Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών
Αλγόριθμοι και Πολυπλοκότητα
Διδάσκοντες: Σ. Ζάχος, Δ. Φωτάκης
4η Σειρά Γραπτών Ασκήσεων - Σχέδιο Λύσεων

Άσκηση 1: Ενημέρωση Ελάχιστου Συνδετικού Δέντρου

Διακρίνουμε 4 περιπτώσεις, ανάλογα με το αν η e ανήκει στο T ή όχι, και αν το βάρος της αυξάνεται ή μειώνεται. Το T μπορεί να αλλάξει μόνο αν (i) $e \in T$ και το βάρος της αυξάνεται, και (ii) $e \notin T$ και το βάρος της μειώνεται.

Στην περίπτωση (i), αφαιρούμε την e, "σπάζοντας" έτσι το T σε δύο συνεκτικές συνιστώσες T_1 και T_2 (ο υπολογισμός των T_1 και T_2 γίνεται π.χ. με BFS σε χρόνο $\Theta(n)$). Σε χρόνο $\Theta(m)$, με ένα πέρασμα των ακμών, υπολογίζουμε την ακμή ελάχιστου βάρους e' που διασχίζει την τομή που ορίζεται από τις κορυφές των T_1 και T_2. Αν $w(e') < \hat{w}(e)$, ενημερώνουμε το T προσθέτοντας την e' και αφαιρώντας την e. Διαφορετικά, διατηρούμε το T.

Στην περίπτωση (ii), προσθέτουμε την e στο T, και βρίσκουμε (π.χ. με DFS σε χρόνο $\mathrm{O}(n)$) τον κύκλο C που σχηματίζεται. Σε χρόνο $\mathrm{O}(n)$, ελέγχουμε αν υπάρχει ακμή $e' \in C$ με βάρος μεγαλύτερο του $\hat{w}(e)$. Αν ναι, ενημερώνουμε το T προσθέτοντας την e και αφαιρώντας την e'. Διαφορετικά, διατηρούμε το T.

Άσκηση 2: Bottleneck Spanning Tree

(α) Με στόχο να καταλήξουμε σε άτοπο, υποθέτουμε ότι υπάρχει Ελάχιστο Συνδετικό Δέντρο (ΕΣΔ) T με bottleneck κόστος μεγαλύτερο από το αντίστοιχο κόστος ενός Bottleneck Spanning Tree (BST) T_b, δηλ. ισχύει ότι $c(T_b) < c(T)$. Συνεπώς υπάρχει ακμή $e^* \in T$ με βάρος $w(e)$ μεγαλύτερο από το βάρος κάθε κάθε ακμής του T_b. Αφαιρούμε την ακμή e^* από το T, "σπάζοντάς" το έτσι σε δύο συνεκτικές συνιστώσες T_1 και T_2. Έστω e μια ακμή του T_b που διασχίζει την τομή που ορίζεται από τις κορυφές των T_1 και T_2 (τέτοια e υπάρχει γιατί το T_b είναι συνδετικό δέντρο). Αφού $w(e^*) > w(e)$, το $(T \setminus \{e^*\}) \cup \{e\}$ είναι ένα συνδετικό δέντρο με συνολικό βάρος μικρότερο από το βάρος του T, άτοπο.

(β) Ο αλγόριθμος διαγράφει όλες τις ακμές με βάρος μεγαλύτερο του B, και ελέγχει (με BFS) αν το γράφημα που απομένει είναι συνεκτικό. Αν το γράφημα που απομένει είναι συνεκτικό, το δέντρο του BFS αποτελεί ένα συνδετικό δέντρο με bottleneck κόστος μικρότερο ή ίσο του B. Ο χρόνος εκτέλεσης είναι γραμμικός ως προς το πλήθος ακμών.

