Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών
Αλγόριθμοι και Πολυπλοκότητα
Διδάσκοντες: Σ. Ζάχος, Δ. Φωτάκης
5η Σειρά Γραπτών Ασκήσεων - Σχέδιο Λύσεων

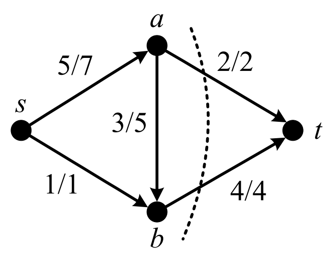
Άσκηση 1: Βελτίωση Οδικού Δικτύου

Μια πρώτη ιδέα που θα μπορούσαμε να δοκιμάσουμε είναι για κάθε νέο δρόμο (u, v), να κατασκευάσουμε ένα νέο δίκτυο G', όπου προσθέτουμε την ακμή (u, v) στο αρχικό δίκτυο G. Για κάθε τέτοιο δίκτυο G', μπορούμε να βρούμε το συντομότερο $s-t$ μονοπάτι με τον αλγόριθμο του Dijkstra. Τέλος, αρκεί να συγκρίνουμε το συντομότερο $s-t$ μονοπάτι από όλα τα δίκτυα G' με το συντομότερο $s-t$ μονοπάτι στο δίκτυο G. Αυτή η λύση χρειάζεται χρόνο $O(|E'|(|E| + |V| \log |V|))$.

Ο παραπάνω αλγόριθμος δεν είναι ο πιο αποδοτικός. Η ιδέα που χρειαζόμαστε είναι ότι ένα μονοπάτι το οποίο μειώνει το κόστος πρέπει να χρησιμοποιεί ακριβώς μία από τις νέες ακμές του συνόλου E' (καθώς μπορούμε να "αγοράσουμε" μόνο μια ακμή). Συνεπώς, αρκεί για κάθε ακμή (u, v), να εξετάσουμε το συντομότερο μονοπάτι που ξεκινά από τον κόμβο s, περνάει από την ακμή (u, v) και καταλήγει στον κόμβο t. Για να υπολογιστεί αυτό το μονοπάτι, χρειαζόμαστε μόνο το συντομότερο μονοπάτι από το s στο u και το συντομότερο μονοπάτι από το v στο t στο δίκτυο G. Το κόστος θα είναι τότε $\mathrm{dist}(s, u) + \ell'(u, v) + \mathrm{dist}(v, t)$.

Μπορούμε να υπολογίσουμε τις αποστάσεις που χρειαζόμαστε με δύο εκτελέσεις του αλγορίθμου του Dijkstra. Αρχικά εκτελούμε τον Dijkstra στο δίκτυο G με αρχική κορυφή s, και υπολογίζουμε τις αποστάσεις $\mathrm{dist}(s, w)$, για κάθε $w \in V$. Για να υπολογίσουμε τις αποστάσεις $\mathrm{dist}(w, t)$, $w \in V$, εκτελούμε τον Dijkstra με αρχική κορυφή t στο δίκτυο που προκύπτει από το G αντιστρέφοντας τη φορά κάθε ακμής. Τέλος, υπολογίζουμε για κάθε ακμή $(u, v) \in E'$, το άθροισμα $\mathrm{dist}(s, u) + \ell'(u, v) + \mathrm{dist}(v, t)$, βρίσκουμε το ελάχιστο τέτοιο άθροισμα, και το συγκρίνουμε με την απόσταση $s-t$ στο αρχικό δίκτυο G. Η χρονική πολυπλοκότητα του αλγορίθμου είναι $\mathrm{O}(|E'| + |E| + |V| \log |V|)$.

Άσκηση 2: Μείωση Μεταφορικής Ικανότητας



Σχήμα 1. Παράδειγμα δικτύου με διαφορετικές χωρητικότητες. Σε κάθε ακμή σημειώνεται η ροή που την διατρέχει στην μέγιστη $s-t$ ροή και η χωρητικότητά της. Η ελάχιστη $s-t$ τομή είναι η $(\{s, a, b\}, \{t\})$ και σημειώνεται με την διακεκομμένη γραμμή. Παρατηρούμε ότι η διαγραφή της ακμής (b, t) μειώνει την μέγιστη ροή κατά 4, ενώ η διαγραφή της ακμής (s, a) μειώνει την μέγιστη ροή κατά 5.

