και έτσι αποδείχθηκε το ζητούμενο.