(γ) Χωρίς βλάβη της γενικότητας, θεωρούμε ότι όλα τα βάρη των ακμών είναι διαφορετικά (αν δεν είναι, επιτυγχάνουμε το ίδιο αποτέλεσμα ορίζοντας έναν "κανόνα προτίμησης" μεταξύ ακμών ίδιου βάρους). Σε χρόνο $\Theta(m)$, υπολογίζουμε την "median" ακμή e_{med}, η οποία έχει βάρος μικρότερο από τις μισές ακμές του γραφήματος και μεγαλύτερο από τις άλλες μισές. Έστω $w(e_{\mathrm{med}}) = B$ το βάρος της "median" ακμής. Με τον αλγόριθμο του (β), ελέγχουμε σε χρόνο $\Theta(m)$ αν το γράφημα έχει bottleneck spanning tree με κόστος μικρότερο ή ίσο του B.

– Αν ναι, εφαρμόζουμε αναδρομικά τον ίδιο αλγόριθμο στο γράφημα που προκύπτει από τη διαγραφή των ακμών με βάρος μεγαλύτερο του B. Το γράφημα αυτό έχει $m/2$ ακμές και είναι συνεκτικό. Το bottleneck spanning tree για το αρχικό γράφημα είναι το bottleneck spanning tree που θα προκύψει ως απάντηση της αναδρομικής κλήσης.

– Αν όχι, συμπτύσσουμε τις συνεκτικές συνιστώσες που υπολόγισε το BFS, και θεωρούμε το γράφημα με κορυφές αυτές τις συνεκτικές συνιστώσες και ακμές όλες τις μεταξύ τους ακμές στο αρχικό γράφημα. Το γράφημα αυτό έχει $m/2$ ακμές το πολύ, αφού όλες οι ακμές με βάρος μικρότερο του B έχουν συμπτυχθεί στις συνεκτικές συνιστώσες, και μπορεί να υπολογισθεί σε χρόνο $\Theta(m)$, με ένα πέρασμα των ακμών. Το bottleneck spanning tree για το αρχικό γράφημα αποτελείται από τις ακμές του bottleneck spanning tree για το γράφημα που προκύπτει από την σύμπτυξη και τις ακμές του BFS-δάσους για κάθε συνεκτική συνιστώσα.

Η αναδρομή σταματά όταν το σύνολο ακμών με βάρος μικρότερο ή ίσο του B αποτελεί ένα συνδετικό δέντρο του γραφήματος. Σε αυτή την περίπτωση, το bottleneck spanning tree έχει bottleneck κόστος ίσο με B.

Η ορθότητα αποδεικνύεται με επαγωγή. Για το χρόνο εκτέλεσης, παρατηρούμε ότι όλες οι ενέργειες εκτός της αναδρομικής κλήσης απαιτούν χρόνο $\Theta(m)$. Συνεπώς, ο χρόνος εκτέλεσης συνολικά δίνεται από την αναδρομική σχέση $T(m) = T(m/2) + \Theta(m)$, $T(1) = \Theta(1)$, η οποία έχει λύση $T(m) = \Theta(m)$.

Άσκηση 3: Bottleneck Shortest Path Tree

(α) Έστω T ένα *Μέγιστο Συνδετικό Δέντρο* (ΜΣΔ, Maximum Spanning Tree) του G, δηλ. ένα συνδετικό δέντρο με μέγιστο συνολικό βάρος. Θα δείξουμε ότι για κάθε ζευγάρι κορυφών u, v, το μοναδικό $u - v$ μονοπάτι του T αποτελεί ένα $u - v$ μονοπάτι μέγιστης χωρητικότητας.

Για να καταλήξουμε σε άτοπο, θεωρούμε ζεύγος κορυφών u, v για τις οποίες το $u - v$ μονοπάτι μέγιστης χωρητικότητας p^* έχει χωρητικότητα μεγαλύτερη από την χωρητικότητα του $u - v$ μονοπατιού p στο T. Έστω e μια ακμή ελάχιστης χωρητικότητας του p. Αφού $C(p^*) > C(p) = w(e)$, όλες οι ακμές του p^* έχουν βάρος μεγαλύτερο του $w(e)$. Αφαιρούμε την e από το T, "σπάζοντας" το έτσι σε δύο συνεκτικές συνιστώσες T_1 και T_2, η μία από τις οποίες περιέχει την u και η άλλη την v. Έστω e' η ακμή του p^* που διασχίζει την τομή που ορίζεται από τις κορυφές των T_1 και T_2. Αφού $w(e') > w(e)$, το $(T \setminus \{e\}) \cup \{e'\}$ είναι ένα συνδετικό δέντρο με συνολικό βάρος μεγαλύτερο από το βάρος του T, άτοπο.

