THE GEOMETRY OF SCHEDULING

NIKHIL BANSAL AND KIRK PRUHS

SIAM Journal on Computing

2014

Δομή της παρουσίασης

Στόχος: Προσεγγιστικός αλγόριθμος για το general scheduling problem (GSP).

1. Εισαγωγή του GSP
2. Εισαγωγή του γεωμετρικού προβλήματος R2C
3. Αναγωγή του GSP στο R2C
4. Προσεγγιστικός αλγόριθμος για το R2C

General Scheduling Problem

Εμπεριέχει διάφορα προβλήματα χρονοπρογραμματισμού.

General Scheduling Problem

Εμπεριέχει διάφορα προβλήματα χρονοπρογραμματισμού.

Επεξεργαστής που επεξεργάζεται n δουλειές j με:

General Scheduling Problem

Εμπεριέχει διάφορα προβλήματα χρονοπρογραμματισμού.

Επεξεργαστής που επεξεργάζεται n δουλειές j με:

- Θετικό ακέραιο χρόνο απελευθέρωσης r_j
- Θετικό ακέραιο μέγεθος (runtime) p_j
- Θετική μη φθίνουσα συνάρτηση κόστους (βάρους) $w_j(t)$

General Scheduling Problem

Εμπεριέχει διάφορα προβλήματα χρονοπρογραμματισμού.

Επεξεργαστής που επεξεργάζεται n δουλειές j με:

- Θετικό ακέραιο χρόνο απελευθέρωσης r_j
- Θετικό ακέραιο μέγεθος (runtime) p_j
- Θετική μη φθίνουσα συνάρτηση κόστους (βάρους) $w_j(t)$

Μη-παραλληλισμός

Ο επεξεργαστής επεξεργάζεται 1 μόνο δουλειά σε δεδομένη χρονική στιγμή.
Μπορεί όμως να αλλάξει δουλειά **σε ακέραιες χρονικές στιγμές**.

General Scheduling Problem

Εμπεριέχει διάφορα προβλήματα χρονοπρογραμματισμού.

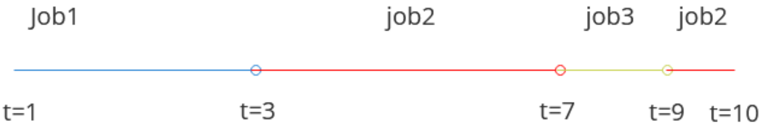
Επεξεργαστής που επεξεργάζεται n δουλειές j με:

- Θετικό ακέραιο χρόνο απελευθέρωσης r_j
- Θετικό ακέραιο μέγεθος (runtime) p_j
- Θετική μη φθίνουσα συνάρτηση κόστους (βάρους) $w_j(t)$

Μη-παραλληλισμός

Ο επεξεργαστής επεξεργάζεται 1 μόνο δουλειά σε δεδομένη χρονική στιγμή.
Μπορεί όμως να αλλάξει δουλειά **σε ακέραιες χρονικές στιγμές**.

Δουλειά που τελειώνει στο t δημιουργεί κόστος $w_j(t)$. Στόχος η ελαχιστοποίηση του αθροιστικού κόστους.



$$\mathbf{cost}=w_1(3)+w_2(10)+w_3(9)$$

Παρατήρηση:

Όλες οι παράμετροι που ορίσαμε εκτός των συναρτήσεων βαρών παίρνουν τιμές στους φυσικούς.

General Scheduling Problem

Εμπεριέχει διάφορα προβλήματα χρονοπρογραμματισμού.

- Weighted flow time: $w_j(t) = w_j(t - r_j)$, w_j σταθερά
- Flow time squared: $w_j(t) = (t - r_j)^2$
- Weighted tardiness: $w_j(t) = w_j max(t - d_j)$

General Scheduling Problem

Εμπεριέχει διάφορα προβλήματα χρονοπρογραμματισμού.

- Weighted flow time: $w_j(t) = w_j(t - r_j)$, w_j σταθερά
- Flow time squared: $w_j(t) = (t - r_j)^2$
- Weighted tardiness: $w_j(t) = w_j max(t - d_j)$

Προφανώς ενδιαφέροντος.

Πριν

- Weighted flow time: Polylog-approximation

General Scheduling Problem

Εμπεριέχει διάφορα προβλήματα χρονοπρογραμματισμού.

- Weighted flow time: $w_j(t) = w_j(t - r_j)$, w_j σταθερά
- Flow time squared: $w_j(t) = (t - r_j)^2$
- Weighted tardiness: $w_j(t) = w_j max(t - d_j)$

Προφανώς ενδιαφέροντος.

Πριν

- Weighted flow time: Polylog-approximation
- Flow time squared: Non of interest

General Scheduling Problem

Εμπεριέχει διάφορα προβλήματα χρονοπρογραμματισμού.

- Weighted flow time: $w_j(t) = w_j(t - r_j)$, w_j σταθερά
- Flow time squared: $w_j(t) = (t - r_j)^2$
- Weighted tardiness: $w_j(t) = w_j max(t - d_j)$

Προφανώς ενδιαφέροντος.

Πριν

- Weighted flow time: Polylog-approximation
- Flow time squared: Non of interest
- Weighted tardiness: Non of interest

General Scheduling Problem

Εμπεριέχει διάφορα προβλήματα χρονοπρογραμματισμού.

- Weighted flow time: $w_j(t) = w_j(t - r_j)$, w_j σταθερά
- Flow time squared: $w_j(t) = (t - r_j)^2$
- Weighted tardiness: $w_j(t) = w_j max(t - d_j)$

Προφανώς ενδιαφέροντος.

Πριν

- Weighted flow time: Polylog-approximation
- Flow time squared: Non of interest
- Weighted tardiness: Non of interest

Με αυτό το paper όλα έχουν $O(\log(\log(P))$-Προσέγγιση, $P = \frac{max}{min}$ job size, σε χρόνο $O(n \log W)$, W max weight.

Rectangle 2D Capacitated (R2C)

Σύνολο σημείων στο $\mathcal{Z}^2$ που απαιτούν να καλυφθούν από ορθογώνια.

Rectangle 2D Capacitated (R2C)

Σύνολο σημείων στο $\mathcal{Z}^2$ που απαιτούν να καλυφθούν από ορθογώνια.

Σημεία $p \in P$ με:

- Θετική ακέραια απαίτηση d_p

Rectangle 2D Capacitated (R2C)

Σύνολο σημείων στο $\mathcal{Z}^2$ που απαιτούν να καλυφθούν από ορθογώνια.

