Εθνικο Μετσοβιο Πολυτεχνειο
Σχολη Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών
Εργαστήριο Λογικής και Επιστήμης Υπολογιστών

Ενοποίηση Σχετικιστικών και Φυσικών Αποδείξεων για την Εύρεση νέων Κάτω Φραγμάτων

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Πλάτων-Λέανδρος Παπαδόπουλος

Επιβλέπων: Παγουρτζής Αριστείδης
Αν. Καθηγητής Ε.Μ.Π.

Αθήνα, Μάρτιος 2017

Εθνικο Μετσοβιο Πολυτεχνειο
Σχολη Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών
Εργαστήριο Λογικής και Επιστήμης Υπολογιστών

Ενοποίηση Σχετικιστικών και Φυσικών Αποδείξεων για την Εύρεση νέων Κάτω Φραγμάτων

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Πλάτων-Λέανδρος Παπαδόπουλος

Επιβλέπων: Παγουρτζής Αριστείδης
Αν. Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 23η Μαρτίου 2017.

.....	.....	.....
Α. Παγουρτζής	**Δ. Φωτάκης**	**Ν. Παπασπύρου**
Αν. Καθηγητής Ε.Μ.Π.	Επ. Καθηγητής Ε.Μ.Π.	Αν. Καθηγητής Ε.Μ.Π.

Αθήνα, Μάρτιος 2017

.....
Πλάτων-Λέανδρος Παπαδόπουλος
Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών

Περίληψη

Στην παρούσα Διπλωματική Εργασία μελετάμε τη σύνδεση των ανομοιόμορφων μοντέλων υπολογισμού με τις παραδοσιακές υπολογιστικές δυνατότητες των μηχανών Turing. Συγκεκριμένα τα πρώτα είναι στενά συνδεδεμένα με τις οικογένειες λογικών κυκλωμάτων και αποδεικνύεται ότι, ακριβώς χάρη στην ανομοιομορφία τους, αποτελούν ένα παντοδύναμο υπολογιστικό μοντέλο, καθιστώντας τα ισχυρότερα από όλες τις γνωστές υπολογιστικές μεθόδους αλλά με το προφανές μειονέκτημα ότι δεν μπορούν να υλοποιηθούν ρεαλιστικά. Εν τούτοις, αποτελούν ένα πολύ αποδοτικό εργαλείο ως μέτρο συμπύκνωσης των υπολογιστικών βημάτων που απαιτεί ο υπολογισμός μιας συνάρτησης και φαίνεται να έχουν αμέτρητες εφαρμογές και συνδέσεις με τις ομοιόμορφες υπολογιστικές κλάσεις. Αποκορύφωμα αυτών, είναι μέχρις στιγμής η στενή σύνδεση της εικασίας ανυπαρξίας χαμηλών κυκλωματικών κλάσεων για υψηλές υπολογιστικές κλάσεις με τη δυνατότητα αποδοτικά ομοιόμορφης παραγωγής ψευδοτυχαιότητας (μία από τις κρισιμότερες παραμέτρους στη σύγχρονη Κρυπτογραφία). Από την άλλη το πεδίο των Κυκλωμάτων φάνηκε να αποτελεί το πιο εύφορο πεδίο για την εύρεση φραγμάτων στις υπολογιστικές δυνατότητες ομοιόμορφων κλάσεων περιορισμένων πόρων (χρόνου, χώρου κ.λπ.), ειδικότερα μετά την απόδειξη της αδυναμίας των απλών επιχειρημάτων αυτοαναφοράς να λύσουν μεγάλα ανοιχτά ερωτήματα της Θεωρίας Πολυπλοκότητας (όπως αν $P = NP$, $P = BPP$ κ.ο.κ.). Εν τούτοις, το ομώνυμο άρθρο περί των Φυσικών Αποδείξεων για αρκετό καιρό είχε απογοητεύσει την επιστημονική κοινότητα όσον αφορά την πιθανότητα ύπαρξης μιας απλής απόδειξης που να χαρακτηρίζει με τεχνικά επιχειρήματα τις συναρτήσεις που επιδέχονται κυκλώματα πολυωνυμικού μεγέθους. Τα εμπόδια αυτά φαίνεται πως έρχεται να άρει μια σχετικά πρόσφατη μέθοδος, η οποία προτάθηκε κατά κύριο λόγο από τον Ryan Williams και η οποία χρησιμοποιεί την παραγωγή μη τετριμμένων αλγορίθμων για κυκλώματα (εκμεταλλεύοντας τεχνικές ιδιότητες συγκεκριμένων κυκλωματικών κλάσεων) και η οποία στη συνέχεια κατασκευάζει μια κυκλωματική δομή που να υπολογίζει σχετικούς αλγόριθμους σε χρόνο που αντιβαίνει παραδοσιακά επιχειρήματα διαγωνιοποίησης. Η μέθοδος αυτή είναι αφενός ελπιδοφόρα επειδή, συνδυάζοντας συντακτικές και σημασιολογικές τεχνικές, φαίνεται η κάθε πλευρά να είναι ικανή να καλύψει τις αδυναμίες που είχε η άλλη (όταν χρησιμοποιούνταν αυτοτελώς), αλλά και αφετέρου επειδή έχει ήδη δείξει τις δυνατότητες της μέσω του πρόσφατου αποτελέσματος (επίσης από τον R.W.) ότι $NEXP \nsubseteq ACC^0$ συνοδευόμενου από μια πληθώρα θεωρημάτων ικανών να παράξουν μια εν δυνάμει μεγάλη ποικιλία νέων κάτω φραγμάτων.

Abstract

In this Diploma Thesis we study the connection between non-uniform computation models and the traditional computational capabilities of Turing Machines. Specifically the former are closely related to Boolean circuit families and it can be proved that, exactly due to their non-uniformity, they consist an omnipotent model, making them more powerful than any known computational methods, having though the obvious drawback of not corresponding to a realistic and implementable model. Even so, they consist a very efficient tool as a way to measure the compressibility of the steps required when computing a function and they seem to have limitless applications and connections to uniform computational classes. Highlight of all these, so far, is the close connection of the small circuits absence for high computational classes conjecture to the effective uniform production of pseudorandomness (one of the most crucial aspects in modern Cryptography). On the other side, the Circuits field seemed to be the most fertile for finding new lower bounds in the computational capacity of uniform classes with bounded resources (time, space etc.), especially after the proof of the impossibility to solve great open problems of Complexity Theory (such as if $P = NP$, $P = BPP$, et al.) using simple self-referential arguments. Nevertheless, the Natural Proofs paper had for a great amount of time discourage the scientific community that it is possible to find a simple proof which characterizes functions that accept circuits of polynomial size using only technical arguments. It seems possible, though, for these barriers to be raised by a relatively recent technique, introduced mainly by Ryan Williams, that exploits the production of non-trivial algorithms for circuits (taking advantage of technical properties of specific circuit classes) and subsequently creates a circuit structure able to compute related algorithms in time that defies traditional diagonalization arguments. This approach is on the one hand promising for, combining syntactic and semantic techniques, it seems that each one is able to cover the weak points of the other (that arise when they are used independently) and on the other hand it has already shown its potential through the recent result (also by R.W.) that $NEXP \nsubseteq ACC^0$ accompanied with a plethora of theorems able to produce, under certain circumstances, a big variety of new lower bounds.

To D.

Ευχαριστίες

Θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα καθηγητή της παρούσας εργασίας, τον κ. Παγουρτζή, όπως επίσης και τα άλλα δύο μέλη της εξεταστικής επιτροπής, τους κ. Φωτάκη και κ. Παπασπύρου, για τη γενικότερη βοήθεια τους τα χρόνια της προπτυχιακής μου πορείας. Επιπλέον, δε θα μπορούσα να παραλείψω να ευχαριστήσω όλα τα μέλη του εργαστηρίου όπως ασφαλώς και το πρεσβύτερο και ιδρυτικό του μέλος, τον κ. Ζάχο για την εύθυμη δημιουργική ατμόσφαιρα που ακούραστα προσφέρει σε αυτό. Ασφαλώς, ιδιαίτερη αναφορά αξίζει στον υπ. διδάκτορα και υπεύθυνο του εργαστηρίου Α. Αντωνόπουλο, ο οποίος ήταν αυτός που συνεισέφερε τα μέγιστα κατά την εκπόνηση της παρούσας εργασίας.

Επί της ευκαιρίας της ολοκλήρωσης της διπλωματικής μου διατριβής, η οποία σηματοδοτεί και την ολοκλήρωση των προπτυχιακών μου σπουδών, θα ήθελα να ευχαριστήσω ιδιαίτερα την οικογένεια και τους φίλους μου, χωρίς τη συμπαράσταση και συντροφικότητα των οποίων δεν θα είχα καταφέρει να φτάσω σε αυτό το σημείο. Τέλος, θα ήθελα να αφιερώσω αυτή την εργασία σε όλους τους ανθρώπους που προσπάθησαν και – επιτυχημένα ή μη – προσέφεραν σε όλα τα επιστημονικά και άλλης φύσεως επιτεύγματα της ανθρώπινης ιστορίας, με ιδιαίτερη μνεία στους C. Barks , A. Christie και E. Oda , το έργο των οποίων στάθηκε θεμελιώδους σημασίας για την ακαδημαϊκή μου πορεία.

Περιεχόμενα

Κεφάλαιο 1

Εισαγωγή

1.1 Γενικό πλαίσιο

Σκοπός της παρούσας διπλωματικής εργασίας είναι η εξερεύνηση της φύσης του υπολογισμού υπό ομοιόμορφα και ανομοιόμορφα μοντέλα και η μελέτη πρόσφατων αποτελεσμάτων που προέκυψαν από συνδυαστικά και λογικά επιχειρήματα των δύο αυτών μοντέλων.

Συγκεκριμένα, ως ομοιόμορφο μοντέλο υπολογισμού ορίζεται αυτό για το οποίο για κάθε είσοδο οποιουδήποτε μήκους έχουμε στη διάθεση μας την ίδια μέθοδο (αλγόριθμο) αντιμετώπισης ώστε να καταλήξουμε στο υπολογιστέο. Χαρακτηριστικό αυτών των μοντέλων είναι ότι ακριβώς λόγω του μικρού (πεπεραμσμένου) μήκους περιγραφής της μεθόδου, εν τέλει δε μπορούν να λύσουν (υπολογίσουν) κάθε δυνατό πρόβλημα. Εν τούτοις, ακριβώς λόγω της πεπερασμένης περιγραφής τους, θεωρούνται ως οι μόνοι υποψήφιοι για μια ρεαλιστική περιγραφή ενός υπολογιστικού μοντέλου. Ένα συνηθισμένο παράδειγμα αυτών των μοντέλων είναι οι κλασικοί υπολογιστές (όταν τους δίνεται δυνατότητα χρήσης απεριόριστης μνήμης).

Από την άλλη, ως ανομοιόμορφο μοντέλο υπολογισμού, εννοούμε αυτό όπου χωρίζουμε τις εισόδους σε ομάδες ανάλογα με το μήκος (ή ενδεχομένως κάποια άλλη τους ιδιότητα) και για κάθε ομάδα έχουμε και διαφορετικό αλγόριθμο υπολογισμού. Χαρακτηριστικό αυτών των μοντέλων είναι ότι επειδή η όλη τους περιγραφή είναι μη πεπερασμένη (αφού υπάρχει διαφορετικός αλγόριθμος για κάθε μήκος εισόδου) και επειδή κάθε μία αφορά πεπερασμένο πλήθος εισόδων, μπορεί να χρησιμοποιηθεί αυτή η άπειρη περιγραφή με τρόπο τέτοιο ώστε να επιλύουν κάθε δυνατό πρόβλημα. Όπως είναι ασφαλώς αναμενόμενο, αυτό το μοντέλο δε θεωρείται ρεαλιστικό και δεν αντιστοιχεί σε μια πρακτικά εφαρμόσιμη μέθοδο. Το πιο σύνηθες παράδειγμα τέτοιου μοντέλου θεωρείται αυτό της οικογένειας λογικών κυκλωμάτων, όπου για κάθε είσοδο, έχουμε και ένα διαφορετικό κύκλωμα-υπολογιστή.

Παρόλα αυτά, χάρη στην ισχυρή διαισθητική τους σύνδεση με τα κάθε είδους κυκλώματα που χρησιμοποιούνται σε τεράστιο φάσμα του τεχνολογικού

τομέα, τα τελευταία έχουν κερδίσει μια περίοπτη θέση τόσο στη μελέτη των ιδιοτήτων τους και κυρίως της παροχής παραλληλίας που παρέχουν, όσο και στη μελέτη της σύνδεσης τους με τα παραδοσιακά ρεαλιστικά μοντέλα υπολογισμού. Ένας ακόμη από τους λόγους που αυτό συνέβη είναι επειδή εν τέλει παρέχουν μια πιο εποπτική εικόνα των υπολογιστικών βημάτων (κάτι που προφανώς δεν είναι δυνατόν από μια πεπερασμένη περιγραφή ενός αλγορίθμου-προγράμματος). Αυτό, από μόνο του στάθηκε αρκετός παράγοντας ώστε να θεωρηθούν από την επιστημονική κοινότητα ως ένα πιο γόνιμο και φιλικό έδαφος για τη μελέτη των δυνατοτήτων ενός υπολογιστικού συστήματος που βρίσκεται υπό κάποιους περιορισμούς πόρων (χρόνου, χώρου, τυχαιότητας κ.ο.κ.).

Η παραπάνω προσέγγιση, έγινε ακόμη πιο στενή όταν άρχισε να διαφαίνεται ότι αποτελέσματα που αφορούσαν μόνο ομοιόμορφες υπολογιστικές κλάσεις αποδεικνύονταν κάνοντας χρήση θεωρημάτων που αφορούσαν ανομοιόμορφα μοντέλα. Το πλέον χαρακτηριστικό παράδειγμα αυτού, είναι η στενή σύνδεση της κυκλωματικής δυσκολίας για ορισμένες υπολογιστικές κλάσεις με τη δυνατότητα ομοιόμορφης παραγωγής ψευδοτυχαιότητας· η ψευδοτυχαιότητα έχει να κάνει με την παραγωγή συμβολοσειρών οι οποίες να μοιάζουν τυχαίες σε έναν περιορισμένων πόρων παρατηρητή και όπως θα δείξουμε αναλυτικότερα έχει θεμελιώδεις συσχετίσεις με την ασφάλεια των σύγχρονων κρυπτογραφικών πρωτοκόλλων.

Πέραν της ψευδοτυχαιότητας ωστόσο, η ανομοιομορφία των κυκλωμάτων (σε παραλληλία με την ανομοιομορφία των πιθανών εισόδων-συμβολοσειρών κάποιου μήκους) φάνηκε να αποτελεί ένα καλό τρόπο για την αναπαράσταση της συμπύκνωσης της πληροφορίας και να δύναται να οδηγήσει σε συμπεράσματα που αφορούσαν κάτω φράγματα και διαχωρισμούς υπολογιστικών κλάσεων (μέσω συνδυαστικών επιχειρημάτων που συνήθως απέκλειαν τη δυνατότητα έκφρασης ενός προβλήματος από ένα κύκλωμα περιορισμένου πλήθους λογικών πυλών κ.ο.κ.). Η προσέγγιση αυτή φάνηκε ιδιαίτερα ελπιδοφόρα, ειδικά μετά την απόδειξη ότι οι παραδοσιακές αυτοαναφορικές μέθοδοι, που χρησιμοποιούνταν μέχρι τότε για το διαχωρισμό κλάσεων και παραγωγή κάτω φραγμάτων, αδυνατούσαν και επίσημα να απαντήσουν στα θεμελιώδη ανοιχτά ερωτήματα της Θεωρητικής Πληροφορικής, όπως το αν $P = NP$ (το οποίο διαισθητικά αφορά το αν ένα πρόβλημα με εύκολα ελέγξιμη λύση έχει και εύκολα κατασκευάσιμη λύση).

Παρότι αυτή η στροφή της επιστημονικής κοινότητας στον τομέα των ανομοιόμορφων μοντέλων απέφερε ορισμένους σπουδαίους καρπούς καθώς και μια μεγάλη ποικιλία αρκετά δύσκολων τεχνικών επιχειρημάτων, εν τέλει μετά από κάποια χρόνια προέκυψε ότι η καθαρή μελέτη των συνδυαστικών ιδιοτήτων κάποιων εκφραστικά ικανών κυκλωματικών κλάσεων δε δύναται να οδηγήσει σε κάποιο αντίστοιχο κάτω φράγμα (εκτός κι αν παραβιάζονται κάποιες εύλογες και ευρεία αποδεκτές κρυπτογραφικές εικασίες). Για ένα σεβαστό χρονικό διάστημα μετά από αυτό το αποτέλεσμα, η επιστημονική κοινότητα έμεινε αμήχανη όσον αφορά τις προοπτικές απόδειξης τέτοιων ανοιχτών ερωτημάτων όπως το $P =^{?} NP$, κάτι που σταδιακά οδήγησε στην όλο και αυξανόμενη πεποίθηση

ότι μια τέτοια απόδειξη (αν υπάρχει) είναι ιδιαίτερα δύσκολη και περίπλοκη, εν μέρει αποθαρρύνοντας σταδιακά την έντονη και πολυπληθή ενασχόληση μαζί τους.

Αλλαγή στο σκηνικό αυτό, μπορεί να έρθει από τη πρόσφατη παρουσίαση μιας νέας προσέγγισης, η οποία αποφεύγει αμφότερα τα προβλήματα και εμπόδια των δύο προηγούμενων και ο εντυπωσιακός τρόπος που το κάνει είναι συνδυάζοντας και τις δύο! Ο λόγος που αυτό πετυχαίνει, όπως θα φανεί λεπτομερέστερα ακολούθως, είναι επειδή αμφότερες οι προσεγγίσεις είχαν κατά κάποιο τρόπο επικεντρωθεί σε μία μόνο μορφή υπολογισμού· η πρώτη καθαρά στην ομοιόμορφη και την υψηλού επιπέδου περιγραφή αλγορίθμων και η δεύτερη στην ανομοιόμορφη και τη συνδυαστική-τεχνική μελέτη των επί μέρους πυλών-πράξεων ενός κυκλώματος. Κάθε μία τέτοια προσέγγιση όμως οδηγούσε σε μια αδυναμία εκφραστικότητας και απαιτούσε την ύπαρξη ενός διαφορετικής φύσεως παραδείγματος. Συνδυάζοντας και τις δύο τεχνικές, όμως, σε μία απόδειξη, τα αδύναμα σημεία της κάθε προσέγγισης καλύπτονται από τα προτερήματα του άλλης, χωρίς να χάνονται τα οφέλη κάθε μιας.

Σε αυτή την εργασία, λοιπόν, θα μελετήσουμε τα δύο αυτά μοντέλα υπολογισμού αρχικά ξεχωριστά και στη συνέχεια ενοποιώντας τα, αρχικά μέσω συμπερασμάτων που εξάγονται από υποθέσεις του ενός για το άλλο και εν τέλει ολοκληρώνοντας με την παρουσίαση του μοτίβου της ενιαίας απόδειξης που παίρνει στοιχεία από αμφότερες τις τεχνικές και οδηγεί όχι μόνο σε ορισμένα σημαντικά αποτελέσματα υπό συγκεκριμένες συνθήκες, όσο και στα πρώτα άνευ όρων αποτελέσματα διαχωρισμού υπολογιστικών και κυκλωματικών κλάσεων, μετά μάλιστα, από μια αρκετά μεγάλη περίοδο μη παραγωγής (τέτοιας κλίμακας) σχετικών θεωρημάτων.

1.2 Δομή

Ακολουθεί η ακριβής δομή καθώς και μια σύντομη περίληψη των επόμενων κεφαλαίων:

- **Κεφάλαιο 2:** *Υπολογιστική Πολυπλοκότητα*
  Εισάγονται τα βασικά μοντέλα ομοιόμορφου υπολογισμού μαζί με τις διάφορες παραλλαγές που οδηγούν στις διάφορες γενικές υπολογιστικές κλάσεις που αναφέρονται στα ακόλουθα Κεφάλαια. Επίσης παρουσιάζονται κάποια κλασικά και θεμελιώδη παραδείγματα διαγωνιοποίησης μαζί με τη βασική απόδειξη που απορρίπτει αυτή τη προσέγγιση από την απόδειξη των μεγάλων ανοιχτών ερωτημάτων της Θεωρίας Πολυπλοκότητας.

- **Κεφάλαιο 3:** *Κυκλωματική Πολυπλοκότητα*
  Εισάγονται τα βασικά μοντέλα ανομοιόμορφου υπολογισμού μαζί με διάφορες παραλλαγές και αναγωγές σε παραδοσιακά μοντέλα υπολογισμού τα οποία διαθέτουν πρόσβαση σε ειδική πληροφορία (συμβουλή). Αποδεικνύονται κάποιες βασικές ιδιότητες και θεωρήματα των κυκλωμάτων

καθώς και κάποια χαρακτηριστικά παραδείγματα κάτω φραγμάτων που προκύπτουν από συνδυαστικές-τεχνικές ιδιότητες. Παράλληλα παρουσιάζεται ένας πολύ κρίσιμος αλγόριθμος δυναμικού προγραμματισμού που εκμεταλλεύεται κυκλωματικές ιδιότητες της κλάσης ACC^0 και που οδηγεί σε ένα από τα κεντρικότερα θεωρήματα του Κεφαλαίου 5.

- **Κεφάλαιο 4:** *Συσχετίσεις Υπολογιστών και Κυκλωμάτων*
  Παρουσιάζεται η στενή σύνδεση των δύο μοντέλων υπολογισμού μαζί με μία μεγάλη πληθώρα θεωρημάτων που γεφυρώνουν υποθέσεις του ενός με συμπεράσματα του άλλου. Ιδιαίτερη αναφορά γίνεται στη στενή σύνδεση των κυκλωματικά δύσκολων κλάσεων με την παραγωγή ψευδοτυχαιότητας και την αποτελεσματική αποτυχαιοποίηση. Στην τελευταία ενότητα παρουσιάζεται με τη βοήθεια των προηγούμενων, ο λόγος που μια καθαρά συνδυαστική τεχνική (όπως αυτές του προηγούμενου Κεφαλαίου) δεν είναι πιθανό να αποδώσει αποτελέσματα για τα μεγάλα ανοιχτά ερωτήματα.

- **Κεφάλαιο 5:** *Τρίτη Γενιά Αποδείξεων*
  Εισάγεται το γενικό μοτίβο της επονομαζόμενης Τρίτης Γενιάς Αποδείξεων, στην οποία γίνεται συνδυασμός των δύο τεχνικών που παρουσιάστηκαν στα προηγούμενα κεφάλαια με τρόπο τέτοιο ώστε να αλληλοσυμπληρώνονται τα εκφραστικά κενά. Επιπλέον παρουσιάζονται κάποια χαρακτηριστικά γενικά θεωρήματα που γεννάν αποτελέσματα κάτω φραγμάτων υπό κάποιες υποθέσεις, καθώς και η άνευ όρων τελική απόδειξη ότι $NEXP \not\subseteq ACC^0$.

- **Κεφάλαιο 6:** *Συμπεράσματα και Επεκτάσεις*
  Παρουσιάζονται μερικά γενικά συμπεράσματα που προκύπτουν από το προαναφερθέν μοτίβο καθώς και ορισμένες πιθανές κατευθύνσεις (μαζί με τα τρέχοντα εμπόδια) στις οποίες μπορούν να επεκταθούν τα προηγούμενα αποτελέσματα.

Η εργασία έχει δομηθεί έτσι, ώστε η αλληλουχία αποτελεσμάτων και κεφαλαίων να έχει τη μέγιστη δυνατή νοηματική και χρονολογική συνέπεια. Γενικά εμπεριέχονται όλα τα Θεωρήματα και Αποδείξεις που οδηγούν στα τελικά αποτελέσματα από πρακτικά μηδενική βάση, με εξαίρεση ορισμένα (κυρίως τεχνικά) Λήμματα που ξεφεύγουν τους σκοπούς της παρούσας εργασίας και τα οποία είτε αποδεικνύονται σε κάποιο Παράρτημα είτε απλά δίνεται σχετική αναφορά στη Βιβλιογραφία για τη πλήρη απόδειξη. Εν τούτοις θεωρείται ότι ο αναγνώστης είναι εξοικιωμένος με θεμελιώδεις έννοιες της Θεωρητικής Πληροφορικής, όπως βασικές μαθηματικές έννοιες (φυσικοί αριθμοί, συμβολοσειρές, συναρτήσεις, σχέσεις κ.α.), γενικές αποδεικτικές μεθόδους (αναδρομή, απαγωγή σε άτοπο κ.α.), πεδία (διακριτά μαθηματικά, πιθανότητες, θεωρία αριθμών κ.α.) καθώς και τους συμβολισμούς big-O. Σε κάθε περίπτωση, για όλα τα προηγούμενα υπάρχει πλούσια σχετική βιβλιογραφία τόσο σε έγγραφη όσο και σε ψηφιακή μορφή.

Κεφάλαιο 2

Υπολογιστική Πολυπλοκότητα

2.1 Μηχανές Turing

Στην προσπάθεια τυποποίησης της έννοιας του αλγορίθμου, ήδη από τις αρχές του 20ου αιώνα είχαν υπάρξει διάφορες προτάσεις και μοντέλα, τα οποία απαρτίζονταν από απλούς κανόνες και δομές και με αναδρομική χρήση αυτών των κανόνων (η οποία αντιστοιχούσε στην εκτέλεση του αλγορίθμου) σχηματιζόταν εν τέλει το υπολογιστέο αποτέλεσμα. Δύο χαρακτηριστικά, περισσότερο μαθηματικά, μοντέλα ήταν οι μ-αναδρομικές συναρτήσεις και ο λ-Λογισμός [Kle37, Chu85]. Αμφότερα αφορούν κυρίως την αναπαράσταση μερικώς ορισμένων συναρτήσεων από τους φυσικούς στους φυσικούς (αριθμούς). Παρότι η ισχύς τους ήταν ισοδύναμη του βέλτιστου (έως τώρα) ρεαλιστικού υπολογιστικού μοντέλου, εν τούτοις είχαν το μεγάλο μειονέκτημα ότι δεν ήταν διαισθητικώς φιλικά μοντέλα στην ανθρωποκεντρική θέαση της δομής ενός αλγορίθμου και η μοντελοποίηση των αντίστοιχων παραμέτρων σε αυτά τα πλαίσια ήταν συχνά αρκετά σύνθετη. Το πρόβλημα αυτό έλυσε ο Turing με την εισήγηση των ομώνυμων μηχανών, οι οποίες αποτέλεσαν το πρώτο μηχανιστικό μοντέλο υπολογισμού και τον πρώτο πλήρη πρόδρομο των σύγχρονων υπολογιστικών μηχανών. Η μεγάλη συνεισφορά των μηχανών Turing ήταν ότι για πρώτη φορά υπήρχε ένα μοντέλο γενικής χρήσης και δυνατοτήτων το οποίο προσομοίωνε ικανοποιητικά (και με τη δέουσα απλότητα) τις πράξεις που εκτελεί κι ένας άνθρωπος όταν εκτελεί κάποιον αλγόριθμο υπολογισμού: τη διαδοχική ανάγνωση και εγγραφή δεδομένων βάσει κάποιων συγκεκριμένων κανόνων μέχρι την εξαγωγή του επιθυμητού αποτελέσματος.

2.1.1 Ορισμός και Παραλλαγές

Ας δούμε, όμως, για λόγους πληρότητας πιο αυστηρά τον ορισμό μιας μηχανής Turing με μία ημι-άπειρη ταινία εργασίας (δηλαδή μία ταινία με άπειρα κελιά,

κάθε ένα εκ των οποίων μπορεί να περιέχει κάθε στιγμή ένα σύμβολο, και η οποία είναι φραγμένη από τη μία πλευρά (όπου κατ΄ αντιστοιχία με απαρίθμηση στους φυσικούς αριθμούς, θεωρούμε ότι δύο κελιά είναι γειτονικά όταν στην αντίστοιχη απαρίθμηση διαφέρουν κατά 1)):

Ορισμός 2.1.1 ([Tur36]). *(Μηχανή Turing)* Μια μηχανή Turing είναι μια τριάδα (Σ, δ, Q), όπου

- Σ είναι ένα πεπερασμένο αλφάβητο συμβόλων που αναγνωρίζει η μηχανή και περιλαμβάνει ένα ειδικό σύμβολο $\hookrightarrow$ (το οποίο βρίσκεται πάντα και αποκλειστικά στο πρώτο κελί της ταινίας) καθώς και το σύμβολο του κενού $\sqcup$.
- δ είναι η (μερική) συνάρτηση μετάβασης $Q \times \Sigma \to Q \times \Sigma \times \{\triangleleft, \triangleright, \diamond\}$
- Q είναι το σύνολο των καταστάσεων, το οποίο περιλαμβάνει μια ειδική αρχική κατάσταση q_{start} καθώς και μία κατάσταση αποδοχής q_{accept}.

Κάθε χρονική στιγμή η διαμόρφωση της μηχανής Turing είναι το σύνολο που περιλαμβάνει τη τρέχουσα κατάσταση, το τρέχον περιεχόμενο της ταινίας και τη θέση του τρέχοντος κελιού ή αλλιώς, όπως λέμε, τη τρέχουσα θέση της κεφαλής που διαβάζει/γράφει στην ταινία. Έτσι διαμόρφωση είναι μία τριάδα (q, T_l, T_r), όπου q η τρέχουσα κατάσταση της μηχανής, T_l η (πεπερασμένη) συμβολοσειρά που περιλαμβάνει όλα τα κελιά της ταινίας από την αρχή της μέχρι το τρέχον κελί που δείχνει η κεφαλή και T_r η (άπειρη) συμβολοσειρά που περιλαμβάνει όλα τα κελιά μετά (δεξιότερα, θεωρώντας ανάγνωση από αριστερά προς τα δεξιά) το τρέχον. Η συνάρτηση μετάβασης είναι αυτή που καθορίζει το πώς πράττει η μηχανή ανάλογα με τη τρέχουσα κατάσταση και το σύμβολο του τρέχοντος κελιού. Συγκεκριμένα, για μια μηχανή M, συμβολίζουμε $(q, T_l, T_r) \vdash_M (q', T_l', T_r')$ αν και μόνο αν $T_l = \alpha \circ u$ με $\alpha \in \Sigma^*, u \in \Sigma$ και επιπλέον $d(q, u) = (q', v, \star)$ όπου αν

1. $\star = \diamond$, τότε $T_l' = \alpha \circ v$ και $T_r' = T_r$,
2. $\star = \triangleleft$, τότε $T_l' = \alpha$ και $T_r' = vT_r$,
3. $\star = \triangleright$, τότε $T_l' = \alpha \circ v \circ w$ και $T_r' = T_r''$ όπου $T_r = w \circ T_r''$.

Θεωρούμε ως αρχική διαμόρφωση την $c_0 = (q, \hookrightarrow, Input)$ όπου $Input$ η συμβολοσειρά που περιέχει την είσοδο μήκους n στα πρώτα n κελιά της ταινίας και όλα τα επόμενα να περιέχουν το σύμβολο κενού $\sqcup$. Επιπλέον, επειδή το σύμβολο $\hookrightarrow$ έχει το ρόλο του τερματικού από τα αριστερά συμβόλου, επιτρέπουμε μόνο κανόνες της μορφής $(q, \hookrightarrow) \to (q', \hookrightarrow, \triangleright)$ και δεν επιτρέπεται εγγραφή του συμβόλου αυτού σε οποιοδήποτε άλλο σημείο της ταινίας, όπως και δεν αποτελεί ποτέ τμήμα της συμβολοσειράς εισόδου. Όπως είδαμε, επειδή η δ είναι (μερική) συνάρτηση, κάθε διαμόρφωση έχει το πολύ μία άλλη στην οποία μπορεί να μεταβεί. Όταν δεν υπάρχει τέτοια διαμόρφωση, θεωρούμε ότι η εκτέλεση

της μηχανής τερματίζει. Θεωρούμε ότι η μηχανή με είσοδο x τερματίζει αν και μόνο αν υπάρχει διαμόρφωση $c_{fin} = (q_{fin}, Fl, F_r)$ τέτοια ώστε $c_0 \vdash_M^* c_{fin}$ (όπου $\vdash_M^*$ η μεταβατική κλειστότητα της $\vdash_M$) και επιπλέον $c_{fin} \not\vdash c$ για κάθε διαμόρφωση c. Αν επιπλέον $q_{fin} = q_{accept}$, τότε θεωρούμε ότι η M αποδέχεται την είσοδο x, ενώ αλλιώς ότι την απορρίπτει. Σε περίπτωση, που δεν υπάρχει κάποια τέτοια c_{fin}, λέμε ότι η M δε τερματίζει με είσοδο x.

2.1.2 Υπολογισιμότητα

Η παραπάνω σχετικά λιτή δομή έχει το πλεονέκτημα της αυξημένης απλότητας, εγείρει ωστόσο ερωτήματα στο κατά πόσο είναι επαρκής υπολογιστικά και αν πιθανόν αυξάνεται η υπολογιστική ισχύς αν επιτρέψουμε περισσότερες δυνατότητες. Ορισμένες από αυτές παρατίθενται στη συνέχεια, μαζί με σύντομες διαισθητικές επεξηγήσεις για ποιο λόγο δεν προσφέρουν κάτι παραπάνω από την προηγούμενη:

- **Περισσότερες καταστάσεις Αποδοχής (ή Απόρριψης)** Δημιουργούμε μία νέα στην οποία να κατευθύνονται όλες οι υπόλοιπες (ή φτιάχνουμε μία κατάσταση-καταβόθρα που προσομοιώνει αυτή της απόρριψης).

- **$k > 1$ ταινίες εργασίας** Αποθηκεύουμε το j-οστό κελί της i-οστής ταινίας στο $(j * k + i)$-οστό κελί της μοναδικής ταινίας και προσαρμόζουμε τους κανόνες αντίστοιχα. Ακόμη και για άπειρες ταινίες (δισδιάστατο πλέγμα αντί για ταινία) κάνουμε μια αντιστοίχιση περιστερoουράς (όμοια με αυτή από το $\mathbb{N}^2$ στο $\mathbb{N}$).

- **$k > 1$ κεφαλές** Αντικαθιστούμε κάθε κεφαλή με μία τιμή-δείκτη στη θέση που δείχνει και ρυθμίζουμε τις υπόλοιπες λειτουργίες αντίστοιχα.

- **Τυχαία Πρόσβαση στα κελιά της ταινίας** Αντίστοιχα με προηγουμένως, προσομοιώνουμε την αντίστοιχη μετατόπιση της κεφαλής στο τυχαίο σημείο που ορίζεται από το πρόγραμμα μέσω κάποιου δείκτη.

Όλες αυτές οι παραλλαγές εν τέλει προκύπτουν, ως προς την εκφραστικότητα, ισοδύναμες με το αρχικό μοντέλο που παρουσιάσαμε παραπάνω [1]. Ακόμη περισσότερο, τα δύο αρχικά μοντέλα των αναδρομικών συναρτήσεων και του λ-Λογισμού (αλλά και παραλλαγές αυτών) προέκυψαν ισοδύναμα των δυνατοτήτων της μηχανής Turing . Συγκεκριμένα λέμε ότι η γλώσσα L υπολογίζεται από μια μηχανή Turing M αν η M τερματίζει για κάθε είσοδο x και αποδέχεται μόνο όταν $x \in L$ (αν υπάρχει τέτοια μηχανή λέμε ότι η L είναι υπολογίσιμη).

[1] Γενικότερα, επειδή εύκολα γίνεται εμφανές ότι στοιχειώδεις λειτουργίες που αφορούν προσομοίωση εκτέλεσης [Pap94, p.36–45] , πολλαπλή εγγραφή/ανάγνωση κ.λπ. μπορούν εν τέλει να περιγραφούν από μια τέτοια παραλλαγμένη μηχανή (κι επομένως και από μια κλασική μηχανή), στα επόμενα δε θα προσκολλάμε στη λεπτομερή περιγραφή του τρόπου που αυτό γίνεται και συνήθως θα περιγράφουμε σε υψηλό επίπεδο τον αντίστοιχο αλγόριθμο.

Η προηγούμενη δήλωση λοιπόν μας εγγυάται για αρχή ότι όλες οι αντίστοιχες παραλλαγές της μηχανής Turing μπορούν να υπολογίσουν το ίδιο σύνολο υπολογίσιμων γλωσσών όπως ο αρχικός ορισμός. Σε ένα πιο γενικό πλαίσιο (όπου π.χ. λαμβάνουμε υπόψιν υπολογισμούς που τερματίζουν μόνο όταν $x \in L$ κ.ο.κ.), έχουμε την κάτωθι θέση, η οποία συνοψίζει τη πίστη μας ότι οι μηχανές Turing αποτελούν το έσχατο υπολογιστικό σύστημα (που να κινείται σε ρεαλιστικά-εφαρμόσιμα πλαίσια) και ότι δεν υπάρχει γενίκευση του που να προσφέρει επιπλέον υπολογιστική ισχύ:

Θέση ([Tur36])**.** *(Church-Turing) Κάθε ρεαλιστικό και πρακτικά εκτελέσιμο υπολογιστικό μοντέλο φυσικών συναρτήσεων είναι ισοδύναμο με μία μηχανή Turing.*

Η ασάφεια της παραπάνω Θέσης σε επί μέρους όρους όπως «ρεαλιστικό», «πρακτικά εκτελέσιμο» και «υπολογιστικό» είναι κατά κύριο λόγο δημιουργική κι επιτήδεια, εφόσον η ίδια η φύση του υπολογισμού είναι κάτι για το οποίο δεν έχουμε αυστηρό ορισμό (κυρίως επειδή δε γνωρίζουμε με αυστηρή απόδειξη τα όρια του – και για διάφορους λόγους πιθανότατα δε μπορούμε να τα βρούμε ποτέ). Επομένως ακολουθώντας την παραπάνω Θέση, θεωρούμε ως υπολογίσιμες όσες συναρτήσεις έχουν μια μηχανή Turing που να τις υπολογίζει.

