ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών
Εργαστήριο Λογικής και Επιστήμης Υπολογισμών
(CORELAB)

Αλγεβρική μεθοδολογία για το χαρακτηρισμό της πολυπλοκότητας Προβλημάτων Ικανοποίησης Περιορισμών

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Παναγιώτη Βούλγαρη

Επιβλέπων: Στάθης Ζάχος
Καθηγητής

Παναγιώτης Βούλγαρης

Αθήνα, 2006

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ
CORELAB

Αλγεβρική μεθοδολογία για το χαρακτηρισμό της πολυπλοκότητας Προβλημάτων Ικανοποίησης Περιορισμών.

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Παναγιώτη Βούλγαρη

Επιβλέπων: Στάθης Ζάχος
Καθηγητής

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την Δευτέρα 17 Ιουλίου 2006.

....	....	....
Ζάχος Σ. Καθηγητής	Σελλής Τ. Καθηγητής	Αφράτη Φ. Καθηγητής

Αθήνα, 2006.

....
Παναγιώτης Βούλγαρης
Ηλεκτρολόγος Μηχανικός και Μηχανικός Ηλεκτρονικών Υπολογιστών

Περίληψη

Συνδυαστικά προβλήματα συναντάμε σε πολλούς τομείς της πληροφορικής. Γραφοθεωρητικά προβλήματα, προβλήματα ικανοποιησιμότητας, προβλήματα scheduling είναι απλώς μερικά παραδείγματα συνδυαστικών προβλημάτων. Όλα αυτά τα προβλήματα μπορούν να αντιμετωπιστούν ως υποκλάσεις ενός γενικευμένου συνδυαστικού προβλήματος του Constraint Satisfaction Problem. Έτσι ερευνώντας τις ιδιότητες του Constraint Satisfaction Problem κερδίζουμε γνώση για μια πολύ μεγάλη και χρήσιμη κλάση προβλημάτων.

Εδώ θα ασχοληθούμε με την πολυπλοκότητα αυτών των προβλημάτων. Συγκεκριμένα η διπλωματική χωρίζεται σε 4 κεφάλαια. Στο πρώτο θα γνωρίσουμε το πρόβλημα, αλλά και θα δούμε εκφράσεις ήδη γνωστών προβλημάτων στη γλώσσα του CSP ώστε να πειστούμε για τη σημαντικότητά του. Στο δεύτερο κεφάλαιο θα ασχοληθούμε με την πολυπλοκότητα της κλάσης των προβλημάτων που μπορούμε να εκφράσουμε με CSP's. Θα δούμε ότι έχουμε ισχυρές ενδείξεις ότι υπάρχει ένα είδος διχοτόμησης της πολυπλοκότητας σε αυτά τα προβλήματα. Συγκεκριμένα ενώ υπάρχουν προβλήματα που είναι στο P αλλά και προβλήματα NP-complete μάλλον δεν υπάρχουν προβλήματα που είναι στο ενδιάμεσο. Στο τρίτο μέρος που είναι και το κύριο μέρος της διπλωματικής θα δούμε πως μπορούμε να χαρακτηρίσουμε υποκλάσεις CSPs που είναι tractable που λύνονται δηλαδή σε P. Θα φτάσουμε τελικά να βρούμε μια αντιστοίχηση μεταξύ σχέσεων και πράξεων και τελικά να χαρακτηρίζουμε Άλγεβρες που αντιστοιχούν σε κλάσεις προβλημάτων CSP ως προς την πολυπλοκότητά τους, μια ιδέα του 2005 που φαίνεται ότι θα ανοίξει νέους δρόμους στην έρευνα στον τομέα. Τέλος θα κλείσουμε με μια παρουσίαση της απόδειξης του Schaefer που πρακτικά άρχισε την έρευνα στο τομέα οπού χαρακτηρίζει όλα τα CSPs κάτω από τον περιορισμό ότι έχουμε σύμπαν δύο στοιχείων.

Abstract

A very common class of problems in Computer Science are Combinatorial problems. Among others graph, satisfiability, scheduling problems are combinatorial problems. What is interesting is that all these problems can be expressed as subclasses of a generalized combinatorial problem the Constraint satisfaction problem. As a result, by examining the properties of the Constraint Satisfaction Problems we are generating knowledge for a very large class of combinatorial problems.

In this thesis we will be concerned with the complexity of such problems. We will do this in 4 chapters. In the first we will be introduced to the problem, and we will see how we can reformulate known combinatorial problems in CSP problems. In the second chapter we will consider the complexity of the problems expressible as CSPs. We will be introduced to the dichotomy conjecture, which states that every problem in CSP is either in P or NP-complete and we will discuss the results that support the conjecture. In the third part, which is the main part of the thesis, we will present the recent algebraic methods used in the field for characterizing the tractability of these problems. We will find relations, operations and finally algebras that correspond to CSPs and we will characterize them as tractable or not. Using Algebras for this characterization is a quite innovative idea introduced in 2005, and it could give a boost to research. In the final chapter we will present one of the first significant results in the field Schaefer's theorem which characterizes all the CSPs with a two-element universe.

Περιεχόμενα

Κεφάλαιο 1

Εισαγωγικά Ορισμοί

Σε αυτό το κεφάλαιο θα γνωρίσουμε το πρόβλημα το οποίο μελετάμε. Πρώτα θα κάνουμε μια γρήγορη εισαγωγή στα αποτελέσματα μέχρι τώρα. Θα δούμε τον τυπικό ορισμό του CSP, καθώς και ισοδύναμα προβλήματα. Να σημειωθεί ότι όταν λέμε ισοδύναμα δεν εννοούμε την ισοδυναμία με την έννοια της αναγωγής αφού τότε όλα τα NP-complete προβλήματα είναι ισοδύναμα. Μπορούμε να πούμε ότι μελετάμε το CSP μέσω διαφορετικών πρισμάτων, όπως είναι αυτό του προβλήματος της ύπαρξης ομομορφισμού. Τέλος για να υπάρξει καλύτερη κατανόηση της γλώσσας του CSP θα εκφράσουμε γνωστά προβλήματα στην γλώσσα του CSP αλλά και στη γλώσσα των ισοδύναμων μορφών του.

Το θέμα της διπλωματικής είναι ο χαρακτηρισμός της πολυπλοκότητας ενός προβλήματος CSP βασιζόμενοι στις ιδιότητες της Γλώσσας Περιορισμών, δηλαδή των σχέσεων που χρησιμοποιούμε ως περιορισμούς. Ειδικότερα θα ασχοληθούμε πότε μπορεί να λυθεί το πρόβλημα στο P και πότε γίνεται NP-complete.

1.1 Εισαγωγή

Σε ένα CSP ο σκοπός είναι να βρούμε μια ανάθεση τιμών σε ένα δοσμένο σύνολο μεταβλητών, που υπόκειται σε περιορισμούς στο ποιες τιμές μπορούν να ανατεθούν συγχρόνως σε συγκεκριμένα υποσύνολα μεταβλητών [28, 30]. Ένα συχνό παράδειγμα ενός τέτοιου προβλήματος είναι αυτό της ικανοποιησιμότητας προτασιακών τύπων [15], οπού στις μεταβλητές πρέπει να ανατεθούν boolean τιμές και οι περιορισμοί είναι από τις προτάσεις.

Το μαθηματικό υπόβαθρο που χρησιμοποιείται για να περιγραφεί τα CSPs έχει στενούς δεσμούς με πολλές περιοχές της θεωρίας υπολογιστών και των μαθηματικών. Συνδέεται με την σχεσιακή θεωρία βάσεων δεδομένων [16], με λογική και συνολοθεωρία [37], και με universal algebra. Ένα survey αυτών των αποτελεσμάτων είναι το [33].

Σε ότι επακολουθήσει θεωρούμε ότι $P \neq NP$, και καλούμε ένα πρόβλημα tractable αν ανήκει στο P. Το CSP είναι γνωστό ότι είναι NP-hard γενικά [28, 30]. Παρόλα αυτά, το να τέθουν όρια στις μορφές των περιορισμών έχει αποδειχθεί ότι διασφαλίζει την tractability [9, 22, 23, 25, 38].

Ένα σημαντικό ανοιχτό πρόβλημα στην περιοχή είναι να χαρακτηριστούν ακριβώς οι μορφές των σχέσεων περιορισμών οι οποίες δημιουργούν tractable κλάσεις προβλημάτων. Αυτό το πρόβλημα είναι σημαντικό από θεωρητική άποψη, αφού βοηθάει στον καθορισμό του ορίου μεταξύ tractability και intractability σε ένα ευρύ φάσμα συνδυαστικών προβλημάτων εύρεσης (λύσης). Είναι επίσης σημαντικό από πρακτικής άποψης, αφού επιτρέπει τη δημιουργία γλωσσών προγραμματισμού περιορισμών που χρησιμοποιούν τη γνώση των tractable κλάσεων περιορισμών ώστε να δίνουν πιο αποδοτικές τεχνικές επίλυσης [34, 27].

Το πρόβλημα του χαρακτηρισμού των tractable περιπτώσεων λύθηκε εξ ολοκλήρου στην ειδική περίπτωση που οι μεταβλητές παίρνουν Boolean τιμές, από τον Schaefer το 1978 [35]. Ο Schaefer απέδειξε ότι για Boolean CSP (τα οποία ονόμασε "γενικευμένα προβλήματα ικανοποιησιμότητας"), υπάρχουν ακριβώς έξι διαφορετικές οικογένειες tractable περιορισμών, και οποιοδήποτε πρόβλημα που περιέχει περιορισμούς εκτός αυτών των έξι οικογενειών είναι NP-complete. Αυτό το σημαντικό αποτέλεσμα είναι γνωστό ως το θεώρημα διχοτόμησης του Schaefer. Παρόμοια αποτελέσματα διχοτόμησης έχουν αποδειχθεί για άλλα συνδυαστικά προβλήματα σε Boolean πεδία τιμών που σχετίζονται με το παραπάνω.

Ο Schaefer [35] έθεσε το ερώτημα του εάν η ανάλυση της πολυπλοκότητας μπορούσε να επεκταθεί σε μεγαλύτερα σύνολα (δηλαδή παραπάνω από μόνο δύο τιμές). Κάποια πρόοδος σημειώθηκε σε αυτό το δρόμο πρόσφατα, και ένας αριθμός από tractable οικογένειες περιορισμών αναγνωρίστηκε, ορισμένων τόσο σε πεπερασμένα όσο και άπειρα σύνολα. Συγκεκριμένα, οι Feder και Vardi [37] χρησιμοποίησαν τεχνικές από λογικό προγραμματισμό και θεωρία συνόλων ώστε να αναγνωρίσουν τρεις διευρυμένες οικογένειες από tractable περιορισμούς, που είναι υπερσύνολα των έξι οικογενειών που όρισε ο Schaefer. Μια σειρά paper του Jeavons και άλλων συγγραφέων απέδειξε ότι οποιαδήποτε tractable κλάση περιορισμών ορισμένων σε πεπερασμένα σύνολα μπορεί να χαρακτηριστεί χρησιμοποιώντας αλγεβρικές ιδιότητες σχέσεων [20, 24, 22, 23]. Αυτή η μεθοδολογία χρησιμοποιήθηκε αργότερα από τον Bulatov ώστε να μπορέσει να πετύχει πλήρη χαρακτηρισμό της πολυπλοκότητας περιορισμών ορισμένων σε σύνολα τριών στοιχείων [3]. Τέλος, σε temporal και spatial λογικής προβλήματα που περιέχουν περιορισμούς σε απειροσύνολα, αρκετές tractable κλάσεις έχουν βρεθεί [31], μάλιστα έχει προκύψει ένα θεώρημα διχοτόμησης για temporal reasoning προβλήματα ορισμένα σε διαστήματα που εκφράζονται με την άλγεβρα διαστημάτων του Allen. Έχει επίσης δειχθεί πρόσφατα ότι η πολυπλοκότητα συγκεκριμένων μορφών περιορισμών σε πεπερασμένα σύνολα τιμών μπορούν να αναλυθούν χρησιμοποιώντας αλγεβρικές ιδιότητες.[5]

Παρόλα αυτά δεν υπάρχει ακόμα πλήρης χαρακτηρισμός της πολυπλοκότητας των περιορισμών σε πεπερασμένα σύνολα με παραπάνω από τρία στοιχεία, ούτε για τυχαία πεπερασμένα σύνολα.

Πρέπει να παρατηρήσουμε ότι τα πιο σημαντικά αποτελέσματα στον τομέα έγιναν όταν χρησιμοποιήθηκαν δυνατότερα και πιο γενικευμένα μαθηματικά εργαλεία. Έτσι στο θεώρημα διχοτόμησης του Schaefer μιλάμε για συντακτικές ιδιότητες των προτασιακών τύπων που σίγουρα δεν μπορεί να χρησιμοποιηθεί σε μη-Boolean περιορισμούς. Για να βρεθεί μια πιο ικανοποιητική γλώσσα και αντίστοιχα μαθηματικά εργαλεία έχουν γίνει διάφορες προσπάθειες [11, 12, 37, 20, 22], η διπλωματική θα ασχοληθεί περισσότερο με τις προσπάθειες που προσανατολίζονται στη χρήση των αλγεβρικών ιδιοτήτων των περιορισμών [20, 22] με κορύφωση το [2] που φτάνει να χαρακτηρίζει αλγεβρικές δομές ως προς tractability.

Το πρώτο βήμα σε αυτή την αλγεβρική θεώρηση χρησιμοποιεί την γνωστή ιδέα ότι δοσμένου ενός αρχικού συνόλου από περιορισμούς - σχέσεις, θα υπάρχουν περισσότερες σχέσεις οι οποίες θα μπορούν να προστεθούν στο σύνολο χωρίς να αλλάξει η πολυπλοκότητα του προβλήματος. Έχει αποδειχθεί ότι μπορούμε να προσθέσουμε όλες τις σχέσεις που μπορούμε να παράγουμε από τις αρχικές χρησιμοποιώντας απλούς κανόνες. Τα μεγαλύτερα σύνολα σχέσεων που μπορούμε να πάρουμε χρησιμοποιώντας αυτούς τους κανόνες ονομάζονται relational clones [13]. Άρα καταλήγουμε ότι είναι αρκετό να αναλύσουμε την πολυπλοκότητα μόνο των συνόλων σχέσεων που είναι relational clones.

Το επόμενο βήμα στην αλγεβρική αντιμετώπιση είναι η παρατήρηση ότι τα relational clones μπορούν να χαρακτηριστούν πλήρως από τους πολυμορφισμούς τους, οι οποίοι είναι αλγεβρικές πράξεις στο ίδιο σύνολο με το οποίο είναι ορισμένες οι σχέσεις [22, 19]. Έτσι όχι μόνο μπορούμε να περιγράφουμε οικογένειες σχέσεων που αλλιώς θα ήταν δύσκολο να περιγράφουν, αλλά μέσω των πράξεων μπορούμε να βρούμε ιδιότητες των σχέσεων που μπορούν να χρησιμοποιηθούν για αποδοτικούς αλγόριθμους. Αυτή η σχέση μεταξύ relational clone και πολυμορφισμών έχει ήδη παίξει ένα σημαντικό ρόλο στην αναγνώριση πολλών tractable κλάσεων και στη δημιουργία κατάλληλων αποδοτικών αλγοριθμικών λύσεων για αυτές [3, 1, 6, 4, 10, 20].

Όμως όπως θα δούμε μπορούμε να πάμε ένα βήμα παρακάτω συσχετίζοντας άλγεβρες και όχι πολυμορφισμούς με το σύνολο των περιορισμών. Έτσι μπορούμε να χρησιμοποιήσουμε θεμελιώδη θεωρήματα για finite άλγεβρες [18, 29, 36], και έτσι να φτάσουμε στα αποτελέσματα του [2].

1.2 Βασικοί Ορισμοί

1.2.1 Σχέσεις και Συναρτήσεις

Παρακάτω θα δοθούν οι σημαντικότεροι ορισμοί που θα χρησιμοποιηθούν μετέπειτα: (ειδικά οι ορισμοί κλάσεων συναρτήσεων είναι εδώ κυρίως για αναφορά και δεν είναι αναγκαίοι μετέπειτα εκτός συγκεκριμένων αποτελεσμάτων)

ΟΡΙΣΜΟΣ 1.2.1.1

Σχέση: Για οποιοδήποτε σύνολο A, και φυσικό αριθμό n, το σύνολο όλων των n-άδων στοιχείων του A συμβολίζεται με A^n. Κάθε υποσύνολο του A^n ονομάζεται σχέση n ορισμάτων πάνω στο A. Το σύνολο όλων των πεπερασμένων σχέσεων στο A συμβολίζεται R_A. □

ΟΡΙΣΜΟΣ 1.2.1.2

Συνάρτηση: Μια συνάρτηση f από ένα σύνολο A σε σύνολο B, συμβολίζεται με $f : A \to B$, και είναι ένα υποσύνολο του $A \times B$ τέτοιο ώστε για κάθε $a \in A$ υπάρχει ακριβώς ένα $b \in B$ τέτοιο ώστε το $(a, b) \in f$. Αν $(a, b) \in f$ γράφουμε $f(a) = b$. Με $=_A$ θα συμβολίζουμε την ισότητα στο A ενώ με $\neq_A$ το διάφορο. □

ΟΡΙΣΜΟΣ 1.2.1.3

Συνάρτηση ν-ορισμάτων: Μια συνάρτηση ν-ορισμάτων σε ένα σύνολο A είναι μια συνάρτηση $f : A^n \to A$. Το σύνολο όλων των συναρτήσεων στο A καλείται O_A. Αν $((a_1, a_2, \ldots, a_n), a_{n+1}) \in f$ γράφουμε $f(a_1, a_2, \ldots, a_n) = a_{n+1}$. □

ΟΡΙΣΜΟΣ 1.2.1.4

Χαρακτηρισμοί συναρτήσεων: Έστω μια συνάρτηση f ν-ορισμάτων από το $A_n \to A$.

- Η f είναι injective αν $f(a) = f(a') \Leftrightarrow a = a'$. Αν n=1 τότε η f καλείται αναδιάταξη.

- Αν $f(a, a, \ldots, a) = a\ \forall a \in A$ τότε η f λέγεται idempotent.

- Αν υπάρχει κάποιος δείκτης $i \in \{1, 2, \ldots, n\}$ τέτοιος ώστε $\forall(a_1,\ a_2,\ \ldots, a_n) \in A^n$ έχουμε $f(a_1, a_2,\ \ldots, a_n) = g(a_i)$, με $g : A \to A$ μη σταθερή, η f καλείται 'essentially unary'. Κάθε συνάρτηση που δεν ανήκει σε αυτές θα λέγεται essentially nonunary, συμπεριλαμβανομένων των σταθερών.

- Αν $n = 2$ και $\forall a_1, a_2, a_3 \in A$ έχουμε $f(f(a_1, a_2), a_3) = f(a_1, f(a_2, a_3))$ (Προσαιτεριστικότητα Associativity) και $f(a_1, a_2) = f(a_2, a_1)$ (Αντιμεταθετικότητα Commutativity), τότε η f λέγεται AC συνάρτηση.

- Αν $n \geq 3$ και υπάρχει κάποιος δείκτης $i \in \{1, 2, \ldots, n\}$ τέτοιος ώστε $\forall a_1,\ a_2 \ldots, a_n \in A$ τέτοια ώστε $|\{a_1, a_2, \ldots, a_n\}| < n$ έχουμε $f(a_1, a_2,\ \ldots, a_n) = a_i$, αλλά η f δεν είναι προβολή, τότε η f λέγεται ημιπροβολή.

- Αν $n = 3$ και $\forall a, a' \in A$ έχουμε $f(a, a, a') = f(a, a', a) = f(a', a, a) = a$, τότε η f λέγεται συνάρτηση πλειονότητας. Η συνάρτηση πλειονότητας στο A:
$$f(x, y, z) = \begin{cases} y \ if \ y = z \\ x \ otherwise \end{cases}$$
θα καλείται 'dual discriminator' στο A και θα συμβολίζεται με μ_A.

- Αν $n = 3$ και $\forall a_1, a_2, a_3 \in A$ έχουμε $f(a_1, a_2, a_3) = a_1 + a_2 + a_3$, όπου $+$ είναι μια συνάρτηση με $n = 2$ στο A ώστε $(A, +)$ να είναι μια Αβελιανή 2-ομάδα, τότε η f ονομάζεται "generalised parity operation".

□

1.2.2 Constraint Satisfaction Problem

Πλέον μπορούμε να ορίσουμε το CSP.

ΟΡΙΣΜΟΣ 1.2.2.1

Γλώσσα περιορισμών: Γλώσσα περιορισμών σε ένα σύνολο A είναι ένα σύνολο σχέσεων στο σύνολο A. $\Gamma = \{\rho_1, \rho_2, \ldots, \rho_n\}$. □

ΟΡΙΣΜΟΣ 1.2.2.2

Πρόβλημα ικανοποίησης περιορισμών : Για κάθε σύνολο A και κάθε γλώσσα περιορισμών Γ στο A, το Πρόβλημα ικανοποίησης περιορισμών $CSP(\Gamma)$ είναι το συνδυαστικό πρόβλημα απόφασης με
Είσοδο: (V, A, C) όπου

- V είναι ένα σύνολο μεταβλητών
- C είναι ένα σύνολο περιορισμών $C_1, \ldots, C_q$.
  Κάθε περιορισμός $C_i \in C$ είναι ένα ζευγάρι (s_i, ρ_i) όπου
  - s_i είναι μια πλειάδα μεταβλητών μεγέθους m_i, που ονομάζεται το πεδίο του περιορισμού.
  - $\rho_i \in \Gamma$ είναι μια σχέση m_i μεταβλητών πάνω στο A, που ονομάζεται σχέση περιορισμού.

Έξοδο: ΝΑΙ, αν υπάρχει λύση, δηλαδή μια συνάρτηση $f(x)$, από το V στο A, τέτοια ώστε, για κάθε περιορισμό $(s_i, \rho_i) \in C$, με $s_i = (x_{i1}, \ldots, x_{im})$, η πλειάδα $(f(x_{i1}), \ldots, f(x_{im}))$ ανήκει στο ρ_i. ΟΧΙ, αλλιώς. □

1.2.3 Πρόβλημα Ομομορφισμών

ΟΡΙΣΜΟΣ 1.2.3.1

Relational structure: Μια “relational structure” είναι ένα ζευγάρι, $(V, E_i\ (i \in I))$, που αποτελείται από ένα μη κενό σύνολο V, και ένα σύστημα, E_i, από πεπερασμένες σχέσεις στο V, που δεικτοδοτούνται από τα στοιχεία του I. Το σύνολο V καλείται το ‘σύμπαν’ της relational structure.

Μια relational structure $S = (V, E_i\ (i \in I))$ καλείται πεπερασμένη αν το V και το i είναι πεπερασμένα. Σε αυτή την περίπτωση θα γράφουμε το S ως $(V, E_1, E_2, \dots, E_{|I|})$ □

ΟΡΙΣΜΟΣ 1.2.3.2

rank function: Η ‘rank function’ μιας relational structure $(V, E_i(i \in I))$, είναι μια συνάρτηση ρ από το I στο σύνολο των μη αρνητικών ακεραίων, έτσι ώστε $\forall i \in I, \rho(i)$ είναι η πολλαπλότητα της E_i.

Μια relational structure S είναι παρόμοια με μια άλλη αν έχουν την ίδια rank function. □

ΟΡΙΣΜΟΣ 1.2.3.3

Ομομορφισμοί: Έστω $S = (V, E_i(i \in I))$ και $S' = (V', E_i{}'\ (i \in I))$ δύο παρόμοιες relational structures, και έστω ρ η κοινή τους rank function.

Ένας ομομορφισμός από την S στην S' είναι μια συνάρτηση $h : V \to V'$ τέτοια ώστε, $\forall i \in I$,

$$(v_1, v_2, \dots, v_{\rho(i)}) \in E_i \Rightarrow (h(v_1), h(v_2), \dots, h(v_{\rho(i)})) \in E_i'.$$

Το σύνολο όλων των ομομορφισμών από το σύνολο S στο S' καλείται $Hom\ (S, S')$. □

Τώρα μπορούμε να ορίσουμε ένα ισοδύναμο με το CSP πρόβλημα αυτό της εύρεσης ομομορφισμού μεταξύ δύο relational structures. Επίσης καλείται και “general combinatorial problem” (GCP).

