Υλοποίηση Αλγορίθμων για AT-free Γραφήματα

Δημήτριος Σίντος

Διπλωματική Εργασία

Επιβλέπων: Λεωνίδας Παληός

Ιωάννινα, Φεβρουάριος, 2024

Τμήμα Μηχ. Η/Υ & Πληροφορικής
Πανεπιστήμιο Ιωαννίνων

Department of Computer Science & Engineering
University of Ioannina

ΕΥΧΑΡΙΣΤΙΕΣ

Ευχαριστώ θερμά τον επιβλέποντα καθηγητή μου κ. Λεωνίδα Παληό, για την άριστη συνεργασία μας και τις εύστοχες παρατηρήσεις του σε κάθε βήμα και δυσκολία της εργασίας αυτής. Επίσης, ιδιαίτερα ευχαριστώ τους γονείς και συγγενείς μου για την συνεχή τους υποστήριξη και καθοδήγηση, καθώς και τους συμφοιτητές και φίλους μου για το ενδιαφέρον και την συμπαράστασή τους.

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΕΡΙΛΗΨΗ

Δημήτριος Σίντος, Δίπλωμα, Τμήμα Μηχανικών Η/Υ και Πληροφορικής, Πολυτεχνική Σχολή, Πανεπιστήμιο Ιωαννίνων, Φεβρουάριος 2024.
Υλοποίηση Αλγορίθμων για AT-free Γραφήματα.
Επιβλέπων: Λεωνίδας Παληός, Καθηγητής.

Αστεροειδές τριάδα (AT σύντομα) είναι ένα σύνολο τριών κορυφών ενός γραφήματος τέτοιο ώστε να υπάρχει μονοπάτι μεταξύ οποιωνδήποτε δύο από αυτές αποφεύγοντας τη γειτονιά της τρίτης. Τα γραφήματα που δεν περιέχουν αστεροιειδή τριάδα ονομάζονται AT-free.Η κατηγορία των AT-free γραφημάτων είναι ένας τύπος γραφήματος για τον οποίο πολλά προβλήματα που είναι NP-πλήρη σε γενικότερα γραφήματα μπορούν να λυθούν σε πολυωνυμικό χρόνο.

Σε αυτή τη διπλωματική εργασία έχουμε υλοποιήσει τρεις αλγορίθμους πολυωνυμικού χρόνου για τα AT-free γραφήματα.

Συγκεκριμένα, τον αλγόριθμο υπολογισμού μέγιστου ανεξάρτητου συνόλου των Hajo Broersma , Ton Kloks , Dieter Kratsch , και Haiko Müller[2]. Τον αλγόριθμο υπολογισμού Ελάχιστου Κυρίαρχου Συνόλου του Dieter Kratsch[18]. Και τον αλγόριθμο για το πρόβλημα του 3-Χρωματισμού του Juraj Stacho[16].

ABSTRACT

Dimitrios Sintos, Diploma, Department of Computer Science and Engineering, School of Engineering, University of Ioannina, Greece, February 2024.
Implementation of Algorithms for AT-free Graphs.
Advisor: Leonidas Palios, Professor.

Asteroidal triple (AT for short) is a set of three vertices of a graph such that there is a path between any two of them avoiding the neighborhood of the third. Graphs that do not contain an asteroidal triple are called AT-free.The class of AT-free graphs is a type of graph for which many problems that are NP-complete in general graphs can be solved in polynomial time.

In this thesis, we have implemented three polynomial-time algorithms for AT-free graphs.

Specifically, the algorithm for computing the maximum independent set of Hajo Broersma, Ton Kloks, Dieter Kratsch, and Haiko Müller[2]. The algorithm for computing the Minimum Dominating Set of Dieter Kratsch[2]. And the algorithm for the 3-Coloring problem of Juraj Stacho[16].