Η μεγαλύτερη δυνατή μείωση στην μέγιστη ροή προκύπτει αν διαγράψουμε οποιεσδήποτε k ακμές από την ελάχιστη $s-t$ τομή. Πράγματι, αφού όλες οι ακμές είναι μοναδιαίας χωρητικότητας, η διαγραφή μιας ακμής μειώνει την μέγιστη ροή το πολύ κατά 1. Η διαγραφή k ακμών από την ελάχιστη τομή μειώνει την ελάχιστη τομή (και συνεπώς και την μέγιστη ροή) κατά k. Η ελάχιστη τομή υπολογίζεται π.χ. με τον αλγόριθμο Edmonds-Karp σε χρόνο $\mathrm{O}(|V||E|^2)$.

Και στην γενική περίπτωση, όπου οι ακμές έχουν διαφορετικές χωρητικότητες, διαγράφοντας μια ακμή της ελάχιστης τομής έχουμε μείωση της μέγιστης ροής ίση με την χωρητικότητα της ακμής. Όπως όμως φαίνεται στο Σχήμα 1, αυτό δεν είναι απαραίτητα το καλύτερο δυνατό.

Άσκηση 3: Προγραμματισμός Εφημεριών

Το πρόβλημα μπορεί να λυθεί με έναν απλό άπληστο αλγόριθμο ως εξής: Εξετάζουμε τους γιατρούς με την σειρά και αναθέτουμε στον γιατρό j να δουλέψει την ημέρα $i \in L_j$ αν και μόνο αν οι γιατροί που μέχρι στιγμής έχουν ανατεθεί στην ημέρα i είναι λιγότεροι από p_i . Η χρονική πολυπλοκότητα του αλγόριθμου είναι $\mathrm{O}(nk)$. Είναι εύκολο να αποδείξουμε ότι ο παραπάνω αλγόριθμος παράγει ένα αποδεκτό πρόγραμμα εφημεριών ανν ένα τέτοιο πρόγραμμα υπάρχει.

Το πρόβλημα γίνεται πιο ενδιαφέρον αν κάθε γιατρός j έχει τον επιπλέον περιορισμό ότι δέχεται να κάνει εφημερία το πολύ ℓ_j, $1 \leq \ell_j \leq |L_j|$, ημέρες από το σύνολο ημερών L_j που δηλώνει διαθέσιμος. Σε αυτή την περίπτωση η λύση του προβλήματος ανάγεται στην εύρεση μιας μέγιστης ροής σε ένα κατάλληλα διαμορφωμένο δίκτυο $G(V, E)$ που αναπαριστά τους περιορισμούς του προβλήματος. Το δίκτυο G κατασκευάζεται ως εξής:

1. Θεωρούμε μια αρχική κορυφή s, μια καταληκτική κορυφή t, ένα σύνολο κορυφών V_1 που περιέχει μια κορυφή για κάθε γιατρό, και ένα σύνολο κορυφών V_2 που περιέχει μια κορυφή για κάθε ημέρα. Το σύνολο κορυφών του δικτύου είναι $V = \{s\} \cup V_1 \cup V_2 \cup \{t\}$, με $|V| = n + k + 2$.
2. Η κορυφή s συνδέεται με ακμή χωρητικότητας ℓ_j με κάθε κορυφή $j \in V_1$.
3. Κάθε κορυφή $i \in V_2$ συνδέεται με ακμή χωρητικότητας p_i με την κορυφή t.
4. Κάθε κορυφή $j \in V_1$ συνδέεται με κάθε κορυφή $i \in L_j$ με ακμή μοναδιαίας χωρητικότητας.
5. Καμία άλλη ακμή δεν υπάρχει στο δίκτυο. Το πλήθος των ακμών του δικτύου είναι $\mathrm{O}(nk)$.

Χωρίς βλάβη της γενικότητας, υποθέτουμε ότι $\sum_{j\in[k]} \ell_j \geq \sum_{i\in[n]} p_i$ (διαφορετικά δεν υπάρχει εφικτό πρόγραμμα εφημεριών τετριμμένα). Προφανώς η μέγιστη $s - t$ ροή στο δίκτυο G είναι μικρότερη ή ίση του $\sum_{i\in[n]} p_i$. Θα δείξουμε ότι υπάρχει αποδεκτό πρόγραμμα εφημεριών ανν η μέγιστη $s-t$ ροή στο δίκτυο G είναι ίση με $\sum_{i\in[n]} p_i$. Συγκεκριμένα, θα δείξουμε ότι κάθε εφικτό πρόγραμμα εφημεριών μπορεί να μετατραπεί σε μια $s - t$ ροή μεγέθους $\sum_{i\in[n]} p_i$, και αντίστροφα.

