



NATIONAL AND KAPODISTRIAN UNIVERSITY OF ATHENS

**SCHOOL OF SCIENCES
DEPARTMENT OF INFORMATICS AND TELECOMMUNICATIONS**

BSc THESIS

Study of Algorithms for Gomory-Hu Trees

Antonis E. Paronis

Supervisor: Stavros Kolliopoulos, Professor NKUA

ATHENS

JUNE 2022



ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

**ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ**

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Μελέτη Αλγορίθμων Κατασκευής Δέντρων Gomory-Hu

Αντώνιος Ε. Παρώνης

Επιβλέπων: Σταύρος Κολλιόπουλος, Καθηγητής ΕΚΠΑ

ΑΘΗΝΑ

ΙΟΥΝΙΟΣ 2022

BSc THESIS

Study of Algorithms for Gomory-Hu Trees

Antonios E. Paronis

S.N.: 1115201500126

SUPERVISOR: Stavros Kolliopoulos, Professor NKUA

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Μελέτη Αλγορίθμων Κατασκευής Δέντρων Gomory-Hu

Αντώνιος Ε. Παρώνης

A.M.: 1115201500126

ΕΠΙΒΛΕΠΩΝ: Σταύρος Κολλιόπουλος, Καθηγητής ΕΚΠΑ

ABSTRACT

This thesis is a study on the research of A. Bhargat, R. Hariharan, T. Kavitha and D. Panigrahi "An $\tilde{O}(mn)$ Gomory-Hu Tree Construction Algorithm for Unweighted Graphs" [12].

We present the notion of graph flow, explain the Multi-Terminal Network Flow problem, as well as the concept of cut-equivalent trees, also known as Gomory-Hu trees, accompanied with a proof of correctness. After we introduce prerequisite concepts, we present two efficient algorithms for determining the edge connectivity [10] and Steiner edge connectivity [12], and how the latter can be utilized in the construction of Gomory-Hu trees for undirected and unweighted graphs in $\tilde{O}(mn)$ time.

Finally, we present the various research results of studies on Gomory-Hu trees, some of their practical applications as well as the open problems in this area.

SUBJECT AREA: Graph Theory

KEYWORDS: Gomory-Hu trees, Maximum Flow - Minimum Cut, Edge Connectivity, Steiner Edge Connectivity, All Pairs Max-Flow Problem

ΠΕΡΙΛΗΨΗ

Το παρόν έργο αποτελεί μια μελέτη πάνω στην έρευνα των A. Bhargat, R. Hariharan, T. Kavitha και D. Panigrahi "An $\tilde{O}(mn)$ Gomory-Hu Tree Construction Algorithm for Unweighted Graphs" [12].

Παρουσιάζουμε τις έννοιες της ροής σε γραφήματα, το πρόβλημα της μέγιστης ροής σε γραφήματα με πολλαπλά τερματικά, καθώς και την έννοια του δέντρου Gomory-Hu, με μια συνοπτική απόδειξη για την ορθότητα του. Αφού εισάγουμε τις αναγκαίες έννοιες, παρουσιάζουμε δύο αποδοτικούς αλγόριθμους ακμοσυνεκτικότητας [10] και Steiner ακμοσυνεκτικότητας [12], και πώς μπορεί ο τελευταίος να χρησιμοποιηθεί για την κατασκευή δέντρων Gomory-Hu για μη κατευθυνόμενα και μη έμβαρα γραφήματα σε σχεδόν γραμμικό χρόνο $\tilde{O}(mn)$.

Τέλος, παρουσιάζουμε αποτελέσματα που έχουν προκύψει από μεταγενέστερες έρευνες που έχουν γίνει πάνω στα Gomory-Hu δέντρα, μερικές πρακτικές τους εφαρμογές καθώς και τα ανοιχτά προβλήματα πάνω σε αυτή τη θεματική περιοχή.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Θεωρία Γραφημάτων

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Δέντρα Gomory-Hu, Μέγιστη Ροή - Ελάχιστη Τομή, Ακμοσυνεκτικότητα, Ακμοσυνεκτικότητα Steiner, Μέγιστη Ροή Πολλαπλών Τερματικών

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να ευχαριστήσω τον επιβλέποντα μου, Σταύρο Κολλιόπουλο για την καθοδήγηση του στην διαδικασία εκπόνησης της πτυχιακής εργασίας καθώς και για την προθυμία του να με βοηθήσει με όποιο εμπόδιο και αν συνάντησα. Ακόμα θα ήθελα να ευχαριστήσω την οικογένεια και τους φίλους μου για την στήριξη και αγάπη τους όλα αυτά τα χρόνια.

ΠΕΡΙΕΧΟΜΕΝΑ

1. ΕΙΣΑΓΩΓΙΚΕΣ ΕΝΝΟΙΕΣ	18
1.1 Μέγιστη ροή γραφήματος	18
1.2 Ακμοσυνεκτικότητα & Ακμοσυνεκτικότητα Steiner	20
1.3 Δέντρα Gomory Hu	20
2. ΑΠΟΔΟΤΙΚΟΙ ΑΛΓΟΡΙΘΜΟΙ ΑΚΜΟΣΥΝΕΚΤΙΚΟΤΗΤΑΣ ΚΑΙ ΚΑΤΑΣΚΕΥΗΣ ΔΕΝΤΡΩΝ GOMORY-HU	31
2.1 Αλγόριθμος του Gabow για ακμοσυνεκτικότητα γραφήματος	31
2.2 Αλγόριθμος εύρεσης τομής Steiner	36
2.2.1 Αμετάβλητες συνθήκες του αλγόριθμου	39
2.2.2 Υπορουτίνες αλγορίθμου	39
2.2.2.1 Υπολογισμός συνόλου εγκλεισμού	39
2.2.2.2 Μετασχηματισμοί δέντρων	43
2.2.2.3 Ισοζυγισμός Βαθμού	45
Τοπικός Ισοζυγισμός Βαθμού	46
Καθολικός Ισοζυγισμός Βαθμού	47
2.2.3 Πολυπλοκότητα Αλγόριθμου	47
2.3 Κατασκευή Δέντρων Gomory-Hu	49
3. ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΗ ΕΡΕΥΝΑ	54
ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΑΣ	56
ΑΝΑΦΟΡΕΣ	59

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

1.1	Ροή ανάμεσα σε τρεις κορυφές ενός γραφήματος	21
1.2	Ροή σε ένα μονοπάτι $x_1, x_2, \dots, x_k, x_{k+1}$	21
1.3	23
1.4	Σύμφωνα με το Θεώρημα 3, η ελάχιστη τομή $x-y$ στο γράφημα με τις συν- θλιμμένες κορυφές είναι ίση με την τομή στο αρχικό γράφημα 1.3	23
1.5	Το αρχικό γράφημα G	26
1.6	Η τομή στο αρχικό γράφημα G	26
1.7	Το Gomory-Hu δέντρο μετά τη πρώτη τομή $e-a$	26
1.8	Η τομή που χωρίζει τα b και d	27
1.9	Το Gomory-Hu δέντρο μετά την τομή $b-d$	27
1.10	Η τομή που χωρίζει τα b και e	27
1.11	Το Gomory-Hu δέντρο μετά την τομή $b-e$	27
1.12	Η τομή που χωρίζει τα a και g	28
1.13	Το Gomory-Hu δέντρο μετά την τομή $a-g$	28
1.14	Η τομή που χωρίζει τα c και a	28
1.15	Το Gomory-Hu δέντρο μετά την τομή $c-a$	28
1.16	Η τομή που χωρίζει τα f και g	28
1.17	Το τελικό Gomory-Hu δέντρο	28
1.18	29
2.1	Αριστερά: Παράδειγμα r -κατευθυνόμενου δέντρου Δεξιά: Παράδειγμα r -επικαλύπτοντος δέντρου χωρίς κατεύθυνση	32
2.2	Παράδειγμα του τελευταίου "δέντρου" T_{k+1} στον αλγόριθμο του Gabow	33
2.3	Παράδειγμα του μετασχηματισμού mate-swap [7]	45
2.4	Παράδειγμα του μετασχηματισμού mate [7]	45

ΠΡΟΛΟΓΟΣ

Αυτή η πτυχιακή εργασία διεξάχθηκε στα πλαίσια απόκτησης του πτυχίου μου στο τμήμα Πληροφορικής και Τηλεπικοινωνιών στο Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών.

1. ΕΙΣΑΓΩΓΙΚΕΣ ΕΝΝΟΙΕΣ

1.1 Μέγιστη ροή γραφήματος

Το πρόβλημα της μέγιστης ροής ενός γραφήματος, ή αλλιώς ενός δικτύου, είναι το πρόβλημα αναζήτησης της μέγιστης ροής από μια κορυφή s , την πηγή, σε μια κορυφή t , την καταβόθρα. Ένα παράδειγμα από την καθημερινότητα είναι το εξής:

Έστω ότι έχουμε ένα δίκτυο σιδηροδρομικών σταθμών V που ενώνονται με σιδηροδρομικές γραμμές που στο σύνολο τους συμβολίζουμε ως E . Έστω ότι μέσω κάθε σιδηρόδρομου μπορεί να μεταφερθεί μια μέγιστη ποσότητα φορτίου ανά ώρα, την οποία συμβολίζουμε ως c_{uv} ανάμεσα σε 2 σταθμούς u και v . Μας ενδιαφέρει να βρούμε τη μέγιστη ποσότητα φορτίου που μπορούμε να μεταφέρουμε από έναν αρχικό σταθμό s σε έναν σταθμό t . Ο σταθμός s έχει θεωρητικά μια πολύ μεγάλη ποσότητα φορτίου και μπορεί συνεχώς να στέλνει φορτίο στους άλλους σταθμούς, ο σταθμός t μπορεί να δέχεται όσο φορτίο λαμβάνει από τους σιδηρόδρομους που καταλήγουν σε αυτόν χωρίς περιορισμό, ενώ οι ενδιάμεσοι σταθμοί δεν μπορούν να στείλουν περισσότερη ποσότητα απ' όση δέχονται ανά ώρα.

Στο παράδειγμα μας, μπορούμε διαισθητικά να καταλάβουμε τι είναι η ροή, αλλά μπορούμε να ορίσουμε την έννοια και πιο αυστηρά.

Δίκτυο είναι ένα κατευθυνόμενο γράφημα $G = (V, E)$ με συνάρτηση βάρους $c : E \mapsto \mathbb{R}_+$ που ονομάζεται χωρητικότητα ακμής. Για συντομία μπορούμε να γράψουμε $c((u, v)) = c_{uv}$. Στα δίκτυα συχνά θεωρούμε έναν κόμβο πηγή (source) και έναν κόμβο καταβόθρα (sink). Ανάμεσα σε αυτούς τους δύο κόμβους μπορούμε να ορίσουμε μια συνάρτηση ροής $f : V \times V \mapsto \mathbb{R}_+$, την οποία μπορούμε να συμβολίσουμε ως $f_{uv} = f(u, v)$ για την οποία ισχύουν οι εξής συνθήκες:

1. Περιορισμός χωρητικότητας: Η ροή που περνά μέσα από μια ακμή δεν μπορεί να ξεπερνά τη χωρητικότητά της, δηλ. $f_{(i,j)} \leq c_{ij}$ ($f_{(i,j)}$ σε αυτή την περίπτωση εννοούμε τη ροή που περνά από την ακμή (i, j) και όχι τη συνολική ροή f_{ij} που μπορεί να είναι μεγαλύτερη από τη χωρητικότητα της ακμής (i, j))
2. Η συνολική ροή που εισέρχεται σε ένα κόμβο πρέπει να είναι ίση με τη ροή που εξέρχεται από αυτόν, εκτός από την πηγή και καταβόθρα στις οποίες η εισερχόμενη και η εξερχόμενη ροή είναι μηδενική αντίστοιχα. Άρα,

$$\sum_{\forall v \in V, (v,u) \in E} f_{vu} = \sum_{\forall v \in V, (u,v) \in E} f_{uv}, \forall u \in V \setminus \{s, t\}$$

3. Όταν αναφερόμαστε στη ροή ανάμεσα σε δύο κορυφές $s-t$, για δύο κορυφές $u, v \in V$, ισχύει

$$\begin{aligned} f(u, v) &= -f(v, u), & \text{αν το γράφημα είναι κατευθυνόμενο} \\ f(u, v) &= f(v, u), & \text{αν το γράφημα είναι μη κατευθυνόμενο} \end{aligned}$$

Η έννοια της ροής υπάρχει και στα μη κατευθυνόμενα γραφήματα με μόνη διαφορά ότι δεν υπάρχει περιορισμός κατεύθυνσης στις συναρτήσεις ροής και χωρητικότητας, δηλαδή οι c και f είναι συμμετρικές συναρτήσεις ($c_{uv} = c_{vu}$).

Μπορούμε να αναφερθούμε στη μέγιστη ροή ανάμεσα σε μια πηγή s και μια καταβόθρα t ως *μέγιστη ροή* $s-t$. Ένας αλγόριθμος που υπολογίζει αυτή τη ροή είναι των Ford-Fulkerson με πολυπλοκότητα $O(|E|f_{max})$ όπου $f_{max} = \max_{(s,t) \in V \times V} f(s,t)$ η μέγιστη ροή $s-t$. Οι Ford-Fulkerson κατά την απόδειξη της ορθότητας του αλγορίθμου, αποδεικνύουν και ένα πολύ σημαντικό θεώρημα. Πρωτού το παρουσιάσουμε είναι σημαντικό να παρουσιάσουμε την έννοια της τομής.

Μια *τομή* (*cut*) $C = (S, V \setminus S)$ ενός γραφήματος $G = (V, E)$ είναι ένα σύνολο ακμών των οποίων η αφαίρεση του αφήνει το γράφημα μη συνεκτικό και διαμερίζει το γράφημα G σε δύο μη κενά και ξένα μεταξύ τους σύνολα κορυφών S και $V \setminus S$ με $S \subset V, S \neq \emptyset$. Στα κατευθυνόμενα γραφήματα, το σύνολο S ονομάζεται σύνολο-πηγή της τομής και η τομή αποτελείται από όλες τις ακμές που κατευθύνονται από το σύνολο-πηγή της τομής προς το σύνολο $V \setminus S$. Στα μη κατευθυνόμενα γραφήματα, η κατεύθυνση δεν μας ενδιαφέρει, επομένως η τομή περιέχει όλες τις κορυφές ανάμεσα στα δύο σύνολα S και $V \setminus S$. Ως *μέγεθος τομής* ορίζουμε το άθροισμα της χωρητικότητας όλων των ακμών της τομής, δηλαδή:

$$C(S, T) = \sum_{u \in S, v \in T, (u,v) \in E} c(u, v), \text{ στα κατευθυνόμενα γραφήματα}$$

$$C(S, T) = \sum_{u \in S, v \in T, \{u,v\} \in E} c(u, v), \text{ στα μη κατευθυνόμενα γραφήματα}$$

Μια *τομή* $s-t$ είναι μια τομή, όπου μετά την αφαίρεση των αντίστοιχων τομών, οι κόμβοι s και t ανήκουν σε διαφορετικά σύνολα S και $V \setminus S$. Στα κατευθυνόμενα γραφήματα, η κορυφή s πρέπει να ανήκει στο σύνολο-πηγή της τομής. Επίσης, στα κατευθυνόμενα γραφήματα, μερικές φορές μας ενδιαφέρει μόνο η κορυφή-πηγή r της τομής. Μια *r -τομή* για ένα κατευθυνόμενο γράφημα $G = (V, E)$ είναι μια τομή $C = (S, V \setminus S)$ όπου η κορυφή r ανήκει στο σύνολο-πηγή S .

Αφού ορίσαμε την έννοια της τομής, μπορούμε να παρουσιάσουμε το θεώρημα των Ford-Fulkerson, που το απόδειξαν για τα δίκτυα με ακέραια χωρητικότητα [16]:

Θεώρημα 1 (Θεώρημα Μέγιστης ροής-Ελάχιστης τομής). *Η μέγιστη ροή ανάμεσα σε δύο κόμβους $s-t$ είναι ίση με την ελάχιστη τομή $s-t$.*

Είναι χρήσιμο να παρατηρήσουμε πως οι προτάσεις του κεφαλαίου ισχύουν και τα έμβαρα και τα μη έμβαρα γραφήματα. Δηλαδή μπορούμε να ορίσουμε την έννοια της ροής και σε ένα γράφημα χωρίς βάρη. Η συνάρτηση χωρητικότητας $c : E \mapsto \mathbb{R}_+$ θα είναι ίση με τη σταθερή συνάρτηση $c(e) = 1, \forall e \in E$. Έτσι μπορούμε να παρατηρήσουμε ότι η έννοια της τομής ταυτίζεται με την έννοια της ακμοσυνεκτικότητας στα μη έμβαρα γραφήματα, καθώς μια ακμή στο σύνολο ακμών της τομής συνεισφέρει κατά 1 στην τομή.

Χρησιμοποιώντας την έννοια της ροής ανάμεσα σε δύο κορυφές, μπορούμε να ορίσουμε το πρόβλημα εύρεσης της μέγιστης ροής ανάμεσα σε πολλαπλά τερματικά (multi-terminal flow). Ένα σύννηθες πρόβλημα είναι να προσπαθήσουμε να βρούμε τη ροή ανάμεσα σε όλες τις κορυφές του γραφήματος. Μια απλοϊκή λύση σε αυτό το πρόβλημα είναι να υπολογίσουμε για όλα τα ζευγάρια τη μέγιστη ροή ανάμεσα τους, δηλαδή χρειαζόμαστε $\binom{n}{2}$ υπολογισμούς μέγιστης ροής σε ένα δίκτυο.

1.2 Ακμοσυνεκτικότητα & Ακμοσυνεκτικότητα Steiner

Ορίσαμε την έννοια της ελάχιστης τομής $s-t$. Θα ορίσουμε διαφορετικούς τύπους ακμοσυνεκτικότητας που θα μας είναι χρήσιμοι αργότερα για τα μη έμβαρα γραφήματα.

Ορισμός 1 (Ακμοσυνεκτικότητα (μη κατευθυνόμενα γραφήματα)). Ακμοσυνεκτικότητα ενός γραφήματος $G = (V, E)$ είναι ο ελάχιστος αριθμός ακμών που πρέπει να αφαιρέσουμε από το γράφημα έτσι ώστε το γράφημα να χωριστεί σε δύο ή περισσότερα συνεκτικά σύνολα.

Η έννοια της ακμοσυνεκτικότητας μπορεί να χρησιμοποιηθεί και στα κατευθυνόμενα γραφήματα:

Ορισμός 2 (Ακμοσυνεκτικότητα (κατευθυνόμενα γραφήματα)). Ακμοσυνεκτικότητα ενός κατευθυνόμενου γραφήματος $G = (V, E)$ είναι ο ελάχιστος αριθμός ακμών που μπορούμε να αφαιρέσουμε από το γράφημα έτσι ώστε το γράφημα να πάψει να είναι ισχυρά συνεκτικό (strongly connected), δηλαδή να υπάρχει έστω ένα ζευγάρι διαφορετικών κορυφών $s, t \in V$ για τις οποίες δεν υπάρχει κατευθυνόμενο μονοπάτι $s-t$ που να τις ενώνει.

Πέρα από την ακμοσυνεκτικότητα ενός γραφήματος μας είναι επίσης χρήσιμη και η ακμοσυνεκτικότητα ανάμεσα σε δύο κορυφές $s-t$:

Ορισμός 3 (Ακμοσυνεκτικότητα κορυφών (σε μη κατευθυνόμενο γράφημα)). Ακμοσυνεκτικότητα δύο κορυφών $s-t$ ενός μη κατευθυνόμενου γραφήματος $G = (V, E)$ είναι ο ελάχιστος αριθμός ακμών που μπορούμε να αφαιρέσουμε από το γράφημα έτσι ώστε οι δύο κορυφές να πάψουν να συνδέονται μέσω ενός μονοπατιού.

Ορισμός 4 (Ακμοσυνεκτικότητα κορυφών (σε κατευθυνόμενο γράφημα)). Ακμοσυνεκτικότητα δύο κορυφών $s-t$ ενός κατευθυνόμενου γραφήματος $G = (V, E)$ είναι ο ελάχιστος αριθμός ακμών που μπορούμε να αφαιρέσουμε από το γράφημα έτσι ώστε οι δύο κορυφές να πάψουν να συνδέονται μέσω ενός κατευθυνόμενου μονοπατιού $s-t$.

Τέλος, ορίζουμε έναν πιο ειδικό τύπο ακμοσυνεκτικότητας, την ακμοσυνεκτικότητα Steiner.

Ορισμός 5 (Ακμοσυνεκτικότητα Steiner). Η ακμοσυνεκτικότητα Steiner για ένα σύνολο κορυφών S ενός κατευθυνόμενου γραφήματος $G = (V, E)$ με $S \subseteq V$ είναι ο ελάχιστος αριθμός ακμών που μπορούμε να αφαιρέσουμε από το γράφημα έτσι ώστε οποιοσδήποτε κορυφές του συνόλου $u, v \in S$ να σταματήσουν να συνδέονται με κατευθυνόμενο μονοπάτι $s-t$, και αντίστοιχα να πάψουν να συνδέονται με ένα μονοπάτι για μη κατευθυνόμενα γραφήματα.

Αντίστοιχα μπορούμε να ορίσουμε την έννοια της τομής Steiner

Ορισμός 6 (Τομή Steiner). Μια τομή Steiner για ένα σύνολο κορυφών S ενός γραφήματος $G = (V, E)$ με $S \subseteq V$ είναι η ελάχιστη τομή $C = (A, V \setminus A)$ στο γράφημα G , τέτοια ώστε να χωρίζει το σύνολο S , δηλαδή $S \not\subseteq A, S \not\subseteq V \setminus A$.

1.3 Δέντρα Gomory Hu

Οι Gomory και Hu δείξαν ότι για ένα μη κατευθυνόμενο γράφημα $G = (V, E)$ είναι δυνατόν να κατασκευάσουμε έναν δέντρο $G' = (V, E')$, όπου τα δύο γραφήματα θα είναι ισοδύναμα ως προς τις τομές. Πιο συγκεκριμένα η ελάχιστη τομή (και η μέγιστη ροή, σύμφωνα με το

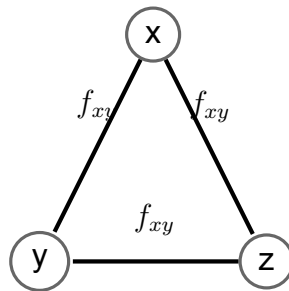
Θεώρημα 1) ανάμεσα σε οποιοσδήποτε δύο κορυφές $u, v \in V$ είναι οι ίδιες και για τα δύο γραφήματα. Παρακάτω παρουσιάζουμε την απόδειξη της ύπαρξης των δέντρων Gomory-Hu, καθώς και την ορθότητα της κατασκευής τους.

Αρχικά παρατηρούμε ότι ισχύει το παρακάτω Λήμμα:

Λήμμα 1. [11] Για να μπορεί μια συνάρτηση $f : V \times V \mapsto \mathbb{R}_+$ να χαρακτηριστεί ως συνάρτηση ροής ενός μη κατευθυνόμενου γραφήματος $G = (V, E)$ πρέπει να ισχύει η συνθήκη:

$$f_{xy} \geq \min [f_{xz}, f_{zy}], \forall x, y, z \in V$$

Απόδειξη. Έστω οι κορυφές $\exists x, y, z \in V$, για τις οποίες $f_{xy} < \min [f_{xz}, f_{zy}]$. Από το Θεώρημα μέγιστης ροής-ελάχιστης τομής 1 έχουμε ότι υπάρχει τομή (A, \bar{A}) , για την οποία $c(A, \bar{A}) = f_{xy}$. Έχουμε δύο ενδεχόμενα:



Σχήμα 1.1: Ροή ανάμεσα σε τρεις κορυφές ενός γραφήματος

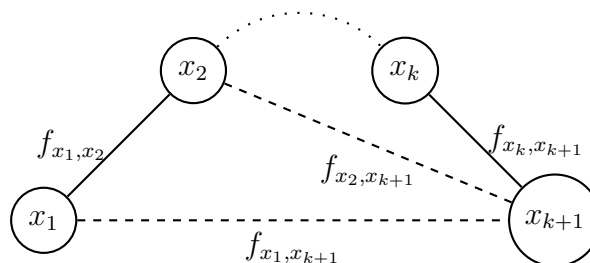
1. $z \in A$

Τότε η τομή (A, \bar{A}) χωρίζει τις κορυφές z και y . Αφού ισχύει αυτό, θα πρέπει να ισχύει ότι η ελάχιστη τομή $z-y$ (που αριθμητικά είναι ίση με f_{zy}), είναι μικρότερη ή ίση από την τομή f_{xy} , δηλαδή $f_{zy} \leq f_{xy}$. Όμως, έχουμε υποθέσει ότι $f_{xy} < \min [f_{xz}, f_{zy}]$, άτοπο.