ΚΕΦΑΛΑΙΟ 1

Εισαγωγή

Η αστεροειδή τριάδα (asteroidal triple) εισήχθη το 1962 για να χαρακτηρίσουν τα interval γραφήματα ως εκείνα τα χορδωτά γραφήματα που δεν περιέχουν μια αστεροειδή τριάδα[19]. Αποτελούν μια μεγάλη κατηγορία γραφημάτων που περιλαμβάνει interval, permutation, trapezoid, και cocomparability γραφήματα. Στη παρούσα εργασία παρουσιάζουμε την υλοποίηση τριών αλγορίθμων για τα AT-free γραφήματα που τρέχουν σε πολυωνυμικό χρόνο. Τον αλγόριθμο των Broersma, H., Kloks, T., Kratsch, D. και Müller, H.[2] για τον υπολογισμό του μέγιστου ανεξάρτητου συνόλου, με χρονική πολυπλοκότητα $O(n^4)$. Τον αλγόριθμο του Dieter Kratsch[18] για τον υπολογισμό του ελάχιστου κυρίαρχου συνόλου, με χρονική πολυπλοκότητα $O(n^6)$. Και τον αλγόριθμο του Juraj Stacho[16] για την επίλυση του προβλήματος του 3-χρωματισμού, με χρονική πολυπλοκότητα $O(n^2m)$

1.1 Βασικοί Ορισμοί

Ακολουθούν βασικοί ορισμοί που απαιτούνται για την παρούσα εργασία.

Ορισμός 1.1. *Γράφημα (Graph)* είναι μια δομή που αποτελείται από ένα σύνολο κορυφών που συνδέονται μεταξύ τους με ένα σύνολο ακμών και το συμβολίζουμε με $G = (V, E)$, όπου V και E είναι τα σύνολα των κορυφών και των ακμών αντίστοιχα.

Ορισμός 1.2. *Μονοπάτι (Path)* μεταξύ δύο κορυφών σε ένα γράφημα ονομάζεται μια ακολουθία διαφορετικών κορυφών, όπου κάθε κορυφή της ακολουθίας συνδέεται με την επόμενή της μέσω ακμής.

Ορισμός 1.3. Έστω γράφημα G. Λέμε ότι ένα σύνολο κορυφών $S \subseteq V(G)$ είναι *ανεξάρτητο σύνολο* του G αν κανένα ζεύγος κορυφών από το S δεν είναι ακμή του G.

Ορισμός 1.4. Ο *αριθμός ανεξαρτησίας* $\alpha(G)$ ενός γραφήματος G, είναι το μέγιστο πλήθος ενός ανεξάρτητου συνόλου του G.

Ορισμός 1.5. Ένα *κυρίαρχο σύνολο* ενός γραφήματος G είναι ένα υποσύνολο κορυφών D τέτοιο ώστε κάθε κορυφή του G είτε ανήκει στο D είτε είναι γειτονική με μία κορυφή που ανήκει στο D.

Ορισμός 1.6. Ο *αριθμός κυριαρχίας* $\gamma(G)$ είναι ο αριθμός των κορυφών στο μικρότερο σύνολο κυριαρχίας του G.

Ορισμός 1.7. Ο *χρωματικός αριθμός* ενός γραφήματος G είναι το ελάχιστο πλήθος χρωμάτων που απαιτούνται για να χρωματιστούν οι κορυφές του G έτσι ώστε να μην υπάρχουν δύο γειτονικές κορυφές με το ίδιο χρώμα.

Ορισμός 1.8. Το *πρόβλημα του 3-χρωματισμού* αφορά εάν ένα γραφήμα G μπορεί να χρωματιστεί χρησιμοποιώντας μόνο τρία χρώματα, έτσι ώστε να μην υπάρχουν δύο γειτονικές κορυφές με το ίδιο χρώμα.

Ορισμός 1.9. Μια *αστεροειδής τριάδα* σε ένα γράφημα αποτελείται από τρεις μη γειτονικές κορυφές, έτσι ώστε να υπάρχει μονοπάτι μεταξύ κάθε ζευγαριού από αυτές που αποφεύγει την κλειστή γειτονιά της τρίτης.

Ορισμός 1.10. Ένα γράφημα ονομάζεται *AT-free* εάν δεν περιέχει καμία αστεροειδή τριάδα.