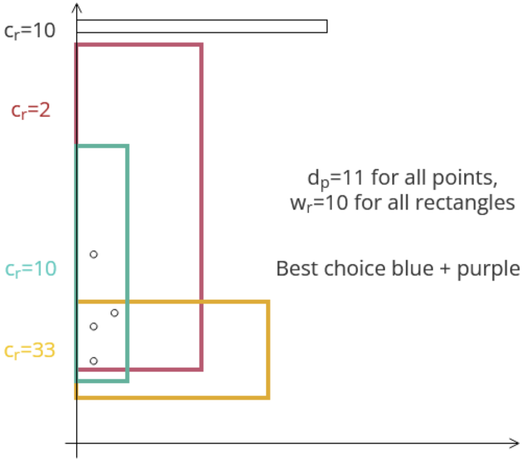
Σημεία $p \in P$ με:

- Θετική ακέραια απαίτηση d_p

Σύνολο ορθογωνίων $r \in R$ με:

- Θετική ακέραια χωρητικότητα (ικανοποιησιμότητα) c_r
- Κάθε ορθογώνιο ικανοποιεί όλα τα σημεία που καλύπτει κατά c_r.
- Θετικό ακέραιο κόστος (βάρος) w_r
- Μια πλευρά προσκολλημένη στον άξονα $y'y$

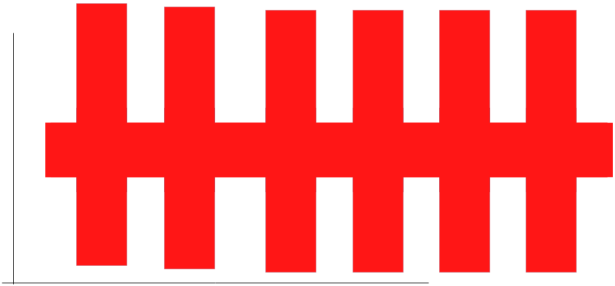
Επιλογή ορθογωνίων ελάχιστου αθροιστικού βάρους που να ικανοποιεί όλα τα σημεία.



d_p=11 for all points,
w_r=10 for all rectangles

Best choice blue + purple

Union complexity: Πλήθος πλευρών και κορυφών ένωσης γεωμετρικών σχημάτων.
Union complexity αυθαίρετης περίπτωσης: $\Omega(k^2)$, k το πλήθος ορθογωνίων.



Για το R2C, $O(k)$. Αυτό θα χρησιμέψει αργότερα.

Παρατηρήσεις και ορολογία για GSP

GSP με deadlines

Έστω ότι κάθε εργασία έχει ένα deadline d_j, πριν από το οποίο πρέπει να ολοκληρωθεί αντί συναρτήσεων βάρους.
Ερώτημα: Είναι ένα δοσμένο διάνυσμα προθεσμιών d_j εφικτό;
Απαντάται τετριμμένα με πολιτική Earliest Deadline First (EDF).

Παρατήρηση

Δοσμένου ενός χρονοπρογραμματισμού των εργασιών του απλού GSP, το GSP με deadlines $d_j := c_j$ είναι προφανώς εφικτό.

Συμπέρασμα

Αρκεί να βρούμε εφικτή ανάθεση προθεσμιών με $\sum_j w_j(d_j)$ ελάχιστο.

Παρατηρήσεις και ορολογία για GSP

Έστω $I = [t1, t2]$ ένα χρονικό διάστημα, πχ [1,3].

Παρατηρήσεις και ορολογία για GSP

Έστω $I = [t1, t2]$ ένα χρονικό διάστημα, πχ [1,3].

- Όρισε μήκος του I, έστω $|I| := t2 - t1$ πχ $|[1,3]|=2$

Παρατηρήσεις και ορολογία για GSP

Έστω $I = [t1, t2]$ ένα χρονικό διάστημα, πχ [1,3].

- Όρισε μήκος του I, έστω $|I| := t2 - t1$ πχ |[1,3]|=2
- $X(I)$: οι δουλείες που φτάνουν "εντός του I", πχ $X([1,3])$ οι δουλειές που φτάνουν όταν $t = 1$ ή 2 ή 3.

Παρατηρήσεις και ορολογία για GSP

Έστω $I = [t1, t2]$ ένα χρονικό διάστημα, πχ [1,3].

- Όρισε μήκος του I, έστω $|I| := t2 - t1$ πχ |[1,3]|=2
- $X(I)$: οι δουλείές που φτάνουν "εντός του I", πχ $X([1,3])$ οι δουλειές που φτάνουν όταν $t = 1$ ή 2 ή 3.
- $P(X(I))$: Το αθροιστικό τους runtime.

Παρατηρήσεις και ορολογία για GSP

Έστω $I = [t1, t2]$ ένα χρονικό διάστημα, πχ [1,3].

- Όρισε μήκος του I, έστω $|I| := t2 - t1$ πχ |[1,3]|=2
- $X(I)$: οι δουλείες που φτάνουν "εντός του I", πχ $X([1,3])$ οι δουλειές που φτάνουν όταν $t = 1$ ή 2 ή 3.
- $P(X(I))$: Το αθροιστικό τους runtime.
- $\xi(1,3)$: Excess time needed to complete all jobs arriving in [1,3]. Runtime δουλειών που φτάνουν στο 1,2,3 - length του [1,3]

Παρατηρήσεις και ορολογία για GSP

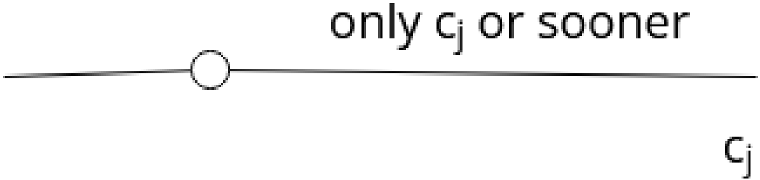
Συνθήκη ώστε οι προθεσμίες να είναι εφικτές

$\forall I$, $|I| \geq$ αθροιστικό runtime δουλειών που φτάνουν στο I και έχουν προθεσμία εντός του.

$\iff$

Συνθήκη ώστε οι προθεσμίες να είναι εφικτές

$\forall I$, αθροιστικό runtime δουλειών που φτάνουν στο I και έχουν προθεσμία μετά του $I \geq$ excess of I $\xi(I)$=runtime δουλειών που φτάνουν στο I - μέγεθος I.


only c_j or sooner
c_j

Αναγωγή GSP σε R2C

Ιδέα:

Θα φτιάξουμε instance του R2C που ανάμεσα σε όλες τις δυνατές προθεσμίες των εργασιών, που ικανοποιούν την συνθήκη, θα επιλέξει αυτές που ελαχιστοποιούν το κόστος.

- Χρήση 2ης μορφής συνθήκης: Οι εργασίες ικανοποιούν τα διαστήματα, όχι το αντίθετο!
- Τα διαστήματα I θα αντιστοιχιστούν σε σημεία και οι εργασίες σε ορθογώνια.
- Θα φτιάξουμε την αναγωγή έτσι ώστε κάθε σημείο να καλύπτεται μόνο από τις εργασίες που φτάνουν στο I και έχουν προθεσμία μετά του I. Δίνοντας στο σημείο $p(I)$ απαίτηση $d_p := \xi(I)$, και στο ορθογώνιο R_j χωρητικότητα (ικανοποιησημότητα) $c_j := p_j$, θα εξασφαλίσουμε την ισχύ της συνθήκης μέσω του R2C.