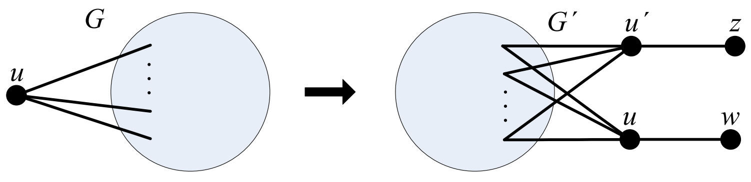
Αρχικά θεωρούμε ένα εφικτό πρόγραμμα εφημεριών $(L'_1, \ldots, L'_k)$. Σε κάθε ακμή (s, j) δρομολογούμε ροή ίση με $|L'_j|$, σε κάθε ακμή (j, i) δρομολογούμε μια μονάδα ροής αν $i \in L'_j$, και μηδενική ροή διαφορετικά, και σε κάθε ακμή (i, t) δρομολογούμε ροή $|\{j : i \in L'_j\}|$. Εξ' ορισμού, η ροή διατηρείται σε κάθε ενδιάμεσο κόμβο. Επιπλέον, αφού το πρόγραμμα εφημεριών $(L'_1, \ldots, L'_k)$ είναι εφικτό, πρέπει $|L'_j| \leq \ell_j$ και $|\{j : i \in L'_j\}| = p_i$. Άρα έχουμε μια εφικτή (και άρα μέγιστη) $s - t$ ροή μεγέθους $\sum_{i\in[n]} p_i$.

Αντίστροφα, θεωρούμε μια ακέραια $s - t$ ροή μεγέθους $\sum_{i\in[n]} p_i$ (από τον αλγόριθμο Ford-Fulkerson, γνωρίζουμε ότι κάθε δίκτυο με ακέραιες χωρητικότητες έχει μια ακέραια μέγιστη ροή). Άρα η ροή σε κάθε ακμή (j, i), $j \in V_1$, $i \in V_2$, είναι είτε 1 είτε 0. Αναθέτουμε σε έναν γιατρό j να εφημερεύει την ημέρα i ανν η ροή στην ακμή (j, i) είναι 1. Το γεγονός ότι το συγκεκριμένο πρόγραμμα εφημεριών είναι εφικτό προκύπτει εύκολα από τις ιδιότητες και το μέγεθος της ροής.

Η χρονική πολυπλοκότητα είναι αυτή που χρειάζεται για να λύσουμε το πρόβλημα της μέγιστης ροής σε ένα δίκτυο με $\Theta(n + k)$ κορυφές, $\mathrm{O}(nk)$ ακμές, και χωρητικότητες $\mathrm{O}(n + k)$.

Άσκηση 4: Αναγωγές και NP-πληρότητα

Όλα τα προβλήματα ανήκουν στο $\mathbf{NP}$, αφού οι υποψήφιες λύσεις τους μπορούν να ελεγχθούν σε πολυωνυμικό χρόνο. Στη συνέχεια παρουσιάζουμε αναγωγές από γνωστά $\mathbf{NP}$ πλήρη προβλήματα, αποδεικνύοντας ότι τα συγκεκριμένα προβλήματα είναι $\mathbf{NP}$-πλήρη.



Σχήμα 2. Σχηματική αναπαράσταση της αναγωγής από το πρόβλημα του κύκλου Hamilton στο πρόβλημα του του Συνδετικού Δέντρου με Ελάχιστο Αριθμό Φύλλων.

Αναγωγή από το πρόβλημα του Κύκλου Hamilton στο πρόβλημα του Συνδετικού Δέντρου με Ελάχιστο Αριθμό Φύλλων. Έστω γράφημα $G(V, E)$ στο οποίο ελέγχουμε την ύπαρξη κύκλου Hamilton. Κατασκευάζουμε γράφημα G' από το G ως εξής: Θεωρούμε μια (αυθαίρετα επιλεγμένη) κορυφή $u \in V$, προσθέτουμε ένα "αντίγραφο" u' της u, και συνδέουμε την u' με όλους τους γείτονες της u στο G (και μόνο με αυτούς). Επιπλέον, προσθέτουμε κορυφή w, που συνδέεται μόνο με την u, και κορυφή z, που συνδέεται μόνο με την u' (βλ. Σχήμα 2). Θέτουμε ακόμη $k = 2$. Προφανώς το G' μπορεί να κατασκευαστεί από το G σε πολυωνυμικό χρόνο.

Θα δείξουμε ότι το G' έχει συνδετικό δέντρο με (το πολύ) 2 φύλλα ανν το G έχει κύκλο Hamilton. Έστω ότι το G έχει κύκλο Hamilton $u - p - u$, όπου p μονοπάτι που περιέχει όλες τις κορυφές του G εκτός της u. Τότε το G' περιέχει το μονοπάτι $w - u - p - u' - z$, το οποίο αποτελεί ένα συνδετικό δέντρο του G' με 2 φύλλα.