2. $z \in \bar{A}$

Αντίστοιχα

Έχοντας δείξει αυτή την πρόταση, μπορούμε επαγωγικά να δείξουμε ότι η ανισότητα ισχύει για x_1, x_2, \dots, x_k όπου οι κορυφές x_1, x_2, \dots, x_k σχηματίζουν ένα μονοπάτι ανάμεσα στα x_1 και x_k . Έστω ότι η ανισότητα ισχύει για $f_{x_1, x_k} \geq \min [f_{x_1, x_2}, f_{x_2, x_3}, \dots, f_{x_{k-1}, x_k}]$, δηλαδή



Σχήμα 1.2: Ροή σε ένα μονοπάτι $x_1, x_2, \dots, x_k, x_{k+1}$

για ένα μονοπάτι μεγέθους $k - 1$. Αρκεί να δείξουμε ότι ισχύει και για $x_1, x_2, x_3, \dots, x_k, x_{k+1}$.

Έχουμε ότι ισχύει $f_{x_2, x_{k+1}} \geq \min [f_{x_2, x_3}, \dots, f_{x_k, x_{k+1}}]$, αφού το x_2, \dots, x_{k+1} είναι μονοπάτι μήκους $k - 1$. Επίσης έχουμε $f_{x_1, x_{k+1}} \geq \min [f_{x_1, x_2}, f_{x_2, x_{k+1}}]$. Έτσι έχουμε $f_{x_1, x_k} \geq \min [f_{x_1, x_2}, f_{x_2, x_3}, \dots, f_{x_{k-1}, x_k}]$. □

Προτού προχωρήσουμε με την απόδειξη ορθότητας των Gomory-Hu δέντρων, θα εισάγουμε την έννοια των *μέγιστων επικαλυπτόντων δέντρων* (*maximal spanning trees*) ενός γραφήματος G και θα αποδείξουμε μια σημαντική ιδιότητα τους. Θυμίζουμε ότι το επικαλύπτον δέντρο ενός γραφήματος $G = (V, E)$ είναι ένα δέντρο $T = (V, E')$ με $E' \subseteq E$, όπου περιέχει όλες τις κορυφές του G και ένα υποσύνολο των ακμών του, έτσι ώστε το T να είναι δέντρο, δηλαδή συνεκτικό και άκυκλο. Αν έχουμε μια συνάρτηση βάρους $w : E \mapsto \mathbb{R}_+$, λέμε ότι το επικαλύπτον δέντρο T έχει "τιμή" ίση με $\sum_{e \in E'} w(e)$.

Ορισμός 7 (Μέγιστο επικαλύπτον δέντρο). Μέγιστο επικαλύπτον δέντρο $T = (V, E')$ του $G = (V, E)$ ονομάζουμε το επικαλύπτον δέντρο T που μεγιστοποιεί την τιμή του δέντρου, δηλαδή το άθροισμα των βαρών των ακμών του.

Μπορούμε να συνοπτικά να αποδείξουμε την εξής ιδιότητα των μέγιστων επικαλυπτόντων δέντρων

Θεώρημα 2. [11] Έστω ένα μη κατευθυνόμενο γράφημα $G = (V, E)$ με συνάρτηση βάρους $w : E \mapsto \mathbb{R}_+$ και ένα μεγιστικό επικαλύπτον δέντρο T του G . Για κάθε ζευγάρι κορυφών s, t του δέντρου, όπου η ακμή (s, t) υπάρχει στο γράφημα G , αλλά δεν υπάρχει στο T , ισχύει η ιδιότητα:

$$w((s, t)) \leq \min [w((s, i)), w((i, j)), \dots, w((k, t))] \quad (1.1)$$

όπου $(s, i), (i, j), \dots, (k, t)$ είναι οι ακμές στο μονοπάτι από το s στο t στο T .

Απόδειξη. Έστω ότι δεν ισχύει η ανισότητα αυτή για το μονοπάτι δύο κορυφών s, t . Υπάρχει, δηλαδή, μια ακμή (u, v) η οποία έχει μικρότερη τιμή βάρους από την ακμή (s, t) , δηλαδή $w((u, v)) < w((s, t))$. Αν ισχύει αυτό, τότε μπορούμε να δημιουργήσουμε ένα δέντρο T' με μεγαλύτερη τιμή από το δέντρο T , αντικαθιστώντας ακμή (u, v) με την ακμή (s, t) στο νέο δέντρο. Αλλά το T είναι μέγιστο, που μας οδηγεί σε άτοπο. Επομένως, ισχύει η ανισότητα του θεωρήματος. \square

Λήμμα 2. [11] Αν μια μη αρνητική, συμμετρική συνάρτηση f ικανοποιεί τη σχέση $f_{xy} \geq \min [f_{xz}, f_{zy}]$, $\forall x, y, z \in V$, τότε υπάρχει μη κατευθυνόμενο γράφημα με συνάρτηση ροής f .

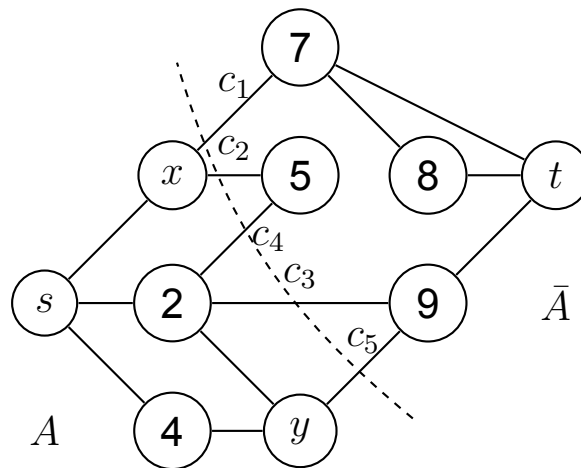
Απόδειξη. Έστω ότι έχουμε ένα γράφημα G στο οποίο κάθε ακμή (x, y) τη συσχετίζουμε με την τιμή $f(x, y)$ και έστω ότι έχουμε ένα μέγιστο επικαλύπτον δέντρο T του γραφήματος G . Λόγω της σχέσης της περιγραφής και της Σχέσης (1.1), έχουμε ως αποτέλεσμα, ότι

$$f(x_1, x_k) = \min [f(x_1, x_2), f(x_2, x_3), \dots, f(x_{k-1}, x_k)]$$

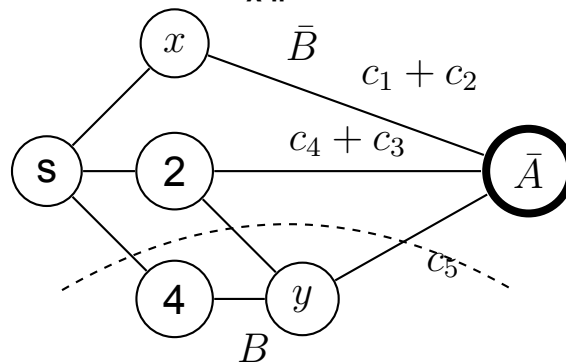
για το μονοπάτι x_1, x_2, \dots, x_k από το x_1 μέχρι το x_k . Έτσι μπορούμε να δούμε ότι, για το δέντρο με χωρητικότητα $c(x, y) = f(x, y)$ για κάθε ακμή μέσα στο δέντρο T , ισχύει ότι έχει συνάρτηση ροής f . Άρα το δέντρο που δημιουργήσαμε είναι το γράφημα που μας ζητείται. \square

Από το Λήμμα 2 όμως παρατηρούμε ότι όλες οι διαφορετικές τιμές που μπορεί να πάρει η συνάρτηση f είναι $n - 1$, όσες και οι ακμές του T . Αυτή είναι η πρώτη ένδειξη για την ύπαρξη των δέντρων Gomory-Hu, διότι υπονοείται ότι μπορεί να χρειαζόμαστε λιγότερες από $\binom{n}{2}$ υπορουτίνες μέγιστης ροής για να βρούμε τη μέγιστη ροή για όλα τα ζευγάρια κορυφών.

Υποθέτουμε ότι έχουμε ένα μη κατευθυνόμενο γράφημα G για το οποίο υπολογίζουμε την ελάχιστη τομή $C = (A, \bar{A})$ ανάμεσα σε δύο κορυφές s, t με $s \in A, t \in \bar{A}$ (Σχήμα 1.3). Κατασκευάζουμε ένα διαφορετικό γράφημα G' , στο οποίο αντικαθιστούμε όλες τις κορυφές



Σχήμα 1.3



Σχήμα 1.4: Σύμφωνα με το Θεώρημα 3, η ελάχιστη τομή $x-y$ στο γράφημα με τις συνθλιμμένες κορυφές είναι ίση με την τομή στο αρχικό γράφημα 1.3

του \bar{A} με μια κορυφή \bar{A} , δηλαδή συνθλίβουμε (contract) το σύνολο \bar{A} σε μια κορυφή στην οποία καταλήγουν όλες οι ακμές της τομής C (Σχήμα 1.4). Οι παράλληλες ακμές που προκύπτουν μπορούν να αντικατασταθούν με μία ακμή με χωρητικότητα ίση με το άθροισμα της χωρητικότητας των παράλληλων ακμών. Σε αυτό το γράφημα θα αποδείξουμε, ότι η μέγιστη ροή ανάμεσα σε δύο οποιοσδήποτε κορυφές u, v (εκτός του \bar{A}) είναι ίση με τη ροή τους στον αρχικό γράφημα.

Λήμμα 3. [11] Έστω μια ελάχιστη τομή $C = (A, \bar{A})$ του μη κατευθυνόμενου γραφήματος G που χωρίζει τις κορυφές $s-t$ με $s \in A, t \in \bar{A}$. Έστω δύο κορυφές $u, v \in A$ και το γράφημα G' στο οποίο έχουμε συνθλίψει το σύνολο \bar{A} . Η μέγιστη ροή ανάμεσα στις δύο κορυφές $u-v$ στο γράφημα G και το G' είναι ίσες μεταξύ τους.

Απόδειξη. Ορίζουμε τα σύνολα

$$\begin{aligned} X &= A \cap B \\ \bar{X} &= A \cap \bar{B} \\ Y &= \bar{A} \cap B \\ \bar{Y} &= \bar{A} \cap \bar{B} \end{aligned}$$

Με (A, \bar{A}) την ελάχιστη τομή για τις κορυφές $s \in A, t \in \bar{A}$ και (B, \bar{B}) η ελάχιστη τομή για τις κορυφές $u \in B, v \in \bar{B}$. Παρατηρούμε ότι τα σύνολα κορυφών X, Y, \bar{X}, \bar{Y} είναι ξένα μεταξύ τους ανά ζευγάρια (η τομή τους είναι ίση με το κενό σύνολο). Θα συμβολίσουμε ως b_{MN} τη συνολική χωρητικότητα των ακμών μεταξύ των συνόλων κορυφών M και N , δηλαδή $b_{MN} = \sum_{a \in M, b \in N} w_{ab}$

ΧβτΓ υποθέτουμε ότι $u \in X, v \in \bar{X}, s \in X$.

1. $t \in Y$. Έτσι έχουμε:

$$\begin{aligned} b_{A\bar{A}} &= b_{XY} + b_{X\bar{Y}} + b_{\bar{X}Y} + b_{\bar{X}\bar{Y}} \\ b_{B\bar{B}} &= b_{X\bar{X}} + b_{X\bar{Y}} + b_{\bar{X}Y} + b_{Y\bar{Y}} \end{aligned}$$

διότι τα σύνολα είναι ξένα μεταξύ τους. (B, \bar{B}) είναι μια ελάχιστη τομή που χωρίζει τα u και v , και αφού $(X \cup Y \cup \bar{Y}, \bar{X})$ χωρίζει u και v , η χωρητικότητά των ακμών της ελάχιστης τομής είναι μικρότερη ή ίση από τη χωρητικότητά οποιασδήποτε άλλης τομής. Αντίστοιχα για τα $s-t$ έχουμε ως ελάχιστη τομή (A, \bar{A}) και $(X \cup \bar{X} \cup \bar{Y}, Y)$ ως τομή που τα χωρίζει. Επομένως, έχουμε:

$$\begin{aligned} b_{B\bar{B}} - b_{X \cup Y \cup \bar{Y}, \bar{X}} &\leq 0 \implies b_{B\bar{B}} - (b_{X\bar{X}} + b_{Y\bar{X}} + b_{\bar{Y}\bar{X}}) \leq 0 \\ &\implies b_{X\bar{Y}} + b_{Y\bar{Y}} - b_{\bar{X}\bar{Y}} \leq 0 \\ b_{A\bar{A}} - b_{X \cup \bar{X} \cup \bar{Y}, Y} &\leq 0 \implies b_{A\bar{A}} - (b_{XY} + b_{\bar{X}Y} + b_{\bar{Y}Y}) \leq 0 \\ &\implies b_{X\bar{Y}} + b_{\bar{X}\bar{Y}} - b_{Y\bar{Y}} \leq 0 \end{aligned}$$

Τότε, από τις δύο σχέσεις, έχουμε ότι $b_{X\bar{Y}} \leq 0 \implies b_{X\bar{Y}} = 0$, επομένως $b_{\bar{X}\bar{Y}} - b_{Y\bar{Y}} = 0$, έτσι η πρώτη ανισότητα είναι ισότητα, επομένως η τομή $(X \cup Y \cup \bar{Y}, \bar{X}) = (X \cup \bar{A}, \bar{X})$ είναι επίσης μια ελάχιστη τομή $u-v$.

2. $t \in \bar{Y}$. Έχουμε:

$$\begin{aligned} b_{B\bar{B}} - b_{X, \bar{X} \cup \bar{Y} \cup Y} &= b_{\bar{X}Y} + b_{Y\bar{Y}} - b_{XY} \leq 0 \\ b_{A\bar{A}} - b_{\bar{Y}, Y \cup \bar{X} \cup X} &= b_{XY} - b_{\bar{X}Y} - b_{Y\bar{Y}} \leq 0 \end{aligned}$$

Ομοίως, αποδεικνύουμε ότι $(X, \bar{X} \cup \bar{A})$ είναι μια ελάχιστη τομή $u-v$

Έτσι βλέπουμε ότι υπάρχει πάντα μια ελάχιστη τομή $u-v$, τέτοια ώστε το σύνολο \bar{A} να βρίσκεται στο ένα μέρος της τομής, επομένως μπορούμε να συνθλίψουμε το \bar{A} σε ένα κόμβο χωρίς να αλλάξει η μέγιστη ροή ανάμεσα στα $u-v$. \square

Λήμμα 4. [11] Η ροή ανάμεσα σε δύο κορυφές s, t είναι

$$\min [v_{i_1}, v_{i_2}, \dots, v_{i_p}]$$

όπου v_i είναι οι τιμές των ακμών στο μονοπάτι ανάμεσα στις κορυφές $s-t$ στο δέντρο Gomory-Hu.

Περιγράφουμε συνοπτικά τον αλγόριθμο για την κατασκευή ενός Gomory-Hu δέντρου \mathcal{T} δεδομένου ενός μη κατευθυνόμενου γραφήματος $G = (V, E)$. Αρχικοποιούμε το δέντρο \mathcal{T} ως μία κορυφή που περιέχει όλες τις κορυφές V . Σε κάθε βήμα του αλγόριθμου επιλέγουμε μια κορυφή S από το δέντρο \mathcal{T} , όπου το S περιέχει πάνω από μία κορυφές του αρχικού γραφήματος G , και επιλέγουμε δύο κορυφές $s, t \in S$, αυθαίρετα. Στη συνέχεια δημιουργούμε ένα γράφημα $G(S)$ με τις εξής ιδιότητες:

1. Το $G(S)$ περιέχει όλες τις κορυφές του S .

2. Έστω όλες οι γειτονικές κορυφές του S στο δέντρο \mathcal{T} , N_1, \dots, N_h . Θεωρούμε την κορυφή S ως ρίζα του \mathcal{T} και παρατηρούμε ότι σχηματίζονται h υποδέντρα κάτω από το S , τα $\mathcal{T}_1, \dots, \mathcal{T}_h$ με ρίζες τις κορυφές N_1, \dots, N_h . Στο $G(S)$ κάθε υποδέντρο \mathcal{T}_i το συνθλίβουμε σε μια κορυφή T_i .

Για παράδειγμα, στο δέντρο στο Σχήμα 1.13, αν υποθέσουμε ότι η κορυφή S είναι η κορυφή gf τότε $N_1 = e$ και $N_2 = ca$, $\mathcal{T}_1 = e - d - b$ και $\mathcal{T}_2 = ca$, και θα διαμορφωθούν οι κορυφές $T_1 = edb$ και $T_2 = ca$.

3. Οι ακμές του $G(S)$ είναι:

- Οι ακμές έχουν τα δύο άκρα τους σε κορυφές του S στο αρχικό γράφημα.
- Οι ακμές που ενώνουν τις κορυφές του S με τις κορυφές T_i . Δηλαδή για κάθε ακμή ανάμεσα σε μια κορυφή $v \in S$ και $\forall u \in \mathcal{T}_i$, βάζουμε στο $G(S)$ μια ακμή $(v, T_{i,v})$ με χωρητικότητα $= \sum_{u \in \mathcal{T}_i} c(u, v)$. Αν μια κορυφή $z \in S, z \neq v$ δεν έχει κοινή ακμή με καμία κορυφή του \mathcal{T}_i , τότε δεν ενώνουμε τα \mathcal{T}_i με τη z στο $G(S)$.

Σε αυτό το νέο γράφημα υπολογίζουμε τη μέγιστη ροή $s-t$. Θα δείξουμε στη συνέχεια ότι η ελάχιστη τομή $s-t$ σε αυτό το γράφημα, θα είναι ελάχιστη και στο αρχικό γράφημα G . Η τομή αυτή χωρίζει το γράφημα $G(S)$ σε δύο μέρη, καθώς και το σύνολο S στα S_1, S_2 με $s \in S_1, t \in S_2$. Μετατρέπουμε το δέντρο \mathcal{T} ως εξής:

- Χωρίζουμε την κορυφή S σε S_1 και S_2 , όπου η κορυφή S_1 έχει τις κορυφές του S που είναι από τη μεριά του s στην τομή και οι υπόλοιπες στο S_2 , και ενώνουμε τα S_1 και S_2 με μια ακμή ίση με το μέγεθος της τομής που υπολογίσαμε για τα $s-t$.
- Ενώνουμε τις νέες κορυφές S_1, S_2 μόνο με τις κορυφές του \mathcal{T} με τις οποίες συνόρευε η κορυφή S στο \mathcal{T} πριν την τομή, χρησιμοποιώντας την ίδια χωρητικότητα της ακμής που τις ένωνε. Όσες κορυφές είναι από τη μεριά του s στην τομή του G_S , τις ενώνουμε με το S_1 στο \mathcal{T} , ενώ όσες είναι από τη μεριά του t , με το S_2 .

Ο αλγόριθμος τερματίζει όταν όλες οι κορυφές του δέντρου \mathcal{T} περιέχουν μόνο μία κορυφή του γραφήματος G . Από αυτή την περιγραφή του αλγορίθμου μπορούμε να κάνουμε δύο παρατηρήσεις:

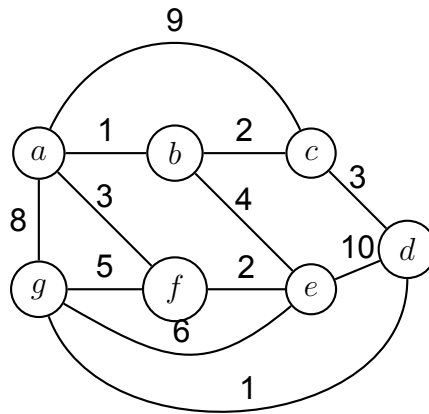
1. Στον αλγόριθμο που περιγράψαμε πιο πάνω τα s και t θα καταλήξουν να ενώνονται με μια ακμή στο \mathcal{T} , το δέντρο Gomory-Hu που κατασκευάσαμε για το G . Αλλά φαίνεται από την περιγραφή ότι δεν είναι απαραίτητο ότι οι s, t ενώνονται στο G . Αυτό σημαίνει ότι το \mathcal{T} δεν είναι επικαλύπτον δέντρο του G .
2. Ο αλγόριθμος έχει έναν γύρο για κάθε ακμή του \mathcal{T} . Έτσι βλέπουμε ότι ο αλγόριθμος έχει $n - 1$ βήματα συνολικά.

Ακολουθεί ένα παράδειγμα κατασκευής ενός δέντρου Gomory-Hu στο γράφημα G .

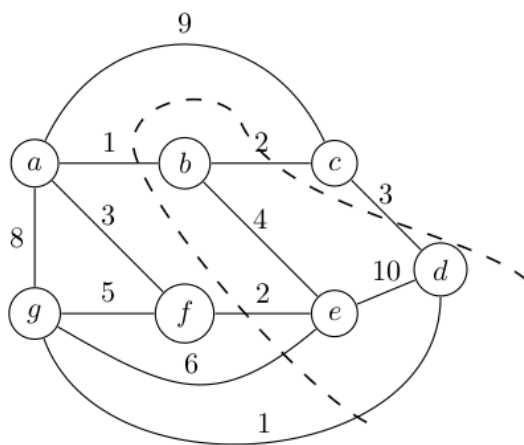
Έστω το γράφημα G στο Σχήμα 1.5.

Επιλέγουμε αυθαίρετα από το αρχικό γράφημα δύο κορυφές e και a και βρίσκουμε την ελάχιστη τομή $e-a$, η οποία είναι ίση με 15. Στο δέντρο Gomory-Hu που κατασκευάζουμε θα χωρίσουμε τα δύο σύνολα τομής σε δύο κορυφές και θα τα ενώσουμε με μια ακμή ίση με την τομή $e-a$.