1.2 Αντικείμενο της Διπλωματικής Εργασίας

Στην παρούσα διπλωματική εργασία, μελετούμε και υλοποιούμε τρεις αλγορίθμους για την επίλυση προβλημάτων σε AT-free γραφήματα.

Ειδικότερα, μελετάμε τον αλγόριθμο των Hajo Broersma , Ton Kloks , Dieter Kratsch , και Haiko Müller[2] για τον υπολογισμό του ανεξάρτητου αριθμού. Τον αλγόριθμο του Dieter Kratsch[18] για τον εντοπισμό ελάχιστου κυρίαρχου συνόλου και τον αλγόριθμο του Juraj Stacho[16] για την αντιμετώπιση του προβλήματος του 3-χρωματισμού σε AT-free γραφήματα.

Ο σκοπός αυτής της εργασίας είναι η βαθύτερη κατανόηση της δομής και των αλγοριθμικών ιδιοτήτων των AT-free γραφημάτων, μέσω της εφαρμογής και επικύρωσης αυτών των αλγορίθμων. Όλοι αυτοί οι αλγόριθμοι επιτυγχάνουν το αποτέλεσμα τους σε πολυωνυμικό χρόνο.

1.3 Σχετικά Ερευνητικά Αποτελέσματα

Η εύρεση της αλγοριθμικής πολυπλοκότητας των ανεξάρτητων συνόλων σε AT-free γραφήματα αποτελεί μια σημαντική πρόκληση. Αν και τα ανεξάρτητα σύνολα είναι ένα κλασσικό NP-πλήρες πρόβλημα, τα AT-free γραφήματα παρουσιάζουν μοναδικές προκλήσεις. Σε αντίθεση με άλλες υποκατηγορίες, όπως τα cocomparability γραφήματα, τα γραφήματα AT-free δεν είναι τέλεια. Ως εκ τούτου, οι πολυωνυμικοί αλγόριθμοι που αναπτύχθηκαν για τέλεια γραφήματα, όπως αυτοί των Grötschel, Lovász και Schrijver[22], δεν εφαρμόζονται σε αυτή την περίπτωση.

Για τον υπολογισμό ενός ελάχιστου κυρίαρχου συνόλου έχουν σχεδιαστεί αλγόριθμοι πολυωνυμικού χρόνου για πολλές κατηγορίες γράφων (βλ.[25, 6]). Για παράδειγμα, υπάρχουν αποδοτικοί αλγόριθμοι για τον υπολογισμό ενός ελάχιστου κυρίαρχου συνόλου για τις ακόλουθες κατηγορίες γραφημάτων: interval graphs [21], strongly chordal graphs[11, 20] , cographs[3], permutation graphs[4, 8, 24, 24], k-polygon graphs [7], cocomparability graphs [14, 17], circular-arc graphs [23] and dually chordal graphs[1]. Ιδιαίτερα ενδιαφέρον είναι ο αλγόριθμος των Breu και Kirkpatrick σχετικά με μια υποκατηγορία των AT-free γραφημάτων. Έχουν δώσει αλγορίθμους $O(nm^2)$ για τον υπολογισμό ελάχιστου κυρίαρχου συνόλου και ενός ολικού ελάχιστου κυρίαρχου συνόλου στα cocomparability γραφήματα[14]

Το πρόβλημα του χρωματισμού είναι ένα από τα πρώτα προβλήματα που γνωρίζουμε ότι είναι NP-hard[10]. Αυτό ισχύει και για ιδικές κλάσεις γραφημάτων όπως τα planar graphs[9], line graphs[15], regular graphs[5] ή ακόμα και για σταθερό αριθμό k χρωμάτων (γνωστό και ώς το πρόβλημα του 3-χρωματισμού)[9].Αντίθετα, για σενάρια με δύο ή λιγότερα χρώματα, το πρόβλημα επιλύεται σε πολυωνυμικό χρόνο. Αυτό ισχύει επίσης για ορισμένες κατηγορίες γραφημάτων με μοναδικές ιδιότητες, όπως τα interval graphs [12], chordal graphs [12], comparability graphs [12], και γενικότερα για τέλεια γραφήματα[13].