(β) Ένα ΜΣΔ υπολογίζεται αντίστοιχα με ένα ΕΣΔ, αν σε κάθε βήμα επιλέγουμε μια μέγιστου βάρους ακμή που διασχίζει μια τομή η οποία δεν διασχίζεται από κάποια από τις ήδη επιλεγμένες ακμές. Έτσι ένα ΜΣΔ μπορεί να υπολογιστεί με τον αλγόριθμο του Kruskal, αν αρχικά οι ακμές έχουν ταξινομηθεί σε φθίνουσα σειρά βάρους, ή με τον αλγόριθμο του Prim, αν σε κάθε βήμα, επιλέγεται και εντάσσεται στο δέντρο η κορυφή με το μεγαλύτερο κόστος σύνδεσης / ένταξης.

Άσκηση 4: Αραιά Γραφήματα με Σύντομα Μονοπάτια

(α) Έστω $e = \{x, y\}$ μια οποιαδήποτε ακμή του G. Αρχικά παρατηρούμε ότι $d_H(x, y) \leq 3w(e)$. Πράγματι, αν η e ανήκει στον H, τότε $d_H(x, y) \leq w(e) \leq 3w(e)$. Αν η e δεν ανήκει στον H, τότε από την περιγραφή του αλγόριθμου, $d_H(x, y) \leq 3w(e)$.

Αν λοιπόν θεωρήσουμε ένα οποιοδήποτε ζεύγος κορυφών u, v και ένα συντομότερο $u - v$ μονοπάτι $p = (u = x_0, x_1, \ldots, x_{k-1}, x_k = v)$, έχουμε ότι

$$3d_G(u, v) = \sum_{i=0}^{k-1} 3w(\{x_i, x_{i+1}\}) \geq \sum_{i=0}^{k-1} d_H(x_i, x_{i+1}) \geq d_H(u, v)$$

(β) Αρχικά παρουσιάζουμε μια επαγωγική λύση. Έστω $T(n)$ το μέγιστο πλήθος ακμών ενός (απλού μη κατευθυνόμενου) γραφήματος G με n κορυφές που δεν έχει κύκλο μήκους 3 ή 4. Θα διατυπώσουμε μια αναδρομική εξίσωση για το $T(n)$ που η λύση της δεν υπερβαίνει το $n^{3/2}$.

Έστω $d^{\max}$ ο μέγιστος βαθμός κορυφής του G, και έστω u κορυφή του G με βαθμό $d^{\max}$. Γνωρίζουμε ότι οι γειτονικές κορυφές της u δεν συνδέονται με ακμή στο G (διαφορετικά θα είχαμε κύκλο μήκους 3), και ότι κάθε κορυφή που δεν είναι γειτονική της u, συνδέεται με ακμή με μία το πολύ γειτονική κορυφή τη u (διαφορετικά θα είχαμε κύκλο μήκους 4). Άρα αν αφαιρέσουμε την u και όλες τις γειτονικές κορυφές της από το G, καθώς και όλες τις ακμές που προσπίπτουν σε αυτές τις κορυφές, προκύπτει ένα γράφημα G' με $n-(d^{\max}+1)$ κορυφές και τουλάχιστον $T(n)-(n-1)$ ακμές. Επειδή το G' επίσης δεν έχει κύκλο μήκους 3 ή 4, ισχύει ότι $T(n-d^{\max}-1) \geq T(n)-n+1$. Χρησιμοποιώντας ότι $d^{\max} \geq 2T(n)/n$ και ότι η $T(n)$ είναι μια αύξουσα συνάρτηση του n, καταλήγουμε ότι

$$T(n) \leq T(n-1-2T(n)/n)+n-1 \qquad (1)$$

με αρχικές συνθήκες $T(1)=0, T(2)=1, T(3)=2, T(4)=3, \ldots$. Με αναλυτικές μεθόδους, μπορούμε να επιβεβαιώσουμε ότι $T(n) < n^{3/2}$. Ειδικότερα, μπορούμε να επιβεβαιώσουμε ότι για $T(n) = n^{3/2}$, η (1) δεν ισχύει όταν $n > 5$, και ότι η (1) δεν ισχύει για συναρτήσεις που αυξάνονται γρηγορότερα από την $n^{3/2}$.