Αναγωγή GSP σε R2C

Λίγο πιο συγκεκριμένα (non final form):

- Για κάθε διάστημα $I = [s, t]$, φτιάχνουμε το σημείο $(s, t) \in Z^2$

Αναγωγή GSP σε R2C

Λίγο πιο συγκεκριμένα (non final form):

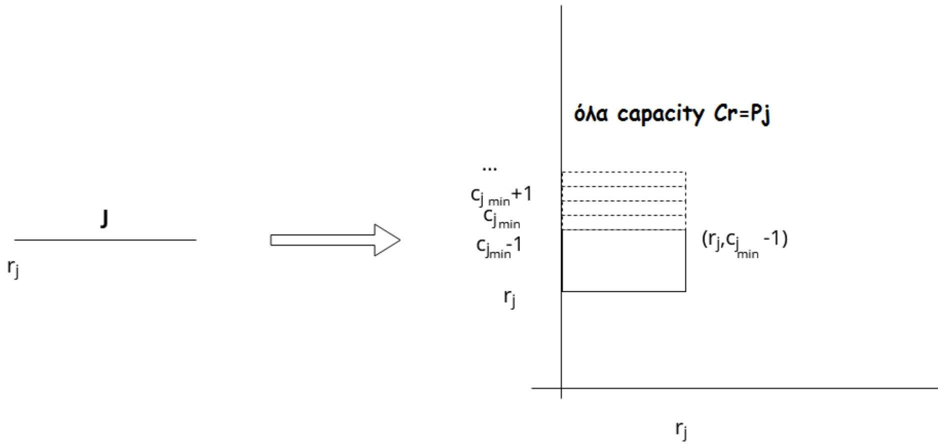
- Για κάθε διάστημα $I = [s, t]$, φτιάχνουμε το σημείο $(s, t) \in Z^2$
- Για κάθε δουλειά j, και κάθε δυνατή στιγμή ολοκλήρωσης της c_j (προθεσμίας d_j) φτιάχνουμε τα ορθογώνια $R_j(c_j) = [0, r_j] \times [r_j, c_j - 1]$

Αναγωγή GSP σε R2C

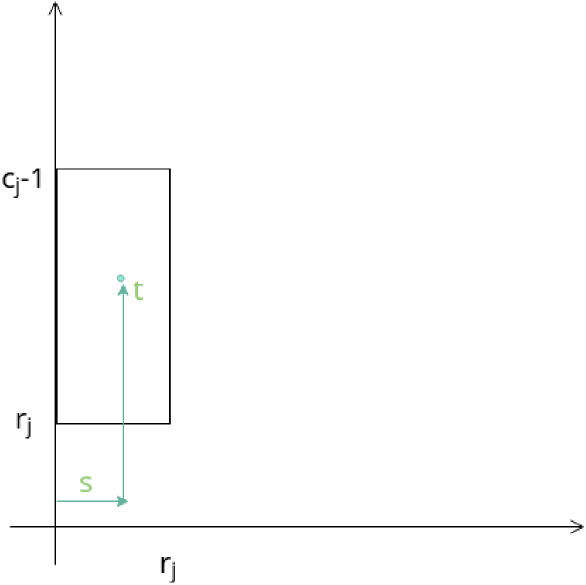
Λίγο πιο συγκεκριμένα (non final form):

- Για κάθε διάστημα $I = [s,t]$, φτιάχνουμε το σημείο $(s,t) \in Z^2$
- Για κάθε δουλειά j, και κάθε δυνατή στιγμή ολοκλήρωσης της c_j (προθεσμίας d_j) φτιάχνουμε τα ορθογώνια $R_j(c_j) = [0, r_j] \times [r_j, c_j - 1]$
- Με κάποιο τρόπο θα αναγκάσουμε το R2C να πάρει μόνο ένα ορθογώνιο για κάθε εργασία.

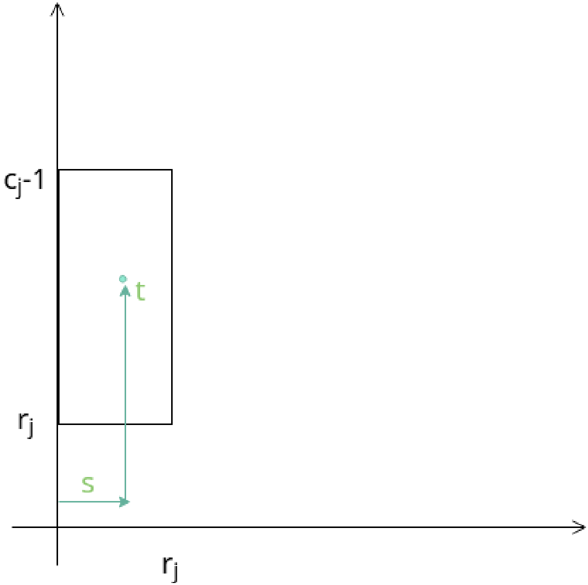
Αναγωγή GSP σε R2C



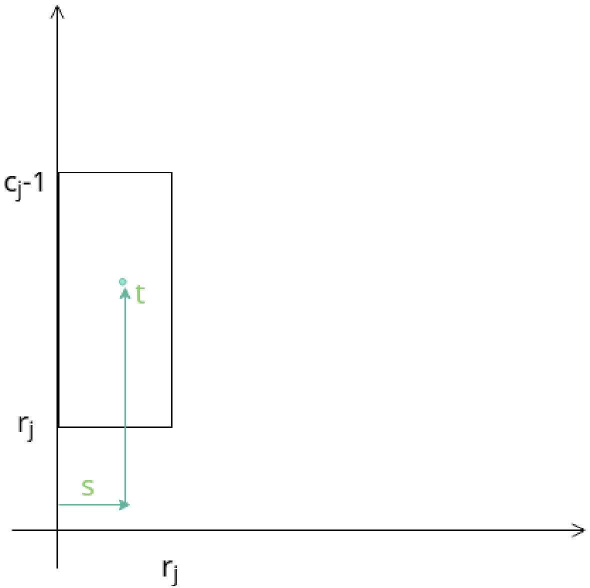
Ποια σημεία (διαστήματα) περιέχει ένα ορθογώνιο;



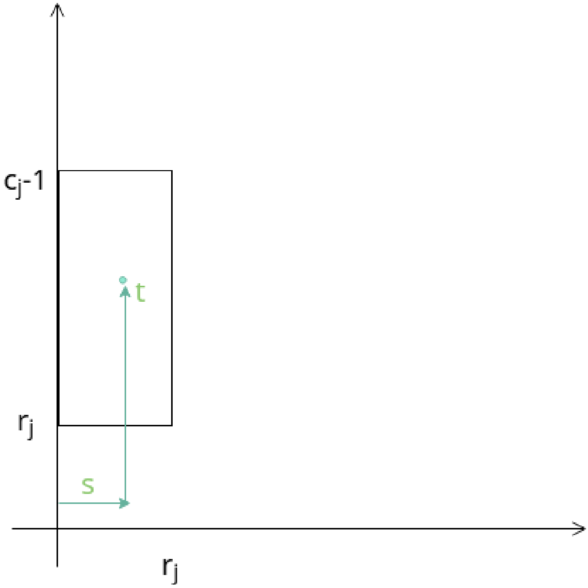
Πιάσε ακριβώς τα διαστήματα I που αρχίζουν $\leq r_j$ και τελειώνουν ανάμεσα στο r_j και το $c_j - 1$.