1.4 Δομή της Διπλωματικής Εργασίας

Η διπλωματική εργασία αναπτύσσεται σε τρία κεφάλαια. Στο κεφάλαιο 2 θα μελετήσουμε του τρεις αλγορίθμους. Στο επόμενο κεφάλαιο ??, παρουσιάζεται η λεπτομερής αναπαράσταση των δεδομένων, οι χρήσιμες δομές δεδομένων και η υλοποίηση κάθε αλγορίθμου. Στο τελευταίο κεφάλαιο ?? γίνεται σύνοψη των ευρημάτων, αναφέρονται δυνατές βελτιώσεις και προτείνεται πώς η υπάρχουσα υλοποίηση θα μπορούσε να τροποποιηθεί για την εκπόνηση πιστοποιητικών που επιβεβαιώνουν την ακρίβεια των αποτελεσμάτων.

ΚΕΦΑΛΑΙΟ 2

Οι Αλγόριθμοι

Σε αυτό το κεφάλαιο θα μελετήσουμε με λεπτομέρεια όλους τους αλγορίθμους που υλοποιήσαμε για τα AT-free γραφήματα. Για κάθε αλγόριθμο θα δώσουμε τον συμβολισμό και τα λήμματα που χρειάζονται για την περιγραφή του. Αμέσως μετά, θα εξηγήσουμε την πολυπλοκότητά του και συγκεκριμένα βήματά του, που θεωρήσαμε πιο ιδιαίτερα. Τέλος θα δώσουμε και ένα απλό παράδειγμα επίλυσης του κάθε αλγορίθμου.

Διευκρινίζουμε ότι δεν αποδεικνύουμε την ορθότητα του κάθε αλγορίθμου που χρησιμοποιούμε. Οι αποδείξεις αυτές βρίσκονται στις αντίστοιχες αναφορές([2], [18], [16])

2.1 Υπολογισμός Μέγιστου Ανεξάρτητου Συνόλου

Συμβολίζουμε τον αριθμό των κορυφών ενός γραφήματος $G = (V, E)$ με n και τον αριθμό των ακμών με m. Υπενθυμίζουμε ότι ένα ανεξάρτητο σύνολο σε ένα γράφημα G είναι ένα σύνολο από ζεύγη με μη γειτονικές κορυφές. Ο αριθμός ανεξαρτησίας ενός γραφήματος G, συμβολίζεται ως $\alpha(G)$ είναι το μέγεθος του μεγαλύτερου ανεξάρτητου συνόλου στο γράφημα. Οι βασικές δομικές ιδιότητες που πρέπει να αναλύσουμε πριν την περιγραφή του αλγορίθμου είναι τα $Components$ και τα $Intervals$.

Σε ένα AT-free γράφημα G όπου το x και το y είναι δύο ξεχωριστές μη γειτονικές κορυφές του G. Συμβολίζουμε με $C^x(y)$ το $Component$ του $G - N[x]$ όπου εμπεριέχεται το y, και με $r(x)$ τον αριθμό των $Components$ του $G - N[x]$.

Ορισμός 2.1. Μια κορυφή $z \in V \setminus \{x, y\}$ είναι μεταξύ των κορυφών x και y εάν οι

x και z βρίσκονται στον ίδιο $Component$ του $G - N[x]$. Εναλλακτικά, η κορυφή z είναι μεταξύ των x και y στο γράφημα G αν υπάρχει μονοπάτι από τον x στον z που αποφεύγει το $N[y]$ και μονοπάτι από τον y στον z που αποφεύγει το $N[x]$.

Ορισμός 2.2. Το διάστημα $I = I(x, y)$ του G είναι το σύνολο όλων των κορυφών του G που βρίσκονται μεταξύ x και y. Συνεπώς, $I(x, y) = C_x(y) \cap C_y(x)$.