Σημείωση. Εμείς είδαμε επίσημα μηχανές Turing οι οποίες μόνο αποδέχονταν ή απέρριπταν μια είσοδο ανάλογα με την τελική κατάσταση. Εννοείται ότι αυτό μπορούμε να το επεκτείνουμε σε συναρτήσεις $\Sigma^* \to \Sigma^*$ θεωρώντας ως έξοδο της μηχανής τη συμβολοσειρά που βρίσκεται στη ταινία όταν η μηχανή καταλήγει στην q_{accept}.

Εδώ είναι ένα καλό σημείο να σημειώσουμε ότι όλα αυτά τα υπολογιστικά μοντέλα ήταν άρρηκτα συνδεδεμένα με μια μαζική προσπάθεια στις αρχές του 20ου αιώνα για προτυποποίηση των αποδεικτικών μεθόδων και εξεύρεση ενός γενικού μοντέλου το οποίο θα παρήγαγε κάθε δυνατή απόδειξη, στα χνάρια του αντίστοιχου αιτήματος του Hilbert στη γνωστή ομιλία του στο γνωστό συνέδριο. Σε συστοιχία όμως με το Θεώρημα Πληρότητας του Gödel , οι μηχανές Turing ήταν υπολογιστικά τόσο ισχυρές σε σημείο που να μη μπορούν να αποδείξουν πληροφορίες για τις ίδιες (ακριβώς όπως ένα αποδεικτικό σύστημα (που περιλαμβάνει την αριθμητική) δε μπορεί να αποδείξει την ορθότητα του μέσα στο ίδιο).

Κατ΄ αρχήν υπάρχει ένα στέρεο επιχείρημα που μας πείθει ότι οι μηχανές Turing δε μπορούν να υπολογίσουν όλες τις $\mathbb{N} \to \{0,1\}$ συναρτήσεις (όπου στα ακόλουθα με 0 συμβολίζουμε την απόρριψη και με 1 την αποδοχή). Επειδή κάθε μηχανή Turing καθορίζεται μονοσήμαντα από τις καταστάσεις της, το πεπερασμένο αλφάβητο και τη συνάρτηση μετάβασης (επίσης πεπερασμένης περιγραφής λόγω του πεπερασμένου πεδίου ορισμού), σημαίνει ότι μπορεί να αντιστοιχιστεί σε μία πεπερασμένη συμβολοσειρά (πάνω σε ένα πεπερασμένο αλφάβητο). Επειδή το σύνολο των πεπερασμένων συμβολοσειρών είναι αριθμήσιμο και των $\mathbb{N} \to \{0,1\}$ συναρτήσεων υπεραριθμήσιμο, προκύπτει ότι υπάρχει τουλάχιστον

μία λογική συνάρτηση που δεν περιγράφεται από καμία μηχανή Turing (για την ακρίβεια το παραπάνω επιχείρημα οδηγεί στο συμπέρασμα ότι όλες εκτός από ένα αμελητέο ποσό δεν υπολογίζονται από κάποια μηχανή Turing).

Αυτό είναι το πρώτο χαρακτηριστικό παράδειγμα όπου κάποιο (σχετικά απλό) επιχείρημα μας εξασφαλίζει την ύπαρξη μιας «δύσκολης» συνάρτησης (όπου εν προκειμένω η δυσκολία ορίζεται από τη μη υπολογισιμότητα), αλλά η εύρεση ενός συγκεκριμένου παραδείγματος ή η απόδειξη ότι κάποια δεδομένη συνάρτηση είναι δύσκολη είναι αρκετά πιο περίπλοκο ζήτημα (και στις περισσότερες περιπτώσεις που ακολουθούν, συνήθως μέχρι σήμερα ανεπίλυτο). Η δυσκολία έγκειται στο γεγονός, ότι ενώ η απόδειξη πως μια συνάρτηση είναι εύκολη έχει συνήθως κάποιο απλό επιχείρημα (π.χ. δίνοντας μια μηχανή Turing που την υπολογίζει και αποδεικνύοντας με κάποια αναδρομικά επιχειρήματα την ορθότητα της), το να αποδειχθεί πως είναι δύσκολη απαιτεί ένα επιχείρημα που περιλαμβάνει αλλά και απορρίπτει όλες τις πιθανές λύσεις (εν προκειμένω ένα επιχείρημα που να αποκλείει κάθε μηχανή Turing από το να την υπολογίζει ορθά για κάθε τιμή).

Έχοντας τα παραπάνω κατά νου, προκύπτει ότι ένα εύλογο επιχείρημα θα πρέπει να περιλαμβάνει μια μηχανή η οποία να αναφέρεται σε όλες τις μηχανές με τρόπο τέτοιο ώστε να μη μπορεί να ισούται με καμία (καθώς αλλιώς θα δημιουργούσε μια αντιφατική αυτοαναφορά). Η μέθοδος αυτή, η οποία είναι γνωστή ως διαγωνιοποίηση (καθώς από κάθε στοιχείο του συνόλου που θέλουμε να αποκλείσουμε παίρνουμε το στοιχείο της διαγωνίου και το αντιστρέφουμε – με τρόπο όμοιο της μεθόδου που ξεχωρίζει τα υπεραριθμήσιμα σύνολα από τα αριθμήσιμα) έχει μεγάλη επιτυχία στην απόδειξη διαφόρων αποτελεσμάτων στον τομέα της υπολογισιμότητας και, όπως θα δούμε στην επόμενη ενότητα, και σε κάποια θεμελιώδη ιεραρχικά αποτελέσματα της πολυπλοκότητας.

Μία πρώτη εφαρμογή αυτής της μεθόδου, φαίνεται στο παρακάτω κλασικό παράδειγμα μιας ρητής συνάρτησης που δε μπορεί να υπολογιστεί από καμία μηχανή Turing .

Θεώρημα 2.1 ([Tur36]). *Το Πρόβλημα Τερματισμού είναι μη υπολογίσιμο. (Όπου το Πρόβλημα Τερματισμού είναι η συνάρτηση που με είσοδο τη περιγραφή μιας μηχανής M και μια συμβολοσειρά x, το αποτέλεσμα είναι αποδοχή αν και μόνο αν η M τερματίζει με είσοδο x.).*

Απόδειξη. Πράγματι, έστω μία μηχανή Turing H που υπολογίζει το Πρόβλημα Τερματισμού, δηλαδή (με κατάχρηση συμβολισμού) $H(\langle "M", x\rangle) = 1 \Leftrightarrow M(x)\downarrow$ (όπου κλασικά το $\downarrow$ δηλώνει τερματισμό και το $\uparrow$ μη τερματισμό). Φτιάχνουμε τη μηχανή $G("M") = H(\langle "M", "M"\rangle)$, δηλαδή αυτήν που με είσοδο μια περιγραφή $"M"$ προσομοιώνει την H με είσοδο το $\langle "M", "M"\rangle$ (προφανώς μπορούμε να κατασκευάσουμε μια τέτοια μηχανή). Τέλος φτιάχνουμε τη μηχανή $D("M")$ η οποία προσομοιώνει το $G("M")$ κι αν αυτό αποδεχτεί, τότε τρέχει επ άπειρον, ενώ αν όχι τερματίζει αποδέχοντας (και πάλι προφανές ότι μπορούμε να κατασκευάσουμε την D δοθείσης της G). Έχουμε δηλαδή ότι $D("M")\downarrow \Leftrightarrow G("M") = 0 \Leftrightarrow M("M")\uparrow$. Δίνοντας, όμως, ως είσοδο στην

D τη περιγραφή του εαυτού της, έχουμε τελικά ότι $D(\text{"}D\text{"}) \downarrow \Leftrightarrow D(\text{"}D\text{"}) \uparrow$ καταλήγοντας σε άτοπο. ■

Με ανάλογες αποδείξεις προκύπτει ότι οποιαδήποτε μη τετριμμένη ιδιότητα που αναφέρεται σε ολόκληρη τη γλώσσα αποδοχής μιας μηχανής Turing είναι μη υπολογίσιμη. Το σημείο-κλειδί όλων αυτών των μη υπολογίσιμων συναρτήσεων είναι η εμπλοκή μιας ιδιότητας η οποία εν τέλει δε τερματίζει. Δηλαδή είναι εγγενές χαρακτηριστικό όλων των πλήρων (κατά Church-Turing) υπολογιστικών μοντέλων να περιλαμβάνουν μερικές συναρτήσεις, δηλαδή που δεν ορίζονται για κάθε είσοδο (όπου στη περίπτωση των μηχανών Turing αυτό ισοδυναμεί με μη τερματισμό). Πράγματι, όλα αυτά τα μοντέλα έχουν κι από ένα περιορισμένο μοντέλο, το οποίο τερματίζει πάντα και που μάλιστα έχει αρκετά μεγάλη υπολογιστική ισχύ (μηχανές Turing με χρονιστή, πρωταρχικές αναδρομικές συναρτήσεις, ελαφρύς λ-Λογισμός κ.ο.κ.). Το πρόβλημα όλων αυτών είναι ότι μπορούμε να κατασκευάσουμε ένα διαγώνιο υπολογισμό που απορρίπτει τη δυνατότητα να περιγράφεται από οποιαδήποτε από αυτές [MM11]. Ο υπολογισμός αυτός, όμως, είναι παντού καλώς ορισμένος και παρόλα αυτά δε μπορεί να περιγράφεται από καμία από αυτές τις μηχανές. Αυτό είναι ένα κρίσιμο σημείο, καθώς ακριβώς εκεί είναι που αποδεχόμαστε ότι τα υπομοντέλα αυτά δεν είναι πλήρη μόνο και μόνο επειδή έχουμε μια ντετερμινιστική διαδικασία η οποία μπορεί να υπολογίσει κάτι που δεν περιγράφεται από αυτά (και ο μόνος λόγος που προχωρήσαμε σε κάποιο «ψηλότερο» μοντέλο είναι επειδή το προηγούμενο δε ταίριαζε στον καθαρά διαισθητικό και καθόλου μαθηματικά αυστηρό ορισμό του τι θεωρούμε υπολογίσιμο – ένα γεγονός που κάνει πιο φανερό τον (ανθρωπο-)κεντρικό ρόλο που παίζει η Θέση Church-Turing σε αυτόν τον τομέα των Μαθηματικών).

2.2 Υπολογιστικές Κλάσεις

2.2.1 Εναλλακτικά Μοντέλα Υπολογισμού

Ως φυσική απόρροια της τελευταίας παρατήρησης της προηγούμενης ενότητας, ήταν αναμενόμενο ότι πέραν του χαρακτηρισμού των γλωσσών ως υπολογίσιμες ή όχι, μας ενδιαφέρει και το πόσους πόρους απαιτεί αυτός ο υπολογισμός. Με πόρους εννοούμε τον χρόνο (δηλαδή πλήθος εφαρμογών της συνάρτησης μετάβασης (βήματα) μέχρις ότου να τερματίσει το πρόγραμμα), τον χώρο (πόσα κελιά της ταινίας (πέραν της εισόδου και εξόδου) χρησιμοποιεί το πρόγραμμα για τον υπολογισμό του), αλλά και άλλα στοιχεία σε παραλλαγές της μηχανής Turing που θα μελετήσουμε στη συνέχεια (όπως μήκος τυχαίων συμβολοσειρών, μήκος μη-ντετερμινιστικών συμβολοσειρών, μήκος συμβολοσειρών συμβουλής, πλήθος κλήσεων σε μαντεία κ.ο.κ.).

Ορισμός 2.2.1. Ορίζουμε λοιπόν τη κλάση $DTIME[f(n)]$[2] ως το σύνολο

[2]Για συγκεκριμένους τεχνικούς λόγους, η f επιτρέπουμε να είναι χρονικά κατασκευάσιμη

των γλωσσών που είναι υπολογίσιμες σε χρόνο $O(f(n))$, δηλαδή που υπάρχει μηχανή Turing που την υπολογίζει και για κάθε είσοδο x απαιτεί το πολύ $O(f(|x|))$ βήματα μέχρι τον τερματισμό της.

Θέλοντας έναν γενικό χαρακτηρισμό των χρονικά εφικτών υπολογίσιμων γλωσσών, οι οποίες να μη περιορίζονται σε μία απλή μικρή συνάρτηση, ορίστηκε η κλάση P των γλωσσών πολυωνυμικού χρόνου, η οποία για διάφορους τεχνικούς λόγους, θεωρείται ως η αντιπροσωπευτικότερη γενική κλάση των υπολογιστικά εφικτών γλωσσών. Η πεποίθηση αυτή, μάλιστα είναι τόσο ευρεία, σε σημείο που το μεγάλο ζητούμενο για το μεγαλύτερο ποσοστό γλωσσών που θα εξετάσουμε είναι να δειχθεί ότι ανήκουν στο P (ή σε κάποια αντίστοιχη υπολογιστική δομή πολυωνυμικών παραμέτρων, π.χ. στο $P_{/poly}$ που θα ορίσουμε στο ακόλουθο Κεφάλαιο). Για τις γλώσσες αυτές δεν υπάρχει ζήτημα υπολογισιμότητας, καθώς για όλες θα υπάρχει κάποιος τετριμμένος αλγόριθμος, που θα είναι όμως εκθετικού χρόνου και κατά βάση δε θα επιδέχεται οποιουδήποτε πρακτικού υπολογισμού σε κάποιο τρέχον ρεαλιστικό μοντέλο! Ορίζονται λοιπόν οι ακόλουθες χαρακτηριστικές κλάσεις:

Ορισμός 2.2.2. Ορίζουμε:

- $P = \bigcup_{c>0} DTIME[n^c]$
- $E = \bigcup_{c>0} DTIME[2^{cn}]$
- $EXP = \bigcup_{c>0} DTIME[2^{n^c}]$
- $quasiP = \bigcup_{c>0} DTIME[n^{log^c n}]$
- $subEXP = \bigcap_{c>0} DTIME[2^{n^c}]$

Σημείωση. Οι παραπάνω κλάσεις αφορούν λογικές συναρτήσεις, αλλά επεκτείνονται με φυσικό τρόπο και για φυσικές συναρτήσεις (όπου ορίζονται αντίστοιχα οι κλάσεις $FP, FEXP$ κ.ο.κ.). Επίσης πάντοτε σε τέτοια πλαίσια θα θεωρούμε ότι η σταθερά c είναι τέτοια ώστε η αντίστοιχη συνάρτηση να είναι κατασκευάσιμη.

Περιορίζοντας τώρα τον χώρο, ορίζουμε αντίστοιχα

Ορισμός 2.2.3. Μια γλώσσα L ανήκει στη κλάση $(D)SPACE[f(n)]$ αν και μόνο αν υπάρχει υπάρχει μηχανή Turing με τρεις ταινίες: μία μόνο ανάγνωσης που τοποθετείται η είσοδος, μία μόνο εγγραφής (που τοποθετείται τυχόν έξοδος) και μία ταινία εργασίας ανάγνωσης/εγγραφής, που να υπολογίζει την L χρησιμοποιώντας μόνο $O(f(n))$ κελιά από την ταινία εργασίας.

αύξουσα φυσική συνάρτηση, δηλαδή τέτοια που να υπάρχει μηχανή Turing που με είσοδο το 1^n να παράγει σε $O(f(n))$ χρόνο το $1^{f(n)}$. Στα ακόλουθα, θα θεωρούμε ότι η f είναι πάντα κατασκευάσιμη (εκτός κι αν αναφέρεται ρητά αλλιώς).

Ορίζονται με το φυσικό τρόπο, οι γλώσσες πολυωνυμικού χώρου που απαρτίζουν τη κλάση $PSPACE$, αλλά εν προκειμένω πολύ μεγάλη σημασία έχουν και οι γλώσσες υπογραμμικού και συγκεκριμένα λογαριθμικού χώρου, οι οποίες απαρτίζουν την κλάση L. Προφανώς δεν υπάρχει λόγος να οριστούν μη τετριμμένες ντετερμινιστικές κλάσεις υπογραμμικού χρόνου, αφού μόνο η ανάγνωση της εισόδου απαιτεί γραμμικό χρόνο, αλλά αυτό δεν ισχύει για την περίπτωση του χώρου, καθώς αυτός μπορεί να ξαναχρησιμοποιηθεί και όπως θα δούμε στη συνέχεια η κλάση L είναι ιδιαίτερα εκφραστική ιδιαίτερα στο κομμάτι των αναγωγών.

Επιστρέφοντας, ένας από τους λόγους που υπήρξε η προσπάθεια προτυποποίησης των αλγοριθμικών μεθόδων ήταν και η προσπάθεια αυτοματοποιημένης παραγωγής και ελέγχου αποδείξεων. Παρότι ο έλεγχος των αποδείξεων (ενός δεδομένου αξιωματικού συστήματος), μπορεί να γίνει εύκολα από μια μηχανή Turing , εν τούτοις η παραγωγή τους φαίνεται να αποτελεί μια πολύ πιο δύσκολη εργασία, καθώς ο χώρος των πιθανών μονοπατιών που μπορεί να πάρει μια απόδειξη δεδομένου μήκους είναι εκθετικός (κάτι που είναι αποτρεπτικό από το να είναι εφικτά υπολογίσιμος με εξαντλητική αναζήτηση). Εν τούτοις, όλα τα ορθά (ως προς το αποδεικτέο) μονοπάτια έχουν ένα κοινό χαρακτηριστικό: όταν βρεθούν, μπορούν να ελεγχθούν για την ορθότητα τους αποδοτικά και αυτοματοποιημένα.

Αυτό το μοτίβο οδήγησε στο να δημιουργηθεί μια νέα κλάση, η οποία αποτυπώνει ακριβώς αυτή τη δυνατότητα:

Ορισμός 2.2.4. Λέμε ότι μία γλώσσα L είναι στο $NTIME[f(n)]$, ακριβώς όταν υπάρχει μία μηχανή[3] M στο $DTIME[n]$ τέτοια ώστε $x \in L$ αν και μόνο αν $\exists y(|y| \leq f(|x|) \wedge M(\langle x, y\rangle) = 1)$.

Το y στον παραπάνω λόγο αναπαριστά την απόδειξη για το ότι $x \in L$ και η M δηλώνει τη μηχανή που ελέγχει αυτοματοποιημένα αν είναι ορθή η απόδειξη για το συγκεκριμένο x (σε χρόνο γραμμικό ως προς την είσοδο που είναι μήκους $f(|x|)$ κι άρα σε συνολικό χρόνο $O(f(n))$). Το y λοιπόν, συχνά αποκαλείται πιστοποιητικό και είναι αυτό που πρέπει να μαντέψουμε για να δείξουμε ότι $x \in L$. Αυτή η έννοια μπορεί να ενσωματωθεί στην ακόλουθη παραλλαγή της μηχανής Turing :

Ορισμός 2.2.5. Μία μη ντετερμινιστική μηχανή Turing είναι μία τριάδα (Σ, Δ, Q), όπου η μόνη διαφορά από τη ντετερμινιστική είναι ότι το Δ δεν είναι πλέον συνάρτηση, αλλά σχέση, δηλαδή είναι υποσύνολο του $(\Sigma \times Q) \times (\Sigma \times Q \times \{\triangleleft, \triangleright, \diamond\})$.

Πλέον η σχέση $\vdash_M$ δεν είναι ντετερμινιστική, με την έννοια ότι από κάθε διαμόρφωση υπάρχουν ενδεχομένως περισσότερες από μία έγκυρες επόμενες

[3]Για τον ακριβή ορισμό πρέπει να αναφέρουμε την ύπαρξη μιας γλώσσας $L \in DTIME[n]$ με μηχανή M_L που την υπολογίζει. Εν τούτοις, χάριν απλότητας, θα κάνουμε συχνά κατάχρηση αυτής της φρασεολογίας.

διαμορφώσεις. Θεωρούμε ότι μία μη ντετερμινιστική μηχανή Turing αποδέχεται μία γλώσσα L αν και μόνο αν για κάθε $x \in L$ υπάρχει ορθή ακολουθία διαμορφώσεων (μήκους όσου και του χρόνου που κάνει η μη ντετερμινιστική μηχανή) που να καταλήγει στην q_{accept}. Η ακολουθία αυτή αντιστοιχεί κατά βάση σε μια κωδικοποίηση του πιστοποιητικού y την οποία μαντεύουμε μη ντετερμινιστικά (αν υπάρχει) και μετά επαληθεύουμε την ορθότητα της. Συγκεκριμένα ισχύει η ακόλουθη πρόταση:

Πρόταση 2.2.1 ([AB09, p.41–42]). *Μία γλώσσα L είναι στο $NTIME[f(n)]$ αν και μόνο αν υπολογίζεται από μία μη ντετερμινιστική μηχανή χρόνου $O(f(n))$.*

Αντίστοιχα με προηγουμένως ορίζονται οι κλάσεις NP, NE, $NEXP$, $subNEXP$, κ.ο.κ. Να σημειωθεί ότι ο μη ντετερμινισμός δε προσφέρει κάποια επιπλέον υπολογιστική ισχύ, εφόσον εύκολα δείχνεται ότι $NTIME[f(n)] \subseteq DTIME[c^{f(n)}]$. Εν τούτοις αποτελεί το πρώτο εναλλακτικό μοντέλο που δεν έχει πολυωνυμική αντιστοίχιση με κάποια παραδοσιακή μηχανή (σε αντίθεση με τις παραλλαγές περισσότερων ταινιών/κεφαλών που είδαμε μέχρι στιγμής). Συγκεκριμένα, ένα από τα μεγαλύτερα ανοιχτά προβλήματα της Θεωρητικής Πληροφορικής είναι κατά πόσο ο μη ντετερμινισμός μπορεί να προσομοιωθεί αποδοτικά από μία ντετερμινιστική μηχανή, ή με άλλα λόγια κατά πόσο ισχύει ότι $P = NP$. Από την επιστημονική κοινότητα, το πιο πιθανό θεωρείται ότι είναι το $P \subsetneq NP$ καθώς σε διαισθητικό επίπεδο, μία από τις πολλές συνέπειες του αντίθετου είναι πως αν υπάρχει απόδειξη για κάποια πρόταση μήκους N, τότε αυτή μπορεί να παραχθεί αυτοματοποιημένα σε χρόνο $poly(N)$ (κάτι που για διαισθητικά προφανείς λόγους δεν είναι το αναμενόμενο). Παρόλα αυτά μέχρι και σήμερα δεν έχει υπάρξει απόδειξη ότι $P \neq NP$ και στα κεφάλαια που ακολουθούν θα δούμε ορισμένα από τα κεντρικότερα σημεία της πορείας αυτών των προσπαθειών.

Όσο λοιπόν δε γνωρίζουμε την ακριβή σχέση του NP με το P, είναι εύλογο να δοκιμάσουμε στη θέση της μηχανής που ελέγχει το πιστοποιητικό να βάλουμε μία επίσης μη ντετερμινιστική μηχανή και αναδρομικά να χτίσουμε έτσι μια ιεραρχία κλάσεων:

Ορισμός 2.2.6. Ορίζουμε

$$\Sigma_0 TIME[f(n)] = \Pi_0 TIME[f(n)] = DTIME[f(n)]$$

και ορίζουμε αναδρομικά:

$L \in \Sigma_{i+1} TIME[f(n)]$ αν και μόνο αν υπάρχει $L' \in \Pi_i TIME[n]$ τέτοια ώστε $x \in L \Leftrightarrow \exists y(|y| \leq f(|x|) \wedge \langle x, y \rangle \in L')$

και $\Pi_{i+1} TIME[f(n)] = co\Sigma_{i+1} TIME[f(n)]$
όπου $L \in coC \Leftrightarrow \exists L' \in C(x \in L \Leftrightarrow x \notin L')$

Εύκολα μπορούμε να δούμε ότι ένας εναλλακτικός ορισμός της κλάσης $\Pi_{i+1} TIME[f(n)]$ είναι να υπάρχει $L' \in \Sigma_i TIME[n]$ τέτοια ώστε $x \in L \Leftrightarrow$

$\forall y(|y| \leq f(|x|) \Rightarrow \langle x, y\rangle \in L')$. Σχετικές κλάσεις είναι οι Σ_i^P και Σ_i^{EXP} που ορίζονται με τον προφανή τρόπο. Συγκεκριμένα αυτή η εναλλαγή ποσοδεικτών στον ορισμό δίνει έναν εύγλωττο τρόπο χαρακτηρισμού των αντίστοιχων κλάσεων, ορίζοντας την αλληλουχία ποσοδεικτών που εφαρμόζονται στα επί μέρους πιστοποιητικά για όταν $x \in L$ και για όταν $x \notin L$ [Zac86]. Έτσι για παράδειγμα η NP χαρακτηρίζεται από το ζεύγος $(\exists, \forall)$, η $coNP$ από το $(\forall, \exists)$, η $\Sigma_2 P$ από το $(\exists\forall, \forall\exists)$ κ.ο.κ.

Εύκολα φαίνεται ότι $V_i^P \subseteq V_{i+1}^P$ για κάθε i (όπου V_i^P οποιαδήποτε από τις κλάσεις Σ_i^P, Π_i^P) κι έτσι ορίζεται μια ιεραρχία, η οποία ονομάζεται Πολυωνυμική Ιεραρχία:

Ορισμός 2.2.7 ([Sto76])**.** Ορίζουμε ως Πολυωνυμική Ιεραρχία την κλάση

$$PH = \bigcup_{i=0}^{\infty} \Sigma_i^P$$

Είναι εύκολο να δούμε ότι $PH \subseteq PSPACE$ (απλώς δοκιμάζουμε διαδοχικά σε πολυωνυμικό χώρο (κι εκθετικό χρόνο) όλους τους δυνατούς συνδυασμούς πιστοποιητικών και σε πολυωνυμικό χρόνο (κι άρα χώρο) επαληθεύουμε ή απορρίπτουμε για κάθε συνδυασμό). Επίσης σε περίπτωση που $P = NP$, τότε όλη η PH καταρρέει στο μηδενικό επίπεδο, δηλαδή $PH = P$ (προκύπτει άμεσα από τους ορισμούς). Παρότι δεν έχει βρεθεί απόδειξη για το αντίστροφο, εν τούτοις για λόγους παρόμοιους με την περίπτωση των P και NP, πιστεύεται ότι η PH είναι γνήσια και δε καταρρέει σε κανένα επίπεδο. Αντίστοιχα ορίζεται και η εκθετική ιεραρχία ως η ένωση όλων των Σ_i^{EXP}.

Όπως είπαμε, ο μη ντετερμινισμός δε φαίνεται να έχει άμεση (χρονική) σύνδεση με κάποιο ντετερμινιστικά εφικτό ισοδύναμο μοντέλο και επομένως ο μη ντετερμινισμός έχει κατά βάση θεωρητικό ενδιαφέρον. Ένα εναλλακτικό μοντέλο, όμως, το οποίο φαίνεται να έχει περισσότερες πιθανότητες να αντιστοιχεί σε κάποιο υπαρκτό υπολογιστικό μοντέλο (και που μέχρι στιγμής δεν έχει απόδειξη ότι είναι πολυωνυμικά ισοδύναμο με μια παραδοσιακή μηχανή Turing) είναι οι τυχαιοκρατικές μηχανές, οι οποίες έχουν πρόσβαση σε μία σειρά τυχαίων δυφίων. Στις περιπτώσεις αυτές επιτρέπουμε μια χαλάρωση στην ορθότητα ή στο χρόνο που χρησιμοποιεί η μηχανή, αλλά με μεγάλη πιθανότητα το αποτέλεσμα να είναι εν τέλει το επιθυμητό. Πιο αυστηρά, έχουμε τον ακόλουθο ορισμό:

Ορισμός 2.2.8. Μία γλώσσα L είναι στο $BPTIME[f(n)]$ αν υπάρχει $L' \in DTIME[n]$ και ισχύει ότι $x \in L$ αν και μόνο αν $\Pr_{y:|y|<f(|x|)}(\langle x, y\rangle \in L') \geq \frac{2}{3}$ και $x \notin L$ αν και μόνο αν $\Pr_{y:|y|<f(|x|)}(\langle x, y\rangle \in L') \leq \frac{1}{3}$

Η αντίστοιχη πολυωνυμική τυχαιοκρατική κλάση είναι η BPP. Να σημειωθεί ότι η σταθερά $\frac{2}{3}$ στον παραπάνω ορισμό είναι αυθαίρετη εφόσον με εφαρμογή φραγμάτων Chernoff μπορεί το σφάλμα να γίνει από $\frac{1}{2} - \frac{1}{poly(n)}$ μέχρι και $2^{-poly(n)}$ (όπου n το μήκος της εισόδου x), χωρίς να αλλοιώνεται η ιδιότητα

του πολυωνυμικού χρόνου [Kab03]. Από εδώ και στο εξής, όταν αναφέρουμε τους όρους «με καλή πιθανότητα» ή «στη μεγάλη πλειοψηφία» θα αναφερόμαστε σε αυτά τα όρια. Ορίζοντας, λοιπόν, τον ποσοδείκτη $\exists^+$ ως αυτόν που δηλώνει την ύπαρξη πολλών πιστοποιητικών (όπου το «πολλών» συμβαδίζει με όσα μόλις είπαμε), τότε έχουμε ότι το BPP χαρακτηρίζεται από το ζεύγος $(\exists^+, \exists^+)$. Αποδεικνύεται ότι $BPP \subseteq \Sigma_2^P$ [Sip83, Lau83], αλλά παραμένει άγνωστη μέχρι στιγμής η ακριβής σχέση του με το NP όπως επίσης και με το P. Να σημειωθεί, επίσης, ότι μπορούμε να πάρουμε τις *promise* εκδοχές των παραπάνω τυχαιοκρατικών κλάσεων (ορίζοντας την $promiseBPP$ κ.ο.κ.) όπου στον παραπάνω ορισμό θέτουμε η μηχανή να έχει την αντίστοιχη γερή τυχαιοκρατική συμπεριφορά μόνο για ένα υποσύνολο $\mathcal{S} \subseteq \{0,1\}^*$ των εισόδων (στις υπόλοιπες εισόδους δε μας ενδιαφέρει η συμπεριφορά της μηχανής).

Μέχρι στιγμής είδαμε παραδείγματα μηχανών που (με αυξημένες ιδιότητες) προσπαθούσαν όλες μόνες τους να υπολογίσουν κάποιο αποτέλεσμα. Αυτό ασφαλώς μπορούμε να το γενικεύσουμε φτιάχνοντας συστήματα όπου η πληροφορία που επιθυμούμε εξάγεται από τη «συζήτηση» δύο ή περισσότερων μηχανών – οι οποίες ωστόσο δε συνεργάζονται (καθώς τότε θα μπορούσαν να αντικατασταθούν τετριμμένα από μία μηχανή που προσομοιώνει όλες τις υπόλοιπες), αλλά προσομοιώνοντας την προσπάθεια απόδειξης ενός επιχειρήματος, αποτελούν ένα ζεύγος που το ένα (ο αποδείκτης) προσπαθεί να πείσει το άλλο για την ορθότητα του επιχειρήματος και ο άλλος (ο ελεγκτής) με κατάλληλα ερωτήματα να το καταρρίψει (και σε περίπτωση που αποτύχει να το αποδεχθεί ως ορθό). Σε τέτοια συστήματα, θεωρούμε ότι κάθε μηχανή έχει μια ταινία στην οποία μπορεί η ίδια να γράφει και όλες οι άλλες να διαβάζουν. Θεωρούμε επίσης για απλότητα ότι κάθε στιγμή μόνο μια μηχανή εργάζεται. Τέλος κάθε φορά που μια μηχανή γράφει κάτι στη ταινία της και η εκτέλεση περνάει σε άλλη μηχανή (κάτι το οποίο σηματοδοτείται από κάποιες αντίστοιχες καταστάσεις ειδικού σκοπού που προσθέτουμε στο Q), λέμε ότι συνέβη μια συνδιαλλαγή. Τέτοια συστήματα ονομάζονται διαλογικά αποδεικτικά συστήματα και ακολουθεί ένας πιο αυστηρός ορισμός των σχετικών κλάσεων:

Ορισμός 2.2.9. Η κλάση $IP[k]$ περιλαμβάνει τις γλώσσες, για τις οποίες υπάρχει τυχαιοκρατική μηχανή Turing πολυωνυμικού χρόνου $\mathcal{V}$, η οποία με είσοδο το x κι έχοντας το πολύ k συνδιαλλαγές με μία φυσική συνάρτηση $\mathcal{P} : \{0,1\}^* \to \{0,1\}^*$:

- Αν $x \in L$, τότε υπάρχει $\mathcal{P}$ τέτοια ώστε από τη παραπάνω διαλογική απόδειξη ο $\mathcal{V}$ να καταλήγει σε αποδοχή με μεγάλη πιθανότητα [4].
- Αν $x \notin L$, τότε για κάθε $\mathcal{P}$, ο $\mathcal{V}$ καταλήγει σε απόρριψη με επίσης μεγάλη πιθανότητα.

Ορίζουμε $IP = \bigcup_{c>0} IP[n^c]$.

[4](όπου το ακριβές φράγμα συμβαδίζει με όσα είπαμε για τη κλάση BPP)

Να σημειωθεί ότι στον παραπάνω ορισμό ο αποδείκτης $\mathcal{P}$ μπορεί να είναι μια αυθαίρετη φυσική συνάρτηση. Επειδή όμως ο ελεγκτής είναι πολυωνυμικού (τυχαιοκρατικού) χρόνου, όλη η προσομοίωση του μπορεί να ελεγχθεί σε πολυωνυμικό χώρο. Άρα ο αποδείκτης, αρκεί να βρίσκει τις απαντήσεις που δίνουν τη μεγαλύτερη πιθανότητα να οδηγήσει σε αποδοχή τον $\mathcal{V}$, τις οποίες μπορεί να υπολογίσει σε πολυωνυμικό χώρο κι άρα αποδεικνύεται, ότι για το IP, αρκεί ο αποδείκτης $\mathcal{P}$ να είναι μια $PSPACE$ ντετερμινιστική μηχανή Turing [Sha92]. Από αυτό προκύπτει ότι η όλη προσομοίωση του διαλογικού συστήματος μπορεί να γίνει επίσης στο $PSPACE$ κι άρα $IP \subseteq PSPACE$. Συγκεκριμένα αποδεικνύεται ότι η εκφραστικότητα της κλάσης IP είναι τέτοια, ώστε οι δύο κλάσεις να είναι ίσες:

Θεώρημα 2.2 ([Sha92])**.** $IP = PSPACE$

Από αυτό γίνεται φανερό, πόσο μεγάλη υπολογιστική ισχύ προσφέρει η αβεβαιότητα που δίνουν τα τυχαία κέρματα και η άγνοια του αποτελέσματος τους από τον $\mathcal{P}$ (καθώς αλλιώς ο $\mathcal{P}$ (όντας παντοδύναμος) θα μπορούσε απλά να δίνει απαντήσεις για τις οποίες ξέρει ότι ο $\mathcal{V}$ αποδέχεται). Εν τούτοις αποδεικνύεται ότι ακόμη και με φανερό το αποτέλεσμα των κερμάτων και πάλι προκύπτουν δύο πολύ ισχυρές χαρακτηριστικές κλάσεις:

Ορισμός 2.2.10. Ορίζουμε ως $AM[k]$ το υποσύνολο του $IP[k]$ για το οποίο τα μηνύματα του ελεγκτή $\mathcal{V}$ είναι τυχαία δυφία και μάλιστα τα μοναδικά στα οποία έχει πρόσβαση ο $\mathcal{V}$.
Ορίζουμε $AM = AM[2]$[5]

Αυτές οι διαλογικές αποδείξεις, λέγονται και διαλογικές αποδείξεις δημόσιων τυχαίων δυφίων δύο γύρων. Είναι εύκολο να δούμε από τον ορισμό ότι με χρήση τελεστών η κλάση AM χαρακτηρίζεται από το ζεύγος $(\exists^+\exists, \exists^+\forall)$ ή με χρήση «τελεστών κλάσεων» [ZP03] με την $\mathcal{BP} \cdot \mathcal{NP}$ (κι αποδεικνύεται αντίστοιχα ότι $AM \subseteq \Pi_2^P$ [Lau83]). Τέλος, ορίζουμε ως MA τις διαλογικές αποδείξεις δημόσιων τυχαίων δυφίων δύο γύρων για τις οποίες πρώτα στέλνει ένα μήνυμα ο αποδείκτης και μετά ο ελεγκτής με χρήση τυχαίων δυφίων αποφασίζει την αποδοχή ή όχι. Αντίστοιχα το MA χαρακτηρίζεται από το ζεύγος $(\exists\exists+, \forall\exists+)$ ή αλλιώς με τη κλάση $\mathcal{N} \cdot \mathcal{BPP}$.

