# Multicut and Integer MulticommodityFlow in Trees

ΑΙΚΑΤΕΡΙΝΗ-ΠΑΝΑΓΙΩΤΑ ΣΤΟΥΚΑ
ΝΙΚΟΛΑΟΣ ΛΑΜΠΡΟΥ

μΠλ∀ 2014

# Minimum multicut

- **Έχουμε** ***$G = (V,E)$ μη κατευθυνόμενο γράφο***

με μη αρνητικές χωρητικότητες $c_e$ *για κάθε* $e \in E$.

- $\{(s_1, t_1), \ldots, (s_k, t_k)\}$ *διακριτά διατεταγμένη ζεύγη κορυφών*
- ***multicut*** *είναι ένα σύνολο από ακμές το οποίο αν το αφαιρέσουμε από γράφημα διαχωρίζονται* $s_i$ *από* $t_i$*, δηλαδή φεύγει τουλάχιστον μία ακμή από κάθε μονοπάτι από το* $s_i$ *στο* $t_i$.
- Το πρόβλημα είναι να βρούμε το multicut με το μικρότερο άθροισμα χωρητικοτήτων $c_e$ των ακμών

# Minimum multicut

- *Minimum s-t cut* ⇨ ειδική περίπτωση *minimum multicut* για κ=1
- *Minimum multicut problem* ⇨ γενίκευση του *multiway cut problem*,(παίρνουμε για ζευγάρια $(s_i, s_j)$ $1 \le i < j \le l$),*που για* κ=3 είναι NP-hard.
- ***Multiway cut problem*** **:**Δίνεται σύνολο από terminals $\mathbf{S} = \{s_1, s_2, \ldots, s_k\} \subseteq V$ , ένα multiway cut είναι ένα σύνολο από ακμές ,το οποίο αν το αφαιρέσουμε από το γράφημα διαχωρίζει τα terminals .Αναζητούμε το multiway cut με το μικρότερο βάρος ακμών.
- Για τη γενική περίπτωση ⇨ O(logk) προσέγγιση
- Για G δέντρο ⇨ 2-προσέγγιση.

# Minimum multicut για G δέντρο

- Αν G είναι δέντρο υπάρχει μοναδικό μονοπάτι από *το* $s_i$ *στο* $t_i$.
- *Minimum multicut problem* παραμένει NP-hard ακόμα και για δέντρα με μοναδιαίες χωρητικότητες και ύψους 1.
- *Απόδειξη*: Αναγάγουμε το *vertex cover* σε αυτό ⇨Παίρνουμε ένα στιγμιότυπο του Vertex cover G=(V,E) και φτιάχνουμε το παρακάτω στιγμιότυπο του *Minimum multicut problem*
- Φτιάχνουμε ένα δέντρο ύψους 1 με κορυφές V∪{r} και ακμές E'={{r,v}/v∈V } με μοναδιαίες χωρητικότητες.Τα ζευγάρια που θέλουμε να διαχωρίσουμε είναι τα {(vi,vj) , ∀{vi,vj} ακμή του G}.
- Ισχύει ότι το G έχει Vertex cover μεγέθους t⇔το T έχει multicut κόστους t.(Για κάθε κορυφή v του vertex cover παίρνουμε ακμή {r,v} στο multicut και αντίστροφα.)
- Η αναγωγή είναι *approximation preserving*.

# Γραμμικό πρόγραμμα για **Minimum multicut** για G δέντρο

- {Minimize $\sum_{e \in E} c_e d_e$
  subject to $\sum_{e \in p_i} d_e \geq 1$, $i \in \{1, . . . , k\}$
  $d_e \in \{0, 1\}$, $e \in E$ }
- $d_e = 1$ αν παίρνουμε e στο multicut ο αλλιώς και $p_i$ το μοναδικό μονοπάτι από το $s_i$ στο $t_i$.
- Έχουμε το LP-relaxation για $d_e \geq 0$ και $d_e \leq 1$ .Το $d_e \leq 1$ μπορεί να παραληφθεί, αφου αν $d_e > 1$, για $d_e = 1$,προκύπει λύση feasible με μικρότερο κόστος.