Ο αλγόριθμος των Broersma, H., Kloks, T., Kratsch, D. και Müller, H. του καθορίζει τον αριθμό ανεξαρτησίας κάθε $Component$ και κάθε $Interval$ χρησιμοποιώντας τις σχέσεις που δίνονται στα Λήμματα 2.1, 2.2 και 2.3.

Λήμμα 2.1. *Έστω ότι $G = (V, E)$ είναι οποιαδήποτε γράφημα. Τότε*

$$\alpha(G) = 1 + \max_{x \in V} \left(\sum_{i=1}^{r(x)} \alpha(C_i^x) \right)$$

όπου $C_1^x, C_2^x, ..., C_r(x)^x$ τα Compontes του $G - N[x]$.

Λήμμα 2.2. *Έστω ότι $G = (V, E)$ είναι ένα AT-free γράφημα. Έστω $x \in V$ και έστω C^x ένα Component του $G - N[x]$. Τότε*

$$\alpha(C^x) = 1 + \max_{y \in C^x} \left(\alpha(I(x, y)) + \sum_i \alpha(D_i^y) \right)$$

όπου τα D_i^y είναι Components του $G - N[y]$ που εμπεριέχονται στο C^x.

Λήμμα 2.3. *Έστω ότι $G = (V, E)$ είναι ένα AT-free γράφημα. Έστω $I = I(x, y)$ είναι ένα $Interval$ του G. Αν $I = \emptyset$ τότε $\alpha(I) = 0$. Αλλιώς*

$$\alpha(I) = 1 + \max_{s \in I} \left(\alpha(I(x, s)) + \alpha(I(s, y)) + \sum_i \alpha(C_i^s) \right)$$

Σε αυτό το σημείο μπορούμε να δώσουμε τον αλγόριθμο για τον υπολογισμό του αριθμού ανεξαρτησίας $\alpha(G)$ για AT-free γραφήματα, ο οποίος είναι βασισμένος στον δυναμικό προγραμματισμό.

Αλγόριθμος 2.1 Αλγόριθμος υπολογισμού αριθμού ανεξαρτησίας σε AT-free γραφήματα

Είσοδος: Ένα AT-free γράφημα G.

Έξοδος: Αριθμός ανεξαρτησίας $\alpha(G)$

1: Για κάθε $x \in V$, υπολόγισε όλα τα *Components* $C_1^x, C_2^x, \ldots, C_{r(x)}^x$
2: Για κάθε ζευγάρι μη γειτονικών κορυφών x και y, υπολόγισε το *Interval* $I(x, y)$.
3: Ταξινόμησε όλα τα *Components* και τα *Intervals* με βάση το μη-αύξοντα αριθμό κορυφών.
4: Υπολόγισε τα $\alpha(C)$ και $\alpha(I)$ για κάθε *Components* C και κάθε *Interval* I με τη σειρά του Βήματος 3.
5: Υπολόγισε το $\alpha(G)$.

Ορισμός 2.3. Υπάρχει αλγόριθμος χρόνου $O(n^4)$ για τον υπολογισμό του ανεξάρτητου αριθμού ενός AT-free γραφήματος.

Για την πολυπλοκότητα του αλγορίθμου μελετάμε κάθε βήμα ξεχωριστά. Το πρώτο βήμα μπορεί να υλοποιηθεί σε χρόνο $O(n(n + m))$ χρησιμοποιώντας έναν γραμμικό αλγόριθμο για τον υπολογισμό των *Components*.