Είδαμε στις διαλογικές αποδείξεις ότι ο κυρίως υπολογισμός γινόταν από τον ελεγκτή και ότι ο αποδείκτης είχε απλά το ρόλο «μαντείου» απαντώντας (σε ουσιαστικά στιγμιαίο χρόνο) τις τυχαίες ερωτήσεις του ελεγκτή. Γενικεύοντας αυτή την έννοια (όπου τώρα το μαντείο παίζει το ρόλο του βοηθού), μπορούμε να ορίσουμε μηχανές με μαντείο, οι οποίες έχουν μια ειδική ταινία εγγραφής ερωτημάτων προς το μαντείο και όταν μεταβαίνουν σε μια ειδική νέα

[5] Αποδεικνύεται ότι k συνδιαλλαγές μπορούν να συμπτυχθούν σε μόλις 2, όπου στέλνει πρώτα ένα μήνυμα ο ελεγκτής και παίρνει μετά την απάντηση από τον αποδείκτη κι εν τέλει αποφασίζει.

κατάσταση q_{query}, το μαντείο επεξεργάζεται στιγμιαία το ερώτημα κι εξάγει σε ένα βήμα το αποτέλεσμα σε μια άλλη ειδική ταινία (στην οποία έχει μόνο πρόσβαση ανάγνωσης η αρχική μηχανή).

Ορισμός 2.2.11. Ορίζουμε ως $DTIME^{\mathcal{O}}[f(n)]$ τη κλάση των γλωσσών που υπολογίζονται σε χρόνο $O(f(n))$ από μια ντετερμινιστική μηχανή που χρησιμοποιεί ως μαντείο τη συνάρτηση $\mathcal{O} : \{0,1\}^* \to \{0,1\}^*$.

Ασφαλώς αλλάζοντας το μοντέλο της ντετερμινιστικής μηχανής με οποιοδήποτε από τα παραπάνω παίρνουμε τις αντίστοιχες κλάσεις $NTIME^{\mathcal{O}}[f(n)]$, $BPTIME^{\mathcal{O}}[f(n)]$ κ.ο.κ. Με το φυσικό τρόπο, ορίζονται επίσης οι κλάσεις $P^{\mathcal{O}}$, $(\Sigma_i P)^{\mathcal{O}}$ κ.ο.κ. Να σημειωθεί ότι η συνάρτηση που υλοποιεί το μαντείο μπορεί να είναι οποιαδήποτε αυθαίρετη συνάρτηση, άρα τώρα είναι το πρώτο εναλλακτικό μοντέλο που βλέπουμε, το οποίο σε πολλές περιπτώσεις δεν είναι καν υπολογίσιμο (π.χ. έχοντας ως μαντείο το Πρόβλημα Τερματισμού). Τέλος μπορούμε να ορίσουμε κλάσεις που αντί για ένα μόνο μαντείο, μπορούν να επιλέξουν από μία κλάση μαντείων.

Ορισμός 2.2.12. Ως $DTIME^{\mathcal{C}}[f(n)]$,για μια κλάση συναρτήσεων $\mathcal{C}$, ορίζουμε την κλάση που προκύπτει από την ένωση όλων των $DTIME^{\mathcal{L}}[f(n)]$ για κάθε $\mathcal{L} \in \mathcal{C}$.

Μας ενδιαφέρουν, όπως είναι αναμενόμενο, περισσότερο οι ιδιότητες κλάσεων που έχουν μαντεία υπολογίσιμες κλάσεις, όπως π.χ. οι P^{NP}, BPP^{NP}, EXP^{NP} κ.ο.κ. Μάλιστα, αποδεικνύεται (με επαγωγή) ότι $\Sigma_i^P = NP^{\Sigma_{i-1}^P}$ [AB09, p.102–103]. Συγκεκριμένα, αν ορίσουμε ως $\Delta_{i+1}^P = P^{\Sigma_i^P}$, τότε είναι σχετικά εύκολο να δείξουμε ότι $\Sigma_i^P \subseteq \Delta_{i+1}^P \subseteq V_{i+1}^P$ (όπου V_i^P οποιοδήποτε από τα Σ_i^P, Π_i^P). Αντίστοιχα αποτελέσματα έχουμε για τα Δ_i^{EXP}.

2.2.2 Πλήρη Προβλήματα

Στον ορισμό των μη ντετερμινιστικών κλάσεων, επιτρέψαμε ο ελεγκτής να είναι γραμμικού χρόνου ώστε να κάνει τόσο χρόνο όσο το μήκος του πιστοποιητικού (εξασφαλίζοντας έτσι συνολικό μη ντετερμινιστικό χρόνο όσο το μήκος αυτό). Ένας επιπλέον λόγος που το κάναμε αυτό, ήταν επειδή για κατάλληλη μορφή του αποδεικτικού, ο έλεγχος του μπορεί να γίνει σε γραμμικό χρόνο και ο τρόπος που γίνεται αυτό είναι εισάγοντας την ίδια την εκτέλεση του ελεγκτή μέσα στο πιστοποιητικό! Όταν το κάνουμε αυτό, αυτός ο ελεγκτής έχει μια γενική μορφή, η οποία είναι κατάλληλη για κάθε NP πρόβλημα (αφού απλώς ελέγχει για το αν πρόκειται για ορθή εκτέλεση μιας μη ντετερμινιστικής μηχανής – ισοδύναμα αν πρόκειται για ορθή απόδειξη βάσει κάποιων αξιωμάτων) και έτσι μπορούμε να πούμε ότι κάθε NP πρόβλημα ανάγεται σε αυτό. Συγκεκριμένα έχουμε τον ακόλουθο ορισμό:

Ορισμός 2.2.13. Μία γλώσσα L είναι NP-πλήρης αν και μόνο αν είναι στο NP και για κάθε γλώσσα $L' \in NP$ υπάρχει αναγωγή $R \in FP$ τέτοια ώστε $x \in L' \Leftrightarrow R(x) \in L$.

Ο παραπάνω ορισμός μπορεί να επεκταθεί και για υπόλοιπες κλάσεις κι έτσι έχουμε π.χ. τον ορισμό των Σ_i^P-πλήρων, των $PSPACE$-πλήρων, των $NEXP$-πλήρων προβλημάτων κ.ο.κ. Στην περίπτωση των P-πλήρων προβλημάτων (τα οποία με τον παραπάνω ορισμό τετριμμένα υπάρχουν) απαιτούμε η R να είναι στο FL, δηλαδή να υπολογίζεται σε λογαριθμικό χώρο. Αποδεικνύεται ότι το SAT που παίρνει ως είσοδο μία λογική φόρμουλα μήκους n (στη βάση $\{AND, OR, NOT\}$) και αποδέχεται μόνο αν υπάρχει απονομή των μεταβλητών της που να την ικανοποιεί είναι NP-πλήρες (για ακριβείς ορισμούς των κανόνων της απονομής αληθείας σε μια λογική πρόταση υπάρχει εκτενής βιβλιογραφία [Goo12, Par14]). Αποδεικνύεται λοιπόν το ακόλουθο:

Θεώρημα 2.3 ([Coo71])**.** *Το SAT είναι NP-πλήρες.*

Απόδειξη. (Σχέδιο) Έστω L ένα NP πρόβλημα και M η ντετερμινιστική μηχανή του ορισμού που ελέγχει το $\langle x, y\rangle$ (δηλαδή $x \in L$ αν και μόνο αν $\exists y M(x, y) = 1$, όπου $|y| < poly(|x|)$. Επειδή η M είναι πολυωνυμικού χρόνου, η διαμόρφωση της κάθε στιγμή είναι επίσης πολυωνυμικού χώρου και μπορεί να περιγραφεί από πολυωνυμικό πλήθος μεταβλητών. Το μόνο που μένει είναι αρχίζοντας από διαμόρφωση που αντιστοιχεί στην αρχική διαμόρφωση c_0 να δείξουμε ότι κάθε επόμενη διαμόρφωση είναι συμβατή με την αμέσως προηγούμενη και ότι η τελική ισοδυναμεί με αποδοχή. Κωδικοποιούμε, λοιπόν, για κάθε στιγμή, κάθε στοιχείο της ταινίας, το τρέχον κελί και την τρέχουσα κατάσταση με πολυωνυμικό πλήθος μεταβλητών, ξεχωριστών για κάθε διαμόρφωση κάθε βήματος. Θέτουμε με κατάλληλους όρους, για τις μεταβλητές της διαμόρφωσης εισόδου, οι μεταβλητές που αντιστοιχούν στο x να ισοδυναμούν με τα αντίστοιχα δυφία, η τρέχουσα κατάσταση να είναι η q_{start}, η κεφαλή να δείχνει στο πρώτο κελί και τα δυφία που αντιστοιχούν στο y να απομένουν ως ελεύθερες μεταβλητές (την απονομή των οποίων ψάχνουμε να βρούμε). Για κάθε επόμενη διαμόρφωση εισάγουμε φρέσκες μεταβλητές για κάθε στοιχείο της και απλώς θέτουμε όρους που να επιβάλλουν η i-οστή τιμή τους να συμβαδίζει με το αντίστοιχο $(i-1)$-οστό στοιχείο (βάσει της συνάρτησης μετάβασης και του αποτελέσματος $\{\diamond, \triangleleft, \triangleright\}$). Αυτό μπορεί να γίνει αποδοτικά και ομοιόμορφα ακριβώς χάρη στη τοπικότητα του υπολογισμού μιας μηχανής Turing για την οποία ισχύουν οι ίδιοι κανόνες τόσο χρονικά (κάθε διαμόρφωση πηγαίνει στην επόμενη βάσει των ίδιων κανόνων) όσο και χωρικά (κάθε μετάβαση αφορά μόνο το τρέχον κελί και τα υπόλοιπα παραμένουν αμετάβλητα). Να σημειωθεί εδώ, ότι επειδή κάθε επόμενη διαμόρφωση καθορίζεται μονοσήμαντα από την προηγούμενη, αν δώσουμε σταθερή απονομή στις μεταβλητές των δυφίων του y, τότε όλες οι υπόλοιπες μεταβλητές έχουν μοναδική απονομή για την οποία ικανοποιούνται. Αν τώρα θέσουμε έναν τελικό όρο στην υπό κατασκευή φόρμουλα που να δηλώνει ότι η τελευταία διαμόρφωση καταλήγει στην q_{accept}, τότε υπάρχει απονομή σε όλες τις μεταβλητές που να ικανοποιεί όλο τον τύπο, αν και μόνο αν υπάρχει απονομή στις μεταβλητές των δυφίων του y (ισοδύναμα αν υπάρχει y) που να καταλήγει τη M σε αποδοχή κι άρα η ισοδυναμία ύπαρξης πιστοποιητικού με την ικανοποιησιμότητα της φόρμουλας που κατασκευάσαμε αποδείχθηκε. Είναι

εύκολο να δούμε ότι τόσο το μήκος όσο και η κατασκευή της φόρμουλας μπορεί να γίνει έτσι ώστε να μη ξεπερνάει τον κύβο του χρόνου που τρέχει η M κι άρα η αναγωγή είναι πράγματι στο FP και η απόδειξη είναι πλήρης. ■

Σημείωση. Είναι σχετικά εύκολο να δειχθεί ότι οι παραπάνω όροι μπορούν όλοι να μεταφραστούν στη μορφή $\bigwedge_{j=1}^{m}(l_{1j} \vee l_{2j} \vee l_{3j})$, όπου l_{kj} οποιοδήποτε από τα x_i ή $\neg x_i$. Το αντίστοιχο πρόβλημα που δέχεται μόνο φόρμουλες σε αυτή τη μορφή, λέγεται $3-SAT$ κι είναι επίσης NP-πλήρες [Coo71].

Σημείωση. Η παραπάνω αναγωγή δημιουργεί εν τέλει ένα ισοδύναμο $3-SAT$ στιγμιότυπο μήκους $poly(T(n))$, όπου $T(n)$ ο χρόνος εκτέλεσης της αντίστοιχης μηχανής. Αποδεικνύουμε στο Παράρτημα Α′.1 ότι με μια πιο έξυπνη κωδικοποίηση μπορούμε να φτιάξουμε ένα ισοδύναμο $3-SAT$ στιγμιότυπο μήκους μόλις $T(n)logT(n)$ [Tse68].

Με παρόμοιο (και ανά περιπτώσεις λιγότερο τετριμμένο) τρόπο [Uma01, CCJK06, Pap94, p.159–180, 409–438] αποδεικνύονται και τα ακόλουθα πλήρη προβλήματα για τις ακόλουθες κλάσεις:

- P: $B(oolean)F(ormula)E(valuation)$ (Δίνεται ως είσοδος μια λογική φόρμουλα και μια απονομή και αποδέχεται αν η απονομή την ικανοποιεί.)
- Σ_i^P: $i-\exists Q(uantified)B(oolean)F(ormula)$ (Δίνεται ως είσοδος μία λογική φόρμουλα με το πολύ i εναλλασσόμενους τελεστές, εκ των οποίων ο πρώτος να είναι $\exists$ και αποδέχεται αν είναι ικανοποιήσιμη.)
- Π_i^P: $i-\forall QBF$ (Δίνεται ως είσοδος μία λογική φόρμουλα με το πολύ i εναλλασσόμενους τελεστές, εκ των οποίων ο πρώτος να είναι $\forall$ και αποδέχεται αν είναι ικανοποιήσιμη.)
- $PSPACE$: $TQBF$ (Δίνεται ως είσοδος μία λογική φόρμουλα με τελεστές και αποδέχεται αν είναι ικανοποιήσιμη.)
- $DTIME[f(n)]$: $BHP[f(n)]$ (Δίνεται ως είσοδος η περιγραφή μιας μηχανής και μια είσοδος και αποδέχεται αν τερματίζει για αυτή την είσοδο σε χρόνο το πολύ $f(n)$.)
- $DTIME^{\mathcal{O}}[f(n)]$: $BHP^{\mathcal{O}}[f(n)]$ (Όπως προηγουμένως, όμως η μηχανή που δίνεται στην είσοδο έχει πρόσβαση σε ένα μαντείο $\mathcal{O}$.)
- $NEXP$: $succinct-SAT$ (Δίνεται ως είσοδος μία λογική φόρμουλα, η οποία είναι συμπυκνώσιμη (υπό την έννοια ότι περιγράφεται από ένα μικρό κύκλωμα) και ζητείται η ικανοποιησιμότητα της – περισσότερα στο Θεώρημα 5.1.)

Γενικά σχεδόν όλες οι κλάσεις που είδαμε έχουν πλήρη προβλήματα [6]. Η μεγάλη σημασία των πλήρων προβλημάτων είναι ότι αποτελούν καλούς αντιπροσώπους των κλάσεων τους. Πράγματι, είναι εύκολο για παράδειγμα να δούμε

[6] Με εξαίρεση την PH, για την οποία εύκολα βλέπουμε ότι αν έχει πλήρες πρόβλημα, τότε καταρρέει σε κάποιο επίπεδο, οπότε πιστεύεται ότι δεν υπάρχει [Pap94, p.428–429].

πως για να δείξουμε ότι $P = NP$, αρκεί να βρούμε πολυωνυμικό αλγόριθμο μόνο για το $3-SAT$ και το ίδιο ισχύει για αντίστοιχες ιδιότητες (οι οποίες να είναι κλειστές ως προς τις πολυωνυμικές αναγωγές) που θέλουμε να δείξουμε για το NP, Σ_i^P, $NEXP$ κ.ο.κ.

2.3 Εφαρμογές Διαγωνιοποίησης

2.3.1 Θεωρήματα Ιεραρχίας

Η διαγωνιοποίηση αποδείχτηκε πολύ χρήσιμο εργαλείο για την εξόρυξη αρνητικών αποτελεσμάτων και συγκεκριμένα για την απόδειξη ότι συγκεκριμένες γλώσσες δεν είναι υπολογίσιμες. Ακολουθώντας την ίδια ακριβώς μέθοδο, μπορούμε να πάρουμε αντίστοιχα αποτελέσματα που να αποδεικνύουν την γνησιότητα των κλάσεων $DTIME[f(n)]$ με την έννοια ότι καμία από αυτές δεν καταρρέει σε κάποια άλλη λιγότερου χρόνου (πάντα μιλώντας για κατασκευάσιμες f).

Θεώρημα 2.4 ([HS65, HS66]). *Αν f, g κατασκευάσιμες $(f(n), g(n) \geq n)$ με $f(n)log(f(n)) = o(g(n))$[7], τότε*

$$DTIME[f(n)] \subsetneq DTIME[g(n)]$$

.

Απόδειξη. Προφανώς αρκεί να δείξουμε ότι υπάρχει $L \in DTIME[g(n)]$ με $L \notin DTIME[f(n)]$. Έστω το αντίθετο κι έστω κατασκευάσιμη h, τέτοια ώστε $f = o(h)$ και $h(n)log(h(n)) = o(g(n))$ (αποδεικνύεται ότι υπάρχει). Τότε έστω η γλώσσα L η οποία δέχεται ως είσοδο τη περιγραφή "M" μίας μηχανής Turing M και αποδέχεται αν και μόνο αν η M("M") απορρίπτει σε χρόνο $h(n)$. Η προσομοίωση μπορεί να γίνει σε χρόνο $h(n)log(h(n))$ κι άρα $L \in DTIME[g(n)]$ κι άρα υπάρχει μια αρκετά μεγάλη περιγραφή μιας μηχανής M', τέτοια ώστε η M' να υπολογίζει την L σε χρόνο $O(f(n))$ και η "M'" να είναι αρκετά μεγάλη (με έξτρα περιττές καταστάσεις) ώστε να έχει τερματίσει σίγουρα σε χρόνο $h(|"M'"|)$. Έχουμε λοιπόν M("M'") αποδέχεται αν και μόνο αν M'("M'") απορρίπτει σε χρόνο $h(n)$, δηλαδή (αφού έχει σίγουρα τερματίσει πιο πριν) αν και μόνο αν M'("M'") απορρίπτει. Όμως η M' υπολογίζει την ίδια συνάρτηση με τη M και καταλήγουμε στο προφανές άτοπο. ■

Το συμπέρασμα από το παραπάνω Θεώρημα, είναι ότι ο χρόνος είναι σημαντική ποσότητα για τις μηχανές Turing υπό την έννοια ότι όσο περισσότερος, τόσο και πιο πολλές συναρτήσεις μπορούν να υπολογιστούν. Στο ίδιο συμπέρασμα (με την ίδια τεχνική) μπορούμε να καταλήξουμε και για τον χώρο:

[7] Προκύπτει ότι εν τέλει αρκεί μόνο $f(n) = o(g(n))$ με μια μικρή παραλλαγή [Für82].

Θεώρημα 2.5 ([Sip78]). *Αν f, g κατασκευάσιμες ($f(n), g(n) \geq logn$) με $f(n) = o(g(n))$, τότε*

$$DSPACE[f(n)] \subsetneq DSPACE[g(n)]$$

.

Κατασκευάζοντας πιο σύνθετες μηχανές που κάνουν πιο εξεζητημένους υπολογισμούς γύρω από την κεντρική προσομοίωση (βασιζόμενους στις υποθέσεις που θέλουμε να καταρρίψουμε), μπορούμε να καταλήξουμε και σε θεωρήματα ιεραρχίας και για κάποια από τα εναλλακτικά μοντέλα υπολογισμού που μελετήσαμε στη προηγούμενη ενότητα:

Θεώρημα 2.6 ([Coo72, SFM78]). *Αν f, g κατασκευάσιμες με $f(n+1) = o(g(n))$, τότε*

$$NTIME[f(n)] \subsetneq NTIME[g(n)]$$

.

Απόδειξη. Το πρόβλημα με τις μη ντετερμινιστικές μηχανές είναι ότι δεν έχουμε τρόπο να προσομοιώνουμε σε αυτές τη μη αποδοχή σε περίπτωση που μία άλλη (αυτή που δίνεται ως είσοδος) ερχόταν σε αποδοχή (όπως κάνουμε στη παραπάνω απόδειξη) καθώς μάλιστα μια τέτοια μέθοδος θα σήμαινε ότι $coNTIME[g(n)] = NTIME[g(n)]$ κάτι το οποίο εικάζεται ψευδές (και που σε κάθε περίπτωση δε γνωρίζουμε κάποια μέθοδο που να το επαληθεύει ώστε να μπορούμε να τη χρησιμοποιήσουμε). Εν τούτοις μπορούμε να κάνουμε αυτή την αντιστροφή ντετερμινιστικά (κι άρα και μη ντετερμινιστικά) σε εκθετικό χρόνο. Αυτό σε συνδυασμό με εξάπλωση μιας τιμής σε εκθετικό εύρος (με τον τρόπο που παρουσιάζεται στη συνέχεια) είναι ικανό να καταλήξει στο ίδιο συμπέρασμα.
Έστω λοιπόν μία συνάρτηση h, τέτοια ώστε $f(n) = o(h(n))$, $h(n+1) = o(g(n))$. Έστω επίσης e η συνάρτηση για την οποία $e(1) = k$ και $e(i+1) = 2^{h(e(i))}$. Με δυαδική αναζήτηση, μπορούμε σε το πολύ $\frac{g(n)}{2}$ χρόνο να βρούμε το i για το οποίο $e(i) < n \leq e(i+1)$. Κατασκευάζουμε λοιπόν τη μη ντετερμινιστική μηχανή N που εκτελεί τα ακόλουθα βήματα:

1. Για οποιαδήποτε είσοδο x, έστω n το μήκος της. Βρες το προαναφερθέν i για αυτό το n σε χρόνο $\frac{g(n)}{2}$.
2. (α′) Αν $e(i) < n < e(i+1)$, τότε προσομοίωσε μη ντετερμινιστικά τη μηχανή M_i με είσοδο 1^{n+1} (δηλαδή της οποίας η περιγραφή ταιριάζει στη δυαδική αναπαράσταση του i βάσει μιας κωδικοποίησης) για χρόνο $h(n+1)$ και αποδέξου αν και μόνο αν έχει τερματίσει και αποδεκτεί μέχρι τότε.
   (β′) Αν $n = e(i+1)$, τότε έλεγξε ντετερμινιστικά αν η M_i αποδέχεται το $1^{e(i)+1}$ σε $h(e(i)+1)$ μη ντετερμινιστικό χρόνο (κι άρα υπάρχουν το πολύ $2^{h(e(i)+1)}$ μονοπάτια προς έλεγχο) οπότε εξ ορισμού της e σίγουρα σε $h(n)$ χρόνο και αν ναι απόρριψε, αλλιώς αποδέξου.

Προκύπτει λοιπόν ότι η μη ντετερμινιστική μηχανή N θέλει χρόνο ίσο με το άθροισμα των δύο βημάτων (δηλαδή $\frac{g(n)}{2}$ και $O(h(n+1))$ αντίστοιχα) κι άρα η αντίστοιχη γλώσσα είναι σίγουρα στο $DTIME[g(n)]$. Επομένως, αν δεν ισχύει το προς απόδειξη, υπάρχει μια αρκετά μεγάλη κωδικοποίηση I τέτοια ώστε η M_I να υπολογίζει την ίδια γλώσσα με την N και να τερματίζει πάντα σε $f(n)$ μη ντετερμινιστικό χρόνο για εισόδους μεγέθους $e(I)$ και πάνω. Έχουμε τότε ότι για κάθε είσοδο x_n μήκους n με $e(I) < n < e(I+1)$ ισχύει $N(x_n) = N(1^n) = M_I(1^{n+1})$ και για $n = e(I+1)$ ότι $N(x_n) = N(1^{e(I+1)}) \neq M_I(1^{e(I)+1})$. Επειδή, όμως, όπως είπαμε η M_I ταυτίζεται σημασιολογικά με τη N προκύπτει από τη πρώτη σχέση ότι η N είναι σταθερή για κάθε είσοδο μήκους $e(I)+1 \leq n \leq e(I+1)$, ενώ από τη δεύτερη ότι $N(1^{e(I)+1}) \neq N(1^{e(I+1)})$ καταλήγοντας σε άτοπο. ■

Δυστυχώς, δεν έχουμε μέχρι σήμερα κάποιο αντίστοιχο ισχυρό Θεώρημα Ιεραρχίας για τις κλάσεις $BPTIME[f(n)]$, κυρίως επειδή δε μπορεί να επιλυθεί τετριμμένα το γεγονός ότι διαβάζοντας μια μηχανή αυτή έχεις τις ιδιότητες μιας τυχαιοκρατικής μηχανής (δηλαδή ότι οι περισσότερες είσοδοι συμφωνούν στο ίδιο αποτέλεσμα)[8].

Η παραπάνω τεχνική, μπορεί με τις κατάλληλες μεταβολές των παραμέτρων να οδηγήσει (με όλο και πιο περίπλοκες αποδείξεις) σε πάρα πολλά ακόμη χρήσιμα συμπεράσματα διαχωρισμού, όπως π.χ. ότι αν $P \neq NP$, τότε υπάρχουν άπειρες ενδιάμεσες κλάσεις NP-ενδιάμεσων προβλημάτων (δηλαδή που δεν είναι ούτε στο P, ούτε NP-πλήρη) [Lad75]. Παρά τις μεγάλες δυνατότητες της, εν τούτοις οδηγεί και σε ένα συμπέρασμα που δυστυχώς την εξαιρεί από τις υποψήφιες να αποδείξουν ότι $P \neq NP$, όπως θα δούμε στην ακόλουθη υποενότητα.

2.3.2 Σχετικιστικές Αποδείξεις

Όλες οι παραπάνω αποδείξεις διαγωνιοποίησης, έχουν ένα κοινό χαρακτηριστικό: Ανεξαρτήτως του μοντέλου που χρησιμοποιείται, υπονοείται πάντα μια μηχανή Turing η οποία εκτελεί τη προσομοίωση μιας άλλης μηχανής (η οποία δίνεται ως είσοδος βάσει κάποιας αποτελεσματικής κωδικοποίησης) και σε χρόνο προσομοίωσης συγκρίσιμο με τον αρχικό. Κοινό χαρακτηριστικό, λοιπόν, αυτών των αποδείξεων είναι ότι κατά βάση λαμβάνουν ως μόνη πληροφορία από αυτές τις μηχανές, την έξοδο τους για συγκεκριμένες εισόδους (τις αντιμετωπίζουν ως μαύρα κουτιά δηλαδή).

Είναι, όμως, αρκετά εύκολο να παρατηρήσουμε ότι στις γενικές αυτές γραμμές, όλες αυτές οι αποδείξεις επεκτείνονται ακόμη και όταν δώσουμε στις μηχανές Turing πρόσβαση σε κάποιο μαντείο. Πράγματι, η προσομοίωση μπορεί να εκτελεστεί κανονικά με χρήση του μαντείου που διαθέτει ο αλγόριθμος προσομοίωσης και σε χρόνο ανάλογο του αρχικού. Είναι εύκολο δηλαδή να

[8] Εν τούτοις έχουν υπάρξει (με πιο σύνθετες αποδείξεις) αποτελέσματα που ξεχωρίζουν το $BPEXP$ από το $QuasiBPP$ [KV87].

δούμε, ότι από την απόδειξη των παραπάνω προκύπτει ότι $TIME^{\mathcal{O}}[f(n)] \subsetneq TIME^{\mathcal{O}}[g(n)]$ (για ντετερμινιστικό ή μη χρόνο) όπου $f(n) = o(g(n))$, οδηγώντας σε αντίστοιχα συμπεράσματα χρονικής (και χωρικής) ιεραρχίας για τα i-οστά επίπεδα $\Sigma_i TIME[f(n)]$, $\Pi_i TIME[f(n)]$, $\Delta_i TIME[f(n)]$. Εν τούτοις η τεχνική του ακόλουθου Θεωρήματος μας εγγυάται, ότι μία απόδειξη σε αυτά τα πλαίσια δεν είναι δυνατόν να ξεχωρίσει κλάσεις που ανήκουν σε διαφορετικά επίπεδα και συγκεκριμένα εν προκειμένω ότι δε μπορεί να ξεχωρίσει το P από το NP.

Θεώρημα 2.7 ([BGS75]). *Υπάρχουν μαντεία $\mathcal{A}$, $\mathcal{B}$, τέτοια ώστε $P^{\mathcal{A}} = NP^{\mathcal{A}}$ και $P^{\mathcal{B}} \neq NP^{\mathcal{B}}$.*

Απόδειξη. Το $\mathcal{A}$ είναι απλώς το μαντείο που με είσοδο $\langle "M", x, 1^t \rangle$ επιστρέφει 1 αν και μόνο αν η M αποδέχεται το x σε 2^t χρόνο (εύκολα βλέπουμε ότι το $\mathcal{A}$ είναι EXP-πλήρες). Τότε, επειδή σε μία $NP^{\mathcal{A}}$ μηχανή όλα τα ερωτήματα στο $\mathcal{A}$ κάθε μονοπατιού είναι πολυωνυμικού μεγέθους και επειδή υπάρχει το πολύ εκθετικό πλήθος μονοπατιών, είναι εύκολο να κατασκευάσουμε (σε πολυωνυμικό χρόνο) τη περιγραφή μιας μηχανής N που θα προσομοιώνει όλα τα μονοπάτια του $NP^{\mathcal{A}}$ (σε εκθετικό χρόνο) και για κάθε ερώτημα (εισόδου πολυωνυμικού μεγέθους) να το απαντάει σε επίσης εκθετικό χρόνο. Όλη η μηχανή αυτή τρέχει επίσης σε εκθετικό χρόνο (ως γινόμενο εκθετικών κομματιών) κι άρα αρκεί ένα ντετερμινιστικό ερώτημα στο μαντείο για τη N με είσοδο x και χρόνο $1^{t'}$ αρκετά μεγάλο (αλλά πολυωνυμικό) ώστε η N να έχει τερματίσει σίγουρα σε χρόνο $2^{t'}$ (κι άρα πράγματι $NP^{\mathcal{A}} \subseteq P^{\mathcal{A}}$).

Το $\mathcal{B}$ είναι, ωστόσο, πιο περίπλοκο. Η διαίσθηση είναι ότι μπορούμε να κάνουμε το $\mathcal{B}$ αρκετά σύνθετο ώστε καμία ντετερμινιστική μηχανή υποεκθετικού χρόνου να μη μπορεί να εντοπίσει κάποιον άσσο της (ακριβώς όπως σε ένα μαύρο κουτί δε μπορούμε να ξέρουμε αν υπάρχει άσσος χωρίς να δοκιμάσουμε κάθε πιθανή είσοδο). Πρέπει, δηλαδή, να προσδιορίσουμε ακριβώς το $\mathcal{B}$, έτσι ώστε η γλώσσα $\{1^n : \exists y(|y| = n \wedge \mathcal{B}(y) = 1)\}$ να μην ανήκει στο $P^{\mathcal{B}}$. Το πρόβλημα αυτό, όμως, για οποιοδήποτε μαντείο (υπολογίσιμο ή μη) είναι επιλύσιμο στο $NP^{\mathcal{B}}$ καθώς απλώς μαντεύει την είσοδο y για την οποία $\mathcal{B}(y) = 1$ και επαληθεύει δοκιμάζοντας στο $\mathcal{B}$. Προχωράμε στην ακριβή περιγραφή του $\mathcal{B}$:
Για κάθε μηχανή M_i ορίζουμε το $\mathcal{B}$ έτσι ώστε το $M_i^{\mathcal{B}}$ να κάνει λάθος για τουλάχιστον ένα μήκος εισόδου n_i. Έστω ότι ισχύει μέχρι κάποιο $i-1$. Θα το δείξουμε για το M_i. Έστω το πρώτο n_i για το οποίο κάθε M_j με $j < i$ δεν έχει κάνει κάποιο ερώτημα στο $\mathcal{B}$ τέτοιου μήκους στα προηγούμενα $i-1$ στάδια και για το οποίο η M_i έχει τερματίσει σε $\frac{2^{n_i}}{2}$ βήματα. Για ό,τι ερώτημα κάνει η M_i στο $\mathcal{B}$ με είσοδο 1^{n_i} (σε το πολύ $\frac{2^{n_i}}{2}$ βήματα) και για το οποίο δεν έχει οριστεί τιμή από προηγούμενα στάδια, ορίζουμε το $\mathcal{B}$ να απαντάει 0 (και για τα άλλα ερωτήματα προφανώς ό,τι έχει ήδη οριστεί). Αν εν τέλει αποδέχεται, θέτουμε όλες τις υπόλοιπες τιμές του $\mathcal{B}$ για μήκος n_i ίσες με 0, ενώ αν απορρίπτει θέτουμε κάποια από τις τουλάχιστον $\frac{2^{n_i}}{2}$ τιμές μήκους n_i που δεν έχει ελέγξει ίση με 1. Τέλος όσες τιμές δεν έχουν οριστεί από τη παραπάνω

διαδικασία τις θέτουμε ίσες με 0. Αυτό σημαίνει ότι οποιαδήποτε πολυωνυμική μηχανή M_i κάνει λάθος για την είσοδο n_i, ολοκληρώνοντας την απόδειξη ότι $NP^{\mathcal{B}} \not\subseteq P^{\mathcal{B}}$. ■

Έτσι είναι απίθανο να καταλήξουμε ότι $P \neq NP$ ή ότι $P = NP$ από μια απόδειξη που «σχετικοποιείται». Συγκεκριμένα το να αντιμετωπίζονται οι μηχανές σαν μαύρα κουτιά δε μπορεί καν να μας απαντήσει για τη σχέση μεταξύ $subEXP$ και του NP (καθώς η προηγούμενη απόδειξη συνεχίζει να λειτουργεί εξίσου ακόμη κι αν τοποθετήσουμε το $subEXP$ στη θέση του P). Ακόμη περισσότερο, υπάρχει ένα αντίστοιχο αποτέλεσμα [Sel12, BBS86, For94, HCC$^+$92] για σχεδόν κάθε ζεύγος κλάσεων που ανήκουν σε διαφορετικά μοντέλα υπολογισμού (ντετερμινιστικά, μη ντετερμινιστικά, χωρικά περιορισμένα, διαλογικά συστήματα) και για τα οποία δεν υπάρχει κάποιος τετριμμένος αποκλεισμός (π.χ. προφανώς $BPTIME[n] \subsetneq NEXP$, εφόσον $BPTIME[n] \subseteq E$, $E \subsetneq EXP$ και $EXP \subseteq NEXP$, αλλά υπάρχουν δύο μαντεία που δείχνουν αντίστοιχα ότι δε μπορεί να προκύψει κάποιο σχετικιστικό συμπέρασμα για τη σχέση π.χ. μεταξύ Σ_5^P και $quasiBPP$).

Παρότι η διαγωνιοποίηση είναι ένα πολύ ισχυρό εργαλείο, έχει γίνει πλέον φανερό ότι δεν είναι ικανό από μόνο του να απαντήσει στα μεγάλα ερωτήματα της Θεωρίας Πολυπλοκότητας. Ο λόγος που δε μπορεί να το κάνει αυτό, είναι επειδή κοιτάει τις μηχανές ως μαύρα κουτιά, για τα οποία λαμβάνει υπόψιν μόνο τις σημασιολογικές τους ιδιότητες, χωρίς να εντρυφήσει καθόλου στις επί μέρους πράξεις μιας υπολογιστικής μηχανής και στους περιορισμούς που επιβάλλει η τοπικότητα και η πεπερασμένη περιγραφή των κανόνων της. Στα χρόνια που ακολούθησαν λοιπόν, υπήρξε ένα ριζικό κύμα μεταστροφής των προσπαθειών κατά των μεγάλων ανοιχτών ερωτημάτων της Θεωρητικής Πληροφορικής, από τις σημασιολογικές κι αφηρημένες αποδείξεις στα συντακτικά και τεχνικά επιχειρήματα που εστίαζαν σε επί μέρους συνδυαστικές ιδιότητες των υπολογιστικών βημάτων. Το πιο πρόσφορο κι ελπιδοφόρο πεδίο για τη μελέτη αυτών φάνηκε να είναι το πεδίο των Κυκλωμάτων, του οποίου τη μεγάλη πορεία και τους βασικότερους σταθμούς θα μελετήσουμε στα κεφάλαια που ακολουθούν.

Κεφάλαιο 3

Κυκλωματική Πολυπλοκότητα

Στο προηγούμενο κεφάλαιο εισάγαμε τις βασικές αρχές του πλέον καθιερωμένου μοντέλου υπολογισμού, των μηχανών Turing . Θα μελετήσουμε τώρα ένα διαφορετικό μοντέλο υπολογισμού, το οποίο όπως θα δούμε δεν αντιστοιχεί σε ρεαλιστικό υπολογισμό (καθώς αλλιώς θα είχαμε κατά κάποιον τρόπο κατάρριψη της θέσης Church-Turing). Εν τούτοις ορίζονται αντίστοιχες κυκλωματικές κλάσεις, οι οποίες μάλιστα φαίνεται να συνδέονται ποικιλοτρόπως με τις καθιερωμένες υπολογιστικές κλάσεις που είδαμε προηγουμένως (τις οποίες συνδέσεις θα μελετήσουμε εκτενέστερα στο επόμενο κεφάλαιο). Παράλληλα το μοντέλο αυτό, αφενός, αποτελεί έναν από τους γλαφυρότερους τρόπους χαρακτηρισμού των προγραμμάτων που επιδέχονται παραλληλοποίηση στην εκτέλεση τους· πέραν τούτου όμως αποτελεί σε κάθε περίπτωση, ένα μοντέλο που έχει το μεγάλο πλεονέκτημα να παρέχει ολόκληρο το σύνολο των υπολογιστικών του πράξεων άμεσα διαθέσιμο, κάτι που επιτρέπει την απόδειξη ορισμένων κάτω φραγμάτων με συνδυαστικούς και τεχνικούς τρόπους (κάποιους από τους οποίους μελετάμε και στο παρόν κεφάλαιο).