ΟΡΙΣΜΟΣ 1.2.3.4

Πρόβλημα ομομορφισμού: Το πρόβλημα ομομορφισμού είναι η κλάση των προβλημάτων απόφασης με:
Είσοδο: Ένα ζευγάρι από παρόμοιες relational structures, (S_1, S_2).
Έξοδο: Αποδέχεται αν υπάρχει ένας ομομορφισμός μεταξύ των δύο, αλλιώς δεν αποδέχεται. □

1.3 Παραδείγματα

1.3.1 Γνωστά προβλήματα στη γλώσσα του CSP

Σ' αυτό το τμήμα θα δείξουμε την ισοδυναμία των παραπάνω προβλημάτων και θα εκφράσουμε γνωστά μας προβλήματα στη γλώσσα των προβλημάτων. Στα προβλήματα γράφων θα συμβολίζουμε με Δ τους κόμβους

$\{\delta_1, \delta_2, \ldots \delta_{|\Delta|}\}$ και E τις πλευρές $\{e_1, e_2, \ldots, e_{|E|}\}$. Ενώ V θα είναι το σύνολο των μεταβλητών.

ΠΑΡΑΔΕΙΓΜΑ 1.3.1.1

Χρωματισμός Γράφων: Η είσοδος του προβλήματος χρωματισμών γράφων είναι ένας γράφος G και ένας ακέραιος q. Το ερώτημα είναι εάν οι κόμβοι του G μπορούν να δεικτοδοτηθούν με q χρώματα έτσι ώστε οι γειτονικές κορυφές να έχουν διαφορετικά χρώματα.

CSP: $A = \{1, \ldots, q\}$ τα χρώματα $V = \{v_1, v_2, \ldots, v_{|\Delta|}\}$ (μια μεταβλητή για κάθε κόμβο), $\Gamma = \{\neq_A\}$ και $C = \{C_1, \ldots, C_{|E|}\}$ με $C_i = (e_i, \neq_A)$ για κάθε $e_i \in E$.

GCP: Εύρεση ομομορφισμού μεταξύ των (G, K_q), με K_q τον πλήρη γράφο με q κόμβους. □

ΠΑΡΑΔΕΙΓΜΑ 1.3.1.2

Εύρεση Κλίκας: Η είσοδος του προβλήματος εύρεσης κλίκας είναι ένας γράφος G και ένας ακέραιος q. Το ερώτημα είναι εάν υπάρχει κλίκα μεγέθους q.

CSP: $A = \{\delta_1, \delta_2, \ldots, \delta_{|\Delta|}\}$ όλοι οι κόμβοι του G, $V = \{v_1, v_2, \ldots v_k\}$ οι επιλεγμένοι κόμβοι για την κλίκα , $\Gamma = \{E\}$ οι κόμβοι και $C = \{C_{1,1}, \ldots, C_{|K|,|K|}\}$ με $C_{i,j} =$ αν $i \neq j$ $((v_i, v_j), E)$ αλλιώς $((v_i, v_j), \neq_A)$.

GCP: Εύρεση ομομορφισμού μεταξύ των (K_q, G). □

ΠΑΡΑΔΕΙΓΜΑ 1.3.1.3

Vertex Cover: Η είσοδος εδώ είναι ένας γράφος G και ακέραιος q. Το ερώτημα είναι αν υπάρχει $V' \subseteq V$ με $|V'| \leq k$, έτσι ώστε για κάθε $(v, w) \in E$ $v \in V'$ ή $w \in V'$.

CSP: $A = \{\delta_1, \delta_2, \ldots, \delta_{|\Delta|}\}$, $V = \{v_1, v_2, \ldots, v_{|\Delta|-q}\}$, $\Gamma = \{\overline{E}\}$, $C = \{\ldots C_{i,j} \ldots\} i, j \leq |\Delta| - q$, με $C_{i,j} = ((v_i, v_j), \overline{E})$.

GCP: Εύρεση ομομορφισμού μεταξύ των $(K_{(|\Delta|-q)}, \overline{G})$, οπού G ο συμπληρωματικός γράφος $(V, \neq_V - E)$. □

ΠΑΡΑΔΕΙΓΜΑ 1.3.1.4

k-Dimesnional matching: Η είσοδος εδώ είναι ένας μια σχέση M πολλαπλότητας k σε ένα σύνολο V. Το ερώτημα είναι αν υπάρχει $M' \subseteq M$ με $|M'| = |V|$, έτσι ώστε οποιαδήποτε $t_1, t_2 \in M'$ δεν συμφωνούν σε καμία συντεταγμένη.

Ορίζουμε εκ των προτέρων την $\neq'_M = M^2 - (t_i, t_j)$ όταν τα (t_i, t_j) συμφωνούν σε κάποια συντεταγμένη.

CSP: $A = M$, $V = \{v_1, v_2, \ldots, v_k\}$, $\Gamma = \{\neq'_M\}$, $C = \{\ldots C_{i,j} \ldots\} i, j \leq k,\ i \neq j$, με $C_{i,j} = ((v_i, v_j), \neq'_M)$.

GCP: Εύρεση ομομορφισμού μεταξύ των $((V, \neq_V), (M, \neq'_M))$. □

ΠΑΡΑΔΕΙΓΜΑ 1.3.1.5

Hamiltonian Circuit: Η είσοδος είναι ένας γράφος G και ζητείται αν υπάρχει κύκλος ο οποίος να περνάει από κάθε κόμβο μια φορά.

Ορίζουμε C_V μια κυκλική αναδιάταξή του V

CSP: $A =$ οι κόμβοι, $V = \{v_1, v_2, \ldots, v_{|\Delta|}\}$ η σειρά με την οποία περνάμε τους κόμβους. $\Gamma = \{E, \neq_V\}$ και $C = \{C_1, \ldots, C_{|\Delta|}\}$ με $C_i = ((v_i, v_{(i+1)\ mod|\Delta|}), E)$, επίσης βάζουμε constraints για να μην αντιστοιχούν δύο κόμβοι σε ίδια σειρά. $(C_{\neq i,j} = ((v_i, v_j), \neq_V))$.

GCP: Εύρεση ομομορφισμού μεταξύ των $((V, C_V, \neq_v), (V, E, \neq_V))$. □

ΠΑΡΑΔΕΙΓΜΑ 1.3.1.6

Bandwidth: Η είσοδος είναι ένας γράφος $G = (\Delta, E)$ με $\Delta = \{\ \delta_1, \delta_2, \ldots, \delta_{|\Delta|}\}$, και ένας θετικός ακέραιος q. Το ερώτημα είναι εάν υπάρχει μια γραμμική διάταξη του V τέτοια ώστε οι γειτονικοί κόμβοι του γράφου να απέχουν το πολύ q στη διάταξη.

Ορίζουμε $B_q = \{(v_i, v_j) \in V^2 || i - j| \leq q\}$.

CSP: $A =$ οι κόμβοι, $V = \{v_1, v_2, \ldots, v_{|\Delta|}\}$ η σειρά με την οποία τους διαλέγουμε. $\Gamma = \{E, \neq_V\}$ και $C = \{C_1, \ldots, C_{|E|}\}$ με $C_i = (e_i, B_q)$, επίσης βάζουμε constraints για να μην αντιστοιχούν δύο κόμβοι σε ίδια σειρά.

GCP: Εύρεση ομομορφισμού μεταξύ $((\Delta, E, \neq_V), (V, B_q, \neq_V))$. □

ΠΑΡΑΔΕΙΓΜΑ 1.3.1.7

H-colorability: Η είσοδος είναι δύο γράφοι $G = (V, E)$ και $G' = (\Delta, E')$. Το ερώτημα είναι αν υπάρχει αντιστοίχηση των κόμβων ώστε γειτονικές κορυφές του G να γίνονται γειτονικές στον G'.

CSP: $A =$ οι κόμβοι του G', $V =$ οι κόμβοι του G'. $\Gamma = \{E'\}$. $C = \{\ldots, C_i, \ldots\}$ $C_i = ((e_i), E')$, $\forall e_i \in E$.

GCP: Ομομορφισμός μεταξύ $((V, E), (\Delta, E'))$. □

ΠΑΡΑΔΕΙΓΜΑ 1.3.1.8

Graph Isomorphism: Η είσοδος είναι δύο γράφοι $G = (\Delta, E)$ και $G' = (\Delta', E')$ με $|V| = |\Delta|$. Το ερώτημα είναι αν υπάρχει αντιστοίχηση των κόμβων ώστε γειτονικές κορυφές του G να γίνονται γειτονικές στον G' και αντίστοιχα για τις μη γειτονικές.

CSP: $A =$ οι κόμβοι του G', $V =$ οι κόμβοι του G. $\Gamma = \{E', \overline{E'}\}$. $C = \{\ldots, C_{i,j}, \ldots\}$ $C_{i,j} =$ εάν $(v_i, v_j) \in E$ $((v_i, v_j), E')$, αλλιώς $((v_i, v_j), \overline{E'})$.

GCP: Ομομορφισμός μεταξύ $((\Delta, E, \overline{E}), (\Delta', E', \overline{E'}))$. □

ΠΑΡΑΔΕΙΓΜΑ 1.3.1.9

Undirected Graph Reachability: Η είσοδος είναι ένας γράφος $G = (\Delta, E)$ και δύο κόμβοι $v, w \in \Delta$. Το ερώτημα είναι αν υπάρχει ένα μονοπάτι στον G που συνδέει τους δύο κόμβους.

Θα εκφράσουμε το συμπληρωματικό πρόβλημα:

CSP: $A = \{0, 1\}$, $V = \varDelta$, $\varGamma = \{=_\varDelta, \neq_\varDelta\}$. $C = \{\ldots, C_{0,i} \ldots, C_1\}$, με $C_i = (e_i, =_\varDelta)$ και $C_1 = ((v, w), \neq_\varDelta)$.

GCP: Ομομορφισμός μεταξύ $(\varDelta, E, \{(v, w)\}), (\{0, 1\}, =_{\{0,1\}}, \neq_{\{0,1\}}))$. □

ΠΑΡΑΔΕΙΓΜΑ 1.3.1.10

Satisfiability: Η είσοδος είναι ένας τύπος, F, σε προτασιακή λογική, δηλαδή ένα conjucntion από προτάσεις, P. Κάθε πρόταση στο P είναι μια disjunction από literals, οπού κάθε literal είναι είτε μια μεταβλητή είτε η άρνησή της. Το ερώτημα είναι αν υπάρχει ανάθεση αληθοτιμών στις μεταβλητές του F τέτοιες ώστε η F να γίνεται αληθής.

Ορίζουμε V το σύνολο των προτασιακών μεταβλητών της F, $E_p = \{(x_1, \ldots, x_{\rho(p)}\}$, με $x_1, \ldots, x_{\rho(p)}$ τις μεταβλητές που εμφανίζονται στην πρόταση p, και R_p οι αντίστοιχες πλειάδες που ικανοποιούν την σχέση p.

CSP: $A = \{0, 1\}$,$V =$ το σύνολο των μεταβλητών, $\varGamma = \{\ldots R_i \ldots\}$ για κάθε πρόταση i, και $C = \{\ldots, C_i, \ldots\}$ με $C_i = ((x_1, x_2, \ldots, x_{\rho(i)}), R_i)$ για κάθε πρόταση i.

GCP: Ομομορφισμός μεταξύ των $((V, E_p(p \in P)), (\{0, 1\}, R_p(p \in P)))$.
□

1.3.2 Αναγωγή CSP ↔ GCP

Μετά από όλα αυτά τα παραδείγματα είμαστε έτοιμοι να δείξουμε και την προφανή πλέον αναγωγή από το ένα πρόβλημα στο άλλο και να πεισθούμε για την ισοδυναμία τους.

ΠΑΡΑΔΕΙΓΜΑ 1.3.2.1

CSP εκφρασμένο ως GCP: Έστω CSP με A,V,$C = \{C_i\}$ με $C_i = (t_i, R_i)$ με το t_i να είναι πλειάδα μεταβλητών και το R_i μια σχέση $\in \varGamma$. Το αντίστοιχο GCP είναι το ερώτημα αν υπάρχει ομομορφισμός μεταξύ των $((V, \{t_1\}, \{t_2\}, \ldots, \{t_{|C|}\}),(A, R_1, R_2, \ldots, R_{|C|}))$. □

ΠΑΡΑΔΕΙΓΜΑ 1.3.2.2

GCP εκφρασμένο ως CSP: Έστω το GCP το ερώτημα εύρεσης ομομορφισμού μεταξύ των δομών $(A, D_1, D_2, \ldots, D_k)$ και $(V, R_1, R_2, \ldots, R_k)$. Το αντίστοιχο CSP έχει $A = A$, $V = V$, και για το *ιστό* στοιχείο της D_j έχει το $C_{\{i,j\}} = (t_i, R_j)$ με t_i την *ιστή* πλειάδα του D_j, και αυτό για κάθε i, j.
□

1.4 Συμπεράσματα

Μέχρι τώρα γνωρίσαμε το πρόβλημά μας και τον "σκοπό" μας, δηλαδή να μπορέσουμε να χαρακτηρίσουμε το εκάστοτε πρόβλημα CSP αν είναι στο

P ή NP-complete με βάση τις ιδιότητες των σχέσεων που ορίζουν τους περιορισμούς. Επίσης είδαμε ένα ισοδύναμο πρόβλημα το GCP ή πρόβλημα ομομορφισμού που σε μερικές περιπτώσεις δίνει πιο μικρές εκφράσεις και σίγουρα πιο όμορφες από μαθηματική άποψη.

Μέχρι τώρα μπορούμε να κάνουμε ένα εύκολο και προφανή διαχωρισμό με βάση τη γραφή του GCP. Έστω $GCP(A, B)$ να είναι το πρόβλημα αν υπάρχει ομομορφισμός από την αλγεβρική δομή A στην B. Έστω ότι θέλουμε να χαρακτηρίσουμε το $GCP(A, all)$ δηλαδή δεδομένης μιας σταθερής αλγεβρικής δομής A να δούμε αν υπάρχει ομομορφισμός σε οποιαδήποτε άλλη. Αυτή η κλάση προβλημάτων όσο γενική και αν φαίνεται είναι στο P. Συγκεκριμένα αν σκεφτούμε το ανάλογο CSP πρόβλημα η αλγεβρική δομή A μας δίνει πόσες μεταβλητές έχουμε έστω c (σταθερά) ενώ η B τις τιμές που μπορούν να πάρουν n. Προφανώς οι διαφορετικές περιπτώσεις τιμών των μεταβλητών είναι n^c και άρα το κόστος του brute force είναι ένα πολυώνυμο ως προς n, άρα σίγουρα λύνεται στο P.

Το βασικό ερώτημα είναι τί γίνεται στην περίπτωση $GCP(All, B)$, δηλαδή με σταθερή τη δομή B να χαρακτηρίσουμε το πρόβλημα όποια και να είναι η δομή A. (Αυτό το πρόβλημα το καλούμε και $CSP(B)$. Αν τα στοιχεία του B είναι c και τα στοιχεία του A n βλέπουμε ότι το κόστος της brute force είναι c^n που είναι εκθετικό, άρα πρέπει να βρούμε πιο έξυπνους αλγορίθμους αν γίνεται όντως κάποιες από αυτές τις περιπτώσεις να λυθούν στο P.

Ας σκεφτούμε τί είναι τελικά η αλγεβρική δομή B. Είναι κάποιες τιμές που μπορούν να πάρουν οι μεταβλητές μας και κάποιες σχέσεις. Τελικά αυτό που μπορούμε να εκμεταλλευτούμε είναι ιδιότητες των σχέσεων που τις κάνουν "πιο εύκολα υπολογιστές". Δηλαδή να βρούμε ιδιότητες που αν εκπληρώνουν οι σχέσεις του B μπορούμε να κάνουμε κάτι πιο έξυπνο από brute force αλγορίθμους. Αυτό είναι τελικά το θέμα που θα μας απασχολήσει.

Κεφάλαιο 2

Complexity of CSP

2.1 SNP

2.1.1 Εισαγωγή

Η Κλάση SNP [26] περιέχει όλα τα προβλήματα που μπορούν να εκφραστούν από μια υπαρξιακή δεύτερης τάξης πρόταση με ένα καθολικό πρώτο μέρος, συγκεκριμένα μία πρόταση της μορφής $(\exists S')(\forall x)\Phi(x,S,S')$, οπού Φ είναι μια πρώτης τάξης πρόταση χωρίς ποσοδείκτες. Δηλαδή η Φ είναι μια πρόταση που χρησιμοποιεί σχέσεις από τα σύνολα S, S' που εφαρμόζονται στις μεταβλητές x, και τα $\neg$,$\vee$, $\wedge$. Το πρόβλημα είναι να αποφασίσουμε αν για κάποια δομή S που έχουμε ως είσοδο, υπάρχει μια δομή S' ορισμένη στο ίδιο πεδίο έτσι ώστε για όλες τις απονομές τιμών στις μεταβλητές του x να είναι αληθές ότι ισχύει $\Phi(x,S,S')$. Θα αναφερόμαστε στις σχέσεις του S ως σχέσεις εισόδου, ενώ στις σχέσεις του S' ως υπαρξιακές σχέσεις. Η τυποποίηση του SNP που θα χρησιμοποιήσουμε παρακάτω (και είναι ισοδύναμη) θα είναι $\exists S'\forall x \wedge_i \neg(a_i \wedge b_i \wedge c_i)$ με $a_i \in S$ και $b_i \in S'$ και πιθανά c_i σχέσεις ισότητας, με η χωρίς $\neg$.

ΠΑΡΑΔΕΙΓΜΑ 2.1.1.1

Μή ύπαρξη κύκλου μεγέθους 3 ως SNP: Η έκφραση θα ήταν $\forall x \forall y \forall z$ $\neg(E(x,y) \wedge E(y,z) \wedge E(z,x))$, (με τις μεταβλητές να αντιστοιχούν σε κόμβους) βλέπουμε οτί δεν χρειάζεται υπαρξιακή δομή αλλά αρκεί μια δομή εισόδου με μόνη σχέση τη σχέση γειτνίασης, δηλαδή τις ακμές. □

ΠΑΡΑΔΕΙΓΜΑ 2.1.1.2

3-colourability: Εδώ το παράδειγμα γίνεται λίγο πιο περίπλοκο. Οι μεταβλητές αντιστοιχούν σε κόμβους, και έχουμε την υπαρξιακή δομή $S' = \{M_1, M_2\}$ με τις μοναδιαίες σχέσεις M_1, M_2 να δίνουν το χρώμα ενός κόμβου. $(\neg M_1(x) \wedge M_2(x)$ το πρώτο $M_1(x) \wedge \neg M_2(x)$ το δεύτερο και $M_1(x) \wedge M_2(x)$ το τρίτο, το $\neg M_1(x) \wedge M_2(x)$ δεν επιτρέπεται). Έχουμε την ακόλουθη έκφραση:

$$\exists M_1, M_2 \forall x, y \begin{cases} \neg(E(x,y) \wedge \neg M_1(x) \wedge M_2(x) \wedge \neg M_1(y) \wedge M_2(y)) \wedge \\ \neg(E(x,y) \wedge M_1(x) \wedge \neg M_2(x) \wedge M_1(y) \wedge \neg M_2(y)) \wedge \\ \neg(E(x,y) \wedge M_1(x) \wedge M_2(x) \wedge M_1(y) \wedge M_2(y)) \wedge \\ \neg(\neg M_1(x) \wedge \neg M_2(x)) \end{cases}$$

□

ΠΑΡΑΔΕΙΓΜΑ 2.1.1.3

Το 3SAT εκφρασμένο ως SNP: Η δομή εισόδου S αποτελείται από τέσσερεις τριαδικές σχέσεις C_0, C_1, C_2, C_3 ορισμένες στο πεδίο $\{0,1\}$, οπού C_i αντιστοιχεί στη διάζευξη τριών μεταβλητών με τις πρώτες i από αυτές με άρνηση. Η υπαρξιακή δομή S' αποτελείται μόνο από μια monadic σχέση T την απονομή αληθείας. Η σχέση που πρέπει να ικανοποιείται είναι ότι για κάθε x_1, x_2, x_3 , αν $C_0(x_1,x_2,x_3)$ τότε $T(x_1) \vee T(x_2) \vee T(x_3)$, και ομοίως για τα υπόλοιπα C_i με άρνηση για τα $T(x_j)$ αν $j \leq i$. □

Ας ασχοληθούμε με το παρακάτω ερώτημα: Ποιες υποκλάσεις του NP έχουν την ίδια υπολογιστική δύναμη όση όλο το NP;

Δηλαδή, ποιες υποκλάσεις του NP είναι τέτοιες ώστε για κάθε πρόβλημα που ανήκει στο NP να υπάρχει ένα ισοδύναμο με πολυωνυμική αναγωγή πρόβλημα στην εν λόγω υποκλάση. Συγκεκριμένα λέμε ότι δύο προβλήματα A και B είναι ισοδύναμα κάτω από πολυωνυμικού χρόνου αναγωγή αν υπάρχει μια πολυωνυμικού χρόνου αναγωγή από το A στο B, αλλά και από το B στο A. Αποδεικνύεται ότι κάθε πρόβλημα του NP είναι ισοδύναμο με ένα πρόβλημα στο SNP κάτω από πολυωνυμικού χρόνου αναγωγή. Αυτό σημαίνει ότι για κάθε πρόβλημα A στο NP, υπάρχει ένα πρόβλημα B στο SNP τέτοιο ώστε με πολυωνυμικό χρόνο να περνάμε από το A στο B και αντίστροφα. Μάλιστα αυτό ισχύει και για πιο περιορισμένες κλάσεις του SNP. Θα αρχίσουμε θεωρώντας ότι οι σχέσεις ισότητας και ανισότητας δεν επιτρέπονται σε μια πρώτης τάξης σχέση, αλλά μόνο σχέσεις από την δομή εισόδου S και την υπαρξιακή δομή S'. Για την κλάση monotone SNP χωρίς ανισότητες, απαιτούμε ότι όλες οι εμφανίσεις της σχέσης εισόδου a_i στο Φ χρησιμοποιούνται χωρίς άρνηση, ώστε τα a_i να μπορούν να θεωρηθούν περιορισμοί, από την άποψη ότι αν ισχύει η a_i για περισσότερα στοιχεία της δομής εισόδου μπορεί να κάνει το πρόβλημα "λιγότερο ικανοποιήσιμο". Παρατηρούμε ότι το 3SAT έχει αυτή την ιδιότητα. Για monadic SNP χωρίς ανισότητες απαιτούμε η υπαρξιακή δομή S' να αποτελείται από monadic σχέσεις μόνο. Ο παραπάνω ορισμός του 3SAT εμπίπτει και σε αυτή την κλάση. Τέλος για monotone monadic SNP με ανισότητα, θεωρούμε ότι η γλώσσα περιέχει και την ισότητα άρα και η ισότητα και η ανισότητα ανήκουν στη Φ. (Αν θεωρήσουμε ότι οι ισότητες και ανισότητες εμφανίζονται με αρνητική πολικότητα, τότε μόνο οι ανισότητες δίνουν μεγαλύτερη δύναμη στη γλώσσα, αφού η πρόταση της μορφής "εάν $x = y$ τότε $\Phi(x,y)$" μπορεί να αντικατασταθεί με "$\Phi(x,x)$".)

Έτσι έχουμε πάρει την κλάση SNP, και ασχολούμαστε με τρεις συντακτικούς περιορισμούς, συγκεκριμένα monotonicity, monadicity και να μη χρησιμοποιούνται ανισότητες. Για την ώρα θα ασχοληθούμε με τις κλάσεις προβλημάτων που δύο από αυτούς τους περιορισμούς ισχύουν.