Επίσης παρουσιάζουμε μια διαφορετική λύση που αποδίδεται στον Paul Erdös. Θεωρούμε ένα οποιοδήποτε (απλό μη κατευθυνόμενο) γράφημα G με n κορυφές που δεν έχει κύκλο μήκους 3 ή 4. Ένα άνω φράγμα στον αριθμό των ακμών του G προκύπτει υπολογίζοντας το πλήθος των διαφορετικών "κερασιών" $K_{1,2}$ (δηλ. επαγόμενων υπογραφημάτων με 3 κορυφές εκ των οποίων οι δύο "ακραίες" συνδέονται με ακμή με την "κεντρική" αλλά όχι μεταξύ τους) με δύο διαφορετικούς τρόπους, και συγκρίνοντας τα αποτελέσματα.

Κάθε κορυφή v βαθμού $d(v)$ αποτελεί το "κέντρο" $\binom{d(v)}{2}$ διαφορετικών "κερασιών". Συνεπώς το πλήθος των διαφορετικών "κερασιών" στον G είναι ίσο με:

$$\sum_{v \in V} \binom{d(v)}{2}$$

Επίσης παρατηρούμε ότι το πλήθος των "κερασιών" του G δεν ξεπερνά το $\binom{n}{2}$ (δηλ. το πλήθος των ζευγαριών κορυφών που μπορούν να αποτελέσουν τα "άκρα" ενός "κερασιού"). Έτσι καταλήγουμε ότι:

$$\binom{n}{2} \geq \sum_{v \in V} \binom{d(v)}{2} \geq n \binom{\overline{d}}{2},$$

όπου $\overline{d} = 2m/n$ είναι ο μέσος βαθμός του G, και η τελευταία ανισότητα ισχύει γιατί η συνάρτηση $\binom{x}{2}$ είναι κυρτή ως προς x. Λύνοντας ως προς $\overline{d}$, προκύπτει ότι $\overline{d} \leq \sqrt{n-1}+1$, δηλ. ότι

$$m \leq \frac{n\sqrt{n-1}+n}{2}$$

(γ) Θα δείξουμε ότι ο H δεν περιέχει κύκλους μήκους 3 ή 4, από το οποίο, λόγω του (β), προκύπτει το ζητούμενο. Στη συνέχεια αποκλείουμε την ύπαρξη κύκλου μήκους 4 στον H. Η ύπαρξη κύκλου μήκους 3 στον H αποκλείεται με τον ίδιο τρόπο.

Έστω ότι υπάρχει στον H κύκλος (x,y,z,w) μήκους 4, και έστω $e_1 = \{x,y\}, e_2 = \{y,z\}, e_3 = \{z,w\}$, και $e_4 = \{w,x\}$ οι αντίστοιχες ακμές. Χωρίς βλάβη της γενικότητας, θεωρούμε ότι $w(e_1) \leq$

$w(e_2) \leq w(e_3) \leq w(e_4)$, και επομένως η e_4 εξετάζεται για προσθήκη στον H, αφού έχουν ήδη επιλεγεί οι ακμές e_1, e_2, και e_3. Την στιγμή που ο αλγόριθμος εξετάζει την e_4, ισχύει για την απόσταση $d_H(x, w)$ των άκρων της στον H ότι:

$$\begin{aligned} d_H(x, w) &\leq d_H(x, y) + d_H(y, z) + d_H(z, w) \\ &\leq w(e_1) + w(e_2) + w(e_3) \leq 3w(e_4) \end{aligned}$$

Επομένως η ακμή e_4 δεν επιλέγεται από τον αλγόριθμο, και ο αντίστοιχος κύκλος δεν μπορεί να σχηματιστεί στον H.