3.1 Ορισμοί

Διαισθητικά ένα υπολογιστικό κύκλωμα είναι το προφανές, δηλαδή μια τοπολογική διάταξη ορισμένων πυλών (από ένα δοθέν σύνολο απλών πυλών που υπολογίζουν κάτι τετριμμένο) οι οποίες συνδυάζουν τις εισόδους με τις εξόδους τους με τέτοιο τρόπο ώστε να καταλήγουν στον υπολογισμό μιας τελικής τιμής που αντιστοιχεί στην έξοδο μιας *boolean* (ή φυσικής) συνάρτησης, την τιμή της οποίας θέλουμε να υπολογίσουμε για την εκάστοτε είσοδο.

Ορισμός 3.1.1. *(Κύκλωμα)* Ορίζουμε ως κύκλωμα C_n n μεταβλητών με εισάριθμο w_i και εξάριθμο w_o πάνω στη βάση B (όπου B ένα σύνολο από λογικές συναρτήσεις $\{0,1\}^{w_i} \to \{0,1\}$) έναν μη κυκλικό, συνεκτικό και κατευθυνόμενο

γράφο, ο οποίος έχει ακριβώς n κόμβους-πηγές (με ετικέτες που αντιστοιχούν στα ονόματα των μεταβλητών εισόδου), έναν ακριβώς κόμβο-καταβόθρα με μία μοναδική εισερχόμενη ακμή (με ετικέτα που αντιστοιχεί στην υπολογιζόμενη συνάρτηση) και με τους υπόλοιπους κόμβους να έχουν ακριβώς w_i εισερχόμενες ακμές και w_o εξερχόμενες, καθώς και μια ετικέτα που αντιστοιχεί σε κάποια από τις βασικές συναρτήσεις της βάσης B. Οι ενδιάμεσοι αυτοί κόμβοι αποκαλούνται συνήθως πύλες.

Από εκεί και πέρα αντιστοιχούμε κάθε κύκλωμα C_n σε μια συνάρτηση $f_{C_n} : \{0,1\}^n \to \{0,1\}$ ορίζοντας αρχικά την εξής αντιστοίχηση των ακμών του κυκλώματος στο σύνολο $\{0,1\}$:

> Για κάθε $x = x_1x_2...x_n$ με $x_i \in \{0,1\}, i = 1,...,n$, οι ακμές που εξέρχονται από τους κόμβους-πηγές έχουν την τιμή που έχει το αντίστοιχο δυφίο της εισόδου.
> Στη συνέχεια, επαγωγικά ορίζουμε την τιμή των w_o εξερχόμενων ακμών ενός κόμβου-πύλη με ετικέτα b ίση με την τιμή $b(y_1,...,y_{w_i})$, όπου $y_1,...,y_{w_i}$ οι τιμές που έχουν λάβει οι εισερχόμενες ακμές. Να σημειώσουμε, ότι από τις υποθέσεις του συνεκτικού, ακυκλικού και κατευθυνόμενου γράφου, προκύπτει ότι ο παραπάνω αλγόριθμος θα τερματίσει έχοντας αποδώσει μία μοναδική καλώς ορισμένη τιμή σε κάθε ακμή του γράφου.

Ορισμός 3.1.2. *(Κυκλωματικός Υπολογισμός)* Ορίζουμε τότε ως έξοδο του κυκλώματος για είσοδο x, την τιμή που έχει αποδοθεί στην μοναδική εισερχόμενη ακμή του κόμβου-καταβόθρα (αφότου έχουμε εφαρμόσει τον παραπάνω αλγόριθμο μέχρι να αποδοθούν τιμές σε όλες τις ακμές) και ως $f_{C_n}(x)$ (ή για απλότητα, όταν δεν προκαλείται σύγχυση,$f_C(x)$ ή ακόμη $C_n(x)$) τη λογική συνάρτηση που αντιστοιχεί κάθε x στην έξοδο του κυκλώματος C_n για αυτή την είσοδο.

Ένας πολύ χρήσιμος σχετικός ορισμός είναι αυτός του πίνακα αληθείας ενός κυκλώματος, ο οποίος αντιστοιχεί στη συμβολοσειρά που παράγεται από τα δυφία εξόδου του κυκλώματος, όταν δώσουμε όλες τις δυνατές εισόδους στο κύκλωμα (σε αλφαριθμητική σειρά):

Ορισμός 3.1.3. Πίνακας αληθείας ενός κυκλώματος C n εισόδων ορίζεται (και συμβολίζεται T_C, ή TT_C ή αν δεν υπάρχει σύγχυση TT) η συμβολοσειρά μήκους 2^n για την οποία ισχύει $T_C[i] = C(i)$ για κάθε $i \in \{0,1\}^n$.

Κάποια χαρακτηριστικά παραδείγματα κυκλωμάτων, είναι αυτά με βάση το σύνολο $B = \{AND, OR, NOT\}$ εισάριθμου 2 και εξάριθμου 1, όπου π.χ. το $AND : \{0,1\}^2 \to \{0,1\}$ αντιστοιχεί στην προφανή συνάρτηση $\{((0,0),0), ((0,1),0),((1,0),0),((1,1),1)\}$ και όμοια οι υπόλοιπες. Να σημειώσουμε εδώ, ότι μπορούμε να ορίσουμε με τον προφανή τρόπο πολλές παραλλαγές στα

επιμέρους σημεία των κυκλωμάτων, για παράδειγμα μπορούμε να επιτρέψουμε απεριόριστο εισάριθμο ή εξάριθμο στις πύλες ή να έχουμε κυκλώματα με μεγαλύτερο πλήθος δυφίων εξόδου, κυκλώματα στα οποία στις εισόδους δίνονται και τα συμπληρώματα των αντίστοιχων μεταβλητών κ.ο.κ και ασφαλώς να επιτρέψουμε πιο εξωτικές πύλες στη βάση μας (οι οποίες όμως, για την περίπτωση του απεριόριστου εισάριθμου, να έχουν μια κοινή πεπερασμένη περιγραφή της λειτουργίας τους η οποία να ισχύει για κάθε μήκος εισόδου). Όπως θα δούμε στη συνέχεια όλες αυτές οι παραλλαγές δεν επηρεάζουν ουσιωδώς την υπολογιστική δυνατότητα των κυκλωμάτων, αλλά συχνά είναι κρίσιμες στις υπολογιστικές δυνατότητες συγκεκριμένων κλάσεων κυκλωμάτων.

Ορισμός 3.1.4. *(Μετρικές κυκλωμάτων)* Επιπλέον ορίζουμε ως μέγεθος $size(C)$ ή $|C|$ του κυκλώματος C το πλήθος των πυλών του και ως βάθος $depth(C)$ το μέγιστο μονοπάτι από έναν κόμβο εισόδου σε έναν κόμβο εξόδου. Επίσης ορίζουμε ως στρώμα βάθους h του κυκλώματος όσους κόμβους του απέχουν απόσταση h από τους κόμβους-πηγές των εισόδων.

Να σημειώσουμε ότι το μέγεθος του κυκλώματος μπορεί να οριστεί και ως το πλήθος των ακμών/συρμάτων του ή ως το άθροισμα αυτών, αλλά επειδή υπάρχει πολυωνυμική σύνδεση μεταξύ τους, όπως θα φανεί στη συνέχεια, για τους σκοπούς που θα το χρησιμοποιήσουμε δεν υπάρχει ουσιώδη διαφορά στο ποιον ορισμό επιλέγουμε.

3.2 Υπολογιστική Ικανότητα Κυκλωμάτων

3.2.1 Υπολογιστές αντί Κυκλωμάτων

Όπως είδαμε, κάθε κύκλωμα ορίζεται μόνο για σταθερό μήκος εισόδου, οπότε αν θέλουμε να το χρησιμοποιήσουμε ως μοντέλο υπολογισμού γενικών συναρτήσεων $f : \mathbb{N} \to \{0,1\}$ θα πρέπει να χρησιμοποιήσουμε οικογένειες κυκλωμάτων. Συγκεκριμένα:

Ορισμός 3.2.1. *(Συνάρτηση Κυκλώματος)* Λέμε ότι μια οικογένεια κυκλωμάτων $\bigcup_{n=1}^{\infty} C_n$ υπολογίζει τη συνάρτηση $f : \mathbb{N} \to \{0,1\}$, όταν για κάθε $x \in \mathbb{N}$ ισχύει $f(x) = C_{|x|}(x)$ όπου $|x|$ το μήκος της εισόδου x (σε δυαδική αναπαράσταση).

Αν θεωρήσουμε ωστόσο αποδεκτή οποιαδήποτε οικογένεια κυκλωμάτων (την οποία συχνά θα αποκαλούμε μη-ομοιόμορφη οικογένεια και τις αντίστοιχες κλάσεις μη-ομοιόμορφες), τότε παίρνουμε ένα κυριολεκτικά πανίσχυρο μοντέλο υπολογισμού καθώς έχουμε την ακόλουθη πρόταση:

Πρόταση 3.2.1. *Κάθε συνάρτηση $f : \mathbb{N} \to \{0,1\}$ είναι υπολογίσιμη από μια μη-ομοιόμορφη οικογένεια κυκλωμάτων το πολύ εκθετικού μεγέθους ως προς το μήκος της εισόδου.*

Απόδειξη. Πράγματι, κάθε λογική συνάρτηση μπορεί να γραφτεί για πεπερασμένο μήκος εισόδου n ως το (λογικό) άθροισμα των ελαχιστόρων στους οποίους επιστρέφει 1. Κάθε ελαχιστόρος χρειάζεται για να σχηματιστεί μια πύλη AND εισάριθμου n (μαζί με συνολικά το πολύ άλλες n πύλες NOT - μία για κάθε πιθανό συμπλήρωμα) και στη συνέχεια απαιτείται μια πύλη OR εισάριθμου το πολύ όσο το πλήθος των απαιτούμενων πυλών AND, δηλαδή το πολύ 2^n. Συνολικά χρειάζεται επομένως ένα κύκλωμα μεγέθους $O(2^n)$ πυλών και βάθους 3. ■

Σημείωση. Όλα αυτά, βεβαίως, στην περίπτωση που υποθέτουμε κυκλώματα απεριόριστου εισάριθμου και με την καθιερωμένη βάση $\{AND, OR, NOT\}$. Όσον αφορά τον εισάριθμο μπορούμε να τον περιορίσουμε σε σταθερή τιμή και τότε το βάθος του κυκλώματος απλώς αυξάνεται σε μία $O(n)$ τιμή χρησιμοποιώντας διαίρει-και-βασίλευε στρατηγική υπολογισμού του τελικού OR, ενώ όσον αφορά τη βάση, αρκεί να έχουμε οποιαδήποτε βάση η οποία είναι πλήρης, δηλαδή η οποία μπορεί να υπολογίσει τις τρεις αυτές συναρτήσεις (είναι εύκολο να δούμε ότι αν θέλουμε, μπορούμε μάλιστα να περιοριστούμε μόνο σε πύλες $NAND$ ή μόνο σε πύλες NOR με έναν σταθερό και μικρό συντελεστή επιβάρυνσης στο τελικό μέγεθος του κυκλώματος – γενικά σε όσες περιπτώσεις θα μελετήσουμε, εκτός κι αν αναφέρεται ρητά αλλιώς, η χρησιμοποιούμενη βάση θα είναι πλήρης). Αποδεικνύεται άλλωστε ότι με μία πιο προσεκτική κατασκευή, κάθε συνάρτηση μπορεί να γραφτεί με ένα κύκλωμα μεγέθους $O(2^n/n)$ [Sha49b, FM05], αλλά δεν χρειάζεται να μπούμε σε λεπτομέρειες επ΄ αυτού.

Ήδη βλέπουμε ότι μια αυθαίρετη οικογένεια κυκλωμάτων μπορεί να είναι απεριόριστα πιο ισχυρή από μια μηχανή Turing κι επομένως δεν αντιστοιχεί σε ένα ρεαλιστικό μοντέλο υπολογισμού (πάντα σύμφωνα με τη θέση των Church-Turing). Να σημειώσουμε μάλιστα ότι το παραπάνω ισχύει για οικογένειες κυκλωμάτων με σχεδόν οποιαδήποτε περιοριστικά χαρακτηριστικά δούμε να θέτουμε για να ορίσουμε τις κλάσεις κυκλωμάτων. Συγκεκριμένα, επειδή όλες οι μη-ομοιόμορφες κλάσεις που θα ορίσουμε θα επιδέχονται τουλάχιστον τα κυκλώματα σταθερού μεγέθους, κι επειδή υπάρχει τετριμμένη οικογένεια κυκλωμάτων σταθερού μεγέθους που αναγνωρίζει τη (προφανώς) μη αποφασίσιμη γλώσσα $L = \{1^n |$ το n-οστό πρόγραμμα τερματίζει για κάθε είσοδο$\}$, προκύπτει ότι όλες αυτές οι κλάσεις θα έχουν μια συνάρτηση που δεν μπορεί να υπολογιστεί από καμία μηχανή Turing . Το μεγαλύτερο ενδιαφέρον μελέτης, όμως, τα κυκλώματα το έχουν στις γλώσσες που περιέχουν μεγάλο πλήθος διαφορετικών στιγμιοτύπων εισόδου για ένα συγκεκριμένο μήκος. Από την άλλη, με την παραπάνω απόδειξη χρειαστήκαμε ένα κύκλωμα εκθετικού μεγέθους για την απεικόνιση μιας τυχαίας συνάρτησης και δε φαίνεται να υπάρχει τρόπος αυτό να μπορεί να συμβεί με μικρότερα κυκλώματα για όλες τις συναρτήσεις (και όπως θα δούμε στη συνέχεια, για τις περισσότερες δε γίνεται), οπότε έχει νόημα να ασχοληθούμε με όσες έχουν όντως αυτή την ικανότητα.

Ορισμός 3.2.2. *(Κλάσεις μεγέθους)* Ορίζουμε ως SIZE(T(n)) το σύνολο

των λογικών συναρτήσεων που επιδέχονται οικογένεια κυκλωμάτων $\bigcup_{n=1}^{\infty} C_n$ τέτοια ώστε $|C_n| < T(n)$ για κάθε (χ.β.τ.γ. αρκετά μεγάλο) n.

Μία πολύ χαρακτηριστική κλάση λογικών συναρτήσεων, είναι αυτή που επιδέχεται κυκλώματα πολυωνυμικού μεγέθους:

Ορισμός 3.2.3. *($P_{/poly}$)*

$$\mathrm{P}_{/\mathrm{poly}} = \bigcup_{\mathrm{c}=1}^{\infty} \mathrm{SIZE}(\mathrm{n}^{\mathrm{c}})$$

3.2.2 Κυκλώματα αντί Υπολογιστών

Προκύπτει το φυσικό ερώτημα, τι συμβαίνει προς την αντίθετη κατεύθυνση. Ήδη είδαμε ότι σίγουρα υπάρχει μια οικογένεια κυκλωμάτων για κάθε συνάρτηση κι άρα και για κάθε συνάρτηση που υπολογίζει μια μηχανή Turing . Με τι αντίτιμο όμως; Αν μια συνάρτηση που απαιτεί πολυωνυμικό χρόνο μέσω μιας μηχανής Turing , απαιτούσε ένα κύκλωμα εκθετικού μεγέθους, τότε τα δύο αυτά μοντέλα υπολογισμού θα μπορούσε να αποτελούν δύο πολύ διαφορετικές οπτικές του υπολογισμού (το οποίο αναμφίβολα θα είχε επιπτώσεις και στον τρόπου που θα συνδέονταν οι δύο αυτές όψεις). Ευτυχώς κάτι τέτοιο δε συμβαίνει καθώς όπως θα δούμε μία μηχανή που χρειάζεται χρόνο $T(n)$ μπορεί να υπολογιστεί με μια οικογένεια κυκλωμάτων μεγέθους $O(T(n)^2)$.

Θεώρημα 3.1. *Αν μια γλώσσα αναγνωρίζεται από μια μηχανή χρόνου $T(n)$, τότε υπάρχει οικογένεια κυκλωμάτων μεγέθους $O(T(n)^2)$ που την υπολογίζει.*

Απόδειξη. Θα κατασκευάσουμε ένα κύκλωμα που έχει $O(T(n))$ βάθος και το i-οστό «στρώμα» αντιστοιχεί στην κατάσταση της μηχανής Turing που προσομοιώνει μετά από i βήματα.

Μπορούμε να φανταστούμε ότι κάθε κατάσταση/στρώμα περιέχει $2 * T(n)$ σύρματα (ακμές του γράφου του κυκλώματος) που φέρουν την πληροφορία που αντιστοιχεί στην ταινία της μηχανής Turing θεωρώντας ότι η κεφαλή βρίσκεται πάντα στο μεσαίο κελί (δεν θα χρειαστούν επιπλέον, καθώς σε χρόνο $T(n)$ δεν θα μπορέσει προφανώς να χρησιμοποιήσει περισσότερα κελιά της ταινίας και επιπλέον όπου και να βρίσκεται η κεφαλή, τα υπόλοιπα $T(n) - 1$ κελιά της θα χωρούν πάντοτε δεξιά ή αριστερά της όταν έχουμε συνολικά $2 * T(n)$ σύρματα), καθώς και $|Q|$ σύρματα (όπου Q το σύνολο των καταστάσεων της μηχανής Τιούριγνκ) που κωδικοποιούν ποια από τις καταστάσεις είναι η τρέχουσα στο αντίστοιχο βήμα. Συνολικά η προσομοίωση κάθε κατάστασης δεν απαιτεί πάνω από $O(T(n))$ σύρματα.

Από εκεί και πέρα, μπορούμε να προχωρήσουμε με επαγωγή. Προφανώς είναι πολύ εύκολο να μοντελοποιήσουμε την αρχική κατάσταση: Έχουμε ότι n από τα $T(n)$ σύρματα περιέχουν την είσοδο, οπότε απλά χρησιμοποιούμε τα

σύρματα της εισόδου για αυτά (τοποθετώντας τα αμέσως μετά το μεσαίο κελί, ώστε να συμβαδίζει και με την αρχική θέση της κεφαλής) ενώ για τα υπόλοιπα απλά τους δίνουμε σταθερή είσοδο που να κωδικοποιεί το κενό (θεωρώντας ότι είτε το έχουμε δεδομένο είτε το κατασκευάζουμε με μια-δυο πύλες)[1]. Όμοια για τα σύρματα της κατάστασης θέτουμε 1 μόνο στο σύρμα που αντιστοιχεί στην αρχική κατάσταση q_0. Αν τώρα έχουμε μοντελοποιήσει μέχρι το i-οστό βήμα, τότε μπορούμε να μοντελοποιήσουμε και την κατάσταση του $(i+1)$-οστού βήματος με ένα κομμάτι κυκλώματος σταθερού βάθους και $T(n)$ πυλών. Αυτό θα το κάνουμε εκμεταλλευόμενοι την υψηλή τοπικότητα του υπολογισμού των μηχανών Turing, που είδαμε και στην απόδειξη της NP-πληρότητας του SAT. Πρώτα από όλα υπάρχει ένα απλό κύκλωμα $O(T(n))$ που μετατρέπει την κατάσταση της ταινίας σε μία νέα κατάσταση που η κεφαλή έχει μετατοπιστεί αριστερά, ένα ακόμη που έχει μετατοπιστεί δεξιά και δύο ακόμη που η κεφαλή δεν έχει μετατοπιστεί αλλά έχει γίνει το τρέχον κελί 0 ή 1. Επομένως για κάθε κελί έχουμε 4 σύρματα, ένα για κάθε πιθανή δράση της μηχανής Turing στο επόμενο βήμα. Εκμεταλλευόμενοι λοιπόν την τοπικότητα του υπολογισμού, του οποίου τα βήματα και οι κανόνες είναι καθολικοί κι εξαρτώμενοι μόνο από την τρέχουσα τιμή του κελιού της κεφαλής, αρκεί να λάβουμε την τιμή του μεσαίου κελιού καθώς και την τιμή της κατάστασης και να αποφασίσουμε ποιο από τα 4 σύρματα είναι αυτό που θα αντιστοιχήσει στην επόμενη κατάσταση. Αυτό ωστόσο για κάθε κελί είναι πολύ απλό με χρήση ενός απλού κυκλώματος σε μορφή πολυπλέκτη το οποίο παίρνει την τρέχουσα κατάσταση και το τρέχον κελί (σύνολο $log|Q|+1$ (ή πιο απλά $|Q|+1$) δυφία στον επιλογέα) κι επιστρέφει τον τύπο της δράσης από τον οποίο διαλέγουμε πάλι με αντίστοιχο κύκλωμα ποιο από τα 4 σύρματα κάθε κελιού είναι το έγκυρο, καθώς και τη νέα κατάσταση. Επειδή το σύνολο των καταστάσεων και κανόνων της μηχανής Turing είναι πεπερασμένο, το παραπάνω κύκλωμα είναι σταθερού μεγέθους για κάθε σύρμα-κελί. Συνολικά για την μοντελοποίηση του $(i+1)$-οστού βήματος χρειάστηκαν λοιπόν $O(T(n))$ πύλες και το επαγωγικό βήμα έχει ολοκληρωθεί.

Τέλος για το $T(n)$-οστό βήμα, έχουμε ότι υπάρχει ένα τετριμμένο σταθερό κύκλωμα που ελέγχει αν η τρέχουσα κατάσταση είναι κατάσταση αποδοχής και επιστρέφει 1 μόνο τότε. Το συνολικό κύκλωμα έχει $T(n)$ στρώματα, κάθε ένα εκ των οποίων έχει $O(T(n))$ πύλες κι επομένως συνολικά χρειάστηκαν $O(T^2(n))$ πύλες.

■

Σημείωση. Ι Το παραπάνω άνω φράγμα ασφαλώς δεν είναι στενό, καθώς αυτό θα υπονοούσε ότι μια μηχανή Turing υπερεκθετικού χρόνου (η οποία όπως έχουμε δει υπάρχει) χρειάζεται και υπερεκθετικό κύκλωμα, ενώ είδαμε ότι δεν υπάρχει ποτέ λόγος να κατασκευάσουμε ένα κύκλωμα μεγέθους μεγαλύτερου από $O(2^n n)$.

[1] Δεδομένου ότι θα χρειαστούμε περισσότερα σύμβολα για τα σύμβολα του αλφαβήτου, κάνουμε μια κατάλληλη κωδικοποίηση ξοδεύοντας περισσότερα από ένα δυφία για κάθε σύμβολο.

Σημείωση. Η Παρατηρεί κανείς, ότι αν και αρκετά εύγλωττη η παραπάνω μοντελοποίηση, εν τούτοις είναι πολύ σπάταλη στο γεγονός ότι σε κάθε βήμα αντιγράφουμε ολόκληρη την ταινία, ενώ η κεφαλή κουνήθηκε μόλις κατά ένα βήμα. Στην πραγματικότητα, υπάρχει μία πιο έξυπνη τοποθέτηση των κελιών της ταινίας έτσι ώστε οι μετατοπίσεις και αντιγραφές να γίνονται για πολλά κελιά λίγες φορές, οδηγώντας συνολικά σε ένα κύκλωμα μεγέθους $O(T(n)logT(n))$ [Für82]. Η απόδειξη παρουσιάζεται στο Παράρτημα Α΄.1.

Από το παραπάνω θεώρημα, προκύπτει άμεσα ότι $\mathrm{P} \subset \mathrm{P}_{/\mathrm{poly}}$. Η σχέση υποσυνόλου είναι ασφαλώς γνήσια, καθώς όπως είδαμε υπάρχουν γλώσσες στο $SIZE(O(1))$ που δεν αποφασίζονται από καμιά (πολυωνυμική ή άλλη) μηχανή Turing (κάτι που είναι εν μέρει αναμενόμενο, δεδομένου ότι σκοπός είναι να εισάγουμε μία νέα μορφή υπολογισμού και όχι μία ακόμη που να είναι ισοδύναμη με το παραδοσιακό μοντέλο). Επίσης, όπως είδαμε κάθε υπολογιστική κλάση είναι υποσύνολο της αντίστοιχης κυκλωματικής με μία πολυωνυμική επιβάρυνση, αλλά τα κυκλώματα αυτά που κατασκευάσαμε χαρακτηρίζονται από πολλή περιττή κι επαναλαμβανόμενη πληροφορία. Τίθεται, λοιπόν, το ερώτημα αν υπάρχει μία μη τετριμμένη κλάση, η οποία να μη μπορεί να υπολογιστεί ομοιόμορφα σε σύντομο χρόνο, αλλά να επιδέχεται σύντομα κυκλώματα (π.χ. αν $EXP \subseteq P_{/poly}$) ή έστω αν το ίδιο ισχύει για μία κλάση για την οποία δε πιστεύουμε ότι έχει πολυωνυμικό αλγόριθμο (π.χ. αν $NP \subseteq P_{/poly}$).

Τέτοιας φύσεως ερωτήματα θα μας απασχολήσουν έντονα στα κεφάλαια που ακολουθούν, αλλά μέχρι στιγμής η μόνη κλάση για την οποία υπάρχει αποδεδειγμένα μία τέτοια οικογένεια πολυωνυμικών κυκλωμάτων χωρίς να γνωρίζουμε πολυωνυμικό αλγόριθμο είναι η BPP.

Θεώρημα 3.2 ([Adl78])**.** $BPP \subseteq P_{/poly}$

Απόδειξη. Μια γλώσσα L στο BPP, ως γνωστόν, έχει μια μηχανή Turing, η οποία για κάθε x με είσοδο το $\langle x, y\rangle$ (όπου $|y| < poly(|x|)$) αποδέχεται για τη μεγάλη πλειοψηφία των y αν και μόνο αν $x \in L$ κι αντίστοιχα απορρίπτει για τη μεγάλη πλειοψηφία των y αν $x \notin L$. Όπως επίσης είδαμε, μπορούμε να μειώσουμε το ποσοστό λάθους μέχρι και σε εκθετικά ως προς το x επίπεδα διατηρώντας τον πολυωνυμικό τυχαιοκρατικό χρόνο. Έστω λοιπόν ότι η M έχει σφάλμα για κάθε είσοδο το πολύ 2^{-n^2} όπου $n = |x|$. Έστω επίσης ότι $|y| < |x|^c$. Έχουμε λοιπόν ότι για κάθε x μήκους n υπάρχουν 2^{n^c} διαφορετικοί τυχαίοι συνδυασμοί των δυφίων του y, εκ των οποίων μόλις οι $2^{n^c-n^2}$ είναι εσφαλμένοι (δηλαδή δίνουν διαφορετικό αποτέλεσμα από ό,τι η πλειοψηφία). Ωστόσο υπάρχουν 2^n διαφορετικές είσοδοι και κάθε μία από αυτές έχει το πολύ $2^{n^c-n^2}$ εσφαλμένα y κι άρα συνολικά υπάρχουν το πολύ $2^{n^c-n(n-1)} < (2^{n^c})$ εσφαλμένα y κι όλα τα υπόλοιπα (από τα 2^{n^c} στο σύνολο) δίνουν σωστό αποτέλεσμα για κάθε είσοδο x. Έστω λοιπόν y^* ένα από αυτά και θα έχουμε τότε $M(x, y^*) = 1$ αν και μόνο αν $x \in L$ για κάθε είσοδο x μήκους n. Όμως, όπως είδαμε, η M επιδέχεται πολυωνυμικά κυκλώματα κι άρα υπάρχει ένα κύκλωμα C_n πολυωνυμικού μεγέθους με είσοδο το $\langle x, y\rangle$ που υπολογίζει την L για εισόδους μήκους n (ως προς το x). Σταθεροποιώντας λοιπόν την είσοδο του y

στο y^* προκύπτει ένα κύκλωμα (προφανώς και πάλι πολυωνυμικού μεγέθους) με είσοδο το x και που υπολογίζει ορθά την L για κάθε τέτοια είσοδο μήκους n και η απόδειξη ολοκληρώθηκε. ■

3.2.3 Μη-ομοιόμορφη Κυκλωματική Ιεραρχία

Η Σημείωση Ι του προηγούμενου ερωτήματος οδηγεί ωστόσο και σε κάποια επόμενα εύλογα ερωτήματα. Πέραν από καλύτερη προσομοίωση των μηχανών Turing (όλων ή κάποιων) με μικρότερα κυκλώματα, μπορεί αυτό να επεκταθεί και στα ίδια τα κυκλώματα; Δηλαδή ισοδύναμα, υπάρχουν κυκλώματα που προσομοιώνουν κάθε λογική συνάρτηση αλλά με υποεκθετικό μέγεθος; Για την ερώτηση αυτή, έχουμε στέρεα αρνητική απάντηση:

Θεώρημα 3.3 ([Sha49b]). *(Μη-ομοιόμορφη κυκλωματική ιεραρχία) Υπάρχει σταθερά c, για την οποία για οποιεσδήποτε φυσικές συναρτήσεις $T_1(n), T_2(n)$ με $2^n/n > T_1(n) > c * T_2(n) > n$, ισχύει*

$$SIZE(T_1(n)) \nsubseteq SIZE(T_2(n))$$

Για την απόδειξη θα χρειαστούμε το ακόλουθο Λήμμα:

Λήμμα 3.3.1. *(Ύπαρξη δύσκολων συναρτήσεων) Υπάρχει σταθερά c, τέτοια ώστε*

$$\{f | f : \mathbb{N} \to \{0,1\}\} \nsubseteq SIZE(\frac{2^n}{cn})$$

Απόδειξη. (Λήμμα 3.3.1) Κάθε κύκλωμα μεγέθους S και n εισόδων, μπορεί να γραφτεί με μία αναπαράσταση μήκους $dSlogS$ (μέσω κάποιας από τις κλασικές κωδικοποιήσεις), επομένως υπάρχουν το πολύ 2^{dSlogS} τέτοια κυκλώματα μεγέθους S. Από την άλλη υπάρχουν 2^{2^n} συναρτήσεις $\{0,1\}^n \to \{0,1\}$, επομένως για $c > d$ έχουμε ότι τα κυκλώματα μεγέθους $\frac{2^n}{cn}$ μπορούν να υπολογίσουν το πολύ $2^{\frac{d2^n}{cn} log(\frac{2^n}{cn})} < 2^{2^n}$, δηλαδή υπάρχει τουλάχιστον μία συνάρτηση που δεν μπορούν να υπολογίσουν και η απόδειξη ολοκληρώθηκε. ■

Σημείωση. Στην πραγματικότητα δεν είναι τουλάχιστον μία, αλλά σχεδόν όλες, καθώς το ποσοστό των υπολογιζόμενων προς όλες είναι

$$\frac{2^{\frac{d}{c}2^n}}{2^{2^n}} = 2^{-\frac{c}{d}2^n}$$

το οποίο πέφτει γρήγορα στο 0.

Χρησιμοποιώντας τώρα το παραπάνω Λήμμα και ρυθμίζοντας κάποιες από τις παραμέτρους, είμαστε σε θέση να αποδείξουμε το αρχικό Θεώρημα.

Απόδειξη. (Θεώρημα 3.3) Από το Λήμμα 3.3.1, έχουμε ότι υπάρχει συνάρτηση $F : \{0,1\}^w \to \{0,1\}$ που δεν υπολογίζεται από κύκλωμα μεγέθους $2^w/cw$, αλλά υπολογίζεται από κύκλωμα μεγέθους $d2^w w$. Διαλέγουμε w τέτοιο ώστε $T_2(n) > d2^w w$ και $T_1(n) < 2^w/cw$ (μπορούμε χάρη στους περιορισμούς που θέσαμε στην υπόθεση) και παίρνουμε τη συνάρτηση n εισόδων που υπολογίζει την F στα πρώτα w δυφία (λόγω των ανισοτήτων που έχουμε θέσει στους περιορισμούς είναι σίγουρα $n \geq w$). Αυτή η συνάρτηση προφανώς είναι στο $SIZE(d2^w w) \subseteq SIZE(T_2(n))$ αλλά όχι στο $SIZE(2^w/cw)$ κι άρα ούτε στο $SIZE(T_1(n))$ ολοκληρώνοντας την απόδειξη. ■

Παρότι, λοιπόν, μεταξύ κυκλωμάτων έχουμε την ανωτέρω καθαρή ιεραρχία, για ακόμη μια φορά δεν έχουμε κάποια αντίστοιχη για τη σχέση μεταξύ υπολογιστικών και κυκλωματικών κλάσεων. Για παράδειγμα, μέχρι και σήμερα, δε γνωρίζουμε κατά πόσο το $DTIME[n^5]$, το $DTIME[n^{logn}]$ ή ακόμη και το $DTIME[2^n]$ είναι υποσύνολο του $SIZE[n^2]$. Αυτή η συσχέτιση, μέχρι και σήμερα, διασαφηνίζεται με έντονα βραδείς ρυθμούς και είναι ένα από τα κυρίως θέματα που θα μας απασχολήσουν στη συνέχεια. Προς το παρόν, μπορούμε να δώσουμε τον ακόλουθο ορισμό της (κυκλωματικής) δυσκολίας μιας λογικής συνάρτησης, η οποία μπορεί να εφαρμοστεί ασφαλώς και στις παραδοσιακές υπολογιστικές κλάσεις.

Ορισμός 3.2.4. *(Δυσκολία λογικής συνάρτησης)* Έστω $f : \{0,1\}^* \to \{0,1\}$ μία λογική συνάρτηση. Ορίζουμε ως δυσκολία αυτής $H_f(n)$ το μέγεθος του ελάχιστου κυκλώματος C_n^f που την υπολογίζει ορθά για όλες τις εισόδους μήκους n ($H_f(n) = min\{|C_n^f||\forall x(|x| = n \Rightarrow C_n^f(x) = f(x))\}$).

3.2.4 Παραγωγή Κυκλωμάτων από Μηχανές Turing

Όπως είδαμε τα κυκλώματα στη μη-ομοιόμορφη έκδοση τους είναι παντοδύναμα υπολογιστικά μοντέλα και συνεπώς δεν φαίνεται να μπορούν να ανταποκριθούν σε κάποιο ρεαλιστικό μοντέλο. Ο λόγος που συμβαίνει αυτό, είναι ότι μπορούμε να έχουμε έναν διαφορετικό αλγόριθμο για κάθε μήκος εισόδου, άρα ο συνολικός αλγόριθμος περιέχει μη πεπερασμένη πληροφορία, ενώ στις παραδοσιακές μηχανές Turing ο αλγόριθμος έχει πεπερασμένη περιγραφή (ίδιο με την περιγραφή της μηχανής). Αυτό γίνεται πιο ξεκάθαρο αν μελετήσουμε μία ελαφρά διαφορετική έκδοση των μηχανών Turing , τις μηχανές Turing με συμβουλή.

Ορισμός 3.2.5. (Μηχανές Turing με συμβουλή) Μία γλώσσα αναγνωρίζεται από μία μηχανή Turing με $s(n)$ δυφία συμβουλής όταν υπάρχει μηχανή M και μία ακολουθία $s_n : \mathbb{N} \to \{0,1\}^{s(n)}$ τέτοια ώστε

$$x \in L \Leftrightarrow M(x, s_{|x|}) = 1$$

Όταν η μηχανή με είσοδο το (x, s_n) τρέχει για χρόνο $T(n)$ (όπου $n = |x|$) (και μόνο τότε) λέμε ότι ανήκει στην κλάση $DTIME(T(n))/s(n)$.

