

Αλγόριθμοι και Πολυπλοκότητα

Εργασία εξαμήνου

Αλγόριθμοι Χρωματισμού Γραφημάτων

Οικονόμου Αναστασία

Αριθμός Μητρώου : 15590 (παλαιό) 1412 (νέο)

Επιβλέπων καθηγητής : Γκόγκος Χρήστος

Άρτα 2020-2021

Περιεχόμενα

Εισαγωγή

Στη παρούσα εργασία στόχος μας είναι η επίλυση του προβλήματος χρωματισμού γραφημάτων με τη χρήση τεσσάρων διαφορετικών αλγορίθμων. Όμως αρχικά θα πρέπει να επισημάνουμε μερικές έννοιες, όπως για παράδειγμα τι ονομάζουμε κλάση P ,τι NP καθώς και τη σημασία της ΝP πληρότητας.

**Κλάση P**

Η γενική κλάση ερωτημάτων για το οποίο κάποιος αλγόριθμος δίνει την απάντηση σε πολυωνυμικό χρόνο ονομάζεται "κλάση P" ή απλούστερα "P". Για κάποια ερωτήματα δεν υπάρχει γνωστός τρόπος για την γρήγορη εύρεση απάντησης, αλλά αν κάποιος διαθέτει πληροφορίες που να αποδεικνύουν ποια είναι η απάντηση, είναι δυνατό να επιβεβαιώσει την απάντηση γρήγορα. Η κλάση των προβλημάτων που μπορούν να "επιβεβαιωθούν" σε πολυωνυμικό χρόνο ονομάζεται κλάση NP.[1]

**Κλάση ΝP**

Η κλάση πολυπλοκότητας N P αντιστοιχεί σε όλα τα προβλήματα για τα οποία υπάρχει πολυωνυμικός ανταιτιοκρατικός (non-deterministic) αλγόριθμος. Υπάρχει όμως ένας ισοδύναμος ορισμός που είναι εξαιρετικά πιο διαισθητικός και δείχνει ακριβώς τη σημασία αυτής της κλάσης πολυπλοκότητας. Ένα πρόβλημα P ανήκει στην κλάση N P αν έχει την ιδιότητα της πολυωνυμικής επαληθευσιμότητας. [2]

**NP-πληρότητα**

Σε αυτό το σημείο θα ορίσουμε την κλάση N PC (N P-Complete) που είναι υποσύνολο της κλάσης N P. Αυτή περιλαμβάνει όλα τα N P-πλήρη προβλήματα που υπό μία έννοια είναι τα πιο δύσκολα της κλάσης N P. Ένα πρόβλημα Π λέγεται N P-πλήρες, όταν ισχύουν τα εξής:

1. Π ∈ N P

2. ∀Ξ ∈ N P ⇒ Ξ ≤ Π

Για να δείξουμε, λοιπόν, ότι ένα πρόβλημα Π είναι N P-πλήρες θα πρέπει να δείξουμε ότι ανήκει στην κλάση N P κατασκευάζοντας έναν πολυωνυμικό επαληθευτή και έπειτα να δείξουμε ότι κάθε πρόβλημα στην κλάση N P ανάγεται στο Π. Το δεύτερο μέρος, όμως, είναι αρκετά πολύπλοκο και για αυτό το λόγο αν αποδείξουμε ότι ισχύει για ένα πρόβλημα, έπειτα κάνοντας αναγωγές από αυτό θα αποδεικνύουμε ότι και άλλα προβλήματα είναι N P-πλήρη. Το γνωστό θεώρημα των Cook-Levin δίνει το πρώτο τέτοιο πρόβλημα που είναι N P-πλήρες. [2]

**Περιγραφή του προβλήματος**

Το πρόβλημα του χρωματισμού γραφήματος είναι ένα NP‐hard πρόβλημα συνδυαστικής βελτιστοποίησης. Αφορά την ανάθεση ενός χρώματος σε κάθε κορυφή ενός γραφήματος έτσι ώστε γειτονικές κορυφές να χρωματίζονται με διαφορετικό χρώμα (όπως στο Σχήμα 1), ενώ παράλληλα χρησιμοποιείται ο ελάχιστος αριθμός διαφορετικών χρωμάτων. Στην παρούσα εργασία ζητείται η υλοποίηση τεσσάρων αλγορίθμων χρωματισμού γραφημάτων και η εφαρμογή τους σε γνωστά προβλήματα από τη βιβλιογραφία.