2.1.2 NP = monotone, monadic SNP με ανισότητα

ΘΕΩΡΗΜΑ 2.1.2.1

NP = monotone, monadic SNP με ανισότητα: Κάθε πρόβλημα στο NP έχει ένα ισοδύναμο (κάτω από πολυωνυμικού χρόνου αναγωγή) πρόβλημα στο monotone monadic SNP με ανισότητες ■

ΑΠΟΔΕΙΞΗ

Οι Hillebrand, Kanellakis, Mairson και Vardi [17] απέδειξαν ότι η monadic Datalog με ανισότητα μπορεί να χρησιμοποιηθεί για να κωδικοποιηθεί σε πολυωνυμικό χρόνο υπολογισμών μιας Turing machine, ακόμα και μη ντετερμινιστικής. Ένα πρόγραμμα Datalog είναι ένας τύπος Φ ο οποίος αποτελείτε από conjunction από τύπους της μορφής $R_0(x_0) \leftarrow R_1(x_1) \wedge \ldots \wedge R_k(x_k)$, όπου τα x_i μπορούν να έχουν κοινές μεταβλητές. Η σχέση R_0 δεν μπορεί να είναι μια σχέση εισόδου, και άρα πρέπει να είναι monadic ή να έχει πολλαπλότητα 0, επίσης οι R_i μπορούν να είναι σχέσεις ανισότητας. Υπάρχει μια συγκεκριμένη $\overline{R_0}$ πολλαπλότητας μηδέν η οποία πρέπει να απορρέει από το πρόγραμμα αν το πρόγραμμα αποδέχεται την είσοδό του, και αυτό σημαίνει ότι η είσοδος δεν γίνεται αποδεκτή από ένα Datalog πρόγραμμα αν $(\exists R)(\forall x)(\Phi(R,S,x) \vee \neg\overline{R_0})$. Παρατηρούμε ότι αυτός ο τύπος F' είναι monotone monadic SNP με ανισότητα. Εδώ το S περιγράφει τον υπολογισμό μιας μη ντετερμινιστικής μηχανής Turing, συμπεριλαμβανομένης της εισόδου, την περιγραφή της κίνησης της κεφαλής στην ταινία της μηχανής, και τις καταστάσεις της μηχανής και των τιμών στην ταινία καθ' όλη τη διάρκεια του υπολογισμού. Θα θέλαμε τώρα να υποθέσουμε ότι ο υπολογισμός αυτός δεν είναι γνωστός από την αρχή, δηλαδή, μόνο η είσοδος της μηχανής δίνεται. η κίνηση της κεφαλής και οι τιμές της ταινίας δεν είναι γνωστές και είναι υπαρξιακά ποσοδεικτούμενες. Δυστυχώς, αυτή η περιγραφή της κίνησης της κεφαλής δεν μπορεί να περιγραφεί με monadic σχέσεις, γιατί καθορίζεται από την είσοδο της μηχανής. Αποφεύγουμε αυτή τη δυσκολία θεωρώντας ότι η μηχανή Turing είναι χωρίς μνήμη (oblivious) δηλαδή η κεφαλή κινείται στο χώρο που αρχικά υπάρχει η είσοδος μπρος πίσω και τελικά αποδέχεται μετά από n^k βήματα για κάποιο k. Μπορούμε τότε να υποθέσουμε ότι η κίνηση της κεφαλής δίνεται σαν μέρος της εισόδου, αφού πρέπει να είναι ανεξάρτητη των τιμών που χρησιμοποιούνται για είσοδο σε μια τέτοια μηχανή χωρίς μνήμη. Έτσι μόνο οι καταστάσεις της μηχανής και οι τιμές της ταινίας πρέπει να ποσοδεικτοθούν υπαρξιακά, δίνοντας έναν monotone monadic SNP με ισότητα τύπο ο οποίος εκφράζει αν η μηχανή αποδέχεται μια συγκεκριμένη είσοδο. Ένας συγκεκριμένος υπολογισμός τότε περιγράφεται με την επιλογή των καταστάσεων της μηχανής αλλά και των τιμών της ταινίας, οι οποίες εκφράζονται με monadic υπαρξιακές σχέσεις οι οποίες μετά χρησιμοποιούνται σαν είσοδοι στο πρόγραμμα Datalog. Ο περιορισμός που πρέπει να ικανοποιείται είναι ότι αν μια κατάσταση έχει χαρακτηριστεί ως η n^k-οστή κατάσταση (αυτό καθορίζεται από το ντετερμινιστικό κομμάτι της μηχανής), τότε πρέπει επίσης να χαρακτηριστεί και ως κατάσταση αποδοχής (αυτό καθορίζεται από το μη ντετερμινιστικό μέρος της μηχανής). Ο monotone monadic SNP με ανισότητα τύπος δεν θα απορρίπτει αν δεν περιγράφει μια είσοδο ακολουθούμενη από τη σωστή κίνηση της κεφαλής και τον επακόλουθο χωρίς μνήμη υπολογισμό, ενώ θα αποδέχεται αν ο αριθμός των κελιών που χρησιμοποιούνται για τον υπολογισμό είναι μικρότερος από n^k, και θα αποδέχεται όταν αποδέχεται και η μηχανή. ■

2.1.3 NP = monadic SNP χωρίς ανισότητα

ΘΕΩΡΗΜΑ 2.1.3.1

NP = monadic SNP χωρίς ανισότητα: Κάθε πρόβλημα στο NP έχει ένα ισοδύναμο (κάτω από πολυωνυμικού χρόνου αναγωγή) πρόβλημα στο monadic SNP χωρίς ισότητα ■

ΑΠΟΔΕΙΞΗ

Αφού η ύπαρξη ενός ισοδύναμου προβλήματος στο monotone monadic SNP με ανισότητα για κάθε πρόβλημα στο NP έχει ήδη δειχθεί, είναι αρκετό να αφαιρέσουμε τις ανισότητες χρησιμοποιώντας ότι δεν έχουμε monotonicity.

Για να γίνει αυτό εισάγουμε μια νέα binary σχέση εισόδου την eq. Προσαρτούμε στον τύπο ένα conjunct που απαιτεί η eq να είναι η σχέση ισότητας με την ιδιότητα ότι αν μια monadic σχέση (εισόδου ή υπαρξιακή) ισχύει σε κάποια στοιχεία, τότε πρέπει να ισχύει και αν κάποιο στοιχείο σε θέση ορίσματος αντικατασταθεί από εα στοιχείο που σχετίζεται με το αρχικό μέσω της eq, τέλος αντικαθιστούμε όλες τις εμφανίσεις του $x \neq y$ με $\neg eq(x, y)$. Έτσι ο τύπος πλέον δεν περιέχει ανισότητες, αλλά περιέχει μια σχέση εισόδου η οποία εμφανίζεται και με θετική και αρνητική πολικότητα, δηλαδή δεν είναι πλέον monotone. Ο τύπος τελικά είναι monadic SNP χωρίς ανισότητα. ■

2.1.4 NP = monotone SNP χωρίς ανισότητα

ΘΕΩΡΗΜΑ 2.1.4.1

NP = monotone SNP χωρίς ανισότητα: Κάθε πρόβλημα στο NP έχει ένα ισοδύναμο (με πολυωνυμικού χρόνου αναγωγή) πρόβλημα στο monotone SNP χωρίς ανισότητα. ■

ΑΠΟΔΕΙΞΗ

Αφού η ύπαρξη ενός ισοδύναμου προβλήματος στο monotone monadic SNP με ανισότητα για κάθε πρόβλημα στο NP έχει ήδη αποδειχθεί, είναι αρκετό να αφαιρέσουμε τις ανισότητες χάνοντας τη monadicity.

Για να γίνει αυτό, η σκέψη είναι οτί με την ισότητα κάποια marked στοιχεία δημιουργούν ένα succ μονοπάτι με το pred να είναι το transitive closure του. Εισάγουμε μια μοναδιαία σχέση εισόδου special, μια διαδική σχέση εισόδου succ, μια μοναδιαία υπαρξιακή σχέση marked, μια διαδική υπαρξιακή σχέση eq, και μια διαδική υπαρξιακή σχέση pred. Απαιτούμε τώρα οτί κάθε special σοιχείο θα είναι marked, και κάθε στοιχείο που σχετίζεται με ένα marked στοιχείο με την succ θα είναι επίσης marked. Απαιτούμε οτί το pred είναι transitive, αλλά δεν σχετίζουμε κανένα στοιχείο δε σχετίζεται με τον εαυτό του, ότι δύο στοιχεία που συνδέονται με την succ συνδέονται και με την pred (με την ίδια φορά), οτί η eq είναι μια σχέση ισοδυναμίας, οτί οποιαδήποτε special στοιχεία σχετίζονται με την eq, και ότι αν δύο στοιχεία σχετίζονται από την eq και αν κάθε ένα σχετίζεται με κάποιο άλλο μέσω της succ (με την ίδια φορά), τότε αυτά τα δύο επίσης σχετίζονται από την eq. Τέλος, περιορίζουμε τον αρχικό τύπο να έχει μόνο marked στοιχεία, και αντικαθιστούμε το $x \neq y$ από $\neq eq(x, y)$, και θεωρούμε οτί μια σχέση ισχύει σε κάποια στοιχεία αν ισχύει σε στοιχεία που σχετίζονται με τα αρχικά μέσω της eq. Να σημειώσουμε εδώ οτί για στοιχεία που είναι ανάγκη να είναι marked, οι σχέσεις eq και pred μπορούν να οριστούν το πολύ με ένα τρόπο, δίνοντας ένα succ μονοπάτι (και τελικά την eq, με pred την transitive closure της). ■

2.2 MMSNP

2.2.1 Εισαγωγή

Τελικά είδαμε ότι (αν $P \neq NP$) οι κλάσεις monotone monadic SNP with inequality, monadic SNP without inequality, και monotone SNP without inequality, είναι ακριβώς ισοδύναμες με την κλάση NP, αφού μέσω αυτών μπορούμε να προσομοιώσουμε μια Turing Machine. Άμεσο αποτέλεσμα των παραπάνω είναι οτί και στις τρεις παραπάνω κλάσεις μπορούμε να χρησιμοποιήσουμε διαγωνοποιήσεις ανάλογες με αυτή του θεωρήματος του Ladner και να αποδείξουμε οτί οι παραπάνω κλάσεις έχουν προβλήματα που είναι στο NP αλλά δεν είναι ούτε στο P ούτε NP-complete.

Όμως το ίδιο δεν ισχύει αν θέσουμε και τους τρεις περιορισμούς ταυτόχρονα. Στο monotone monadic SNP without inequality (που ονομάζουμε για συντομία MMSNP), δεν μπορούμε να χρησιμοποιήσουμε διαγωνιοποιήσεις της μορφής του Ladner για να πάρουμε ανάλογα αποτελέσματα. Έτσι η κλάση MMSNP είναι μια αρκετά μεγάλη κλάση προβλημάτων για τα οποία μπορεί να ισχύει η διχοτόμιση σε P και NP-complete ή τουλάχιστον δεν μπορούμε να αποδείξουμε το αντίθετο. Μάλιστα μπορούμε να ορίσουμε απλούστερα τα MMSNP ως εξής: Αν μία σχέση (είτε υπαρξιακή είτε εισόδου) εμφανίζεται και θετική και αρνητική τότε πρέπει να είναι monadic.

Το ακόμα πιο ενδιαφέρον είναι οτί πρακτικά με την κλάση MMSNP όπως θα αποδείξουμε παρακάτω υπολογίζουμε πρακτικά ότι μπορούμε να υπολογίσουμε με την κλάση προβλημάτων που μπορούν να εκφραστούν ως CSPs. Άρα τελικά η κλάση προβλημάτων CSP όχι μόνο είναι μια κλάση που πολύ πιθανώς ισχύει η διχοτόμιση σε P και NP-complete αλλά είναι και μια από τις μεγαλύτερες κλάσεις που ισχύει αυτό.

2.2.2 MMSNP $\cong$ CSP

Είναι εύκολο να δούμε οτί $CSP \subseteq MMSNP$. Έστω A το σύμπαν του εν λόγω CSP. Εισάγουμε για κάθε στοιχείο a του σύμπαντος μια υπαρξιακή monadic σχέση $M_a(x)$ με σημασία ότι η μεταβλητή x έχει την τιμή a Τέλος δημιουργούμε τον τύπο Φ έτσι ώστε να περιέχει όλους τους περιορισμούς και να απαιτεί οτί μόνο μια τιμή έχει ανατεθεί σε κάθε μεταβλητή.

Μπορεί να αποδειχθεί ότι $CSP \subset MMSNP$ δηλαδή ότι η σχέση είναι αυστηρή, και υπάρχει πρόβλημα του MMSNP που δεν μπορεί να εκφραστεί ως CSP. Παρόλα αυτά η υπολογιστική δύναμη που προσφέρουν είναι πρακτικά η ίδια αφού μπορούμε με πιθανοτικό πολυωνυμικό αλγόριθμο να περάσουμε από το MMSNP στο CSP. (Η απόδειξη είναι αρκετά περίπλοκη και δεν θα την αναλύσουμε εδώ αλλά μπορεί να βρεθεί στο [37]).

Κεφάλαιο 3

Άλγεβρα

Όπως έχουμε ήδη αναφέρει σκοπός μας είναι να χαρακτηρίσουμε προβλήματα CSP ως προς την πολυπλοκότητά τους. Σε αυτό το κεφάλαιο θα δούμε πως μπορούμε να τα χαρακτηρίσουμε μέσω αλγεβρικών δομών που ορίζονται από αυτά. Σε κάθε υποενότητα θα δούμε διαφορετικές δομές που μπορούν να μας δώσουν πληροφορίες.

3.1 Γλώσσες Περιορισμών

3.1.1 Ορισμοί

ΟΡΙΣΜΟΣ 3.1.1.1

Tractable πρόβλημα: Θα λέμε ένα πρόβλημα tractable αν υπάρχει ντετερμινιστικός πολυωνυμικού χρόνου αλγόριθμος που λύνει όλες τις πιθανές εισόδους του προβλήματος. $\square$

ΟΡΙΣΜΟΣ 3.1.1.2

Tractable γλώσσα περιορισμών: Για κάθε σύνολο A, μια πεπερασμένη γλώσσα περιορισμών $\Gamma \subseteq R_A$ λέγεται tractable αν το CSP(Γ) είναι tractable. Μια άπειρη γλώσσα περιορισμών λέγεται tractable αν κάθε πεπερασμένο υποσύνολό της είναι tractable. $\square$

ΟΡΙΣΜΟΣ 3.1.1.3

NP-complete γλώσσα περιορισμών: Μια γλώσσα περιορισμών Γ λέγεται NP-complete αν $\exists \Delta \subset \Gamma$ με Δ πεπερασμένο έτσι ώστε το $CSP(\Delta)$ να είναι NP-Complete. $\square$

3.1.2 Παραδείγματα

ΠΑΡΑΔΕΙΓΜΑ 3.1.2.1

Χρωματισμός γράφων: Όπως είδαμε πιο πάνω ο q-Χρωματισμός γράφων εκφράζεται μέσω της γλώσσας περιορισμών $\{\neq_q\}$. Γνωρίζουμε όμως ότι ενώ για $q \leq 2$ το πρόβλημα λύνεται στο P ενώ αλλιώς είναι NP-complete. Άρα η $\{\neq_q\}$ είναι tractable για $q \leq 2$ διαφορετικά είναι NP-complete. □

ΠΑΡΑΔΕΙΓΜΑ 3.1.2.2

H-Colorability: Αυτό το πρόβλημα έχει πλήρως χαρακτηριστεί και συγκεκριμένα ξέρουμε ότι για H μη κατευθυνόμενο γράφο όταν είναι bipartite ή έχει βρόγχο τότε το πρόβλημα είναι tractable, αλλιώς είναι NP-complete.

Γνωρίζοντας ήδη οτι εδώ $\Gamma = \{E_H\}$, ξέρουμε ότι μια συμμετρική διμελής σχέση E είναι tractable όταν ανήκει σε bipartite γράφο ή σε ένα γράφο με βρόγχο. □

Αν μια πεπερασμένη ή άπειρη γλώσσα περιορισμών Γ έχει την ιδιότητα $CSP(\Gamma)$ είναι tractable θα ονομάζεται globally tractable. Εμφανώς μια γλώσσα περιορισμών που είναι globally tractable είναι tractable, με βάση τον παραπάνω ορισμό, αλλά δεν είναι άμεσα εμφανές γιατί το αντίστροφο θα έπρεπε να ισχύει για όλες τις μη πεπερασμένες γλώσσες. Έστω κάποια άπειρη γλώσσα περιορισμών Γ, μπορεί να ισχύει ότι για κάθε πεπερασμένο υποσύνολο $\Delta \subseteq \Gamma$ υπάρχει ένας πολυωνυμικός αλγόριθμος $Alg(\Delta)$ που επιλύει το $CSP(\Delta)$, αλλά δεν υπάρχει γενικός αλγόριθμος πολυωνυμικού χρόνου που να λύνει το $CSP(\Gamma)$. Παρόλα αυτά δεν γνωρίζουμε παραδείγματα οπού ισχύει αυτό και υποθέτουμε οτι κάθε tractable γλώσσα περιορισμών είναι και globally tractable.

ΠΑΡΑΔΕΙΓΜΑ 3.1.2.3

Γ_{LIN}: Έστω A ένα πεπερασμένο πεδίο, και έστω Γ_{LIN} μια γλώσσα περιορισμών που περιέχει όλες τις σχέσεις ορισμένες στο A που είναι λύσεις σε κάποιο σύστημα γραμμικών εξισώσεων στο A. Κάθε σχέση στο Γ_{LIN}, και άρα κάθε $CSP(\Gamma_{LIN})$, μπορεί να αναπαρασταθεί από ένα σύνολο γραμμικών εξισώσεων στο A. Πράγματι αν $\rho \in \Gamma_{LIN}$, τότε είναι ο χώρος λύσεων ενός συστήματος γραμμικών εξισώσεων που παίρνουμε από τα παρακάτω βήματα:

1ό Πάρε ένα τυχαίο στοιχείο $\overline{a_0} = (a_{01}, \dots a_{0n}) \in \rho$, και θέσε $\rho_0 = \{\bar{b} - \overline{a_0} | \bar{b} \in \rho\}$.

2ό Για κάθε μέλος $(a_1, \dots, a_n)$ του ρ_0, δημιούργησε την εξίσωση: $a_1x_1 + \dots + a_nx_n = 0$, και βρες μια βάση $r^{\perp}$, του χώρου λύσεων του συστήματος εξισώσεων που προέκυψε.

3ό Για κάθε $(b_1, \dots, b_n) \in \rho^{\perp}$, δώσε την εξίσωση $b_1x_1 + \dots + b_nx_n = b_0$, οπού $b_0 = b_1a_{01} + \dots + b_na_{0n}$.

Εφόσον το $CSP(\Gamma_{LIN})$ μπορεί να λυθεί σε πολυωνυμικό χρόνο (π.χ με Gaussian elimination), είναι επακόλουθο ότι το Γ_{LIN} είναι (globally) tractable γλώσσα περιορισμών. □

ΘΕΩΡΗΜΑ 3.1.2.1

Διχοτόμησης του Schaefer: (Η απόδειξη του θεωρήματος δίνεται στο τελευταίο κεφάλαιο). Όμως μπορούμε από τώρα να εκφράσουμε τα αποτελέσματα στη γλώσσα μας:

Μια γλώσσα περιορισμών ορισμένη στο $A = \{0, 1\}$ είναι tractable αν (τουλάχιστον) ένα από τα παρακάτω ισχύει:

1. Κάθε σχέση στο Γ περιέχει ένα tuple που είναι όλο 0.

2. Κάθε σχέση στο Γ περιέχει ένα tuple που είναι όλο 1.

3. Κάθε σχέση στο Γ μπορεί να οριστεί από ένα τύπο σε conjunctive normal form οπού κάθε conjunct έχει το πολύ μια μεταβλητή με άρνηση.

4. Κάθε σχέση στο Γ μπορεί να οριστεί από ένα τύπο σε conjunctive normal form οπού κάθε conjunct έχει το πολύ μια μεταβλητή χωρίς άρνηση.

5. Κάθε σχέση στο Γ μπορεί να οριστεί από μια CNF με κάθε conjunct να περιέχει το πολύ 2 literal.

6. Κάθε σχέση στο Γ είναι το σύνολο λύσεων ενός συστήματος γραμμικών εξισώσεων στο πεπερασμένο field $GF(2)$.

Αλλιώς είναι NP-complete. ■

Βλέπουμε λοιπόν ότι το $CSP(B)$ όταν το B περιέχει μόνο 2 στοιχεία είναι πλήρως ορισμένο[35]. Επίσης όπως αναφέρθηκε και στην εισαγωγή πλέον έχει πλήρως χαρακτηριστεί και για 3 στοιχεία [3]. Αλλά για μεγαλύτερο πληθάριθμο δεν έχουμε γενικά αποτελέσματα ακόμα.

ΠΑΡΑΔΕΙΓΜΑ 3.1.2.4

Γ_{ZOA}**:** Μια από τις πρώτες γλώσσες περιορισμών που χαρακτηρίστηκε πλήρως (και δεν ήταν ορισμένη σε boolean) ήταν η Γ_{ZOA} των "0,1,all" σχέσεων που περιγράφονται στο [9]. Το σύνολο Γ_{ZOA} περιέχει όλες τις σχέσεις ορισμένες σε κάποιο συγκεκριμένο σύνολο A με τις εξής μορφές:

1. Όλες τις μοναδιαίες σχέσεις

2. Όλες τις διμελείς της μορφής $A_1 \times A_2$ με $A_1, A_2 \subseteq A$.

3. Όλες τις διμελείς σχέσεις της μορφής $\{(\alpha, \pi(\alpha) | \alpha \in A_1\}$ για κάποιο υποσύνολο A_1 του A και κάποια αναδιάταξη π του A.

4. Όλες τις διμελείς σχέσεις της μορφής $\{(a, b) \in A_1 \times A_2 | a = a_1 \vee b = a_2\}$ για κάποια υποσύνολα A_1, A_2 του A και κάποια στοιχεία $a_1 \in A_1$, $a_2 \in A_2$

Έχει δειχθεί στο [9] ότι το $CSP(\Gamma_{ZOA})$ όχι μόνο είναι tractable αλλά και για οποιαδήποτε διμελή σχέση ρ προσθαίσουμε που δεν είναι στο Γ_{ZOA} τότε το $CSP(\Gamma_{ZOA} \cup \{\rho\})$ είναι NP-complete. □

3.2 Relational Clones

3.2.1 Ορισμοί

Για να περιγράψει τις tractable γλώσσες περιορισμών ο Schaefer χρησιμοποίησε τις συντακτικές ιδιότητες των προτασιακών τύπων που αντιστοιχούσαν σε Boolean σχέσεις. Σε μη Boolean σχέσεις προφανώς αυτή η μέθοδος δεν μπορεί να χρησιμοποιηθεί. Χρειαζόμαστε λοιπόν μια επαρκή γλώσσα οπού μπορούμε να εκφράσουμε τις ιδιότητες μιας γλώσσας περιορισμών που είναι υπεύθυνες για την πολυπλοκότητα των αντίστοιχων CSPs.

Ένα χρήσιμο πρώτο βήμα στο να αντιμετωπίσουμε αυτό το πρόβλημα είναι να σκεφτούμε ποιες πρόσθετες σχέσεις μπορούμε να έχουμε σε μια γλώσσα περιορισμών χωρίς να αλλάξουν την πολυπλοκότητα της συγκεκριμένης κλάσης προβλημάτων. Αυτή η τεχνική χρησιμοποιήθηκε ευρέως στην ανάλυση Boolean CSPs [35], και στην ανάλυση temporal και spatial περιορισμών [31], και εισήχθη στην ανάλυση περιορισμών σε arbitrary πεπερασμένα σύνολα στο [19].

Για να χρησιμοποιήσουμε αυτή την τεχνική πρώτα ορίζουμε μία μέθοδο για να παράγουμε νέες σχέσεις από τις δοσμένες. Αυτή η μέθοδος είναι ένας τρόπος μέσω λογικών τύπων και των δοσμένων σχέσεων να ορίζουμε νέες. Για να ορίσουμε τέτοιους λογικούς τύπους θα χρησιμοποιήσουμε τη συνηθισμένη αντιστοιχία μεταξύ σχέσεων και predicates: Μια σχέση αποτελείται από όλα τα tuples που κάνουν το αντίστοιχο predicate να αληθές. (Θα χρησιμοποιούμε το ίδιο σύμβολο για τις αντίστοιχες σχέσεις και predicates, αφού το σε ποιο αναφερόμαστε θα είναι εμφανές από τα συμφραζόμενα.)

ΟΡΙΣΜΟΣ 3.2.1.1

Relational Clone: Μια γλώσσα περιορισμών $\Gamma \subseteq R_A$ λέγεται relational clone αν περιέχει όλες τις σχέσεις που μπορούν να εκφραστούν με προταβάθμιους τύπους που χρησιμοποιούν:

1. Σχέσεις από το $\Gamma \cup \{=_A\}$
2. σύζευξη και
3. υπαρξιακούς ποσοδείκτες.