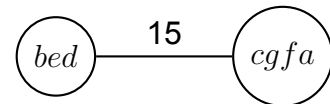
Στο νέο δέντρο Gomory-Hu επιλέγουμε αυθαίρετα μία από τις δύο συνθλιμμένες κορυφές, έστω την bed , για να τη χωρίσουμε περαιτέρω. Δημιουργούμε ένα γράφημα με όλες τις



Σχήμα 1.5: Το αρχικό γράφημα G



Σχήμα 1.6: Η τομή στο αρχικό γράφημα G

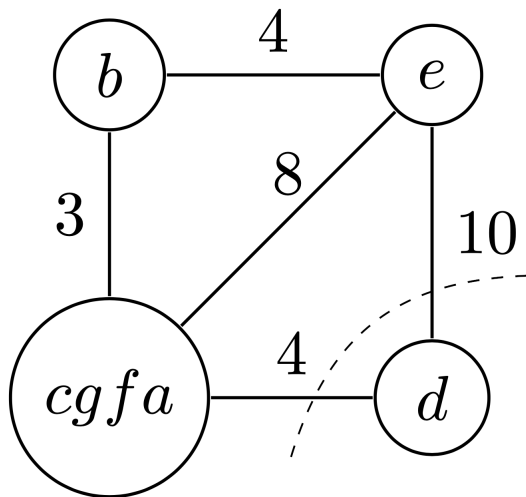


Σχήμα 1.7: Το Gomory-Hu δέντρο μετά τη πρώτη τομή $e-a$

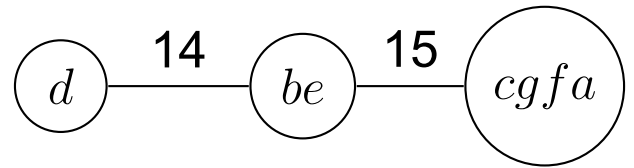
κορυφές του bed , b , e και d , και την κορυφή $cgfa$ ως μία κορυφή. Οι ακμές ανάμεσα στα b , e και d μένουν όπως στο αρχικό γράφημα, ενώ ενώνουμε την $cgfa$ με τις υπόλοιπες κορυφές όπως περιγράψαμε παραπάνω. Σε αυτό το γράφημα, 1.8, υπολογίζουμε την ελάχιστη τομή ανάμεσα σε δύο κορυφές από το bed , τις οποίες επιλέγουμε αυθαίρετα πάλι. Έστω ότι επιλέγουμε τις b και d , έτσι σχηματίζεται η τομή η οποία φαίνεται στο Σχήμα 1.8. Τα be βρίσκονται στο ίδιο μέρος της τομής άρα στο Gomory-Hu δέντρο θα είναι στην ίδια κορυφή και η κορυφή $cgfa$ είναι στο ίδιο μέρος της τομής με τα be επομένως η ακμή με βάρος 15 θα ενώνει την $cgfa$ με την be , όπως φαίνεται στο 1.9

Καλούμαστε τώρα να επιλέξουμε μία από τις κορυφές του δέντρου Gomory-Hu για να τη χωρίσουμε. Η κορυφή d δεν μπορεί να επιλεγεί καθώς αποτελείται από 1 μόνο κορυφή και έτσι απομένουν οι be και $cgfa$. Αυθαίρετα επιλέγουμε την κορυφή be και διαμορφώνουμε κατάλληλα το νέο γράφημα στο οποίο θα τρέξουμε τον υπολογισμό ελάχιστης τομής ανάμεσα στις κορυφές b και e . Παρατηρούμε ότι το γράφημα 1.10 είναι διαφορετικό από το προηγούμενο γράφημα που τρέξαμε τον υπολογισμό παρόλο που έχουν τις ίδιες κορυφές. Αυτό ισχύει διότι η κορυφή d δεν ενώνεται με την be στο δέντρο 1.9, επομένως δεν μπορούμε στο γράφημα ενώσουμε τις be με την d .

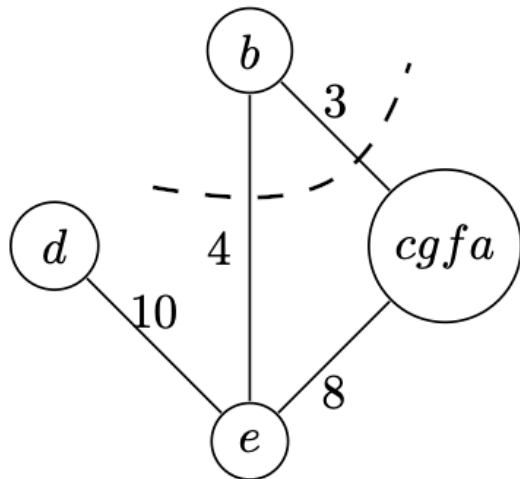
Στο Gomory-Hu δέντρο που έχει σχηματιστεί, όπως φαίνεται στο Σχήμα 1.11, τώρα έχει μείνει μόνο μια κορυφή που αποτελείται από περισσότερες από μία κορυφές του γραφήματος. Οι κορυφές εκτός της κορυφής $cgfa$ θα συνθλιφτούν κατάλληλα σε μία κορυφή.



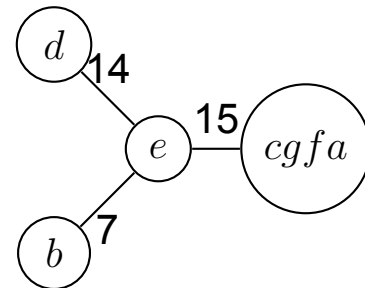
Σχήμα 1.8: Η τομή που χωρίζει τα b και d



Σχήμα 1.9: Το Gomory-Hu δέντρο μετά την τομή $b-d$



Σχήμα 1.10: Η τομή που χωρίζει τα b και e



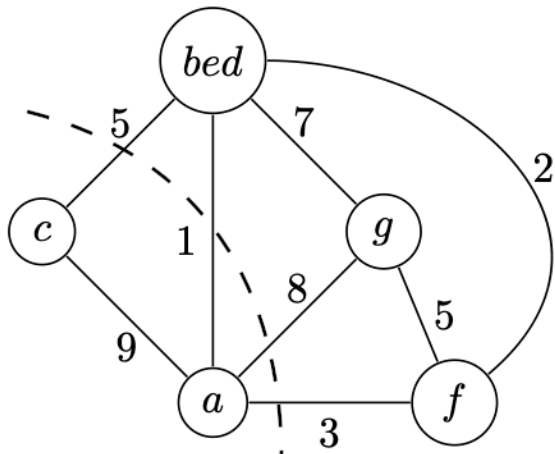
Σχήμα 1.11: Το Gomory-Hu δέντρο μετά την τομή $b-e$

Με την ίδια λογική που περιγράψαμε πιο πάνω θα χωρίσουμε στο Gomory-Hu δέντρο την $cgfa$ στις κορυφές gf και ca και θα τις ενώσουμε με μια ακμή με βάρος ίσο με την τομή $a-g$ που βρήκαμε.

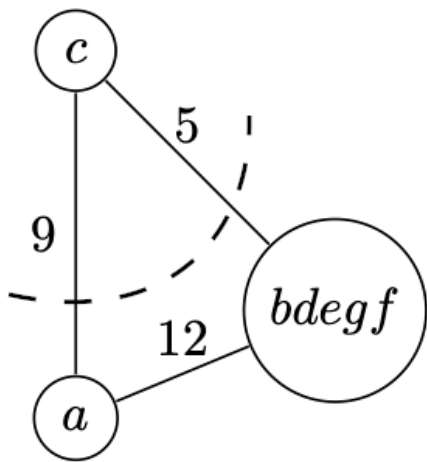
Στο Σχήμα 1.14 βλέπουμε ότι οι κορυφές d, e, b και gf συνθλιβονται σε μία κορυφή. Θα μπορούσαμε να αφήσουμε τις κορυφές αυτές να συνδέονται όπως στο Σχήμα 1.13 και οι κορυφές c και a να συνδέονται μόνο με την κορυφή gf . Τα δύο αυτά γραφήματα θα ήταν ισοδύναμα ως προς την ελάχιστη τομή των κορυφών c και a , επομένως επιλέγουμε να συνθλίψουμε τις κορυφές για να δημιουργήσουμε ένα πιο απλό γράφημα.

Επιλέγουμε την τελευταία κορυφή του δέντρου, την gf , για να τη χωρίσουμε. Συνθλιβουμε τις κορυφές bed και ac στο γράφημα και τις ενώνουμε κατάλληλα με τις g και f . Η κορυφή f αποκόπτεται από όλες τις κορυφές με την τομή στο Σχήμα 1.16, επομένως ενώνεται μόνο με την g με μια κορυφή ίση με την τομή που βρήκαμε, ενώ η g μένει συνδεδεμένη με τις κορυφές e και a . Είναι ενδιαφέρον να παρατηρήσουμε ότι η ακμή (g, c) που σχηματίστηκε στο γράφημα 1.17 δεν υπάρχει στο αρχικό γράφημα, επομένως το δέντρο Gomory-Hu του G δεν είναι υπογράφημα του G .

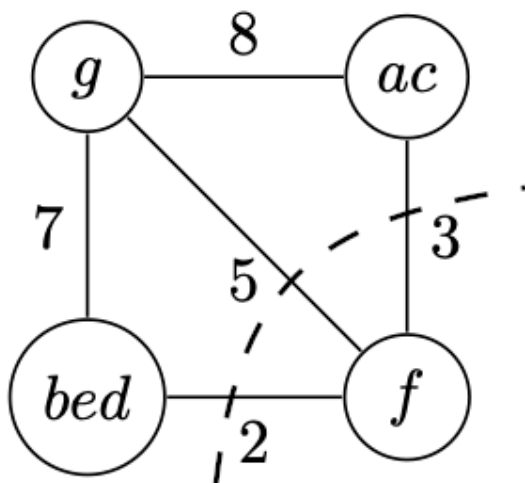
Στο Σχήμα 1.17 βλέπουμε το τελικό δέντρο Gomory-Hu του G .



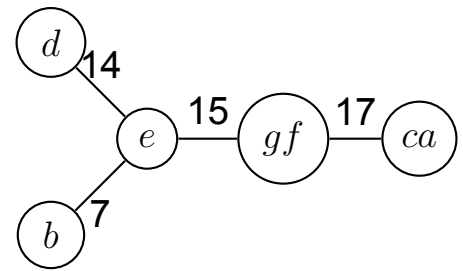
Σχήμα 1.12: Η τομή που χωρίζει τα a και g



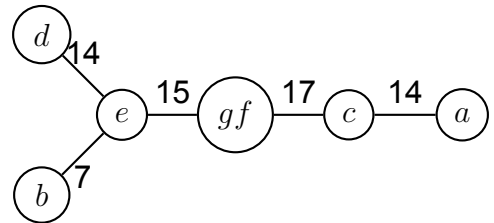
Σχήμα 1.14: Η τομή που χωρίζει τα c και a



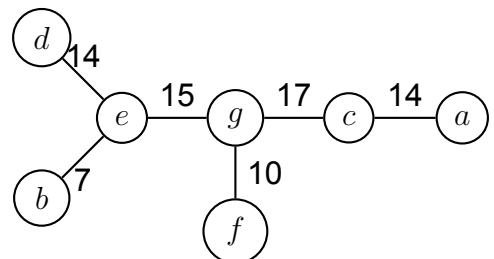
Σχήμα 1.16: Η τομή που χωρίζει τα f και g



Σχήμα 1.13: Το Gomory-Hu δέντρο μετά την τομή $a-g$



Σχήμα 1.15: Το Gomory-Hu δέντρο μετά την τομή $c-a$



Σχήμα 1.17: Το τελικό Gomory-Hu δέντρο

Θα αποδείξουμε τώρα το Λήμμα 4, με το οποίο ολοκληρώνεται η απόδειξη ορθότητας της κατασκευής των Gomory-Hu δέντρων. Θυμίζουμε ότι σκοπός μας είναι να αποδείξουμε ότι η ροή ανάμεσα σε δύο κορυφές s, t στο γράφημα G είναι ίση την τιμή

$$\min [v_{i_1}, v_{i_2}, \dots, v_{i_p}]$$

στο Gomory-Hu δέντρο.

Απόδειξη. Ξεκινάμε με την παρατήρηση ότι κάθε ακμή στο μονοπάτι ανάμεσα στις κορυφές s, t στο δέντρο Gomory-Hu αναπαριστά μια τομή που χωρίζει τις κορυφές s, t . Επομένως:

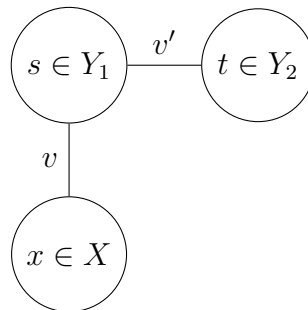
$$f_{st} \leq \min [v_{i_1}, v_{i_2}, \dots, v_{i_p}]$$

αφού η ροή ανάμεσα σε δύο κορυφές φράζεται από οποιαδήποτε τομή που χωρίζει αυτές (λόγω 1)

Αρκεί να αποδείξουμε ότι $f_{st} \geq \min [v_{i_1}, v_{i_2}, \dots, v_{i_p}]$. Αρχικά θα αποδείξουμε ότι, σε κάθε στάδιο κατασκευής του δέντρου Gomory-Hu, για οποιαδήποτε ακμή με τιμή v και άκρα τα σύνολα X και Y υπάρχουν κορυφές στα X και Y όπου η ροή ανάμεσα τους είναι v .

Είναι εμφανές ότι αυτό ισχύει για την πρώτη τομή, όπου χωρίζουμε το V σε δύο σύνολα, καθώς η τιμή της ακμής είναι η ελάχιστη τομή (άρα και μέγιστη ροή) ανάμεσα σε δύο κορυφές στα σύνολα αυτά. Έστω ότι χωρίζουμε το γράφημα σε δύο σύνολα X και Y με $x \in X, y \in Y$ και έστω $f(x, y) = v$.

Στο επόμενο βήμα χωρίζουμε το σύνολο το Y περαιτέρω, με την ελάχιστη τομή $s-t$ (Y_1, Y_2) και έστω ότι Y_1 συνδέεται με το X . Έχουμε τα $x \in X, s \in Y_1$ και $t \in Y_2$. Για την ακμή $f(s, t) = v'$, τα s και t είναι οι δύο κόμβοι που έχουν την αντίστοιχη ροή.



Σχήμα 1.18

Αρκεί τώρα να βρούμε το ζευγάρι κορυφών στα σύνολα X και Y_1 , όπου έχουν μέγιστη ροή μεταξύ τους ίση με v , όπως φαίνεται στο Σχήμα 1.18. Παίρνουμε τις εξής δύο περιπτώσεις:

1. $y \in Y_1$

$x \in X$ και $y \in Y_1$, επομένως τα x και y είναι το ζευγάρι κορυφών που αναζητάμε.

2. $y \in Y_2$

Παρατηρούμε ότι στη μία μεριά της τομής $s-t$ με τιμή v' βρίσκονται οι κορυφές s, x ενώ στην άλλη μεριά οι κορυφές t, y . Έτσι, λόγω του Λήμματος 3, ξέρουμε ότι αν στο G συνθλίψουμε τις κορυφές του συνόλου Y_2 , τότε η ελάχιστη τομή $x-s$ ανάμεσα στις κορυφές δεν επηρεάζεται. Έτσι, ακόμα και να θέσουμε τη χωρητικότητα όλων των ακμών στο Y_2 σε ∞ , η ροή f_{xs} δεν επηρεάζεται.

Έτσι, για τη ροή f' στο νέο γράφημα στο οποίο έχουμε κάνει την τροποποίηση αυτή, έχουμε:

$$\begin{aligned} f'(x, s) &= f(x, s) \\ f'(x, y) &\geq f(x, y) = v \\ f'(y, t) &= \infty \\ f'(t, s) &\geq f(t, s) = v' \end{aligned}$$

Η ανισότητα ισχύει διότι $f'(x, y) \geq f(x, y)$ η ροή ανάμεσα στα x και y μπορεί να έχει επηρεαστεί ή όχι από την αλλαγή της χωρητικότητας στις ακμές εντός του Y_2 , αλλά η αύξηση της χωρητικότητας δεν μπορεί να έχει επιφέρει τη μείωση της ροής ανάμεσα στα x και y . Έτσι έχουμε, λόγω της τριγωνικής ανισότητας του Λήμματος 1:

$$\begin{aligned} f'(x, s) &\geq \min [f'(x, y), f'(y, t), f'(t, s)] \\ &= \min [f'(x, y), f'(t, s)] \quad \text{αφού } f'(y, t) = \infty \end{aligned}$$

Χρησιμοποιώντας τις 4 ανισότητες παραπάνω:

$$f(x, s) = f'(x, s) \geq \min [v, v']$$

Η τομή v' χωρίζει τα x και y . Η τομή v είναι η ελάχιστη τομή x - y επομένως $v' \geq v \implies v = \min [v, v']$. Άρα, $f(x, s) \geq v$.

Η τομή v χωρίζει τα x και s , άρα $f(x, s) \leq v$, επομένως $f(x, s) = v$. Άρα x, s είναι οι δύο κόμβοι που αναζητάμε.

Έτσι, έχουμε δείξει ότι η συνθήκη που θέσαμε ισχύει σε κάθε βήμα της κατασκευής· τα άκρα κάθε ακμής περιέχουν ένα ζευγάρι κορυφών που έχουν μέγιστη ροή ίση με την τιμή της ακμής στο δέντρο. Αυτό ισχύει και μετά το πέρας του τελευταίου βήματος, όπου όλες οι κορυφές αντιπροσωπεύουν μια κορυφή στο αρχικό γράφημα. Δηλαδή, κάθε ακμή στο τελικό δέντρο Gomory-Hu, έχει τιμή ίση με τη μέγιστη ροή των δύο άκρων της. Άρα, εφαρμόζοντας την ανισότητα από το Λήμμα 1 στο τελικό δέντρο έχουμε:

$$f_{st} \geq \min [v_{i_1}, v_{i_2}, \dots, v_{i_p}]$$

Έχοντας δείξει και την αντίθετη ανισότητα, τελικά έχουμε ότι στο τελικό μας δέντρο ισχύει:

$$f_{st} = \min [v_{i_1}, v_{i_2}, \dots, v_{i_p}]$$

□

Έτσι, ολοκληρώνουμε την απόδειξη ορθότητας της κατασκευής των δέντρων Gomory-Hu.

2. ΑΠΟΔΟΤΙΚΟΙ ΑΛΓΟΡΙΘΜΟΙ ΑΚΜΟΣΥΝΕΚΤΙΚΟΤΗΤΑΣ ΚΑΙ ΚΑΤΑΣΚΕΥΗΣ ΔΕΝΤΡΩΝ GOMORY-HU

Στο παρόν κεφάλαιο, περιγράφουμε τον αλγόριθμο εύρεσης ακμοσυνεκτικότητας σε μη κατευθυνόμενο γράφημα του Gabow [10], και βασισμένοι σε αυτόν θα παρουσιάσουμε τον αλγόριθμο εύρεσης τομής Steiner σε μη κατευθυνόμενο γράφημα των Bhalgat et al. [12]. Μετά την ανάλυση των αλγορίθμων, θα παρουσιάσουμε πως ο αλγόριθμος των Bhalgat et al. μπορεί να χρησιμοποιηθεί ως μια αποδοτική υποροουτίνα για κατασκευή Gomory-Hu δέντρων [12].

Και οι δύο αλγόριθμοι βασίζονται σε τεχνικές πακεταρίσματος δέντρων. *Πακετάρισμα δέντρων (tree packing)* ονομάζουμε το πρόβλημα ανάλυσης ενός γραφήματος σε δέντρα T_1, T_2, \dots, T_k όπου για τα δέντρα $T_j = (V, E_j)$ ισχύει $E_j \subset E, 1 \leq j \leq k$ και τα σύνολα ακμών E_j είναι ξένα μεταξύ τους ανά δύο. Οι αλγόριθμοι εφαρμόζονται σε μη κατευθυνόμενα γραφήματα, αλλά χειρίζονται τα γραφήματα ως κατευθυνόμενα. Αυτό το κάνουν αντικαθιστώντας κάθε ακμή χωρίς κατεύθυνση με δύο ακμές αντίθετων κατευθύνσεων. Η επεξεργασία αυτή του γραφήματος δεν αλλάζει την ακμοσυνεκτικότητα μεταξύ των κορυφών του γραφήματος, όπως φαίνεται από το παρακάτω Θεώρημα 3:

Θεώρημα 3. [7] Ένα κατευθυνόμενο γράφημα Euler G μπορεί να μετατραπεί σε μη κατευθυνόμενο γράφημα G' τέτοιο ώστε κάθε τομή στο G' να είναι ακριβώς η διπλάσια σε μέγεθος από την αντίστοιχη τομή στο G . Αντίστοιχα, ένα μη κατευθυνόμενο γράφημα G μπορεί να μετατραπεί σε ένα κατευθυνόμενο Euler γράφημα έτσι ώστε κάθε τομή να έχει το ίδιο μέγεθος με την αντίστοιχη τομή στο G .

Από αυτό το σημείο και μετά, οποιαδήποτε αναφορά γίνεται σε γραφήματα για τους δύο αλγόριθμους, θα αναφέρεται σε κατευθυνόμενα γραφήματα, εκτός αναφερθούμε ρητά σε μη κατευθυνόμενα γραφήματα. Είναι, επίσης, σημαντικό να σημειωθεί ότι όταν αναφερόμαστε σε δέντρα T_1, \dots, T_i σε οποιοδήποτε βήμα στους παρακάτω αλγορίθμους πακεταρίσματος, συχνά θα αναφέρουμε το τελευταίο T_i του βήματος αυτού ως δέντρο, ενώ στη πραγματικότητα θα αποτελεί δάσος. Τέλος, λόγω της μεθόδου που χρησιμοποιούν οι δύο αλγόριθμοι (πακετάρισμα δέντρων), μπορούμε να εφαρμόσουμε μια προεπεξεργασία στα μη κατευθυνόμενα γραφήματα (πριν τη μετατροπή κάθε ακμής σε δύο αντίθετων κατευθύνσεων) σύμφωνα με το θεώρημα 4 των Nagamochi και Ibaraki [19]:

Θεώρημα 4. [19] Δεδομένου ενός μη κατευθυνόμενου γραφήματος $G = (V, E)$, μπορούμε σε χρόνο $O(n+m)$ να βρούμε μια (ταξινομημένη) ακολουθία από δάση με τις εξής ιδιότητες:

1. Κάθε ακμή είναι μέσα σε κάποιο δάσος.
2. Κάθε ζευγάρι κορυφών που δεν συνδέεται σε κάποιο δάσος, το ζευγάρι αυτό δεν συνδέεται και σε κανένα άλλο επόμενο δάσος.
3. Τουλάχιστον i ακμές από κάθε τομή $\geq i$ είναι παρούσες στα πρώτα i δάση.

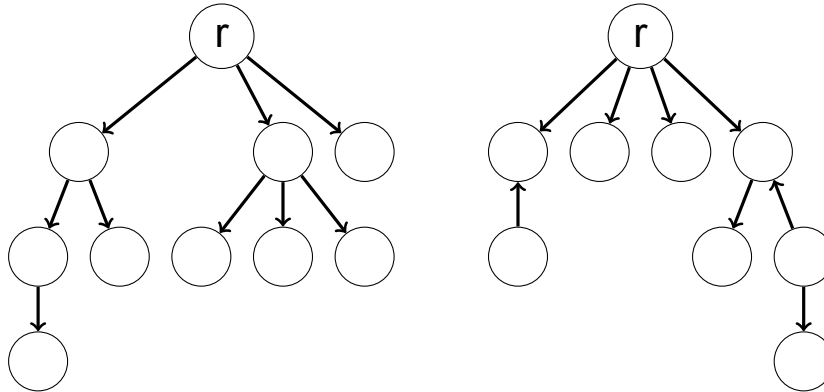
2.1 Αλγόριθμος του Gabow για ακμοσυνεκτικότητα γραφήματος

Ορίζουμε μερικές έννοιες που θα μας βοηθήσουν να κατανοήσουμε καλύτερα τον αλγόριθμο που θα παρουσιάσουμε:

Ορισμός 8. r -κατευθυνόμενο δέντρο (r -arborescence) είναι ένα κατευθυνόμενο δέντρο όπου όλες οι ακμές κατευθύνονται μακριά από τη ρίζα r

Ορισμός 9. r -επικαλύπτον δέντρο χωρίς κατεύθυνση (directionless r -spanning tree) ονομάζουμε το κατευθυνόμενο επικαλύπτον δέντρο ενός γραφήματος G , όπου η ρίζα r έχει εισερχόμενο βαθμό 0.

Στον παραπάνω Ορισμό 9, οι μόνοι μας περιορισμοί είναι η μη ύπαρξη ακμών που να καταλήγουν στη ρίζα r και ότι δεν υπάρχουν κύκλοι στο γράφημα. Η φράση "χωρίς κατεύθυνση" δηλώνει ότι δεν υπάρχει ο περιορισμός ως προς τις υπόλοιπες ακμές του γραφήματος να κατευθύνονται "μακριά" από τη ρίζα r , όπως φαίνεται στο παράδειγμα 2.1.