# Dual program –maximum *multicommodity flow* in *G*,

- {Maximize $\sum^{k}_{i=1} f_i$
  subject to $\sum_{i:\, e\in p_i} f_i \le c_e,\ e\in E$
  $f_i \ge 0,\ i\in\{1, \ldots, k\}$ }
- *Ορίζουμε στα αντίστοιχα ζευγάρια του multicut problem $(s_i, t_i)$ commodities και τα $f_i$ απεικονίζουν την ποσότητα των commodities που διαβιβάζεται μέσω του $p_i$.*
- Από το *θεώρημα ασθενούς δυικότητας* μία feasible λύση του multicommodity flow δίνει ένα κάτω φράγμα για το minimum fractional multicut που είναι ≤ minimum integral multicut.
- Από *ισχυρή δυικότητα* το minimum fractional multicut =maximum multicommodity flow.

# Integer multicommodity flow

- Έχουμε το γράφημα G και τα ζευγάρια *$\{(s_1, t_1), \ldots, (s_k, t_k)\}$ ,όπως στο* minimum multicut problem με τη διαφορά ότι οι χωρητικότητες των ακμών είναι ακέραιες. Ορίζουμε commodity για κάθε *$(s_i, t_i)$ και θέλουμε να μεγιστοποιήσουμε το άθροισμα των ποσοτήτων των commodities που διαβιβάζεται μέσω του $p_i$ μονοπατιού, με την προϋπόθεση ότι ότι δεν υπερβαίνουμε τη χωρητικότητα κάθε ακμής και ότι στέλνουμε ακέραια ποσότητα καθενός commodity.*
- *Το παραπάνω πρόβλημα για G δέντρο περιγράφεται από το γραμμικό πρόγραμμα του* maximum *multicommodity flow για $f_i$ ακέραια.*
- *Είναι NP-hard ακόμα και για δέντρο ύψους 3.*

# Complementary slackness conditions

- Για G δέντρο με ακέραιες χωρητικότητες ισχύει:
  $OPT_{\text{integ mult fl}} \leq OPT_{\text{max mult fl}} = OPT_{\text{min mult}} \leq OPT_{\text{integ min mult}}$

- Έστω το *primal* και *dual* γραμμικό πρόγραμμα :

$\{min\ c^T x$ $\{\ max\ b^T y$

$Ax \geq b\ ,\ x \geq 0\ \}$ $A^T y \leq c\ , y \geq 0\}$

$με\ A \in R^{m \times n}, b \in R^m, c \in R^n$

- ***Primal complementary slackness conditions***

  Έστω $\alpha \geq 1$. Για $1 \leq j \leq n$: $x_j = 0$ ή $c_j/\alpha \leq \sum_{i=1}^{m} a_{ij} y_i \leq c_j$

- ***Dual complementary slackness conditions***

  Έστω $\beta \geq 1$.

  Για $1 \leq i \leq m$: $y_i = 0$ ή $b_i \leq \sum_{i=1}^{n} a_{ij} xj \leq \beta \cdot b_i$,

# Complementary slackness conditions

- *Εάν x και y είναι primal και dual feasible λύσεις αντίστοιχα που ικανοποιούν τις* ***complementary slackness conditions*** *τότε*

$$\sum_{j=1}^{n} c_j x_j \leq \alpha \cdot \beta \cdot \sum_{i=1}^{m} b_i y_i$$

- *Απόδειξη:* $\sum_{j=1}^{n} c_j x_j \leq \alpha \cdot \sum_{j=1}^{n} (\sum_{i=1}^{m} a_{ij} y_i \cdot x_j) = \alpha \cdot \sum_{i=1}^{m} \sum_{j=1}^{n} (a_{ij} y_i \cdot x_j) = \alpha \cdot \sum_{i=1}^{m} [(\sum_{j=1}^{n} a_{ij} x_j) \cdot y_i] \leq \alpha \cdot \beta \cdot \sum_{i=1}^{m} b_i y_i$
- *Δηλαδή x είναι* $\alpha \cdot \beta$ *προσεγγιστική για το primal, αφού* $\sum_{i=1}^{m} b_i y_i \leq OPT_{dual} = OPT_{primal}$, *άρα* $\sum_{j=1}^{n} c_j x_j \leq \alpha\beta OPT_{primal}$