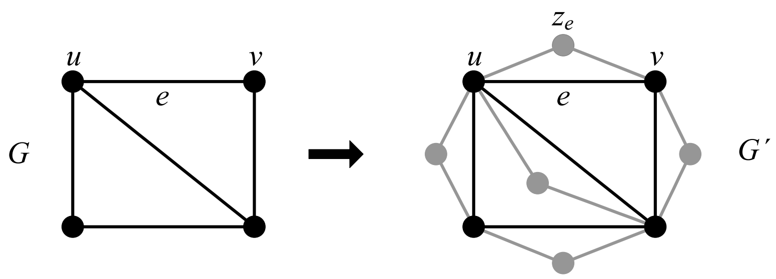
Αντίστροφα, έστω ότι το G' έχει συνδετικό δέντρο με (το πολύ) 2 φύλλα. Κατ΄ ανάγκη, αυτό θα είναι ένα μονοπάτι που περιέχει όλες τις κορυφές του G' και έχει άκρα τις κορυφές w και z. Συνεπώς το γράφημα G' περιέχει μονοπάτι $w - u - p - u' - z$, όπου p μονοπάτι που περιέχει όλες τις κορυφές του G' εκτός των u, u', w, z. Τότε ο κύκλος $u - p - u$ αποτελεί κύκλο Hamilton για το γράφημα G.

Παρατήρηση: Στο πρόβλημα του Μονοπατιού Hamilton, δίνεται ένα γράφημα G, και ελέγχουμε αν υπάρχει (απλό) μονοπάτι που διέρχεται από όλες τις κορυφές του G. Η παραπάνω κατασκευή ουσιαστικά αποτελεί μια αναγωγή από το πρόβλημα του Κύκλου Hamilton στο πρόβλημα του Μονοπατιού Hamilton. Έτσι αποδεικνύουμε ότι το πρόβλημα του Μονοπατιού Hamilton είναι **NP**-πλήρες. Το πρόβλημα του Συνδετικού Δέντρου με Ελάχιστο Αριθμό Φύλλων ουσιαστικά αποτελεί γενίκευση του Μονοπατιού Hamilton, και συνεπώς είναι και αυτό **NP**-πλήρες.

Αναγωγή του Καλύμματος Κορυφών στο Κυρίαρχο Σύνολο. Έστω γράφημα $G = (V, E)$ στο οποίο ελέγχουμε αν υπάρχει κάλυμμα κορυφών C με $|C| \leq k$. Χωρίς βλάβη της γενικότητας, θεωρούμε ότι το γράφημα G είναι συνεκτικό (ως άσκηση, να αιτιολογήσετε αυτό το σημείο).

Κατασκευάζουμε γράφημα G' ώστε κάθε κυρίαρχο σύνολο του G' να αντιστοιχεί σε ένα κάλυμμα κορυφών του G, και αντίστροφα. Το G' προκύπτει από το G προσθέτοντας ένα "παράλληλο" μονοπάτι μήκους 2 για κάθε ακμή του G. Ειδικότερα, το G' περιέχει όλες τις κορυφές και τις ακμές του G. Επιπλέον, για κάθε ακμή $e = \{u, v\} \in E$, το G' περιέχει μια επιπλέον κορυφή z_e και δύο επιπλέον ακμές $\{u, z_e\}$ και $\{z_e, v\}$ που συνδέουν τις u και v με μονοπάτι μήκους 2 μέσω της z_e (βλ. Σχήμα 3). Προφανώς το G' μπορεί να κατασκευαστεί από το G σε πολυωνυμικό χρόνο.

Θα δείξουμε ότι το γράφημα G έχει κάλυμμα κορυφών C με $|C| \leq k$ ανν το γράφημα G' έχει κυρίαρχο σύνολο D με $|D| \leq k$. Πράγματι, κάθε κάλυμμα κορυφών C του G αποτελεί κυρίαρχο σύνολο για το G' γιατί: (α) κάθε κορυφή $u \in V$, είτε ανήκει στο C είτε (εφόσον δεν είναι απομονωμένη) συνδέεται με κορυφή του C (στο G, και άρα στο G'), και (β) κάθε κορυφή z_e συνδέεται με κάποια κορυφή του C στο G' (αφού η αντίστοιχη ακμή e "καλύπτεται" από το C στο G).



Σχήμα 3. Ένα παράδειγμα για την αναγωγή από το πρόβλημα του Καλύμματος Κορυφών στο πρόβλημα του Κυρίαρχου Συνόλου. Οι ακμές και κορυφές που προστίθενται στο G' σημειώνονται γκρίζες.