Σχήμα 2.1: Αριστερά: Παράδειγμα r -κατευθυνόμενου δέντρου
 Δεξιά: Παράδειγμα r -επικαλύπτοντος δέντρου χωρίς κατεύθυνση

Το πρόβλημα του πακεταρίσματος δέντρων συνδέεται με την ακμοσυνεκτικότητα ενός γραφήματος μέσω του παρακάτω θεωρήματος του Edmonds:

Θεώρημα 5 (Θεώρημα του Edmonds [9]). Η ελάχιστη τιμή c_{min} μιας r -τομής είναι ίση με τον μέγιστο αριθμό των r -επικαλυπτόντων δέντρων χωρίς κατεύθυνση με ξένα σύνολα ακμών, έτσι ώστε κάθε κορυφή $v \neq r$ να έχει συνολικό εισερχόμενο βαθμό c_{min} σε όλα τα δέντρα.

Έστω ένα κατευθυνόμενο γράφημα $G = (V, E)$. Ορίζουμε ως *αντίστροφο γράφημα* (*transpose graph*) του G και συμβολίζουμε με $G^R = (V, E^R)$ το κατευθυνόμενο γράφημα όπου έχουμε αντιστρέψει όλες τις ακμές του G , δηλαδή $E^R = \{(u, v) : (v, u) \in E\}$.

Η ελάχιστη τομή $\lambda(G)$ ενός κατευθυνόμενου γραφήματος G έχει μέγεθος ίσο με το μέγεθος της r -τομής στο αρχικό γράφημα ή στο αντίστροφο γράφημα για μια αυθαίρετα επιλεγμένη κορυφή $r \in V(G)$, δηλαδή $\lambda(G) = \min \{r\text{-con}(G), r\text{-con}(G^R)\}$, όπου $r\text{-con}(D)$ το μέγεθος της ελάχιστης r -τομής στο κατευθυνόμενο γράφημα D .

Έστω το αρχικό μας μη κατευθυνόμενο γράφημα G_u και το κατευθυνόμενο γράφημα G_d , το οποίο προκύπτει αντικαθιστώντας κάθε ακμή του G_u με δύο ακμές αντίθετης κατεύθυνσης, δηλαδή $E(G_d) = \{(u, v), (v, u) : \{u, v\} \in E(G_u)\}$.

Για το κατευθυνόμενο γράφημα G_d που έχουμε δημιουργήσει ισχύει ότι $G_d = G_d^R$, αφού για κάθε ακμή στο G_d υπάρχει η αντίστροφη της στο G_d . Επομένως στο κατευθυνόμενο γράφημα G_d , $\lambda(G_d) = r\text{-con}(G_d)$. Επίσης, λόγω του Θεωρήματος 3, ισχύει ότι $\lambda(G_d) = \lambda(G_u)$. Επομένως, μέσω του Θεωρήματος 5, μπορούμε να βρούμε την ελάχιστη τομή $\lambda(G_u)$ του αρχικού μη κατευθυνόμενου γραφήματος G_u , εφαρμόζοντας τεχνικές πακεταρίσματος στο G_d .

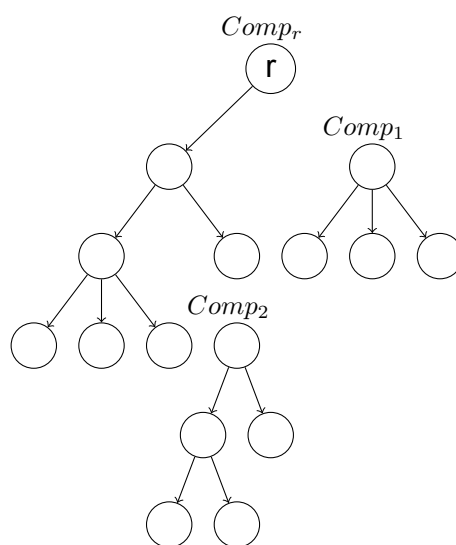
Έτσι, η ελάχιστη τομή σε ένα μη κατευθυνόμενο γράφημα είναι ίση με μια r -τομή, για μια οποιαδήποτε κορυφή r , στο κατευθυνόμενο γράφημα που έχουμε μετατρέψει κάθε μη κατευθυνόμενη ακμή σε δύο ακμές αντίθετης κατεύθυνσης.

Ο αλγόριθμος του Gabow ορίζει και προσεγγίζει το πρόβλημα του πακεταρίσματος r -επικαλυπτόντων δέντρων χωρίς κατεύθυνση μέσω της θεωρίας μητροειδών. Ορίζει έναν round-robin αλγόριθμο, ο οποίος δέχεται σαν είσοδο ένα γράφημα G , και όταν ολοκληρωθεί επιστρέφει ένα σύνολο κορυφών $S \subset V(G)$, το οποίο είναι το ένα μέρος της ελάχιστης τομής $(V(G) \setminus S, S)$.

Πιο συγκεκριμένα δίνει έναν αλγόριθμο που χτίζει r -επικαλύπτοντα δέντρα χωρίς κατεύθυνση T_1, \dots, T_i σε κάθε γύρο του αλγορίθμου, όπου το r έχει επιλεγθεί αυθαίρετα. Για την κατασκευή του i -οστού δέντρου χρησιμοποιεί τα προηγούμενα $i - 1$ δέντρα. Ένας γύρος i ξεκινάει με το δέντρο T_i να είναι το σύνολο κορυφών V χωρίς καμία ακμή ανάμεσα τους και κάθε κορυφή είναι ένα *συνεκτικό στοιχείο (component)*. Σε κάθε βήμα του γύρου αυτού, προσπαθούμε να ενώσουμε κάθε συνεκτικό στοιχείο με ένα άλλο, υποδιαιρώντας έτσι τον αριθμό των συνεκτικών στοιχείων. Βλέπουμε έτσι ότι σε κάθε γύρο, έχουμε $O(\log n)$ βήματα. Θα προσπαθήσουμε να ενώσουμε τα δέντρα "ανάποδα", δηλαδή ενώ θα επεξεργαζόμαστε το συνεκτικό στοιχείο K θα προσπαθούμε να βρούμε μια ακμή από ένα άλλο συνεκτικό στοιχείο που κατευθύνεται προς την K για να τα ενώσουμε.

Σε κάθε βήμα έχουμε τις εξής αμετάβλητες συνθήκες:

1. Ο εισερχόμενος βαθμός του r είναι 0
2. Κάθε συνεκτικό στοιχείο στο T_i έχει ακριβώς μία κορυφή v η οποία έχει εισερχόμενο βαθμό $i - 1$ σε όλα τα δέντρα T_1, \dots, T_i . Αυτή την κορυφή την ονομάζουμε *ελλιπή (deficient)* κορυφή. Μπορούμε άτυπα να αναφέρουμε την ελλιπή κορυφή ενός συνεκτικού στοιχείου ως τη ρίζα του.
3. Κάθε κορυφή που δεν είναι "ρίζα" ενός συνεκτικού στοιχείου στο T_i , έχει συνολικό εισερχόμενο βαθμό i σε όλα τα δέντρα T_1, \dots, T_i .



Σχήμα 2.2: Παράδειγμα του τελευταίου "δέντρου" T_{k+1} στον αλγόριθμο του Gabow

Όπως είπαμε, σε κάθε βήμα, διαλέγουμε ένα συνεκτικό στοιχείο (με σειρά round-robin), και προσπαθούμε να το ενώσουμε με ένα άλλο. Ισχύει ότι μια ελλιπής κορυφή έχει μια

αχρησιμοποίητη ακμή. Μια *αχρησιμοποίητη ακμή* είναι μια ακμή που δεν είναι σε κανένα δέντρο T_1, \dots, T_i και καταλήγει σε μια ελλiptή κορυφή v του T_i .

Έστω ότι χρησιμοποιούμε αυτή την αχρησιμοποίητη ακμή $e = (u, v)$, με v την ελλiptή κορυφή του συνεκτικού στοιχείου K που διαχειριζόμαστε σε αυτό το βήμα. Διακρίνουμε δύο περιπτώσεις:

1. $u \notin K$. Αν προσθέσουμε αυτή την ακμή στο T_i , τότε δεν παραβιάζουμε καμιά από τις συνθήκες που θέσαμε παραπάνω, ούτε χαλάμε τη δενδρική μορφή του T_i και ενώνουμε δύο συνεκτικά στοιχεία, επομένως προχωράμε στο επόμενο συνεκτικό στοιχείο.
2. $u \in K$. Αν το u είναι μέρος του συνεκτικού στοιχείου που χειριζόμαστε, τότε δημιουργείτε κύκλος. Είναι ενδιαφέρον να παρατηρήσουμε ότι η ακμή $u \rightarrow v$ δημιουργεί κύκλο αν τη προσθέσουμε σε οποιοδήποτε δέντρο T_1, \dots, T_i , αφού υπάρχει ήδη ένα μονοπάτι uv στα δέντρα αυτά. Ο κύκλος που δημιουργείται σε ένα δέντρο με τη προσθήκη μια ακμής e ονομάζεται *θεμελιώδες κύκλωμα (fundamental circuit)*. Μπορούμε να αφαιρέσουμε μια ακμή m που ανήκει στο θεμελιώδες κύκλωμα που δημιουργεί η προσθήκη της ακμής $u \rightarrow v$ στο δέντρο T_j , $1 \leq j \leq i$ και να την αντικαταστήσουμε με την $u \rightarrow v$. Τώρα έχουμε μια άλλη ακμή που πρέπει με κάποιον τρόπο να χρησιμοποιήσουμε έτσι ώστε να πετύχουμε το σκοπό μας, να ενώσουμε δύο συνεκτικά στοιχεία. Αν το θεμελιώδες κύκλωμα που δημιουργήθηκε ανήκει στο δέντρο T_i , τότε αυτό που έχουμε καταφέρει είναι να αλλάξουμε την ελλiptή κορυφή σε μια άλλη κορυφή του K , σε μια από τις κορυφές του θεμελιώδους κυκλώματος. Στην περίπτωση που το κύκλωμα δεν ανήκει στο δέντρο T_i , έχουμε ελευθερώσει μια άλλη ακμή, η οποία, στη συνέχεια, μέσω μιας σειράς μετασχηματισμών στα δέντρα μας, μπορεί να μας δώσει μια ακμή με την οποία θα πετύχουμε τον σκοπό μας, δηλαδή να ενώσουμε δύο συνεκτικά στοιχεία. Στην παρακάτω ενότητα, θα αναφέρουμε ποιοι είναι οι επιτρεπόμενοι μετασχηματισμοί που να εφαρμοστούν στα δέντρα με τη χρήση μιας ελεύθερης ακμής.

Ως ελεύθερη ακμή στην παραπάνω διαδικασία αναφέρουμε μια ακμή που μπορούμε να τη χρησιμοποιήσουμε είτε για να ενώσουμε δύο συνεκτικά στοιχεία, είτε σε έναν μετασχηματισμό που θα μας δώσει μια άλλη ελεύθερη ακμή. Την παραπάνω διαδικασία αναζήτησης μιας ακμής που θα ενώσει το συνεκτικό στοιχείο K με ένα άλλο, την ορίζουμε ως *υπολογισμό κλειστότητας (Closure Computation)* και δίνουμε παρακάτω περισσότερες λεπτομέρειες για αυτήν.

Υπολογισμός Κλειστότητας

Η υπορουτίνα υπολογισμού κλειστότητας δέχεται ως είσοδο το γράφημα G , το σύνολο των δέντρων T_1, \dots, T_i και το συνεκτικό στοιχείο K του δέντρου T_i και ξεκινάει των υπολογισμό ενός συνόλου ακμών M . Το σύνολο αυτό περιέχει όλες τις ακμές που μπορούμε να απελευθερώσουμε. Πιο συγκεκριμένα χρησιμοποιεί δύο δομές στον υπολογισμό αυτό:

- Ένα σύνολο M με όλες τις ακμές που μπορούν να απελευθερωθούν και αρχικοποιείται με όλες τις αχρησιμοποίητες ακμές που κατευθύνονται προς την ελλiptή κορυφή v του K .
- Ένα σύνολο C με όλες τις κορυφές των ακμών που ανήκουν στο M και το αρχικοποιείται με την ελλiptή κορυφή του K .

Έπειτα χρησιμοποιεί δύο κανόνες μετασχηματισμών για να προσθέσει και άλλες ακμές στο M .

Swap Ο μετασχηματισμός αυτός προσθέτει μια ακμή $e \in M$ σε ένα δέντρο T_j και απελευθερώνει κάθε ακμή στο θεμελιώδες κύκλωμα που δημιουργείται. Η αντικατάσταση αυτή έχει ως αποτέλεσμα να μην παραβιαστεί καμία από τις αμετάβλητες συνθήκες του αλγορίθμου του Gabow που περιγράψαμε. Έτσι, προσθέτουμε όλες τις ακμές από το θεμελιώδες κύκλωμα που δημιουργείται στο σύνολο M και τις κορυφές του στο σύνολο C .

Incidence Ο μετασχηματισμός αυτός προσθέτει κάθε αχρησιμοποίητη ακμή (w, v) για κάθε ακμή $(u, v) \in M$. Δηλαδή αντικαθιστά μια ακμή που μπορούμε να απελευθερώσουμε με μια αχρησιμοποίητη ακμή που κατευθύνεται προς την ίδια κορυφή. Εισάγουμε την (w, v) στο M και την w στο C .

Ο Gabow δείχνει ότι υπάρχουν δύο πιθανά αποτελέσματα του υπολογισμού κλειστότητας. Στη πρώτη περίπτωση, ο υπολογισμός μπορεί να λήξει επιτυχώς, με το που θα βρεθεί μια ακμή που μπορούμε να την εισάγουμε στο T_i και να ενώσει το K με κάποιο άλλο συνεκτικό στοιχείο. Αυτή η ακμή μπορεί να προστεθεί στο T_i μετά από μια σειρά μετασχηματισμών στα δέντρα, έτσι ώστε να μην παραβιάζεται καμία συνθήκη. Σε περίπτωση αποτυχίας του υπολογισμού, δηλαδή σε περίπτωση που δεν βρέθηκε ακμή που να μπορεί να ενώσει το K με ένα άλλο συνεκτικό στοιχείο, τότε ο αλγόριθμος του Gabow μας επιστρέφει την r -τομή $(V \setminus C, C)$ μεγέθους $k = i - 1$. Έως τότε έχουμε ήδη χτίσει k δέντρα, κάτι το οποίο επιβεβαιώνει το Θεώρημα 5. Η απόδειξη ότι η τομή αυτή είναι μεγέθους $k = i - 1$ βασίζεται στις τρεις παρακάτω προτάσεις:

1. Κάθε κορυφή στο C έχει εισερχόμενο βαθμό k εκτός από την ελλειπή κορυφή του K που έχει εισερχόμενο βαθμό $k - 1$ στα δέντρα T_1, \dots, T_k
2. Οι κορυφές στο C είναι όλες γειτονικές (*contiguous*) στα δέντρα T_1, \dots, T_k . Με τον όρο γειτονικές σε αυτή την περίπτωση εννοούμε ότι όλες οι κορυφές είναι απευθείας συνδεδεμένες μεταξύ τους. Δηλαδή, δεν υπάρχει δέντρο στο οποίο μια κορυφή δεν συνδέεται απευθείας με μία από τις κορυφές που σχηματίζουν γειτονιά.
3. Όλες οι ακμές που δεν είναι μέσα στα T_1, \dots, T_k , αλλά κατευθύνονται προς μια κορυφή στο C , είναι όλες μέσα στο υπογράφημα που σχηματίζουν οι κορυφές του C . Έτσι, όλες οι ακμές που κατευθύνονται προς το C από το $V \setminus C$ ανήκουν στα T_1, \dots, T_k .

Μπορούμε εδώ να σημειώσουμε ότι ο υπολογισμός κλειστότητας δεν επιφέρει καμιά αλλαγή στα δέντρα. Απλά υπολογίζει ένα σύνολο κορυφών S με την ιδιότητα ότι όλες οι αχρησιμοποίητες ακμές που κατευθύνονται προς αυτό, έχουν και τα δύο τους άκρα στο S και αν αυτό δεν μπορεί να βρεθεί, μας δίνει μέσω του M , ένα σύνολο πιθανών μετασχηματισμών που θα οδηγήσει στο να ενωθούν δύο συνεκτικά στοιχεία. Παρόλα αυτά, η εύρεση αυτής της σειράς μετασχηματισμών και η εφαρμογή τους στα δέντρα θα γίνει σε επόμενο βήμα.

Είναι εύκολο να συμπεράνουμε από αυτές τις ιδιότητες, ότι το σύνολο C είναι μια τομή μεγέθους $k - 1$.

Αρχικά παρατηρούμε από τις πρώτες 2 ιδιότητες ότι ο εισερχόμενος βαθμός του C μέσα στα δέντρα είναι $k - 1$. Αυτό ισχύει διότι ο συνολικός εισερχόμενος βαθμός των κορυφών

του $|C|$ είναι $(|C| - 1) \cdot k + (k - 1)$ αφού οι κορυφές του C μέσα στα δέντρα έχουν εισερχόμενο βαθμό k , εκτός από την ελλιπή κορυφή που έχει εισερχόμενο βαθμό $k - 1$. Το C περιέχει τουλάχιστον $k(|C| - 1)$ ακμές που είναι εξ' ολοκλήρου μέσα στο C , αφού είναι η ένωση θεμελιωδών κύκλων σε όλα τα δέντρα. Έτσι, προκύπτει ότι το C έχει το πολύ $k - 1$ εισερχόμενο βαθμό στα δέντρα που έχουμε χτίσει. Η τρίτη ιδιότητα μας εγγυάται ότι ο εισερχόμενος βαθμός σε όλο το γράφημα είναι $k - 1$.

2.2 Αλγόριθμος εύρεσης τομής Steiner

Αρχικά θα δώσουμε μια γενική περιγραφή του αλγορίθμου για τομές Steiner που προτείνεται στην έρευνα των Bhalgat et al. [12] και θα αποδείξουμε τις συνθήκες που μας επιτρέπουν να καταλάβουμε την ορθότητα και πολυπλοκότητα του αλγορίθμου στο πλαίσιο του αλγορίθμου κατασκευής δέντρων Gomory-Hu.

Ο αλγόριθμος εύρεσης τομής Steiner βασίζεται στον αλγόριθμο του Gabow που παρουσιάσαμε νωρίτερα (Ενότητα 2.1). Σε σχέση με τον αλγόριθμο του Gabow, ο αλγόριθμος βασίζεται σε μια λιγότερο περιοριστική μορφή του θεωρήματος πακεταρίσματος γραφήματος.

Θεώρημα 6 (Θεώρημα πακεταρίσματος γραφήματος [5]). *Δεδομένου ενός κατευθυνόμενου γραφήματος Euler $G = (V, E)$ και μιας οποιασδήποτε κορυφής $r \in V$, υπάρχουν k δέντρα χωρίς κατεύθυνση ασύνδετα ως προς τις ακμές T_1, \dots, T_k με την κορυφή r ως ρίζα, έτσι ώστε κάθε κορυφή $v \neq r, v \in V$ να εμφανίζεται $con(v)$ φορές σε όλα τα δέντρα, και να έχει εισερχόμενο βαθμό $con(v)$ σε αυτά, όπου $con(v)$ είναι η ακμοσυνεκτικότητα $r \rightarrow v$ και $k = \max_{v \in V \setminus \{r\}} con(v)$.*

Είναι χρήσιμο να παρατηρήσουμε ότι το θεώρημα δεν περιορίζει την εμφάνιση μιας κορυφής παραπάνω από μία φορές σε ένα δέντρο T_i . Θα δούμε ότι θα χρησιμοποιήσουμε αυτή την ιδιότητα για μια συγκεκριμένη κατηγορία κορυφών.

Ο αλγόριθμος αρχικά μοιάζει με τον αλγόριθμο του Gabow, που περιγράψαμε παραπάνω (Ενότητα 2.1). Χτίζει ξένα ως προς τις ακμές δέντρα T_1, \dots, T_k σε κάθε γύρο, των οποίων οι κορυφές ικανοποιούν τη συνθήκη του Θεωρήματος 6 και σε κάθε βήμα του γύρου προσπαθούμε να ενώσουμε κάποια συνεκτικά στοιχεία με άλλα, υποδιαιρώντας έτσι τον αριθμό τους. Ο αλγόριθμος του Gabow, όταν "κολλήσει", δηλαδή όταν το σύνολο C συγκλίνει, κατά την κατασκευή του δέντρου T_k θα επιστρέψει κατευθείαν ένα σύνολο κορυφών C που εκφράζει μια τομή μεγέθους $k - 1$. Αυτό δεν μας είναι χρήσιμο όμως: ο αλγόριθμος πρέπει να εντοπίσει μια τομή Steiner, δηλαδή να χωρίσει ένα υποσύνολο κορυφών S ενός γραφήματος σε δύο μέρη· ο αλγόριθμος του Gabow δεν μας εγγυάται κάτι τέτοιο, δηλαδή μπορεί το S να είναι υποσύνολο των δύο παραγόμενων συνόλων της τομής, δηλαδή το S δεν θα χωριστεί από την τομή. Από το Θεώρημα 3, όμως, ξέρουμε ότι αν υπάρχει μια τομή C που να χωρίζει το γράφημα έτσι ώστε όλες οι κορυφές του S να είναι στη μία μεριά της τομής, τότε υπάρχει και μια τομή, αντίστοιχου μεγέθους, που μπορεί να χωρίσει το σύνολο S στο γράφημα που έχουμε συνθλίψει τη μεριά της τομής C που δεν περιέχει το S . Ως αποτέλεσμα, μπορούμε να θεωρήσουμε το C ως μια κορυφή για το υπόλοιπο των υπολογισμών μας. Αυτό το συνθλιμμένο σύνολο κορυφών θα το ονομάζουμε *μαύρη κορυφή*. Ο Gabow μας εγγυάται ότι το σύνολο κορυφών που προκύπτουν από τον υπολογισμό συνόλου εγκλεισμού θα είναι γειτονικές μεταξύ τους σε όλα τα δέντρα, επομένως μπορούμε να τις συνθλίψουμε σε όλα τα δέντρα σε μια μαύρη κορυφή b . Ο αλγόριθμος προς μελέτη, χρησιμοποιεί διαφορετικές μεθόδους υπολογισμού του συνόλου εγκλεισμού και εφαρμόζει

διαφορετικούς μετασχηματισμούς στα δέντρα, αλλά αποδεικνύεται ότι οι αλλαγές αυτές στον αλγόριθμο δεν θα επηρεάσουν την ιδιότητα της γειτνίασης των κορυφών.

Όπως αναφέρθηκε, ο αλγόριθμος τομής Steiner, μπορεί να συνθλίψει ένα σύνολο κορυφών σε μια μαύρη κορυφή. Ο αλγόριθμος χρησιμοποιεί δύο ειδών κορυφές: τις άσπρες και τις μαύρες. Οι μαύρες κορυφές είναι *υπέρ-κορυφές* (*super-vertices*), δηλαδή αποτελούνται από πολλές άλλες κορυφές, άσπρες ή μαύρες, ενώ οι άσπρες κορυφές αντιπροσωπεύουν κορυφές που βρίσκονται στο αρχικό μας γράφημα.