Κοιτώντας τις πλευρές: Το I θα πιαστεί ακριβώς από τις δουλείες που αρχίζουν $\geq s$ και αρχίζουν $\leq t$ και τελειώνουν μετά του t.



Αυτές είναι ακριβώς οι δουλείες της συνθήκης.



Αναγωγή GSP σε R2C

Λήμμα

Η συνθήκη ικανοποιείται για το I αν και μόνο αν το $p(I)$ ικανοποιείται στο R2C.

Έαν το R2C ήταν αυτό, η αναγωγή θα είχε τελειώσει, και χωρίς απώλεια στον λόγο προσέγγισης. Όμως δεν έχουμε εξασφαλίσει ότι για κάθε δουλειά j θα επιλεχθεί 1 μόνο ορθογώνιο.

Αναγωγή GSP σε R2C

Λήμμα

Η συνθήκη ικανοποιείται για το I αν και μόνο αν το $p(I)$ ικανοποιείται στο R2C.

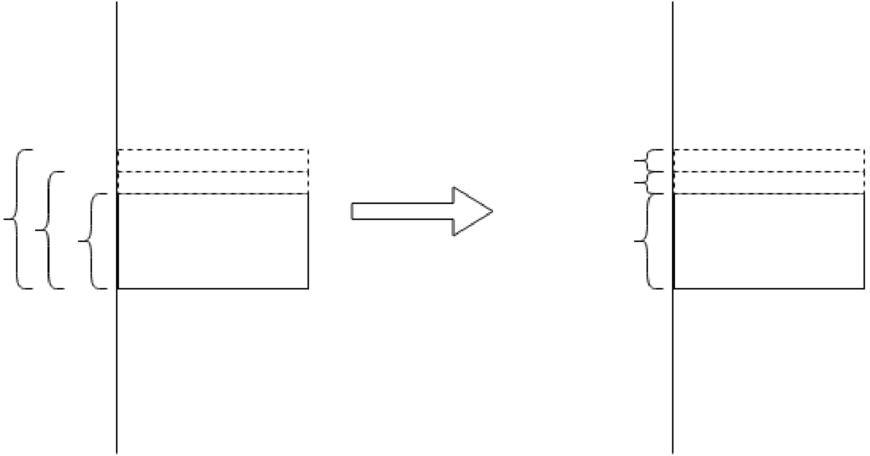
Έαν το R2C ήταν αυτό, η αναγωγή θα είχε τελειώσει, και χωρίς απώλεια στον λόγο προσέγγισης. Όμως δεν έχουμε εξασφαλίσει ότι για κάθε δουλειά j θα επιλεχθεί 1 μόνο ορθογώνιο.

Παρατήρηση:

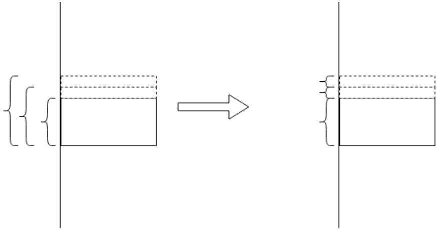
Για την δουλειά j και κάποιο c_j, το $R_j(c_j)$ είναι αυστηρά μικρότερο του $R_j(c_j+1)$, που είναι αυστηρά μικρότερο του $R_j(c_j+2)$...

Αναγωγή GSP σε R2C

Τμηματοποίηση ορθογωνίων; Τα κάνουμε ξένα. Αν για την δουλειά j το μεγαλύτερο ορθογώνιο είναι το $R_j(c_j)$, θέτουμε προθεσμία στο αρχικό πρόβλημα $d_j = c_j$.



Αναγωγή GSP σε R2C

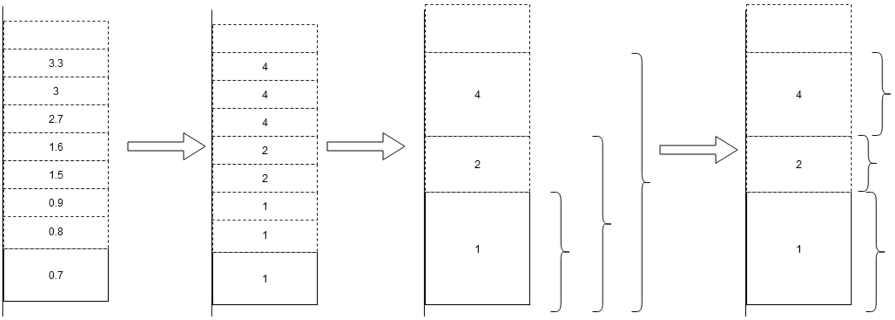


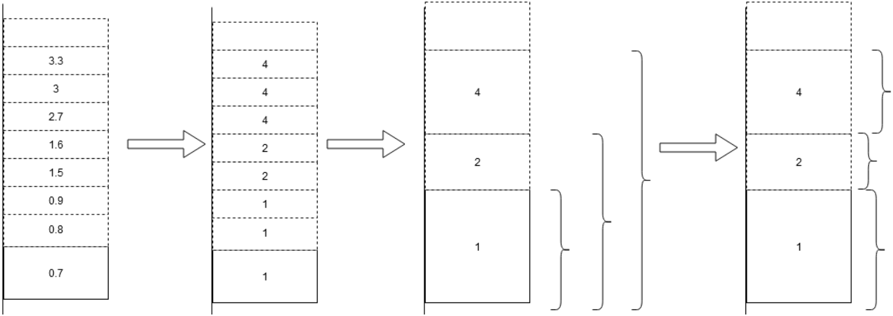
Προβληματικό:

- W το πλήθος ορθογώνια. Εκθετικά το πλήθος!
- Τι γίνεται με τα κόστη;

Λύση:

Υποθέτω ότι $w_j(t)$ είναι μόνο δυνάμεις τους 2.