# Complementary slackness conditions για το Minimum multicut

- Για το *primal* γραμμικό πρόγραμμα του *Minimum multicut για G δέντρο και το* Dual του (γραμμικό πρόγραμμα του *maximum multicommodity flow* στο *G),καθώς και α=1 και β=2, έχουμε* αντίστοιχα:
- ***Primal conditions***: $\forall\ e \in E,\ d_e \neq 0 \Rightarrow \sum_{i:\, e \in p_i} f_i = c_e$ *,δηλαδή κάθε ακμή που παίρνουμε στο multicut πρέπει να είναι κορεσμένη.*
- ***Relaxed dual conditions*: Για $i \in \{1, \ldots, k\}$,** $f_i \neq 0 \Rightarrow 1 \leq \sum_{e \in p_i} d_e \leq 2$ *, δηλαδή από κάθε μονοπάτι που μεταφέρει μη μηδενική ροή παίρνουμε τουλάχιστον μία και το πολύ δύο ακμές.*

# ΙΔΙΟΤΗΤΕΣ ΚΑΙ ΟΡΙΣΜΟΙ ΓΙΑ ΑΛΓΟΡΙΘΜΟ ΓΙΑ MINIMUM MULTICUT

- *Εάν x και y είναι primal και dual feasible λύσεις αντίστοιχα που ικανοποιούν τις* ***complementary slackness conditions*** *τότε*

$\sum_{e\in E} c_e d_e \leq 2 \cdot \sum^{k}_{i=1} f_i$

- *lca(u, v)* $\stackrel{def}{=}$ ο κοινός πρόγονος με το μικρότερο βάθος (η ρίζα έχει βάθος 0) $\stackrel{def}{=}$ η κορυφή με το μικρότερο βάθος στο μονοπάτι u-v.

- Στον παρακάτω αλγόριθμο θα πάρουμε αρχηκά την λύση $d_e=f_e=0\ \forall\ e \in E$ ,η οποία είναι *feasible* για το *dual* ,αλλά infeasible για το *primal* ,και θα προσπαθήσουμε να βελτιώσουμε το optimality της *dual λύσης* και το feasibility της λύσης για το *primal*.(*primal -dual method*)
- Θα αποδείξουμε ότι για τις λύσεις αυτές ισχύουν οι *Complementary slackness conditions* και λόγω του θεωρήματος θα πάρουμε τις προσεγγίσεις που θέλουμε.

# ΑΛΓΟΡΙΘΜΟΣ ΓΙΑ MINIMUM MULTICUT

- ***Algorithm 18.4 (Multicut and integer multicommodity flow in trees)***
- *1.* ***Initialization***: $f \leftarrow 0$; $D \leftarrow \emptyset$.
- *2.* ***Flow routing***: **For each vertex *v*, in non increasing order of depth, do:**

  For each pair $(s_i, t_i)$ such that $lca(s_i, t_i) = v$, greedily route integral flow from $s_i$ to $t_i$.

  Add to *D all edges that were saturated in the current iteration in* arbitrary order.
- 3. Let $e_1, e_2, \ldots, e_l$ *be the ordered list of edges in D.*
- *4.* ***Reverse delete***: **For** $j = l$ **down to 1 do:**

  If $D - \{e_j\}$ is a multicut in *G*, then $D \leftarrow D - \{e_j\}$.
- 5. Output the flow and multicut *D*.