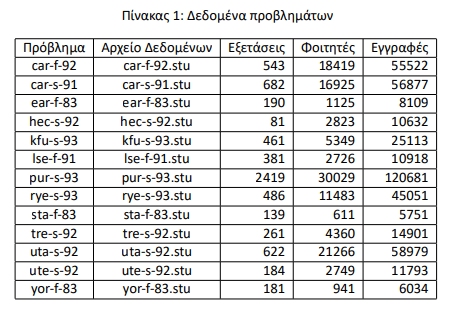
Το πρόβλημα χρωματισμού γραφήματος τυπικά ορίζεται ως εξής. Δεδομένου ενός μη κατευ‐ θυνόμενου απλού γραφήματος G = (V, E) με ένα σύνολο κορυφών V και ένα σύνολο ακμών E, ζητείται η ανάθεση σε κάθε κορυφή v ∈ V ενός ακεραίου c(v) ∈ {1, 2, ..., k} έτσι ώστε το k να ελαχιστοποιείται και να ισχύει ότι c(v) ≠ c(u) ∀{v, u} ∈ E. Το πρόβλημα συναντάται σε μεγάλο αριθμό πρακτικών εφαρμογών όπως ο χρονοπρογραμ‐ ματισμός εκπαιδευτικών ιδρυμάτων (educational timetabling), ο χρονοπογραμματισμός αθλητι‐ κών γεγονότων (sports scheduling), η ανάθεση συχνοτήτων (frequency assignment), η ανάθεση καταχωρητών στους μεταγλωττιστές (compiler register allocation) και άλλα. Πολλοί αλγόριθμοι χρωματισμού γραφημάτων έχουν προταθεί τα τελευταία 50 έτη. Στην πα‐ ρούσα εργασία θα εξεταστούν τέσσερις αλγόριθμοι που ανήκουν στις λεγόμενες κατασκευαστι‐ κές τεχνικές (constructive techniques). Οι κατασκευαστικές τεχνικές δημιουργούν λύσεις βήμα προς βήμα, αναθέτοντας στη σειρά, σε κάθε κορυφή, ένα χρώμα, πιθανά εφαρμόζοντας οπι‐ σθοχώρηση κατά τη διαδικασία. Οι αλγόριθμοι που θα εξεταστούν είναι ο αλγόριθμος first fit, ο αλγόριθμος DSATUR, ο αλγόριθμος Recursive Largest First και ο αλγόριθμος backtracking DSATUR. Πληροφορίες για τους ανωτέρω αλγορίθμους μπορούν να βρεθούν στο άρθρο [LTMG12] καθώς και στις αναφορές του ίδιου άρθρου.

**Προσεγγίσεις επίλυσης**

**Δεδομένα προβλήματος**

Το πρόβλημα χρονοπρογραμματισμού εξετάσεων αφορά φοιτητές που έχουν πραγματοποιήσει εγγραφές σε εξετάσεις μαθημάτων. Για κάθε εξέταση διατίθεται μια λίστα από φοιτητές και κάθε φοιτητής μπορεί να είναι εγγεγραμμένος σε μια ή περισσότερες εξετάσεις. Κάθε εξέταση θα πρέπει να τοποθετηθεί σε μια περίοδο εξέτασης και η λύση του προβλήματος συνίσταται στην ανάθεση όλων των εξετάσεων στο μικρότερο δυνατό αριθμό περιόδων έτσι ώστε να μην υπάρχουν συγκρούσεις, δηλαδή να μην υπάρχουν φοιτητές που θα έπρεπε να συμμετάσχουν σε εξετάσεις σε περισσότερα του ενός μαθήματα στην ίδια περίοδο. Ως δεδομένα του προβλήματος θα χρησιμοποιηθούν τα δεδομένα του προβλήματος χρονοπρογραμματισμού εξετάσεων Toronto τα οποία είναι διαθέσιμα προς μεταφόρτωση στη διεύθυνση : <https://github.com/chgogos/datasets/blob/main/UETT/toronto.zip>.