3.3
3
2.7
1.6
1.5
0.9
0.8
0.7
4
4
4
2
2
1
1
1
4
2
1
4
2
1

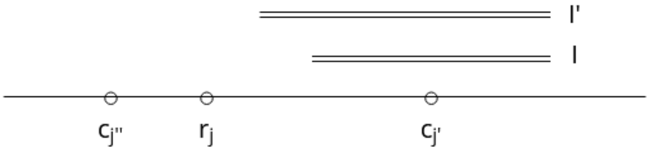


- 1ο βήμα: Απώλεια 2 στον λόγο προσέγγισης αναγωγής
- 3ο βήμα: Απώλεια 2 στον λόγο προσέγγισης αναγωγής

Αναγωγή GSP σε R2C

Παρατήρηση

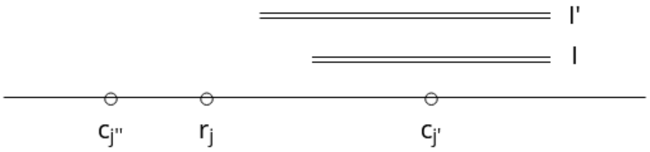
- Αφού παίρνουμε τις εργασίες να έχουνε $w_j(t)$ μόνο δύναμη του 2, μπορούμε κάλλιστα αντί να λάβουμε υπόψη κάθε δυνατό χρόνο λήξης, να λάβουμε υπόψη μόνο χρόνο λήξης ακριβώς πριν το κόστος αυξηθεί.
- Έχοντας αυτό υπόψη, κάθε εργασία έχει $\log W$ το πλήθος το πολύ πιθανά τέλη.
- Ομοίως δεν χρειάζεται να λάβουμε υπόψη κάθε δυνατό διάστημα I.



Αναγωγή GSP σε R2C

Παρατήρηση

Αρκεί να έχουμε ένα διάστημα από κάθε κενό υποδιάστημα σε κάθε κενό υποδιάστημα, από κάθε ώρα έναρξης ή υποψήφιας λήξης σε κάθε ώρα έναρξης ή υποψήφιας λήξης, και από κάθε ώρα έναρξης ή υποψήφιας λήξης σε κάθε κενό υποδιάστημα.



Αναγωγή GSP σε R2C

Χρόνος;

- Δεν χρειάζεται να λάβουμε υπόψη κάθε διάστημα I, αρκούν αυτά τα οποία ξεκινάνε και τελειώνουν με έναρξη ή λήξη εργασιών.
- Κάθε εργασία έχει logW το πλήθος το πολύ πιθανά τέλη, και 1 αρχή.

Λόγος προσέγγισης

- Αν η βέλτιστη λύση του GSP δίνει v, η βέλτιστη λύση του R2C δίνει το πολύ 4v.
- Αν η βέλτιστη λύση του R2C δίνει v, η βέλτιστη λύση του GSP δίνει το πολύ v.
- Λόγος προσέγγισης αναγωγής 4.

Αναγωγή GSP σε R2C

Αν η βέλτιστη λύση S του GSP δίνει v, υπάρχει βέλτιστη λύση S' του R2C δίνει το πολύ 4v.

Έστω σε αυτή την λύση, η j λήγει την χρονική στιγμή Cj. Στο πρώτο εγχείρημα αναγωγής που είδαμε, επιλέγοντας για κάθε j το $R_j(Cj)$, πετυχαίνουμε λύση ίδιου κόστους, η οποία είναι εφικτή αφού $\forall I$, αθροιστικό runtime δουλειών που φτάνουν στο I και έχουν προθεσμία μετά του $I \geq$ excess of I $\xi(I)$ συνεπάγεται ότι τα αντίστοιχα $R_j(Cj)$ ικανοποιούνε όλα τα σημεία. Πηγαίνοντας από το αρχικό εγχείρημα στο τελικό, έχουμε το πολύ απώλεια λόγου προσέγγισης 4 και ίδια ή καλύτερη ικανοποίηση σημείων.

Αναγωγή GSP σε R2C

Αν η βέλτιστη λύση S του R2C δίνει v, υπάρχει βέλτιστη λύση S' του R2C δίνει το πολύ v.

Έστω η δουλειά j, για την οποία στο R2C επιλέχθηκαν κάποια ορθογώνια, και έστω t η τετμημένη του τελευταίου αυτών. Στο αρχικό εγχείρημα, επιλέγουμε το $R_j(t)$. Αυτό πετυχαίνει ίδια ή καλύτερη κάλυψη με κόστος $w_j(t)$, ίδιο ή καλύτερο. Συνεπάγεται ότι η επιλογή προθεσμίας t για την δουλειά j ικανοποιεί την συνθήκη $\forall I$, αθροιστικό runtime δουλειών που φτάνουν στο I και έχουν προθεσμία μετά του I $\geq$ excess of I $\xi(I)$ με κόστος $w_j(t)$, ίδιο ή καλύτερο.

Επίλυση του R2C

Ειδική περίπτωση

Αν το r_j είναι ίδιο για κάθε δουλειά, το R2C είναι μονοδιάστατο και ανάγεται στο generalized caching problem, το οποίο έχει πολυωνυμικό αλγόριθμο με λόγο προσέγγισης 4. Άρα σε αυτή την περίπτωση πετυχαίνουμε λόγο προσέγγισης 16.

Επίλυση του R2C

Με λύση του γραμμικού προγράμματος και στρογγύλευση του σε ακέραια λύση.

Ακέραιο πρόγραμμα R2C

$$min \sum_r w_r x_r$$

$$s.t \sum_{r \in R : p \in r} c_r x_r \geq d_p, \forall p$$

$$x_r \in \{0,1\}, \forall r$$

Επίλυση του R2C

Χάσμα ακεραιότητας

Το χαλαρωμένο LP έχει αυθαίρετα μεγάλο χάσμα έστω και για 1 σημείο: Για 1 σημείο, το πρόβλημα μας είναι ακριβώς το Knapsack Cover, του οποίου το (αντίστοιχο) απλούστερο δυνατό γραμμικό πρόγραμμα έχει αυθαίρετα μεγάλο χάσμα.

Knapsack cover (KC)

Δεδομένων σακιδίου χωρητικότητας B και αντικειμένων με βάρη w_i και χωρητικότητες c_i, βρείτε συλλογή αντικειμένων ελάχιστου βάρους με χωρητικότητα $\geq B$.

Αντιμετώπιση

Θα χρησιμοποιήσουμε τεχνικές που λέγονται KC inequalities.

Επίλυση του R2C

Παρατήρηση

Χαλάρωση ενός γραμμικού προγράμματος (που διατηρεί ακέραιες λύσεις) αυξάνει το integrality gap του. Ενδυνάμωση ενός γραμμικού προγράμματος (που διατηρεί ακέραιες λύσεις) μειώνει το integrality gap του.

Γραμμικό πρόγραμμα R2C

$$min \sum_{r \in \mathcal{R}} w_r x_r$$

$$s.t \sum_{r \in \mathcal{R}: p \in r} c_r x_r \geq d_p, \forall p \in \mathcal{P}$$

$$x_r \in [0,1], \forall r \in \mathcal{R}$$

Επίλυση του R2C

Εξήγηση

Για S κενό, έχουμε την προηγούμενη ανισότητα. Σχετικά με άλλα S: Για κάθε p, και για κάθε υποσύνολο ορθογωνίων S, ακόμα και αν επιλεχθούν όλα τα S και αν περιέχουν το p, τουλάχιστον μία απαίτηση $d_p - c(S)$ πρέπει να καλυφθεί από τα υπόλοιπα ορθογώνια. Τα 2 γραμμικά προγράμματα είναι ισοδύναμα.