Αντίστροφα, έστω D ένα κυρίαρχο σύνολο στο G'. Χωρίς βλάβη της γενικότητας, υποθέτουμε ότι $D \subseteq V$ (διαφορετικά, κάθε $z_e \in D$, όπου $e = \{u, v\}$, "κυριαρχεί" μόνο στις z_e, u, v, έτσι μπορούμε να αντικαταστήσουμε την z_e με μία από τις u, v στο D). Κάθε κορυφή z_e συνδέεται με κάποια κορυφή του D στο G'. Επομένως κάθε ακμή $e \in E$ του G έχει τουλάχιστον ένα άκρο της στο D. Άρα το D αποτελεί ένα κάλυμμα κορυφών του G.

Αναγωγή του Καλύμματος Κορυφών στο Δέντρο Steiner. Έστω γράφημα $G = (V, E)$ στο οποίο ελέγχουμε αν υπάρχει κάλυμμα κορυφών C με $|C| \leq k$. Δημιουργούμε ένα πλήρες γράφημα G' με $|V| + |E|$ κορυφές, μία κορυφή u για κάθε $u \in V$, και μια κορυφή z_e για κάθε $e \in E$.

Στο G', μήκος 1 έχουν οι ακμές που συνδέουν είτε κορυφές u, v που αντιστοιχούν σε κορυφές του G, είτε μια κορυφή z_e, που αντιστοιχεί σε ακμή $e = \{u, v\}$ του G, με τα άκρα της u και v. Τα μήκος των υπόλοιπων ακμών του G' καθορίζεται από τις αποστάσεις των άκρων τους στο υπογράφημα του G' που περιλαμβάνει μόνο τις ακμές μήκους 1 (βλ. Σχήμα 4). Έτσι:

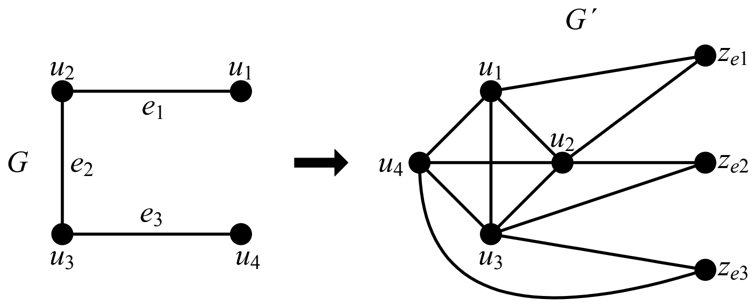
- Μήκος 2 έχουν οι ακμές του G' που συνδέουν είτε δύο κορυφές z_e και $z_{e'}$ για τις οποίες οι αντίστοιχες ακμές e και e' έχουν κοινό άκρο στο G, είτε δύο κορυφές u και z_e που η μια αντιστοιχεί σε κορυφή και η άλλη σε ακμή του G .
- Μήκος 3 έχουν οι ακμές του G' που συνδέουν δύο κορυφές z_e και $z_{e'}$ για τις οποίες οι αντίστοιχες ακμές e και e' δεν έχουν κοινό άκρο στο G.

Εξ' ορισμού, τα μήκη των ακμών ικανοποιούν την τριγωνική ανισότητα. Θέτουμε ακόμη $B = k + |E| - 1$. Στο στιγμιότυπο του Δέντρου Steiner ελέγχουμε αν υπάρχει συνεκτικό υπογράφημα του G' που περιλαμβάνει τις κορυφές του συνόλου $Z = \{z_e : e \in E\}$, δηλ. όλες τις κορυφές του G' που αντιστοιχούν σε ακμές του G, και έχει συνολικό μήκος μικρότερο ή ίσο του B.

Η κατασκευή του G' από το G και ο υπολογισμός του μήκους των ακμών μπορούν να γίνουν σε πολυωνυμικό χρόνο. Θα δείξουμε ότι το G έχει κάλυμμα κορυφών C με $|C| \leq k$ ανν στο G' υπάρχει δέντρο Steiner που καλύπτει τις κορυφές του Z και έχει συνολικό μήκος μικρότερο ή ίσο του $k + |E| - 1$.

Έστω κάλυμμα κορυφών C του G με $|C| \leq k$. Θεωρούμε το υπογράφημα του G' που περιλαμβάνει τις κορυφές του C, $k - 1$ ακμές μήκους 1 που τις συνδέουν μεταξύ τους, τις κορυφές του Z, και για κάθε κορυφή $z_e \in Z$, την ακμή μήκους 1 που συνδέει την z_e με το άκρο της που ανήκει στο C (αν και τα δύο άκρα της e ανήκουν στο C, συνδέουμε την z_e με ένα από τα δύο άκρα). Το υπογράφημα αυτό είναι συνεκτικό γιατί το C "καλύπτει" όλες τις ακμές του G, περιλαμβάνει όλες τις κορυφές του Z, και έχει συνολικό βάρος $k + |E| - 1$.