Οι πρωτοβάθμιοι τύποι που χρησιμοποιούν μόνο σύζευξη και υπαρξιακούς ποσοδείκτες λέγονται primitive positive (pp) τύποι. □

Για κάθε γλώσσα περιορισμών Γ, υπάρχει ένα μοναδικό relational clone που περιέχει τη Γ, το οποίο συμβολίζεται με $\langle \Gamma \rangle$ και καλείται relational clone που παράγεται από τη Γ. Το σύνολο $\langle\Gamma\rangle$ εμπεριέχει όλες τις σχέσεις που ορίζονται από pp-τύπους χρησιμοποιώντας σχέσεις στο Γ μαζί με τη σχέση της ισότητας.

3.2.2 Παραδείγματα - Αποτελέσματα

ΠΑΡΑΔΕΙΓΜΑ 3.2.2.1

Κλείσιμο σε 2 διμελής σχέσεις: Έστω η Boolean γλώσσα περιορισμών $\Gamma = \{R_1, R_2\}$, οπού $R_1 = \{(0,1), (1,0), (1,1)\}$ και $R_2 = \{(0,0), (0,1), (1,0)\}$.

Είναι προφανές να ελέγξουμε ότι κάθε Binary σχέση μπορεί να εκφραστεί από ένα pp-τύπο που περιέχει τα R_1 και R_2. Για παράδειγμα, η σχέση $R_3 = \{(0,0), (1,0), (1,1)\}$ μπορεί να εκφραστεί από τον τύπο $R_3 = \exists y R_1(x,y) \wedge R_2(y,z)$. Άρα το relational clone που παράγεται από τη Γ, $\langle\Gamma\rangle$, περιέχει όλες τις 16 διμελείς Binary σχέσεις.

Μάλιστα μπορεί να αποδειχθεί ότι το $\langle\Gamma\rangle$ περιέχει ακριβώς αυτές τις Bool σχέσεις (οποιασδήποτε πολλαπλότητας) που μπορούν να εκφραστούν σαν μια σύζευξη μοναδιαίων ή διμελών Boolean σχέσεων [36]. □

Υπάρχουν αρκετοί διαφορετικοί αλλά ισοδύναμοι ορισμοί των relational clones [13], ένας από αυτούς χρησιμοποιήθηκε στο [19] για να αποδειχθεί το παρακάτω θεώρημα. Εδώ δίνεται μια απόδειξη με τον ορισμό που δώσαμε πιο πάνω.

ΘΕΩΡΗΜΑ 3.2.2.1

Αναγωγής στη παράγουσα γλώσσα περιορισμών: Για κάθε σύνολο σχέσεων Γ και κάθε πεπερασμένο σύνολο $\Delta \subseteq \langle\Gamma\rangle$, υπάρχει μια πολυωνυμικού χρόνου αναγωγή από το $CSP(\Delta)$ στο $CSP(\Gamma)$. ■

ΑΠΟΔΕΙΞΗ

Έστω $\Delta = \{\rho_1, \ldots, \rho_k\}$ ένα πεπερασμένο σύνολο σχέσεων ορισμένων στο πεπερασμένο σύνολο A, όπου κάθε ρ_i μπορεί να εκφραστεί ως ένας pp-τύπος που χρησιμοποιεί τις σχέσεις από το Γ και την σχέση ισότητας στο A , $=_A$. Παρατηρούμε ότι μπορούμε να σταθεροποιήσουμε αυτές τις αναπαραστάσεις.

Έστω $(V, A, C) \in CSP(\Delta)$ θα την μετατρέψουμε ως εξής: Για κάθε περιορισμό $(\sigma, \rho) \in C$, με $s = (v_1, \ldots, v_l)$ και το ρ είναι αναπαραστήσιμο από ένα pp-τύπο.

$$\rho(v_1, \ldots, v_l) = \exists u_1, \ldots, u_m(\rho_1(w_1^1, \ldots, w_{l_1}^1) \wedge \ldots \wedge \rho_n(w_1^n, \ldots, w_{l_n}^n))$$

όπου $w_1^1, \ldots, w_{l_1}^1, \ldots, w_1^n, \ldots, w_{l_n}^n \in \{v_1, \ldots, v_l, u_1, \ldots, u_m\}$,

1. προσθέτουμε τις βοηθητικές μεταβλητές $u_1, \ldots, u_m$ στο V (αλλάζοντας ονόματα αν χρειαστεί ώστε κανένα να μην επαναλαμβάνεται)

2. προσθέτουμε περιορισμούς $((w_1^1, \ldots, w_{l_1}^1, \ldots, w_1^n, \ldots, w_{l_n}^n), \rho_n)$ στο C.

3. Αφαιρούμε το (s, ρ) από το C.

Μπορούμε εύκολα να αποδείξουμε ότι με αυτή τη διαδικασία έχουμε ένα ισοδύναμο πρόβλημα στο (V, A, C) και ανήκει στο $CSP(\Gamma \cup \{=_A\})$. Μάλιστα εφόσον όλες οι αναπαραστάσεις σχέσεων από το Δ είναι σταθερές αυτός ο μετασχηματισμός μπορεί να γίνει σε γραμμικό χρόνο ως προς την είσοδο. Τέλος, όλοι οι περιορισμοί της μορφής $((v_1, v_2), =_A)$ μπορούν να απαλειφθούν αντικαθιστώντας όλες τις αναφορές στην v_1 με τη v_2. Αυτός ο μετασχηματισμός μπορεί επίσης να γίνει σε πολυωνυμικό χρόνο. ■

ΠΟΡΙΣΜΑ 3.2.2.1

Ένα σύνολο σχέσεων Γ είναι tractable αν και μόνο αν το relational clone $\langle \Gamma \rangle$ είναι tractable. Ομοίως , το Γ είναι NP-complete αν και μόνο αν το $\langle \Gamma \rangle$ είναι NP-complete. ■

Αυτό το αποτέλεσμα απλοποιεί το πρόβλημα του χαρακτηρισμού tractable γλωσσών περιορισμών στο να χαρακτηρίζουμε relational clones.

ΠΑΡΑΔΕΙΓΜΑ 3.2.2.2

Γ_{ZOA}: Ας ξαναθυμηθούμε την tractable γλώσσα περιορισμών των 0 /1 /all σχέσεων, Γ_{ZOA}.

Παρατηρούμε ότι για οποιοδήποτε πεπερασμένο σύνολο A, το σύνολο των $0/1/all$ σχέσεων στο A περιέχει μόνο μοναδιαίες και διμελής σχέσεις και άρα είναι πεπερασμένη. Όμως, σύμφωνα με το παραπάνω πόρισμα ότι το relational clone $\langle \Gamma_{ZOA} \rangle$ είναι επίσης tractable. Αυτό είναι ένα απειροσύνολο σχέσεων που περιέχει σχέσης οποιασδήποτε πολλαπλότητας. Στη πραγματικότητα το σύνολο $\langle \Gamma_{ZOA} \rangle$ είναι ακριβώς οι σχέσεις που ορίζονται στο [25]. Μάλιστα το σύνολο των 0/1/all σχέσεων, Γ_{ZOA}, είναι ακριβώς το σύνολο των μοναδιαίων και διμελών σχέσεων στο relational clone $\langle \Gamma_{ZOA} \rangle$. □

3.2.3 Ένας διαφορετικός ορισμός των relational clones

Είναι αρκετά ενδιαφέρον να δούμε τον ορισμό που δίνεται εκτός των άλλων και στα [19, 22] που δεν είναι από τον κόσμο της πρωτοβάθμιας λογικής αλλά από τις relational databases. Συγκεκριμένα έχουμε:

ΟΡΙΣΜΟΣ 3.2.3.1

Πράξεις σε σχέσεις: Ορίζουμε τις παρακάτω σχέσεις στις σχέσεις:

- Έστω R μια n-αδική και S μια m-αδική σχέση ορισμένες στο A. Το καρτεσιανό γινόμενο $R \times S$ ορίζεται να είναι η $(n+m)$-αδική σχέση:

$$R \times S = \{(t[1], t[2], \ldots, t[n+m]) | ((t[1], t[2], \ldots, t[n]) \in R) \wedge$$

$$((t[n+1], t[n+2], \ldots, t[n+m]) \in S)\}.$$

- Έστω R να είναι μια n-αδική σχέση ορισμένη στο A. Και $1 \leq i, j \leq n$. Η επιλογή ισότητας $\sigma_{i=j}(R)$ ορίζεται να είναι η n-αδική σχέση:

$$\sigma_{i=j}(R) = \{t \in R | t[i] = t[j]\}.$$

- Έστω R να είναι μια n-αδική σχέση σε ένα σύνολο A. Έστω $i_1, \ldots, i_m$ να είναι μια υπακολουθία του $1, \ldots, n$. Η προβολή $\pi_{i_1,\ldots,i_m}(R)$ ορίζεται να είναι η m-αδική σχέση:

$$\pi_{i_1,\ldots,i_m}(R) = \{(t[i_1], \ldots, t[i_m]) | t \in R\}.$$

□

Είναι γνωστό ότι η συνδυασμένη χρήση δύο περιορισμών σε ένα CSP μπορεί να παραχθεί αν κάνουμε ένα join [7] στους δύο περιορισμούς [16]. Το επόμενο αποτέλεσμα είναι μια συνέπεια του ορισμού του join σε σχέσεις.

ΛΗΜΜΑ 3.2.3.1
Οποιοδήποτε join των σχέσεων R και S μπορεί να υπολογιστεί χρησιμοποιώντας το Καρτεσιανό γινόμενο, τις πράξεις $\sigma()$ και $\pi()$ στις σχέσεις R και S. ■

Έτσι μπορούμε να ορίσουμε το $\langle \Gamma \rangle$ με βάση τις παραπάνω σχέσεις.

3.3 Πολυμορφισμοί

3.3.1 Ορισμοί

Δείξαμε στην προηγούμενη ενότητα οτι το να αναλύσεις την πολυπλοκότητα οποιασδήποτε γλώσσας περιορισμών ήταν αρκετό να σκεφτόμαστε μόνο τα relational clones. Αυτό μειώνει αρκετά τα είδη των γλωσσών που αναλύουμε. 'Ομως, αμέσως προκύπτει το ερώτημα πώς μπορούμε να αναπαραστήσουμε και να περιγράψουμε τα relational clones. Για αρκετά relational clones τα μόνα σύνολα που τα παράγουν είναι αρκετά περίπλοκα, και σε μερικές περιπτώσεις δεν έχουν βρεθεί σύνολα που τα παράγουν.

Ευτυχώς όμως φαίνεται ότι υπάρχει ένας εναλλακτικός τρόπος να απεικονίζουμε και να περιγράφουμε τα relational clones, χρησιμοποιώντας πράξεις (συναρτήσεις). Εδώ μπορεί ο αναγνώστης να ανατρέξει στο πρώτο κεφάλαιο που δίνονται οι ορισμοί των συναρτήσεων. Η παρακάτω παρουσίαση ακολουθεί τα [29, 36]

Πρώτα περιγράφουμε μια σημαντική αλγεβρική σχέση μεταξύ πράξεων και σχέσεων. Παρατηρούμε ότι οποιαδήποτε πράξη ορισμένη στο A μπορεί να επεκταθεί με συγκεκριμένο τρόπο σε μια πράξη tuples του A. Συγκεκριμένα για μια m-αδική f και μια συλλογή από tuples $\overline{a_1}, \dots \overline{a_m} \in A^n$, όπου $\overline{a_i} = (a_{1i}, \dots, a_{ni})$, ορίζουμε:

$$f(\overline{a_1}, \dots, \overline{a_m}) = ((f(a_{11}, \dots, a_{1m}), \dots, f(a_{n1}, \dots, a_{nm}))$$

ή πιο γραφικά μπορούμε να το δούμε ώς:

$$\begin{array}{cc} & \begin{array}{ccccc} & \overline{a_1}, & \overline{a_2}, & \dots & \overline{a_m} \\ & \shortparallel & \shortparallel & & \shortparallel \end{array} \\ f(\overline{a_1}, \dots, \overline{a_m}) = & \begin{pmatrix} f(& a_{11}, & a_{12}, & \dots & a_{1m} &) & = b_1 \\ f(& \vdots & \ddots & & \vdots &) & \vdots \\ f(& \vdots & & \ddots & \vdots &) & \vdots \\ f(& a_{n1}, & a_{n2}, & \dots & a_{nm} &) & = b_n \end{pmatrix} = \overline{b} \end{array}$$

ΟΡΙΣΜΟΣ 3.3.1.1

Πολυμορφισμοί[13, 36]: Μια m-αδική πράξη $f \in O_A$ διατηρεί μια n-αδική σχέση $\rho \in R_A$ (ή η f είναι πολυμορφισμός της r, ή η ρ είναι αναλλοίωτη της f) εάν $f(\overline{a_1}, \dots, \overline{a_m}) \in \rho \ \forall \overline{a_1}, \dots, \overline{a_m} \in \rho$.

Για οποιαδήποτε σύνολα $\mathit{\Gamma} \subseteq R_A$ και $F \subseteq O_A$, έστω:

$$Pol(\mathit{\Gamma}) = \{f \in O_A | f \text{ διατηρεί κάθε σχέση του } \mathit{\Gamma}\},$$

$$Inv(F) = \{\rho \in R_A | \rho \text{ είναι αναλλοίωτη για κάθε πράξη από το } F\}.$$

□

Επισημαίνουμε ότι οι τελεστές Pol και Inv δημιουργούν μια Galois correspondence μεταξύ των R_A και O_A. Εισαγωγή στις ιδιότητες αυτής της σχέσης μπορούν να βρεθούν στο [13]. Πρώτον έχουμε $Inv(F) = Inv(Pol(Inv(F)))$, για κάθε σύνολο από πράξεις F. Σύνολα της μορφής $Inv(F)$ είναι ακριβώς relational clones όπως φαίνεται στο παρακάτω αποτέλεσμα:

ΠΡΟΤΑΣΗ 3.3.1.1

Για κάθε σύνολο A, και κάθε $F \subseteq O_A$, το σύνολο $Inv(F)$ είναι ένα relational clone. Από την άλλη κάθε relational clone μπορεί να αναπαρασταθεί στη μορφή $Inv(F)$ για κάποιο $F \subseteq O_A$. Συγκεκριμένα, για κάθε $\mathit{\Gamma} \subseteq R_A$, $\langle \mathit{\Gamma} \rangle = Inv(Pol(\mathit{\Gamma}))$. Συγκεκριμένα, για οποιοδήποτε $\mathit{\Gamma} \subseteq R_A$, $\langle \mathit{\Gamma} \rangle = Inv(Pol(\mathit{\Gamma}))$.
■

Χρησιμοποιώντας την πάνω πρόταση μπορούμε να μεταφράσουμε το αρχικό πρόβλημα μας του να χαρακτηρίσουμε tractable σύνολα σχέσεων σε ένα ισοδύναμο πρόβλημα για σύνολα από πράξεις. Πρώτον, ορίζουμε τι σημαίνει για ένα σύνολο πράξεων να είναι tractable ή NP-complete.

ΟΡΙΣΜΟΣ 3.3.1.2

Χαρακτηρισμός συνόλου πράξεων: Το $F \subseteq O_A$ είναι tractable εάν το $Inv(F)$ είναι tractable. Αντίστοιχα ένα σύνολο $F \subseteq O_A$ λέγεται NP-complete εάν το $Inv(F)$ είναι NP-complete. □

Και έτσι πλέον το αρχικό μας πρόβλημα μετασχηματίζεται στο εξής: Να χαρακτηριστούν όλα τα tractable σύνολα πράξεων σε πεπερασμένα σύνολα.

Σε πολλές περιπτώσεις η περιγραφή ενός συνόλου από πράξεις δίνει ένα σύντομο και συνεπή τρόπο να περιγράψουμε το αντίστοιχο relational clone, όπως το επόμενο παράδειγμα δείχνει.

3.3.2 Παραδείγματα - Αποτελέσματα

ΠΑΡΑΔΕΙΓΜΑ 3.3.2.1

Έκφραση του Γ_{ZOA} μέσω πολυμορφισμού: Ας ξαναθυμηθούμε τη Γ_{ZOA} των $0/1/all$ σχέσεων που ορίστηκε στο παραπάνω και επεκτάθηκε στο (Γ_{ZOA}). Αποδείχθηκε στο [22] ότι το (Γ_{ZOA}) είναι ακριβώς η γλώσσα περιορισμών που αποτελείται από όλες τις σχέσεις που είναι αναλλοίωτες της "dual discriminator" πράξης d, που ορίζεται ως:

$$d(x, y, z) = \begin{cases} y \; if \; y = z \\ x \; otherwise \end{cases}$$

Άρα αυτή η άπειρη tractable γλώσσα περιορισμών, που είναι αρκετά δύσκολο να περιγράφει με λεπτομέρεια, μπορεί να παρασταθεί πολύ απλά ως το σύνολο των σχέσεων που είναι αναλλοίωτες του tractable συνόλου πράξεων $\{d\}$. □

Σε πολλές περιπτώσεις, έχει δειχθεί οτι η παρουσία μιας μοναδικής πράξης που ικανοποιεί συγκεκριμένες ιδιότητες είναι αρκετή για να είναι σίγουρη η tractability όλου του συνόλου των πράξεων.

ΠΑΡΑΔΕΙΓΜΑ 3.3.2.2

Ιδιότητα που αποδεικνύει tractability: Έστω μια πράξη $f(x, y)$ που ικανοποιεί τις τρεις παρακάτω συνθήκες να καλείται semilattice operation: (Να σημειώσουμε ότι χρησιμοποιείται και ο όρος ACI - (associativity, commutativity, idepotency)).

- $f(x, f(y, z)) = f(f(x, y), z)$ (associativity),
- $f(x, y) = f(y, x)$ (commutativity),
- $f(x, x) = x$ (idempotency).

Το θεώρημα 16 του [21] αποδεικνύει ότι για ένα πεπερασμένο σύνολο A, και σύνολο πράξεων $F \subseteq O_A$ που περιέχει μια semilatice πράξη είναι tractable. □

Σε αντίθεση θα ασχοληθούμε με ιδιότητες σχέσεων οι οποίες δείχνουν NP-completeness των γλωσσών περιορισμών.

ΠΡΟΤΑΣΗ 3.3.2.1

Για κάθε πεπερασμένο σύνολο A και οποιοδήποτε $\Gamma \subseteq R_A$, αν το $Pol(\Gamma)$ περιέχει essentially unary πράξεις μόνο, τότε το $CSP(\Gamma)$ είναι NP-complete. ■

ΠΑΡΑΔΕΙΓΜΑ 3.3.2.3

NP-completeness μέσω πολυμορφισμού: Θεωρήστε σχέση N ορισμένη στο σύνολο $\{0,1\}$, που ορίζεται από:

$$N = \{0,1\}^3 \backslash \{(0,0,0),(1,1,1)\}.$$

Μπορεί να δειχθεί ότι το $Pol(\{N\})$ περιέχει μόνο essentially operations και άρα το $CSP(\{N\})$ είναι NP-complete, από την πιο πάνω πρόταση.

Εδώ να πούμε ότι το $CSP(\{N\})$ αποδείχθηκε πρώτα από τον Schaefer ότι ήταν NP-complete, και είναι μια ειδική περίπτωση του NOT-ALL-EQUAL SATISFIABILITY problem [32, 15]. □

Οι Boolean πράξεις, δηλαδή οι πράξεις στο $A = \{0,1\}$, έχουν μελετηθεί πολύ καλά [36]. Συγκεκριμένα, είναι γνωστό ότι αν μια Boolean γλώσσα περιορισμών αν δεν ανήκει σε μια από τις έξι κλάσεις που όρισε ο Schaefer, τότε όλοι οι πολυμορφισμοί της είναι essentially unary πράξεις. Έτσι μπορούμε να επαναδιατυπώσουμε το θεώρημα διχοτόμησης του Schaefer ώστε να πάρουμε ένα πλήρη χαρακτηρισμό των Boolean πράξεων.

ΠΟΡΙΣΜΑ 3.3.2.1

Ένα σύνολο Boolean πράξεων είναι tractable αν περιέχει μια essentially nonunary πράξη. Αλλιώς είναι NP-complete. ■

3.4 Tractability σε Άλγεβρες

3.4.1 Ορισμοί

Έχει ήδη αποδειχθεί στην προηγούμενη ενότητα οτι το πρόβλημα της ανάλυσης της πολυπλοκότητας μιας γλώσσας περιορισμών μπορεί να μεταφραστεί στο πρόβλημα της ανάλυσης την πολυπλοκότητας ενός συνόλου από πράξεις οι οποίες διατηρούν όλες τις σχέσεις της γλώσσας. Στη περίπτωση Boolean συνόλου, είναι αρκετό για να αποκτήσουμε έναν πλήρη χαρακτηρισμό της πολυπλοκότητας, αλλά σε μεγαλύτερα σύνολα χρειαζόμαστε δυνατότερα αναλυτικά εργαλεία. Όπως θα δούμε στο επόμενο παράδειγμα:

ΠΑΡΑΔΕΙΓΜΑ 3.4.1.1

Παράδειγμα πράξης που δεν μπορούμε να χαρακτηρίσουμε.: Θεωρήστε τη binary πράξη $\circ$ στο σύνολο $\{0,1,2\}$ που ορίζεται από τον παρακάτω πίνακα Cayley:

$\circ$	0	1	2
0	0	1	1
1	1	1	0
2	2	2	2

Αυτή η πράξη δεν εμπίπτει σε καμία από τις γνωστές tractable κλάσεις και δεν είναι essentially unary. Αρα δεν μπορούμε να καθορίσουμε την πολυπλοκότητα αυτής της πράξης χρησιμοποιώντας τα εργαλεία που έχουμε μέχρι τώρα. □

Σε αυτή την ενότητα θα κάνουμε το τελευταίο βήμα της αλγεβρικής μεθόδου, θα περάσουμε από σύνολα πράξεων σε άλγεβρες.

ΟΡΙΣΜΟΣ 3.4.1.1

Άλγεβρα: Μια άλγεβρα είναι ένα διατεταγμένο ζεύγος $\mathcal{A} = (A, F)$ τέτοια ώστε το A να είναι ένα μη κενό σύνολο και F να είναι μια οικογένεια από πεπερασμένες πράξεις στο A. Το σύνολο A καλείται το σύμπαν του $\mathcal{A}$, και οι πράξεις από το F καλούνται βασικές. Μια άλγεβρα με πεπερασμένο σύμπαν καλείται πεπερασμένη άλγεβρα. □

Για να κάνουμε τη μετάβαση από σύνολα πράξεων σε άλγεβρες πρέπει απλά να σημειώσουμε ότι το σύνολο πράξεων F σε ένα σταθερό σύνολο A μπορεί να αντιστοιχιστεί στην άλγεβρα (A, F). Άρα, θα ορίσουμε τί σημαίνει μια άλγεβρα να είναι tractable με βάση το αν οι βασικές πράξεις είναι tractable.

ΟΡΙΣΜΟΣ 3.4.1.2

Tractability σε άλγεβρες: Μια άλγεβρα $\mathcal{A} = (A, F)$ λέγεται tractable αν η F είναι tractable, και αντίστοιχα λέγεται NP-complete αν η F είναι NP-complete. □

Χρησιμοποιώντας τον παραπάνω ορισμό μπορούμε τώρα να μεταφράσουμε το θεώρημα διχοτόμησης του Schaefer στο χαρακτηρισμό των αλγεβρών που ορίζονται σε δισύνολα.

ΠΟΡΙΣΜΑ 3.4.1.1

Μια άλγεβρα που είναι ορισμένη σε εάν σύμπαν δύο στοιχείων είναι NP-Complete αν όλες οι βασικές πράξεις της είναι essentially unary. Αλλιώς είναι tractable. ■

Το πρώτο προτέρημα του να χρησιμοποιούμε άλγεβρες αντί από σύνολα πράξεων είναι οτι μπορούμε να χρησιμοποιήσουμε έτοιμη θεωρία και δομές άλγεβρας ώστε να πάρουμε νέα συμπεράσματα για την πολυπλοκότητα των γλωσσών περιορισμών. Ένα άλλο προτέρημα είναι ότι οι πεπερασμένες άλγεβρες έχουν ήδη μελετηθεί εκτενώς, και υπάρχει ήδη σημαντική θεωρία πάνω στις δομές της [29, 36, 18]. Μέσω αυτών των ιδεών θα καταλήξουμε σε χρήσιμα αποτελέσματα για τον χαρακτηρισμό αλγεβρών.