Γραμμικό πρόγραμμα R2C

$$min \sum_{r \in \mathcal{R}} w_r x_r$$

$$s.t \sum_{r \in \mathcal{R} \setminus S : p \in r} c_r x_r \geq d_p - c(S), \qquad \forall p \in \mathcal{P}, S \subseteq \mathcal{R}$$

$$x_r \in [0, 1], \qquad \forall r \in \mathcal{R}$$

Επίλυση του R2C

Εξήγηση

Η "κοπή" του c_r σε $d_p - c(S)$ προφανώς δεν αλλάζει την εφικτότητα ακέραιων λύσεων. Περιορίζει όμως τις γραμμικές λύσεις.

Γραμμικό πρόγραμμα R2C

$$min \sum_{r \in \mathcal{R}} w_r x_r$$

$$s.t \sum_{r \in \mathcal{R} \setminus S : p \in r} \min\{c_r, d_p - c(S)\} x_r \geq d_p - c(S), \quad \forall p \in \mathcal{P}, S \subseteq \mathcal{R}$$

$$x_r \in [0,1], \quad \forall r \in \mathcal{R}$$

Επίλυση του R2C

Εξήγηση

Negative values bad.

Γραμμικό πρόγραμμα R2C

$$min \sum_{r \in \mathcal{R}} w_r x_r$$

$$s.t \sum_{r \in \mathcal{R} \setminus S : p \in r} \min\{c_r, \max(0, d_p - c(S))\} x_r \geq d_p - c(S), \qquad \forall p \in \mathcal{P}, S \subseteq \mathcal{R}$$

$$x_r \in [0,1], \qquad \forall r \in \mathcal{R}$$

Επίλυση του R2C

Επίλυση LP

Μπορούμε να υπολογίσουμε, και το κάνουμε, μία $(1+\epsilon)$ προσέγγιση x της βέλτιστης λύσης του LP [1]. Παρεμπιπτόντως, οι περιορισμοί ικανοποιούνται ακριβώς.

Γραμμικό πρόγραμμα R2C

$$min \sum_{r \in \mathcal{R}} w_r x_r$$

$$s.t \sum_{r \in \mathcal{R} \setminus S : p \in r} \min\{c_r, \max(0, d_p - c(S))\} x_r \geq d_p - c(S), \quad \forall p \in \mathcal{P}, S \subseteq \mathcal{R}$$

$$x_r \in [0,1], \quad \forall r \in \mathcal{R}$$

Επίλυση του R2C

Κατασκευή στρογγύλευσης

Έστω β μία μικρή σταθερά, και το $\beta = 1/12$ επαρκεί. Προς κόστος στο λόγο προσέγγισης το πολύ 12, για κάθε r με $x_r \geq 1/12$, θέτουμε $x_r = 1$, επιλέγουμε δηλαδή αυτά τα ορθογώνια. Έστω S_p το σύνολο τέτοιων ορθογωνίων που περιέχουν το p. Θεωρώντας τα παρμένα, τα αφαιρούμε, μαζί με σημεία πλέον ικανοποιημένα.

Γραμμικό πρόγραμμα R2C

$$min \sum_{r \in \mathcal{R}} w_r x_r$$

$$s.t \sum_{r \in \mathcal{R} \setminus S : p \in r} \min\{c_r, \max(0, d_p - c(S))\} x_r \geq d_p - c(S), \qquad \forall p \in \mathcal{P}, S \subseteq \mathcal{R}$$

$$x_r \in [0,1], \qquad \forall r \in \mathcal{R}$$

Επίλυση του R2C

Επόμενα βήματα

- Από το x, με σταθερό κόστος στον λόγο προσέγγισης, φτιάχνουμε το x'.
- Από το x', με $O(\log\log(P))$ κόστος στον λόγο προσέγγισης, φτιάχνουμε ακέραια λύση x_{int} με την ιδιότητα

$$\sum_{r \in \mathcal{R} \setminus S_p : p \in r} c_r x_{r\,int} \geq d_p - c(S_p), \qquad \forall p \in \mathcal{P}$$

- Μαζί με το $\cup_p S_p$, αυτό θα δώσει ακέραια λύση για το πρόβλημα (στην απλή μορφή του).

Επίλυση του R2C

Κατασκευή στρογγύλευσης

Το εξής ίσχυε πριν, και συνεχίζει να ισχύει:

$$\sum_{r \in \mathcal{R} \setminus S_p : p \in r} \min\{c_r, \max(0, d_p - c(S_p))\} x_r \geq d_p - c(S_p), \qquad \forall p \in \mathcal{P}$$

Οι υπόλοιπες ανισότητες δεν μας ενδιαφέρουν πλέον.

Γραμμικό πρόγραμμα R2C

$$min \sum_{r \in \mathcal{R}} w_r x_r$$

$$s.t \sum_{r \in \mathcal{R} \setminus S : p \in r} \min\{c_r, \max(0, d_p - c(S))\} x_r \geq d_p - c(S), \qquad \forall p \in \mathcal{P}, S \subseteq \mathcal{R}$$

$$x_r \in [0, 1], \qquad \forall r \in \mathcal{R}$$

Επίλυση του R2C

Κατασκευή στρογγύλευσης

Για την ακρίβεια μπορεί να αφαιρεθεί το max τώρα.

$$\sum_{r \in \mathcal{R} \setminus S_p : p \in r} \min\{c_r, d_p - c(S_p)\} x_r \geq d_p - c(S_p), \quad \forall p \in \mathcal{P}$$

Γραμμικό πρόγραμμα R2C

$$min \sum_{r \in \mathcal{R}} w_r x_r$$

$$s.t \sum_{r \in \mathcal{R} \setminus S : p \in r} \min\{c_r, \max(0, d_p - c(S))\} x_r \geq d_p - c(S), \quad \forall p \in \mathcal{P}, S \subseteq \mathcal{R}$$

$$x_r \in [0,1], \quad \forall r \in \mathcal{R}$$

Επίλυση του R2C

Κατασκευή στρογγύλευσης

Πριν κάναμε scale up τα $x_r \geq 1/12$ σε 1. Κάνε scale up τα υπόλοιπα x_r κατά 12 φορές (συνολικά κόστος λόγου προσέγγισης 12), για να πάρεις το x'. $x'_r \in [0, 1]$

Ισχύει πως

$$\sum_{r \in \mathcal{R} \setminus S_p : p \in r} \min\{c_r, d_p - c(S_p)\} x'_r \geq 12(d_p - c(S_p)), \quad \forall p \in \mathcal{P}$$

Επίλυση του R2C

Κατασκευή στρογγύλευσης

Επαναορίζουμε την απαίτηση d'_p του p ως $d_p - c(S_p)$ στρογγυλεμένο πάνω προς την κοντινότερη ακέραια δύναμη του 2.
Επαναορίζουμε την χωρητικότητα c'_r του r ως c_r στρογγυλεμένο κάτω προς την κοντινότερη ακέραια δύναμη του 2.