Θα απαριθμήσουμε μερικές χρήσιμες ιδιότητες των μαύρων κορυφών. Έστω μια μαύρη κορυφή b και ότι μέχρι τώρα έχουμε χτίσει T_1, \dots, T_{i-1} δέντρα και χτίζουμε τώρα το δέντρο T_i :

1. Η b δεν μπορεί να περιέχει τη ρίζα r
2. Η b έχει εισερχόμενο βαθμό $j \leq i - 1$
3. Η b δημιουργήθηκε κατά την κατασκευή του δέντρου T_{j+1} , δηλαδή κατά την κατασκευή του δέντρου T_{j+1} βρέθηκε μια τομή στο γράφημα, μεγέθους j , αλλά δεν αφού δεν χώριζε το σύνολο κορυφών Steiner, συνθλίψαμε την b σε μια κορυφή για να συνεχίσουμε τους υπολογισμούς μας.
4. Όλες οι κορυφές που ανήκουν στο b είναι γειτονικές σε όλα τα δέντρα. Αυτή η ιδιότητα δεν αλλάζει, διότι οποιαδήποτε ενέργεια πραγματοποιείται στο σύνολο κορυφών b , και όχι στις κορυφές μέσα στην b .
5. Η b έχει $con(b)$ εμφανίσεις συνολικά σε όλα τα δέντρα, και μπορεί να εμφανίζεται περισσότερες από μία φορές σε κάποια δέντρα.
6. Κάθε άσπρη κορυφή w μέσα στην b έχουν $con(w) = j$, όσο και η ίδια η b . Αυτό ισχύει διότι οι άσπρες κορυφές είναι ουσιαστικά οι κορυφές που χωρίζει η τομή που εκφράζει η κορυφή b από την κορυφή r .
7. Οι μαύρες κορυφές b' μέσα στην b έχουν $con(b') \leq j$. Αυτό ισχύει διότι η b' μπορεί να έχει ανακαλυφθεί στον ίδιο γύρο με την b ($con(b') = j$) ή σε προηγούμενο γύρο $con(b') < j$.

Μερικές ιδιότητες των μαύρων κορυφών που περιγράψαμε, θα δούμε ότι είναι όμοιες με της ιδιότητες των συνόλων κλειστότητας του αλγόριθμου. Αυτό ισχύει διότι η μαύρες κορυφές προκύπτουν από τα (μεγιστικά) σύνολα κλειστότητας.

Παραθέτουμε επίσης μερικές ιδιότητες των άσπρων κορυφών, δεδομένου ότι βρισκόμαστε στην κατάσταση όπου έχουμε χτίσει T_1, \dots, T_{i-1} δέντρα και χτίζουμε τώρα το δέντρο T_i :

1. Έστω μια άσπρη κορυφή w που δεν ανήκει σε καμία μαύρη κορυφή, δηλαδή βρίσκεται στα δέντρα μας. Η κορυφή αυτή μπορεί να έχει $con(w) \geq i$ από την r . Διαφορετικά, $con(w) = i - 1$, αλλά προς το παρόν δεν έχουμε ανακαλύψει κάποια τομή. Μέχρι το τέλος της κατασκευής του T_i θα έχουμε βρει μια τέτοια τομή.
2. Οι άσπρες κορυφές που δεν ανήκουν σε καμία μαύρη κορυφή εμφανίζονται ακριβώς i φορές συνολικά σε όλα τα δέντρα, ακριβώς μια φορά σε κάθε δέντρο.
3. Για τις άσπρες κορυφές w που ανήκουν σε μια μαύρη κορυφή b με $con(b) = j$, ισχύει $con(w) = con(b)$.

Σε κάθε τομή που βρίσκουμε η οποία δεν είναι η τομή Steiner που αναζητάμε, τη συνθλίβουμε σε μια μαύρη κορυφή b . Έτσι, προκύπτει το πρόβλημα ότι το b έχει εισερχόμενο βαθμό k κατά την κατασκευή του δέντρου T_{k+1} , αλλά έχει $k + 1$ εμφανίσεις. Λύση σε αυτό το πρόβλημα δίνει η μέθοδος *Ισοζυγισμού Βαθμού (Degree Balancing)*, η οποία μας επιτρέπει να συνεχίσουμε τον αλγόριθμο μέχρι την εύρεση μιας Steiner τομής. Η μέθοδος Ισοζυγισμού Βαθμού μας βοηθάει επίσης να κρατήσουμε τις αμετάβλητες συνθήκες που περιγράφουμε στην Ενότητα 2.2.1. Επίσης, είναι σημαντική γιατί ο αλγόριθμος αντιμετωπίζει τις μαύρες κορυφές, και πιο συγκεκριμένα τα μονοπάτια από μαύρες κορυφές, ως ακμές. Επίσης, με τη λήξη της μεθόδου, θα μειωθούν οι εμφανίσεις της προβληματικής κορυφής b κατά 1 και θα μπορούμε να συνεχίσουμε τον αλγόριθμο μας, αφού θα ισχύει το Θεώρημα 6.

Παρακάτω παρουσιάζουμε τον ψευδοκώδικα για ένα ολόκληρο βήμα του αλγόριθμου:

Αλγόριθμος 1 Αλγόριθμος ενός βήματος

- 1: **Για** συνεκτικό στοιχείο $Comp$ **κάνε**
 - 2: Αρχικοποιούμε το $Seed(Comp)$ με την υπέρ-ακμή που σχετίζεται με το $Comp$, αλλά αν δεν έχει, με τις αχρησιμοποίητες ακμές της ελλιπούς κορυφής του $Comp$.
 - 3: **Επανάλαβε**
 - 4: Τρέχουμε τον υπολογισμό κλειστότητας στο $Comp$ με το $Seed(Comp)$ για να βρούμε ένα νέο σύνολο κλειστότητας B .
 - 5: **Αν** $B \cap S \neq \emptyset$ **τότε**
 - 6: **Επέστρεψε** B ως την τομή Steiner για το σύνολο S
 - 7: **Τέλος**
 - 8: Συνθλίβουμε το B σε μια μαύρη κορυφή και τρέχουμε τον τοπικό ισοζυγισμό βαθμού για να ισοζυγίσουμε τον βαθμό του B , το οποίο έχει ως αποτέλεσμα να δημιουργηθεί μια υπέρ-ακμή την οποία θέτουμε ως $Seed(Comp)$
 - 9: **Μέχρι** να τερματίσει ο υπολογισμός συνόλου εγκλεισμού τερματίσει πρώτου βρεθεί ένα σύνολο εγκλεισμού ή ο ισοζυγισμός βαθμού τερματίσει γιατί κατά τον ισοζυγισμό βγαίνουμε εκτός του $Comp$
 - 10: **Τέλος επανάληψης**
 - 11: Τρέχουμε το καθολικό ισοζυγισμό βαθμού για όλα τα συνεκτικά στοιχεία
 - 12: **Για κάθε** συνεκτικό σύνολο $Comp$ **κάνε**
 - 13: Τρέχουμε τον υπολογισμό κλειστότητας
 - 14: **Τέλος επανάληψης**
 - 15: Βρίσκουμε μια αλυσίδα μετασχηματισμών για κάθε συνεκτικό στοιχείο $Comp$, και εφαρμόζουμε αυτούς τους μετασχηματισμούς για ένα μέρος των συνεκτικών στοιχείων, υποδιαιρώντας έτσι τον αριθμό των συνεκτικών στοιχείων κατά ένα σταθερό συντελεστή.
-

Όπως έχουμε πει, οι μαύρες κορυφές αντιπροσωπεύουν μια τομή μεγέθους $\leq k$. Μετά την εύρεση μιας μαύρης κορυφής, ο αλγόριθμος την αντιμετωπίζει ως μέρος μιας ακμής. Πιο συγκεκριμένα, ένα μονοπάτι από μαύρες ακμές ανάμεσα σε δύο άσπρες κορυφές είναι μια *υπέρ-ακμή (super edge)*. Ένα μονοπάτι μαύρων κορυφών που καταλήγει σε μία μόνο άσπρη κορυφή ονομάζεται *μερική υπέρ-ακμή (partial super edge)*.

2.2.1 Αμετάβλητες συνθήκες του αλγόριθμου

1. Κάθε συνεκτικό στοιχείο K του τελευταίου δέντρου T_{k+1} , εκτός από το συνεκτικό στοιχείο που περιέχει την κορυφή r , περιέχει το πολύ μια ελλiptή άσπρη κορυφή της οποίας ο συνολικός εισερχόμενος βαθμός είναι k . Αν δεν υπάρχει μια τέτοια κορυφή, τότε υπάρχει μια υπέρ-ακμή που έχει και τα δύο της άκρα στο στοιχείο.
2. Μια άσπρη κορυφή εμφανίζεται ακριβώς μία φορά σε κάθε δέντρο T_1, \dots, T_{k+1} .
3. Μια άσπρη κορυφή, εκτός της ρίζας r , έχει συνολικό εισερχόμενο βαθμό k είτε $k+1$ μέσα σε όλα τα δέντρα, συμπεριλαμβανομένων και των υπέρ-ακμών που μπορεί να έχουν προκύψει σε κάποιο βήμα.
4. Κάθε μαύρη κορυφή b έχει συνολικές εμφανίσεις και εισερχόμενο βαθμό ίσο με $con(b)$ στα δέντρα, συμπεριλαμβανομένων και των υπέρ-ακμών που μπορεί να έχουν προκύψει σε κάποιο βήμα, με $con(b) \leq k$. Κάθε μαύρη κορυφή μπορεί να εμφανίζεται πάνω από μια φορές μέσα σε ένα δέντρο και δεν περιλαμβάνει το r .
5. Όλες οι εμφανίσεις μια μαύρης κορυφής στα δέντρα T_1, \dots, T_{k+1} έχει βαθμό το πολύ 2, και όλες οι κορυφές με βαθμό 1 υπάρχουν μέσα στο T_1 .

Ακολουθούν μερικές σημειώσεις που θα μπορούσαμε να κάνουμε πάνω σε αυτές τις συνθήκες:

1. Σε κάθε περίπτωση, όταν βρισκόμαστε στο βήμα υπολογισμού του συνόλου εγκλεισμού, βρισκόμαστε σε μία από αυτές τις περιπτώσεις. Επομένως, ως seed της μεθόδου χρησιμοποιούνται είτε οι αχρησιμοποίητες ακμές κατευθυνόμενες προς την ελλiptή κορυφή του K , είτε την υπέρ-ακμή που σχετίζεται με αυτό.
5. Η μαύρη κορυφή όπως είπαμε είναι μια τομή που ανακαλύψαμε κατά τη δημιουργία του δέντρου $con(b)$, μεγέθους $con(b)$. Η ιδέα ότι η κορυφή αυτή μπορεί να σχετίζεται με περισσότερες από μία εμφανίσεις, έχει να κάνει με το γεγονός ότι η κορυφή αυτή αντιπροσωπεύει μια ακμή
6. Όταν παύει να ισχύει η συνθήκη αυτή, αφού έχουμε συνθλίψει μια μαύρη κορυφή, τη διορθώνουμε με τη μέθοδο του ισοζυγισμού βαθμού.

2.2.2 Υπορουτίνες αλγορίθμου

Ο αλγόριθμος έχει τις παρακάτω σημαντικές υπορουτίνες, οι οποίες φαίνονται στον Αλγόριθμο 1:

2.2.2.1 Υπολογισμός συνόλου εγκλεισμού

Όπως ο αλγόριθμος του Gabow, ο υπολογισμός γίνεται για κάθε συνεκτικό στοιχείο του δέντρου T_{k+1} ξεχωριστά. Ο υπολογισμός του συνόλου εγκλεισμού δέχεται ως είσοδο (seed) είτε μια υπέρ-ακμή που και τα δύο της άκρα βρίσκονται στο συνεκτικό στοιχείο, είτε ένα σύνολο των αχρησιμοποίητων ακμών κατευθυνόμενες προς την ελλiptή κορυφή v του συνεκτικού στοιχείου. Ξεκινάει εισάγοντας όλες τις κορυφές στο μονοπάτι της υπέρ-ακμής ή

όλες τις κορυφές των αχρησιμοποίητων ακμών στα βοηθητικά σύνολα του υπολογισμού συνόλου εγκλεισμού.

Υπολογίζουμε ένα σύνολο εγκλεισμού για ένα συνεκτικό στοιχείο $Comp$ στο δέντρο T_{k+1} . Το σύνολο εγκλεισμού που θα υπολογίσουμε είναι ένα ελάχιστο σύνολο μαύρων και άσπρων κορυφών B με τις παρακάτω ιδιότητες:

1. Στην περίπτωση που ως είσοδο/seed στο υπολογισμό έχουμε μια υπέρ-ακμή, το B περιέχει όλες τις μαύρες κορυφές και τις δύο άσπρες κορυφές της υπέρ-ακμής. Διαφορετικά, αν σαν είσοδο έχει τις ακμές που κατευθύνονται προς μια ελλιπή κορυφή, τότε το B περιέχει αυτή την ελλιπή κορυφή.
2. Το σύνολο εγκλεισμού B δεν περιέχει την κορυφή r .
3. Όλες οι άσπρες κορυφές του B ανήκουν στο $Comp$.
4. Για όλα τα δέντρα T_i , οι εμφανίσεις στο T_i όλων των κορυφών του B είναι γειτονικές.
5. Όλες οι αχρησιμοποίητες ακμές που κατευθύνονται σε κορυφές στο B έχουν και τα δύο άκρα τους στο B .

Ο υπολογισμός έχει δύο πιθανά αποτελέσματα:

1. Όπως στον αλγόριθμο του Gabow, αν ο υπολογισμός "κολλήσει", δηλαδή τερματίσει επειδή το σύνολο C συγκλίνει, επιστρέφεται ένα σύνολο C που αντιπροσωπεύει μια τομή μεγέθους k στο γράφημα μας. Αυτή την τομή στη συνέχεια θα τη συνθλίψουμε σε μια μαύρη κορυφή. Σε αυτή την περίπτωση θα χρειαστεί να κληθεί η υπορουτίνα ισοζυγισμού βαθμού για τη νέα μαύρη κορυφή.
2. Διαφορετικά αν στο σύνολο C πάει να μπει μια κορυφή εκτός του συνεκτικού στοιχείου που επεξεργαζόμαστε, τότε το b είναι μεγιστική μαύρη κορυφή ως προς το συνεκτικό της στοιχείο και μπορούμε μέσω κάποιον μετασχηματισμών στα δέντρα να ενώσουμε το συνεκτικό στοιχείο που επεξεργαζόμαστε με ένα άλλο.

Αν το σύνολο που βρέθηκε στη πρώτη περίπτωση έχει κοινές κορυφές με το σύνολο Steiner S , τότε τερματίζουμε ολόκληρο τον αλγόριθμο, γιατί σε αυτή την περίπτωση έχουμε βρει μια τομή Steiner μεγέθους k που χωρίζει το σύνολο S στη μέση, και τερματίζουμε τον αλγόριθμο.

Από εδώ και πέρα, θα αναφέρουμε ως *τρέχον γράφημα* το γράφημα που επεξεργαζόμαστε με τις μαύρες κορυφές συνθλιμμένες. Επίσης, για κάθε υποσύνολο $B \in V$ θα ορίσουμε τον αριθμό των *ενιαίων περιοχών* ως εξής: Έστω το γράφημα \mathcal{R} το οποίο σχηματίζεται από τις εμφανίσεις των κορυφών σε όλα τα δέντρα συν όλες τις υπέρ-ακμές, και το υπογράφημα που δημιουργείται από τις εμφανίσεις των κορυφών του συνόλου κορυφών B στο \mathcal{R} . Ο αριθμός των συνεκτικών συνόλων που σχηματίζεται στο υπογράφημα είναι ο αριθμός των *ενιαίων περιοχών* στο B .

Έχοντας ορίσει αυτές τις έννοιες θα δείξουμε το παρακάτω θεώρημα:

Θεώρημα 7. [7] Το σύνολο B που βρίσκει ο υπολογισμός κλειστότητας είναι μια τομή μεγέθους k .

Λήμμα 5. [7] Έστω ένα σύνολο $B \subseteq V$ όπου $r \notin B$. Έστω x αριθμός των ενιαίων περιοχών για το B , y ο αριθμός των ελλιπών κορυφών στο B και z ο αριθμός των υπέρ-ακμών που είναι ολόκληρες μέσα σε μια ενιαία περιοχή. Δεδομένων των αμετάβλητων συνθηκών που ορίσαμε, ο εισερχόμενος βαθμός του B στο \mathcal{R} είναι ακριβώς $x - z - y$. Επίσης, αν το B περιέχει τουλάχιστον μια άσπρη κορυφή τότε $x - z - y \geq k$.

Απόδειξη. Έστω q ο αριθμός των εμφανίσεων των κορυφών του B στο \mathcal{R} . Όλες οι κορυφές εκτός των ελλιπών κορυφών, έχουν εισερχόμενο βαθμό ίσο με τον αριθμό εμφανίσεων τους: οι μαύρες έχουν αριθμό εμφανίσεων ίσο με τον εισερχόμενο βαθμό τους (αμετάβλητη Συνθήκη 4) και οι μη ελλιπείς άσπρες κορυφές έχουν $k + 1$ εμφανίσεις (αμετάβλητη Συνθήκη 2) και εισερχόμενο βαθμό $k + 1$ (αμετάβλητη Συνθήκη 3). Έτσι έχουμε συνολικό άθροισμα εισερχόμενου βαθμού των κορυφών του B ίσο με $q - y$.

Υπολογίζουμε τώρα τον εισερχόμενο βαθμό στο B στο γράφημα \mathcal{R} μόνο από ακμές εντελώς μέσα στο B . Έχουμε ότι όλες οι κορυφές συνεισφέρουν σε αυτό τον αριθμό όσο και οι εμφανίσεις τους, δηλαδή q . Όμως, κάθε ενιαία περιοχή έχει μια κορυφή που είναι σαν "ρίζα" της περιοχής και δεν έχει κάποια ακμή που να είναι εντελώς μέσα στο B και να κατευθύνεται προς αυτήν, συνεισφέροντας αρνητικά στο άθροισμα. Συνεπώς, έχουμε x συνολική αρνητική συνεισφορά στο άθροισμα, όσες και οι ενιαίες περιοχές. Τέλος, στο \mathcal{R} συμπεριλαμβάνουμε και τις, z στο σύνολο τους, υπέρ-ακμές οι οποίες είναι ολόκληρες μέσα στις έννοιες περιοχές, με αποτέλεσμα να δημιουργείται ένας κύκλος που "ακυρώνει" την αρνητική συνεισφορά που αναφέραμε για την ενιαία περιοχή αυτή. Επομένως, τελικά έχουμε $q - (x - z)$ εισερχόμενο βαθμό στο B στο γράφημα \mathcal{R} μόνο από ακμές εντελώς μέσα στο B .

Η διαφορά ανάμεσα στον συνολικό εισερχόμενο βαθμό των κορυφών του B και τον εσωτερικό εισερχόμενο βαθμό του B είναι ο εισερχόμενος βαθμός του στο γράφημα \mathcal{R} , δηλαδή $(q - y) - (q - (x - z)) = x - z - y$.

Για την ανισότητα παρατηρούμε ότι το x προκύπτει από δύο ειδών ενιαίες περιοχές: αυτές που έχουν προκύψει από τα δέντρα T_1, \dots, T_k (x_1) και αυτές που έχουν προκύψει από το δάσος δέντρο T_{k+1} (x_2). Οι ενιαίες περιοχές στα T_1, \dots, T_k δέντρα είναι περισσότερες ή ίσες από τα δέντρα, δηλαδή $x_1 \geq k$, διότι το σύνολο B μπορεί να μην έχει γειτονικές κορυφές για όλα τα δέντρα, επομένως να δημιουργεί περισσότερες από μία ενιαίες περιοχές για ένα δέντρο. Επίσης, τα συνεκτικά στοιχεία στο δέντρο T_{k+1} μπορούν να έχουν είτε μια ελλιπή κορυφή, είτε να έχουν μια συσχετισμένη υπέρ-ακμή, είτε να περιέχουν την κορυφή r . Βλέπουμε έτσι ότι οι ενιαίες περιοχές στο T_{k+1} είναι λιγότερες ή ίσες από τις ελλιπείς κορυφές και τις υπέρ-ακμές των συνεκτικών στοιχείων στο B . Επομένως, $x_2 \geq z + y$. Άρα έχουμε, $x = x_1 + x_2 \geq k + y + z \implies x - y - z \geq k$ \square

Λήμμα 6. [7] Δεδομένων των αμετάβλητων συνθηκών 1-5 και των ιδιοτήτων 1-4 για ένα σύνολο εγκλεισμού B , το B έχει εισερχόμενο βαθμό k στο \mathcal{R} , όπως και στα δέντρα T_1, T_2, \dots, T_{k+1} .

Απόδειξη. Έστω ότι το B αποτελεί σύνολο εγκλεισμού στο συνεκτικό στοιχείο $Comp$. Οι ιδιότητες 1-4 δείχνουν ότι $x = k + 1$, δηλαδή οι ενιαίες περιοχές είναι το σύνολο κορυφών σε όλα τα δέντρα. Από την ιδιότητα 3 έχουμε ότι $z = 1, y = 0$ ή $y = 1, z = 0$. Λόγω της ιδιότητας 2, ισχύει το Λήμμα 5 και έτσι ισχύει η πρώτη πρόταση του Λήμματος. Από τη στιγμή που η ιδιότητα ισχύει για το \mathcal{R} , το οποίο περιέχει όλα τα δέντρα T_1, T_2, \dots, T_{k+1} στην υπέρ-ακμές, λόγω της ιδιότητας 1, δηλαδή ότι το B περιέχει και την υπέρ-ακμή του $Comp$, έχουμε ότι ισχύει και η δεύτερη πρόταση του Λήμματος. \square

Λήμμα 7. [7] Δεδομένων των αμετάβλητων συνθηκών 1-5, ένα σύνολο εγκλεισμού B εκφράζει μια τομή μεγέθους k στο τρέχον γράφημα. Επίσης, το B εκφράζει μια ελάχιστη τομή στο τρέχον γράφημα που χωρίζει τις άσπρες κορυφές του B με τη ρίζα r .

Απόδειξη. Λόγω του Λήμματος 6, ο εισερχόμενος βαθμός στο \mathcal{R} είναι k . Λόγω της ιδιότητας 5, ο εισερχόμενος βαθμός του B στο τρέχον γράφημα είναι k , διότι μόνο οι αχρησιμοποίητες ακμές διαφέρουν ανάμεσα στο \mathcal{R} και στο τρέχον γράφημα. Λόγω του 5, βλέπουμε ότι ο εισερχόμενος βαθμός του κάθε συνόλου εγκλεισμού B φράσσεται από το k , επομένως το B όταν ισχύουν οι αμετάβλητες Συνθήκες 1-5, είναι μια ελάχιστη τομή ανάμεσα στο r και τις άσπρες κορυφές στο B . \square

Στο παραπάνω Λήμμα αναφέρουμε ότι το B εκφράζει μια τομή k ανάμεσα στις άσπρες κορυφές του B και την κορυφή r . Διαισθητικά, μπορούμε να πούμε ότι αν αφαιρέσουμε όλες τις ακμές που κατευθύνονται προς το B , διαχωρίζουμε και μια άσπρη κορυφή w του B από το r . Αυτό ισχύει λόγω της ιδιότητας 5, δηλαδή όλες οι αχρησιμοποίητες ακμές προς την w είναι μέσα στο B , επομένως μόνο οι k ακμές του B συνδέουν την w με την r .

Λήμμα 8. [7] *Δεδομένων των αμετάβλητων συνθηκών 1-5, ένα σύνολο εγκλεισμού B εκφράζει μια τομή μεγέθους k στο αρχικό γράφημα. Επίσης, το B εκφράζει μια ελάχιστη τομή στο αρχικό γράφημα που χωρίζει τις άσπρες κορυφές του B με τη ρίζα r .*

Απόδειξη. Αν αποσυμπιέσουμε τις μαύρες κορυφές στο τρέχον γράφημα, τότε "παιρνουμε" το αρχικό γράφημα. Επομένως, από το Λήμμα 7, αποδεικνύεται η πρώτη πρόταση του λήμματος.