Στη μελέτη μας είναι χρήσιμο να περιγράψουμε μια σχέση ισοδυναμίας που συνδέει τις άλγεβρες που αντιστοιχούν στην ίδια γλώσσα περιορισμών. Όπως προείπαμε οι απεικονίσεις Pol, Inv έχουν την ιδιότητα $Inv(Pol(Inv(F))) = Inv(F)$, και έτσι μπορούμε να επεκτείνουμε το σύνολο πράξεων F στο σύνολο $Pol(Inv(F))$ χωρίς να αλλάζουμε την αντίστοιχο σύνολο των αναλλοίωτων σχέσεων. Το σύνολο $Pol(Inv(F))$ αποτελείται από όλες τις πράξεις που μπορούμε να πάρουμε από τις πράξεις της F, μαζί με το σύνολο των προβολών με την πράξη της σύνθεσης συναρτήσεων [13]. (Εάν η f είναι n-αδική πράξη στο A, και $g_1, \ldots, g_n$ είναι k-αδικές πράξεις στο A, τότε η σύνθεση της f και των $g_1, \ldots, g_n$ είναι η k-αδική πράξη h στο A που ορίζεται ως $h(a_1, \ldots, a_k)$ $f(g_1(a_1, \ldots, a_k), \ldots, g_n(a_1, \ldots, a_k))$.)

ΟΡΙΣΜΟΣ 3.4.1.3

Πράξεις όροι: Για μία άλγεβρα $\mathcal{A} = (A, F)$, μια πράξη f στο A θα λέγεται πράξη όρος της $\mathcal{A}$ εαν $f \in Pol(Inv(F))$. Το σύνολο όλων των πράξεων όρων της $\mathcal{A}$ θα συμβολίζεται με $Term(\mathcal{A})$. □

Δύο άλγεβρες με το ίδιο σύμπαν ονομάζονται ισοδύναμες ως προς τους όρους αν έχουν το ίδιο σύνολο πράξεων όρων. Παρατηρούμε οτι, για οποιαδήποτε άλγεβρα $\mathcal{A} = (A, F)$, έχουμε $Inv(F) = Inv(Term(\mathcal{A}))$, έτσι δύο άλγεβρες είναι ισοδύναμες ως προς τους όρους αν και μόνο εάν έχουν το ίδιο σύνολο από αναλλοίωτες σχέσεις. Ως αποτέλεσμα χρειάζεται να χαρακτηρίσουμε τις άλγεβρες μέχρι το επίπεδο ισοδυναμίας όρων.

ΠΑΡΑΔΕΙΓΜΑ 3.4.1.2

Χαρακτηρισμός ομάδων: Μια ομάδα είναι μια άλγεβρα με τρεις βασικές πράξεις: έναν δυαδικό πολλαπλασιασμό, τη μοναδιαία πράξη του αντίθετου, και μια σταθερή πράξη της μονάδας [14]. Ένα coset μιας ομάδας είναι μια σχέση που είναι αναλλοίωτη της τριαδικής πράξης όρου $t(x, y, z) = xy^{-1}z$. Αποδεικνύεται οτι οποιαδήποτε γλώσσα περιορισμών αποτελείται από cosets μιας πεπερασμένης ομάδας είναι tractable. Άρα οποιοδήποτε πεπερασμένη ομάδα είναι tractable, και κάθε πεπερασμένη άλγεβρα που έχει την τριαδική πράξη t ως όρο είναι επίσης tractable. □

3.4.2 Ειδικές κλάσεις αλγεβρών

Σε αυτό το τμήμα θα δείξουμε οτι, όταν μελετάμε την tractability των πεπερασμένων αλγεβρών, μπορούμε να περιορίσουμε την προσοχή μας σε ειδικές περιπτώσεις - κλάσεις από άλγεβρες.

ΟΡΙΣΜΟΣ 3.4.2.1

Surjuctive (Επί) άλγεβρες.: Καλούμε μια άλγεβρα surjective (επί) αν όλες οι πράξεις όροι είναι surjective (επί). □

Είναι εύκολο να ελέγξουμε αν μια πεπερασμένη άλγεβρα είναι επί αν και μόνο αν οι μοναδιαίοι πράξεις όροι είναι όλοι επί και άρα αποτελούν μια ομάδα μεταθέσεων.

Έχει δειχθεί στο [19] οτι οποιασδήποτε μοναδιαίος πολυμορφισμός μπορεί να εφαρμοσθεί στο σύνολο των σχέσεων χωρίς να αλλάξει την πολυπλοκότητα του συνόλου.

ΠΡΟΤΑΣΗ 3.4.2.1

Για ένα σύνολο σχέσεων Γ, και οποιαδήποτε μοναδιαία πράξη $f \in Pol(\Gamma)$, έστω $f(\Gamma)$ να είναι το σύνολο των σχέσεων $\{f(\rho)|\rho \in \Gamma\}$, οπού $f(\rho) = \{f(\overline{a})|\overline{a} \in \rho\}$. Το σύνολο Γ είναι tractable αν και μόνο εάν το $f(\Gamma)$ είναι tractable, και το Γ είναι NP-complete αν και μόνο εάν το $f(\Gamma)$ είναι NP-complete.

Κάθε άλγεβρα $\mathcal{A} = (A, F)$ η οποία δεν είναι επί θα έχει μια μοναδιαία πράξη όρο f η οποία δεν θα είναι επί, και άρα θα έχει πεδίο τιμών U, οπού $U \subset A$ (αυστηρό υποσύνολο). Με το να εφαρμόσουμε αυτή την πράξη σε όλες τις σχέσεις στο $Inv(F)$, όπως περιγράφεται από την παραπάνω σχέση μπορούμε να πάρουμε ένα σύνολο σχέσεων ορισμένων στο U χωρίς να αλλάξουμε την πολυπλοκότητα. Η άλγεβρα που αντιστοιχεί στις νέες σχέσεις μπορεί να δειχθεί ότι είναι μια term induced άλγεβρα της $\mathcal{A}$, η οποία ορίζεται ακολούθως. ■

ΟΡΙΣΜΟΣ 3.4.2.2

Term induced άλγεβρα: Έστω $\mathcal{A} = (A, F)$ μια άλγεβρα, και έστω U ένα μη κενό υποσύνολο του A. Η term induced άλγεβρα $\mathcal{A}|_U$ ορίζεται ως $(U, Term(\mathcal{A})|_U)$, οπού $Term(\mathcal{A})|_U = \{g|_U : g \in Term(\mathcal{A})$ και η g διατηρεί το $U\}$. □

Με την επιλογή μιας μοναδιαίας πράξης όρου f με πεδίο τιμών U με το μικρότερο πληθάριθμο, μπορούμε να είμαστε σίγουροι οτι η term induced άλγεβρα $\mathcal{A}|_U$ είναι επί. Άρα έχουμε το επόμενο θεώρημα:

ΘΕΩΡΗΜΑ 3.4.2.1

Αναγωγή κάθε άλγεβρας σε μια επί-άλγεβρα: Για κάθε πεπερασμένη άλγεβρα $\mathcal{A}$, υπάρχει ένα σύνολο U τέτοιο ώστε:

- η άλγεβρα $\mathcal{A}|_U$ είναι επί και
- η άλγεβρα $\mathcal{A}$ είναι tractable αν και μόνο εάν η $\mathcal{A}|_U$ είναι tractable, και είναι NP-complete ανν η $\mathcal{A}|_U$ είναι NP-complete.

■

Το παραπάνω θεώρημα αποδεικνύει ότι μπορούμε να περιορίσουμε την προσοχή μας σε επί άλγεβρες. Το επόμενο θεώρημα αποδεικνύει ότι σε πολλές περιπτώσεις αρκεί να ενδιαφερόμαστε για αυτές τις επί άλγεβρες οι οποίες είναι και idempotent.

ΟΡΙΣΜΟΣ 3.4.2.3

Idempotent απλοποίηση: Θυμόμαστε ότι μια πράξη f ορισμένη στο A, καλείται idempotent όταν ικανοποιεί $f(x, \ldots, x) = x$ για κάθε $x \in A$. Η πλήρης idempotent απλοποίηση μιας άλγεβρας $\mathcal{A} = (A, F)$ είναι η άλγεβρα $(A, Term_{id}(\mathcal{A}))$, οπού $Term_{id}(\mathcal{A})$ αποτελείται από όλες τις idempotent πράξεις από το $Term(\mathcal{A})$. □

Παρατηρούμε οτι μια πράξη f σε ένα σύνολο A είναι idempotent ανν διατηρεί όλες τις σχέσεις του συνόλου $\Gamma_{CON} = \{\{(a)\}|a \in A\}$, που αποτελείται από όλες τις μοναδιαίες ενός στοιχείου σχέσεις του A. Άρα, $Inv(Term_{id}(\mathcal{A}))$ είναι το relational clone που παράγεται από $Inv(F) \cup \Gamma_{CON}$.

Για το επόμενο θεώρημα χρειαζόμαστε το ακόλουθο βοηθητικό λήμμα του [36]

ΛΗΜΜΑ 3.4.2.1

Έστω $\mathcal{A} = (\{a_1, a_2, \ldots, a_k\}, F)$ να είναι μια πεπερασμένη άλγεβρα της οποίας οι μοναδιαίες πράξεις όροι είναι μια ομάδα μεταθέσεων G. Τότε η σχέση ρ_G, που ορίζεται ως:

$$\rho_G = \{(g(a_1), \ldots, g(a_k))|g \in G\},$$

ανήκει στο $Inv(F)$. ■

ΑΠΟΔΕΙΞΗ

Από την πρόταση 1.3 του [36] έχουμε ότι οι σχέσεις της μορφής ρ_G διατηρούνται από όλες τις πράξεις της άλγεβρας $\mathcal{A}$ και άρα ανήκουν στο $Inv(F)$. ■

ΘΕΩΡΗΜΑ 3.4.2.2

Ισοδυναμία idempotent Αλγεβρών με τις αρχικές: Μια πεπερασμένη επί άλγεβρα $\mathcal{A}$ είναι tractable ανν η πλήρης idempotent απλοποίησή της $\mathcal{A}_0$ είναι tractable. Ομοίως για τον χαρακτηρισμό NP-complete. ■

ΑΠΟΔΕΙΞΗ

Έστω $\mathcal{A} = (A, F)$ να είναι μια πεπερασμένη επί άλγεβρα, και έστω $\mathcal{A}_0$ να είναι η πλήρης idempotent απλοποίηση της $\mathcal{A}$.

Όπως παρατηρήσαμε πιο πάνω, $Inv(Term(\mathcal{A}_0))$ είναι το relational clone που παράγεται από το σύνολο $Inv(F) \cup \Gamma_{CON}$, όπου $\Gamma_{CON} = \{\{(a)\}|a \in A\}$. Άρα η $\mathcal{A}_0$ είναι tractable ανν $Inv(F) \cup \Gamma_{CON}$ είναι tractable, και $\mathcal{A}_0$ είναι NP-complete αν και μόνο αν $Inv(F) \cup \Gamma_{CON}$ είναι NP-complete. Στην υπόλοιπή απόδειξη θα δείξουμε ότι το $CSP(Inv(F) \cup \Gamma_{CON})$ είναι σε πολυωνυμικό χρόνο ισοδύναμο με το $CSP(Inv(F))$.

Προφανώς, κάθε είσοδος της μορφής $CSP(Inv(F))$ μπορεί να θεωρηθεί είσοδος της μορφής $CSP(Inv(F) \cup \Gamma_{CON})$, άρα υπάρχει μια σταθερού χρόνο αναγωγή από το $CSP(Inv(F))$ στο $(CSP(Inv(F) \cup \Gamma_{CON})$.

Για το αντίστροφο αποτέλεσμα, έστω $\mathcal{P} = (V, A, C)$ μια είσοδος της μορφής $CSP(Inv(F) \cup \Gamma_{CON})$ και έστω $\mathcal{P}'$ να είναι το πρόβλημα με είσοδο (V', A, C'), όπου $V' = V \cup \{v_a | a \in A\}$ (κάθε μια από τις μεταβλητές v_a είναι νέα μεταβλητή του V). Για να αποκτήσουμε τους περιορισμούς C', παίρνουμε τις αρχικούς περιορισμούς C του $\mathcal{P}$, και αντικαθιστούμε κάθε μοναδιαίο περιορισμό της μορφής $(v, \{(a)\})$ του C με τον περιορισμό $((v, v_a), =_A)$. Τέλος προσθέτουμε τον περιορισμό $((v_{a_1}, \ldots, v_{a_k}), \rho_G)$, οπού $a_1, \ldots, a_k$ είναι τα στοιχεία του A (σε κάποια σειρά) και ρ_G είναι η σχέση που ορίστηκε στο παραπάνω λήμμα. Παρατηρούμε ότι το $\mathcal{P}'$ είναι μια είσοδος του $CSP(Inv(F))$ και οτι αυτή η κατασκευή μπορεί να γίνει σε πολυωνυμικό χρόνο.

Ισχυριζόμαστε τώρα ότι το πρόβλημα $\mathcal{P}'$ έχει μια λύση c, τότε υπάρχει μία λύση ϕ τέτοια ώστε $\phi(v_a) = a$ για κάθε $a \in A$. Για να αποδείξουμε τον ισχυρισμό, παρατηρούμε ότι από τον ορισμό του ρ_G, υπάρχει μια $g \in G$ τέτοια ώστε $\psi(v_a) = g(a)$ για κάθε $a \in A$. Αφού το G είναι μια ομάδα, η αντίστροφη πράξη $g^{-1} \in G$, που σημαίνει ότι είναι μια πράξη όρος της $\mathcal{A}$. Αυτό σημαίνει ότι κάθε σχέση στο $Inv(F)$ είναι αναλλοίωτη της g^{-1}, και άρα $\phi = g^{-1}\psi$ είναι επίσης μια λύση στο $\mathcal{P}'$, και έχει την ιδιότητα $g^{-1}\psi(v_a) = a$ για κάθε $a \in A$.

Κάθε λύση ϕ η οποία ικανοποιεί αυτή τη συνθήκη προφανώς ικανοποιεί και όλου τους περιορισμούς της C, και άρα ο περιορισμός της ϕ στο V είναι μια λύση του $\mathcal{P}$. Από την άλλη, αν ψ είναι μια λύση του $\mathcal{P}$, τότε η επέκτασή της ψ' στο V' ώστε $\psi'(v_a) = a$, για κάθε $a \in A$, είναι μια λύση του $\mathcal{P}'$. Άρα αυτή η κατασκευή είναι μια πολυωνυμικού χρόνου αναγωγή από το $CSP(Inv(F) \cup \Gamma_{CON})$ στο $CSP(Inv(F))$. ■

Το παραπάνω θεώρημα μπορεί να εκφραστεί με όρους των γλωσσών περιορισμών ως:

ΠΟΡΙΣΜΑ 3.4.2.1

Έστω Γ μια γλώσσα περιορισμών ορισμένη σε ένα πεπερασμένο σύνολο A, και έστω $\Gamma_{CON} = \{\{(a)\} | a \in A\}$ να είναι το σύνολο όλων των μοναδιαίων με ένα στοιχείο σχέσεων στο A.

Αν όλοι οι μοναδιαίοι πολυμορφισμοί του Γ είναι μεταθέσεις, τότε το Γ είναι tractable αν και μόνο αν το $\Gamma \cup \Gamma_{CON}$ είναι tractable, και το ίδιο για τον χαρακτηρισμό NP-complete. ■

Το παραπάνω πόρισμά έχει ως ενδιαφέρουσα συνέπεια τη σύνδεση των προβλημάτων απόφασης με τα προβλήματα εύρεσης λύσης. Έχουμε περιγράψει το CSP ως ένα πρόβλημα απόφασης, με ερώτημα αν υπάρχει μια λύση. Όμως το αντίστοιχο πρόβλημα εύρεσης λύσης, πολλές φορές εμφανίζεται πρακτικά. Θα δείξουμε τώρα ότι οι tractable περιπτώσεις των δύο προβλημάτων είναι ίδιες. (Παρατηρούμε οτι οι tractable περιπτώσεις του προβλήματος εύρεσης λύσης είναι αυτές που ανήκουν στην κλάση FP.)

ΠΟΡΙΣΜΑ 3.4.2.2

Ένα πρόβλημα απόφασης $CSP(G)$ είναι tractable αν και μόνο αν το αντίστοιχο πρόβλημα εύρεσης λύσης μπορεί να λυθεί σε πολυωνυμικό χρόνο. ■

ΑΠΟΔΕΙΞΗ

Προφανώς, αν το πρόβλημα εύρεσης λύσης είναι πολυωνυμικό τότε και το πρόβλημα απόφασης είναι.

Για το αντίστροφο, έστω Γ ένα tractable σύνολο σχέσεων ορισμένων σε ένα πεπερασμένο σύνολο A. Επιλέγοντας ένα μοναδιαίο πολυμορφισμό f του Γ, του οποίου το πεδίο τιμών U έχει το μικρότερο πληθάριθμο, μπορούμε να πάρουμε το αντίστοιχο σύνολο σχέσεων $\Gamma' = f(\Gamma)$ ορισμένων στο U, έτσι ώστε κάθε μοναδιαίος πολυμορφισμός του Γ' να είναι μετάθεση. Από τα παραπάνω η Γ' είναι επίσης tractable.

Τώρα θεωρούμε μια είσοδο $\mathcal{P} = (V, U, C)$ του $CSP(\Gamma)$. Από την επιλογή του Γ, μπορούμε να αποφασίσουμε σε πολυωνυμικό χρόνο (με βάση το μέγεθος της $\mathcal{P}$) αν αυτή η είσοδος έχει λύση. Έστω ότι έχει. Τότε η είσοδος $\mathcal{P}' = (V, U, C')$, που αποκτούμε αν αντικαταστήσουμε κάθε σχέση περιορισμού ρ με την αντίστοιχη σχέση $f(\rho)$, έχει επίσης λύση, και μάλιστα κάθε λύση του $\mathcal{P}'$ είναι λύση και του $\mathcal{P}$.

Αφού το $\mathcal{P}'$ έχει λύση, συνεπάγεται ότι για κάθε $v \in V$ υπάρχει κάποιο $a \in A$ για το οποίο μπορούμε να προσθαίσουμε τον περιορισμό $((v), \{(a)\})$ και να έχουμε ακόμη επιλύσιμη είσοδο. Άρα, αν ασχοληθούμε με μία μεταβλητή κάθε φορά, και κάθε πιθανή τιμή $a \in A$ για αυτή τη μεταβλητή, μπορούμε να πάρουμε έναν περιορισμό για κάθε μεταβλητή κάθε φορά, και τελικά μια λύση για το $\mathcal{P}'$. Ελέγχοντας για επιλυσημότητα σε κάθε πιθανή τιμή όλων των μεταβλητών χρειάζεται να λύσουμε το πολύ $|V||U|$ φορές εισόδους του $CSP(\Gamma' \cup \Gamma_{CON})$, και έτσι μπορούμε να βρούμε σε πολυωνυμικό χρόνο σε σχέση με το $\mathcal{P}'$ μια λύση του. ■

3.4.3 Δομές στις άλγεβρες

Τα αποτελέσματα που παρουσιάζονται εδώ σχετίζουν την πολυπλοκότητα μια πεπερασμένης άλγεβρας με την πολυπλοκότητα των υποαλγεβρών και των ομομορφικών απεικονίσεων της [29, 13, 8]. Σε πολλές περιπτώσεις, μπορούμε να χρησιμοποιήσουμε αυτά τα αποτελέσματα για να ανάγουμε το πρόβλημα της ανάλυσης της πολυπλοκότητας μια άλγεβρας σε ένα παρόμοιο πρόβλημα που εμπεριέχει μια άλγεβρα με μικρότερο σύμπαν.

ΟΡΙΣΜΟΣ 3.4.3.1

Υποάλγεβρα: Έστω $\mathcal{A} = (A, F)$ μια άλγεβρα και B ένα υποσύνολο του A τέτοιο ώστε, για κάθε $f \in F$ και για οποιαδήποτε $b_1, \ldots, b_n \in B$, όπου n είναι η πληθικότητα της f, έχουμε ότι $f(b_1, \ldots, b_n) \in B$. Τότε η άλγεβρα $\mathcal{B} = (B, F|_B)$ καλείται υποάλγεβρα της $\mathcal{A}$, οπού $F|_B$ περιέχει τους περιορισμούς όλων των πράξεων του F στο B. Αν $B \neq A$, τότε το B λέγεται αυστηρή υποάλγεβρα. □

ΘΕΩΡΗΜΑ 3.4.3.1

Χαρακτηρισμός μέσω υποάλγεβρας: Έστω $\mathcal{A}$ μια πεπερασμένη άλγεβρα.

1. Αν $\mathcal{A}$ είναι tractable τότε κάθε υποάλγεβρα της είναι επίσης.
2. Αν η $\mathcal{A}$ έχει μια NP-complete υποάλγεβρα, είναι και η ίδια NP-complete.

■

ΑΠΟΔΕΙΞΗ

Έστω $\mathcal{B} = (B, F|_B)$ μια υποάλγεβρα του $\mathcal{A} = (A, F)$. Είναι εύκολο να ελέγξουμε αν $Inv(F|_B) \subseteq Inv(F)$. Άρα , $CSP(Inv(F|_B))$ μπορεί να αναχθεί στο $CSP(Inv(F))$ σε σταθερό χρόνο. Τα παραπάνω αποτελέσματα είναι άμεσες συνέπειες της παραπάνω αναγωγής. ■

ΟΡΙΣΜΟΣ 3.4.3.2

Ομομορφισμοί αλγεβρών: Έστω $\mathcal{A}_1 = (A_1, F_1)$ και $\mathcal{A}_2 = (A_2, F_2)$ τέτοιες ώστε $F_1 = \{f_i^1 | i \in I\}$ και $F_2 = \{f_i^2 | i \in I\}$, όπου τα f_i^1, g_i^2 είναι n_i-αδικά, για κάθε $i \in I$.

Μια απεικόνιση $\phi : A_1 \to A_2$ καλείται ομομορφισμός από το $\mathcal{A}_1$ στο $\mathcal{A}_2$ εάν:

$$\phi(f_i^1(a_1, \ldots, a_{n_i})) = f_i^2(\phi(a_1), \ldots, \phi(a_{n_i}))$$

ισχύει για κάθε $i \in I$ και για όλα τα $a_1, \ldots, a_{n_i} \in A_1$.

Αν η απεικόνιση είναι και επί, τότε η $\mathcal{A}_2$ λέγεται ομομορφική εικόνα της $\mathcal{A}_a$. □

ΘΕΩΡΗΜΑ 3.4.3.2

Χαρακτηρισμός μέσω ομομορφικών εικόνων: Έστω $\mathcal{A}$ μια πεπερασμένη άλγεβρα.

1. Εάν $\mathcal{A}$ είναι tractable, τότε κάθε ομομορφική εικόνα του $\mathcal{A}$ είναι.
2. Εάν η $\mathcal{A}$ έχει μια ομομορφική εικόνα που είναι NP-complete είναι και η ίδια NP-complete.

■

ΑΠΟΔΕΙΞΗ

Έστω $\mathcal{B} = (B, F_B)$ μια ομομορφική εικόνα του $\mathcal{A} = (A, F_A)$ και έστω ϕ ο αντίστοιχος ομομορφισμός. Θα δείξουμε ότι για κάθε πεπερασμένο $\Gamma \subseteq Inv(F_B)$, το $CSP(\Gamma)$ ανάγεται σε γραμμικό χρόνο στο $CSP(\Gamma')$ για κάποιο πεπερασμένο $\Gamma' \subseteq Inv(F_A)$.

Για $\rho \in Inv(F_B)$, θέτουμε $\phi^{-1}(\rho) = \{\overline{a}|\phi(\overline{a} \in \rho\}$. Είναι προφανές ότι η $\phi^{-1}(\rho)$ είναι αναλλοίωτη της ίδιας πολλαπλότητας με την ρ. Μπορεί άμεσα να αποδειχθεί οτι $\phi^{-1}(\rho) \in Inv(F_A)$. Έστω $\Gamma' = \{\phi^{-1}(\rho)|\rho \in \Gamma\}$. Τότε Γ' είναι ένα πεπερασμένο υποσύνολο του $Inv(F_A)$.

Παίρνουμε μια είσοδο $\mathcal{P} = (V, B, C)$ του $CSP(\Gamma)$ και κατασκευάζουμε μια είσοδο $\mathcal{P}' = (V, A, C')$ του $CSP(\Gamma')$ όπου $C' = \{(s, \phi^{-1}(\rho))|(s, \rho) \in C\}$.