Ισχύει πως

$$\sum_{r \in \mathcal{R} \setminus S_p : p \in r} \min\{c_r, d_p - c(S_p)\} x'_r \geq 12(d_p - c(S_p)), \quad \forall p \in \mathcal{P} \implies$$

$$\sum_{r \in \mathcal{R} \setminus S_p : p \in r} \min\{c'_r, d_p - c(S_p)\} x'_r \geq \frac{12}{2}(d_p - c(S_p)), \quad \forall p \in \mathcal{P} \implies$$

$$\sum_{r \in \mathcal{R} \setminus S_p : p \in r} \min\{c'_r, d'_p\} x'_r \geq \frac{12}{4} d'_p = 3d'_p, \quad \forall p \in \mathcal{P}$$

Επίλυση του R2C

Παρατήρηση

Σύνολο ορθογωνίων που ικανοποιεί d'_p μετά τις αλλαγές θα ικανοποιεί $d_p - c(S_p)$ πριν από αυτές. Αρκεί να ικανοποιήσουμε d'_p.

- Ορθογώνιο τάξης i: $c'_r = 2^i c'_{min}$. Το πολύ $O(\log(P))$ τάξεις.
- Σημείο τάξης i: $d'_p = 2^i c'_{min}$ (δυνητικά αρνητική).
- p βαρύ εάν καλύπτεται από ορθογώνια ίσης ή μεγαλύτερης τάξης στην LP λύση.
- p ελαφρύ αλλιώς.

$\mathcal{R}' := \mathcal{R} \setminus S_p$

Θα καλύψουμε τα ελαφρά και τα βαριά σημεία με διαφορετικές μεθόδους.

Επίλυση του R2C

Βαρύ p

$$\sum_{r\in\mathcal{R}':c'_r\geq d'_p} \min\{c'_r, d'_p\}x'_r \geq d'_p \iff$$

$$\sum_{r\in\mathcal{R}':c'_r\geq d'_p} d'_p x'_r \geq d'_p \iff$$

$$\sum_{r\in\mathcal{R}':c'_r\geq d'_p} x'_r \geq 1$$

Επίλυση του R2C

Ελαφρύ p

Από τις εξισώσεις

$$\sum_{r \in \mathcal{R}' : c'_r \geq d'_p} \min\{c'_r, d'_p\} x'_r < d'_p$$

$$\sum_{r \in \mathcal{R}' : p \in r} \min\{c'_r, d'_p\} x'_r \geq 3d'_p, \qquad \forall p \in \mathcal{P}$$

Έχουμε

$$\sum_{r \in \mathcal{R}' : c'_r < d'_p} c'_r x'_r \geq 2d'_p$$

Κάλυμμα βαρέων σημείων

Με αναγωγή στο Rectangle 3dimensions uncapacitated (R3U)

Ομοίως με R2C

Σύνολο σημείων στο Z^3 που απαιτούν να καλυφθούν από κυβοειδή.

Κάλυμμα βαρέων σημείων

Με αναγωγή στο Rectangle 3dimensions uncapacitated (R3U)

Ομοίως με R2C

Σύνολο σημείων στο Z^3 που απαιτούν να καλυφθούν από κυβοειδή.
Σημεία $p = (x, y, z)$. Σύνολο κυβοειδών $r =[0, x_r] \times [y_r^1, y_r^2] \times [0, z_r]$ με:

- Κάθε ορθογώνιο ικανοποιεί όλα τα σημεία που καλύπτει κατά c_r.
- Θετικό ακέραιο κόστος (βάρος) w_r
- Μια επιφάνεια προσκολλημένη στο επίπεδο $x - y$ μία στο $y - z$

Επιλογή κυβοειδών ελάχιστου αθροιστικού βάρους που να ικανοποιεί όλα τα σημεία. Δεν υπάρχουν απαιτήσεις, χωρητικότητες. Απλό κάλυμμα.

Κάλυμμα βαρέων σημείων

Με αναγωγή στο Rectangle 3dimensions uncapacitated (R3U)

Αναγωγή καλύμματος βαρέων σημείων σε R3U

Δοσμένων βαρεών σημείων (x, y), φτιάξε στο R3U το (x, y, d'_p). Δοσμένων ορθωγωνίων $[0, x] \times [y_1, y_2]$ φτιάξε κυβοειδές $[0, x] \times [y_1, y_2] \times [0, c'_r]$ βάρους w_r. Φαίνεται ότι ένα 3D σημείο καλύπτεται από κυβοειδές αν και μόνο αν το αντίστοιχο 2D καλύπτεται από το αντίστοιχο ορθογώνιο και $c'_r > d'_p$.

Παρατήρηση

Κυβοειδή έχουν το πολύ $O(\log(P))$ διαφορετικά βάθη.

Κάλυμμα βαρέων σημείων

Παρατήρηση

Καθώς για το x'

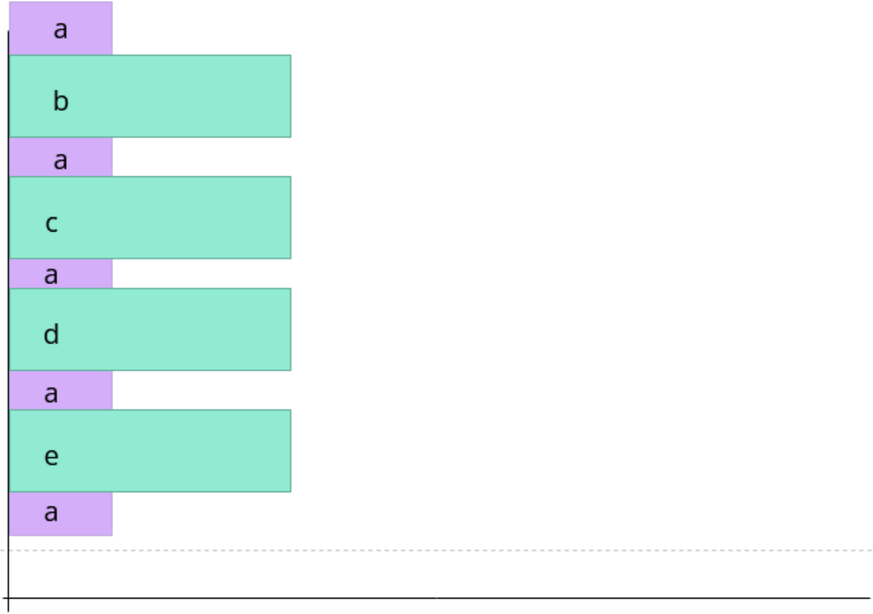
$$\sum_{r \in \mathcal{R}' : c'_r \geq d'_p} x'_r \geq 1$$

x' είναι και (γραμμική) λύση του R3U.