Αυτό που λέει ο παραπάνω ορισμός είναι ότι για κάθε μήκος εισόδου, η μηχανή έχει πρόσβαση σε μια (αυθαίρετη) πληροφορία μήκους $s(n)$ για την εκτέλεση της. Η ισχύς της διαισθητικής περιγραφής της προηγούμενης παραγράφου, λοιπόν, μπορεί να αποδειχθεί από την ακόλουθη πρόταση, την οποία δείχνουμε εδώ για τη κλάση $P_{/poly}$, αλλά μπορεί να επεκταθεί με το φυσικό τρόπο για παρόμοιες:

Πρόταση 3.2.2 ([KL82]). *Μια γλώσσα ανήκει στο $P_{/poly}$ αν και μόνο αν αναγνωρίζεται από μια πολυωνυμική μηχανή που δέχεται πολυωνυμική συμβουλή.*

Απόδειξη. Πράγματι αν ανήκει στο $P_{/poly}$, τότε δίνουμε ως συμβουλή την περιγραφή του κυκλώματος σε μια μηχανή η οποία προσομοιώνει το κύκλωμα που διαβάζει από τη συμβουλή για την είσοδο που δέχθηκε (προφανώς σε πολυωνυμικό χρόνο). Αντίστροφα η μηχανή, εφόσον είναι πολυωνυμικού χρόνου υπολογίζεται από μια πολυωνυμική οικογένεια κυκλωμάτων σύμφωνα με την προσομοίωση που είδαμε στο Θεώρημα 3.1, η οποία ωστόσο τροποποιείται έτσι ώστε στην αρχική κατάσταση πέρα από τα δυφία της εισόδου, έχουμε συνδέσει σταθερά και τα δυφία της συμβουλής. ■

Η δυνατότητα της άπειρης πληροφορίας της συμβουλής λοιπόν, όπως φαίνεται, είναι αρκετή για να οδηγήσει σε ένα μοντέλο που δεν ανταποκρίνεται σε κάποιο πραγματικό (πάντα σύμφωνα με τη θέση των Church-Turing). Ένας τρόπος λοιπόν να ασχοληθούμε με πιο ρεαλιστικά μοντέλα είναι οι ομοιόμορφες οικογένειες κυκλωμάτων, δηλαδή αυτές που έχουν μια μηχανή Turing, η οποία παράγει την περιγραφή του C_n με είσοδο το 1^n. Αν για παράδειγμα υποχρεώσουμε η μηχανή αυτή να τρέχει σε πολυωνυμικό χρόνο, τότε η αντίστοιχη κλάση $P_{/poly}^{uniform}$ ισούται προφανώς με το P. Μεγαλύτερο ενδιαφέρον παρουσιάζουν οι περιπτώσεις που ζητάμε η μηχανή που παράγει το κύκλωμα να ανήκει επιπλέον στο $L = LOGSPACE$, οπότε και οι αντίστοιχες οικογένειες αποκαλούνται λογαροχωρο-ομοιόμορφες. Συγκεκριμένα έχουμε την ακόλουθη ισχυρότερη πρόταση:

Πρόταση 3.2.3. *Μια γλώσσα έχει λογαροχώρο-ομοιόμορφα κυκλώματα αν και μόνο αν είναι στο P.*

Απόδειξη. (Σχέδιο) Προφανώς επειδή $L \subseteq P$ ισχύει ότι οι αντίστοιχες οικογένειες είναι στο P. Από την απόδειξη του Θεωρήματος 3.1 προκύπτει ωστόσο ότι και κάθε γλώσσα στο P είναι λογαροχωρο-ομοιόμορφη, καθώς α) σε κάθε στρώμα τοποθετούμε τις ίδιες πύλες και β) σε κάθε σύρμα-κελί εφαρμόζουμε τα ίδια σταθερά κυκλώματα επιλογής της επόμενης κατάστασης τους, οπότε παίρνοντας έναν λογαριθμικού χώρου δείκτη d μπορούμε με *modulo* πράξεις (που εκτελούνται σε λογαροχώρο και λογαροχρόνο) να προσδιορίσουμε ποια είναι η αντίστοιχη d-οστή πύλη ή σύνδεση του κυκλώματος. Επομένως με είσοδο 1^n, το μόνο που χρειάζεται είναι ένας απαριθμητής που επιστρέφει το i-οστό στοιχείο του κυκλώματος όπου το i χωράει και μπορεί να αυξάνεται προφανώς σε

λογαροχώρο (εφόσον το συνολικό κύκλωμα είναι πολυωνυμικό). Εν τέλει προκύπτει ότι όλα μπορούν να παραχθούν σε λογαροχώρο κι άρα ολοκληρώνεται η απόδειξη. ■

3.3 Κλάσεις Κυκλωμάτων

Αντίστοιχα με τις χρονικές (ή χωρικές) κλάσεις των μηχανών Turing , μπορούμε να ορίσουμε και κλάσεις στις οικογένειες κυκλωμάτων που υπολογίζουν μια συνάρτηση. Εν τούτοις η φύση των κυκλωμάτων, όπως είδαμε, είναι διαφορετική κι έτσι για αρχή δεν έχει νόημα να ορίσουμε μια γενική κλάση της μορφής $EXPSIZE$ (κατ΄ αντιστοιχία με τη χρονική κλάση EXP) καθώς όλα τα κυκλώματα θα ανήκουν σε αυτήν (αντίστοιχα δεν έχει πολύ νόημα να ορίσουμε μια κλάση $LOGSIZE$ καθώς η είσοδος είναι μέρος του κυκλώματος κι άρα δε γίνεται ένα κύκλωμα να έχει υπογραμμικό μέγεθος – εν τούτοις ορίζεται η κλάση $L_{/poly}$ [Aar05] η οποία θα μπορούσε να θεωρηθεί το ανάλογο αυτής).

Από την άλλη, ένας από τους λόγους που αναπτύχθηκε η θεωρία των κυκλωμάτων είναι η στενή της σύνδεση με τη παραλληλοποίηση των προγραμμάτων, όπου τότε ο χρόνος εκτέλεσης ισούται με το χρόνο που χρειάζονται τα παράλληλα προγράμματα να εκτελεστούν, ο οποίος αντιστοιχεί διαισθητικά στο βάθος του κυκλώματος. Τα κριτήρια με τα οποία ταξινομούμε τις κλάσεις στα κυκλώματα είναι, λοιπόν, όχι σκέτο το συνολικό μέγεθος, αλλά αυτό σε συνδυασμό με το βάθος του κυκλώματος αλλά και με το είδος των πυλών, το οποίο όπως θα φανεί στη συνέχεια είναι από τους πιο κρίσιμους παράγοντες στον χαρακτηρισμό μιας κλάσης.

Έχουμε ήδη ορίσει τη πιο γενική κλάση $P_{/poly}$ που ισούται με τις γλώσσες που επιδέχονται οικογένειες κυκλωμάτων πολυωνυμικού μεγέθους (θεωρώντας τη καθιερωμένη βάση $B = \{AND, OR, NOT\}$ και εισάριθμο ίσο με 2 – οποιαδήποτε τροποποίηση σε αυτές τις παραμέτρους δεν μεταβάλλει τη συγκεκριμένη κλάση, όπως ελέγχεται εύκολα).

Θα ορίσουμε τώρα ορισμένες ενδιαφέρουσες και ευρεία μελετημένες κλάσεις οικογενειών κυκλωμάτων. Να σημειώσουμε ότι από εδώ και πέρα αναφερόμαστε πάντοτε σε μη ομοιόμορφες κλάσεις και δε θα το διευκρινίζουμε περαιτέρω. Εξαίρεση αποτελεί η κλάση NC, η οποία επειδή σύμφωνα με την καθιερωμένη σημειογραφία, αντιστοιχεί σε ομοιόμορφη κλάση κυκλωμάτων, θα τη σημειώνουμε με NC_{non-u} για να είναι βέβαιη η διάκριση.

Ορισμός 3.3.1. *(NC^d_{non-u})* Η κλάση NC^d_{non-u} είναι το σύνολο των γλωσσών που αποφασίζονται από μια μη ομοιόμορφη οικογένεια κυκλωμάτων που χρησιμοποιεί την καθιερωμένη βάση και εισάριθμο 2, η οποία έχει κυκλώματα πολυωνυμικού μεγέθους $O(n^c)$ και βάθους $O(log^d n)$.

Ορισμός 3.3.2. *(AC^d)* Η κλάση AC^d είναι το σύνολο των γλωσσών που αποφασίζονται από μια μη ομοιόμορφη οικογένεια κυκλωμάτων που χρησιμο-

ποιεί την καθιερωμένη βάση και απεριόριστο εισάριθμο, η οποία έχει κυκλώματα πολυωνυμικού μεγέθους $O(n^c)$ και βάθους $O(log^d n)$.

Η μεγάλη διαφορά μεταξύ των κλάσεων NC^d_{non-u}, AC^d φαίνεται ήδη για $d = 0$ όπου έχουμε σταθερό βάθος. Τότε το NC^0_{non-u} μπορεί να εξαρτάται από το πολύ 2^d δυφία της εισόδου κι άρα δεν μπορεί να υπολογίσει καμία ιδιαίτερα ενδιαφέρουσα γλώσσα (για παράδειγμα είναι εύκολο να δείξει κανείς ότι δεν αποφασίζει τη γλώσσα $\{1^n | n \in \mathbb{N}\}$, η οποία προφανώς ανήκει στην AC^0).

Ορισμός 3.3.3. *(TC^d)* Η κλάση TC^d είναι το σύνολο των γλωσσών που αποφασίζονται από μια μη ομοιόμορφη οικογένεια κυκλωμάτων που χρησιμοποιεί τη βάση $B = \{MAJ, NOT\}$ (όπου $MAJ(x_1, ..., x_n) = 1 \Leftrightarrow \sum_{i=1}^{n} x_i \geq \frac{n}{2}$ η συνάρτηση πλειοψηφίας) και απεριόριστο εισάριθμο, η οποία έχει κυκλώματα πολυωνυμικού μεγέθους $O(n^c)$ και βάθους $O(log^d n)$.

Η τελευταία αυτή κλάση δεν θα μας απασχολήσει έντονα, αλλά έχει ιδιαίτερο ενδιαφέρον για το γεγονός ότι κατέχει πρωταγωνιστικό ρόλο στα νευρωνικά δίκτυα (όπου ένα σήμα παίρνει μια τιμή ανάλογα με το αν η είσοδος προσπερνάει ένα κατώφλι). Μπορούμε εύκολα να δείξουμε ότι ισχύει το ακόλουθο (βλ. [CK02]):

Θεώρημα 3.4.

$$NC^d_{non-u} \subseteq AC^d \subseteq TC^d \subseteq NC^{d+1}_{non-u}$$

Απόδειξη. Είναι προφανές ότι ισχύει η πρώτη σχέση $NC^d_{non-u} \subseteq AC^d$.
Για τη δεύτερη σχέση αρκεί να δείξουμε ότι η πύλη MAJ σε συνδυασμό με τη NOT μπορεί να υπολογίσει τις AND και OR. Πράγματι είναι εύκολο να επαληθεύσουμε ότι ισχύει $AND(x_1, x_2, ..., x_n) = MAJ(x_1, 0, x_2, 0, ..., 0, x_n)$ κι άρα αρκεί να αντικαταστήσουμε κάθε πύλη AND n εισόδων με μια πύλη MAJ $2n-1$ εισόδων, εκ των οποίων οι $n-1$ να ισούνται σταθερά με 0, χωρίς να μεταβάλλουμε την τάξη του βάθους ή του μεγέθους (κι άρα παραμένουμε στο TC^d). Παρόμοια έχουμε $OR(x_1, x_2, ..., x_n) = MAJ(x_1, 1, x_2, 1, ..., 1, x_n)$.
Για τη τρίτη σχέση, αρκεί να παρατηρήσουμε ότι μπορούμε να κατασκευάσουμε ένα NC^1_{non-u} κύκλωμα λογαριθμικού βάθους και πολυωνυμικού μεγέθους που υπολογίζει πόσους άσσους έχει η είσοδος κι ένα ακόμη μικρό κύκλωμα επίσης λογαριθμικού βάθους που επιστρέφει 1 αν είναι πάνω από τους μισούς (ή γενικότερα αν προσπερνάν ένα κατώφλι). Αντικαθιστώντας λοιπόν κάθε πύλη MAJ με ένα τέτοιο κύκλωμα (το οποίο θα έχει το πολύ $O(n^c)$ εισόδους και $O(log(n^c)) = O(logn)$ βάθος) προκύπτει ότι το συνολικό μέγεθος παραμένει πολυωνυμικής τάξης, ενώ το βάθος θα έχει πολλαπλασιαστεί με έναν συντελεστή $O(logn)$ κι άρα θα έχουμε κατασκευάσει ένα ισοδύναμο κύκλωμα που ανήκει στην κλάση NC^{d+1}_{non-u}. ■

Ο λόγος που ζητάμε τα κυκλώματα να έχουν κάθε φορά πολυλογαριθμικό βάθος (δηλαδή της τάξης $O(log^d n)$ για κάποιο n) είναι επειδή αυτά τα κυκλώματα (δηλαδή όσα ανήκουν στην κλάση $NC = \cup_{d=0}^{\infty} NC^d$) χαρακτηρίζουν τα αποδοτικώς παραλληλοποιήσιμα προγράμματα [AB09, p.116–118]. Πράγματι ένα

πρόγραμμα που παραλληλοποιείται σε $poly(n)$ επεξεργαστές, έκαστος εκ των οποίων χρειάζεται $polylog(n)$ χρόνο για να εκτελεστεί μπορεί να μοντελοποιηθεί από ένα κύκλωμα που συνδυάζει τη δουλειά όλων αυτών των επεξεργαστών (και που προφανώς θα έχει επίσης βάθος $polylog(n)$ χάρη στο Θεώρημα 3.1 και ασφαλώς πολυωνυμικό μέγεθος). Αντίστροφα, επειδή σε ένα κύκλωμα που ανήκει στο NC κάθε στρώμα πυλών (δηλαδή κάθε σύνολο που περιέχει όλες τις πύλες του κυκλώματος που απέχουν την ίδια απόσταση από τις εισόδους) δεν θα έχει παραπάνω από $poly(n)$ μέγεθος, μπορούμε να θεωρήσουμε $poly(n)$ επεξεργαστές δικτυωμένους έτσι ώστε κάθε ένας να είναι υπεύθυνος για τον υπολογισμό μιας σειράς από πύλες (βάθους το πολύ $polylog(n)$) και να προωθεί κάθε φορά τα αποτελέσματα του στους κατάλληλους επεξεργαστές, παίρνοντας συνολικά $polylog(n)$ χρόνο.

3.4 Οι κλάσεις AC^0 και ACC^0

Είδαμε τις σχέσεις υποσυνόλου που επικρατούν μεταξύ των κλάσεων που ορίσαμε, για τις οποίες ωστόσο δεν αναφέραμε τίποτα για το κατά πόσο είναι σχέσεις γνήσιου υποσυνόλου και κατά πόσο ορίζεται μια αντίστοιχη ιεραρχία βάθους. Γνωρίζουμε σίγουρα ότι το NC^0_{non-u} είναι γνήσιο υποσύνολο του AC^0 χάρη στο τετριμμένο παράδειγμα που δώσαμε παραπάνω. Τι συμβαίνει ωστόσο με τις κλάσεις AC^0 και TC^0; Με μία μακρά πιο περίτεχνη απόδειξη μπορούμε να δείξουμε ότι είναι επίσης σχέση γνήσιου υποσυνόλου. Και εκεί είναι που σταματάει η τωρινή γνώση μας για τη γνησιότητα αυτών των σχέσεων. Αυτό επεκτείνεται και στις υπολογιστικές κλάσεις, καθώς με όσα γνωρίζουμε μέχρι στιγμής, θα μπορούσε να είναι συνεπές ότι ολόκληρο το P, NP, PH, αλλά μέχρι και ολόκληρο το $NEXP$ ανήκει στο TC^0!

3.4.1 $AC^0 \subsetneq TC^0$

Για να δείξουμε τη γνησιότητα της σχέσης, θα αποδείξουμε ότι ένα συγκεκριμένο πρόβλημα δεν είναι στο AC^0 και μετά θα παρουσιάσουμε ένα TC^0 κύκλωμα που το αποφασίζει. Το πρόβλημα αυτό θα είναι το πρόβλημα αρτιότητας $PARITY = \{x | \text{το } x \text{ έχει άρτιο πλήθος από 1}\}$.

Θεώρημα 3.5 ([FSS84, Ajt83])**.**

$$PARITY \notin AC^0$$

Απόδειξη. Η απόδειξη εκμεταλλεύεται την εξής ιδιότητα των AC^0 κυκλωμάτων: Όταν ένα ποσοστό (που τείνει στο 1) από τυχαία δυφία εισόδου σταθεροποιείται σε μια τυχαία τιμή, τότε το προκύπτον κύκλωμα είναι σταθερό (δηλαδή έχει έξοδο σταθερά 0 ή σταθερά 1 για κάθε είσοδο) με θετική πιθανότητα. Σε αυτή την περίπτωση το $PARITY$ σίγουρα δεν ανήκει στο AC^0 καθώς εκτός

κι αν σταθεροποιήσουμε όλες τις εισόδους, το προκύπτον κύκλωμα προκύπτει σταθερό με μηδενική πιθανότητα, εφόσον προφανώς αρκεί η αλλαγή ενός δυφίου για να αλλάξει το αποτέλεσμα αυτής της συνάρτησης. Θα χρειαστούμε το ακόλουθο Λήμμα, του οποίου η απόδειξη παρατίθεται στο Παράρτημα Α΄.2.

Λήμμα 3.5.1 ([Has86]). *Έστω f μια λογική συνάρτηση n μεταβλητών, η οποία εκφράζεται ως μία $k-DNF$. Τότε αν εφαρμόσουμε μια τυχαία σταθεροποίηση σε $(1-q)*n$ τυχαία επιλεγμένα δυφία της εισόδου, η πιθανότητα να μη μπορεί να γραφτεί η προκύπτουσα συνάρτηση ως $s-CNF$ (για $s \geq 2$) είναι φραγμένη από το $(q*k^C)^{s/2}$ (όπου C μία αρκετά μεγάλη σταθερά).*

Να σημειώσουμε ότι το παραπάνω λήμμα ισχύει και για τη δυϊκή του πλευρά (δηλαδή ισχύει κι αν εναλλάξουμε τα DNF με CNF, απλώς αντί για f παίρνοντας το συμπλήρωμα της).

Σημείωση. Για το υπόλοιπο της απόδειξης θεωρούμε ότι τα AC^0 κυκλώματα μας είναι σε μορφή δέντρου, δηλαδή έχουμε εξάριθμο ίσο με 1 (μπορούμε να το κάνουμε απλά επαναλαμβάνοντας το υποκύκλωμα όσες φορές είναι και ο εξάριθμος – λόγω του σταθερού βάθους στο τέλος θα έχουμε ένα πολυωνυμικό κύκλωμα ίδιου βάθους), ότι δεν έχουμε πύλες NOT αλλά αντ΄ αυτού δίνονται στην είσοδο μαζί με τα δυφία της εισόδου και τα συμπληρώματα τους (μπορούμε να το κάνουμε αν σπρώξουμε τις πύλες NOT προς τα κάτω (εναλλάσσοντας το είδος των πυλών μεταξύ AND και OR ή διπλασιάζοντας υποκυκλώματα αν χρειάζεται όμοια με προηγουμένως)) καθώς και ότι κάθε στρώμα έχει τον ίδιο τύπο πυλών και παίρνει εισόδους μόνο από το αμέσως πιο κάτω στρώμα, όπως και ότι το είδος εναλλάσσεται ανά στρώμα μεταξύ AND και OR (προσθέτοντας πλεονάζουσες πύλες πολυωνυμικού πλήθους και αυξάνοντας το βάθος κατά μία σταθερά)· επίσης στο πιο χαμηλό επίπεδο (των εισόδων) έχουμε χ.β.τ.γ. πύλες AND εισάριθμου 1. Όλες οι παραπάνω μετατροπές μπορούν να γίνουν χωρίς να χάσει το κύκλωμα την AC^0 ιδιότητα του, οπότε πλέον θεωρούμε ότι το κύκλωμα έχει μέγεθος n^b και βάθος d διατηρώντας τις παραπάνω ιδιότητες.

Η γενική ιδέα, είναι ότι σε κάθε στρώμα θα αφήνουμε ελεύθερο ένα μικρό ποσοστό των εισόδων, της τάξης του n^c (με $c < 1$) κι άρα θα έχουμε $q_i = n_i/n_{i-1}$, όπου $n_0 = n$ και $n_i = n^{c^i}$, άρα $q_i = n^{-(1-c)c^{i-1}}$. Για $c = 1/2$ έχουμε $n_i = n^{2^{-i}}$ και $q_i = n^{-2^{-i}}$. Έχουμε μία τελευταία ελεύθερη μεταβλητή, η οποία είναι το k_i που συμβολίζει το τι μεγέθους θέλουμε να είναι οι CNF (ή DNF) στο i-οστό επίπεδο. Από τις υποθέσεις έχουμε $k_0 = 1$. Θα θέσουμε λοιπόν $k_i = Cb2^i$ για λόγους που θα φανούν από τις πράξεις στη συνέχεια.
Έχουμε σε κάποιο βήμα λοιπόν ότι στο δεύτερο στρώμα υπάρχουν συναρτήσεις OR και στο πρώτο AND εισάριθμου k_i, άρα κάθε OR του δεύτερου στρώματος εκφράζεται ως μία $k_i - DNF$ κι άρα από το Λήμμα 3.5.1 έχουμε ότι έχει

πιθανότητα τουλάχιστον

$$\begin{aligned}&1-(q_{i+1}*k_i^C)^{k_{i+1}/2}=\\=&1-\left(n^{-2^{-(i+1)}}*(Cb2^i)^C\right)^{Cb2^{i+1}/2}=\\=&1-n^{-Cb/2}*(Cb2^i)^{C^2b2^i}\end{aligned}$$

να μπορεί να εκφραστεί ως μία $k_{i+1}-CNF$. Επειδή τώρα έχουμε το πολύ n^b πύλες σε αυτό το στρώμα προκύπτει ότι υπάρχει πιθανότητα τουλάχιστον

$$\begin{aligned}&1-n^b*n^{-Cb/2}*(Cb2^i)^{C^2b2^i}=\\=&1-(Cb2^i)^{C^2b2^i}/n^{b(C/2-1)}\geq\\\geq&1-(Cb2^d)^{C^2b2^d}/n^{b(C/2-1)}\geq\\\geq&1-Dn^{-b}\end{aligned}$$

για κάποια σταθερά D και αρκετά μεγάλο n. Αφού λοιπόν σε κάθε στρώμα υπάρχει τουλάχιστον τόση πιθανότητα να μπορούμε να εκφράσουμε κάθε k_i-DNF ως μία $k_{i+1}-CNF$ προκύπτει ότι συμπτύσσοντας την νέα πύλη OR του δεύτερου στρώματος με αυτές του τρίτου, ότι το συνολικό βάθος του κυκλώματος πέφτει σε κάθε βήμα κατά 1 και ότι ο νέος εισάριθμος του κυκλώματος αυξάνεται σε k_{i+1}. Επομένως παίρνοντας το γινόμενο, προκύπτει ότι υπάρχει μία θετική πιθανότητα να μπορεί μετά από $d-2$ βήματα το κύκλωμα να γραφτεί ως μία $k_{d-2}-C(D)NF$. Επειδή ωστόσο $k_{d-2}<n$ μία τέτοια $C(D)NF$ μπορεί πολύ εύκολα να σταθεροποιηθεί (π.χ. αν είναι CNF αρκεί να θέσουμε ίσο με 0 τους k_{d-2} όρους του πρώτου ελαχιστόρου, ώστε να γίνει 0 ολόκληρη η συνάρτηση). Από την άλλη αυτό αποκλείεται να μπορεί να ισχύει για την $PARITY$ για τους λόγους που προαναφέραμε κι άρα καταλήγουμε ότι $PARITY\notin AC^0$. ■

Το παραπάνω αρκεί για να γνωρίζουμε πλέον ότι $P\nsubseteq AC^0$, αλλά στην πραγματικότητα μπορούμε να δείξουμε κι ότι $AC^0\subsetneq TC^0$ παρουσιάζοντας ένα απλό κύκλωμα στο TC^0 που ελέγχει αν το πλήθος των 1 στην είσοδο είναι $0,2,4,...$ (με χρήση κατάλληλων MAJ πυλών) και παίρνοντας το OR όλων αυτών (προφανώς πολυωνυμικού μεγέθους και σταθερού βάθους).

3.4.2 ACC^0

Από ό,τι φάνηκε η απλή γλώσσα $PARITY=\{x|\sum x_i=0\ (mod\ 2)\}$ ήταν αρκετή για να «ρίξει» την κλάση AC^0 από υποψήφια να περιέχει μη τετριμμένες κλάσεις πολυπλοκότητας. Προτού πάει κανείς στη TC^0 είναι εύλογο να αναρωτηθεί κατά πόσο ήταν χαρακτηριστική η συγκεκριμένη συνάρτηση, δηλαδή κατά πόσο μια πύλη της μορφής mod_2 είναι αρκετή ώστε αν την δίναμε στην κλάση AC^0 αυτή να γινόταν υποψήφια ώστε να μπορούσε να υπολογίσει συναρτήσεις του TC^0 ή του P,NP κ.ο.κ. Ορίζουμε λοιπόν την ακόλουθη γενικότερη κλάση κυκλωμάτων:

Ορισμός 3.4.1. Η κλάση $ACC^d(p_1, p_2, ..., p_s)$ ορίζεται ακριβώς όπως η AC^d επεκτείνοντας όμως την βάση B προσθέτοντας τις πύλες mod_{p_1}, mod_{p_2}, ..., $modp_s$ (όπου $mod_p(x_1, x_2, ..., x_n) = 0$ αν και μόνο αν το άθροισμα των x_i είναι $0\ (mod\ p)$).

Ορισμός 3.4.2. Η κλάση ACC^d ορίζεται ως η ένωση όλων των κλάσεων $ACC^d(p_1, p_2, ..., p_s)$ για όλα τα πιθανά πεπερασμένα σύνολα $s \in \mathbb{N}$ φυσικών αριθμών.

Σημείωση. Ισχύει $AC^d \subset ACC^d$ και βάσει του Θεωρήματος 3.5 είναι σίγουρα $AC^0 \subsetneq ACC^0(2)$ (και με παρόμοιο τρόπο μπορούμε άλλωστε να δείξουμε ότι $AC^0 \subsetneq ACC^0(p)$ για κάθε p).
Επίσης ισχύει $ACC^d \subset TC^d$. Αυτό προκύπτει αν αντικαταστήσουμε κάθε mod_p πύλη με κατάλληλες MAJ πύλες που ελέγχουν αν το άθροισμα των άσσων είναι 0, p, $2p$,..., $\lfloor n/p \rfloor * p$ και πάρουμε το OR όλων αυτών (θα χρειαστούν το πολύ $n/2$ τέτοιες πύλες οι οποίες τοποθετούνται παράλληλα κι άρα το συνολικό βάθος το πολύ να διπλασιαστεί, οπότε προκύπτει TC^d κύκλωμα).

Ήδη το γεγονός ότι ορίσαμε την κλάση ACC με ποικιλία πυλών mod μας προδιαθέτει για το γεγονός ότι το να εισάγουμε μόνο την πύλη mod_2 δεν είναι αρκετό. Πράγματι έχουμε το ακόλουθο Θεώρημα, του οποίου η απόδειξη παραλείπεται:

Θεώρημα 3.6 ([Raz87, Smo87])**.** *Για δύο διαφορετικούς πρώτους p, q η συνάρτηση MOD_p δεν ανήκει στο $ACC^0(q)$.*

Το επόμενο βήμα είναι να δούμε την κατάσταση για το $ACC^0(c)$ όπου c κάποιος σύνθετος. Γενικά αν $c = p_1^{a_1} * ... * p_s^{a_s}$ η ανάλυση του σε γινόμενο πρώτων παραγόντων, τότε έχουμε ότι $ACC^0(c) = ACC^0(p_1^{a_1}, ..., p_s^{a_s})$.
Πράγματι, έχουμε

$$mod_c(x_1, ..., x_n) = OR\left(mod_{p_1^{a_1}}(x_1, ..., x_n), ..., mod_{p_s^{a_s}}(x_1, ..., x_n)\right)$$

και αντίστροφα

$$mod_{p_i^{a_i}}(x_1, ..., x_n) = mod_c(x_1, ..., x_1, x_2, ..., x_2, \ ... \ , x_n, ..., x_n)$$

όπου κάθε μεταβλητή επαναλαμβάνεται $c/p_i^{a_i}$ φορές στην mod_c. Εν τούτοις δεν γνωρίζουμε ποια είναι η σχέση μεταξύ $ACC^0(p^a)$ και $ACC^0(p^b)$ (με $a < b$). Προφανώς ισχύει $ACC^0(p^a) \subseteq ACC^0(p^b)$ με μια κατασκευή όμοια με παραπάνω, αλλά δεν φαίνεται να υπάρχει κάποιος προφανής τρόπος για την αντίστροφη κατεύθυνση και για την ακρίβεια δεν ξέρουμε καν αν το mod_4 μπορεί να γραφεί με ένα $ACC^0(2)$ κύκλωμα.

Δυστυχώς η έλλειψη προόδου παρατηρείται ακόμη κι όταν επιτρέπουμε ACC^0 κυκλώματα με τον ελάχιστο σύνθετο που περιέχει δύο (διαφορετικούς) πρώτους, τον 6. Με την τωρινή γνώση θα μπορούσε π.χ. ολόκληρο το NP

να ανήκει στο $ACC^0(6)$ (βλ. και [BBR94]). Για την ακρίβεια μέχρι και πριν λίγα χρόνια δεν γνωρίζαμε καν τη σχέση μεταξύ $NEXP$ και $ACC^0(6)$, μέχρι που αποδείχθηκε επιτέλους από τον Ryan Williams ότι $NEXP \nsubseteq ACC^0$, την οποία απόδειξη θα μελετήσουμε στα κεφάλαια που ακολουθούν.

3.4.3 Εκφραστικότητα της ACC^0

Μία γενική κατεύθυνση στην απόδειξη κάτω φραγμάτων στην θεωρία κυκλωματικής πολυπλοκότητας είναι να βρίσκουμε μια ιδιότητα που έχει κάθε κύκλωμα μιας συγκεκριμένης κλάσης κυκλωμάτων και στη συνέχεια να αποδεικνύουμε ότι μια συνάρτηση που ανήκει σε μια άλλη κλάση δεν έχει αυτή την ιδιότητα, ξεχωρίζοντας έτσι τις δύο κλάσεις. Αυτή ήταν ακριβώς και η μέθοδος που ακολουθήθηκε για να δειχθεί ότι $AC^0 \subsetneq ACC^0$. Όπως θα δούμε αργότερα, αυτή η μέθοδος είναι στενά συνδεδεμένη με μια γενικότερη κατηγορία αποδείξεων, τις επονομαζόμενες φυσικές αποδείξεις, οι οποίες έχουν αυτή την ονομασία ακριβώς λόγω του διαισθητικά φυσικού τρόπου διαχωρισμού δύο κλάσεων εκμεταλλευόμενοι μια χαρακτηριστική ιδιότητα περιορισμένης εκφραστικότητας της «χαμηλής» κλάσης.

Όπως είπαμε, μέχρι και σήμερα η κατάσταση είναι μάλλον αποκαρδιωτική όσον αφορά τη πρόοδο που έχουμε σχετικά με την ακριβή εκφραστικότητα της κλάσης ACC^0 (αν αναλογιστεί μάλιστα κανείς ότι επικρατεί η γενική «πίστη» μεταξύ των ερευνητών ότι $MAJ \notin ACC^0$ και ότι αυτή τη στιγμή δε γνωρίζουμε καν τη ακριβή σχέση του ACC^0 με το EXP ή με το $P/poly$). Κάθε χαρακτηριστική ιδιότητα των ACC^0 κυκλωμάτων, λοιπόν, είναι πολύ μεγάλης σημασίας για το τρέχον ερευνητικό τοπίο.

Μία από τις πιο ενδελεχώς μελετημένες ιδιότητες κάθε οικογένειας κυκλωμάτων είναι αυτή που είναι εμπνευσμένη από το θεμελιώδες ερώτημα της θεωρίας πολυπλοκότητας και δεν είναι άλλο από το $CKT - SAT$ που ορίζεται με τον προφανή τρόπο:

Ορισμός 3.4.3. *(CKT-SAT)* Ως $CKT - SAT$ ορίζεται η γλώσσα $\{C|$το C αναπαριστά κύκλωμα το οποίο έχει έξοδο 1 για κάποια είσοδο$\}$. Ανάλογο ορισμό έχουμε όταν περιορίζουμε το C να ανήκει σε κάποια συγκεκριμένη κλάση κυκλωμάτων.

Πρόταση 3.4.1. *Το $CKT-SAT$ (στη γενική του μορφή) είναι NP-πλήρες.*

Απόδειξη. Πράγματι, είναι προφανώς στο NP αφού δοθέντος μιας τιμής που το κύκλωμα γίνεται 1, μπορούμε σε πολυωνυμικό χρόνο να προσομοιώσουμε το κύκλωμα για αυτή την είσοδο και να την επαληθεύσουμε. Επίσης είναι προφανώς NP-πλήρες, αφού κάθε $3 - SAT$ στιγμιότυπο αντιστοιχεί σε μία AND κάποιων OR όρων, οπότε ορίζουν ένα προφανές ισοδύναμο με τη $3-SAT$ κύκλωμα (που προφανώς παράγεται σε πολυωνυμικό χρόνο). ■

Αναμένουμε λοιπόν ότι μία οικογένεια κυκλωμάτων που είναι αρκετά εκφραστική ώστε να περιέχει το P θα ορίζει ένα $CKT - SAT$ που θα είναι

πολύ δύσκολο (σε συμβαδισμό με την SETH υπόθεση, ότι $NP \nsubseteq subEXP$). Επομένως ένα αποτέλεσμα που βελτιώνει το $CKT - SAT$ μιας κλάσης από την τετριμμένη εκθετική λύση ως προς οποιαδήποτε παράμετρο, είναι ένα αποτέλεσμα που μάλλον δεν θα πρέπει να ληφθεί απαρατήρητο και ενδεχομένως να οδηγεί σε ένα καινούριο αποτέλεσμα διαχωρισμού.

Η κλάση ACC^0 παρά τη μέχρι στιγμής μυστηριώδη της φύση, ανήκει ευτυχώς σε αυτή την κατηγορία κλάσεων που επιδέχονται μία (ελαφριά) βελτιστοποίηση από την τετριμμένη επίλυση του $CKT - SAT$. Συγκεκριμένα έχουμε το ακόλουθο θεώρημα:

Θεώρημα 3.7 ([Wil11])**.** *Για κάθε d, υπάρχει ένα $\delta > 0$ τέτοιο ώστε το $CKT - SAT$ για κυκλώματα ACC^0 βάθους d, n εισόδων και 2^{n^δ} μεγέθους να ανήκει στο $DTIME[2^{n-n^\delta}]$.*

Σημείωση. Εμείς ορίσαμε γενικά την κλάση ACC να είναι πολυωνυμικού μεγέθους, αλλά μπορούμε να την επεκτείνουμε με το φυσικό τρόπο να έχει οποιοδήποτε μέγεθος και πάντα σταθερό βάθος.

Απόδειξη. Για την απόδειξη θα χρειαστούμε τον ορισμό του $SYM+$ κυκλώματος καθώς και ένα βασικό Λήμμα που αφορά αυτά τα κυκλώματα.

Ορισμός 3.4.4. *($SYM+$ κυκλώματα)* Ορίζουμε ως $SYM+$ κύκλωμα βάθους d και μεγέθους s μία δυάδα (P, Θ) τέτοια ώστε το P να αντιστοιχεί σε ένα πολυώνυμο n ακέραιων μεταβλητών βαθμού d και ακέραιων συντελεστών μικρότερων ή ίσων του s και το Θ να είναι μία συνάρτηση από τους ακεραίους στο $\{0, 1\}$.
Λέμε ότι ένα $SYM+$ κύκλωμα (P, Θ) υπολογίζει μια λογική συνάρτηση f n μεταβλητών (ή είναι ισοδύναμη με αυτήν) αν $f(x) = \Theta(P(x_1, ..., x_n))$ για κάθε $x = x_1...x_n \in \{0, 1\}^n$ (δηλαδή η Θ έχει το ρόλο της μετατροπής του ακέραιου πολυώνυμου σε μια λογική συνάρτηση θεωρώντας ως 1 ορισμένους από τους ακέραιους που μπορεί να βγάλει το πολυώνυμο P). Η απεικόνιση του P γίνεται ως η λίστα των αντίστοιχων μη μηδενικών συντελεστών του πολυωνύμου και η απεικόνιση του Θ ως η λίστα των ακεραίων στους οποίους επιστρέφει 1 (για συναρτήσεις n μεταβλητών, η μέγιστη τιμή που μπορεί να πάρει το $P(x)$ είναι $s * (n + 1)^d$ (αφού μπορούν να σχηματιστούν το πολύ $(n + 1)^d$ όροι βαθμού d με n μεταβλητές και κάθε ένας μπορεί να πάρει την τιμή 0 ή 1) κι άρα η Θ αρκεί να έχει μέγεθος $O(s * n^d)$).

Η σημασία των παραπάνω κυκλωμάτων φαίνεται από το ακόλουθο κρίσιμο Λήμμα την απόδειξη του οποίου παραθέτουμε στο Παράρτημα Α΄.3.

Λήμμα 3.7.1 ([Yao90, BT94, AG91])**.** *Δοθέντος ενός κυκλώματος C βάθους d, n εισόδων και μεγέθους S, υπάρχει αλγόριθμος χρόνου $S^{log^D S}$ που επιστρέφει ισοδύναμο $SYM+$ κύκλωμα βαθμού $log^D S$ και μεγέθους $S^{log^D S}$ (για κάποια σταθερά D που εξαρτάται από το d, δηλαδή $D = D(d)$).*

Έστω λοιπόν ένα βάθος d. Ορίζουμε τότε ως $\delta = \delta(d) \leq \frac{1}{3(1+D(d+1))}$ (όπου D η συνάρτηση του παραπάνω Λήμματος) και θεωρούμε ότι έχουμε ένα ACC^0 κύκλωμα C βάθους d, n εισόδων και μεγέθους $S \leq 2^{n^\delta}$. Ζητούμενο είναι να ελέγξουμε αν είναι ικανοποιήσιμο, δηλαδή αν έχει είσοδο για την οποία δίνει έξοδο 1. Περιορίζουμε το κύκλωμα C σε ένα κύκλωμα C' με $n - 2n^\delta$ εισόδους το οποίο φτιάχνουμε παραθέτοντας 2^{2n^δ} αντίτυπα του C όπου σε κάθε ένα δίνουμε ως είσοδο τις $n - 2n^\delta$ εισόδους του C' και μία εκ των 2^{2n^δ} δυνατών τιμών των υπόλοιπων δυφίων σε έκαστο εκ των 2^{2n^δ} αντιτύπων του C κι εν τέλει παίρνουμε το OR όλων αυτών. Το C' είναι προφανώς ένα ACC^0 κύκλωμα μεγέθους $S' = 2^{2n^\delta} * S + c \leq 2^{3n^\delta}$ με βάθος $d + 1$ και $n - 2n^\delta$ εισόδους, το οποίο είναι ικανοποιήσιμο αν και μόνο αν είναι ικανοποιήσιμο και το C. Η τετριμμένη προσέγγιση θα ήταν να δοκιμάσουμε όλες τις δυνατές 2^{n-2n^δ} εισόδους οδηγώντας σε συνολικό χρόνο 2^{n+n^δ}, οπότε θα ήταν μάταιη η μέχρι τώρα διαδικασία.