Για το αντίστροφο, θεωρούμε συνδετικό υπογράφημα T του G' που περιλαμβάνει όλες τις κορυφές του Z και έχει συνολικό μήκος μικρότερο ή ίσο του $k + |E| - 1$. Το σημαντικό είναι πως



Σχήμα 4. Ένα παράδειγμα για την αναγωγή από το πρόβλημα του Καλύμματος Κορυφών στο πρόβλημα του Δέντρου Steiner. Από τις ακμές του (πλήρους) γραφήματος G' εμφανίζονται μόνο αυτές με μήκος 1. Από τις ακμές του G' που δεν εμφανίζονται, η ακμή $\{e_1, e_3\}$ έχει μήκος 3, και όλες οι υπόλοιπες έχουν μήκος 2.

μπορούμε να θεωρήσουμε ότι το T είναι δέντρο και περιλαμβάνει μόνο ακμές του G' με μήκος 1. Αν δεν ισχύει αυτό, μπορούμε να μετατρέψουμε το T σε ένα δέντρο T' που περιλαμβάνει όλες τις κορυφές του Z, έχει μόνο ακμές μήκους 1, και συνολικό μήκος μικρότερο ή ίσο από αυτό του T. Η μετατροπή μπορεί να γίνει ως εξής:

– Αν το T περιλαμβάνει ακμή $\{z_e, z_{e'}\}$ μήκους 3, όπου $e = \{u, v\}$ και $e' = \{u', v'\}$ ακμές στο G, την αντικαθιστούμε με τις ακμές $\{z_e, u\}$, $\{u, u'\}$, και $\{u', z_{e'}\}$, όλες μήκους 1.
– Αν το T περιλαμβάνει ακμή μήκους 2, αυτή:
  • Είτε θα είναι της μορφής $\{z_e, z_{e'}\}$, όπου οι αντίστοιχες ακμές $e = \{u, v\}$ και $e' = \{u, v'\}$ έχουν κοινό άκρο στο G (εδώ το u). Τότε αντικαθιστούμε την ακμή $\{z_e, z_{e'}\}$ με τις ακμές $\{z_e, u\}$ και $\{u, z_{e'}\}$, αμφότερες μήκους 1.
  • Είτε θα είναι της μορφής $\{z_e, w\}$, όπου $e = \{u, v\}$ ακμή και w κορυφή του G. Τότε αντικαθιστούμε την ακμή $\{z_e, w\}$ με τις ακμές $\{z_e, u\}$ και $\{u, w\}$, αμφότερες μήκους 1.

Αν το γράφημα που προκύπτει δεν είναι δέντρο, μπορούμε να το μετατρέψουμε σε δέντρο χωρίς να αυξήσουμε το βάρος του.

Εστιάζουμε λοιπόν στην περίπτωση που το T περιλαμβάνει μόνο ακμές μήκους 1. Αφού το συνολικό μήκος του T είναι μικρότερο ή ίσο του $k + |E| - 1$, το T περιλαμβάνει το πολύ $k + |E|$ κορυφές. Αφού το T περιλαμβάνει όλες τις κορυφές του Z (το πλήθος τους είναι $|E|$), το σύνολο C των κορυφών του T που αντιστοιχούν σε κορυφές του G αποτελείται από k κορυφές το πολύ. Επειδή το T είναι συνεκτικό και οι κορυφές του Z δεν συνδέονται μεταξύ τους με ακμές μήκους 1, κάθε κορυφή $z_e \in Z$ συνδέεται με ακμή μήκους 1 με κάποια κορυφή του C. Από τον ορισμό των μηκών, αυτή είναι κάποιο άκρο της ακμής e στο G. Δηλαδή, για κάθε ακμή e του G, τουλάχιστον ένα από τα άκρα της (αντιστοιχεί σε κορυφή που) ανήκει στο C. Άρα το C ορίζει ένα κάλυμμα κορυφών στο G με αριθμό κορυφών μικρότερο ή ίσο του k.