Για την λύση του R3U χρειάζεται να επικαλεστούμε το union complexity των κυβοειδών. Ας αρχίσουμε με ορθογώνια.

union complexity

Για κάθε συλλογή k ορθογωνίων όπως το R2C, το union complexity είναι $O(k)$.


a
b
a
c
a
d
a
e
a

Κάλυμμα βαρέων σημείων

union complexity R3U

Διαίσθηση: Έστω k κυβοειδή. Υπάρχουν $O(\log P)$ πιθανά βάθη, δυνάμεις του 2. Έστω 2 διαδοχικά τέτοια, 2^i, 2^{i+1}. Κοιτώντας την "φέτα" εντός τους, η φέτα αυτή αντιστοιχεί σε ορθογώνια. Καθώς υπάρχουν $O(\log P)$ φέτες για κάθε κυβοειδές, έχουμε $O(k \log P)$ union complexity.

Θεώρημα [2]

Έστω ένα geometric set multicover problem n σημείων, με union complexity k sets (κυβοειδών πχ) να είναι $kh(k)$ ($k \log(P)$) ή λιγότερο. Τότε έχουμε μία πολυωνυμική $O(\log h(n))$-προσέγγιση ($\log \log P$), και αυτή είναι και $O(\log h(n))$-προσέγγιση της βέλτιστης τιμής της (standard) γραμμικής χαλάρωσης του προβλήματος.

Κάλυμμα βαρέων σημείων

Λήμμα

Καθώς το x' είναι γραμμική λύση του R3U, έχουμε βρει μία προσεγγιστική λύση του R3U, δηλαδή του καλύμματος βαρέων σημείων με λόγο προσέγγισης $O(\log \log P)$.

Κάλυμμα ελαφριών σημείων

Rectangle 2D Multicover (R2M)

Το R2C αλλά με capacities 1.

Αναγωγή

Δοσμένων ελαφριών σημείων, φτιάξε instances του R2M, έστω B_l, για κάθε κλάση χωρητικότητας $l = 0, 1, \ldots \log P$. Η B_l περιέχει όλα τα ελαφριά σημεία, και μόνο τα ορθογώνια τάξης l (χωρητικότητας $2^l c'_{min}$). Τα κόστοι των ορθογωνίων είναι ίδια, όμως τα d_p είναι για την instance B_l,

$$d_p^l := \left\lfloor \sum_{r : p \in r, r \in B_l} x'_r \right\rfloor$$

δηλαδή το πλήθος των ορθογωνίων τάξης l που καλύπτουν το p στην (χαλαρωμένη γραμμική) λύση x', rounded down.

Κάλυμμα ελαφριών σημείων

Λήμμα

Το x' περιορισμένο σε ορθογώνια τάξης l είναι εφικτή λύση του instance B_l. Εξ'ορισμού της επιλογής των d_p^l.

Θεώρημα [2]

Έστω ένα geometric set multicover problem n σημείων, με union complexity k sets (ορθογωνίων πχ) να είναι $kh(k)$ $(k \cdot O(1))$ ή λιγότερο. Τότε έχουμε μία πολυωνυμική $O(\log h(n))$-προσέγγιση $(O(1))$, και αυτή είναι και $O(\log h(n))$-προσέγγιση της βέλτιστης τιμής της (standard) γραμμικής χαλάρωσης του προβλήματος.

Έχουμε O(1)-προσεγγιστική λύση του B_l για το x' περιορισμένο σε ορθογώνια τάξης l.

Κάλυμμα ελαφριών σημείων

Λήμμα

Έστω η ένωση των ακέραιων λύσεων (των επιλεγμένων ορθογωνίων) κάθε instance B_l. Αυτή είναι O(1)-προσέγγιση του x'.

Θεώρημα

Έστω η ένωση των ακέραιων λύσεων (των επιλεγμένων ορθογωνίων) κάθε instance B_l. Αυτή ικανοποιεί τα ελαφριά σημεία του R2C.

Intuition

Λόγω του ορισμού του d_p^l, το p χάνει 1 μονάδα ικανοποίησης ανά l το πολύ. Καθώς έχουμε

$$\sum_{r \in \mathcal{R}' : c'_r < d'_p} c'_r x'_r \geq 2d'_p$$

μικρό ποσό απώλειας είναι ανεκτό.

Επίλυση του R2C

Overview επίλυσης R2C

- Από το x, με σταθερό κόστος στον λόγο προσέγγισης, φτιάξαμε το x'.
- Από το x', με $O(\log\log(P))+O(1)$ κόστος στον λόγο προσέγγισης, φτιάξαμε ακέραια λύση $x_{int} = x^1_{int} \cup x^2_{int}$ με την ιδιότητα

$$\sum_{r \in \mathcal{R} \setminus S_p : p \in r} c_r {x^1}_{r\,int} \geq \sum_{r \in \mathcal{R} \setminus S_p : p \in r} c'_r {x^1}_{r\,int} \geq d'_p \geq d_p - c(S_p), \forall p \in \mathcal{P}_{heavy}$$

$$\sum_{r \in \mathcal{R} \setminus S_p : p \in r} c_r {x^2}_{r\,int} \geq \sum_{r \in \mathcal{R} \setminus S_p : p \in r} c'_r {x^2}_{r\,int} \geq d'_p \geq d_p - c(S_p), \forall p \in \mathcal{P}_{light}$$

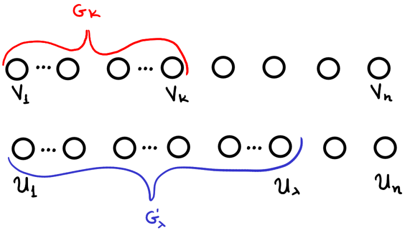
- Μαζί με το $\cup_p S_p$, αυτό δίνει ακέραια λύση για το πρόβλημα (στην απλή μορφή του).

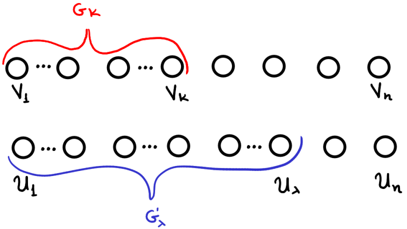
Κάλυμμα ελαφριών σημείων

Συμπεράσματα

- Βρήκαμε $O(\log \log P)$ προσέγγιση για το GSP, με την γεωμετρική μέθοδο.
- Μπορούμε να θέσουμε το LP του GSP δίχως γεωμετρικές αναγωγές.
- Μπορούμε στρογγυλεύοντας με βάση αυτό να πετύχουμε O(1)-προσέγγιση;
- Στο εύρος των γνώσεων μας, δεν έχει βρεθεί O(1)-προσέγγιση ούτε έχει δειχτεί ότι το integrality gap του είναι χειρότερο.

Ευχαριστούμε!





Βιβλιογραφία

1. R. D. Carr, L. Fleischer, V. J. Leung, and C. A. Phillips, Strengthening integrality gaps for capacitated network design and covering problems, in Proceedings of the ACM-SIAM Symposium on Discrete Algorithms, 2000, pp. 106–115.
2. N. Bansal and K. Pruhs, Weighted geometric set multi-cover via quasi-uniform sampling, in Proceedings of the European Symposium on Algorithms, 2012, pp. 145–156.