Κάνοντας όμως χρήση του Λήμματος 3.7.1 μπορούμε σε χρόνο

$$S'^{log^{D(d+1)}S'} \leq 2^{(3n^\delta)^{(1+D(d+1))}} \leq 2^{(3n)^{1/3}} < 2^{\sqrt{n}}$$

να βρούμε ένα $SYM+$ ισοδύναμο του C' κύκλωμα, μεγέθους το πολύ $2^{\sqrt{n}}$, βαθμού $(3n^\delta)^{D(d+1)} < \sqrt{n}$ και συνολικά $n - 2n^\delta$ εισόδων. Αρκεί να βρούμε έναν αποδοτικό τρόπο να υπολογίσουμε όλες τις τιμές που βγάζει το αντίστοιχο πολυώνυμο κι έχουμε τελειώσει.

Έστω λοιπόν ένα πολυώνυμο με αντίστοιχους συντελεστές p_y, δηλαδή

$$P(x_1, ..., x_n) = \sum_{y \in \{0,1\}^n} p_y * \prod_{i=1}^{n} x_i^{y_i}$$

(γενικά δεν δίνουμε σε κάποια μεταβλητή δύναμη μεγαλύτερη του 2, αφού $x^c = x$ για $x \in \{0, 1\}$). Θέλουμε να φτιάξουμε τη λίστα που περιέχει τις 2^n τιμές του P. Θα δείξουμε με επαγωγή ότι μπορούμε να το κάνουμε σε χρόνο $2^n poly(n, logm)$, όπου $m = max|P|$. Έστω ότι ισχύει μέχρι $n - 1$. Έχουμε όμως ότι

$$P(x_1, ..., x_{n-1}, 0) = \sum_{y \in \{0,1\}^{n-1}} p_{y0} * \prod_{i=1}^{n-1} x_i^{y_i}$$

και

$$P(x_1, ..., x_{n-1}, 1) = \sum_{y \in \{0,1\}^{n-1}} p_{y0} * \prod_{i=1}^{n-1} x_i^{y_i} + \sum_{y \in \{0,1\}^{n-1}} p_{y1} * \prod_{i=1}^{n-1} x_i^{y_i}$$

και ότι οι λίστες των δύο πολυωνύμων που εμφανίζονται παραπάνω, μπορούν να κατασκευαστούν έκαστη σε χρόνο $2^{n-1} * (n - 1 + logs + c)^c$. Τότε η

τιμή του τελικού πολυωνύμου για τις μισές τιμές $x_1...x_{n-1}0$ είναι ήδη έτοιμη, ενώ για τις υπόλοιπες αρκεί να προσθέσουμε τα αντίστοιχα στοιχεία των δύο λιστών (όπου κάθε πρόσθεση χρειάζεται $O(logs)$ χρόνο). Συνολικά απαιτούνται $2*2^{n-1}poly(n,logm)+2^n*poly(n,logm) = 2^n poly(n,logm)$ (αποφύγαμε την πλήρη απεικόνιση των συντελεστών του $poly(n,logm)$ για λόγους απλότητας, αλλά είναι προφανές ότι επειδή έχουμε το πολύ n βάθος επαναλήψεων, η τελική έκφραση στο επίπεδο του n είναι πράγματι κάποιο φραγμένο πολυώνυμο).

Επομένως μπορούμε να υπολογίσουμε όλες τις τιμές του τελικού $SYM+$ κυκλώματος σε χρόνο $O(2^{n-2n^\delta})poly(n)$ και στη συνέχεια κάνοντας έναν έλεγχο στο $o(2^{n-n^\delta})$ μεγέθους Θ να βρούμε αν είναι ικανοποιήσιμο το αρχικό κύκλωμα C. Αθροίζοντας, χρειαστήκαμε χρόνο $O(2^{3n^\delta})$ για τη δημιουργία του C', στη συνέχεια χρειαστήκαμε χρόνο $O(2^{\sqrt{n}})$ για τη μετατροπή του σε $SYM+$ κύκλωμα και εν τέλει χρειαστήκαμε χρόνο $O(2^{n-2n^\delta}poly(n))$ για την παραγωγή όλων των τιμών του. Συνολικά χρειαστήκαμε $o(2^{n-n^\delta})$ χρόνο και η απόδειξη ολοκληρώθηκε. ■

Ασφαλώς ο τελικός αλγόριθμος παραμένει απρόσιτος, αλλά δεν παύει να υπονοεί μια αδυναμία των ACC^0 κυκλωμάτων να υπολογίσουν απόλυτα δύσκολες συναρτήσεις, με την έννοια των συναρτήσεων που ως προς το $CKT-SAT$ συμπεριφέρονται ως μαύρα κουτιά και όπου ο βέλτιστος αλγόριθμος είναι η δοκιμή όλων των πιθανών τιμών (η οποία είναι η τρέχουσα εικόνα που έχουμε για τις μεγαλύτερες κλάσεις κυκλωμάτων). Έστω και αυτή η μικρή βελτίωση, όμως, θα σταθεί αρκετή (αλλά και κρίσιμη) ώστε να διαχωρίσουμε τη κλάση $NEXP$ από την ACC^0.

Κεφάλαιο 4

Συσχετίσεις Υπολογιστών και Κυκλωμάτων

Μέχρι στιγμής μελετήσαμε τους ορισμούς και μερικά σημαντικά αποτελέσματα δύο φύσει διαφορετικών μοντέλων υπολογισμού, του υπολογισμού μέσω μηχανών Turing (ο οποίος αντιστοιχεί στο καθιερωμένο μοντέλο καθολικής υπολογιστικής ισχύος σύμφωνα με τη θέση Church-Turing) και του υπολογισμού μέσω οικογενειών κυκλωμάτων (ο οποίος στη μη-ομοιόμορφη έκδοση του αντιστοιχεί σε μη ρεαλιστικό μοντέλο για τους λόγους που έχουμε αναφέρει). Παρά τη διαφορετική τους φύση ωστόσο, αποτελέσματα στον έναν τομέα συνάγουν σημαντικά αποτελέσματα στον άλλον. Κατά κάποιον τρόπο, είδαμε ήδη ένα τέτοιο αποτέλεσμα στο προηγούμενο κεφάλαιο, αφού π.χ. διαχωρίσαμε τη κλάση $P_{/poly}$ από την AC^0 επειδή δείξαμε την ύπαρξη μιας γλώσσας που επιδέχεται πολυωνυμικό αλγόριθμο απόφασης (που άρα ανήκει στο $P_{/poly}$), η οποία ωστόσο δεν ανήκει στο AC_0 (ακόμη και σε σχέση με την TC^0, ο τρόπος απόδειξης ύπαρξης των αντίστοιχων κυκλωμάτων ήταν κατασκευαστικός – δηλαδή προέκυψε από την ύπαρξη ενός σχετικού αλγορίθμου). Σε αυτό το κεφάλαιο λοιπόν θα μελετήσουμε μερικές από τις πιο χαρακτηριστικές συνδέσεις του ενός πεδίου με το άλλο και θα λήξουμε με μία σύνδεση, η οποία για χρόνια έπαυσε τις ελπίδες για μια (εύκολη) απόδειξη ότι $NP \nsubseteq P_{/poly}$.

4.1 Συνέπειες Κυκλωματικών Υποθέσεων

4.1.1 Περίπτωση ύπαρξης εύκολων κυκλωμάτων

Είδαμε ότι $P \subseteq P_{/poly}$, αλλά ότι το $P_{/poly}$ περιέχει γλώσσες πολύ πιο δύσκολες από κάθε πολυωνυμική μηχανή. Μια εύλογη ερώτηση είναι λοιπόν κατά πόσο ισχύει το ίδιο για πιο ισχυρές γλώσσες, π.χ. αν ισχύει ότι $NP \subseteq P_{/poly}$. Γενικά πιστεύεται ότι μια πολυωνυμική πληροφορία μόνο για το μήκος εισόδου δεν είναι αρκετή ώστε να υπολογίσουμε σε πολυωνυμικό χρόνο προβλήματα που παίρνουν υπερπολυωνυμικό χρόνο (άρα ούτε NP-πλήρη προβλήματα για τα

οποία πιστεύουμε ότι ανήκουν σε αυτή την κατηγορία). Δυστυχώς, το μόνο που έχουμε καταφέρει να αποδείξουμε μέχρι στιγμής είναι ότι για τα NP-πλήρη προβλήματα χρειάζονται κυκλώματα (κάποιας βάσης) μεγέθους τουλάχιστον $5n$ [IM02]. Θα μπορούσε επομένως να υπάρχει γραμμική οικογένεια κυκλωμάτων που περιέχει όλο το NP (η ίδια υπόθεση για το P είναι μάλιστα γνωστή ως εικασία του Kolmogorov και γενικά πιστεύεται ότι δεν θα ισχύει [Lip94]).

Εν τούτοις είμαστε σε θέση να βγάλουμε συμπεράσματα που προκύπτουν από τέτοιες υποθέσεις και που αφορούν μόνο υπολογιστικές κλάσεις, οπότε αρνητικά αποτελέσματα σε αυτές θα συνάγουν αρνητικά αποτελέσματα και στις κυκλωματικές κλάσεις.

Θεώρημα 4.1 ([KL80]). *Αν $NP \subseteq P_{/poly}$, τότε $PH = \Sigma_2^P$.*

Απόδειξη. Έστω ότι $NP \subseteq P_{/poly}$. Τότε υπάρχει μια πολυωνυμική οικογένεια κυκλωμάτων n εξόδων που με είσοδο ένα $3-SAT$ (χ.β.τ.γ. μήκους n και το πολύ n μεταβλητών) στιγμιότυπο, επιστρέφει μία απονομή αληθείας για το $3-SAT$ αν είναι ικανοποιήσιμο, αλλιώς μια οποιαδήποτε τιμή. Για να δείξουμε ότι $PH = \Sigma_2^P$ αρκεί να δείξουμε ότι $\Sigma_3^P = \Sigma_2^P$ [Pap94, p.424-429]. Μία Σ_3^P γλώσσα L, όμως, χαρακτηρίζεται από μία μηχανή M πολυωνυμικού χρόνου, για την οποία $x \in L \Leftrightarrow \exists y \forall z \exists w M(x,y,z,w) = 1$ όπου το μήκος των μεταβλητών των ποσοδεικτών είναι φραγμένο από μία σταθερή δύναμη του μήκους εισόδου. Δεδομένων των x, y, z το $\exists w M(x,y,z,w) = 1$ είναι ένα NP πρόβλημα, οπότε υπάρχει πολυωνυμικό κύκλωμα C το οποίο για κάθε x, y, z επιστρέφει το w για το οποίο $M(x,y,z,w) = 1$ (αν υπάρχει τέτοιο).

Επομένως έχουμε $x \in L \Leftrightarrow \exists \langle y, C \rangle \forall z M(x,y,z,C(x,y,z)) = 1$ (η αντίστροφη κατεύθυνση είναι προφανές ότι ισχύει) και επειδή όλοι οι ποσοδείκτες είναι πολυωνυμικά φραγμένοι, προκύπτει ότι υπάρχει μια μηχανή M' που τρέχει συνολικά πολυωνυμικό χρόνο και για την οποία έχουμε ότι ισχύει $x \in L \Leftrightarrow \exists y' = \langle y, C \rangle \forall z M'(x,y',z) = 1$ όπου η M' απλά προσομοιώνει το $M(x,y,z,C(x,y,z))$ κι άρα πράγματι $L \in \Sigma_2^P$. ■

Η παραπάνω, δηλαδή ότι $NP \subseteq P_{/poly}$, είναι η αμέσως επόμενη λιγότερο «ακραία» υπόθεση από το ότι $NP = P$, που σημαίνει ότι μπορεί το NP να μην είναι εφικτά επιλύσιμο, αλλά υπάρχουν προγράμματα (που προσομοιώνουν τα αντίστοιχα κυκλώματα) που πετυχαίνουν την επίλυση $3-SAT$ στιγμιοτύπων μέχρι κάποιου πολυωνυμικού μεγέθους (ως προς το μήκος του προγράμματος). Ακόμη και μια τέτοια υπόθεση όμως, όπως φαίνεται παραπάνω, οδηγεί στην κατάρρευση της πολυωνυμικής ιεραρχίας στο δεύτερο επίπεδο (που επίσης δεν θεωρείται πιθανό επειδή θα είχε επιπτώσεις παρόμοιες με αυτές που θα είχε το $P = NP$).

Είναι προφανές ότι η ισχύς της παραπάνω πρότασης επεκτείνεται αν αντί για NP πάρουμε οποιοδήποτε ψηλότερο επίπεδο της πολυωνυμικής ιεραρχίας. Προκύπτει λοιπόν το ερώτημα τι συμβαίνει με υψηλότερες κλάσεις. Για αρχή, έχουμε το ακόλουθο παρόμοιο Θεώρημα:

Θεώρημα 4.2 ([KL80]). *Αν $EXP \subseteq P_{/poly}$, τότε $EXP = \Sigma_2^P$.*

Απόδειξη. Για κάθε γλώσσα $L \in DTIME[2^{n^c}]$ (με αντίστοιχη μηχανή M) έχουμε ότι υπάρχει ένα πρόγραμμα M' χρόνου $O(2^{n^d})$ (με $d > c$) το οποίο για είσοδο $\langle x, i, j\rangle$ με $|i|, |j| \leq O(|x|^c)$, επιστρέφει το j-οστό στοιχείο της διαμόρφωσης της M' στο i-οστό βήμα (η M' απλά προσομοιώνει τη M με είσοδο x για $i < 2^{n^c}$ βήματα και επιστρέφει το j-οστό στοιχείο της i-οστής διαμόρφωσης, όπου π.χ. αν $j \leq 2 * 2^{n^c}$ τότε το στοιχείο ισούται με το αντίστοιχο κελί της μηχανής (όπου η κεφαλή βρίσκεται πάντα στο $j = 2^{n^c}$) και για μεγαλύτερο j (το πολύ άλλα $log|Q|$ δυφία) επιστρέφεται η τρέχουσα κατάσταση της μηχανής).

Όμως, από την υπόθεση, προκύπτει ότι υπάρχει ένα $P_{/poly}$ κύκλωμα C που επιστρέφει αυτά τα στοιχεία. Επιπλέον υπάρχει και τρόπος να εξακριβώσουμε την ορθότητα αυτής της μηχανής. Για βήμα 0, αρκεί να εξακριβώσουμε ότι για κάθε j η διαμόρφωση της M είναι αυτή της αρχικής κατάστασης (επαληθεύσιμο σε γραμμικό χρόνο για κάθε j). Για τα επόμενα βήματα, αρκεί να επαληθεύσουμε ότι για κάθε j το στοιχείο στην αντίστοιχη θέση συμβαδίζει με την παλιά διαμόρφωση αφότου έχει εφαρμοστεί το βήμα της μηχανής που αναμένουμε (μετατόπιση κατά μία θέση ή αλλαγή του δυφίου της κεφαλής – πάντα σύμφωνα με τους κανόνες της M). Αυτό για κάθε j μπορούμε επίσης να το κάνουμε σε πολυωνυμικό χρόνο, αφού χρειαζόμαστε το πολύ δύο ερωτήσεις στο C μία για το τρέχον j-οστό στοιχείο του i-οστού βήματος και μία για το $j + s$-οστό στοιχείο του $i - 1$-οστού βήματος (όπου $s \in \{-1, 0, 1\}$ ανάλογα με την τρέχουσα εναλλαγή που πρέπει να γίνει και που μπορούμε να ξέρουμε εύκολα από την $(i-1)$-οστή διαμόρφωση). Τέλος, αφότου επαληθεύσουμε και το τελευταίο βήμα, αρκεί να εξακριβώσουμε αν η κατάσταση που βρίσκεται είναι κατάσταση αποδοχής (που γίνεται σε σταθερό χρόνο).

Όλη η παραπάνω διαδικασία, λοιπόν, μαντεύει για μια είσοδο x το πολυωνυμικό κύκλωμα που περιγράφει τη διαμόρφωση της μηχανής M σε κάθε βήμα και εξακριβώνει για κάθε $\langle i, j\rangle$ ότι ήταν το σωστό κύκλωμα (δηλαδή δίνει ορθές διαμορφώσεις), οπότε και δίκαια αποφαίνεται στο τέλος αν $x \in L$. Η διαδικασία αυτή, μπορεί να τυποποιηθεί σε μια πολυωνυμική μηχανή M'' για την οποία θα έχουμε εν τέλει $x \in L \Leftrightarrow \exists C \forall \langle i, j\rangle M''(x, C, \langle i, j\rangle)$ όπου τα μήκη των C, $\langle i, j\rangle$ είναι πολυωνυμικά φραγμένα από το μήκος της εισόδου $|x|$. Επομένως πράγματι $L \in \Sigma_2^P$ και η απόδειξη ολοκληρώθηκε. ■

Στο ίδιο πνεύμα μπορούμε να αποδείξουμε επιπλέον το ακόλουθο θεώρημα:

Θεώρημα 4.3 ([BFL91, BFNW93])**.** *Αν $EXP \subseteq P_{/poly}$, τότε $EXP = MA$.*

Απόδειξη. Είδαμε προηγουμένως ότι αν $EXP \subseteq P_{/poly}$, τότε $EXP = \Sigma_2^P$ κι άρα κατ' επέκταση $\Sigma_2^P = PSPACE = IP = EXP \subseteq P_{/poly}$. Επομένως κάθε ντετερμινιστικό πρόβλημα εκθετικού χρόνου αντιστοιχεί σε στιγμιότυπο του $TQBF$ και επιπλέον επιδέχεται διαλογική απόδειξη, στην οποία γνωρίζουμε ότι ο Αποδείκτης είναι πολυωνυμικού χώρου. Όμως τότε, ο Αποδείκτης μπορεί να αντικατασταθεί από ένα πολυωνυμικό κύκλωμα, λόγω της υπόθεσης. Άρα στη διαλογική υπόθεση που λύνει το EXP πρόβλημα, αρκεί να δώσουμε στον

Ελεγκτή όλες τις αντιδράσεις του Αποδείκτη κωδικοποιημένες στο πολυωνυμικό κύκλωμα που αντιστοιχεί στην εκτέλεση του και στη συνέχεια ο Ελεγκτής (χρησιμοποιώντας τυχαία δυφία) προσομοιώνει όλο το διάλογο της διαλογικής απόδειξης χρησιμοποιώντας αυτό το κύκλωμα στη θέση των απαντήσεων του Αποδείκτη. Επομένως αν η συμβολοσειρά εισόδου ανήκει σε αυτή την EXP γλώσσα, τότε θα υπάρχει ένα κύκλωμα (πολυωνυμικού μεγέθους κι άρα το οποίο απαιτεί πολυωνυμικό χρόνο για τον υπολογισμό της εξόδου του για κάθε είσοδο) το οποίο όταν δίνεται στον Ελεγκτή με μεγάλη πιθανότητα αποδέχεται (καθώς απλώς προσομοιώνει την αντίστοιχη διαλογική απόδειξη), ενώ αν η συμβολοσειρά δεν ανήκει, τότε προφανώς δεν υπάρχει τέτοιο κύκλωμα (καθώς αν υπήρχε θα μπορούσε να χρησιμοποιηθεί αυτούσιο για τις απαντήσεις του Αποδείκτη στην κανονική διαλογική απόδειξη, οδηγώντας σε άτοπο). Άρα μετατρέπουμε το EXP πρόβλημα στο MA πρόβλημα όπου ο Αποδείκτης δίνει μία φορά στην αρχή τη πολυωνυμική κωδικοποίηση της λειτουργίας του στον Ελεγκτή και μετά ο δεύτερος προσομοιώνει με αυτήν όλο το διάλογο βγάζοντας το τελικό αποτέλεσμα κι έτσι έχουμε δείξει ότι $EXP \subseteq MA$ κι άρα ασφαλώς ότι $EXP = MA$. ■

Παρατηρούμε στα παραπάνω παραδείγματα πως ο τρόπος που χρησιμοποιείται η υπόθεση ότι μια «μεγάλη» κλάση έχει $P_{/poly}$ κυκλώματα δεν είναι για πραγματικό υπολογισμό, όσο περισσότερο για εύκολα υπολογίσιμες πληροφορίες που πρέπει να επαληθευτούν. Είναι, δηλαδή, ένα καλό σημείο να σημειώσουμε πως τα μικρά κυκλώματα ως μοντέλο υπολογισμού δεν θα πρέπει να αντιμετωπίζονται τόσο ως επεξεργαστές, όσο ως αποδοτικά συμπυκνωμένη πληροφορία εκθετικού μήκους.

Ένα ερώτημα που προκύπτει είναι τι παραπάνω μπορούμε να αποκομίσουμε από αυτές τις συνεπαγωγές (των οποίων το κομμάτι της υπόθεσης είναι πιθανότατα ψευδές). Θα μπορούσε κανείς να υποθέσει ότι, με μία λίγο πιο περίπλοκη απόδειξη, οδηγούμαστε σε ακόμη περισσότερο μη αναμενόμενα αποτελέσματα, π.χ. ότι $P = NP$ - ενισχύοντας την άποψη ότι οι υποθέσεις αυτές δεν πρέπει να είναι αληθείς. Εν τούτοις εδώ (αλλά και σε κάποιες χαρακτηριστικές επόμενες περιπτώσεις), αυτή δεν είναι η κατάσταση, καθώς εύκολα μπορούμε να δούμε ότι αν $EXP \subseteq P_{/poly}$, τότε $P \neq NP$. Πράγματι αν ίσχυε το συμπέρασμα, τότε όλη η πολυωνυμική ιεραρχία θα κατέρρεε στο P και τότε (αφού $EXP = \Sigma_2^P$) θα είχαμε $EXP = P$ που αντιφάσκει με την ντετερμινιστική χρονική ιεραρχία (οπότε στη περίπτωση που $EXP \subseteq P_{/poly}$ όχι μονο $P \neq NP$, αλλά και $NP \nsubseteq DTIME[f(n)]$ για $f(f(n)) = o(2^n)$).

Ένα επόμενο ερώτημα είναι μέχρι πού μπορούμε να «υψώσουμε» αυτά τα συμπεράσματα υποθέσεων και κατά πόσο αξίζει αυτό; Για αρχή να σημειώσουμε ότι αν αν μπορούμε να ανέβουμε μέχρι ένα αντίστοιχο συμπέρασμα της μορφής $\Sigma_2^{EXP} \subseteq P_{/poly} \Rightarrow \Sigma_2^{EXP} \subseteq \Sigma_2^P$, τότε δείχνουμε απ' ευθείας ότι η υπόθεση δεν ισχύει, αφού αλλιώς έχουμε παραβίαση της Σ_2 χρονικής ιεραρχίας (η οποία αποδεικνύεται όμοια με τη μη ντετερμινιστική ιεραρχία, εφόσον όπως είπαμε αυτές οι αποδείξεις είναι ανεκτές στα μαντεία και $\Sigma_2 TIME[T(n)] =$

$NTIME^{NTIME[n]}[T(n)]$).

Προτού φτάσουμε όμως σε αυτό, ας δούμε πρώτα τι συμβαίνει με την κλάση $NEXP$. Αν επιχειρήσουμε να μιμηθούμε την απόδειξη για το EXP, τότε θα προσκρούσουμε στην αρχική κατάσταση, η οποία πέρα από την είσοδο περιέχει και τη «μαντεψιά» του πιστοποιητικού εκθετικού μήκους. Θα σκεφτεί κανείς ότι αφού $NEXP \subseteq P_{/poly}$, τότε υπάρχει και πολυωνυμικό κύκλωμα που παράγει το πιστοποιητικό και τότε τελειώσαμε. Εν τούτοις υπάρχει ένα λεπτό σημείο εδώ. Εμείς ορίσαμε όλες τις κλάσεις μόνο με προβλήματα απόφασης που επιστρέφουν ένα δυφίο. Επομένως η υποχρέωση για το κύκλωμα που επιστρέφει το πιστοποιητικό θα ήταν να επιστρέφει 1 για είσοδο $\langle x, i\rangle$ αν υπάρχει πιστοποιητικό για το x που στην i-οστή θέση έχει 1. Το θέμα είναι ότι δεν έχουμε κάποια εγγύηση ότι το πιστοποιητικό είναι μοναδικό και επομένως μπορεί η συνολική κατασκευή να επιστρέφει ένα πιστοποιητικό από το οποίο να προκύπτει ότι $x \notin L$, ενώ το μόνο που συνέβαινε είναι ότι το κύκλωμα C αναφερόταν σε διαφορετικά όντως έγκυρα πιστοποιητικά για κάθε i. Αυτό είναι ένα ζήτημα που θα σταθεί καίριας σημασίας στα όρια των αποδείξεων που έχουμε μέχρι σήμερα και το οποίο θα διευθετήσουμε εκτενώς στην επόμενη ενότητα. Πριν φτάσουμε όμως εκεί, είναι ένα καλό σημείο να κάνουμε μια επισκόπηση κάποιων συμπερασμάτων που προκύπτουν στην περίπτωση που δεν υπάρχουν εύκολα κυκλώματα για υπολογιστικά δύσκολες κλάσεις (το οποίο είναι και το τρέχον αναμενόμενο από τη πλευρά της επιστημονικής κοινότητας).

4.1.2 Περίπτωση ανυπαρξίας εύκολων κυκλωμάτων

Μέχρι στιγμής έχουμε εξάγει συμπεράσματα που αφορούν την περίπτωση όπου «δύσκολες» ομοιόμορφες υπολογιστικές κλάσεις (οι οποίες είτε αποδεδειγμένα έχουν είτε πιστεύεται ότι έχουν εκθετική χρονική δυσκολία (π.χ. EXP και NP αντίστοιχα)) επιδέχονται οικογένειες κυκλωμάτων πολυωνυμικού μεγέθους. Πολύ χρήσιμα συμπεράσματα, ωστόσο, μπορούμε να εξάγουμε και για την περίπτωση όπου δεν ισχύει αυτό (τα οποία παρουσιάζουν και μεγαλύτερο ενδιαφέρον ως πιο πιθανά να ισχύουν). Συγκεκριμένα οι ιδέες που κυριαρχούν σε αυτή την περίπτωση εκμεταλλεύονται αυτή την υπόθεση δυσκολίας με τεχνικές που προσομοιάζουν μοτίβα από την Κρυπτογραφία.

Συγκεκριμένα στην Κρυπτογραφία, σε αντίθεση με άλλους κλάδους της Θεωρητικής Πληροφορικής, ο κύριος σκοπός είναι η εκμετάλλευση της αδυναμίας (ή ανυπαρξίας) εύκολης λύσης κάποιων κλάσεων προβλημάτων παρά η προσπάθεια εξεύρεσης μίας. Ο τρόπος που αυτό συμβαίνει είναι ότι παίρνοντας ως δεδομένα ορισμένα κάτω φράγματα στη δυνατότητα αποδοτικής επίλυσης ορισμένων δύσκολων προβλημάτων, μετατρέπονται οι πληροφορίες προς κρυπτογράφηση σε στιγμιότυπα αυτών των προβλημάτων, με τρόπο τέτοιο ώστε οποιοσδήποτε τρίτος (με περιορισμένους χρονικούς πόρους) να μη μπορεί να εξάγει αυτή τη πληροφορία, με συντριπτικά μεγάλη πιθανότητα.

Στην τέλεια κρυπτογράφηση, αυτό μπορεί να συμβεί με απόλυτη επιτυχία, απλώς δημιουργώντας μία τυχαία συμβολοσειρά (μήκους όσο και το κείμενο

προς κρυπτογράφηση) και εφαρμόζοντας δυφίο προς δυφίο την πράξη xor με το προς κρυπτογράφηση κείμενο. Με εύκολη ανάλυση [Sha49a] προκύπτει ότι το παραγόμενο κείμενο δε μπορεί να αποκρυπτογραφηθεί από κανέναν τρίτο που δεν έχει το κλειδί (δηλαδή την τυχαία συμβολοσειρά). Από την άλλη πλευρά, αυτό έχει ένα μειονέκτημα (το οποίο έχει οδηγήσει στη μη χρήση αυτής της μεθόδου παρά μόνο σε πολύ σπάνιες περιπτώσεις): απαιτεί τη δημιουργία μιας πραγματικά τυχαίας συμβολοσειράς τόσο μεγάλης όσο και του κειμένου, κάτι που για πρακτικούς λόγους (ίσως και για θεωρητικούς) δεν είναι πάντοτε δυνατό.

Μια πιο βιώσιμη λύση, είναι η χρήση συμβολοσειρών που μοιάζουν με ομοιόμορφα τυχαίες, αλλά παράγονται από πολύ λιγότερα τυχαία δυφία (π.χ. πολυωνυμικά ή λογαριθμικά λιγότερα). Το ζήτημα είναι ότι τότε οι ψευδοτυχαίες αυτές συμβολοσειρές που παράγονται ίσως να κάνουν πιο εύκολη τη πλήρη (ή και ευριστικά μερική) δοκιμή τους και να οδηγούν στο σπάσιμο του κρυπτοκειμένου από κάποιον τρίτο που χρησιμοποιεί σχετικά λίγους πόρους. Θέλουμε λοιπόν στην Κρυπτογραφία να μπορούμε να παράξουμε κλειδιά (εν προκειμένω τυχαίες συμβολοσειρές) συνδυάζοντας λίγους πόρους κατά τη δημιουργία του κρυπτοκειμένου (εν προκειμένω λίγη τυχαιότητα) και μεγάλη δυσκολία εξεύρεσης του κλειδιού από τρίτους που περιορίζονται από εύλογο ποσό πόρων (εν προκειμένω εφικτό-πολυωνυμικό χρόνο).

Ήδη το παραπάνω παράδειγμα αποτελεί μια χαρακτηριστική περίπτωση όπου η (ψευδο-)τυχαιότητα παίζει κρίσιμο ρόλο στα πλαίσια κάποιου αλγορίθμου. Συγκεκριμένα, ζητούμενο ήταν η ύπαρξη ενός αλγορίθμου που με (σχεδόν) ντετερμινιστικό τρόπο κατασκευάζει μεγάλες συμβολοσειρές που να φαίνονται τυχαίες σε όλους τους υπόλοιπους (χρονικά) εφικτούς αλγόριθμους. Μια τέτοια διεργασία, ονομάζεται γενικά ψευδοτυχαία γεννήτρια και προφανώς πέρα από τη πρακτική της εφαρμογή σε παραδείγματα όπως τα παραπάνω, έχει και ιδιαίτερη εφαρμογή σε συμπεράσματα που αφορούν τη δυνατότητα ντετερμινιστικής επίλυσης προβλημάτων για τα οποία γνωρίζουμε μόνο τυχαιοκρατικούς εφικτούς αλγόριθμους (πιο συγκεκριμένα στη σχέση του BPP με το P). Αλγόριθμοι που ανήκουν σε αυτή την κατηγορία λέμε ότι χρησιμοποιούν τεχνικές αποτυχαιοποίησης και όπως θα δούμε, σχετίζονται στενά με τις ψευδοτυχαίες γεννήτριες.

Σε αυτό το σημείο μπορούμε να δώσουμε τον ακριβή ορισμό μιας ψευδοτυχαίας γεννήτριας.

Ορισμός 4.1.1. *(Ψευδοτυχαία Γεννήτρια)* Έστω $z(n)$ μία (Turing κατασκευάσιμη και αύξουσα) συνάρτηση $\mathbb{N} \to \mathbb{N}$. Μία συνάρτηση $G \in FDTIME[2^n]$ ($\{0,1\}^* \to \{0,1\}^*$) είναι μία z-ψευδοτυχαία γεννήτρια αν $|G(x)| = z(|x|)$ και επιπλέον για κάθε κύκλωμα C μεγέθους $z^3(|x|)$ ισχύει

$$\left|\Pr[C(G(\boldsymbol{U}_{|\boldsymbol{x}|})) = 1] - \Pr[C(\boldsymbol{U}_{\boldsymbol{z}(|\boldsymbol{x}|)}) = 1]\right| \leq \frac{1}{9}$$

όπου $\boldsymbol{U}_{|\boldsymbol{x}|}$, $\boldsymbol{U}_{\boldsymbol{z}(|\boldsymbol{x}|)}$ ομοιόμορφα τυχαίες δυαδικές συμβολοσειρές μήκους $|x|$ και $z(|x|)$ αντίστοιχα.

Το κύκλωμα C στον παραπάνω ορισμό, αντιστοιχεί στον χρονικά περιορισμένο τρίτο (με εν προκειμένω περισσότερους πόρους από μία απλή μηχανή Turing) για τον οποίον, το ζητούμενο είναι να έχει συμπεριφορά ως προς τις ψευδοτυχαίες συμβολοσειρές, όμοια με αυτή των αληθινά τυχαίων (το οποίο εκφράζεται από την τελευταία ανισότητα). Ασφαλώς οι σταθερές που εμφανίζονται στον παραπάνω ορισμό είναι αυθαίρετες και μπορούν να αλλάξουν κατά το δοκούν (στα αναμενόμενα λογικά πλαίσια, π.χ. το φράγμα της ανίσωσης πρέπει να είναι γνησίως μικρότερο του 1/2 κ.ο.κ.).

Ήδη από την εισαγωγική ανάλυση, έχει γίνει προφανές ότι σκοπός μας είναι να χρησιμοποιήσουμε τέτοιες ψευδοτυχαίες συμβολοσειρές για να «ξεγελάσουμε» BPP προγράμματα και να πάρουμε εικόνα της εξόδου τους με πολύ λιγότερες δοκιμές τυχαίων δυφίων από όσες θα απαιτούσε η πλήρης δοκιμή όλων των συνδυασμών τους. Συγκεκριμένα έχουμε την ακόλουθη πρόταση:

Πρόταση 4.1.1 ([Yao82, NW94, IW97]). *Έστω z όπως παραπάνω και l (πολυωνυμικά υπολογίσιμη) φυσική συνάρτηση. Αν υπάρχει z-ψευδοτυχαία γεννήτρια G, τότε έχουμε $BPTIME[z(l(n))] \subseteq DTIME[2^{O(l(n))}]$.*

Απόδειξη. Έστω ότι ισχύουν όλες οι υποθέσεις. Έχουμε ότι μία γλώσσα στο $BPTIME[z(l(n))]$ χρειάζεται το πολύ $z(l(n))$ τυχαία δυφία και ότι δοθέντων αυτών μπορεί να υπολογιστεί από ένα κύκλωμα C μεγέθους το πολύ $z^2(l(n))$. Δίνοντας στη G λοιπόν όλες τις εισόδους μήκους $l(n)$ θα πάρουμε εξ ορισμού στην έξοδο $2^{l(n)}$ τυχαίες συμβολοσειρές μήκους $z(l(n))$, οι οποίες «ξεγελούν» όλα τα κυκλώματα μεγέθους $z^3(l(n))$, συγκεκριμένα έχουμε ότι για το παραπάνω κύκλωμα C αν για κάποια είσοδο αποδέχεται, τότε για τα 2/3 των συνδυασμών των τυχαίων δυφίων οδηγείται σε αποδοχή και δεδομένου ότι η συλλογή των συμβολοσειρών που δίνει η ψευδογεννήτρια έχει σφάλμα το πολύ 1/9, τουλάχιστον τα $\frac{5}{9} > \frac{1}{2}$ αυτών των ψευδοτυχαίων συμβολοσειρών οδηγούν σε αποδοχή (όμοια και συμμετρικά για την περίπτωση της απόρριψης). Επομένως μπορούμε να φτιάξουμε με χρήση της z όλες τις $2^{l(n)}$ συμβολοσειρές (έκαστη εκ των οποίων απαιτεί χρόνο $O(2^{l(n)})$) σε συνολικό χρόνο $2^{O(l(n))}$ και από αυτές να τις δοκιμάσουμε όλες στο κύκλωμα C μεγέθους $z^2(l(n))$ (σε συνολικό ντετερμινιστικό χρόνο $z^2(l(n))2^{l(n)} = 2^{O(l(n))}$). Στη συνέχεια αποδεχόμαστε ή απορρίπτουμε ανάλογα με το τι κάνει η πλειοψηφία αυτών των ψευδοτυχαίων συμβολοσειρών και η απόδειξη ολοκληρώθηκε. ■

Αν λοιπόν, για παράδειγμα, διαθέτουμε μία $z(n) = 2^{cn}$ ψευδοτυχαία γεννήτρια, τότε θέτοντας $l(n) = logn$ καταλήγουμε ότι $BPP = P$. Το θέμα παραμένει ασφαλώς ότι δεν έχουμε μέχρι στιγμής απόδειξη για οποιαδήποτε $FDTIME[2^n]$ συνάρτηση ότι πληροί τις ζητούμενες προϋποθέσεις. Η μεγάλη συσχέτιση όμως του ορισμού της ψευδοτυχαιότητας με το μέγεθος των κυκλωμάτων που πρέπει να «ξεγελάει», έχει οδηγήσει στο ακόλουθο Θεώρημα, η τεχνική απόδειξη του οποίου παραλείπεται:

Θεώρημα 4.4 ([Uma03, STV01]). *Έστω μια γλώσσα $f \in DTIME[2^{O(n)}]$ δυσκολίας $H_f(n) \leq z(n)$ για κάποια z (αύξουσα και Turing κατασκευάσιμη).*

Τότε υπάρχει (κατασκευάσιμη) $(z^c(cn))$*-ψευδοτυχαία γεννήτρια για κάποια σταθερά* $c > 0$.