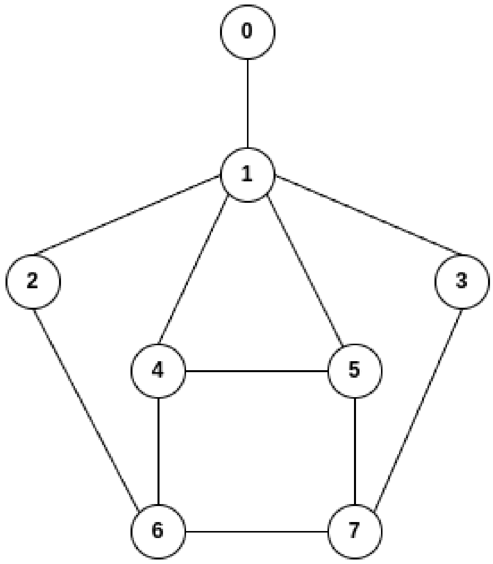
Το βήμα 2 υπολογίζει *Intervals* για τις μη γειτονικές κορυφές x και y, χρησιμοποιώντας την τομή των συνιστωσών $C_x(y)$ και $C_y(x)$. Η διαδικασία, που εκτελείται σε χρόνο O(n) για κάθε *Interval*, οδηγεί σε συνολική χρονική πολυπλοκότητα $O(n^3)$. Η υλοποίηση αξιοποιεί ένα λεξικό με αντικείμενα της κλάσης *Interval*, παρέχοντας έναν αποτελεσματικό τρόπο διαχείρισης και υπολογισμού εντός το πολύ n^2 *Component* και *Intervals*.

Με την χρήση του Bucket sort το βήμα 3 υλοποιείται σε $O(n^2)$.

Το σημείο φραγμού για τη χρονική πολυπλοκότητα του αλγορίθμου μας είναι το βήμα 4 αφού εκτελείται σε $O(n^4)$. Για κάθε *Component* C_x του $G - N[x]$ και μια κορυφή $y \in C_x$, ο αλγόριθμος πρέπει να υπολογίσει τα *Components* του $G - N[y]$ που περιέχονται στην C_x. Αυτό γίνεται σε χρόνο $O(|C_x|)$ για σταθερές κορυφές x και y στο C_x, με αποτέλεσμα ο συνολικός χρόνος να είναι $O(n^3)$ για όλα τα *Compoents*. Θεωρώντας ένα *Interval* $I(x, y)$ και μια κορυφή $s \in I$, ο αλγόριθμος πρέπει να αθροίσει τους αριθμούς ανεξαρτησίας των *Components* C_s του $G - N[s]$ που περιέχονται στο I. Η λειτουργία αυτή απαιτεί χρόνο $O(|I(x, y)|)$ για ένα σταθερό $I(x, y)$ και μια κορυφή $s \in I$. Ο συνολικός χρόνος για τον υπολογισμό του $\alpha(I)$ για όλα τα διαστήματα είναι $O(n^3)$.

Είναι σαφές ότι το βήμα 5 μπορεί να γίνει σε χρόνο $O(n^2)$. Έτσι ο χρόνος εκτέλεσης του αλγορίθμου μας είναι $O(n^4)$.

Παρακάτω παραθέτουμε ένα παράδειγμα εκτέλεσης του αλγορίθμου για το γράφημα 2.1.



Σχήμα 2.1: Ένα AT-free γράφημα με 7 κόμβους

Μετά την εκτέλεση του πρώτου βήματος τα *Components* που υπολογίσαμε φαίνονται στον πίνακα 2.1

Πίνακας 2.1: Components του γραφήματος μετά την εκτέλεση του πρώτου βήματος

Component	**Σύνολο κορυφών**	**Alpha**
C_1^2	{3, 4, 5, 7}	-
C_2^2	{0}	-
C_1^5	{2, 3, 4, 6}	-
C_2^5	{0}	-
C_1^1	{6, 7}	-
C_1^0	{2, 3, 4, 5, 6, 7}	-
C_1^4	{2, 6}	-
C_2^4	{5}	-
C_3^4	{0}	-
C_1^7	{0, 1, 2, 3}	-
C_1^3	{2}	-
C_2^3	{5, 7}	-
C_3^3	{0}	-
C_1^6	{0, 1, 4, 5}	-

Για το δεύτερο βήμα του αλγορίθμου, παίρνοντας όλους τους συνδυασμούς μη γειτονικών κορυφών για το παρών γράφημα. Τα *Intervals* που προκύπτουν φαίνονται στον παρακάτω πίνακα2.2.