Εάν ψ είναι μια λύση του $\mathcal{P}'$, τότε η $\phi\psi$ είναι λύση του $\mathcal{P}$. Όμοια, αν ξ είναι μια λύση του $\mathcal{P}$, τότε οποιαδήποτε συνάρτηση $\psi : V \to A$ τέτοια ώστε $\phi\psi(v) = \xi(v)$ για κάθε $v \in V$ είναι λύση της $\mathcal{P}'$. ■

Τώρα θα δώσουμε δύο παραδείγματα χρήσης των παραπάνω θεωρημάτων χαρακτηρισμού. Τα παραδείγματα δείχνουν οτι και τα δύο αποτελέσματα μπορούν να είναι χρήσιμα (ανεξάρτητα) για να χαρακτηρίσουν την πολυπλοκότητα συγκεκριμένων αλγεβρών ανάγοντας το πρόβλημα σε μία άλγεβρα ορισμένη σε μικρότερο σύνολο.

ΠΑΡΑΔΕΙΓΜΑ 3.4.3.1

Χρήση αλγεβρικών δομών για NP-completeness: Έστω $\mathcal{A}$ να είναι η idempotent άλγεβρα $(\{0, 1, 2\}, \circ)$, όπου $\circ$ είναι η ακόλουθη σχέση:

$\circ$	0	1	2
0	0	1	1
1	1	1	0
2	2	2	2

Χρησιμοποιώντας το θεώρημα για τις ομομορφικές εικόνες, θα δείξουμε ότι η $\mathcal{A}$ είναι NP-complete ακόμη και αν όλες οι υποάλγεβρές του είναι tractable.

Παρατηρούμε οτι, οι ισότητες $0 \circ 2 = 1$, $1 \circ 2 = 0$, $0 \circ 1 = 1 \circ 0 = 1$ δείχνουν, ότι η $\mathcal{A}$ έχει μόνο μια αυστηρή υποάλγεβρα η οποία έχει μόνο ένα στοιχείο, την άλγεβρα $\mathcal{B} = (\{0, 1\}, \circ|_{\{0,1\}})$. Είναι εύκολο να δείξουμε ότι η $\circ|_{\{0,1\}}$ είναι μια semilattice πράξη στο $\{0, 1\}$. Έτσι, η άλγεβρα $\mathcal{B}$ είναι tractable.

Από την άλλη, ας θεωρήσουμε την άλγεβρα $\mathcal{C} = (C, *)$, όπου $C = \{a, b\}$ και για όλα τα $x, y \in \{a, b\}$, $x * y = x$. Είναι εύκολο να δούμε ότι η απεικόνιση $\phi : \{0, 1, 2\} \to C$ τέτοια ώστε $\phi(0) = \phi(1) = a$, και $\phi(2) = b$, είναι ομομορφισμοί από το $\mathcal{A}$ στο $\mathcal{C}$. Η $\mathcal{C}$ είναι NP-complete,(όλες οι βασικές πράξεις είναι essentially unary και είμαστε στο $\{0, 1\}$). Και άρα η $\mathcal{A}$ είναι NP-complete. □

ΠΑΡΑΔΕΙΓΜΑ 3.4.3.2

Χρήση αλγεβρικών δομών για NP-completeness: Έστω $\mathcal{A}$ να είναι η idempotent άλγεβρα $(\{0, 1, 2\}, \circ)$, όπου $\circ$ είναι η ακόλουθη σχέση:

$\circ$	0	1	2
0	0	1	1
1	0	1	1
2	1	1	2

Χρησιμοποιώντας το θεώρημα χαρακτηρισμού μέσω υποαλγεβρών θα δείξουμε ότι η $\mathcal{A}$ είναι NP-complete ακόμη και αν όλες οι μικρότερες ομομορφικές απεικονίσεις της είναι tractable.

Αφού οι άλγεβρες με μόνο ένα στοιχείο είναι σίγουρα tractable, αρκεί να ασχοληθούμε με τις ομομορφικές εικόνες $\mathcal{B}$ της $\mathcal{A}$ που έχουν δύο στοιχεία. Έστω $\mathcal{B} = (\{a, b\}, *)$ όπου $*$ είναι μια δυαδική πράξη στο $\{a, b\}$ και υποθέτουμε ότι η ϕ είναι ένας ομομορφισμός από το $\mathcal{A}$ στο $\mathcal{B}$. Τότε έχουμε $\phi(x \circ y) = \phi(x) \circ \phi(y)$ για όλα τα $x, y \in \{0, 1, 2\}$.

Έχουμε τις εξής περιπτώσεις τώρα:

1. $\phi(0) = \phi(1) = a$, $\phi(2) = b$. Σε αυτή την περίπτωση έχουμε:

$$a * a = \phi(0) * \phi(0) = \phi(0 \circ 0) = \phi(0) = a$$

$$a * b = \phi(0) * \phi(2) = \phi(0 \circ 2) = \phi(1) = a$$

$$b * a = \phi(2) * \phi(0) = \phi(2 \circ 0) = \phi(1) = a$$

$$b * b = \phi(2) * \phi(2) = \phi(2 \circ 2) = \phi(2) = b$$

   Είναι εύκολο να δούμε ότι η $*$ είναι μια semilattice πράξη στο $\{a, b\}$. Άρα η $\mathcal{B}$ είναι tractable.

2. Έστω $\phi(0) \neq \phi(1)$. Τότε έχουμε οτι $\phi(2) \in \{\phi(0), \phi(1)\}$. Εάν $\phi(2) = \phi(0)$, τότε:

$$\phi(0) = \phi(0 \circ 0) = \phi(0) * \phi(0) = \phi(0) * \phi(2) = \phi(0 \circ 2) = \phi(1).$$

   Ενώ αν $\phi(2) = \phi(1)$, τότε:

$$\phi(0) = \phi(1 \circ 0) = \phi(1) * \phi(0) = \phi(2) * \phi(0) = \phi(2 \circ 0) = \phi(1)$$

Έτσι η δεύτερη περίπτωση είναι αδύνατη, και αποδείξαμε ότι όλες οι μικρότερες ομομορφικές εικόνες του $\mathcal{A}$ είναι tractable.

Από την άλλη, η άλγεβρα $\mathcal{A}$ έχει την υποάλγεβρα $\mathcal{A}' = (\{0, 1\}, \circ)$ τέτοια ώστε $x \circ y = y$ για όλα τα $x, y \in \{0, 1\}$. Όμως η $\mathcal{A}'$ είναι NP-complete (όλες οι βασικές πράξεις είναι essentially unary και είμαστε στο $\{0, 1\}$) και άρα και η $\mathcal{A}$ είναι NP-complete. □

3.4.4 Αποτελέσματα

Η προηγούμενη υποενότητα έδειξε ότι μια tractable άλγεβρα πρέπει να έχει την ιδιότητα ότι όλες οι υποάλγεβρες και οι ομομορφικές εικόνες της είναι tractable. Οπότε, μια πρώτη ερώτηση που έρχεται φυσικά είναι αν θα μπορούσαμε να χαρακτηρίσουμε την πολυπλοκότητα όλων των αλγεβρών που δεν έχουν μικρότερες υποάλγεβρες και ομομορφικές εικόνες. Αν μπορούσαμε να χαρακτηρίσουμε όλες αυτές τις άλγεβρες θα μπορούσε να είναι μια βάση αναγωγής για οποιοδήποτε άλλο πρόβλημα.

ΟΡΙΣΜΟΣ 3.4.4.1

Απλή άλγεβρα, αυστηρά απλή άλγεβρα: Μια πεπερασμένη άλγεβρα θα λέγεται απλή αν όλες οι μικρότερες ομομορφικές εικόνες της έχουν μόνο ένα στοιχείο, και αυστηρά απλή αν είναι απλή και όλες οι αυστηρές υποάλγεβρες της έχουν επίσης ένα στοιχείο. □

Τελικά για αυτές τις άλγεβρες αποδεικνύεται το εξής:

ΘΕΩΡΗΜΑ 3.4.4.1

Χαρακτηρισμός απλών επί αλγεβρών: Μια πεπερασμένη απλή επί άλγεβρα είναι NP-complete αν όλες οι πράξεις όροι είναι essentially unary. Αλλιώς είναι tractable. ■

Ένας τέτοιος χαρακτηρισμός για όλες τις άλγεβρες θα σήμαινε πρακτικά ότι το πρόβλημ λύθηκε. Δυστυχώς κάτι τέτοιο δεν ισχύει όμως φαίνεται ότι ισχύει το ακόλουθο:

Υπόθεση: Μια πεπερασμένη idempotent άλγεβρα $\mathcal{A}$ είναι NP-complete αν κάποια υποάλγεβρά έχει ομομορφική άλγεβρα της οποίας όλες οι πράξεις είναι Αλλιώς είναι tractable.

Είναι σημαντικό ότι για όλες τις ήδη χαρακτηρισμένες κλάσεις που δοκιμάστηκαν η απλή αυτή ιδιότητα ισχύει (Συμπεριλαμβανομένων των αποτελεσμάτων του Schaefer του 1978 αλλά και του Bulatov 2002). Μάλιστα αν τελικά αποδειχθεί η υπόθεση και αφού έχει ήδη αποδείχθει ότι κάθε άλγεβρα μπορεί να αναχθεί σε idempotent ίσως φτάσουμε στο τέλος του δρόμου του χαρακτηρισμού.

Ένα άλλο σημαντικό αποτέλεσμα που απορέει από αυτή την υπόθεση είναι ότι μπορούμε σε πολυωνυμικό χρόνο σε σχέση με το Γ να αποφασίσουμε αν είναι στο P ή NP-complete μηχανιστικά. Πράγμα εξαιρετικά χρήσιμο και ενδιαφέρον από θεωρητικής πλευράς. Δηλαδή το να μπορείς σε πολυωνυμικό χρόνο να αποφασίζεις αν ένα πρόβλημα μπορεί να λυθεί σε πολυωνυμικό χρόνο ή χρειάζεσαι αναγκαστικά να πας στο NP.

Κεφάλαιο 4

Θεώρημα Schaefer

4.1 Πλήρης απόδειξη

Σ' αυτό το κεφάλαιο θα παρουσιαστούν οι αυτούσιες αποδείξεις από τα paper για δύο λόγους. Πρώτον προφανώς την κατανόησή τους, και δεύτερον για να γίνει εμφανής η διαφοροποίηση μεταξύ των μαθηματικών εργαλείων που χρησιμοποιούνται.

4.1.1 Εισαγωγή

Το CSP όταν οι μεταβλητές παίρνουν τιμές από ένα σύνολο δύο στοιχείων έχει μελετηθεί πλήρως ως προς την tractability ήδη από το 1978 από τον Thomas J. Schaefer. Δηλαδή όταν το A έχει πληθάριθμο 2 το $CSP(\Gamma)$ γνωρίζουμε αν είναι P ή NP-complete. Παρακάτω γίνεται παρουσίαση της απόδειξης. Αξίζει να σημειωθεί ότι το πρόβλημα στο paper του Schaefer φαίνεται σαν μια γενίκευση του satisfiability problem μια και ο ορισμός και το ενδιαφέρον για το ακόμα πιο γενικό CSP προήλθε πιο μετά (αφού η αρχή ήταν αυτό το paper). Η παρουσίαση παρακάτω θα γίνει με την αρχική σημειολογία, και όπου κριθεί αναγκαίο θα εκφραστεί και στη σημερινή γλώσσα του CSP. Παρακάτω παρουσιάζονται οι συμβολισμοί που θα χρησιμοποιηθούν στην απόδειξη:

ΟΡΙΣΜΟΣ 4.1.1.1

S-τύπος, S-ικανοποιησιμότητα, $SAT(S)$: Έστω $S = \{R_1, \ldots R_m\}$ ένα πεπερασμένο σύνολο από λογικές σχέσεις. (Μια λογική σχέση ορίζεται ως ένα υποσύνολο του $\{0,1\}^{\kappa}$). Έστω S-τύπος να είναι μια conjunction από προτάσεις της μορφής $R_i(\xi_1, \xi_2, \ldots.)$, οπού $\xi_1, \xi_2, \ldots$ είναι μεταβλητές, ο αριθμός των οποίων είναι η πολλαπλότητα της R_i, $i \in (1, \ldots, m)$. Το να αποφασίσεις αν ένας S-τύπος είναι ικανοποιήσιμος είναι το πρόβλημα S-ικανοποιησιμότητας. Συμβολίζουμε με $SAT(S)$ το σύνολο όλων των ικανοποιήσιμων S-τύπων. $\square$

ΟΡΙΣΜΟΣ 4.1.1.2

$(0-1)$-**ικανοποιήσιμη, ασθενώς θετική-αρνητική, bijunctive,affine:**

- Μια σχέση R είναι 0-ικανοποιήσιμη (1-ικανοποιήσιμη) αν το $(0,\ldots 0) \in R$ $((1,\ldots 1) \in R)$.
- Μια σχέση είναι ασθενώς θετική (αρνητική) αν η $R(x_1,\ldots)$ είναι λογικά ισοδύναμη με ένα CNF τύπο ο οποίος έχει το πολύ μια μεταβλητή με άρνηση (χωρίς άρνηση) σε κάθε conjunct.
- Μια σχέση R είναι bijunctive αν η $R(x_1,\ldots)$ είναι λογικά ισοδύναμη με ένα CNF τύπο που έχει 2 μεταβλητές σε κάθε conjunct.
- Μια σχέση R είναι affine αν $R(x_1,\ldots)$ είναι λογικά ισοδύναμη με ένα σύστημα γραμμικών εξισώσεων στο δισύνολο $\{0,1\}$, δηλαδή αν $R(x_1,\ldots)$ είναι λογικά ισοδύναμο με μια conjunction από τύπους της μορφής $\xi_1 \oplus \ldots \oplus \xi_n = 0$ και $\xi_1 \oplus \ldots \oplus \xi_n = 1$ με $\oplus$ να συμβολίζει την πρόσθεση mod 2.

□

ΟΡΙΣΜΟΣ 4.1.1.3

Τύποι: Με τον όρο τύπος θα εννοούμε ένα καλώς ορισμένο τύπο από μεταβλητές, σταθερές, λογικούς συνδέσμους, παρενθέσεις, σύμβολα λογικών σχέσεων και υπαρξιακούς και καθολικούς ποσοδείκτες. Η ιδέα είναι να μπορούμε να χρησιμοποιήσουμε οποιοδήποτε συμβολισμό μας είναι χρήσιμος για την έκφραση σχέσεων μεταξύ προτασιακών μεταβλητών.

- Μεταβλητή θα θεωρήσουμε ένα στοιχείο από το σύνολο $\{x, x_0, x_1 \ldots, y, y_0\ ,\ldots,\ z, \ldots u\ ,\ldots v,\ \ldots\}$.Οι μεταβλητές όπως και οι τύποι είναι αλφαριθμητικά (πχ το x_{18} είναι ένα αλφαριθμητικό τριών συμβόλων).
- Σταθερά είναι ένα από τα σύμβολα $0, 1$ (1=αληθές, 0 =ψευδές).
- Λογικός σύνδεσμος είναι ένα από τα σύμβολα $\vee, \wedge,\ \neg, \rightarrow,\ \equiv, \neq$ με τις συνήθεις σημασίες τους
- Κάθε λογική σχέση R έχει το αντίστοιχο σύμβολο λογικής σχέσης
- Οι ποσοδείκτες $\exists x, \forall x$ έχουν τη συνήθη σημασία.
- Literal είναι μια μεταβλητή η μια μεταβλητή με άρνηση.

□

ΟΡΙΣΜΟΣ 4.1.1.4

$Var(A)$**:** Αν A είναι ένας τύπος, τότε με $Var(A)$ συμβολίζουμε το σύνολο των ελεύθερων (που δεν έχουν ποσοδειχθεί) μεταβλητών που εμφανίζονται στο A. Μια ανάθεση στο A είναι μια συνάρτηση $s : Var(A) \to \{0,1\}$. Λέμε ότι η ανάθεση s ικανοποιεί το A αν η s κάνει το A αληθές. $\square$

ΟΡΙΣΜΟΣ 4.1.1.5

$Sat(A)$: $Sat(A)$ είναι το σύνολο όλων των αναθέσεων s που ικανοποιούν το A. Δύο τύποι A και B είναι λογικά ισοδύναμοι αν $Var(A) = Var(B)$ και $Sat(A) = Sat(B)$. $\square$

ΟΡΙΣΜΟΣ 4.1.1.6

$K^A_{i,V}$: Έστω Α τύπος με $V \subseteq Var(A)$, και $i \in \{0,1\}$. Τότε $K^A_{i,V}$ είναι η ανάθεση $s : Var(A) \to \{0,1\}$ που ορίζεται ως $s(\xi) = i$ αν και μόνο αν $\xi \in V$. Συνήθως γράφουμε απλώς $K_{i,V}$ με το A να βγαίνει από τα συμφραζόμενα. Το K_i συμβολίζει την ανάθεση που έχει την σταθερή τιμή i (και πάλι το A βγαίνει από τα συμφραζόμενα). $\square$

ΟΡΙΣΜΟΣ 4.1.1.7

$A[^{\xi}_{w}]$: Αν A ένας τύπος, ξ μια μεταβλητή, και w είναι ένα literal, τότε $A[^{\xi}_{w}]$ συμβολίζει τον τύπο που σχηματίζεται από το A αν αντικαταστήσουμε κάθε εμφάνιση της ξ με το w. Αν V είναι ένα σύνολο μεταβλητών, τότε $A[^{V}_{w}]$ συμβολίζει το αποτέλεσμα της αντικατάστασης με w κάθε εμφάνισης μεταβλητής του V. Πολλαπλές αντικαταστάσεις συμβολίζονται με εκφράσεις όπως $A[^{V}_{w},^{V'}_{w'},^{V''}_{w''}]$ και με νόημα προφανές. $\square$

ΟΡΙΣΜΟΣ 4.1.1.8

$Gen(S)$: Το σύνολο των υπαρξιακά ποσοδεικτούμενων S-τύπων με σταθερές συμβολίζεται με $Gen(S)$. Συγκεκριμένα, $Gen(S)$ είναι το μικρότερο σύνολο τύπων τέτοιων ώστε:

- $\forall R \in S, R(x_1, \ldots) \in Gen(S)$.
- $\forall A, B \in Gen(S)$ και όλες τις μεταβλητές ξ, η τα παρακάτω είναι όλα στο $Gen(S) : A \wedge B, A[^{\xi}_{\eta}], A[^{\xi}_{0}], A[^{\xi}_{1}]$ και $(\exists \xi)A$.

Με $Gen^+(S)$ συμβολίζουμε το σύνολο όλων των τύπων που είναι λογικά ισοδύναμοι με κάποιο τύπο του $Gen(S)$. $\square$

ΟΡΙΣΜΟΣ 4.1.1.9

$[A]$: Αν A είναι ένα τύπος τότε με $[A]$ συμβολίζουμε τη λογική σχέση που ορίζει η A, όταν οι μεταβλητές παίρνονται σε λεξικογραφική σειρά. $\square$

ΟΡΙΣΜΟΣ 4.1.1.10

$Rep(S)$: Τέλος συμβολίζουμε με $Rep(S) := \{[A] : A \in Gen(S)\}$. Δηλαδή $Rep(S)$ είναι το σύνολο των σχέσεων που μπορούν να εκφραστούν με ποσοδεικτούμενους S-τύπους με σταθερές. Παρατηρούμε ότι αν $S \subseteq S'$, τότε $Rep(S) \subseteq Rep(S')$. $\square$

4.1.2 Θεώρημα κλάσεων του $Rep(S)$

ΘΕΩΡΗΜΑ 4.1.2.1

Θεώρημα κλάσεων του $Rep(S)$: Έστω S ένα οποιοδήποτε σύνολο λογικών σχέσεων. Αν το S ικανοποιεί έναν από τους παρακάτω περιορισμούς $(a) - (d)$ παρακάτω τότε το $Rep(S)$ ικανοποιεί τον ίδιο περιορισμό. Αλλιώς το $Rep(S)$ είναι το σύνολο όλων των λογικών σχέσεων.

(a) Κάθε σχέση στο S είναι ασθενώς αρνητική

(b) Κάθε σχέση στο S είναι ασθενώς θετική

(c) Κάθε σχέση στο S είναι affine.

(d) Κάθε σχέση στο S είναι bijunctive.

■

Το υπόλοιπο της υποενότητας είναι η απόδειξη αυτού του θεωρήματος.

ΛΗΜΜΑ 4.1.2.1

Έστω R μια λογική σχέση και $A := R(x_1, \ldots)$. Τότε η R είναι affine αν και μόνο αν για κάθε $s_1, s_2, s_3 \in Sat(A), s_1 \oplus s_2 \oplus s_3 \in Sat(A)$. ■

ΑΠΟΔΕΙΞΗ

Χρησιμοποιούμε το ακόλουθο γεγονός, που αποδεικνύεται με απλή γραμμική άλγεβρα. Αν K είναι ένα field, τότε το $D \subseteq K^n$ είναι σύνολο λύσεων ενός συστήματος γραμμικών εξισώσεων στο K αν και μόνο αν $\forall b_1, b_2, b_3 \in D$ και $\forall c_1, c_2, c_3 \in K$ με $c_1 + c_2 + c_3 = 1$, τότε $c_1b_1 + c_2b_2 + c_3b_3 \in D$. Σε περίπτωση που το K είναι το $\{0, 1\}$, αυτή η συνθήκη είναι ισοδύναμη με "το άθροισμα οποιωνδήποτε τριών στοιχείων του D είναι στοιχείο του D." Αφού η R είναι affine αν και μόνο εάν A είναι ισοδύναμη με ένα σύστημα γραμμικών εξισώσεων στο $\{0, 1\}$, το λήμμα συνεπάγεται από αυτό το γεγονός. Επίσης αποδεικνύεται ότι ο πληθικός αριθμός μιας affine σχέσης είναι δύναμη του 2 που μπορεί να χρησιμοποιηθεί για να αποδείξει ότι κάποια σχέση δεν είναι affine. ■

Τώρα θα εισάγουμε συμβολισμό για το επόμενο λήμμα: Αν ξ είναι μια μεταβλητή, χρησιμοποιούμε τον συμβολισμό (ξ, i) για να δηλώσουμε το literal ξ αν $i = 1$ και $\neg\xi$ αν i=0. Όπως συνηθίζεται, το $\overline{\alpha}$ δηλώνει το συμπληρωματικό literal του α, δηλαδή το literal $(\xi, 1 - i)$ όταν $\alpha = (\xi, i)$. Λέμε ότι το literal $\alpha = (\xi, i)$ είναι συμβατό με ένα τύπο A αν $s(\xi) = i$ για κάποιο $s \in Sat(A)$. Λέμε οτι η ανάθεση s συμφωνεί με το literal α αν $\alpha = (\xi, s(\xi))$ για κάποια μεταβλητή ξ. Ένα σύνολο από literals είναι συμβατό αν δεν συμπεριλαμβάνει τα α και $\overline{\alpha}$ για κάποιο literal α. Αν s είναι μια ανάθεση και Q είναι ένα συνεπές σύνολο από literal, συμβολίζουμε με $s\#Q$ την ανάθεση η οποία διαφέρει από το s μόνο στο σύνολο $\{\xi : (\xi, 1 - s(\xi)) \in Q\}$.

ΟΡΙΣΜΟΣ 4.1.2.1

$Imp_A(\alpha)$: Έστω α ένα literal και A ένας τύπος. Ορίζουμε $Imp_A(\alpha)$ να είναι το σύνολο όλων των literals β τέτοιων ώστε κάθε $s \in Sat(A)$ που συμφωνεί με το α συμφωνεί επίσης και με το β. Έτσι, το $Imp_A(\alpha)$ είναι το σύνολο των literal που "επάγονται" από το literal α. Για παράδειγμα, αν $A = x \vee y$, τότε $(y, 1) \in Imp_A((x, 0))$. □

ΟΡΙΣΜΟΣ 4.1.2.2

Σύνολο αλλαγής: Έστω A ένας τύπος και $s \in Sat(A)$. Ένα σύνολο αλλαγής για το (A, s) είναι ένα σύνολο $V \subseteq Var(A)$ τέτοιο ώστε $s \oplus K_{1,V} \in Sat(A)$. Δηλαδή, V είναι ένα σύνολο αλλαγών για το (A,s) αν και μόνο αν η ανάθεση που διαφέρει από το s μόνο στο V ικανοποιεί το A. □

ΛΗΜΜΑ 4.1.2.2

Έστω R μια λογική σχέση και $A := R(x_1, \ldots)$. Τότε τα παρακάτω είναι ισοδύναμα:

(a) R είναι bijunctive

(b) Για κάθε $s \in Sat(A)$, αν V_1 και V_2 είναι σύνολα αλλαγής για το (A,s) τότε είναι και το $V_1 \cap V_2$

(c) Για κάθε $s \in Sat(A)$ και κάθε literal α το οποίο είναι συμβατό με το A, $s\#Imp_A(\alpha) \in Sat(A)$.