Αναγωγή του Κυρίαρχου Συνόλου στην Χωροθέτηση Υπηρεσιών. Έστω γράφημα $G = (V, E)$ στο οποίο ελέγχουμε αν υπάρχει κυρίαρχο σύνολο D με $|D| \leq k$. Δημιουργούμε ένα πλήρες γράφημα G' με σύνολο κορυφών V (ίδιο με αυτό του G). Το κόστος κάθε κορυφής του G' είναι ίσο με 2. Οι ακμές του G' που υφίστανται στο G έχουν μήκος 1, οι υπόλοιπες έχουν μήκος 2. Εξ΄ ορισμού τα μήκη των ακμών ικανοποιούν την τριγωνική ανισότητα. Θέτουμε ακόμη $B = |V| + k$.

Το G', μαζί με τα μήκη των ακμών, το κόστη των κορυφών, και το B αποτελούν ένα στιγμιότυπο για το πρόβλημα Χωροθέτησης Υπηρεσιών. Δεδομένου του G, το στιγμιότυπο αυτό υπολογίζεται σε πολυωνυμικό χρόνο. Θα δείξουμε ότι το G έχει κυρίαρχο σύνολο D με $|D| \leq k$ ανν υπάρχει λύση για την Χωροθέτηση Υπηρεσιών στο G' με συνολικό κόστος μικρότερο ή ίσο του $|V| + k$.

Πράγματι, κάθε κυρίαρχο σύνολο D, $|D| \leq k$, στο G ορίζει μια λύση για την χωροθέτηση υπηρεσιών στο G' με συνολικό κόστος μικρότερο ή ίσο του $|V| + k$. Το κόστος εγκατάστασης της υπηρεσίας στις κορυφές του D είναι $2|D|$, και το κόστος εξυπηρέτησης των κορυφών που δεν ανήκουν στο D είναι $|V| - |D|$, αφού όλες συνδέονται με ακμή μήκους 1 με κάποια κορυφή του D (επειδή το D είναι κυρίαρχο σύνολο στο G).

Αντίστροφα, έστω μια λύση για την Χωροθέτηση Υπηρεσιών στο G' η οποία ορίζεται από το σύνολο $C \subseteq V$ και έχει συνολικό κόστος μικρότερο ή ίσο του $|V| + k$. Χωρίς βλάβη της γενικότητας, θεωρούμε ότι κάθε κορυφή του $V \setminus C$ συνδέεται με ακμή μήκους 1 με κάποια κορυφή του C (αν υπάρχει $v \in V \setminus C$ που συνδέεται με ακμές μήκους 2 με όλες τις κορυφές του C, μπορούμε να προσθέσουμε την v στο C χωρίς να αυξήσουμε το συνολικό κόστος). Αφού το συνολικό κόστος είναι $2|C| + (|V| - |C|) \leq |V| + k$, έχουμε ότι $|C| \leq k$. Επιπλέον κάθε κορυφή που δεν ανήκει στο C συνδέεται (στο G) με ακμή με κάποια κορυφή του C. Συνεπώς το C αποτελεί ένα κυρίαρχο σύνολο του G με k κορυφές το πολύ.

Άσκηση 5*: Φωλιασμένα Διαστήματα

Αφού τα s_i είναι ταξινομημένα, αρκεί να υπολογίσουμε για πόσα ζεύγη t_i, t_j, με $i < j$, ισχύει ότι $t_j \leq t_i$. Για κάθε τέτοιο ζεύγος t_i, t_j, έχουμε ότι $s_i \leq s_j$ και $t_j \leq t_i$, άρα $[s_j, t_j] \subseteq [s_i, t_i]$. Το πλήθος αυτών των ζευγών μπορεί να υπολογιστεί σε χρόνο $\mathrm{O}(n \log n)$ όπως στην Άσκηση 4 της 2ης Γραπτής Εργασίας.

Άσκηση 6*: Το πρόβλημα της βιβλιοθήκης

Πρόκειται για το πρόβλημα της επιλογής των σελίδων που διατηρούνται στην “γρήγορη” μνήμη σε υπολογιστές με ιεραρχία μνήμης (το πρόβλημα είναι γνωστό ως *paging* ή *cache maintenance*). Η βέλτιστη πολιτική είναι είναι να επιστρέφουμε το βιβλίο που θα χρειαστούμε τελευταίο (από αυτά που έχουμε). Δείτε την Ενότητα 4.3, σελ. 164-171, στο βιβλίο των Kleinberg και Tardos.

Άσκηση 7*: Κοπή Υφασμάτων

Θα χρησιμοποιήσουμε δυναμικό προγραμματισμό. Για την αρχή της βελτιστότητας, θεωρούμε μια βέλτιστη λύση που κόβει το ύφασμα π.χ. στην θέση k, $1 \leq k < X$, στην οριζόντια διάσταση. Τότε η βέλτιστη εκμετάλλευση του υφάσματος προκύπτει αν θεωρήσουμε την βέλτιστη εκμετάλλευση των δύο τμημάτων υφάσματος που προκύπτουν (ένα με διαστάσεις $k \times Y$ και ένα με διαστάσεις $(X - k) \times Y$). Μάλιστα η βέλτιστη εκμετάλλευση για τα δύο τμήματα υπολογίζεται ανεξάρτητα.