Από το παραπάνω Θεώρημα μαζί με την προηγούμενη Πρόταση μπορούμε να καταλήξουμε στα ακόλουθα συμπεράσματα που συνδέουν την όλο και υψηλότερη κυκλωματική δυσκολία συναρτήσεων με την όλο και αποδοτικότερη αποτυχαιοποίηση τυχαιοκρατικών αλγορίθμων.

Πρόταση 4.1.2. *Έστω* $E = DTIME[2^{O(n)}]$. *Τότε*

- *αν* $E \nsubseteq P_{/poly}$, *τότε* $BPP \subseteq subEXP$,
- *αν* $E \nsubseteq subEXPSIZE$, *τότε* $BPP \subseteq quasiP$ *και πιο συγκεκριμένα*
- *αν* $E \nsubseteq subESIZE$, *τότε* $BPP = P$

Απόδειξη. Πράγματι, έστω $E \nsubseteq P_{/poly}$. Τότε υπάρχει $f \in E$, τέτοια ώστε $H_f(n) \geq n^c$ για κάθε c. Άρα από προηγούμενο Θεώρημα, υπάρχει γεννήτρια που είναι n^d-ψευδοτυχαία γεννήτρια για κάθε d κι άρα από προηγούμενη Πρόταση για $z(n) = n^{bd}$ και $l(n) = n^{1/d}$ (για αυθαίρετο $b > 0$) ότι $BPTIME[n^b] \subseteq DTIME[2^{O(n^{1/d})}]$ για κάθε $d > 0$, από το οποίο καταλήγουμε ότι πράγματι $BPP \subseteq subEXP$. Με παρόμοιο τρόπο αποδεικνύονται και τα υπόλοιπα. ■

Σε κάθε περίπτωση λοιπόν, έχουμε και κάποιο όφελος ανάλογα με το πού ανήκει το EXP. Αν είναι εντός του $P_{/poly}$, τότε η πολυωνυμική ιεραρχία καταρρέει στο δεύτερο επίπεδο, ενώ αν δεν είναι, τότε οι τυχαιοκρατικοί αλγόριθμοι δεν είναι δραματικά πιο ισχυροί από τους ντετερμινιστικούς (και μάλιστα όσο πιο «μακριά» από το $P_{/poly}$ βρίσκονται τόσο πιο εύκολη είναι η αποτυχαιοποίηση). Ακόμη κι έτσι όμως, δεν έχουμε καταλήξει ακόμη σε κάποιο στέρεο άνευ όρων συμπέρασμα. Τα παραπάνω αποτελούν συμπεράσματα, έκαστο εκ των οποίων συνάγεται από κάποιο κάτω φράγμα για τις εκθετικές συναρτήσεις, κάτι που προς το παρόν δεν έχουμε δει για καμιά κλάση. Στην ενότητα που ακολουθεί θα προχωρήσουμε προς αυτή την κατεύθυνση.

4.2 Μη ντετερμινισμός και κάτω φράγματα

4.2.1 Συμπύκνωση πληροφορίας

Επιστρέφοντας στη προσπάθεια εξερεύνησης των υποθέσεων όπου μια δύσκολη κλάση περιγράφεται από πολυωνυμικά κυκλώματα (με απώτερο σκοπό την απόρριψη τους ως ψευδείς), αυτό που θέλαμε να δείξουμε για να συνεχίσουμε την πορεία μας, είναι ότι αν $NEXP \subseteq P_{/poly}$, τότε υπάρχει συμπυκνώσιμο πιστοποιητικό για κάθε $NEXP$ στιγμιότυπο. Με άλλα λόγια για κάθε γλώσσα $L \in NEXP$ και για κάθε x έχουμε $x \in L$ αν και μόνο αν υπάρχει πολυωνυμικό κύκλωμα C, του οποίο ο πίνακας αληθείας είναι ένα στιγμιότυπο του x για την L.

Προτού συνεχίσουμε ας δούμε τον ακόλουθο ορισμό της συμπύκνωσης:

Ορισμός 4.2.1. Λέμε ότι μια πληροφορία (συμβολοσειρά) I είναι $z(|I|)$ - συμπυκνώσιμη (με $z(|I|) = O(|I|)$), αν υπάρχει κύκλωμα C μεγέθους $z(|I|)$ και $log|I|$ εισόδων, τέτοια ώστε $T_C = I$ (όπου T_C ο πίνακας αληθείας του C). Λέμε ότι μια οικογένεια πληροφοριών είναι $z(N)$-συμπυκνώσιμη, αν κάθε πληροφορία της (μήκους N) είναι $z(N)$-συμπυκνώσιμη.

Δηλαδή μια πληροφορία είναι τόσο πιο συμπυκνώσιμη όσο πιο μικρά κυκλώματα υπάρχουν που να την κατασκευάζουν μέσω του πίνακα αληθείας τους. Επίσης αν $z(N) = polylog(N)$, τότε συχνά θα αναφέρουμε ότι είναι απλά συμπυκνώσιμη, χωρίς να προσδιορίζουμε την ακριβή συνάρτηση. Να παρατηρήσουμε επίσης, ότι κάθε πληροφορία είναι $O(N)$-συμπυκνώσιμη, αφού υπάρχει ένα προφανές κύκλωμα μεγέθους $O(N)$ που έχει αποθηκεύσει όλη τη πληροφορία σε μορφή πυλών. Εδώ το N αναφέρεται στο μήκος της πληροφορίας που αντιστοιχεί σε 2^n αν $n = logN$ οι είσοδοι του αντίστοιχου κυκλώματος. Στη συνέχεια θα περιγράφουμε το μέτρο της συμπύκνωσης χρησιμοποιώντας αυτή τη μεταβλητή (το πλήθος των δυφίων εισόδου) ως μεταβλητή αναφοράς (κι έτσι θα έχουμε π.χ. $poly(n), 2^{o(n)}$ συμπυκνώσιμες πληροφορίες κ.ο.κ.)

Μια εύλογη ερώτηση είναι αν όλες οι πληροφορίες είναι συμπυκνώσιμες κι αν όχι, πόσες από αυτές είναι; Η απάντηση, όμως, έχει ήδη δοθεί από το Λήμμα 3.3.1, όπου δείξαμε ότι σχεδόν όλες οι συναρτήσεις $\{0,1\}^{logn} \to \{0,1\}$ (κάθε μία εκ των οποίων αντιστοιχεί αμφιμονοσήμαντα σε μία συμβολοσειρά μήκους n) θέλουν κυκλώματα μεγέθους τουλάχιστον $\Omega(n/logn)$ για να παραχθούν και μόνο ένα πολύ μικρό ποσοστό που τείνει στο 0 έχει μικρότερα κυκλώματα.

Μπορεί κανείς να σκεφτεί τη συμπύκνωση μιας συμβολοσειράς ως μια γενίκευση της συμπίεσης, της τεχνικής που χρησιμοποιείται καθημερινά σχεδόν σε όλες τις μεταφορές μεγάλων αρχείων στα σύγχρονα υπολογιστικά συστήματα. Γενικά όλα τα καθιερωμένα μοντέλα συμπίεσης μπορούν να μοντελοποιηθούν από μια γραμματική χωρίς συμφραζόμενα, η οποία έχει ένα μοναδικό κανόνα για κάθε μη τερματικό σύμβολο (κι άρα επίσης δεν έχει κύκλους στους κανόνες της) και η οποία έχει μια μοναδική λέξη, τη συμβολοσειρά που θέλουμε να συμπιέσουμε [KY00, CLL$^+$05]. Ως μήκος της συμπίεσης θεωρείται τότε το μήκος της γραμματικής. Είναι εύκολο να δείξουμε ότι αν μια συμβολοσειρά είναι συμπιέσιμη μέσω μιας γραμματικής τέτοιου μεγέθους, τότε υπάρχει και συμπύκνωση αυτής με το πολύ πολυωνυμική επιβάρυνση σε σχέση με το μήκος της συμπίεσης (προκύπτει από την ύπαρξη γρήγορου αλγορίθμου που επιστρέφει το i-οστό δυφίο μιας λέξης που παράγεται από γραμματική χωρίς συμφραζόμενα).

Ωστόσο η συμπύκνωση είναι γνησίως πιο ισχυρή από τη συμπίεση. Αυτό διαισθητικά είναι αναμενόμενο, αφού σε συμπιέσιμες συμβολοσειρές είναι εύκολο να βρεθεί αν υπάρχει άσσος, ενώ σε συμπυκνώσιμες θεωρούμε πως μάλλον όχι (αφού πρόκειται για NP-πλήρες πρόβλημα). Επιπλέον μπορεί να αποδειχθεί και τυπικά, αν θεωρήσουμε τη συμβολοσειρά μήκους $nlogn$ που περιέχει διαδοχικά όλες τις n συμβολοσειρές μήκους $logn$ (με κάποιο ενδιάμεσο διαχωριστικό σύμβολο). Πράγματι, τότε είναι εύκολο να δούμε ότι δε μπορούμε να έχουμε

επανάληψη κάποιου τερματικού συμβόλου μήκους μεγαλύτερου από $logn$ κι άρα πρέπει να υπάρχουν τουλάχιστον $\Omega(n)$ διαφορετικά μη τερματικά σύμβολα στη γραμματική συμπίεσης. Ο κύριος λόγος, όπως φαίνεται και από το παραπάνω παράδειγμα, είναι ότι οι συμπιέσιμες πληροφορίες έχουν πολλά επαναλαμβανόμενα σημεία (επειδή έχουν μικρό μέγεθος γραμματικής κι άρα μη τερματικών συμβόλων που αντιστοιχούν σε υποσυμβολοσειρές) ενώ οι συμπυκνώσιμες αρκεί να έχουν μία υπολογιστικά επαναλαμβανόμενη ιδιότητα (στο παραπάνω παράδειγμα, η ιδιότητα αυτή είναι ότι πρόκειται για μια απλή αριθμητική ακολουθία που υπολογίζεται από μικρά κυκλώματα).

Σε κάθε περίπτωση, η συμπύκνωση είναι η πιο γενική μέθοδος αναφοράς σε πληροφορία η οποία μπορεί να «χωρέσει» σε μια μικρότερη αυτόνομη πληροφορία (με την έννοια ότι για να εξάγουμε ένα δυφίο της αρχικής πληροφορίας δεν χρειαζόμαστε υπολογιστικό χρόνο μεγαλύτερο του μήκους της συμπυκνωμένης πληροφορίας). Υπό αυτό το πρίσμα γίνεται εμφανές γιατί οι περισσότερες πληροφορίες δεν είναι συμπυκνώσιμες· αν ήταν τότε ένα μεγάλο ποσοστό του $\{0,1\}^n$ θα ήταν αντιστοιχίσιμο στο $\{0,1\}^{polylog(n)}$.

Ο έσχατος τρόπος συμπύκνωσης ασφαλώς, δεν είναι άλλος από την Kolmogorov πολυπλοκότητα μιας συμβολοσειράς [Kol63], η οποία αντιστοιχεί στο ελάχιστο μήκος της περιγραφής μιας μηχανής Turing , η οποία παράγει αυτή τη συμβολοσειρά (με κενή είσοδο). Ασφαλώς η κυκλωματική πολυπλοκότητα είναι υποσύνολο της Kolmogorov πολυπλοκότητας (με την έννοια ότι κάθε κύκλωμα μπορεί προφανώς να αντιστοιχηθεί σε μία μηχανή Turing που περιέχει το κύκλωμα κωδικοποιημένο στις καταστάσεις της), αλλά δεν ισχύει το αντίστροφο. Αυτό, επειδή υπάρχουν συμβολοσειρές που είναι κυκλωματικά δύσκολες, οι οποίες όμως μπορούν να παραχθούν από μία ντετερμινιστική μηχανή Turing (η οποία απλά ψάχνει όλες τις συμβολοσειρές μέχρι να βρει μια δύσκολη), η περιγραφή της οποίας είναι ασφαλώς πολύ μικρή (λόγω της επαναληψιμότητας του αλγορίθμου) παρότι ασφαλώς ο χρόνος εκτέλεσης της είναι μεγάλος. Στο σημείο αυτό υπάρχει και μία έντονη συσχέτιση με τη κυκλωματική δυσκολία εκθετικών κλάσεων. Πράγματι αν $EXP \not\subseteq P/poly$, τότε παίρνουμε ένα άμεσο στιγμιότυπο κυκλωματικά δύσκολης συμβολοσειράς που παράγεται ομοιόμορφα από μια εκθετική μηχανή (με προφανώς μικρή περιγραφή) κι άρα με χαμηλή Kolmogorov πολυπλοκότητα.

4.2.2 Συμπυκνώσιμα πιστοποιητικά για το $NEXP$

Θα δούμε τώρα ότι αν το $NEXP$ επιδέχεται σύντομα κυκλώματα, τότε υπάρχουν και συμπυκνώσιμα πιστοποιητικά για κάθε στιγμιότυπο του.

Θεώρημα 4.5 ([IKW02]). *Αν $NEXP \subseteq P_{/poly}$, τότε κάθε ικανοποιήσιμο στιγμιότυπο ενός $NEXP$ προβλήματος επιδέχεται τουλάχιστον ενός πολυωνυμικά συμπυκνώσιμου πιστοποιητικού.*

Απόδειξη. Έστω ότι $NEXP \subseteq P_{/poly}$. Έστω επιπλέον ότι δεν υπάρχει πάντα πολυωνυμικά συμπυκνώσιμο πιστοποιητικό, δηλαδή ότι για κάθε $c > 0$ για άπει-

ρα πολλά n, υπάρχει είσοδος x, μήκους n, για την οποία υπάρχει πιστοποιητικό y μήκους 2^{n^d} (σε ένα $NEXP$ πρόβλημα με κατηγόρημα ελέγχου R - δηλαδή $x \in L \Leftrightarrow \exists y : R(x, y)$ το οποίο τρέχει σε επίσης χρόνο 2^{n^d}), το οποίο ωστόσο δεν είναι n^c συμπυκνώσιμο. Όμως τότε, το πιστοποιητικό αυτό ενέχει μέσα του δυσκολία, η οποία δε μπορεί να περιγραφεί από κανένα πολυωνυμικό κύκλωμα κι άρα από Θεώρημα 4.4 έχουμε ότι η συμβολοσειρά αυτή μπορεί να χρησιμοποιηθεί για παραγωγή τυχαίων δυφίων, τα οποία «ξεγελούν» όλα τα πολυωνυμικά κυκλώματα. Από την άλλη όμως, έχουμε λόγω της υπόθεσης ότι $EXP \subseteq P_{/poly}$ κι άρα από Θεώρημα 4.3 έχουμε ότι $EXP = MA$. Αντί λοιπόν να χρησιμοποιηθούν τυχαία δυφία στη προσομοίωση, με εκμετάλλευση του μη ντετερμινισμού, θα μαντέψουμε το δύσκολο y και θα χρησιμοποιήσουμε αυτό για την παραγωγή των τυχαίων δυφίων. Η επαλήθευση του y όμως μπορεί να γίνει από το R σε χρόνο 2^{n^d}. Προκύπτει επομένως ο ακόλουθος μη ντετερμινιστικός αλγόριθμος για κάθε EXP πρόβλημα χρόνου 2^{n^b} δοθείσας της προαναφερθείσας εισόδου x: Μάντεψε ένα πιστοποιητικό y μήκους 2^{n^d} καθώς και το κύκλωμα του αποδείκτη μεγέθους n^B (όπου το B είναι σταθερά που εξαρτάται μόνο από το b). Επαλήθευσε το πιστοποιητικό y μέσω του R σε χρόνο 2^{n^d} και στη συνέχεια χρησιμοποίησε το ως πίνακα αληθείας για τη προσομοίωση των τυχαίων δυφίων και της MA απόδειξης μέσω του μαντεμένου κυκλώματος του Αποδείκτη (που απαιτεί συνολικά πολυωνυμικό χρόνο). Ο παραπάνω αλγόριθμος χρειάζεται λοιπόν το πολύ 2^{3n^d} μη ντετερμινιστικό χρόνο κι άρα από υπόθεση επιδέχεται κυκλώματα μεγέθους n^D για τα οποία μπορούμε να θέσουμε ως σταθερή είσοδο το x που δημιουργεί στιγμιότυπο που έχει μόνο δύσκολα πιστοποιητικά (ασφαλώς για τα n για τα οποία υπάρχει τέτοιο x (τα οποία είναι άπειρα) - για τα υπόλοιπα βάζουμε οποιοδήποτε x) παίρνοντας εν τέλει οικογένειες κυκλωμάτων μεγέθους n^D οι οποίες προσομοιώνουν για άπειρα μήκη εισόδου n οποιοδήποτε EXP πρόβλημα σωστά. Ωστόσο μπορούμε τετριμμένα με διαγωνιοποίηση να φτιάξουμε αλγόριθμο χρόνου $2^{n^{D+1}}$ που να ελέγχει όλα τα κυκλώματα μεγέθους n^D και να έχει ως πίνακα αληθείας τη πρώτη συμβολοσειρά που δε παράγεται από κανένα τέτοιο κύκλωμα, οδηγώντας σε άτοπο. ■

Από το παραπάνω Θεώρημα προκύπτει εύκολα η ακόλουθη πρόταση:

Πρόταση 4.2.1. *Αν $NEXP \subseteq P_{/poly}$, τότε $NEXP = EXP$.*

Απόδειξη. Από προηγουμένως έχουμε ότι αν $NEXP \subseteq P_{/poly}$, τότε υπάρχει c, τέτοιο ώστε τα ικανοποιήσιμα στιγμιότυπα να έχουν τουλάχιστον ένα n^c-συμπυκνώσιμο πιστοποιητικό. Επομένως για κάθε είσοδο δοκιμάζουμε όλα τα δυνατά 2^{n^c} κυκλώματα μεγέθους n^c και δοκιμάζουμε τον πίνακα αληθείας τους ως υποψήφιο πιστοποιητικό. Αν κάποιο επαληθεύσει το R, ασφαλώς αποδεχόμαστε, ενώ αν δε βρεθεί κανένα σημαίνει από υπόθεση ότι είναι μη ικανοποιήσιμο το στιγμιότυπο, οπότε απορρίπτουμε. Συνολικά χρειαστήκαμε $2^{poly(n)}$ ντετερμινιστικό χρόνο κι άρα $NEXP \subseteq EXP$, δηλαδή $NEXP = EXP$. ■

Στο σημείο αυτό μπορούμε μάλιστα να εξάγουμε το πρώτο σίγουρο αρνητικό αποτέλεσμα:

Πρόταση 4.2.2. $\Sigma_2^{EXP} \not\subseteq P_{/poly}$

Απόδειξη. Έστω ότι $\Sigma_2^{EXP} \subseteq P_{/poly}$. Όμως τότε $EXP = \Sigma_2^P$ κι άρα με ένα επιχείρημα παραγεμίσματος (ανάλογο αυτού στο [AB09]) καταλήγουμε ότι $EXP^{EXP} = \Sigma_2^{EXP}$. Όμως το EXP^{EXP} δεν έχει πολυωνυμικά κυκλώματα (αφού υπάρχει σε αυτή τη κλάση τετριμμένος αλγόριθμος που ψάχνει όλες τις συμβολοσειρές μήκους 2^n και παίρνει τη πρώτη που δεν περιγράφεται από πολυωνυμικό κύκλωμα). Στην πραγματικότητα, όπως θα δούμε λίγο παρακάτω το Σ_2^{EXP} είναι ακόμη πιο δύσκολο κυκλωματικά. ■

Γενικά κλειδί της απόδειξης του παραπάνω Θεωρήματος είναι ότι όταν υποθέτουμε πως $NEXP \subseteq P/poly$, προκύπτει ότι είναι τόσες πολλές οι κλάσεις που επιδέχονται επίσης πολυωνυμικά κυκλώματα, σε σημείο που το μόνο που χρειάζεται ιδιαίτερης μέριμνας είναι τα τυχαία δυφία. Συγκεκριμένα αν μπορούμε να παράξουμε τυχαία δυφία χωρίς τη χρήση εκθετικού μη ντετερμινισμού, τότε λόγω των παραπάνω, θα μπορούσαμε να εκτελέσουμε το $NEXP$ σε υποεκθετικό μη ντετερμινιστικό χρόνο, οδηγώντας σε άτοπο από μη ντετερμινιστική χρονική ιεραρχία. Συγκεκριμένα έχουμε το ακόλουθο Θεώρημα:

Θεώρημα 4.6 ([IKW02]). *Αν $promiseBPP \subseteq subNEXP$, τότε $NEXP \not\subseteq P_{/poly}$.*

Απόδειξη. Έστω ότι ισχύει τόσο η υπόθεση όσο και $NEXP \subseteq P_{/poly}$. Όμως, τότε έχουμε από τη προηγούμενη πρόταση ότι ένα $NEXP$ πρόβλημα χρόνου 2^{n^c} έχει EXP πρόβλημα χρόνου 2^{n^d} κι άρα μία διαλογική απόδειξη MA χρόνου n^D. Επομένως το μόνο που χρειάζεται είναι να μαντέψουμε το κύκλωμα μεγέθους το πολύ $n^{D'}$ και στη συνέχεια να ελέγξουμε αν για τους περισσότερους συνδυασμούς των τυχαίων δυφίων οδηγούμαστε σε αποδοχή για αυτό το κύκλωμα. Αυτό όμως δεν είναι άλλο από ένα $promiseBPP$ πρόβλημα με είσοδο ένα συνολικό κύκλωμα μεγέθους $poly(n)$ (και $poly(n)$ τυχαίων δυφίων) κι επομένως χρειάζεται $f(poly(n))$ μη ντετερμινιστικός χρόνος για την επίλυση του όπου f μία υποεκθετική συνάρτηση, η οποία αντιστοιχεί στο χρόνο που απαιτεί η επίλυση του αντίστοιχου $promiseBPP$ στιγμιότυπου. Επομένως συνολικά χρειαστήκαμε υποεκθετικό μη ντετερμινιστικό χρόνο για την προσομοίωση της διαλογικής απόδειξης, η οποία επιλύει το αρχικό $NEXP$ πρόβλημα κι άρα καταλήγουμε ότι $NEXP \subseteq subNEXP$ καταλήγοντας σε άτοπο, από μη ντετερμινιστική χρονική ιεραρχία. ■

Να σημειωθεί ότι χρειάστηκε στην υπόθεση να πάρουμε το $promiseBPP$ κι όχι απλώς το BPP, καθώς στην περίπτωση αποδοχής θα υπάρχουν (πέραν του σωστού κυκλώματος) και κυκλώματα Αποδεικτών που δεν οδηγούν σε αποδοχή (ή απόρριψη) στη μεγάλη πλειοψηφία των συνδυασμών των τυχαίων δυφίων. Εν τούτοις υπάρχει ένα αντίστοιχο αποτέλεσμα για το BPP

[KI03], συγκεκριμένα ότι αν $BPP = P$, τότε είτε $NEXP \not\subseteq P_{/poly}$, είτε η $Permanent$ δεν υπολογίζεται από πολυωνυμικά αριθμητικά κυκλώματα (ισοδύναμα $\#P \not\subseteq AlgP_{/poly}$). Στο σημείο αυτό, έχει γίνει πλέον φανερή η στενή συσχέτιση της ψευδοτυχαιότητας με τη κυκλωματική δυσκολία εκθετικών κλάσεων. Όπως είχαμε δει προηγουμένως, έχουμε ότι όσο πιο κυκλωματικά δύσκολο είναι το EXP, τόσο πιο εύκολο είναι το BPP, ενώ ρυθμίζοντας κατάλληλα τις παραμέτρους του προηγούμενου θεωρήματος καταλήγουμε ότι όσο πιο εύκολο είναι το $promiseBPP$, τόσο πιο δύσκολο είναι το $NEXP$. Τα συμπεράσματα αυτά μάλιστα επεκτείνονται και σε υποεκθετικές κλάσεις (ή υπερπολυωνυμικά κυκλώματα), αλλάζοντας κατάλληλα τις παραμέτρους των υποθέσεων. Επιπλέον συνεχίζουν να έχουν αντίστοιχες μορφές (με πιο χαλαρές ή ισχυρές υποθέσεις-συμπεράσματα) ακόμα κι όταν παίρνουμε τη $promise$ ή μη εκδοχή του BPP.

Η παραπάνω ιδιότητα των κυκλωμάτων αποτελεί μία από τις σημαντικότερες τους, καθώς θεμελιώνει πλέον ακράδαντα τη σημασία αυτού του μοντέλου ακόμη κι αν δεν αντιστοιχεί σε ρεαλιστικό τρόπο υπολογισμού. Πράγματι μέχρι στιγμής είχαμε αποτελέσματα της μορφής που υπέθεταν ότι μία ομοιόμορφη κλάση επιδέχεται εύκολα κυκλώματα και οδηγούμασταν σε ένα «παράδοξο» συμπέρασμα όπως ότι η πολυωνυμική ιεραρχία καταρρέει. Εν τούτοις αυτό θα αποτελούσε χρήσιμη πληροφορία, μόνο αν εν τέλει υπήρχαν εύκολα κυκλώματα για αυτές τις κλάσεις (το οποίο δεν είναι το τρέχον αναμενόμενο από την επιστημονική κοινότητα) και συνεπώς με μια κυνική σκοπιά, δε συνεισφέρουν κάτι καινούριο τέτοιες συνεπαγωγές. Με άλλα λόγια, το αναμενόμενο συμπέρασμα (δηλαδή ότι $EXP \not\subseteq P_{/poly}$) δε μας δίνει μέσω αυτών των θεωρημάτων κάποια πληροφορία για την πολυωνυμική ιεραρχία (καθώς σε τελική ανάλυση, «πρακτική» αξία έχουν μόνο τα συμπεράσματα που αναφέρονται σε μηχανές Turing).

Αυτό ήρθε να διορθωθεί σε μεγάλο ποσοστό από τα συμπεράσματα ότι όσο πιο δύσκολα κυκλώματα υπάρχουν για εκθετικές κλάσεις, τόσο πιο εύκολα γίνεται η αποτυχαιοποίηση. Επομένως πλέον σε κάθε περίπτωση έχουμε ένα συμπέρασμα από τον κόσμο των κυκλωμάτων, για τον κόσμο των υπολογιστών. Γνωρίζουμε πλέον ότι ένα από τα δύο, η προσθήκη πολλών ποσοδεικτών ή η προσθήκη τυχαίων ψηφίων, δεν αυξάνει δραματικά την υπολογιστική δυσκολία. Ασφαλώς, και μόνο από τα παραπάνω ο κόσμος των κυκλωμάτων είναι ένας τομέας που έχει θεμελιώσει αδιαμφισβήτητα την θεωρητική (αλλά και πρακτική) του αξία. Το τελευταίο που απέμενε είναι να συνδεθεί μόνιμα με έναν ξεκάθαρο τομέα των ομοιόμορφων κλάσεων και αυτό πλέον ολοκληρώθηκε από το παραπάνω Θεώρημα, όπου έγινε καθαρή η αντιστοίχιση (σε μορφή σχεδόν ισοδυναμίας) της αποτυχαιοποίησης με τη κυκλωματική δυσκολία ομοιόμορφων κλάσεων.

Ο τρόπος μάλιστα που έγινε αυτή η αντιστοίχιση ήταν στενά συνδεδεμένος με τη βασική έννοια της ψευδοτυχαιότητας, όπου σκοπός δεν είναι η παραγωγή κάποιας αληθινά τυχαίας κατανομής, αλλά μιας κατανομής που να φαίνεται τυχαία σε συγκεκριμένες εύκολες κλάσεις (ξοδεύοντας όσο το δυνατόν λιγότερους πόρους για τη παραγωγή της). Το τελικό συμπέρασμα λοιπόν είναι ότι

τα κυκλώματα δεν είναι χρήσιμα ως ένα ρεαλιστικό μοντέλο υπολογισμού. Είναι χρήσιμα ως ένα εργαλείο που μετράει και κατατάσσει την πολυπλοκότητα που εμπεριέχουν και μπορούν να παράξουν κλάσεις που πηγάζουν από τέτοια ρεαλιστικά μοντέλα.

4.2.3 Κάτω όρια για μεγαλύτερες κλάσεις

Είδαμε στη προηγούμενη υποενότητα, πόσο διαφορετικά πρέπει να προσεγγίσουμε την περίπτωση του $NEXP$ ώστε να καταλήξουμε στο συμπέρασμα ότι αν επιδέχεται οικογένεια πολυωνυμικών κυκλωμάτων, τότε ισούται με το Σ_2^P, ένα συμπέρασμα στο οποίο καταλήξαμε για ανάλογες κλάσεις από το NP και πάνω. Ένα φυσικό ερώτημα που προκύπτει είναι μέχρι ποια κλάση μπορούμε να υψώσουμε αυτό το γενικό συμπέρασμα και από ποιο σημείο και πάνω έχουμε τη μη ύπαρξη πολυωνυμικών κυκλωμάτων χωρίς υποθέσεις. Ας μελετήσουμε για αρχή την περίπτωση του EXP^{NP}, δηλαδή τη κλάση των προβλημάτων που λύνονται σε εκθετικό χρόνο με τη χρήση ενός SAT μαντείου.

Θεώρημα 4.7. *Αν* $EXP^{NP} \subseteq P_{/poly}$, *τότε* $EXP^{NP} = \Sigma_2^P$.

Απόδειξη. Έστω μία EXP^{NP} γλώσσα L και x μία είσοδος μήκους n, η οποία τρέχει σε μηχανή M χρόνου 2^{n^c} η οποία χρησιμοποιεί SAT μαντείο. Υπάρχει λοιπόν πολυωνυμικό κύκλωμα C_1, το οποίο δέχεται ως είσοδο μία τούπλα $\langle x, j\rangle$ κι επιστρέφει την απάντηση της j-οστής ερώτησης στο SAT μαντείο όταν τρέξουμε τη M με είσοδο x (και 0 αν δεν υπάρχει j-οστή ερώτηση). Προφανώς το μήκος του j είναι το πολύ n^c κι άρα η τούπλα εισόδου και κατ΄ επέκταση το κύκλωμα C_1 έχει πολυωνυμικό μέγεθος ως προς το n, έστω n^d. Έστω τώρα το εξής πρόγραμμα που παίρνει ως είσοδο μία τούπλα $\langle x, C, j\rangle$ και αφού ελέγξει ότι το C είναι ορθή κωδικοποίηση ενός κυκλώματος n^c εισόδων και μεγέθους n^d, κατασκευάζει τον πίνακα αληθείας του (σε χρόνο $2^{n^c}|C|$), έστω y. Στη συνέχεια τρέχει τη M για $j-1$ ερωτήματα στο SAT θεωρώντας ότι οι απαντήσεις του μαντείου είναι αυτές του αντίστοιχου δυφίου του y (χωρίς να ρωτάει στα αλήθεια το μαντείο) και ελέγχει αν το j-οστό ερώτημα στο μαντείο έχει απάντηση ίδια με το j-οστό δυφίο του y. Έχουμε λοιπόν ένα SAT ερώτημα, το οποίο προσδιορίζεται πλήρως από τα $\langle x, C, j\rangle$ και μήκους το πολύ 2^{n^c}. Από προηγουμένως, λοιπόν έχουμε ότι αν είναι ικανοποιήσιμο τότε έχει n^b συμπυκνώσιμο πιστοποιητικό κι άρα αρκεί να τα ψάξουμε όλα για να ελέγξουμε αν ικανοποιείται (να σημειωθεί ότι το b εξαρτάται μόνο από το c κι όχι από το j). Εν τέλει, για κάθε είσοδο $\langle x, C, j\rangle$, το παραπάνω πρόγραμμα κάνει χρόνο το πολύ $2^{n^{\max\{b,c\}+1}}$, έστω 2^{n^f}, όπου n το μήκος του $|x|$. Το πρόγραμμα λοιπόν που παίρνει ως είσοδο το $\langle x, C\rangle$ κι ελέγχει αν τα δυφία του y (που παράγεται από το C) συμβαδίζουν με όλες τις κλήσεις στο SAT μαντείο θα εκτελέσει το πολύ 2^{n^c} κλήσεις στο παραπάνω εκθετικό πρόγραμμα κι άρα θα χρειαστεί συνολικά 2^{n^A} χρόνο, για κάποιο $A > 0$. Επομένως συνοψίζοντας έχουμε ότι με είσοδο x μήκους n υπάρχει κύκλωμα C μεγέθους n^d, τέτοιο ώστε το παραπάνω

ντετερμινιστικό πρόγραμμα ελέγχει σε χρόνο 2^{n^A} αν το x ανήκει στην L χρησιμοποιώντας και ταυτόχρονα ελέγχοντας την ορθότητα των απαντήσεων του υποτιθέμενου μαντείου SAT στα αντίστοιχα ερωτήματα. Επειδή λοιπόν γίνεται και έλεγχος των αντίστοιχων απαντήσεων, έχουμε ότι μόνο μία συμβολοσειρά μπορεί να ικανοποιεί τα παραπάνω (αυτή που αντιστοιχεί στις (μοναδικές) απαντήσεις του SAT μαντείου) και ότι σίγουρα υπάρχει μία τέτοια. Ο τελικός αλγόριθμος, λοιπόν, παίρνει ως είσοδο το x και δοκιμάζει όλα τα κυκλώματα C μέχρις ότου ο πίνακας αληθείας κάποιου να επαληθεύεται για όλο το μήκος της προσομοίωσης. Μόλις βρεθεί (σίγουρα θα συμβεί κάποτε λόγω των παραπάνω), ο αλγόριθμος επιστρέφει αποδοχή ή απόρριψη αντίστοιχα με το τι κάνει η προσομοίωση. Ο τελικός αλγόριθμος θα πάρει το πολύ $2^{n^A+n^d}poly(n)$ ντετερμινιστικό χρόνο κι άρα $EXP^{NP} \subseteq EXP$ κι επομένως $EXP^{NP} = \Sigma_2^P$. ■

Μέχρι στιγμής έχουμε δει κυρίως υποθέσεις για ομοιόμορφες κλάσεις, οι οποίες αν επιδέχονται πολυωνυμικά κυκλώματα, τότε οδηγούν στο να καταρρέει η πολυωνυμική ιεραρχία στο δεύτερο επίπεδο, αποτελώντας μια ισχυρή ένδειξη ότι δε πρέπει να είναι αυτή η περίπτωση. Θα δούμε τώρα ορισμένες κλάσεις, οι οποίες χωρίς υποθέσεις δεν επιδέχονται πολυωνυμικά κυκλώματα. Πρώτα από όλα δεδομένου ότι οι περισσότερες συναρτήσεις $\{0,1\}^n \to \{0,1\}$ όχι μόνο δεν έχουν πολυωνυμικά κυκλώματα, αλλά μάλιστα απαιτούν κυκλώματα εκθετικού μεγέθους (όπως έχουμε δει) αρκεί να τις ψάξουμε όλες και να πάρουμε την πρώτη που δεν έχει υποεκθετικό κύκλωμα (σίγουρα υπάρχει μία – οι περισσότερες μάλιστα είναι τέτοιες). Επομένως υπάρχει πρόγραμμα 2^{2^n} που κάνει το παραπάνω και προφανώς δεν έχει πολυωνυμικά κυκλώματα και μάλιστα εύκολα βλέπουμε ότι $EXP \not\subseteq subEXPSIZE$. Με μία πιο προσεκτική διατύπωση του παραπάνω αλγορίθμου μπορούμε να «κατεβάσουμε» αυτό το συμπέρασμα στην κλάση $EXP^{NP^{NP}}$.

Πρόταση 4.2.3. $EXP^{NP^{NP}} \not\subseteq SIZE[subEXP]$

Απόδειξη. Πράγματι έστω $R(x)$ το κατηγόρημα που γίνεται αληθές όταν υπάρχει συμβολοσειρά y αλφαριθμητικά μικρότερη του x για την οποία δεν υπάρχει υποεκθετικό κύκλωμα (ισοδύναμα κάθε κύκλωμα υποεκθετικού μεγέθους έχει πίνακα αληθείας που διαφέρει σε τουλάχιστον ένα δυφίο από τη y). Το $R(x)$ όταν το x είναι πίνακας αληθείας εύκολα βλέπουμε ότι ανήκει στο NP^{NP} κι άρα αρκεί να αρχίσουμε από την $x = 10^{2^n-1}$ και με διαδοχικά ερωτήματα στο R να βρούμε με δυαδική αναζήτηση την ελάχιστη συμβολοσειρά που δεν επιδέχεται συμπύκνωση από υποεκθετικό κύκλωμα. Από αυτήν εύκολα μπορούμε να φτιάξουμε ένα $EXP^{NP^{NP}}$ κατηγόρημα που επιστρέφει 1 όταν το j-οστό ψηφίο αυτής της μοναδικής συμβολοσειράς είναι 1 και το οποίο ασφαλώς δεν θα επιδέχεται πολυωνυμικού κυκλώματος. ■

Σημείωση. Στην πραγματικότητα, η παραπάνω απόδειξη μπορεί να χρησιμοποιηθεί αυτούσια, ώστε να εφαρμοστεί σε γενικότερες κλάσεις αλλά και με μεγαλύτερη λεπτότητα. Συγκεκριμένα αν $\mathcal{C}$ μία κλάση (με $\mathcal{C} \subseteq EXP$) για την

οποία ισχύει ότι $DTIME[g(n)] \subseteq \mathcal{C} \Rightarrow DTIME[g(n^c)] \subseteq \mathcal{C}$ (για κάθε $c > 0$ και ότι υπάρχει τουλάχιστον μία τέτοια $g(n) > n$ με $DTIME[g(n)] \subseteq \mathcal{C}$), τότε $\mathcal{C}^{NP^{NP}} \not\subseteq SIZE[f(n)]$, για οποιαδήποτε f με $DTIME[f(n)] \subseteq \mathcal{C}$.