# Feasibility λύσεων

- ο αλγόριθμος εξετάζει κάθε κορυφή ⇨ θα εξεταστεί σίγουρα η κορυφή lca ($s_i$, $t_i$) και θα προστεθεί στο D τουλάχιστον μία ακμή από το μοναδικό μονοπάτι $s_i - t_i$.(στέλνουμε ακέραια ροή μέχρι να κορεστεί τουλάχιστον μία ακμή και βάζουμε στο D τις κορεσμένες ακμές) ⇨ Το σύνολο D που δίνει ο αλγόριθμος είναι multicut ( ή αλλιώς τα $\{d_e\}$, για $e \in E$ είναι feasible λύση για το primal(integer))
- Στον αλγόριθμο σταματάμε να στέλνουμε ροή (ακέραια) μόλις κορεστεί μία ακμή ⇨ δεν έχουμε υπερχείλιση ⇨ Η λύση $\{f_e\}$ για $e \in E$ είναι feasible για το integer multicommodity flow, άρα και fractional feasible λύση για το multicommodity flow).

# ικανοποίηση για τις **Relaxed dual conditions**

- **Lemma 18.5**: Έστω ($s_i$, $t_i$) ένα ζευγάρι με fi≠0, και lca($s_i$, $t_i$) = v. Παίρνουμε το πολύ μία ακμή στο multicut D σε καθένα από τα μονοπάτια ,$s_i$-v και $t_i$ –v ⇨ ικανοποιούνται οι **Relaxed dual conditions**: **Για** $i \in \{1, \dots, k\}$, $f_i \neq 0 \Rightarrow 1 \leq \sum_{e \in p_i} d_e \leq 2$

- ΑΠΟΔΕΙΞΗ: Θα αποδείξουμε ότι υπάρχει το πολύ μία ακμή στο μονοπάτι $s_i$-v.
- 1)Έστω ότι στο D έχουν παραμείνει στο τέλος του αλγορίθμου οι ακμές e, e' με e να έχει μεγαλύτερο βάθος(πιο μακριά από ρίζα) στο $s_i$-v,άρα και στο $s_i$-$t_i$

# ΣΥΝΕΧΕΙΑ ΑΠΟΔΕΙΞΗΣ

- *2)Όταν ελέγχεται η e για το αν θα αφαιρεθεί από το D στο* **Reverse delete στάδιο** *,η e'(είτε έχει ελεγχθεί πιο πριν είτε όχι) βρίσκεται στο D ⇨ υπάρχει ένα ζευγάρι $(s_j,t_j)$ τω e είναι μοναδική του μονοπατιού $s_j$–$t_j$ στο D εκείνη τη στιγμή ,ώστε e να είναι απαραίτητη για multicut και να μην αφαιρεθεί .*
- *3)Έστω u $\stackrel{def}{=}$ lca($s_j$, $t_j$) .Έχουμε ότι e' δεν βρίσκεται στο $s_j$–$t_j$ μονοπάτι ⇨ u έχει μεγαλύτερο βάθος από e' ⇨ u μεγαλύτερο βάθος από v ⇨ u εξετάζεται πρώτα από την v στον αλγόριθμο ⇨ οι ακμές που μπήκαν στο multicut σε αυτό το στάδιο εξετάζονται στο* **Reverse delete στάδιο** *για το αν θα παραμείνουν στο D μετά από αυτές που μπήκαν κατά την εξέταση της v κορυφής.*