Πίνακας 2.2: Intervals του γραφήματος μετά την εκτέλεση του δεύτερου βήματος

Intervals	**Σύνολο κορυφών**	**Alpha**
I(2, 5)	{3, 4}	-
I(2, 7)	{3}	-
I(5, 2)	{3, 4}	-
I(5, 6)	{4}	-
I(0, 7)	{2, 3}	-
I(0, 6)	{4, 5}	-
I(7, 2)	{3}	-
I(7, 0)	{2, 3}	-
I(6, 5)	{4}	-
I(6, 0)	{4, 5}	-

Όπως φαίνεται και στους πίνακες 2.3 και 2.4,μετά το τρίτο βήμα τα $Components$ και $Intervals$ ταξινομούνται με βάση το πλήθος των κορυφών τους, από το μικρότερο στο μεγαλύτερο.

Πίνακας 2.3: Ταξινόμηση Components με βάση το σύνολο κορυφών

Components	**Σύνολο Κορυφών**	**Alpha**
C_2^2	$\{0\}$	-
C_2^5	$\{0\}$	-
C_3^4	$\{0\}$	-
C_3^3	$\{0\}$	-
C_2^4	$\{5\}$	-
C_1^7	$\{0, 1, 2, 3\}$	-
C_1^1	$\{6, 7\}$	-
C_1^4	$\{2, 6\}$	-
C_1^3	$\{2\}$	-
C_1^2	$\{3, 4, 5, 7\}$	-
C_1^5	$\{2, 3, 4, 6\}$	-
C_1^6	$\{0, 1, 4, 5\}$	-
C_1^0	$\{2, 3, 4, 5, 6, 7\}$	-

Πίνακας 2.4: Ταξινόμηση Intervals με βάση το σύνολο κορυφών

Intervals	**Σύνολο Κορυφών**	**Alpha**
I(2, 7)	$\{3\}$	-
I(5, 6)	$\{4\}$	-
I(7, 2)	$\{3\}$	-
I(6, 5)	$\{4\}$	-
I(2, 5)	$\{3, 4\}$	-
I(5, 2)	$\{3, 4\}$	-
I(0, 7)	$\{2, 3\}$	-
I(0, 6)	$\{4, 5\}$	-
I(7, 0)	$\{2, 3\}$	-
I(6, 0)	$\{4, 5\}$	-

Χρησιμοποιώντας τα λήμματα 2.1, 2.2 και εφαρμόζοντας τεχνικές δυναμικού προγραμματισμού, μετά την εκτέλεση του βήματος τέσσερα έχουν υπολογιστεί τα alpha όλων των *Components* και *Intervals*. Τα αποτελέσματα φαίνονται στους πίνακες 2.5 και 2.6

Πίνακας 2.5: Πίνακας Components μετά την εκτέλεση του βήματος 4 του αλγορίθμου

Components	**Σύνολο Κορυφών**	**Alpha**
C_2^2	{0}	1
C_2^5	{0}	1
C_3^4	{0}	1
C_3^3	{0}	1
C_2^4	{5}	1
C_1^7	{0, 1, 2, 3}	3
C_1^1	{6, 7}	1
C_1^4	{2, 6}	1
C_1^3	{2}	1
C_1^2	{3, 4, 5, 7}	2
C_1^5	{2, 3, 4, 6}	2
C_1^6	{0, 1, 4, 5}	3
C_1^0	{2, 3, 4, 5, 6, 7}	3

Πίνακας 2.6: Πίνακας Intervals μετά την εκτέλεση του βήματος 4 του αλγορίθμου

Intervals	**Σύνολο Κορυφών**	**Alpha**
I(2, 7)	{3}	1
I(5, 6)	{4}	1
I(7, 2)	{3}	1
I(6, 5)	{4}	1
I(2, 5)	{3, 4}	1
I(5, 2)	{3, 4}	1
I(0, 7)	{2, 3}	2
I(0, 6)	{4, 5}	2
I(7, 0)	{2, 3}	2
I(6, 0)	{4, 5}	2

2.2 Υπολογισμός Ελάχιστου Κυρίαρχου Συνόλου

2.3 Το Πρόβλημα του 3-Χρωματισμού

ΒΙΒΛΙΟΓΡΑΦΙΑ

[1] A. Brandstädt, V.D. Chepoi, and F.F. Dragan. The algorithmic use of hypertree structure and maximum neighbourhood orderings. *Discrete Applied Mathematics*, 82(1-3):43–77, 1998.