■

ΑΠΟΔΕΙΞΗ

$(a) \rightarrow (b)$: Έστω ότι η R είναι bijunctive. Έτσι, το A είναι λογικά ισοδύναμο με κάποιο τύπο B ο οποίος είναι conjunction από προτάσεις της μορφής $(\alpha \rightarrow \beta)$, με α, β literal. Έστω $s \in Sat(A)$ δοσμένο και έστω V_1, V_2 σύνολα αλλαγής για το (A, s). Έστω Q το μικρότερο σύνολο literal τέτοιο ώστε:

- $\{(\eta, 1 - s(\eta)) : \eta \in V_1 \cap V_2\} \in Q$.
- όταν $\alpha \in Q$ και $(\alpha \rightarrow \beta)$ ή $(\overline{\beta} \rightarrow \overline{\alpha})$ είναι ένα conjunct του B για κάποιο literal β, τότε $\beta \in Q$.

Προφανώς κάθε ανάθεση $t \in Sat(B)$ η οποία διαφέρει από το s σε όλα τα $V_1 \cap V_2$ πρέπει να συμφωνεί με κάθε literal στο Q. Εφόσον $s \oplus K_{1,V_1}$ είναι μια τέτοια ανάθεση, το Q είναι συμβατό και δεν μπορεί να περιέχει literal $(\eta, 1 - s(\eta))$ με $\eta \notin V_2$. Οπότε, $s\#Q = s \oplus K_{1,V_1 \cap V_2}$. Είναι εύκολο να δείξουμε οτι $s\#Q$ ικανοποιεί κάθε conjunct του B, και άρα $s\#Q = s \oplus K_{1,V_1 \cap V_2} \in Sat(B) = Sat(A)$. Άρα $V_1 \cap V_2$ είναι ένα σύνολο αλλαγής για το (A, s).

$(b) \rightarrow (c)$: Έστω ότι το (b) ισχύει. Έστω $s \in Sat(A)$, και έστω α να είναι ένα literal συμβατό με το A. Θέλουμε να δείξουμε οτι $s\#Imp_A(\alpha) \in Sat(A)$.

Έστω οτι το s δεν είναι συμβατό με το α, δηλαδή $\alpha = (\xi, 1 - s(\xi))$ για κάποια μεταβλητή ξ. Έστω $W := \{\eta : (\eta, 1 - s(\eta)) \in Imp_A(\alpha)\}$ δηλαδή, W είναι το σύνολο όλων των μεταβλητών στις οποίες το $Imp_A(\alpha)$ διαφωνεί με το s. Ισχυριζόμαστε ότι $W = \cap\{V : V$ είναι ένα σύνολο αλλαγών για το (A, s) και $\xi \in V\}$. Προσέχουμε οτι κάθε σύνολο αλλαγών που περιέχει το ξ πρέπει επίσης να περιέχει και όλο το W, αφού το W αποτελείται από όλες τις μεταβλητές που αναγκαστικά πρέπει να αλλάξουν αν αλλάξει το ξ. Από την άλλη, αν κάποια μεταβλητή η δεν περιέχεται στο W, τότε υπάρχει κάποια ανάθεση t ώστε $t(\xi) \neq s(\xi)$ αλλά $t(\eta) = s(\eta)$, έτσι ώστε το η δεν περιέχεται στο σύνολο αλλαγής $\{\zeta : t(\zeta) \neq s(\zeta)\}$. Αυτό αποδεικνύει τον ισχυρισμό. Τώρα με αλλεπάλληλες χρήσεις της υπόθεσης (b), το W είναι ένα σύνολο αλλαγής για το (A, s). Έτσι, $s \oplus K_{1,W} = s\#Imp_A(\alpha) \in Sat(A)$.

$(c) \rightarrow (a)$: Υποθέτουμε οτι για κάθε $s \in Sat(A)$ και κάθε literal α το οποίο είναι συμβατό με το A, $s\#Imp_A(a) \in Sat(A)$. Ορίζουμε B να είναι η conjunction των $\{(\alpha \rightarrow \beta) : \alpha, \beta$ literal και $\beta \in Imp_A(\alpha)\}$. Σημειώνουμε οτι $Var(B) = Var(A)$, αφού το B έχει το conjunct $(\xi \rightarrow \xi)$ για κάθε $\xi \in Var(A)$. Ισχυριζόμαστε οτι το B είναι λογικά ισοδύναμο με το A, και άρα η R είναι bijunctive. Πρέπει να δείξουμε ότι $Sat(B) = Sat(A)$. Προφανώς $Sat(A) \subseteq Sat(B)$, αφού κάθε ανάθεση που ικανοποιεί το A πρέπει να ικανοποιεί και κάθε conjunct του B. Απομένει να δείξουμε οτι $Sat(B) \subseteq Sat(A)$. Έστω ότι δεν ισχύει, δηλαδή υπάρχει $s_1 \in Sat(B) - Sat(A)$. Επιλέγουμε $s_2 \in Sat(A)$ ώστε $|W|$ είναι μέγιστο, με $W = \{\eta : s_1(\eta) = s_2(\eta)\}$. Επιλέγουμε $\xi \in Var(A) - W$, και έστω $\alpha := (\xi, s_1(\xi))$. Το literal α είναι συμβατό με το A, γιατί αν το α ήταν ασύμβατο με το A, το B θα είχε ένα conjunct που θα επιβεβαίωνε το γεγονός(δηλαδή, αν $\alpha = (\xi, 0)$ είναι ασύμβατο με το A, τότε το B έχει ένα conjunct $(\neg\xi \rightarrow \xi)$, και αν $\alpha = (\xi, 1)$ είναι ασύμβατο με το A, τότε το B έχει ένα conjunct $(\xi \rightarrow \neg\xi)$), και αυτό θα ανάγκαζε $s_2(\xi) = s_1(\xi)$. Έστω $s_3 := s_2\#Imp_A(\alpha)$. από υπόθεση το s_3 ικανοποιεί το A. Υποστηρίζουμε ότι για κάθε $\eta \in W, s_3(\eta) = s_2(\eta)$. Για να το δούμε αυτό, υποθέτουμε $\eta \in W$ με $s_3(\eta) \neq s_2(\eta)$. Αφού τώρα $(\eta, 1 - s_2(\eta)) \in Imp_A(\alpha)$, το B θα έχει ένα conjunct $(\alpha \rightarrow (\eta, 1 - s_2(\eta)))$, ή ισοδύναμα $((\xi, s_1(\xi)) \rightarrow (\eta, 1 - s_1(\eta)))$. Αυτό το conjunct δεν ικανοποιείται από το s_1, αναιρώντας την υπόθεση οτι το s_1 ικανοποιεί το B. Αυτό αποδεικνύει τον ισχυρισμό. Έτσι το s_3 συμφωνεί με το s_1 σε όλα τα $W \cup \xi$. το οποίο αναιρεί την υπόθεση ότι το s_2 μεγιστοποιεί το $|W|$. Και με αυτήν την εις άτοπο απαγωγή τελειώνει η απόδειξη. ■

ΟΡΙΣΜΟΣ 4.1.2.3

***i*-κλείσιμο:** Έστω A ένας τύπος, έστω $i \in \{0, 1\}$, και έστω $V \subseteq Var(A)$. Ορίζουμε το i-κλείσιμο του V σε σχέση με το A να είναι το σύνολο $Cl_{i,A}(V)$ $:= \{\xi \in Var(A) : \forall s \in Sat(A)$ τέτοιο ώστε $s|V \equiv i, s(\xi) \equiv i\}$. Δηλαδή $Cl_{0,A}(V)$ $(Cl_{1,A}(V))$ είναι το σύνολο όλων των μεταβλητών που αναγκαστικά γίνονται ψευδείς (αληθείς) όταν όλες οι μεταβλητές του V γίνονται ψευδείς (αληθείς). Είναι εύκολο να δούμε οτι $V \subseteq Cl_{i,A}(V)$, και οτι $V \subseteq V'$

συνεπάγεται $Cl_{i,A}(V) \subseteq Cl_{i,A}(V')$, για κάθε $V, V' \subseteq Var(A), i \in \{0,1\}$. Καλούμε το σύνολο $V \subseteq Var(A)$ i-κλειστό στο A αν $V = Cl_{i,A}(V)$. Επίσης καλούμε το V i-συμβατό στο A αν υπάρχει κάποιο $s \in Sat(A)$ τέτοιο ώστε $s|V \equiv i$. Λέμε οτι το V είναι i-μηκλειστό (i-ασύμβατο) στο A αν το V δεν είναι i-κλειστό (i-συμβατό) στο A. □

ΛΗΜΜΑ 4.1.2.3

Έστω R μια λογική σχέση και $A := R(x_1, \ldots)$. Τότε

- Η R είναι ασθενώς θετική αν και μόνο αν όταν $V \subseteq Var(A)$ είναι 0-συμβατή και 0-κλειστή για το A, $K_{0,V} \in Sat(A)$
- R είναι ασθενώς αρνητική αν και μόνο αν όταν $V \subseteq Var(A)$ είναι 1-συμβατή και 1-κλειστή για $A, K_{1,V} \in Sat(A)$.

■

ΑΠΟΔΕΙΞΗ

Πρέπει να αποδείξουμε το πρώτο μέρος. Η απόδειξη του 2ού βγαίνει όμοια. Αν R είναι κενό, τότε το λήμμα ισχύει προφανώς, οπότε θεωρούμε R μη κενό.

($\Rightarrow$): Έστω ότι το R είναι ασθενώς θετικό. Έτσι το A είναι λογικά ισοδύναμο με κάποιο CNF τύπο A' ο οποίος έχει το πολύ μια μεταβλητή με άρνηση σε κάθε conjunct. Αρκεί να δείξουμε ότι αν $V \subseteq Var(A')$ είναι 0-συμβατό και 0-κλειστό για το A', τότε $K_{0,V} \in Sat(A')$. Έστω V είναι ένα τέτοιο σύνολο και υποθέτουμε ότι $K_{0,V} \notin Sat(A')$. Έστω C ένα conjunct της A' στο οποίο η $K_{0,V}$ αποτυγχάνει. Έστω U το σύνολο από μεταβλητές χωρίς άρνηση του C. Εφόσον $K_{0,V}$ αποτυγχάνει στο C, $U \subseteq V$. Αν το C δεν έχει μεταβλητές με άρνηση, τότε δεν ισχύει ότι το V είναι 0-συμβατό για το A'. Αλλιώς, έστω η μόνη μεταβλητή με άρνηση στο C. Είναι προφανές οτι $\eta \in Cl_{0,A'}(U)$. Επίσης, αφού $K_{0,V}$ αποτυγχάνει στο C, $\eta \notin V$. Αλλά τότε η V δεν είναι 0-κλειστή στο A'. Αυτό αποδεικνύει το γεγονός οτι $K_{0,V} \in Sat(A')$.

($\Leftarrow$): Υποθέτουμε $K_{0,V} \in Sat(A)$ για όλα τα 0-κλειστά και 0-συμβατά σύνολα $V \subseteq Var(A)$. Έστω A' να είναι το conjunction όλων των προτάσεων $\{(\xi_1 \vee \ldots \vee \xi_n) : \xi_1, \ldots, \xi_n$ είναι 0-μηικανοποιήσιμα για το A $\} \cup \{(\neg\eta \vee \xi_1 \vee \ldots \vee \xi_\nu) : \eta \in Cl_{0,A}(\{\xi_1, \ldots, \xi_\nu\})\}$. Εφόσον κάθε μεταβλητή περιέχεται στο ίδιο της το 0-κλείσιμο, A' έχει ένα conjunct $(\neg\xi \vee \xi)$ για κάθε $\xi \in Var(A)$, και άρα $Var(A') = Var(A)$. Ισχυριζόμαστε ότι το A' είναι λογικά ισοδύναμο με το A. Για να δείξουμε οτι $Sat(A) \subseteq Sat(A')$, υποθέτουμε οτι $s \notin Sat(A')$. Έστω C να είναι ένα conjunct της A' στο οποίο η s αποτυγχάνει. Αν το C είναι της μορφής $(\xi_1 \vee \ldots \vee \xi_n)$ τότε $s(\xi_1) = \ldots = s(\xi_n) = 0$ και από τον ορισμό του $A'\{\xi_1, \ldots, \xi_n\}$ είναι 0-μησυμβατό στο A, έτσι, $s \notin Sat(A)$. Αλλιώς το C είναι της μορφής $(\neg\eta \vee \xi_1 \vee \ldots \vee \xi_n)$, και άρα $s(\eta) = 1$, $s(\xi_1) = \ldots = s(\xi_n) = 0$. Τότε από ορισμό του A', $\eta \in Cl_{0,A}(\xi_1, \ldots, \xi_n)$, και άρα $s \notin Sat(A)$. Αυτό αποδεικνύει οτι $Sat(A) \subseteq Sat(A')$. Μετά δείχνουμε οτι $Sat(A') \subseteq Sat(A)$. Έστω $s \notin Sat(A)$. Έστω $V = \{\xi : s(\xi) = 0\}$. Από την ιδιότητα που έχουμε υποθέσει για την A, η V είναι είτε 0-μησυμβατή ή 0-μηκλειστή στο A. Αν είναι 0-μησυμβατή, τότε $(\xi_1 \vee \ldots \vee \xi_n)$ είναι conjunct της A', όπου $V = \{\xi_1, \ldots, \xi_\nu\}$, και άρα $s \notin Sat(A')$. Αν V είναι 0-μηκλειστή, έστω $\eta \in Cl_{0,A}(V) - V$. Τότε $(\neg\eta \vee \xi_1 \vee \ldots \vee \xi_n)$ είναι ένα conjunct της A', και άρα $s \notin Sat(A')$. Αυτό αποδεικνύει οτι $Sat(A') \subseteq Sat(A)$. Άρα $Sat(A) = Sat(A')$ και άρα A' είναι λογικά ισοδύναμο με το A. Και τελικά η R είναι ασθενώς θετική. ■

ΛΗΜΜΑ 4.1.2.4

Έστω R μια λογική σχέση. Αν R δεν είναι ασθενώς αρνητική, τότε $Rep(\{R\}) \cap \{[x \neq y], [x \vee y]\} \neq \emptyset$. Αν R δεν είναι ασθενώς θετική, τότε $Rep(\{R\}) \cap \{[x \neq y], [\neg x \vee \neg y]\} \neq \emptyset$. ■

ΑΠΟΔΕΙΞΗ

Έστω R μια λογική σχέση η οποία δεν είναι ασθενώς αρνητική, και έστω $A := R(x_1, \ldots)$. Από προηγούμενο λήμμα (ενότητα 4.1.2 σελίδα 51) υπάρχει ένα σύνολο $V \subseteq Var(A)$ το οποίο είναι 1-συμβατό και 1-κλειστό ώστε $K_{1,V} \notin Sat(A)$. Έστω $\overline{V} := Var(A) - V$. Επιλέγουμε $W \subseteq \overline{V}$ τέτοιο ώστε να έχει τη μεγαλύτερη πληθικότητα έτσι ώστε το $K_{0,W} \in Sat(A)$. Είναι προφανές από τους ορισμούς οτι $1 \leq |W|(|\overline{V}|$. Επιλέγουμε $\xi \in \overline{V} - W$, και έστω s μια ανάθεση που ικανοποιεί το A έτσι ώστε $s|V \equiv 1$ και $s(\xi) = 0$. Ένα τέτοιο s υπάρχει γιατί το V είναι 1-κλειστό και 1-συμβατό. Για $j \in \{0, 1\}$ ορίζουμε $W_j := \{\eta \in W : s(\eta) = j\}$, $\overline{W_j} := \{\eta \in Var(A) - W : s(\eta) = j\}$, και έστω $B := A[^{W_0}_{0}, ^{W_1}_{x}, ^{\overline{W_0}}_{y}, ^{\overline{W_1}}_{1}]$. W_1 είναι μη κενό αφού το $|W|$ είναι μέγιστο. $\overline{W_0}$ είναι μη κενό αφού περιέχει το ξ. Άρα τα x και y υπάρχουν στο B.

Προφανώς $[B] \in Rep(\{R\})$. Επίσης, το $[B]$ περιέχει το $(0, 1)$ και το $(1, 0)$, γιατί το A είναι ικανοποιήσιμο από $K_{0,W}$ και το s ταυτοχρόνως. Τέλος το $[B]$ δεν περιέχει το $(0, 0)$ λόγω του οτι το $|W|$ είναι μέγιστο. Τελικά ανάλογα με το αν το $(1, 1)$ περιέχεται στο $[B]$, το $[B]$ είναι είτε το $[x \neq y]$ ή το $[x \vee y]$. Αυτό αποδεικνύει το λήμμα στην περίπτωση που η R είναι ασθενώς αρνητική. Όμοια και για τις ασθενώς θετικές. ■

ΠΟΡΙΣΜΑ 4.1.2.1

Αν η S έχει κάποια σχέση η οποία δεν είναι ασθενώς αρνητική και μια που δεν είναι ασθενώς θετική τότε $[x \neq y] \in Rep(S)$. ■

ΑΠΟΔΕΙΞΗ

Έστω $R, R' \in S$ με R μη ασθενώς θετική και R' μη ασθενώς αρνητική. Έστω οτι $[x \neq y] \notin Rep(S)$. Τότε από το προηγούμενο λήμμα, $[x \vee y]$ και $[\neg x \vee \neg y]$ είναι στο $Rep(S)$. Άρα το $Rep(S)$ περιέχει και το $[(x \vee y) \wedge (\neg x \vee \neg y)]$, το οποίο είναι το $[x \neq y]$. ■

ΛΗΜΜΑ 4.1.2.5

Λήμμα αντικατάστασης με άρνηση: Έστω $[x \neq y] \in Rep(S)$. Τότε το $Gen^+(S)$ είναι κλειστό στη αντικατάσταση με άρνηση, δηλαδή, αν $A \in Gen^+(S)$ και ξ, η είναι μεταβλητές, $A[^{\xi}_{\neg\eta}] \in Gen^+(S)$. ■

ΑΠΟΔΕΙΞΗ

Από υπόθεση $Gen^+(S)$ περιέχει τον τύπο $x \neq y$. Παρατηρήστε οτι $A[^{\xi}_{\neg\eta}]$ είναι λογικά ισοδύναμο με $(\exists u)(A[^{\xi}_{u}] \wedge \eta \neq u)$, με u να είναι μια μεταβλητή η οποία δεν υπάρχει στο A. Άρα, $A[^{\xi}_{\neg\eta}] \in Gen^+(S)$. ■

Με τον όρο "λογική σχέση 3 στοιχείων" εννοούμε μια σχέση με ακριβώς 3 στοιχεία ορισμένη στο 2^2. Προφανώς αυτές είναι 4 οι $[x \vee y], [\neg x \vee y], [x \vee \neg y]$ και $[\neg x \vee \neg y]$.

ΛΗΜΜΑ 4.1.2.6

Έστω R μια σχέση η οποία δεν είναι affine. Τότε το $Rep(\{R, [x \neq y]\})$ περιέχει όλες τις λογικές σχέσεις 3 στοιχείων. ■

ΑΠΟΔΕΙΞΗ

Αρκεί να δείξουμε οτι η $Rep(\{R, [x \neq y]\})$ περιέχει κάποια τέτοια σχέση αφού με την αντικατάσταση με άρνηση μπορούμε να πάρουμε όλες τις άλλες.

Έστω $A := R(x_1, \ldots)$. Χρησιμοποιώντας προηγούμενο λήμμα (ενότητα 4.1.2 σελίδα 48) έστω s_0, s_1, s_2 αναθέσεις που ικανοποιούν το A έτσι ώστε το $s_0 \oplus s_1 \oplus s_2$ δεν ικανοποιεί το A. Δημιουργούμε το A' από το A αντιστρέφοντας κάθε εμφάνιση μεταβλητής στο σύνολο $\{\eta : s_0(\eta) = 1\}$. Χρησιμοποιώντας το λήμμα της αρνητικής αντικατάστασης, $A' \in Gen^+(\{R, [x \neq y]\})$. Ορίζουμε $s'_j := s_i \oplus s_0$, με $i = 1, 2$. Παρατηρούμε οτι μια ανάθεση t ικανοποιεί το A' αν και μόνο αν $t \oplus s_0$ ικανοποιεί το A. Έτσι K_0 (δηλαδή η ανάθεση 0 σε όλες τις μεταβλητές), οι s'_1, s'_2 ικανοποιούν το A' , αλλά η $s'_1 \oplus s'_2$ όχι. Για $i, j = 0, 1$, έστω $V_{i,j} := \{\xi \in Var(A') : s'_1(\xi) = i$ και $s'_2(\xi) = j\}$ και έστω $B := A[^{V_{0,0}}_{0}{}^{V_{0,1}}_{,x}{}^{V_{1,0}}_{,y}{}^{V_{1,1}}_{,z}]$. Προφανώς, $B \in Gen^+(\{R, [x \notin y]\})$. Υποθέτουμε χωρίς απώλεια της γενικότητας ότι x,y,z εμφανίζονται στο B. (Αν κάποιο δεν εμφανίζεται, έστω το x εισάγουμε ένα conjunct $(\exists w)(w \neq x)$ απλώς για να εμφανιστεί.) Από οτι έχουμε πει για το A', το $[B]$ περιέχει τα $(0,0,0), (0,1,1), (1,0,1)$ αλλά όχι το $(1,1,0)$. Υποθέτουμε ότι $Rep(\{R, [x \neq y]\})$ δεν περιέχει καμία σχέση τριών στοιχείων. Τότε η $[B]$ πρέπει να περιέχει το $(0,1,0)$ ή αλλιώς $[(\exists x)B]$ είναι $\{(0,0), (1,1), (0,1)\}$. Επίσης το $[B]$ πρέπει να περιέχει τα $(1,0,0)$ ή αλλιώς $[(\exists y)B]$ είναι $\{(0,0), (0,1), (1,0)\}$, και αυτή η αντίθεση τελειώνει την απόδειξη. ■

ΛΗΜΜΑ 4.1.2.7

Έστω R μια λογική σχέση η οποία δεν είναι bijunctive. Τότε το $Rep(\{R, [x \neq y], [x \vee y]\})$ περιέχει τη σχέση "ακριβώς ένα από τα x, y, z." ■

ΑΠΟΔΕΙΞΗ

Έστω $A := R(x_1, \ldots)$. Σύμφωνα με λήμμα (ενότητα 4.1.2 σελίδα 49)

, υπάρχει $s_0 \in Sat(A)$ και $U, V \subseteq Var(A)$ τέτοια ώστε U και V είναι σύνολα αλλαγών για το (A, s), αλλά $U \cap V$ δεν είναι σύνολο αλλαγών για το (A, s). Δημιουργούμε το A' από το A αντιστρέφοντας κάθε εμφάνιση κάθε μεταβλητής στο σύνολο $\{\eta : s_0(\eta) = 1\}$. Χρησιμοποιώντας το λήμμα της αντίστροφης αντικατάστασης $A' \in Gen^+(\{R, [x \neq y]\})$. Παρατηρούμε ότι τα U και V είναι σύνολα αλλαγής για το (A', K_0), οπού K_0 είναι η ανάθεση που σε όλες τις μεταβλητές αποδίδει το 0, αλλά το $U \cap V$ δεν είναι. (Επίσης το K_0 ικανοποιεί την A'). Ορίζουμε: $B := A'[^{Var(A')-(U\cup V)}_{0}, ^{U\cap V}_{x}, ^{U-V}_{y}, ^{V-U}_{z}]$. Με βάση τα παραπάνω για τα σύνολα αλλαγής για το (A', K_0), το $[B]$ περιέχει τα $(0,0,0), (1,1,0)$ και $(1,0,1)$ αλλά όχι το $(1,0,0)$. Τώρα ορίζουμε το $B' := B[^{x}_{\neg x}] \wedge (\neg x \vee \neg y) \wedge (\neg y \vee \neg z) \wedge (\neg z \vee \neg x)$. Σύμφωνα με το λήμμα την αντικατάστασης με αντιστροφή, $B' \in Gen^+(\{R, [x \neq y], [x \vee y]\})$. Είναι εύκολο να ελέγξουμε οτι $[B'] = \{(1,0,0), (0,1,0), (0,0,1)\}$. Δηλαδή, $[B']$ είναι η σχέση "ακριβώς ένα από τα x, y, z." ■

ΛΗΜΜΑ 4.1.2.8

Έστω R η λογική σχέση "ακριβώς ένα από τα x, y, z". Τότε $Rep(\{R\})$ είναι το σύνολο όλων των λογικών σχέσεων. ■

ΑΠΟΔΕΙΞΗ

Ορίζουμε:

$$A := (\exists u_1, u_2, u_3, u_4, u_5, u_6)$$

$$(R(x, u_1, u_4) \wedge R(y, u_2, u_4) \wedge R(u_1, u_2, u_5) \wedge R(u_3, u_4, u_6) \wedge R(z, u_3, 0))$$

$$B := R(x, y, 0)$$

Είναι προφανές να αποδείξουμε ότι το A είναι λογικά ισοδύναμο με το $(x \vee y \vee z)$ και ότι το B είναι λογικά ισοδύναμο με το $x \neq y$. Έστω η λογική σχέση Q δοσμένη με $Q = [C]$ για κάποιο προτασιακό τύπο C. Με το να εισάγει μια νέα υπαρξιακά ποσοδεικτούμενη μεταβλητή για κάθε binary λογικό σύνδεσμο του C, μπορούμε να δημιουργήσουμε τον τύπο $(\exists y_1, \ldots, y_m)D$, ισοδύναμη με την C, οπού D είναι μια conjunction προτάσεων κάθε μια από τις οποίες με το πολύ 3 μεταβλητές (και άρα η D μπορεί να επεκταθεί σε CNF μορφή με το πολύ 3 literal ανά conjunct.) Τώρα είναι προφανές, χρησιμοποιώντας τους τύπους A και B, να μετατρέψουμε το D σε έναν ισοδύναμο τύπο στο $Gen(\{R\})$. Προφανώς $Q \in Rep(\{R\})$. Άρα $Rep(\{R\})$ είναι το σύνολο όλων των λογικών προτάσεων.