Αν το τρέχον γράφημα δεν έχει μαύρες κορυφές, τότε ταυτίζεται με το αρχικό γράφημα. Αν το τρέχον γράφημα περιέχει μαύρες κορυφές, λόγω επαγωγής, λέμε ότι κάθε σύνολο εγκλεισμού B' που συσχετίζεται με μια μαύρη κορυφή εκφράζει μια ελάχιστη τομή ανάμεσα στην κορυφή r και τις άσπρες κορυφές μέσα στο B' στο αρχικό γράφημα. Εφαρμόζουμε το Λήμμα 3 για μία μαύρη κορυφή και δείχνουμε ότι υπάρχει μια τομή στο αρχικό γράφημα, όπου η μαύρη κορυφή είναι ολόκληρη στο ένα μέρος της τομής, δηλαδή η μαύρη κορυφή δεν χωρίζεται από την τομή αυτή. Έτσι μπορούμε να εφαρμόσουμε επαναληπτικά το Λήμμα 3: η μία πλευρά της ελάχιστης τομής που προέκυψε από την προηγούμενη εφαρμογή του λήμματος δεν περιέχει τη μαύρη κορυφή της προηγούμενης επανάληψης. Σε αυτή την πλευρά, μπορούμε να εφαρμόσουμε πάλι το λήμμα και να αποδείξουμε ότι υπάρχει μια ελάχιστη τομή ανάμεσα στο r και τις άσπρες κορυφές του B που δεν χωρίζει μια άλλη μαύρη κορυφή στο αρχικό γράφημα. Ακολουθώντας αυτή την διαδικασία επαναληπτικά για όλες τις μαύρες κορυφές, δείχνουμε ότι υπάρχει τομή που δεν χωρίζει καμία μαύρη κορυφή στο αρχικό γράφημα. Η τομή αυτή αντιστοιχίζεται σε μια τομή στο τρέχον γράφημα με το ίδιο μέγεθος. Επομένως, λόγω του Λήμματος 7, το σύνολο εγκλεισμού B εκφράζει μια ελάχιστη τομή που χωρίζει το r από τις άσπρες κορυφές που ανήκουν στο B και στο αρχικό γράφημα. \square

Ορισμός 10 (Ελαχιστική ελάχιστη τομή Steiner). *Ορίζουμε ως ελαχιστική ελάχιστη τομή Steiner μια ελάχιστη τομή Steiner $(B, V \setminus B)$ στο αρχικό γράφημα για την οποία η κορυφή $v, v \in B, r \in V \setminus B$ και δεν υπάρχει κάποιο γνήσιο υποσύνολο του B που να περιέχει το v και να είναι ελάχιστη τομή Steiner.*

Στον ορισμό παραπάνω, ο όρος "ελαχιστική" αναφέρεται στο μέγεθος του συνόλου κορυφών B , ενώ ο όρος "ελάχιστη" στο μέγεθος της τομής.

Λήμμα 9. [7] *Έστω k η τιμή της ελάχιστης τομής Steiner. Μια τομή $(B, V \setminus B)$ είναι ελαχιστική ελάχιστη τομή Steiner στο αρχικό γράφημα αν και μόνο αν το B εμφανίζεται ως σύνολο εγκλεισμού κατά την κατασκευή του δέντρου T_{k+1} και περιέχει μια άσπρη κορυφή.*

Απόδειξη. Υποθέτουμε ότι το B είναι μια ελαχιστική ελάχιστη τομή Steiner και το v μια άσπρη κορυφή η οποία χωρίζεται από την κορυφή r μέσω αυτής της τομής. Βλέπουμε ότι

το v έχει ακμοσυνεκτικότητα από το r ίση με το μέγεθος της τομής B , δηλαδή k . Έστω ένα σύνολο εγκλεισμού B' στο οποίο μπαίνει η κορυφή v κατά την κατασκευή ενός δέντρου. Σύμφωνα με το Λήμμα 8, αυτό το δέντρο δεν μπορεί να είναι άλλο εκτός από το T_{k+1} , αλλιώς η ελάχιστη τομή στο αρχικό γράφημα δεν θα ήταν μεγέθους k . Χρησιμοποιώντας αυτό, καθώς και την ιδιότητα της ελαχιστικότητας των συνόλων B (εξ' ορισμού ελαχιστικής ελάχιστης τομής Steiner) και του B' (εξ' ορισμού του συνόλου εγκλεισμού), βλέπουμε ότι $B = B'$. Αρκεί τώρα να δείξουμε την αντίθετη μεριά της συνεπαγωγής.

Υποθέτουμε ότι το B εμφανίζεται ως σύνολο εγκλεισμού κατά την κατασκευή του δέντρου T_{k+1} και περιέχει μια άσπρη κορυφή v . Λόγω του Λήμματος 8, το B είναι μια ελάχιστη τομή Steiner στο αρχικό γράφημα. Έστω ένα σύνολο $B' \subset B, v \in B'$. Εξ' ορισμού, το σύνολο εγκλεισμού είναι ελαχιστικό σύνολο που ικανοποιεί τις Συνθήκες 1-5 του συνόλου εγκλεισμού. Άρα το B' παραβιάζει μία από τις Συνθήκες 1-5 και πιο συγκεκριμένα μία από τις 1,4 ή 5, διότι το υποσύνολο ενός συνόλου εγκλεισμού B δεν μπορεί να περιέχει κορυφές που δεν περιείχε το B , την r ή άσπρες κορυφές εκτός του $Comp$. Θα δείξουμε ότι η παραβίαση τουλάχιστον μίας από τις Συνθήκες 1,4 ή 5 συνεπάγεται ότι το B' είναι τομή μεγέθους $\geq k + 1$.

Έστω το B' παραβιάζει μόνο την ιδιότητα 5. Τότε έχουμε ότι και πάλι ισχύει το Λήμμα 6, έτσι το B έχει εισερχόμενο βαθμό k στα δέντρα. Αλλά, λόγω του ότι το παραβιάζεται η Συνθήκη 5, ο εισερχόμενος βαθμός είναι μεγαλύτερος στο αρχικό γράφημα, καθώς υπάρχει τουλάχιστον μία αχρησιμοποίητη ακμή που κατευθύνεται προς το B' .

Έστω ότι παραβιάζει την ιδιότητα 1. Τότε οι ποσότητες z, y από το Λήμμα 5, δηλαδή ο αριθμός των υπέρ-ακμών και ο αριθμός των ελλিপών κορυφών μηδενίζονται ($y = z = 0$). Για το x έχουμε ότι ο αριθμός των ενιαίων περιοχών δεν μπορεί να μειωθεί σε αυτή την περίπτωση, επομένως $x \geq k + 1$. Άρα το μέγεθος της τομής του B' στο \mathcal{R} είναι $x - y - z \geq k + 1$, επομένως και στο τρέχον γράφημα, άρα και στο αρχικό.

Έστω ότι παραβιάζεται η ιδιότητα 4. Τότε έχουμε τουλάχιστον ένα δέντρο T_j , στο οποίο οι εμφανίσεις των κορυφών του B , δεν είναι γειτονικές, επομένως οι ενιαίες περιοχές που σχηματίζονται είναι περισσότερες από τον αριθμό των δέντρων, δηλαδή $x > k + 1$. Όπως έχουμε ήδη πει, λόγω της ιδιότητας 1, τα $z + y$ είναι το πολύ 1, έτσι $x - y - z > k \implies x - y - z > k + 1$. Επομένως το B' είναι τομή μεγέθους τουλάχιστον $k + 1$ στο \mathcal{R} , άρα και στο τρέχον γράφημα και στο αρχικό γράφημα. \square

2.2.2.2 Μετασχηματισμοί δέντρων

Οι μετασχηματισμοί των δέντρων έχουν ως αποτέλεσμα να αντικαταστήσουμε τις ελεύθερες ακμές και υπέρ-ακμές που έχουμε με άλλες· αυτή η διαδικασία θα συνεχίσει ώσπου να βρεθούν ακμές τέτοιες ώστε η πρόσθεση τους στο δέντρο T_{k+1} να έχει ως αποτέλεσμα να υποδιαιρεθούν τα συνεκτικά στοιχεία του δέντρου T_{k+1} και να μην παραβιάσουμε καμιά συνθήκη που έχουμε θέσει. Η διαδικασία μοιάζει πολύ με τη διαδικασία μετασχηματισμού των δέντρων στον αλγόριθμο του Gabow. Η διαφορά είναι ότι πέρα από ακμές που κατευθύνονται προς ελλίπεις κορυφές, τώρα έχουμε και υπέρ-ακμές.

Ξεκινάμε έχοντας τα δέντρα T_1, \dots, T_k και το δέντρο T_{k+1} που ικανοποιούν τις αμετάβλητες Συνθήκες 1-5 και τα συνεκτικά στοιχεία $Comp_1, Comp_2, \dots, Comp_h$ στο T_{k+1} . Κάθε συνεκτικό στοιχείο έχει είτε μια υπέρ-ακμή ή ένα σύνολο ακμών που κατευθύνονται προς την ελλίπη κορυφή του συνεκτικού στοιχείου. Σε κάθε βήμα έχουμε ένα σύνολο e_1, e_2, \dots, e_h , όπου το κάθε e_i είναι η υπέρ-ακμή ή μία από τις ακμές που συσχετίζονται με το $Comp_i$. Μια σειρά μετασχηματισμών, όπως θα τους ορίσουμε παρακάτω, ικανοποιεί τις παρακάτω

συνθήκες:

1. Κάθε μετασχηματισμός έχει ως αποτέλεσμα τη μετατροπή ενός δέντρου και την αλλαγή ενός ή δύο e_i .
2. Κάθε e_i είναι είτε μια υπέρ-ακμή ή μια ακμή που κατευθύνεται προς μια άσπρη κορυφή.
3. Τουλάχιστον μία άσπρη κορυφή-άκρο του e_i ανήκει στο $Comp_i$.
4. Οι ακμές (ή υπέρ-ακμές) e_i είναι ελεύθερες, όπως περιγράψαμε και πιο πάνω στον αλγόριθμο του Gabow, δηλαδή δεν ανήκουν σε κανένα δέντρο και βρήκαμε μέσω μετασχηματισμών πάνω στις αρχικές e_i ακμές.

Παρακάτω θα ορίσουμε τους μετασχηματισμούς των δέντρων:

Swap Ανταλλάσσουμε μια υπέρ-ακμή e_i με μια ακμή ή μια υπέρ-ακμή f η οποία ανήκει στο θεμελιώδες κύκλωμα που δημιουργεί η εισαγωγή της e_i στο δέντρο. Η f αντικαθιστά την e_i .

Mate-Swap Ταιριάζουμε μια υπέρ-ακμή e_i με μια υπέρ-ακμή f σε κάποιο δέντρο, όπως φαίνεται στο Σχήμα 2.3. Η f δεν πρέπει να είναι στον θεμελιώδες κύκλωμα της e_i και πρέπει να έχει μια κοινή μαύρη κορυφή με την e_i . Αυτή η αλλαγή θα έχει ως αποτέλεσμα να δημιουργηθούν δύο υπέρ-ακμές, οι οποίες θα εισαχθούν στο δέντρο αυτό. Από το θεμελιώδες κύκλωμα που θα δημιουργηθεί, μπορούμε να εξάγουμε μια υπέρ-ακμή, η μια ακμή που κατευθύνεται προς μια άσπρη κορυφή, η οποία θα αντικαταστήσει την e_i .

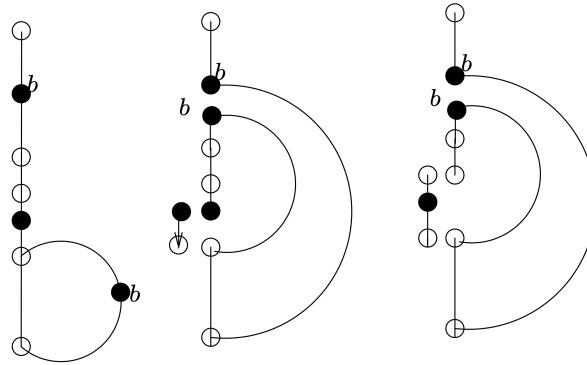
Incidence Αντικαθιστάμε μια ακμή e_i που κατευθύνεται προς μια άσπρη κορυφή w με μια άλλη, μη ελεύθερη ακμή εκτός των δέντρων που και αυτή κατευθύνεται προς την w .

Mate Ταιριάζουμε δύο υπέρ-ακμές e_i και e_j όπως φαίνεται στο Σχήμα 2.4. Οι δύο υπέρ-ακμές πρέπει να έχουν μια κοινή μαύρη κορυφή.

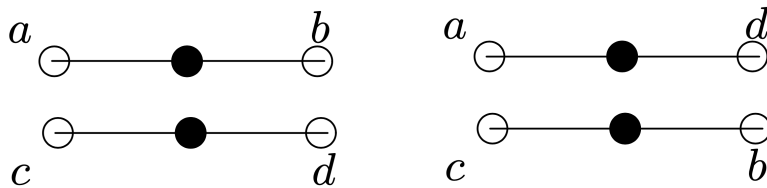
Join Υποθέτουμε ότι έχουμε μια ακμή e_i με ένα άκρο τη μαύρη κορυφή b και μια εμφάνιση της b ως φύλλο. Εξάγουμε τη μερική υπέρ-ακμή που συσχετίζεται με την εμφάνιση-φύλλο της b και την ενώνουμε με την e_i στο b , δημιουργώντας μια υπέρ-ακμή e_i .

Join-Swap Υποθέτουμε ότι έχουμε μια ακμή e_i με ένα άκρο τη μαύρη κορυφή b και μια εμφάνιση της b ως φύλλο. Εισάγουμε την e_i ταυτίζοντας την b της e_i με την εμφάνιση φύλλο της b . Από το θεμελιώδες κύκλωμα που δημιουργείται, μπορούμε να εξάγουμε μια οποιαδήποτε ακμή ή υπέρ-ακμή.

Connect Έστω ότι έχουμε μια υπέρ-ακμή e_i με ένα άκρο στο συνεκτικό στοιχείο $Comp_j$, $j \neq i$ (το ένα άκρο είναι σίγουρα στο $Comp_i$). Εισάγουμε στο δέντρο T_{k+1} την e_i και ενώνουμε τα δύο στοιχεία δημιουργώντας ένα καινούριο συνεκτικό στοιχείο, το οποίο θα συσχετίζεται με την υπέρ-ακμή e_j .



Σχήμα 2.3: Παράδειγμα του μετασχηματισμού mate-swap [7]



Σχήμα 2.4: Παράδειγμα του μετασχηματισμού mate [7]

2.2.2.3 Ισοζυγισμός Βαθμού

Ο αλγόριθμος έχει δύο υπορουτίνες ισοζυγισμού βαθμού: τον *τοπικό ισοζυγισμό βαθμού (local degree balancing)* και τον *καθολικό ισοζυγισμό βαθμού (global degree balancing)*. Ο τοπικός ισοζυγισμός, όπως υποδηλώνει το όνομα του ισοζυγίζει τον βαθμό των εμφανίσεων των μαύρων κορυφών στο συνεκτικό στοιχείο $Comp$ που επεξεργαζόμαστε, ενώ ο καθολικός επεξεργάζεται όλες τις μαύρες κορυφές που μπορεί να έχουν προκύψει για όλα τα συνεκτικά στοιχεία $Comp$.

Τα μονοπάτια μαύρων κορυφών χρησιμοποιούνται ως υπέρ-ακμές στον αλγόριθμο μας, με τη μη-κατευθυνόμενη έννοια, δηλαδή δεν μας ενδιαφέρει η κατεύθυνση των ακμών στο μονοπάτι αυτό. Επίσης, η μαύρη κορυφή b που έχουμε ανακαλύψει στα προηγούμενα βήματα έχει $k + 1$ εμφανίσεις σε όλα τα δέντρα, καθώς είναι παρών και σε όλα τα T_{k+1} δέντρα, και εισερχόμενο βαθμό k , αφού εκφράζει μια τομή μεγέθους k . Αυτό παραβιάζει την αμετάβλητη συνθήκη 4.

Άλλη μια διαφορά ανάμεσα στις δύο μεθόδους είναι ότι ο τοπικός ισοζυγισμός αποσκοπεί στο να βρεθεί μια μεγιστική μαύρη κορυφή και αν αυτή δεν βρεθεί, να ισοζυγίσει τον βαθμό της μαύρης κορυφής που επεξεργάζεται, ώστε να συνεχίσει ο υπολογισμός κλειστότητας, ώστε να προκύψει μια μεγιστική μαύρη κορυφή από τη μία ή την άλλη μέθοδο. Ο καθολικός ισοζυγισμός βαθμού σκοπεύει στο να ισοζυγίσει τις μαύρες κορυφές που δεν κατάφερε να ισοζυγίσει η τοπική μέθοδος, ώστε να χρησιμοποιηθεί η μέθοδος του υπολογισμού κλειστότητας και να βρεθεί ένα μετασχηματισμός που θα οδηγήσει στη μείωση των συνεκτικών στοιχείων του τελευταίου δέντρου.

Ξεκινάμε το βήμα του ισοζυγισμού (Βήμα 8 στον Αλγόριθμο 1) με μια μέθοδο απόφασης. Με αυτή την μέθοδο θα αποφασίσουμε αν μπορούμε να πραγματοποιήσουμε τον τοπικό ισοζυγισμό, ή αν θα βγούμε από την επανάληψη (Βήμα 3) και θα εκτελέσουμε τον καθολικό ισοζυγισμό στη μαύρη κορυφή αργότερα (Βήμα 11). Έστω ότι έχουμε σε ένα δέντρο T_i μια εμφάνιση του b , το σύνολο εγκλεισμού που βρήκαμε στο προηγούμενο βήμα και έχουμε

συνθλίψει σε μία κορυφή, με βαθμό $d + 2$, $d > 0$ και τις κορυφές u_1, \dots, u_{d+2} να είναι οι άσπρες κορυφές των μερικών υπέρ-ακμών που καταλήγουν στο b στο T_i . Αφού το b έχει $k + 1$ εμφανίσεις και συνολικό βαθμό σε όλα τα δέντρα ίσο με $2k$ (το κατευθυνόμενο γράφημα που κατασκευάσαμε είναι γράφημα Euler, επομένως ο εισερχόμενος βαθμός ισούται με τον εξερχόμενο, άρα ο συνολικός βαθμός ισούται με $2k$), τότε υπάρχουν d εμφανίσεις του b που είναι φύλλα και μπορούμε να τις ταιριάξουμε με μία από τις d περισσευούμενες ακμές της εμφάνισης του b με βαθμό $d + 2$. Αφαιρούμε αυτές τις d μερικές υπέρ-ακμές των φύλλων b και τις βάζουμε στο δέντρο T_i . Έστω ότι αυτές τις μερικές υπέρ-ακμές τις συμβολίζουμε με e_1, \dots, e_d και την άσπρη υπέρ-ακμή στο άλλο άκρο τους με w_1, \dots, w_d . Ελέγχουμε αν το b είναι μεγιστική μαύρη κορυφή με τον εξής τρόπο:

Διασχίζουμε όλες κορυφές w_j και την b προς τα πάνω, με σειρά round-robin, σημειώνοντας τις άσπρες κορυφές που έχουμε διασχίσει και προσπερνώντας τις μαύρες. Για κάθε προσπέλαση, λέμε ότι είναι επιτυχής όταν συναντήσει μια σημειωμένη άσπρη κορυφή ή τη μαύρη κορυφή b , διαφορετικά αν συναντήσει μια κορυφή εκτός του $Comp$ αποτυγχάνει.

Η διαδικασία έχει δύο συνθήκες τερματισμού. Όταν συναντήσουμε δύο διασχίσεις που έχουν αποτύχει, στην προκείμενη περίπτωση υπάρχει τουλάχιστον μία άσπρη κορυφή στους θεμελιώδεις κύκλους που είναι εκτός του $Comp$. Διαφορετικά, όλες οι άσπρες κορυφές στους θεμελιώδεις κύκλους που σχηματίστηκαν είναι μέσα στο $Comp$.

Η διαδικασία αυτή θα τρέξει για όλα τα δέντρα που έχουμε μια εμφάνιση με βαθμό μεγαλύτερο του 2. Το αποτέλεσμα της μεθόδου μας δίνει τη μια σημαντική πληροφορία: όταν μετά τον ισοζυγισμό τρέξουμε τον υπολογισμό κλειστότητας, ο υπολογισμός κλειστότητας δεν θα βρει ένα σύνολο εγκλεισμού με επιτυχία αν σε έστω ένα δέντρο βρήκαμε μια κορυφή εκτός του $Comp$. Βλέποντας τον Αλγόριθμο 1, με αυτόν τον τρόπο αποφασίζουμε αν μπορούμε στον επόμενο υπολογισμό κλειστότητας να βρούμε έναν μετασχηματισμό δέντρων που θα έχει ως αποτέλεσμα να μην τελειώσουμε τον υπολογισμό κλειστότητας.

Τοπικός Ισοζυγισμός Βαθμού Συνεχίζοντας από το σημείο που σταματήσαμε στο προηγούμενο βήμα, ο τοπικός ισοζυγισμός ξεκινάει με την κορυφή b να έχει $d + 2 + d = 2d + 2$ υπέρ-ακμές να προσπίπτουν σε αυτή.

Στη προηγούμενη μέθοδο απόφασης, σημειώναμε τις διασχισμένες κορυφές και η προσπέλαση από μια κορυφή w_i έληγε όταν συναντούσαμε μια διασχισμένη κορυφή. Έχουμε τις εξής περιπτώσεις:

1. Η προσπέλαση της κορυφής w_i σταμάτησε γιατί κατέληξε στην κορυφή b .
2. Η προσπέλαση της κορυφής w_i σταμάτησε γιατί κατέληξε σε κορυφή που είχε σημειωθεί από την προσπέλαση από την κορυφή $u \neq b$.
3. Η προσπέλαση της κορυφής w_i σταμάτησε γιατί κατέληξε σε μια κορυφή που σημειώθηκε από την κορυφή b .

Παρατηρούμε ότι σε όλες τις περιπτώσεις, έχουμε βρει ένα μονοπάτι από το w_i στο b . Επομένως, μπορούμε να βρούμε πάροουμε μια οποιαδήποτε υπέρ-ακμή που προσπίπτει στο b , η οποία όμως να μην ανήκει στο μονοπάτι αυτό, και να την ενώσουμε με την $w_i - b$ ενώνοντας τις δύο εμφανίσεις της b . Έτσι έχουμε μειώσει τον βαθμό του b κατά 1.

Τέλος, θα πάροουμε δύο μερικές υπέρ-ακμές και θα τις ενώσουμε μεταξύ τους πάνω στην b . Η υπέρ-ακμή που δημιουργείται, μπορεί να χρησιμοποιηθεί ως seed για τον επόμενο υπολογισμό κλειστότητας.

Καθολικός Ισοζυγισμός Βαθμού Η υπορουτίνα του καθολικού ισοζυγισμού βαθμού ισοζυγίζει τον βαθμό όλων των μεγιστικών μαύρων κορυφών που έχουμε βρει σε αυτό το γύρο, σε όλα τα συνεκτικά στοιχεία που έχουμε επεξεργαστεί. Έστω το σύνολο $\mathcal{B} = \{b_1, \dots, b_r\}$ των μεγιστικών μαύρων κορυφών που πρέπει να ισοζυγίσουμε στο δέντρο T_i . Για κάθε κορυφή b_j στο δέντρο T_i λέμε ότι έχει βαθμό $d_j > 2$. Σκοπός μας είναι να ταιριάξουμε τις $d_j - 2$ υπέρ-ακμές των εμφανίσεων-φύλλων του b_j με υπέρ-ακμές που προσπίπτουν στο b_j . Στόχος μας είναι πάλι να ισοζυγίσουμε τις ακμές των μαύρων κορυφών που δεν είναι ισοζυγισμένες και να το κάνουμε αυτό σε όλα τα δέντρα T_1, \dots, T_k χωρίς να δημιουργηθούν κύκλοι και να μην αποσυνδεθούν, καθώς και στα συνεκτικά στοιχεία του T_{k+1} χωρίς να δημιουργηθούν κύκλοι ή να αυξηθεί ο αριθμός των συνεκτικών στοιχείων (δηλαδή κάποιο συνεκτικό στοιχείο αποσυνδεθεί).