[2] Kloks T. Kratsch D. Broersma, H. and H. Müller. Independent sets in asteroidal triple-free graphs. *SIAM Journal on Discrete Mathematics*, 12(2):276–287, 1999.

[3] D.G. Corneil and Y. Perl. Clustering and domination in perfect graphs. *Discrete Appl. Math.*, 9:27–39, 1984.

[4] D.G. Corneil and L. Stewart. Dominating sets in perfect graphs. *Discrete Math.*, 86:145–164, 1990.

[5] D.P. Dailey. Uniqueness of colorability and colorability of planar 4-regular graphs are np-complete. *Discrete Math.*, 30:289–293, 1980.

[6] Johnson D.S. The np-completeness column: an ongoing guide. *J. Algorithms*, 5:147–160, 1984.

[7] E.S. Elmallah and L.K. Stewart. Independence and domination in polygon graphs. *Discrete Appl. Math.*, 44:65–77, 1993.

[8] M. Farber and M. Keil. Domination in permutation graphs. *J. Algorithms*, 6:309–321, 1985.

[9] Johnson D. Stockmeyer L. Garey, M.R. Some simplified np-complete graph problems. *Theor. Comput. Sci.*, 1:237–267, 1976.

[10] Johnson D.S. Garey, M.R. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, New York, 1979.

[11] Chang G.J. Labelling algorithms for domination problems in sunfree chordal graphs. *Discrete Appl. Math.*, 22:21–34, 1988.

[12] M.C. Golumbic. *Algorithmic Graph Theory and Perfect Graphs*. North-Holland, Amsterdam, 2nd edn edition, 2004.

[13] Lovász L. Schrijver A. Grötschel, M. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica*, 1:169–197, 1981.

[14] D.G. Kirkpatrick H. Breu. Algorithms for dominating and steiner set problems in cocomparability graphs. manuscript, 1993.

[15] I. Holyer. The np-completeness of edge-coloring. *SIAM J. Comput.*, 10:718–720, 1981.

[16] 3-Colouring AT-Free Graphs in Polynomial Time. Juraj stacho. *Algorithmica*, 64(3):384–399, 2012.

[17] D. Kratsch and L. Stewart. Domination on cocomparability graphs. *SIAM J. Discrete Math.*, 6:400–417, 1993.

[18] Dieter Kratsch. Domination and total domination on asteroidal triple-free graphs. *Discrete Applied Mathematics*, 99(1-3):111–123, 2000.

[19] C. G. Lekkerkerker and J. Ch. Boland. Representation of a finite graph by a set of intervals on the real line. *Fund. Math*, 51(1):45–64, 1964.

[20] Farber M. Domination, independent domination, and duality in strongly chordal graphs. *Discrete Appl. Math.*, 7:115–130, 1984.

[21] Chang M.-S. Efficient algorithms for the domination problems on interval and circular-arc graphs. *SIAM Journal on Computing*, 27(6):1671–1694, 1998.

[22] Sau-I. Mertzios, G.B. and S. Zaks. The recognition of tolerance and bounded tolerance graphs. *SIAM Journal on Computing*, 40(5):1234–1257, 2011.

[23] Chang M.S. Efficient algorithms for the domination problems on interval and circular-arc graphs. *SIAM J. Comput.*, 27:1671–1694, 1998.

[24] Liang-Y.D. Dhall S.K. Rhee, C. and S. Lakshmivarahan. An o(n + m)-time algorithm for finding a minimum-weight dominating set in a permutation graph. *SIAM J. Comput.*, 25:404–419, 1996.

[25] Arnborg S. Efficient algorithms for combinatorial problems on graphs with bounded decomposability — a survey. *Bit Numerical Mathematics*, 25(1):1–23, 1985.