Έτσι το βέλτιστο κέρδος $G(x, y)$, $1 \leq x \leq X$, $1 \leq y \leq Y$, που μπορούμε να έχουμε από ένα κομμάτι υφάσματος με διαστάσεις $x \times y$ δίνεται από την αναδρομική σχέση:

$$G(x,y) = \begin{cases} 0 & \text{αν για κάθε } i \in [n],\, a_i > x \vee b_i > y \\ \max\{\ \max_{1 \leq k < x}\{G(k,y) + G(x-k,y)\}, & \\ \qquad \max_{1 \leq k < y}\{G(x,k) + G(x,k-y)\}, & \text{διαφορετικά} \\ \qquad \max_{i : a_i \leq x \wedge b_i \leq y}\{c_i\}\ \} & \end{cases}$$

Το βέλτιστο κέρδος $G(X, Y)$ υπολογίζεται από την παραπάνω αναδρομική εξίσωση σε χρόνο $\mathrm{O}(X \cdot Y \cdot (n + X + Y))$ με εφαρμογή δυναμικού προγραμματισμού.

Άσκηση 8*: Αντιπροσωπεία Φορητών Υπολογιστών

Θα χρησιμοποιήσουμε δυναμικό προγραμματισμό. Για την αρχή της βελτιστότητας, θεωρούμε μια βέλτιστη ακολουθία παραγγελιών $(p_1, \ldots, p_{n-1}, p_n)$ για n ημέρες, όπου μετά το τέλος της n-οστής ημέρας δεν υπάρχουν αδιάθετοι υπολογιστές. Αν $d_n > S$, τότε η μόνη τιμή που μπορεί να πάρει το p_n σε μια βέλτιστη λύση είναι d_n (όλοι οι υπολογιστές που πωλούνται την n-οστή ημέρα παραγγέλνονται την ίδια ημέρα με κόστος K). Σε αυτή την περίπτωση, η υπακολουθία $(p_1, \ldots, p_{n-1})$ είναι μια βέλτιστη πολιτική παραγγελιών για τις $n-1$ πρώτες ημέρες, όπου στο τέλος της $(n-1)$-οστής ημέρας δεν υπάρχουν αδιάθετοι υπολογιστές. Αν $d_n \leq S$, το p_n μπορεί να είναι είτε d_n (όταν $K \leq d_n C$), οπότε ισχύει ότι και παραπάνω, είτε 0 (όταν $K > d_n C$), οπότε η υπακολουθία $(p_1, \ldots, p_{n-1})$ είναι μια βέλτιστη πολιτική παραγγελιών για τις $n-1$ πρώτες ημέρες, υπό την προϋπόθεση ότι στο τέλος της ημέρας $n-1$ υπάρχουν d_n αδιάθετοι υπολογιστές (που θα αποθηκευθούν με κόστος $d_n C$).

Έστω λοιπόν $\text{cost}[i, s]$, $1 \leq i \leq n$, $0 \leq s \leq S$, το ελάχιστο κόστος παραγγελιών για τις πρώτες i ημέρες, υπό την προϋπόθεση ότι στο τέλος της ημέρας i υπάρχουν s διαθέσιμοι υπολογιστές (το κόστος αποθήκευσής τους δεν συμπεριλαμβάνεται στο $\text{cost}[i, s]$). Σύμφωνα με τα παραπάνω, το $\text{cost}[i, s]$ δίνεται από την αναδρομική σχέση:

$$\text{cost}[i, s] = \begin{cases} \min\{\text{cost}[i-1, s+d_i] + (s+d_i)C, \text{cost}[i-1, 0] + K\} & \text{αν } i \geq 2 \text{ και } s + d_i \leq S \\ \text{cost}[i-1, 0] + K & \text{αν } i \geq 2 \text{ και } s + d_i > S \\ K & \text{αν } i = 1 \end{cases}$$

Το βέλτιστο κόστος δίνεται από το $\text{cost}[n, 0]$, το οποίο υπολογίζεται από την παραπάνω εξίσωση σε χρόνο $\mathrm{O}(nS)$. Η ακολουθία των παραγγελιών μπορεί να υπολογισθεί στον ίδιο χρόνο ανατρέχοντας στις επιλογές που οδήγησαν στον υπολογισμό του βέλτιστου κόστους $\text{cost}[n, 0]$ (οι λεπτομέρειες αφήνονται ως άσκηση).