Ορίζουμε την έννοια των υποδέντρων-απογόνων (descendant subtrees) μια μαύρης κορυφής $b_j \in \mathcal{B}$ ως το υποδέντρο με ρίζα μια άσπρη κορυφή v που είναι παιδί της b , προσπερνώντας τις μαύρες κορυφές. Παρατηρούμε ότι το b έχει $d_j - 1$ υποδεντρα-απόγονους. Έστω ένα τέτοιο υποδέντρο το οποίο δεν περιέχει καμιά άλλη κορυφή από το \mathcal{B} και καμιά από τις άσπρες κορυφές των υπέρ-ακμών των εμφανίσεων-φύλλων της b_j . Αυτό το υποδέντρο το ονομάζουμε φιλικό υποδέντρο-απόγονο. Τότε, μπορούμε να ταιριάξουμε την υπέρ-ακμή (b_j, v) , όπου το v είναι η ρίζα του φιλικού υποδέντρου-απόγονου, με μια από τις υπέρ-ακμές των εμφανίσεων-φύλλα της b_j , έστω (w, b_j) . Ενώνουμε την ακμή (w, b_j) στην εμφάνιση της κορυφής w στο δέντρο T_i , και ενώνουμε το φιλικό υποδέντρο-απόγονο του b_j με τη μερική υπέρ-ακμή (w, b_j) . Έτσι δημιουργούμε μια εμφάνιση της κορυφής b_j με βαθμό 2 και μια υπέρ-ακμή $w - b_j - v$ που καταλήγει στο φιλικό υποδέντρο-απόγονο. Αφού ολοκληρώσουμε αυτή τη διαδικασία, μπορούμε να συνθλίψουμε για τη συνέχεια του καθολικού ισοζυγισμού το υποδέντρο με ρίζα v καθώς και την υπέρ-ακμή $w - b_j - v$ σε μια κορυφή, αφού δεν πρόκειται να αλλάξει και να επηρεάσει τους υπολογισμούς μας.

2.2.3 Πολυπλοκότητα Αλγόριθμου

Παρακάτω θα χρησιμοποιήσουμε τον συμβολισμό $\|B\|$ συμβολίσουμε τον αριθμό των ακμών που έχουν και τα δύο τους άκρα στο σύνολο κορυφών B . Επίσης, ως $w(B)$ θα συμβολίσουμε τον αριθμό των άσπρων κορυφών στο σύνολο κορυφών B .

Οι υπορουτίνες που περιγράψαμε πιο πάνω έχουν τις εξής πολυπλοκότητες:

Υπολογισμός κλειστότητας Ο υπολογισμός κλειστότητας έχει δύο πιθανά αποτελέσματα. Είτε να βρεθεί ένα σύνολο εγκλεισμού B , είτε να βοηθήσει να βρεθεί μια σειρά μετασχηματισμών που θα ενώσουν ένα μέρος των συνεκτικών στοιχείων. Στη πρώτη περίπτωση, έχει πολυπλοκότητα $O(\|B\| + k)$, στη δεύτερη $O(nk)$.

Τοπικός ισοζυγισμός Ο τοπικός ισοζυγισμός έχει δύο πιθανά αποτελέσματα, να ισοζυγίσει τη μαύρη κορυφή που προέκυψε, είτε να σταματήσει γιατί βρέθηκε μια μεγιστική μαύρη κορυφή. Στη πρώτη περίπτωση έχει χρονική πολυπλοκότητα $O(w(B)k)$ και στη δεύτερη περίπτωση $O(w(Comp)k)$.

Καθολικός ισοζυγισμός Ο καθολικός ισοζυγισμός επεξεργάζεται ένα δέντρο τη φορά σε ένα βήμα του αλγορίθμου. Συνθλίβει υπέρ-ακμές και τα δέντρα απογόνους κατά την εκτέλεση του, έτσι για ένα δέντρο δεν πρόκειται να ξανά επεξεργαστεί την ίδια υπέρ-ακμή. Επομένως, έχουμε πολυπλοκότητα $O(n)$ για να ισοζυγίσουμε τις μαύρες κορυφές ενός δέντρου, και συνολική πολυπλοκότητα $O(nk)$ για να ισοζυγίσουμε όλες τις μαύρες κορυφές σε όλα τα T_1, \dots, T_{k+1} δέντρα.

Μετασχηματισμοί δέντρων Η εύρεση της σωστής αλυσίδας μετασχηματισμών για ενωθούν όλα τα συνεκτικά στοιχεία που μπορούμε να ενώσουμε, καθώς και η εκτέλεση τους, έχει χρονική πολυπλοκότητα $O(n + m)$.

Βλέπουμε ότι στον Αλγόριθμο 1 ο **υπολογισμός κλειστότητας** καλείται σε δύο διαφορετικές περιπτώσεις:

1. καθώς αναζητάμε μια μεγιστική μαύρη κορυφή (Βήμα 4). Σε αυτή την περίπτωση, έχουμε χρονική πολυπλοκότητα $O(\|B\| + k)$. Σε όλο τον αλγόριθμο, ο όρος $\|B\|$ αθροίζει συνολικά σε (m) , διότι με τη λήξη του υπολογισμού κλειστότητας, το σύνολο B θα συνθλιφτεί, και οι ακμές του δεν θα ξαναχρησιμοποιηθούν. Ο όρος k σε όλο το γύρο αθροίζει σε $O(nk)$, διότι κάθε συνεκτικό στοιχείο έχει τουλάχιστον μια άσπρη κορυφή.
2. καθώς αναζητάμε σειρά μετασχηματισμών για τα δέντρα (Βήμα 13). Όλη η επανάληψη (Βήμα 12) έχει συνολική πολυπλοκότητα $O(nk)$.

Ο **τοπικός ισοζυγισμός** καλείται πιθανώς πολλές φορές για κάθε συνεκτικό στοιχείο. Έστω ότι σε διαδοχικές κλήσεις βρίσκουμε τα σύνολα εγκλεισμού B_1, \dots, B_h . Για ένα βήμα η συνολική πολυπλοκότητα του τοπικού ισοζυγισμού για ένα συνεκτικό στοιχείο θα είναι $\sum_{i=1}^h O(w(B_i)k) + O(w(Comp)k)$. Καθώς οι άσπρες κορυφές όλων των μαύρων κορυφών που ισοζυγίζουμε σε ένα συνεκτικό στοιχείο $Comp$ δεν επαναλαμβάνονται (αφού οι κορυφές θα συνθλιφτούν στην B_j), τότε το πρώτο μέρος αθροίζει σε $O(w(Comp)k)$, επομένως όλη πολυπλοκότητα για ένα $Comp$ είναι $O(w(Comp)k)$, άρα η συνολική πολυπλοκότητα για όλα τα συνεκτικά στοιχεία, δηλαδή για έναν ολόκληρο γύρο, είναι $O(nk)$.

Το βήμα του μετασχηματισμού έχει πολυπλοκότητα $O(n + m)$, για την εύρεση και εφαρμογή του μετασχηματισμού για όλα τα συνεκτικά στοιχεία. Λόγω της προεπεξεργασίας Nagamochi-Ibaraki, με τις ιδιότητες που ορίζει το Θεώρημα 4, περιορίζοντας τις ακμές που χρησιμοποιούμε στον i -οστό γύρο μόνο στις $O(ni)$ ακμές των πρώτων i Nagamochi-Ibaraki δασών, μπορούμε να φράξουμε τον αριθμό των χρησιμοποιούμενων ακμών σε $m = O(nk)$, άρα το βήμα του μετασχηματισμού έχει χρονική πολυπλοκότητα $O(nk)$. Η προεπεξεργασία αυτή έχει μια επιβάρυνση της τάξης $O(m)$ στον τελικό αλγόριθμο.

Λόγω των παραπάνω, ο αλγόριθμός για κάθε βήμα έχει πολυπλοκότητα $O(nk)$. Σε κάθε βήμα, ο αριθμός των συνεκτικών στοιχείων ενός δέντρου υποδιαιρείται, άρα έχουμε $O(\log n)$ βήματα για την κατασκευή ενός δέντρου T_i . Επιπλέον, στα κατευθυνόμενα γραφήματα ισχύει ότι $m \geq kn$, αφού κάθε κορυφή έχει τουλάχιστον k εισερχόμενο βαθμό. Αντίστοιχα, αφού το αρχικό μας μη κατευθυνόμενο γράφημα έχει τον μισό αριθμό ακμών m από το κατευθυνόμενο που δημιουργήσαμε, ισχύει ότι $m \geq kn/2$. Άρα $O(kn) = O(m)$ · μπορούμε να εφαρμόσουμε αυτό το συμπέρασμα στην πολυπλοκότητα του αλγόριθμου, επομένως έχουμε:

Θεώρημα 8. [7] *Ο συνολικός χρόνος για την εύρεση μιας τομής Steiner για ένα σύνολο S μπορεί να βρεθεί σε χρόνο $O(nk \log n + m) = O(m \log n)$, όπου k το μέγεθος της τομής Steiner.*

Γνωρίζουμε, επιπλέον, ότι μπορούμε να βρούμε την ελάχιστη τομή που χωρίζει μια κορυφή v από μια κορυφή r χτίζοντας k δέντρα, όπου k είναι το μέγεθος της ελάχιστης τομής που χωρίζει τις κορυφές $v-r$. Επομένως, δεδομένης μιας ρίζας r και ενός συνόλου S , μπορούμε να βρούμε για κάθε κορυφή $v \in S$, την ελάχιστη τομή που τη χωρίζει από τη ρίζα r , χτίζοντας k δέντρα, όπου k είναι η μέγιστη ελάχιστη τομή ανάμεσα σε μια κορυφή $v \in S$ και στο r . Επομένως έχουμε συνολική χρονική πολυπλοκότητα $O(nk^2 \log n)$.

Θεώρημα 9. [7] Για κάθε σύνολο κορυφών $X \subseteq V$ ενός γραφήματος G , μπορούμε να βρούμε την ελάχιστη τομή που χωρίζει κάθε κορυφή $v \in X$ από τη ρίζα r σε χρόνο $O(nk^2 \log n + m) = O(mk \log n)$.

2.3 Κατασκευή Δέντρων Gomory-Hu

Παρουσιάσαμε έναν αλγόριθμο που βρίσκει μια τομή Steiner για ένα σύνολο S στο γράφημα G . Θυμίζουμε τις παρακάτω ιδιότητες του αλγόριθμου για τομές Steiner οι οποίες θα είναι χρήσιμες για την κατασκευή των Gomory-Hu δέντρων. Το σύνολο B του υπολογισμού κλειστότητας που βρίσκουμε κατά την κατασκευή του i -οστού δέντρου είναι μια τομή Steiner μεγέθους $i - 1$ για το σύνολο S που θέλαμε να χωρίσουμε και οι άσπρες κορυφές του B έχουν ακμοσυνεκτικότητα $i - 1$ με το r . Επίσης, το νέο σύνολο B μπορούμε να το συνθλίψουμε και αυτό και να συνεχίσουμε τους υπολογισμούς μας, μέχρι να βρούμε ένα άλλο σύνολο εγκλεισμού B . Ο αλγόριθμος που δείξαμε έχει σχεδόν γραμμική (quasi-linear) χρονική πολυπλοκότητα $\tilde{O}(mc \log n)$, όπου c είναι το μέγεθος της τομής. Η αποδοτικότητά του αλγόριθμου κατασκευής των δέντρων Gomory-Hu έγκειται στο γεγονός ότι μπορούμε ανάμεσα σε γύρους να επαναχρησιμοποιήσουμε τους υπολογισμούς από προηγούμενα βήματα.

Πιο συγκεκριμένα, στον κλασικό αλγόριθμο των Gomory-Hu, όπως τον έχουμε περιγράψει, επιλέγουμε αυθαίρετα ένα ζευγάρι κορυφών μέσα σε μία υπέρ-κορυφή S , εκτελούμε έναν υπολογισμό μέγιστης ροής για αυτές τις κορυφές σε ένα τροποποιημένο γράφημα $G(S)$, που περιέχει όλες τις υπέρ-κορυφές του Gomory-Hu δέντρου \mathcal{T} συνθλιμμένες εκτός από την S , και χωρίζουμε την υπέρ-κορυφή ανάλογα στο \mathcal{T} . Ο αλγόριθμος των Bhalgat et al. που περιγράψαμε πιο πάνω αντιμετωπίζει το σύνολο S ως ένα σύνολο Steiner, δηλαδή επιλέγει αυθαίρετα μια κορυφή r ως τη ρίζα των δέντρων και εκτελεί τον αλγόριθμο εύρεσης της ελάχιστης Steiner τομής για το σύνολο S . Αφού επεξεργαστούμε την κορυφή S με τον αλγόριθμο Steiner, τη προσθέτουμε στο δέντρο \mathcal{T} και συνεχίζουμε με τις υπόλοιπες κορυφές του Gomory-Hu δέντρου που έχουν πάνω από μια κορυφή.

Προκειμένου να περιγράψουμε πως χειριζόμαστε την κορυφή S στην παραπάνω περιγραφή, εισάγουμε κάποιες έννοιες.

Ορισμός 11 (Μάρτυρες ελαχιστικής ελάχιστης τομής). Έστω η ελαχιστική ελάχιστη τομή $(B, V(G(S)) - B)$ που έχουμε βρει σε σχέση με τη ρίζα r στο γράφημα $G(S)$. Κάθε κορυφή $v \in B$ ονομάζεται μάρτυρας ελαχιστικής ελάχιστης τομής. Για μια τομή B συμβολίζουμε το σύνολο των μαρτύρων της ως $w(B)$.

Ορισμός 12 (Laminar οικογένεια συνόλων). Μια οικογένεια συνόλων \mathcal{F} ονομάζεται laminar αν κάθε ζεύγος συνόλων είναι είτε ξένα μεταξύ τους είτε σχετίζονται μέσω της σχέσης υποσυνόλου, δηλαδή $S \in \mathcal{F}(S)$ και $S_1, S_2 \in \mathcal{F}(S), \forall S_1, S_2 \in S \iff S_1 \cap S_2 = \emptyset$ ή $S_1 \cap S_2 = S_1$ ή $S_1 \cap S_2 = S_2$.

Βλέπουμε επομένως ότι τα σύνολα ελαχιστικών ελάχιστων τομών σχηματίζουν μια οικογένεια Laminar με αρχικό σύνολο το σύνολο που περιέχει όλες τις κορυφές. Λόγω της σχέσης που ορίζει η οικογένεια Laminar μπορούμε να δομήσουμε την οικογένεια αυτή ως ένα δέντρο. Δηλαδή, ρίζα θα είναι το σύνολο όλων των κορυφών και κάθε κορυφή αντιπροσωπεύει μια ελαχιστική ελάχιστη τομή B που έχει ως παιδί μια άλλη τομή υποσύνολο του B . Είναι φυσικό μια τομή να έχει πολλές τομές των οποίων είναι υποσύνολο B_1, \dots, B_n , για τις οποίες θα ισχύει $B \subset B_1 \subset B_2 \subset \dots \subset B_n$, αλλά ως γονέα στο δέντρο που σχηματίζεται θα έχει το σύνολο αμέσως "μικρότερο" σύνολο σε αυτή την σχέση, το B_1 .

Αφού τρέξουμε τον αλγόριθμο τομής Steiner για ένα σύνολο S στο $G(S)$ και βρούμε την ακμοσυνεκτικότητα όλων των κορυφών σε σχέση με την κορυφή r , αντικαθιστούμε την κορυφή S με ένα δέντρο, έστω ότι το ονομάζουμε \mathcal{L} , που έχει την ίδια δομή με το δέντρο που περιγράψαμε πιο πάνω. Κάθε κορυφή του \mathcal{L} που συσχετίζεται με μια τομή B τη συμβολίζουμε ως \mathcal{L}_B . Η S μετά την τομή στο Gomory-Hu δέντρο διαμορφώνεται ως εξής:

- Η κορυφή \mathcal{L}_B αντί για όλες τις κορυφές που ανήκουν στην τομή B , περιέχει μόνο τις κορυφές-μάρτυρες του B που ανήκουν στο αρχικό γράφημα, δηλαδή τις κορυφές $w(B) \cap V$.
- Στη ρίζα του \mathcal{L} , που θεωρείται η τετριμμένη τομή που περιέχει όλες τις κορυφές, ορίζουμε ως $w(B) \cap V = \{r\}$.
- Έστω ότι η συνθλιμμένη κορυφή T_i του $G(S)$, που είναι κορυφή που δεν ανήκει στο αρχικό γράφημα G , είναι μάρτυρας του B , $T_i \in w(B)$. Το υποδέντρο T_i θα ενωθεί με την κορυφή \mathcal{L}_B στο δέντρο Gomory-Hu με μια ακμή με βάρος ίσο με το βάρος της ακμής που ένωνε την S με το T_i πριν το βήμα αυτό.
- Έστω οι κορυφές \mathcal{L}_B και \mathcal{L}'_B όπου η \mathcal{L}_B είναι γονέας της \mathcal{L}'_B στο \mathcal{L} . Οι ακμή που τις συνδέει στο Gomory-Hu δέντρο θα έχει βάρος ίσο με το μέγεθος της τομής B' .

Στο νέο δέντρο Gomory-Hu, επιλέγουμε μια κορυφή S που περιέχει περισσότερες από μία κορυφές και ξανατρέχουμε τον αλγόριθμο στο $G(S)$, και επαναλαμβάνουμε αυτή τη διαδικασία μέχρι να καταλήξουμε σε ένα δέντρο Gomory-Hu με κορυφές που περιέχουν μόνο μία κορυφή του αρχικού γραφήματος G . η διαδικασία της επιλογής μπορεί να υλοποιηθεί με μια ουρά όπως βλέπουμε στον παρακάτω Αλγόριθμο 2.

Αλγόριθμος 2 Ο τελικός αλγόριθμος Gomory-Hu με τη χρήση του υπολογισμού εύρεσης τομής Steiner

- 1: Αρχικοποιούμε το δέντρο Gomory-Hu $\mathcal{T} = (\{V\}, \emptyset)$
 - 2: Αρχικοποιούμε την ουρά Q με το στοιχείο (V, G)
 - 3: **Όσο** Q δεν είναι άδεια **κάνε**
 - 4: Παίρνουμε το πρώτο αντικείμενο $(S, G(S))$ από την ουρά Q
 - 5: **Αν** $|S| > 1$ **τότε**
 - 6: Επιλέγουμε τυχαία μια κορυφή r από το σύνολο S
 - 7: Με ρίζα την r τρέχουμε τον υπολογισμό Steiner ώστε να βρούμε το δέντρο Laminar των ελάχιστων τομών στο γράφημα $G(S)$ και αντικαθιστούμε την S με το δέντρο \mathcal{L} στο \mathcal{T} , όπως περιγράψαμε πιο πάνω.
 - 8: **Για κάθε** τομή B που βρήκαμε στον παραπάνω υπολογισμό **κάνε**
 - 9: Εισάγουμε το στοιχείο $(B, G(B))$ στην ουρά Q , όπου το $G(B)$ είναι το $G(S)$ αφού συνθλίψουμε τις κορυφές $G(S) - B$ σε μία κορυφή.
 - 10: **Τέλος επανάληψης**
 - 11: **Τέλος**
 - 12: **Τέλος επανάληψης**
-

Ας δούμε τώρα τα υποπροβλήματα που δημιουργούνται στον Αλγόριθμο 2, δηλαδή στοιχεία που μπαίνουν στην ουρά Q και σε αυτά τρέχουμε έναν υπολογισμό ελάχιστης τομής Steiner, και ας τα χωρίσουμε σε επίπεδα. Το πρώτο επίπεδο $Layer(0)$ περιέχει το πρόβλημα (V, G) . Το δεύτερο επίπεδο $Layer(1)$ αποτελείται από όλες τις κορυφές που βρήκαμε κατά την επεξεργασία της κορυφής του επιπέδου $Layer(0)$. Το ίδιο ισχύει για κάθε επίπεδο

i , δηλαδή το επίπεδο $Layer(i)$ περιέχει όλα τα υποπροβλήματα που δημιουργήθηκαν από ελάχιστες τομές που βρήκαμε σε κορυφές του επιπέδου $Layer(i - 1)$. Επίσης για κάθε επίπεδο $Layer(i - 1)$ υπάρχει ένα σχετικό δέντρο Gomory-Hu. Αν επεξεργαστούμε όλες τις κορυφές σε αυτό το δέντρο, βρίσκουμε το Gomory-Hu δέντρο για το επίπεδο $Layer(i)$. Θα παρουσιάσουμε τώρα την απόδειξη της χρονικής πολυπλοκότητας του αλγόριθμου.

Θεώρημα 10. [7] *Η χρονική πολυπλοκότητα του αλγόριθμου κατασκευής του δέντρου Gomory-Hu για το γράφημα $G = (V, E)$ είναι $\tilde{O}(mc)$, όπου $c = \max_{u,v \in V} c(u, v)$.*

Η αποδοτικότητα του αλγόριθμου αποδίδεται εν μέρει στην ιεραρχία των υποπροβλημάτων που δημιουργείται. Πιο συγκεκριμένα, όλα τα υποπροβλήματα σε ένα επίπεδο $Layer(i)$ έχουν συνολική πολυπλοκότητα $\tilde{O}(mc)$. Εκ πρώτης όψεως, αυτό ισχύει αν τα υποπροβλήματα που ανήκουν στο επίπεδο $Layer(i)$ δεν μοιράζονται καμιά ακμή μεταξύ τους. Αυτό όμως δεν ισχύει. Παρόλα αυτά, μπορεί να αποδειχτεί σε κάθε επίπεδο χρησιμοποιούνται συνολικά $O(m)$ ακμές συνολικά σε όλα τα υποπροβλήματα με αποτέλεσμα να μην αλλάζει η χρονική πολυπλοκότητα.

Λήμμα 10. [7] *Ο συνολικός αριθμός όλων των ακμών που χρησιμοποιούνται στα υποπροβλήματα ενός επιπέδου είναι $O(m)$.*

Απόδειξη. Οι ακμές που επαναχρησιμοποιούνται σε υποπροβλήματα του ίδιου επιπέδου είναι οι ακμές που ανήκουν στις τομές που δημιουργήθηκαν στο προηγούμενο επίπεδο. Για παράδειγμα, έστω το Gomory-Hu δέντρο \mathcal{T} που αντιστοιχεί στο επίπεδο $Layer(i)$. Έστω οι κορυφές S και S' που ενώνονται από την ακμή e στο \mathcal{T} . Η ακμή e χρησιμοποιείται και στα δύο γραφήματα $G(S)$ και $G(S')$ στους υπολογισμούς που γίνονται για το επίπεδο $Layer(i)$, και μπορεί να διαχωριστεί σε πολλαπλές ακμές, των οποίων ο αριθμός φράσσεται από το βάρος της ακμής στο \mathcal{T} .

Αρκεί να αθροίσουμε για όλες τις ακμές e , τον αριθμό των τομών στις οποίες η e είναι παρούσα. Αυτό το άθροισμα φράσσεται από το άθροισμα $\sum_{i=1}^{n-1} c_i$, όπου $c_i, = 1, 2, \dots$ είναι το βάρος των $n - 1$ ακμών του τελικού Gomory-Hu δέντρου.

Θέτουμε ως ρίζα του Gomory-Hu δέντρου $\mathcal{T} = (V, \mathcal{E})$ μια τυχαία κορυφή και ορίζουμε τη συνάρτηση $l : \mathcal{E} \mapsto V$ τέτοια ώστε l να αντιστοιχίζει την ακμή e στο άκρο της e που βρίσκεται πιο μακριά από τη ρίζα. Είναι εύκολο να δούμε ότι η l είναι συνάρτηση 1-1. Για κάθε ακμή $e = (u, v) \in E$, το βάρος της e στο δέντρο Gomory-Hu αντιστοιχεί στην τομή $c(u, v)$. Χωρίς βλάβη της γενικότητας, λέμε για απλότητα ότι $l(e) = u$.