Τα δεδομένα Toronto αποτελούνται από 13 προβλήματα και πληροφορίες για κάθε πρόβλημα παρουσιάζονται στον Πίνακα 1. Τα αρχεία δεδομένων (κατάληξη .stu) διαθέτουν για κάθε σπουδαστή μια γραμμή που περιέχει τους αριθμούς των μαθημάτων στα οποία είναι εγγεγραμμένος χωρισμένους μεταξύ τους με κενά. Η πρώτη γραμμή του αρχείου αντιστοιχεί στον πρώτο σπουδαστή, η δεύτερη γραμμή στο δεύτερο σπουδαστή κ.ο.κ. Για παράδειγμα το αρχείο car‐f‐92.stu περιέχει 18419 σειρές δεδομένων και ξεκινά με τις ακόλουθες σειρές: 0170 0156 0281 0006 0154 0156 0383 0534 0535 0536 0275 0091 0160 0164 ... που σημαίνουν ότι ο φοιτητής 1 έχει εγγραφεί στο μάθημα 0170, ο φοιτητής 2 έχει εγγραφεί στο μάθημα 0156, ο φοιτητής 3 έχει εγγραφεί στο μάθημα 0281, ο φοιτητής 4 έχει εγγραφεί στο μάθημα 0006, ο φοιτητής 5 στα μαθήματα 0154 0156 κ.ο.κ



Θεωρώντας κάθε εξέταση ως κόμβο ενός γραφήματος και κάθε ακμή ανάμεσα σε δύο κόμβους να υποδηλώνει την ύπαρξη κοινών φοιτητών ανάμεσα στις δύο εξετάσεις που βρίσκονται στα άκρα της ακμής, το πρόβλημα μπορεί να θεωρηθεί ως πρόβλημα χρωματισμού γραφήματος όπου κάθε χρώμα είναι και μια περίοδος εξέτασης.

**Στατιστικά στοιχεία προβλημάτων**

Θα εμφανίζονται τα ακόλουθα στατιστικά στοιχεία για καθένα από τα 13 προβλήματα του Toronto dataset:

1. Αριθμός κορυφών.

2. Πυκνότητα. Για τον υπολογισμό της πυκνότητας θα πρέπει να κατασκευαστεί ο πίνακας συγκρούσεων.

Ο πίνακας συγκρούσεων είναι ένας δισδιάστατος πίνακας c στον οποίο κάθε στοιχείο cij = 1 αν η εξέταση i βρίσκεται σε σύγκρουση με την εξέταση j ενώ ισχύει ότι cij = 0 σε άλλη περίπτωση. Η πυκνότητα συγκρούσεων υπολογίζεται διαιρώντας τον αριθμό των στοιχείων του πίνακα συγκρούσεων που έχουν την τιμή 1 με το συνολικό πλή‐ θος των στοιχείων του πίνακα. 3. Για τους βαθμούς (degrees) των κορυφών η ελάχιστη τιμή (min), η διάμεσος τιμή (median), η μέγιστη τιμή (max), η μέση τιμή (mean) καθώς και ο συντελεστής διακύμανσης (CV=coefficient of variation) που ορίζεται ως η τυπική απόκλιση προς τη μέση τιμή.

**Επίλυση προβλήματος.**

* Επίλυση προβλήματος με first fit

Ο αλγόριθμος first fit είναι ένας άπληστος (greedy) αλγόριθμος που λαμβάνει κάθε κορυφή και την αναθέτει στο μικρότερο αριθμό χρώματος που δεν προκαλεί σύγκρουση, δημιουργώ‐ ντας νέα χρώματα όταν χρειάζεται. Οι κορυφές μπορούν αρχικά να ταξινομηθούν σε φθίνουσα σειρά βαθμού, όπως έχει προταθεί στο [WP67] και ο χρωματισμός των κορυφών να γίνει από την κορυφή με τον υψηλότερο βαθμό προς την κορυφή με το χαμηλότερο βαθμό.

ΒΙΒΛΙΟΓΡΑΦΙΑ

[1] <https://el.wikipedia.org/wiki/%CE%A0%CF%81%CF%8C%CE%B2%CE%BB%CE%B7%CE%BC%CE%B1_P%3DNP>

[2] <https://repository.kallipos.gr/bitstream/11419/4015/1/chapter10Final.pdf>