■

ΑΠΟΔΕΙΞΗ

Απόδειξη του αρχικού θεωρήματος.

Πρώτα δείχνουμε οτι αν η S δεν ικανοποιεί καμία από τις περιπτώσεις (a) - (d) του πρώτου θεωρήματος, $Rep(S)$ είναι το σύνολο όλων των λογικών σχέσεων. Υποθέτουμε οτι η S δεν ικανοποιεί κανένα από τα (a) - (d). Τότε το

S περιέχει κάποια σχέση R_1 που δεν είναι ασθενώς θετική, κάποια σχέση R_2 που δεν είναι ασθενώς αρνητική, κάποια R_3 που δεν είναι affine, και κάποια R_4 που δεν είναι bijunctive. Ακολουθώντας τώρα την πορεία των προηγούμενων λημμάτων $[x \neq y] \in Rep(\{R_1, R_2\})$, $[x \vee y] \in Rep(\{R_1, R_2, R_3\})$ και τέλος $Rep(\{R_1, R_2, R_3, R_4\})$ περιέχει τη σχέση "ακριβώς ένα από τα x,y,z" και έτσι είναι το σύνολο όλων των λογικών σχέσεων. Έτσι το $Rep(S)$ είναι το σύνολο όλων των λογικών σχέσεων. Απομένει να δείξουμε οτι αν το S ικανοποιεί έναν από τους περιορισμούς (a)-(d) τότε το ίδιο ισχύει και για το $Rep(S)$. (Εύκολη απόδειξη η οποία δεν χρειάζεται για περαιτέρω). ■

4.1.3 Το θεώρημα διχοτόμησης

ΘΕΩΡΗΜΑ 4.1.3.1

Θεώρημα διχοτόμησης στο πρόβλημα ικανοποιησιμότητας: Έστω S ένα πεπερασμένο σύνολο λογικών σχέσεων. Αν το S ικανοποιεί κάποιους από τους παρακάτω περιορισμούς τότε το $SAT(S)$ είναι αποφασίσιμο σε πολυωνυμικό χρόνο. Αλλιώς το $SAT(S)$ είναι log-complete στο NP.

(a) Κάθε σχέση στο S είναι 0-ικανοποιήσιμη.

(b) Κάθε σχέση στο S είναι 1-ικανοποιήσιμη.

(c) Κάθε σχέση στο S είναι ασθενώς θετική

(d) Κάθε σχέση στο S είναι ασθενώς αρνητική.

(e) Κάθε σχέση στο S είναι affine.

(f) Κάθε σχέση στο S είναι bijunctive.

■

ΛΗΜΜΑ 4.1.3.1

Έστω S ένα πεπερασμένο σύνολο λογικών σχέσεων. Αν το S ικανοποιεί έναν από τους περιορισμούς (a)-(d) του βασικού θεωρήματος της προηγούμενης υποεννότητας, τότε το $SAT_C(S)$ είναι αποφασίσιμο σε πολυωνυμικό χρόνο. Αλλιώς το $SAT_C(S)$ είναι log-complete στο NP. ■

ΑΠΟΔΕΙΞΗ

(a) Υποθέστε ότι κάθε σχέση στο S είναι ασθενώς αρνητική. Τότε το SAT(S) είναι αποφασίσιμο με τον παρακάτω αλγόριθμο:

- Δοσμένου S-τύπου A, αντικαθιστούμε κάθε conjunct του A με έναν ισοδύναμο CNF τύπο A' που έχει το πολύ μια μεταβλητή με άρνηση σε κάθε conjunct.

- Αν κάθε conjunct του A' περιέχει μια μεταβλητή χωρίς άρνηση αποδέξου.
- Αλλιώς, έστω $(\neg\xi)$ ένα conjunct του A'. Αν το (ξ) είναι επίσης conjunct του A' ΑΠΕΡΡΙΨΕ. Αλλιώς, κόψε κάθε conjunct στο οποίο το $\neg\xi$ εμφανίζεται και το ξ από κάθε conjunct που εμφανίζεται χωρίς άρνηση. (Αν το A' είναι κενό ΑΠΟΔΟΧΗ).
- Πίσω στο 2.

(b) Όταν κάθε σχέση είναι ασθενώς αρνητική είναι παρόμοια με την (a).

(c) Υποθέτουμε οτι κάθε σχέση στην S είναι affine. Τότε για να αποφασίσουμε αν ένας δοσμένος S-τύπος A είναι ικονοποιήσιμος, μετατρέπουμε το A δε ένα ισοδύναμο σύστημα γραμμικών εξισώσεων στο $\{0,1\}$ και λύνουμε το σύστημα με Gaussian elimination. (Μειώνουμε τις μεταβλητές κατά μια κάθε φορά μέχρι είτε όλες οι μεταβλητές να φύγουν ή να καταλήξουμε σε $0 = 1$.) Αυτός είναι γνωστός πολυωνυμικός αλγόριθμος.

(d) Υποθέτουμε ότι κάθε σχέση στην S είναι bijunctive. Τότε για να αποφασίσουμε αν ένας δοσμένος S-τύπος είναι ικανοποιήσιμος, τον μετατρέπουμε σε ισοδύναμο CNF τύπο με το πολύ 2 literal ανά conjunct και χρησιμοποιούμε την Davis-Putnam διαδικασία [DP], η οποία όπως αναφέρεται και στο [C] αποφασίζει την ικανοποιησιμότητα ενός τέτοιου τύπου σε πολυωνυμικό χρόνο.

Στα (a)- (d) πάνω, δώσαμε πολυωνυμικούς αλγορίθμους για το $SAT(S)$. Για το $SAT_C(S)$, δηλαδή αν οι τύποι περιέχουν σταθερές, είναι προφανές πως μετατρέπουμε τους αλγορίθμους. Υποθέστε τώρα ότι το S δεν ικανοποιεί κανένα από τους περιορισμούς (a)- (d). Θα δείξουμε ότι το $SAT_C(S)$ είναι NP-complete δείχνοντας $SAT3 \leq_{log} SAT_C(S)$, οπού SAT3 είναι το σύνολο από ικανοποιήσιμους CNF τύπους που έχουν το πολύ 3 literal ανά conjunct, ένα πολύ γνωστό NP-complete πρόβλημα.

Έστω R_0, R_1, R_2, R_3 είναι οι 3 θέσεων λογικές σχέσεις που ορίζονται ως $R_0(x,y,z) \equiv (x \vee y \vee z), R_1(x,y,z) \equiv (\neg x \vee y \vee z), R_2(x,y,z) = (\neg x \vee \neg y \vee z), R_3(x,y,z) \equiv (\neg x \vee \neg y \vee \neg z)$. Για $i = 0,1,2,3$ έστω $F_i(x,y,z)$ ένα τύπος στο $Gen(S)$ που είναι λογικά ισοδύναμος με τον $R_i(x,y,z)$. Τέτοιοι τύποι υπάρχουν από το βασικό θεώρημα της προηγούμενης υποενότητας. Δοσμένου ενός CNF τύπου A με το πολύ 3 literal σε κάθε conjunct, δημιουργούμε έναν ισοδύναμο τύπο A' αντικαθιστώντας κάθε conjunct του A με έναν από τους τύπους F_i με τις κατάλληλες μεταβλητές να έχουν αντικατασταθεί. Δημιουργούμε το A'' από το A' διαγράφοντας όλους τους ποσοδείκτες, αφού είμαστε σίγουροι ότι όλες οι ποσοδεικτούμενες μεταβλητές είναι διαφορετικές μεταξύ τους και από όλες τις άλλες ελεύθερες μεταβλητές. Παρατηρούμε ότι το A'' είναι ικανοποιήσιμο αν και μόνο αν A είναι ικανοποιήσιμο. Δεν είναι δύσκολο να δείξουμε ότι το A'' είναι log-space υπολογίσιμο από το A. Αυτό αποδεικνύει οτι $SAT3 \leq_{log} SAT_C(S)$ Προφανώς αφού $SAT_C(S) \in NP$, είναι επακόλουθο ότι το πρόβλημα $SAT_C(S)$ είναι log-complete στο NP. ■

Ορίζουμε "χωρίς-σταθερές" ανάλογα του $Gen(S)$ και του $Rep(S)$, ως ακολούθως. Έστω $Gen_{NC}(S) := \{A \in Gen(S) :$ δεν εμφανίζονται μεταβλητές στο $A\}$, $Rep_{NC}(S) := \{[A] : A \in Gen_{NC}(S)\}$. Η λογική σχέση R είναι συμπληρωματική αν είναι κλειστή ως προς συμπλήρωμα, δηλαδή αν για κάθε $(a_1, \ldots, a_m) \in R$, $(1 - a_1, \ldots, 1 - a_m) \in R$. Το ακόλουθο λήμμα καθορίζει τη σχέση μεταξύ των $SAT(S)$ και $SAT_C(S)$.

ΛΗΜΜΑ 4.1.3.2

Αν $Rep_{NC}(S)$ περιέχει τα $[x]$ και $[\neg x]$, τότε $Rep(S) = Rep_{NC}(S)$, και άρα $SAT_C(S)$ και $SAT(S)$ έχουν την ίδια πολυπλοκότητα. ∎

Όπως το επόμενο λήμμα δείχνει η υπόθεση του λήμματος αποτυγχάνει μόνο σε πολύ περιορισμένες συνθήκες. Έτσι για τα περισσότερα σύνολα $S, SAT(S)$ και $SAT_C(S)$ έχουν την ίδια πολυπλοκότητα.

ΛΗΜΜΑ 4.1.3.3

Έστω S να είναι ένα σύνολο από μη κενές λογικές σχέσεις. Τότε τουλάχιστον ένα από τα ακόλουθα ισχύει:

(a) Κάθε σχέση στο S είναι ικανοποιήσιμη από το $(0, \ldots 0)$.

(b) Κάθε σχέση στο S είναι ικανοποιήσιμη από το $(1, \ldots 1)$.

(c) $[x]$ και $[\neg x]$ περιέχονται στο $Rep_{NC}(S)$.

(d) $[x \neq y] \in Rep_{NC}(S)$.

Μάλιστα, αν το (c) αποτυγχάνει και το (d) ισχύει, κάθε σχέση της S είναι συμπληρωματική. ∎

ΑΠΟΔΕΙΞΗ

Υποθέτουμε όπως υπόθηκε πιο πάνω ότι όλες οι σχέσεις στο S είναι μη κενές. Υποθέτουμε ότι το (a) και το (b) αποτυγχάνουν. Θα δείξουμε οτι το (c) ή το (d) ισχύει.

• Πρώτη περίπτωση. Κάθε σχέση στο S είναι 0- ή 1-ικανοποιήσιμη. Σε αυτή την περίπτωση, αφού τα (a) και (b) αποτυγχάνουν, υπάρχει κάποιο $R_0 \in S$ το οποίο είναι 0-ικανοποιήσιμο αλλά όχι 1-ικανοποιήσιμο, και κάποιο $R_1 \in S$ το οποίο είναι 1-ικανοποιήσιμο αλλά όχι 0-ικανοποιήσιμο. Έστω $A_i := R_i(x_1 \ldots)$, έχουμε $[A_i[^{Var(A_i)}_{x}]] = (i)$ για $i = 0, 1$, άρα, $[x]$ και $[\neg x]$ είναι στο $Rep_{NC}(S)$,και άρα το (c) ισχύει.

• Δεύτερη περίπτωση. Το S περιέχει κάποια μη κενή σχέση R η οποία δεν είναι ούτε 0-ικανοποιήσιμη ούτε 1-ικανοποιήσιμη. Σε αυτή την περίπτωση, έστω $A := R(x_1, \ldots)$ και επιλέγουμε $s \in Sat(A)$. Έστω: $B := A[^{s^{-1}(0)}_{x_0}{}^{s^{-1}(1)}_{,x_1}]$ Αφού το R δεν είναι 0 ή 1-ικανοποιήσιμο τα x_0, x_1 υπάρχουν στο B και το $[B]$ είναι είτε το $(0,1)$ ή $(0,1),(1,0)$. Αν ισχύει το πρώτο $[(\exists x_0)B] = [x]$ και

$[(\exists x_1)B] = [\neg x]$ και άρα το (c) ισχύει. Αλλιώς $[B] = [x \neq y]$ και ισχύει το (d). Άρα το (c) ή το (d) ισχύουν πάντα.

Τώρα υποθέτουμε ότι το (c) δεν ισχύει ενώ το (d) ισχύει. Ισχυριζόμαστε οτι $[x] \notin Rep_{NC}(S)$. Γιατί αν $[x] \in Rep_{NC}(S)$, θα είχαμε $[(\exists x)(x \wedge x \neq y)] = [\neg y]$, ενώ το (c) δεν ισχύει. Ομοίως, $[\neg x] \notin Rep_{NC}(S)$. Έστω ότι υπάρχει μια R που δεν είναι συμπληρωματική. Έστω $A := R(x_1, \ldots)$ και επιλέγουμε $s \in Sat(A)$ τέτοιο ώστε $\overline{s} \notin Sat(A)$, όπου $\overline{s}$ είναι η συμπληρωματική ανάθεση του s. Ορίζουμε έναν τύπο B όπως πάνω (στη δεύτερη περίπτωση). Τώρα το x_0 εμφανίζεται στη B αλλιώς $[B] = [x]$. Ομοίως και το x_1. Έτσι $Var(B) = \{x_1, x_2\}$ και το $[B]$ περιέχει τα $(0, 1)$ αλλά όχι $(1, 0)$. Όμως τώρα το $[B]$ πρέπει να περιέχει το $(0, 0)$, γιατί αλλιώς το $[(\exists x_0)B]$ είναι $[x]$. Επίσης το $[B]$ περιέχει το $(1, 1)$ αλλιώς το $[(\exists x_1)B]$ είναι το $[\neg x]$. Άρα $[B]$ είναι το $[x \rightarrow y]$. Αλλά τότε $[\exists x)((x \rightarrow y) \wedge (x \neq y))]$ είναι το $[y]$, και με εις άτοπο απαγωγή τελειώνει η απόδειξη. ■

Τελικά μπορούμε να αποδείξουμε το θεώρημα διχοτόμησης του $SAT(S)$.

ΑΠΟΔΕΙΞΗ

Υποθέτουμε οτι δεν είναι κάθε σχέση στην S 0-ικανοποιήσιμη ή κάθε σχέση 1-ικανοποιήσιμη. Δηλαδή τα $(a) - (b)$ του θεωρήματος αποτυγχάνουν. Από το παραπάνω λήμμα έχουμε δύο περιπτώσεις να εξετάσουμε.

• Πρώτη περίπτωση. $[x]$ και $[\neg x] \in Rep_{NC}(S)$

Τότε μπορούμε να αντικαταστήσουμε εύκολα έναν S-τύπο με σταθερές από έναν χωρίς σταθερές και χρησιμοποιώντας τα αποτέλεσμα του πιο πάνω λήμματος (ενότητα 4.1.3 σελίδα 55) .

• Δεύτερη περίπτωση. $[x \neq y] \in Rep_{NC}(S)$, και κάθε σχέση στην S είναι συμπληρωματική.

Έστω ένας S-τύπος με σταθερές και δοσμένο A. Έστω A' να είναι $A'' \wedge (y_0 \neq y_1)$, οπού A'' είναι το A αφού αντικαταστήσουμε κάθε εμφάνιση του 0 από το y_0 και κάθε εμφάνιση του 1 από το y_1, με y_0, y_1 νέες μεταβλητές. Άρα, A' είναι ένας S-τύπος χωρίς σταθερές, και από συμπληρωματικότατα το A' είναι ικανοποιήσιμο αν και μόνο αν το A είναι ικανοποιήσιμο και πάλι φτάνουμε στο ζητούμενο αποτέλεσμα με το λήμμα για το πρόβλημα με σταθερές. ■

Βιβλιογραφία

[1] Mak'tsev A. Bulatov. Constraints are tractable. Technical Report PRG-RR-02-05, Computing Laboratory, University of Oxford, Oxford, UK, 2002.

[2] P. Jeavons A. Bulatov and A.Krokhin. Classifying the complexity of constraints using finite algebras. *SIAM journal of Computing*, 34:720–742, 2005.

[3] A.Bulatov. A dichotomy theorem for constraints on a three-element set. In *Proceedings of the 43rd IEEE Symposium on Foundations of Computer Science —FOCS'02 (Vancouver,BC,Canada, 2002)*, pages 649–658, 2002.

[4] A.Krokhin A.Bulatov and P. Jeavons. The complexity of maximal constraint languages. In *Proceedings of the 33rd ACM Symposium on Theory of Computing —STOC'01 (2001)*, pages 667–674, 2001.

[5] M. Bodirsky and J. Nesetril. Constraint satisfaction with countable homogeneous templates. In *Proceedings of Computer Science Logic and the 8th Kurt Godel Colloquium, Lecture Notes in Comput. Sci. 2804, Springer-Verlag — (Berlin, 2003)*, pages 44–57, 2003.

[6] A. Bulatov and P. Jeavons. Tractable constraints closed under a binary operation. Technical Report PRG-TR-12-00, Computing Laboratory, University of Oxford, Oxford, UK, 2002.

[7] E.F. Codd. A relational model of data for large shared databanks. *Communications of the ACM*, 13(6):377–387, 1970.

[8] P.M. Cohn. *Universal Algebra.* Harper & Row, 1965.

[9] M.C. Cooper, D.A. Cohen, and P.G. Jeavons. Characterising tractable constraints. *Artificial Intelligence*, 65:347–361, 1994.

[10] V. Dalmau. A new tractable class of constraint satisfaction problems. In *Proceedings of the 6th International Symposium on Artificial Intelligence and Mathematics*, 2000.

[11] V. Dalmau. Constraint satisfaction problems in non-deterministic logarithmic space. In *Proceedings of 29th International Colloquium on Automata, Languages and Programming —ICALP'02, Lecture Notes in Comput. Sci. 2380, Springer-Verlag (Berlin, 2002)*, pages 414–425, 2002.

[12] V. Dalmau and J. Pearson. Set functions and width 1 problems. In *Proceedings of the 5th International Conference on Constraint Programming —CP'99, Lecture Notes in Comput. Sci. 1713, Springer-Verlag (Berlin, 1999)*, pages 159–173, 1999.

[13] K. Denecke and S. Wismath. *Universal Algebra and Applications in Theoretical Computer Science*. Champman and HALL/CRC Press,Boca Raton,FL, 2002.

[14] J. B. Fraleigh, editor. *A first course in Abstract Algebra*. Addison-Wesley Publishing Company, Inc., 1988.

[15] M. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, San Francisco, CA., 1979.

[16] M. Gyssens, P.G. Jeavons, and D.A. Cohen. Decomposing constraint satisfaction problems using database techniques. *Artificial Intelligence*, 66(1):57–89, 1994.

[17] Gerd G. Hillebrand, Paris C. Kanellakis, Harry G. Mairson, and Moshe Y. Vardi. Tools for Datalog boundedness. pages 1–12, 1991.

[18] D. Hobby and R. McKenzie. The structure of finite algebras. *Contemporary Mathematics,AMS,Providence,RI*, 76, 1988.

[19] P.G. Jeavons. On the algebraic structure of combinatorial problems. Technical Report CSD-TR-95-15, Computer Science Department, Royal Holloway, University of London, Egham , Surrey , UK, 1995. to appear in Theoretical Computer Science.

[20] P.G. Jeavons, D.A. Cohen, and M.C. Cooper. Constraints, consistency and closure. *Artificial Intelligence*, 101(1-2):251–265, 1998.

[21] P.G. Jeavons, D.A. Cohen, and M. Gyssens. A unifying framework for tractable constraints. In *Proceedings 1st International Conference on Constraint Programming—CP'95 (Cassis, France, September 1995)*, volume 976 of *Lecture Notes in Computer Science*, pages 276–291. Springer-Verlag, 1995.

[22] P.G. Jeavons, D.A. Cohen, and M. Gyssens. Closure properties of constraints. Technical Report CSD-TR-96-15, Computer Science Department, Royal Holloway, University of London, Egham, Surrey, UK, 1996. also appears in Journal of the ACM.

[23] P.G. Jeavons and M.C. Cooper. Tractable constraints on ordered domains. *Artificial Intelligence*, 79(2):327–339, 1995.

[24] P. Jonsson and C. Bäckström. A unifying approach to temporal constraint reasoning. *Artificial Intelligence*, 102:143–155, 1998.

[25] L. Kirousis. Fast parallel constraint satisfaction. *Artificial Intelligence*, 64:147–160, 1993.

[26] P. Kolaitis and M. Vardi. The decision problem for the probabilities of higher-order properties. In *STOC '87: Proceedings of the nineteenth annual ACM conference on Theory of computing*, pages 425–435, New York, NY, USA, 1987. ACM Press.

[27] D. Lesaint, N. Azarmi, R. Laithwaite, and P. Walker. Engineering dynamic scheduler for work manager. *BT Technology Journal*, 16:16–29, 1998.

[28] A.K. Mackworth. Consistency in networks of relations. *Artificial Intelligence*, 8:99–118, 1977.

[29] R.N. McKenzie, G.F. McNulty, and W.F. Taylor. *Algebras, Lattices and Varieties*, volume I. Wadsworth and Brooks, California, 1987.

[30] U. Montanari. Networks of constraints: Fundamental properties and applications to picture processing. *Information Sciences*, 7:95–132, 1974.

[31] B. Nebel and H-J. Burckert. Reasoning about temporal relations: a maximal tractable subclass of Allen's interval algebra. *Journal of the ACM*, 42:43–66, 1995.

[32] C.H. Papadimitriou. *Computational Complexity.* Addison-Wesley, 1994.

[33] J.K. Pearson and P.G. Jeavons. A survey of tractable constraint satisfaction problems. Technical Report CSD-TR-97-15, Royal Holloway, University of London, July 1997.

[34] L. Purvis and P. Jeavons. Constraint tractability theory and its application to the product development process for a constraint-based scheduler. Technical report, Xerox Corporation, Webster, NY, USA, December 1998. (Submitted to PACT'99).

[35] T.J. Schaefer. The complexity of satisfiability problems. In *Proceedings 10th ACM Symposium on Theory of Computing (STOC)*, pages 216–226, 1978.

[36] A. Szendrei. *Clones in Universal Algebra*, volume 99 of *Seminaires de Mathematiques Superieures.* University of Montreal, 1986.

[37] T.Feder and M.Vardi. The computational structure of monotone monadic snp and constraint satisfaction: A study through datalog and group theory. *SIAM journal of Computing*, 28:57–104, 1998.

[38] P. van Beek. On the minimality and decomposability of row-convex constraint networks. In *Proceedings AAAI-92 (San Jose, CA)*, pages 447–452, 1992.