Εφόσον, η τομή $c(u, v)$ είναι η ελάχιστη τομή $u - v$, και η τομή $(u, V - u)$ με μέγεθος $deg(u)$ είναι μια $u - v$ τομή η οποία όμως δεν είναι ελάχιστη, ισχύει ότι $c(u, v) \leq deg(u) = deg(l(e))$. Αθροίζοντας για όλες τις ακμές $e \in \mathcal{E}$ και δεδομένου ότι η l είναι 1-1, έχουμε $\sum_{e \in \mathcal{E}} w(e) \leq \sum_{v \in V} deg(v) = 2m$. \square

Ένα υποπρόβλημα έχει χρονική πολυπλοκότητα $O(m'c \log n)$, λόγω του Θεωρήματος 9, όπου m' ο αριθμός των ακμών του γραφήματος του υποπροβλήματος. Λόγω του Λήμματος 10, για τη διεκπεραίωση όλων των υποπροβλημάτων ενός επιπέδου απαιτείται $O(mc \log n)$. Επομένως, η συνολική πολυπλοκότητα του αλγόριθμου εξαρτάται από τον αριθμό των επιπέδων. Παρατηρούμε ότι

Λήμμα 11. [7] *Έστω το υποπρόβλημα $(C, G(C))$ και το υποπρόβλημα γονέας $(P, G(P))$ και έστω $|P|$ και $|C|$ ο αριθμός των κορυφών που είναι παρόν στον αρχικό γράφημα (και δεν είναι κορυφή που περιέχει πάνω από μία κορυφές στο \mathcal{T}) στα σύνολα P και C . Τότε $Pr(|C| \geq |P|/2) \leq 1/2$.*

Απόδειξη. Η τομή C είναι μια ελαχιστική ελάχιστη τομή που χωρίζει την κορυφή r από μια κορυφή $v \in C$ στο πρόβλημα $(P, G(P))$. Παρατηρούμε την τομή $(C, G(P) \setminus C)$ από τη μεριά του v . Βλέπουμε ότι το όλες οι κορυφές στο σύνολο C έχουν ακμοσυνεκτικότητα από το v μεγαλύτερη από το μέγεθος της τομής του C , που είναι ίση με την ακμοσυνεκτικότητα $r-v$.

Ταξινομούμε τις κορυφές του P με βάση την ακμοσυνεκτικότητά τους από το v στο $G(P)$ με αύξουσα σειρά. Αν η κορυφή r βρίσκεται στις πρώτες $|P|/2$ κορυφές της ταξινόμησης, τότε οι κορυφές μετά από την r είναι η κορυφές που ανήκουν στο υποπρόβλημα C . Αφού η r διαλέχτηκε τυχαία από τις $|P|$ κορυφές, η πιθανότητα αυτή είναι μικρότερη του $1/2$, επομένως αντιστοιχεί σε $Pr(|C| \geq |P|/2) \leq 1/2$. \square

Θεώρημα 11. [7] *Με πιθανότητα $1 - 1/n$, ο αριθμός των επιπέδων υποπροβλημάτων είναι $O(\log n)$*

Απόδειξη. Θεωρούμε ένα μονοπάτι με μήκος l στο δέντρο των υποπροβλημάτων που ξεκινάει από το υποπρόβλημα (V, G) στο επίπεδο $Layer(0)$ και εκτείνεται μέχρι το υποπρόβλημα φύλλο $(C_l, G(C_l))$. Κάθε ακμή στο έχει τα άκρα τις σε ένα υποπρόβλημα $(C_j, G(C_j))$ στο επίπεδο $Layer(j)$ και στο υποπρόβλημα γονέα του $(C_j, G(C_j))$, το $(C_{j-1}, G(C_{j-1}))$ στο επίπεδο $Layer(j-1)$. Θα ορίσουμε την τυχαία μεταβλητή X_j ως εξής:

$$X_j = \begin{cases} 1, & |C_j| \geq |C_{j-1}|/2 \\ 0, & \text{διαφορετικά} \end{cases}$$

Επομένως, η τ.μ. X_j μας δείχνει αν στο υποπρόβλημα στο επίπεδο $Layer(j)$ του μονοπατιού, είχαμε έναν "καλό" γύρο, δηλαδή, αν επιλέξαμε σωστά τη ρίζα r στο $(C_j, G(C_j))$ έτσι ώστε να υποδιπλασιάσουμε το μέγεθος του υποπροβλήματος στο επόμενο επίπεδο ($X_j = 0$), ή αν είχαμε έναν "κακό" γύρο ($X_j = 1$). Παρατηρούμε ότι οι τ.μ. X_1, X_2, \dots, X_j είναι ανεξάρτητες μεταξύ τους, διότι η τιμή της τ.μ. εξαρτάται αποκλειστικά από την τυχαία επιλογή της ρίζας r .

Έστω ότι $l > C \log_2 n, C > 0$. Ένας καλός γύρος υπό διπλασιάζει το μέγεθος του προβλήματος, επομένως μπορούμε να έχουμε το πολύ $\log_2 n$ καλούς γύρους στα πρώτα $\lfloor C \log_2 n \rfloor$ επίπεδα. Επομένως, για τα πρώτα $\lfloor C \log_2 n \rfloor$ επίπεδα πρέπει να έχουμε τουλάχιστον $\lfloor (C-1) \log_2 n \rfloor$ κακούς γύρους. Ορίζουμε ως $X = \sum_{i=0}^{\lfloor C \log_2 n \rfloor} X_i$ τον αριθμό των κακών γύρων που έχουμε στα πρώτα $\lfloor C \log_2 n \rfloor$ επίπεδα. Έτσι προκύπτει:

$$\begin{aligned} E[X] &= \sum_{i=1}^{\lfloor C \log_2 n \rfloor} E[X_i], \text{ λόγω της ανεξαρτησίας των τ.μ.} \\ &\leq \sum_{i=1}^{\lfloor C \log_2 n \rfloor} Pr[X_i = 1], X_i \text{ είναι τ.μ. Bernoulli} \\ &\leq \frac{C}{2} \log_2 n = U \end{aligned}$$

Αφού X_i είναι ανεξάρτητες τ.μ. που παίρνει τιμές 0 ή 1, με $\mu = E[X]$ και $\mu \leq U$, λόγω των ορίων Chernoff (για φραγμένη αναμενόμενη τιμή) [22], για $\delta > 0$ έχουμε:

$$Pr[X \geq (1 + \delta)U] < \left(\frac{e^\delta}{(\delta + 1)^{\delta+1}} \right)^U$$

Θέτουμε ως $C = 11$ και επιλέγουμε το δ κατάλληλα ($\delta = \frac{9}{11}$). Έτσι προκύπτει:

$$\begin{aligned} Pr[X > \frac{20}{11} \frac{11}{2} \log n] &< \left(\frac{e^\delta}{(\delta+1)^{\delta+1}} \right)^U < n^{-2} \implies \\ Pr[X > 10 \log n] &< n^{-2} \end{aligned}$$

Ένα τέτοιο δέντρο μπορεί να έχει το πολύ n . Έτσι χρησιμοποιώντας την ιδιότητα της υποπροσθετικότητας (union bound), η πιθανότητα να έχουμε έστω ένα φύλλο με βάθος $> 11 \log n$ είναι:

$$\begin{aligned} Pr[\text{ένα φύλλο είναι σε βάθος } > 11 \log n] &\leq \\ &= Pr\left[\bigcup_{i=1}^n \text{το φύλλο } i \text{ είναι σε βάθος } > 11 \log n\right] \\ &\leq \sum_{i=1}^n Pr[\text{το φύλλο } i \text{ είναι σε βάθος } > 11 \log n] \\ &\leq n \cdot n^{-2} \leq \frac{1}{n} \end{aligned}$$

Έτσι με πιθανότητα μεγαλύτερη του $1 - \frac{1}{n}$ το δέντρο των υποπροβλημάτων έχει βάθος $O(\log n)$.

□

Επομένως, ο αλγόριθμος έχει χρονική πολυπλοκότητα $O(mk \log n)$ με πιθανότητα $1 - \frac{1}{n}$. Μπορούμε να αγνοήσουμε τον όρο $\log n$, καθώς ο ρυθμός αύξησης του είναι πολύ μικρότερος από τον γραμμικό χρόνο, επομένως μπορούμε να χρησιμοποιήσουμε τον συμβολισμό $\tilde{O}(mk)$.

3. ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΗ ΕΡΕΥΝΑ

Στην Ενότητα 2.2 παρουσιάσαμε έναν αποδοτικό αλγόριθμο των Bhalgat et al. [12] για την κατασκευή Gomory-Hu δέντρων για μη έμβαρα γραφήματα σε χρόνο καλύτερο από $O(n) \cdot T(n, m)$ όπου $T(n, m)$ χρόνος υπολογισμού της ελάχιστης τομής/μέγιστης ροής σε ένα γράφημα με n κορυφές και m ακμές. Ο αλγόριθμος έχει σχεδόν γραμμική χρονική πολυπλοκότητα $\tilde{O}(mk)$ με μεγάλη πιθανότητα, όπου k είναι η μέγιστη τιμή της ελάχιστης τομής ανάμεσα σε όλα τα ζεύγη κορυφών. Στα μη έμβαρα γραφήματα ισχύει ότι $n < k$, επομένως μπορούμε να θεωρήσουμε ότι η πολυπλοκότητα ανάγεται σε $\tilde{O}(mn)$. Επίσης, μπορεί να εφαρμοστεί και σε έμβαρα γραφήματα, όπου όλα τα βάρη είναι ακέραιοι, μετατρέποντας μια ακμή με βάρος w σε w παράλληλες ακμές, καθώς στον αλγόριθμο δεν υπάρχει κάποια υπόθεση που βασίζεται στη μη ύπαρξη παράλληλων ακμών. Παρόλα αυτά η χρονική πολυπλοκότητα $\tilde{O}(mk)$ σε αυτή την περίπτωση δεν είναι αποδοτική, καθώς μπορεί $k \gg n$. Εν συντομία, ο αλγόριθμος θεωρείται αποδοτικός για μη πυκνά γραφήματα, δηλαδή για γραφήματα όπου $m = o(n^2)$, αλλά είναι κυβικής πολυπλοκότητας για πυκνά γραφήματα όπου $m = \Theta(n^2)$.

Πολλές μελέτες στη σχεδίαση αποδοτικών αλγορίθμων για την κατασκευή δέντρων Gomory-Hu αντιμετωπίζουν τη δυσκολία του προβλήματος μέσω της τυχαιότητας επομένως ένα ανοιχτό πρόβλημα για πολύ καιρό ήταν η εύρεση ενός ντετερμινιστικού αποδοτικού αλγορίθμου.

Η μελέτη γύρω από τους αλγόριθμους κατασκευής Gomory-Hu δέντρων έχει συγκεντρώσει επιστημονικό ενδιαφέρον με αποτέλεσμα την ανάπτυξη αποτελεσματικών αλγορίθμων. Οι έρευνες επικεντρώνονται κυρίως σε συγκεκριμένες κλάσεις γραφημάτων όπως απλά γραφήματα [17] [3] [24], μη έμβαρα [15] [4], επίπεδα (planar) [8] και φραγμένου δενδροπλάτους [2] (bounded treewidth).

Για όλες τις κλάσεις γραφημάτων, αλλά και στη γενική περίπτωση (έμβαρα, μη απλά γραφήματα), πλέον έχει σπάσει το κυβικό όριο. Πιο συγκεκριμένα για τη κλάση των μη έμβαρων γραφημάτων, λόγω της έρευνας των Abboud, Krauthgamer και Trabelsi [3], μπορούμε να κατασκευάσουμε ένα δέντρο Gomory-Hu σε σχεδόν τετραγωνικό χρόνο. Η έρευνα αυτή, επίσης, αλλάζει τη μέχρι τώρα κλασική προσέγγιση του προβλήματος. Οι περισσότεροι αλγόριθμοι κατασκευής δέντρων Gomory-Hu, όπως ο αλγόριθμος των Bhalgat et al. που αναλύσαμε, βασίζονται στην τυχαία επιλογή μιας κορυφής και έπειτα λύνουν το πρόβλημα Μέγιστης Ροής Μοναδικής Πηγής (single source maximum flow problem) με έναν αποδοτικό τρόπο. Η τυχαία επιλογή μιας κορυφής ως σημείο βάσης (pivot) της υπορουτίνας αλγόριθμου οδηγεί με μεγάλη πιθανότητα στο λογαριθμικό βάθος του δέντρου υπολογισμού. Η μέθοδος των Abboud et al. εισάγει μια τεχνική επιλογής του σημείου βάσης με ντετερμινιστικό τρόπο, απαλείφοντας έτσι την εξάρτηση της αποδοτικότητας του αλγορίθμου από την τυχαιότητα.

Είναι σημαντικό να σημειωθεί ότι για τη γενική περίπτωση, δηλαδή για την περίπτωση μη κατευθυνόμενων γραφημάτων, έχει αναπτυχθεί αλγόριθμος [1] με πολυπλοκότητα $\tilde{O}(n^2)$, ο οποίος σπάει το όριο που έθεσαν οι Gomory και Hu [11], δηλαδή $O(n) \cdot T(m, n)$. Επίσης, έχουν αναπτυχθεί αλγόριθμοι με σχεδόν γραμμική χρονική πολυπλοκότητα για συγκεκριμένες κλάσεις γραφημάτων, όπως για τα επίπεδα γραφήματα [8] και γραφήματα φραγμένου δενδροπλάτους [2].

Μερικές από τις πρακτικές εφαρμογές των Gomory-Hu δέντρων είναι στον τομέα της μηχανικής όρασης [23], στη βιολογία [21], τμηματοποίηση ιστοσελίδων [18] κα. Επίσης στα δίκτυα και τις τηλεπικοινωνίες, σύμφωνα με τον Hu [14], τα Gomory-Hu δέντρα είναι η

βέλτιστη δομή για το πρόβλημα του Βέλτιστου Επικαλύπτοντος Δέντρου Επικοινωνίας (Optimum Communication Spanning Tree Problem), δηλαδή το επικαλύπτον δέντρο ενός γραφήματος που ελαχιστοποιεί τη συνάρτηση κόστους επικοινωνίας. Αυτό έχει αποτελέσει αφορμή να μελετηθούν οι ιδιότητες των γραφημάτων που επιτρέπουν την ύπαρξη υπο-γραφημάτων Gomory-Hu [20].

Ενδιαφέρον πρόβλημα σχετικό με τα δέντρα Gomory-Hu, επίσης, είναι και η ανάλυση ευαισθησίας ελάχιστων τομών γραφημάτων, δηλαδή πώς οι αλλαγές στα γραφήματα, όπως η προσθήκη μιας ακμής σε ένα γράφημα, επηρεάζει τη μέγστη ροή/ελάχιστη τομή ανάμεσα σε όλα τα ζεύγη κορυφών. Τα δέντρα Gomory-Hu είναι μια δομή που μπορεί να απαντάει σε ερωτήματα ελάχιστων τομών για ένα γράφημα, αλλά δεν υπάρχει τετριμμένος τρόπος για να ανανεώνεται το δέντρο Gomory-Hu σύμφωνα με τις αλλαγές στο αρχικό δέντρο. Το πρόβλημα αυτό έχει μελετηθεί από τη σκοπιά της δημιουργίας δυναμικών δέντρων Gomory-Hu [13], αλλά έχουν ερευνηθεί και άλλες δομές που θα μπορούσαν να λύσουν αποδοτικά αυτό το πρόβλημα [6].

Για πολύ καιρό, τα ανοιχτά προβλήματα σχετικά με τα δέντρα Gomory-Hu ήταν αν μπορεί να αναπτυχθεί ένας ντετερμινιστικός αλγόριθμος για την κατασκευή δέντρων Gomory-Hu και αν μπορεί να σπάσει το κυβικό όριο για τους αλγόριθμους Gomory-Hu. Όπως έχουμε αναφέρει ήδη, αυτά τα ερωτήματα έχουν πλέον απαντηθεί επιτυχώς [1, 3] και έχουν αναπτυχθεί σχετικοί αλγόριθμοι. Παρόλα αυτά, ένα ανοιχτό ερώτημα που παραμένει είναι το αν υπάρχει ντετερμινιστικός, αποδοτικός αλγόριθμος για έμβαρα, μη απλά γραφημάτων και αν οι τεχνικές απαλοιφής της τυχαιότητας μπορούν να χρησιμοποιηθούν στη γενική περίπτωση των μη κατευθυνόμενων γραφημάτων.

ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΑΣ

Ξενόγλωσσος Όρος	Ελληνικός Όρος
Source	Πηγή
Sink	Καταβόθρα
Cut	Τομή
Multi-terminal Flow	Ροή ανάμεσα σε Πολλαπλά Τερματικά
Strongly Connected	Ισχυρά Συνεκτικό
Maximal Spanning Tree	Μέγιστο Επικαλύπτον Δέντρο
Contract	Συνθλίβω
Tree Packing	Πακετάρισμα δέντρων
r -arborescence	r -κατευθυνόμενο δέντρο
Directionless r -spanning Tree	r -επικαλύπτον Δέντρο χωρίς Κατεύθυνση
Transpose Graph	Αντίστροφο Γράφημα
Component	Συνεκτικό Στοιχείο
Deficient	Ελλιπής
fundamental Circuit	Θεμελιώδες Κύκλωμα
Closure Computation	Υπολογισμός Κλειστότητας
Contiguous	Γειτονικές
Super-vertices	Υπέρ-κορυφές
Degree Balancing	Ισοζυγισμού Βαθμού
Super Edge	Υπέρ-ακμή
Partial Super Edge	Μερική Υπέρ-ακμή
Seed	Είσοδος
Local Degree Balancing	Τοπικός Ισοζυγισμός βαθμού
Global Degree Balancing	Καθολικός Ισοζυγισμός βαθμού
Descendant Subtrees	Υποδέντρα Απόγονοι
Quasi-linear	Ψευδογραμμικός
Unionbound	Ιδιότητα Υποπροσθετικότητας
Planar	Επίπεδος
Bounded Treewidth	Φραγμένο Δενδροπλάτος

Single Source Maximum Flow Problem	Πρόβλημα Μοναδικής Πηγής Μέγιστης Ροής
Pivot	Σημείο Βάσης
Optimum Communication Spanning Tree Problem	Πρόβλημα Βέλτιστου Επικαλύπτοντος Δέντρου Επικοινωνίας

ΑΝΑΦΟΡΕΣ

- [1] Amir Abboud, Robert Krauthgamer, Jason Li, Debmalya Panigrahi, Thatchaphol Saranurak, and Ohad Trabelsi. Gomory-hu tree in subcubic time. *CoRR*, abs/2111.04958, 2021.
- [2] Amir Abboud, Robert Krauthgamer, and Ohad Trabelsi. Cut-Equivalent Trees are Optimal for Min-Cut Queries. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 105–118, 2020.
- [3] Amir Abboud, Robert Krauthgamer, and Ohad Trabelsi. APMF < APSP? Gomory-Hu Tree for Unweighted Graphs in Almost-Quadratic Time. In *62nd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2021, Denver, CO, USA, February 7-10, 2022*, pages 1135–1146. IEEE, 2021.
- [4] Amir Abboud, Robert Krauthgamer, and Ohad Trabelsi. New Algorithms and Lower Bounds for All-Pairs Max-Flow in Undirected Graphs. *Theory Comput.*, 17:1–27, 2021.
- [5] Jørgen Bang-Jensen, András Frank, and Bill Jackson. Preserving and Increasing Local Edge-Connectivity in Mixed Graphs. *SIAM Journal on Discrete Mathematics*, 8(2):155–178, 1995.
- [6] Surender Baswana, Shiv Gupta, and Till Knollmann. Mincut Sensitivity Data Structures for the Insertion of an Edge. In Fabrizio Grandoni, Grzegorz Herman, and Peter Sanders, editors, *28th Annual European Symposium on Algorithms (ESA 2020)*, volume 173 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 12:1–12:14, Dagstuhl, Germany, 2020. Schloss Dagstuhl–Leibniz-Zentrum für Informatik.
- [7] Anand Bhalgat, Richard Cole, Ramesh Hariharan, Telikepalli Kavitha, and Debmalya Panigrahi. Efficient Algorithms for Steiner Edge Connectivity Computation and Gomory-Hu Tree Construction for Unweighted Graphs. Unpublished full version of [12], 2008.
- [8] Glencora Borradaile, Piotr Sankowski, and Christian Wulff-Nilsen. Min St-Cut Oracle for Planar Graphs with Near-Linear Preprocessing Time. *ACM Trans. Algorithms*, 11(3), jan 2015.
- [9] Jack Edmonds. *Submodular Functions, Matroids, and Certain Polyhedra*, pages 11–26. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003.
- [10] H.N. Gabow. A Matroid Approach to Finding Edge Connectivity and Packing Arborescences. *Journal of Computer and System Sciences*, 50(2):259–273, 1995.
- [11] Ralph E. Gomory and T. C. Hu. Multi-Terminal Network Flows. *Journal of The Society for Industrial and Applied Mathematics*, 9:551–570, 1961.
- [12] Ramesh Hariharan, Telikepalli Kavitha, Debmalya Panigrahi, and Anand Bhalgat. An $\tilde{O}(Mn)$ Gomory-Hu Tree Construction Algorithm for Unweighted Graphs. In *Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing, STOC '07*, page 605–614, New York, NY, USA, 2007. Association for Computing Machinery.
- [13] Tanja Hartmann and Dorothea Wagner. Fast and simple fully-dynamic cut tree construction. In Kun-Mao Chao, Tsan-sheng Hsu, and Der-Tsai Lee, editors, *Algorithms and Computation*, pages 95–105, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.
- [14] T. C. Hu. Optimum communication spanning trees. *SIAM Journal on Computing*, 3(3):188–195, 1974.
- [15] David R. Karger and Matthew S. Levine. Fast augmenting paths by random sampling from residual graphs. *SIAM Journal on Computing*, 44(2):320–339, 2015.
- [16] Jon Kleinberg and Eva Tardos. *Algorithm Design*, pages 346–350. Addison-Wesley Longman Publishing Co., Inc., USA, 2005.
- [17] Jason Li, Debmalya Panigrahi, and Thatchaphol Saranurak. A Nearly Optimal All-Pairs Min-Cuts Algorithm in Simple Graphs. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1124–1134, 2022.
- [18] Xinyue Liu, Hongfei Lin, and Ye Tian. Segmenting webpage with gomory-hu tree based clustering. *J. Softw.*, 6(12):2421–2425, 2011.
- [19] Hiroshi Nagamochi and Toshihide Ibaraki. A Linear-Time Algorithm for Finding a Sparse k-Connected Spanning Subgraph of a k-Connected Graph. *Algorithmica*, 7(5 & 6):583–596, 1992.

- [20] Guyslain Naves and F. Bruce Shepherd. When do gomory-hu subtrees exist? *CoRR*, abs/1807.07331, 2018.
- [21] Nurcan Tuncbag, F. Sibel Salman, Ozlem Keskin, and Attila Gursoy. Analysis and network representation of hotspots in protein interfaces using minimum cut trees. *Proteins: Structure, Function, and Bioinformatics*, 78(10):2283–2294, 2010.
- [22] David P. Williamson and David B. Shmoys. *The Design of Approximation Algorithms*, chapter 5, pages 129–130. Cambridge University Press, 2011.
- [23] Z. Wu and R. Leahy. An optimal graph theoretic approach to data clustering: theory and its application to image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11):1101–1113, 1993.
- [24] Tianyi Zhang. Faster Cut-Equivalent Trees in Simple Graphs. *CoRR*, abs/2106.03305, 2021.