# ΣΥΝΕΧΕΙΑ ΑΠΟΔΕΙΞΗΣ

- 4) $f_i \neq 0$ ⇨ *e μπαίνει στο D κατά το στάδιο εξέτασης της v κορυφής ή μετά ,γιατί αν είχε μπει σε προηγούμενο στάδιο ( κατά την εξέταση της u) θα είχε κορεστεί* ⇨ *δεν θα μπορούσαμε να στείλουμε ροή στο $s_i$–$t_i$ μονοπάτι.*
- *5) Άρα κατά την εξέταση της u μπαίνει e'' από το μονοπάτι $s_j$-$t_j$.*
- *6) v μικρότερο βάθος από u* ⇨*e'' μπαίνει πρώτα από e στο D* ⇨ *e εξετάζεται πρώτη στο* **Reverse delete στάδιο**
 ⇨*e'' είναι στο D όταν εξετάζεται e.*
- *7)ΑΤΟΠΟ αφού έχουμε υποθέσει ότι e είναι μοναδική του μονοπατιού $s_j$–$t_j$ στο D εκείνη τη στιγμή .*

# Σύνολο D ⇨ 2-προσεγγιστική λύση για το integer minimum multicut problem

- Έχουμε ότι :

  οι ***Primal conditions***: $\forall\ e \in E,\ d_e \neq 0 \Rightarrow \sum_{i:\ e \in p_i} f_i = c_e$ *ικανοποιούνται, αφού στο D παίρνουμε μόνο κορεσμένες ακμές.*
- Άρα για τις λύσεις του αλγορίθμου ισχύει $\sum_{e \in E} c_e d_e \leq 2 \cdot \sum_{i=1}^{k} f_i$
- Επίσης ισχύει $\sum_{i=1}^{k} f_i \leq OPT_{\text{max mult flow}} = OPT_{\text{min mult}}$
- Άρα $\sum_{e \in E} c_e d_e \leq 2 \cdot OPT_{\text{min mult}} \leq 2 \cdot OPT_{\text{int min mult}}$ και το σύνολο D που δίνει ο αλγόριθμος είναι 2-προσεγγιστική λύση για το integer minimum multicut problem

## $\{f_e\}$ *για* $e \in E$ είναι ½- προσέγγιση του *integer multicommodity flow problem*

- $\sum_{e \in E} c_e d_e \leq 2 \cdot \sum^{k}_{i=1} f_i \Leftrightarrow \sum^{k}_{i=1} f_i \geq 1/2 \sum_{e \in E} c_e d_e$
  $\geq 1/2 \cdot \mathrm{OPT}_{\text{min mult}} = 1/2 \cdot \mathrm{OPT}_{\text{max mult flow}}$
  $\geq 1/2 \cdot OPT_{\textit{integ mult fl}}$

- *Και επειδή* $f_i$ που επιστρέφει ο αλγόριθμος είναι ακέραια έχουμε ότι $\{f_e\}$ *για* $e \in E$ είναι ½- προσέγγιση του *integer multicommodity flow problem*

- *ΠΟΡΙΣΜΑ :Για G δέντρο με ακέραιες χωρητικότητες :*
  $OPT_{\textit{integ mult fl}} \leq \mathrm{OPT}_{\text{min mult}} \leq 2 \cdot OPT_{\textit{integ mult fl}}$

# ΑΠΟΔΕΙΞΗ ΠΟΡΙΣΜΑΤΟΣ

- ΑΠΟΔΕΙΞΗ:

$$OPT_{\text{integ mult fl}} \le \text{OPT}_{\text{max mult flow}} = \text{OPT}_{\text{min mult}}$$

$$\text{OPT}_{\text{min mult}} \le \sum_{e\in E} c_e d_e \le 2 \cdot \sum_{i=1}^{k} f_i \le 2 \cdot OPT_{\text{integ mult fl}}$$

- $\sum_{i=1}^{k} f_i \le OPT_{\text{integ mult fl}}$ *,αφού* $\{f_e\}$ *για* $e \in E$ *είναι feasible λύση*

*για το integer multicommodity flow problem.*

- $OPT_{\text{min mult}} \le \sum_{e\in E} c_e d_e$ ,αφού $\{d_e\}$ για $e \in E$ *feasible λύση για minimum multicut.*

# ΠΗΓΕΣ


- Approximation Algorithms ***Vijay V. Vazirani*** College of Computing Georgia Institute of Technology
- The design of Approximation Algorithms David P.Williamson,David B. Shmoys