Παρατηρούμε ότι ο μόνος λόγος που τοποθετήσαμε το κατηγόρημα R σε μορφή μαντείου (και δε πήραμε αντ΄ αυτού κατ΄ ευθείαν τη κλάση $NEXP^{NP}$) είναι για να εξασφαλίσουμε ότι παίρνουμε μια μοναδική δύσκολη συμβολοσειρά (καθώς σε ένα κύκλωμα θα έχουμε πρόσβαση σε ένα δυφίο της μόνο κάθε φορά) και να αποφύγουμε το πρόβλημα που συναντήσαμε στη μη τετριμμένη ύπαρξη συμπυκνώσιμων πιστοποιητικών για το $NEXP$, δηλαδή την αναφορά σε διαφορετικά πιστοποιητικά για κάθε δυφίο. Για παράδειγμα, αν ο παραπάνω αλγόριθμος έβρισκε απλά ένα x το οποίο να μην έχει υποεκθετικό κύκλωμα (έκφραση που ανήκε στο Σ_2^{EXP}) τότε ένα κύκλωμα που επιστρέφει διαρκώς 1, θα ήταν σωστό, αφού είναι τετριμμένο ότι πάντα υπάρχει μία δύσκολη συμβολοσειρά η οποία να έχει 1 στο j-οστό της ψηφίο. Η ανάγκη λοιπόν της αναφοράς σε μία κοινή δύσκολη συμβολοσειρά έχει επιτρέψει μέχρι στιγμής να δείξουμε μόνο ότι $NEXP^{NP} \not\subseteq SIZE[f(n)]$ όπου f μια half-exponential συνάρτηση, δηλαδή τέτοια ώστε $f(f(n)) \leq 2^{poly(n)}$ (το οποίο περιλαμβάνει συναρτήσεις όπως τις $n^{a(n)}$, $2^{logn^{logn^{logn}}}$ αλλά και αρκετά ψηλότερες, όχι όμως συναρτήσεις σαν τις $2^{\sqrt{n}}$ ή $2^{n^{(1/loglogn)}}$).

Πρόταση 4.2.4 ([Kan82]). *$NEXP^{NP} \not\subseteq SIZE[f(n)]$ για οποιαδήποτε half-exponential συνάρτηση f.*

Απόδειξη. Έστω ότι δεν ισχύει κι έστω f^* η αντίστοιχη half-exponential συνάρτηση που το ικανοποιεί και $f = \omega(f^*)$ μια οποιαδήποτε άλλη half-exponential συνάρτηση. Έχουμε πρώτα από όλα από τη Σημείωση του Θεωρήματος 4.2.3 ότι $DTIME[f(n)]^{NP^{NP}} \not\subseteq SIZE[f(n^c)]$ για κάθε $c > 0$. Έχουμε επίσης ότι το $DTIME[f(n)]^{NP^{NP}}$ είναι υποσύνολο του $NTIME[f(n)]^{NP^{NP}}$, κι άρα για μια γλώσσα που ανήκει σε αυτή τη κλάση, έχουμε ότι υπάρχει ένα (χ.β.τ.γ. γραμμικού χρόνου) κατηγόρημα R τέτοιο ώστε $x \in L \Leftrightarrow \exists y \forall z \exists w R(x, y, z, w)$ με $|y|, |z|, |w| \leq f(n)$. Δοθέντων όμως των x, y, z μένει να αποφασίσουμε αν $\exists w R(x, y, z, w)$ κι επειδή από υπόθεση προκύπτει άμεσα $NP \subseteq SIZE[f(n)]$ έχουμε ότι υπάρχει κύκλωμα μεγέθους $O(f(n))$ τέτοιο ώστε για κάθε ικανοποιήσιμη λογική έκφραση μεγέθους n να επιστρέφει (σε n εξόδους) μια τέτοια αποτίμηση που την ικανοποιεί. Το $\exists w R(x, y, z, w)$ μπορεί να επιλυθεί από ένα κύκλωμα μεγέθους $f(f(n))$ (δεδομένου ότι η είσοδος $\langle x, y, z\rangle$ είναι $O(f(n))$) κι άρα (όμοια με την απόδειξη του Θεωρήματος 4.2) το πρόβλημα μπορεί να γραφτεί στη μορφή $\exists \langle C_{fof}, y\rangle \forall z R(x, y, z, C_{fof}(x, y, z))$ κι άρα έχουμε $NTIME[f(n)]^{NP^{NP}} \subseteq NTIME[O(f(f(n)))]^{NP}$. Επομένως εφόσον η f είναι half-exponential και από υπόθεση έχουμε $NEXP^{NP} \subseteq SIZE[f^*(n)]$, καταλήγουμε εν τέλει ότι $DTIME[f(n)]^{NP^{NP}} \subseteq SIZE[f^*(n)]$ για οποιαδήποτε half-exponential f, το οποίο όπως είδαμε από την παραπάνω Σημείωση δεν αληθεύει. ■

Η πιο «χαμηλή» κλάση που γνωρίζουμε μέχρι στιγμής ότι δεν ανήκει στο $P_{/poly}$ είναι η $MAEXP$ [BFT98] και η απόδειξη σχετίζεται με αυτή του Θεωρήματος 4.6, καθώς όπως φάνηκε εκεί ο συνδυασμός μη ντετερμινισμού με τυχαιότητα (όπως ακριβώς έχει η κλάση $MAEXP$) οδηγεί στον αποκλεισμό εύκολων κυκλωμάτων. Από εκεί και πέρα ανοιχτά παραμένουν τα ερωτήματα (ως προς την ύπαρξη γενικών πολυωνυμικών κυκλωμάτων) για πιο χαμηλές κλάσεις όπως οι $EXP, NEXP$ αλλά ακόμη και για την EXP^{NP} και όπως θα δούμε στην επόμενη ενότητα ακόμη λιγότερο ελπιδοφόρα είναι τα συμπεράσματα που αφορούν μη εκθετικές κλάσεις, όπως την $QuasiP$ ή την NP, $PSPACE$ κ.ο.κ. Το πιο πρόσφατο επίτευγμα στη προσπάθεια απόδειξης ανομοιόμορφων κάτω φραγμάτων έγινε μόλις πριν λίγα χρόνια κι αφορά τη κλάση $NEXP$ με την ACC^0, τη πλήρη απόδειξη του οποίου θα δούμε στο επόμενο κεφάλαιο.

4.3 Φυσικές Αποδείξεις

Οι αποδείξεις που είδαμε μέχρι στιγμής περιέχουν έντονα στοιχεία διαγωνιοποίησης. Συγκεκριμένα, δείξαμε κάτω φράγματα για (υπερ-)εκθετικές κλάσεις, στις οποίες αυτό που κάναμε ήταν να απαριθμούμε όλα τα πιθανά εύκολα κυκλώματα και να παίρνουμε έναν πίνακα αληθείας που δεν προκύπτει από κανένα από όλα αυτά. Η μέθοδος αυτή φαίνεται να λειτούργησε (εν μέρει) για κλάσεις οι οποίες μπορούσαν να χωρέσουν την απαρίθμηση όλων αυτών, αλλά δε φαίνεται να μπορούν να δώσουν αποτελέσματα για μικρότερες κλάσεις όπως την NP, τη $PSPACE$ ή ακόμη και για την EXP.

Η επόμενη εύλογη προσέγγιση λοιπόν είναι να δείξουμε ότι κάποιες υψηλές υπολογιστικές κλάσεις δεν έχουν εύκολα κυκλώματα, βρίσκοντας μια ιδιότητα, η οποία να είναι πολύ δύσκολη για να μπορούν να την έχουν εύκολα κυκλώματα, αλλά να τυχαίνει να την έχουν οι υψηλές κλάσεις που θέλουμε να αποκλείσουμε από αυτά. Όπως θα δούμε και αναλυτικότερα στη συνέχεια, αυτή ήταν και η μεθοδολογία που ακολουθήσαμε για να δείξουμε ότι το P δεν είναι στο AC^0· το P μπορούσε να υπολογίσει αν μια συμβολοσειρά έχει άρτιο πλήθος άσσων, μια ιδιότητα η οποία φάνηκε να είναι πολύ δύσκολη για να μπορεί να ελεγχθεί από ένα AC^0 κύκλωμα. Προκύπτει λοιπόν το ερώτημα, αν μπορούμε να βρούμε μια συνάρτηση αντίστοιχη της αρτιότητας (και ασφαλώς αρκετά πιο δύσκολη από αυτή), τέτοια ώστε να είναι υπολογίσιμη στο NP ή στο EXP, αλλά με ανάλυση των πολυωνυμικών κυκλωμάτων να προκύπτει ότι αυτή δε μπορεί να υπολογιστεί από κανένα τέτοιο (καταλήγοντας αντίστοιχα ότι $NP \nsubseteq P_{/poly} \Rightarrow NP \neq P$ και $EXP \nsubseteq P_{/poly} \Rightarrow BPP \subseteq subEXP$). Μια τέτοια απόδειξη, ακριβώς λόγω της φυσικότητας της ιδέας της (αλλά και των πολλών επιτυχιών που είχε στην απόδειξη κάτω φραγμάτων για χαμηλότερες κυκλωματικές κλάσεις), ονομάστηκε από τους Razborov και Rudich φυσική απόδειξη [RR97]. Δυστυχώς, στο ίδιο άρθρο που έδωσαν αυτή την ονομασία, έδειξαν κιόλας ότι κάτω από ορισμένες εύλογες συνθήκες, αυτές οι αποδείξεις δεν δύνανται να ξεχωρίσουν το NP (και αντίστοιχες κλάσεις) από το $P_{/poly}$.

4.3.1 Ορισμός Φυσικών Αποδείξεων

Η προτυποποίηση ενός είδους αποδείξεων με τον τυπικό ορισμό θα ήταν αφενός δύσκολη καθώς θα έπρεπε να οριστεί πάνω σε ένα απλό σύστημα αξιωμάτων και κανόνων και αφετέρου δύσχρηστη καθώς είτε θα ήταν πολύ περιορισμένος ο ορισμός (με μικρό πλήθος εφαρμογών), είτε τόσο αφηρημένος που θα έκανε πολύ δύσκολο τον έλεγχο για το αν μια κλασική (υψηλού επιπέδου) απόδειξη ανήκει σε αυτή την κατηγορία. Αντ΄ αυτού, λοιπόν, θα ορίσουμε ως φυσικές αποδείξεις αυτές που χρησιμοποιούν κάποια φυσική συνδυαστική ιδιότητα (ο ακριβής ορισμός των οποίων δίνεται παρακάτω). Ας παρατηρηθεί ότι ακόμη κι αυτός ο ορισμός (της χρήσης κάποιας ιδιότητας από μία απόδειξη) δεν είναι καθόλου αυστηρός μαθηματικά, αλλά εν προκειμένω τα αντίστοιχα συμπεράσματα δεν επηρεάζονται, καθώς όσα θεωρήματα ακολούθως αναφέρονται καταχρηστικά σε ύπαρξη φυσικών αποδείξεων, μπορούν να αντικατασταθούν από την ύπαρξη φυσικών συνδυαστικών ιδιοτήτων.

Οι εν λόγω ιδιότητες, αναφέρονται σε συναρτήσεις και δεδομένου ότι θα χρησιμοποιηθούν για την απόδειξη κάτω φραγμάτων σε κυκλωματικές συναρτήσεις, αφορούν τους πίνακες αληθείας αυτών. Έτσι για n εισόδους έχουμε ότι το C_n είναι ένα υποσύνολο του $F_n = \{f_n : \{0,1\}^n \to \{0,1\}\}$ ή ισοδύναμα των $\{0,1\}^{2^n}$ συμβολοσειρών (μια ιδιότητα λοιπόν απαρτίζεται από τα $C = \cup_{n=0}^{\infty} C_n$). Όπως έχει φανεί από τα παραπάνω, θέλουμε αυτή η ιδιότητα να αφορά συναρτήσεις που δε μπορούν να περιγραφούν από πολυωνυμικά κυκλώματα ή αλλιώς να είναι χρήσιμη κατά του $P_{/poly}$.

Ορισμός 4.3.1. *(Χρησιμότητα)* Λέμε ότι η ιδιότητα C_n είναι χρήσιμη κατά της κυκλωματικής κλάσης $P_{/poly}$ αν οποιαδήποτε ακολουθία συναρτήσεων $f_1, f_2, ...f_n, ...$ με $f_n \in C_n$ (για άπειρα n) δεν ανήκει στο $P_{/poly}$.

Ασφαλώς, όπως είναι αναμενόμενο, οι παραπάνω ιδιότητες πρέπει να περιοριστούν ακόμη περισσότερο εφόσον η μη ύπαρξη μιας τέτοιας γενικής ιδιότητας στο NP θα σήμαινε ανυπαρξία απόδειξης ότι $NP \nsubseteq P_{/poly}$ (στο τρέχον αξιωματικό σύστημα) το οποίο θα ήταν αποτέλεσμα πολύ σπουδαιότερο. Δεδομένου μάλιστα ότι το άτοπο που προαναγγείλαμε θα προκύψει από σύγκρουση με μια κρυπτογραφική υπόθεση, χρειαζόμαστε τους ακόλουθους επιπλέον περιορισμούς (οι οποίοι θα σταθούν καίριας σημασίας για αυτό). Προτού προχωρήσουμε σε αυτούς όμως, να σημειωθεί ότι ο παραπάνω ορισμός μπορεί να χρησιμοποιηθεί αυτούσιος για τον χαρακτηρισμό αποδείξεων χρήσιμων κατά οποιασδήποτε κυκλωματικής κλάσης $\mathcal{C}$, απλώς αντικαθιστώντας στον παραπάνω ορισμό το $P_{/poly}$ με αυτήν. Ορίζουμε, λοιπόν, επιπλέον:

Ορισμός 4.3.2. *(Κατασκευασιμότητα)* Λέμε ότι μια ιδιότητα C_n είναι κατασκευάσιμη αν ο έλεγχος $f_n \in C_n$ μπορεί να γίνει στο P (με είσοδο $|f_n|$, δηλαδή σε χρόνο $2^{O(n)}$).

Ορισμός 4.3.3. *(Πληθωριστικότητα)* Λέμε ότι μια ιδιότητα C_n είναι πληθωρική αν ένα μη αμελητέο ποσοστό όλων των συναρτήσεων $\{0,1\}^{2^n}$ την έχει,

συγκεκριμένα αν

$$\frac{|C_n|}{|F_n|} \geq \frac{1}{poly(N)}$$

όπου $N = 2^n$.

Εκ πρώτης όψεως, οι παραπάνω περιορισμοί μπορεί να φαίνονται πολύ περιοριστικοί ή επιτηδευμένοι, εν τούτοις περικλείουν μια μεγάλη οικογένεια αποδείξεων και συγκεκριμένα, σχεδόν όλες τις συνδυαστικές τεχνικές που έχουν δώσει κυκλωματικά κάτω φράγματα. Ακόμη και σε περιπτώσεις που αυτή η ιδιότητα δεν είναι ευκρινώς ορισμένη ή η αντίστοιχη κυρίαρχη συνδυαστική ιδιότητα δε πληροί μια από τις παραπάνω ιδιότητες, έχει προκύψει ότι εμπεριέχει ως υποσύνολο μια τέτοια κατασκευάσιμη και πληθωρική ιδιότητα.

Η κατασκευασιμότητα περιορίζει ασφαλώς τις επιτρεπτές ιδιότητες αναγκάζοντας τις να είναι πεπερασμένα ελέγξιμες, τους δίνει ωστόσο αρκετή υπολογιστική ισχύ, συγκεκριμένα πολυωνυμική ως προς το μέγεθος του TT, το οποίο (δεδομένου ότι θέλουμε να αναφερθούμε σε συνδυαστικές ιδιότητες πολυωνυμικών κυκλωμάτων n εισόδων) είναι αρκετό για σχεδόν όλες τις συνδυαστικές αποδείξεις (μέχρι την εποχή που γράφτηκε το αντίστοιχο άρθρο). Να σημειωθεί ότι αν αλλάξουμε στον παραπάνω ορισμό τη κλάση P με οποιαδήποτε άλλη Γ παίρνουμε τον ορισμό των Γ-φυσικών αποδείξεων (όταν λείπει ο χαρακτηρισμός υπονοείται $\Gamma = P$).

Η πληθωριστικότητα μπορεί αρχικά να φαίνεται ως μια παράταιρη ιδιότητα, η οποία ενδεχομένως να περιορίσει πολύ τις ιδιότητες που θα μελετήσουμε, εν τούτοις η απαίτηση να περιέχει μεγάλο ποσοστό των συναρτήσεων, συμβαδίζει με την ιδέα ενός λογικού μέτρου κυκλωματικής δυσκολίας (βλ. [RR97, s.5]). Πράγματι διαισθητικά, για οποιοδήποτε μέτρο το οποίο αυξάνει αναλογικά με το πλήθος των πυλών, αν αυτό δίνει μεγάλη δυσκολία για μια συνάρτηση f_1, θα πρέπει να δίνει μεγάλη δυσκολία (τουλάχιστον τη μισή) και για την $f_1 \oplus f_2$ ή για την f_2, αφού μία πύλη $\oplus$ μεταξύ αυτών των δύο δίνει ένα κύκλωμα με μέτρο το άθροισμα των δύο επιμέρους (των $f_1 \oplus f_2$ και f_2), λόγω της αναλογικότητας του μέτρου, και που ισούται (σημασιολογικά) με την δύσκολη f_1. Άρα δεδομένου ότι στο παραπάνω (μη αυστηρό) παράδειγμα η f_2 είναι τυχαία συνάρτηση (οπότε το ίδιο ισχύει και για την $f_1 \oplus f_2$), προκύπτει ότι αν ένα αναλογικό μέτρο δίνει δυσκολία για μία συνάρτηση, οφείλει να δίνει αντίστοιχη δυσκολία για σχεδόν τις μισές από αυτές, ικανοποιώντας την πληθωριστικότητα. Να σημειωθεί και εδώ, ότι μπορούμε να αλλάξουμε τη πυκνότητα της ιδιότητας με οποιοδήποτε π_n επιθυμούμε, αλλά για εδώ θα περιοριστούμε στις περιπτώσεις όπου είναι $\frac{1}{2^{O(n)}}$. Σε κάθε περίπτωση, μέσω των παραπάνω τριών ιδιοτήτων που πρέπει να πληροί ταυτόχρονα μια ιδιότητα ώστε να θεωρηθεί φυσική, έχουμε ορίσει καλώς την έννοια των Γ-φυσικών αποδείξεων κατά της κλάσης $\mathcal{C}$ (πυκνότητας π_n).

Έχουμε ήδη δει μία περίπτωση φυσικής απόδειξης και δεν είναι άλλη από την απόδειξη ότι το $PARITY$, η γλώσσα δηλαδή που αποδέχεται όσες συμβολοσειρές έχουν άρτιο πλήθος άσσων, δεν ανήκει στο AC^0. Η κυρίως ιδέα που χρησιμοποιήθηκε ήταν ότι όποτε σε ένα AC^0 κύκλωμα σταθεροποιούμε ένα

μεγάλο ποσοστό των δυφίων εισόδου (έστω k), τότε το προκύπτον κύκλωμα είναι σταθερό με θετική πιθανότητα (και επειδή η ιδιότητα αυτή δε συμβαδίζει με το $PARITY$ για προφανείς λόγους, προέκυψε ο διαχωρισμός του P από το AC^0). Έστω λοιπόν C_n η ιδιότητα που έχει όλες τις συμβολοσειρές για τις οποίες όταν σταθεροποιούμε k τυχαία δυφία δεν προκύπτει σταθερή συνάρτηση με θετική πιθανότητα.
Χρησιμότητα: Η παραπάνω ιδιότητα είναι προφανώς χρήσιμη κατά του AC^0, καθώς από την ανάλυση που προέκυψε στην απόδειξη του Θεωρήματος 3.5, τα AC^0 κυκλώματα σταθεροποιούνται με θετική πιθανότητα σταθεροποιώντας k τυχαία δυφία.
Κατασκευασιμότητα: Το αν $f_n \in C_n$ μπορεί να ελεγχθεί σε πολυωνυμικό χρόνο (ως προς 2^n), καθώς απλώς παίρνουμε όλες τις 2^k σταθεροποιήσεις από $\binom{n}{k}$ συνδυασμούς (συνολικού πλήθους $2^k * \binom{n}{k} < 3^n = 2^{O(n)}$) και για κάθε μία ελέγχουμε αν η προκύπτουσα συνάρτηση είναι σταθερή (προφανώς σε χρόνο το πολύ $2^{O(n)}$). Αν βρεθεί έστω και ένας συνδυασμός για τον οποίον να ισχύει αυτό (θετική πιθανότητα), τότε απορρίπτουμε, αλλιώς αποδεχόμαστε (σε συνολικό χρόνο $poly(2^n)$).
Πληθωριστικότητα: Μένει να δείξουμε ότι οι περισσότερες συναρτήσεις την έχουν αυτή την ιδιότητα. Πράγματι, κάθε συνάρτηση που δεν την έχει, έχει τουλάχιστον μία σταθεροποίηση για την οποία δεν ισχύει. Επομένως, επειδή έχουμε σταθεροποίηση στα $n-k$ εναπομείναντα δυφία, προκύπτει ότι από τις 2^n θέσεις, υπάρχουν μόλις δύο επιλογές για τις 2^{n-k} (σταθερά 0 ή σταθερά 1) κι άρα υπάρχουν το πολύ $2^{2^n-2^{n-k}+1}$ τέτοιες απονομές (αφότου έχουν καθοριστεί τα k δυφία σταθεροποίησης). Συνολικά δεν μπορούν να υπάρχουν περισσότερες από $2^{2^n-2^{n-k}+1} * 2^k * \binom{n}{k} < 2^{2^n-2^{n-k}+5n}$ τέτοιες συναρτήσεις κι άρα το ποσοστό των συναρτήσεων που έχουν την ιδιότητα C_n προς όλες είναι τουλάχιστον $1 - 2^{-(2^{n-k-1})} >> \frac{1}{2}$.

Συγκεκριμένα ο έλεγχος της C_n μπορεί να γίνει απόλυτα παραλληλοποιήσιμος σε σταθερό βάθος, καθώς ο έλεγχος για το αν απομένει μία σταθερή συνάρτηση για δεδομένη σταθεροποίηση κάποιων δυφίων μπορεί να γίνει με κάποιο τετριμμένο κύκλωμα πολυωνυμικού μεγέθους (ως προς το 2^n) και σταθερού βάθους. Παίρνοντας την τομή όλων αυτών των $2^k * \binom{n}{k}$ κυκλωμάτων προκύπτει ένα AC^0 κύκλωμα που ελέγχει αν μια δοθείσα συνάρτηση έχει την εν λόγω ιδιότητα, οπότε κατά βάση επρόκειτο για μία AC^0-φυσική απόδειξη κατά του AC^0.

Πέραν τούτου του παραδείγματος, υπάρχει πληθώρα άλλων παραδειγμάτων φυσικών αποδείξεων (βλ. [RR97, s.3]), όπως αυτή που ξεχωρίζει το $ACC(q)$ από το $ACC(p)$ (για p, q πρώτους) αλλά και για διάφορα άλλα προϋπήρχοντα αποτελέσματα που αφορούσαν το ελάχιστο μέγεθος της φόρμουλας μιας συγκεκριμένης συνάρτησης κ.ο.κ. τα οποία μπορούν να βρεθούν αναλυτικά στο αντίστοιχο άρθρο (αλλά και σε επόμενα που ακολούθησαν και έδειξαν τη μεγάλη πληθώρα αποδείξεων που φυσικοποιούνταν).

Υπάρχουν ωστόσο και ορισμένες κατηγορίες αποδείξεων, οι οποίες δεν είναι φυσικές και συγκεκριμένα αυτές που είδαμε σε αυτό το κεφάλαιο και τα οποία συνδέονται στενά με μετρητικά επιχειρήματα κι επιχειρήματα διαγωνιοποίησης. Πράγματι η κυρίαρχη ιδιότητα σε αυτές τις αποδείξεις ήταν αυτή που δεχόταν όσες συμβολοσειρές δεν επιδέχονται πολυωνυμικό κύκλωμα. Προφανώς πληρούν την χρησιμότητα και πληθωριστικότητα (για τους λόγους που έχουμε δει ενδελεχώς στα προηγούμενα), αλλά δεν πληρούν την κατασκευασιμότητα, καθώς, τουλάχιστον μέχρι τώρα, δε γνωρίζουμε κάποιον αλγόριθμο που να αποδέχεται μόνο μη πολυωνυμικές συμβολοσειρές (ο τετριμμένος αλγόριθμος εξερευνά όλα τα πολυωνυμικά κυκλώματα (για κάθε πολυώνυμο) κι άρα απαιτεί χρόνο $2^{n^{\omega(1)}}$).

Με ανάλογη σκοπιά, όσες αποδείξεις χρησιμοποιούσαν ένα επιχείρημα διαγωνιοποίησης, πληρούσαν την κατασκευασιμότητα, αλλά όχι την πληθωριστικότητα καθώς αναφέρονταν κάθε φορά σε μία συγκεκριμένη συνάρτηση που επιτελούσε συγκεκριμένη λειτουργία. Για παράδειγμα αν η ιδιότητα αφορούσε τις συναρτήσεις που επιλύουν σωστά το SAT, έχουμε την κατασκευασιμότητα μέσω του προφανούς τετριμμένου αλγόριθμου (εκμεταλλεύοντας το γεγονός ότι η είσοδος είναι μεγέθους 2^n) αλλά όχι την πληθωριστικότητα εφόσον μόνο η συγκεκριμένη συνάρτηση την ικανοποιεί. Για λόγους πληρότητας, ας αναφερθεί απλώς ότι η πληθωριστικότητα φαίνεται να είναι και ο λόγος που επίσης δε φυσικοποιείται η μεγάλη κατηγορία των αποδείξεων που αναφέρονται σε μονότονα κυκλώματα [RR97, s.2].

4.3.2 Φράγματα και ανυπαρξία Φυσικών Αποδείξεων

Σε κάθε περίπτωση, προσπερνώντας τις προφανείς μεθόδους διαγωνιοποίησης, οι φυσικές αποδείξεις φαίνονταν να αποτελούν την μητέρα των μεθόδων που προσπαθούν με συνδυαστικά επιχειρήματα (κάποιων εύλογων ορίων) να διαχωρίσουν μια κλάση από μία κυκλωματική κλάση και μέχρι και την έκδοση του αντίστοιχου άρθρου, μία τέτοια προσέγγιση θεωρούνταν ένας καλός υποψήφιος για την απόδειξη του $P \neq NP$ (συγκεκριμένα του $NP \nsubseteq P_{/poly}$). Εν τούτοις, από ό,τι δείχθηκε, μια τέτοια φυσική συνδυαστική ιδιότητα που ξεχωρίζει εύκολες από δύσκολες συναρτήσεις υπονοεί αυτόματα και την ύπαρξη ενός αλγορίθμου που ξεχωρίζει τυχαίες από ψευδοτυχαίες συμβολοσειρές, αποκλείοντας την ύπαρξη κρυπτογραφικά πολύ δύσκολων συναρτήσεων (μια εικασία αρκετά πιο ισχυρή από τη SETH).

Συγκεκριμένα, από την ανάλυση που είδαμε στην ενότητα για την αποτυχαιοποίηση του BPP, για να έχουμε αποδοτικές κρυπτογραφικές συναρτήσεις, χρειαζόμαστε την εύκολη παραγωγή ισχυρά ψευδοτυχαίων συμβολοσειρών. Παρότι εκεί επιτρέψαμε στις γεννήτριες να διαθέτουν εκθετικό χρόνο για την παραγωγή αυτών, προφανώς δεν είναι αυτή η περίπτωση για τις πρακτικές εφαρμογές, όπου το πολύ πολυωνυμικός χρόνος είναι επιτρεπτός (με αντίστοιχη σμίκρυνση στο μέγεθος της παραγόμενης συμβολοσειράς). Τα παραπάνω μπορούν να τυποποιηθούν με τον παρακάτω ορισμό:

Ορισμός 4.3.4. *(Δυσκολία Αποδοτικής Ψευδοτυχαίας Γεννήτριας)* Έστω μία ψευδοτυχαία γεννήτρια $G_k : \{0,1\}^k \to \{0,1\}^{2k}$ πολυωνυμικού χρόνου. Ορίζουμε ως δυσκολία της, $H(G_k)$, το ελάχιστο S, για το οποίο υπάρχει κύκλωμα C μεγέθους S, τέτοιο ώστε

$$|\Pr[C(\boldsymbol{x_{2k}})] - \Pr[C(G_k(\boldsymbol{x_k}))]| \geq \frac{1}{S}$$

(όπου οι πιθανότητες ορίζονται πάνω στις ομοιόμορφες τυχαίες μεταβλητές επί του $\{0,1\}^{2k}$ και $\{0,1\}^k$ αντίστοιχα, οι οποίες συμβολίζονται κλασικά με έντονα).

Όπως και προηγουμένως δηλαδή, η δυσκολία της ψευδοτυχαίας γεννήτριας εξαρτάται από το πόσο μεγάλο είναι το ελάχιστο κύκλωμα που μπορεί να ξεχωρίσει μια αληθινά τυχαία συμβολοδειρά από μία που παράγεται ψευδοτυχαία από την εν λόγω γεννήτρια. Παρότι δεν υπάρχει ακόμη σχετική απόδειξη, μια πολύ ευρεία εικασία (που στηρίζεται πέραν των άλλων σε λόγους παρόμοιους λοιπών ισχυρών εικασιών, όπως ότι $P \neq NP$ και $EXP \nsubseteq P_{/poly}$) είναι ότι υπάρχουν μέχρι και εκθετικά δύσκολες πολυωνυμικές ψευδοτυχαίες γεννήτριες (δηλαδή τέτοιες ώστε $H(G_k) = \Omega(2^{k^a})$). Συγκεκριμένα για τη κλασική γεννήτρια του διακριτού λογάριθμου [BM84], η οποία χρησιμοποιείται ευρέως στην Κρυπτογραφία, εικάζεται ότι είναι $a = \frac{1}{3}$. Όπως είναι φυσικό, η κατάρριψη αυτής της εικασίας θα είχε αφενός ισχυρότατες επιπτώσεις στην Θεωρητική Κρυπτογραφία και την ύπαρξη μονόδρομων συναρτήσεων, αλλά δεν θα άφηνε ανεπηρεάστη και πληθώρα πρακτικών εφαρμογών του σύγχρονου κόσμου που στηρίζονται σε παρόμοιες (μέχρι σήμερα αναπόδεικτες) εικασίες.

Υπάρχει λοιπόν το ακόλουθο Θεώρημα που (δεχόμενοι την παραπάνω εικασία) αποκλείει την ύπαρξη φυσικών αποδείξεων κατά του $P_{/poly}$.

Θεώρημα 4.8 ([RR97])**.** *Εκτός κι αν κάθε $P_{/poly}$ οικογένεια ψευδοτυχαίων γεννητριών $\{0,1\}^k \to \{0,1\}^{2k}$ έχει υποεκθετική δυσκολία, δεν υπάρχει $P_{/poly}$-φυσική απόδειξη κατά του $P_{/poly}$.*

Απόδειξη. Έστω G μία τέτοια γεννήτρια k εισόδων κι έστω G_0, G_1, οι αντίστοιχες $\{0,1\}^k \to \{0,1\}^k$ συναρτήσεις για τις οποίες $G(x) \equiv G_0(x)G_1(x)$. Μπορούμε πλέον να ορίσουμε τη συνάρτηση $g(x)(y) = \#1.G_y(x)$, δηλαδή το πρώτο δυφίο της $G_y(x)$, η οποία ορίζεται ως η διαδοχική εφαρμογή των δύο υπογεννητριών κατ΄ αντιστοιχία με τα δυφία του y, συγκεκριμένα $G_y(x) \equiv G_{y_n} \circ ... \circ G_{y_1}(x)$. Έστω ότι το x είναι μεγέθους k και το y μεγέθους $n = k^a$. Επειδή κάθε G_i είναι στο $P_{/poly}$, προκύπτει εύκολα ότι η συνάρτηση $g(x)$ είναι επίσης στο $P_{/poly}$ για κάθε x.
Έστω τώρα ότι υπάρχει $P_{/poly}$-φυσική συνδυαστική ιδιότητα C_n κατά του $P_{/poly}$, τέτοια ώστε $C_n(f_n) = 0$ για κάθε f_n που δέχεται πολυωνυμικά κυκλώματα (για αρκετά μεγάλες τιμές του n). Μα όμως τότε $C_n(g(x)) = 0$ για οποιοδήποτε x και επειδή η C_n είναι πληθωρική, θα ισχύει $\Pr[C_n(\boldsymbol{f_n})] > 2^{-O(n)}$, οπότε έχουμε

$$|\Pr[C_n(\boldsymbol{f_n}) = 1] - \Pr[C_n(g(\boldsymbol{x})) = 1]| > 2^{-O(n)}$$

όπου όμως το C_n μπορεί να υπολογιστεί με $P_{/poly}$ κυκλώματα κι άρα μεγέθους επίσης $2^{O(n)}$. Το τελικό ζήτημα για να καταλήξουμε στο ζητούμενο είναι ότι εμείς χρειαζόμαστε μια τέτοια σχέση για τη G, την οποία όμως μπορούμε να εξάγουμε, διατηρώντας το φράγμα $S = 2^{O(n)}$, με τον τρόπο που παρατίθεται στο Παράρτημα Α΄.4.
Προκύπτει επομένως ότι η δυσκολία της G είναι $2^{O(n)} = 2^{O(k^a)}$. Επειδή, όμως, το a ήταν αυθαίρετο (κι άρα $H(G_k) < 2^{k^a}$ για κάθε a), εν τέλει προκύπτει ότι η G είναι πράγματι υποεκθετικής δυσκολίας. ■

Σημείωση. Στην πραγματικότητα, η παραπάνω απόδειξη μπορεί να επεκταθεί αυτούσια μέχρι και για $P_{/qPoly}$-φυσικές αποδείξεις κατά του $P_{/poly}$ (όπου $P_{/qPoly}$ οι γλώσσες που επιδέχονται οικογένειες κυκλωμάτων κβασιπολυωνυμικού μεγέθους (δηλαδή $n^{log^c n}$)).

Όπως φάνηκε από την παραπάνω απόδειξη, κρίσιμη για τη μη ύπαρξη φυσικών αποδείξεων ήταν η υπόθεση για την ύπαρξη πολύ δύσκολων ψευδοτυχαίων γεννητριών. Παρότι πρόκειται για μια γερή εικασία, βέβαια αποτελέσματα έχουμε ασφαλώς μόνο για κλάσεις για τις οποίες έχουμε αποδεδειγμένα δύσκολες συναρτήσεις. Ένα παράδειγμα τέτοιας κλάσης είναι το AC^0 για το οποίο έχουμε ήδη δει μια δύσκολη συνάρτηση (την $PARITY \in ACC^0(2)$) από την οποία μπορεί να κατασκευαστεί μια ψευδοτυχαία γεννήτρια αποδεδειγμένα επαρκής ώστε να αποκλείει την ύπαρξη AC^0-φυσικών αποδείξεων κατά του $ACC^0(2)$ [RR97, s.4].

Ακόμη και χωρίς την ύπαρξη αντίστοιχων άνευ συνθηκών δύσκολων ψευδοτυχαίων συναρτήσεων για το $P_{/poly}$, το γεγονός της εκτίμησης μιας τέτοιας εικασίας ως αληθινής και οι συνέπειες που συνεπάγει το παραπάνω Θεώρημα αποτελούν έναν πολύ καλό οδηγό για το ποιες κατευθύνσεις είναι πιο πιθανό να μην οδηγήσουν σε κάποιο αποτέλεσμα έναν ερευνητή (ειδικά αν πιστεύει στην παραπάνω εικασία). Παρότι, γενικά, θεωρήθηκε από την κοινότητα ως σημάδι ότι δεν είναι ο χώρος των κυκλωμάτων ο πλέον κατάλληλος για την απόδειξη ότι $P \neq NP$, εν τούτοις υπάρχει και η αντίθετη ερμηνεία ότι η παραπάνω απόδειξη απλώς έδωσε περισσότερες οδηγίες ως προς το ποιες πορείες πρέπει να ακολουθηθούν. Συγκεκριμένα, δεδομένου ότι μια συνθήκη χρησιμότητας είναι προφανώς απαραίτητη για μια τέτοια συνδυαστική ιδιότητα, θα πρέπει να αποκλειστεί είτε η πληθωριστικότητα είτε η κατασκευασιμότητα.

Ο αποκλεισμός της πληθωριστικότητας, θα σημαίνει ότι το μέτρο που θα χρησιμοποιηθεί είτε δεν θα είναι αναλογικό είτε δε θα εφαρμόζεται ορθά σε όλες τις (μη εύκολες) συναρτήσεις. Αυτό, διαισθητικά, υπονοεί ότι δε μπορεί να πρόκειται για ιδιότητα που στηρίζεται σε μεγάλο βαθμό σε απλές συνδυαστικές ιδέες που να συμβαδίζουν με μια παραδοσιακή ερμηνεία του μεγέθους ενός κυκλώματος. Από την άλλη ο αποκλεισμός της κατασκευασιμότητας θα σημαίνει ότι αφορά μια ιδιότητα των πινάκων αληθείας που δε μπορεί να ελεγχθεί σε πολυωνυμικό χρόνο (ως προς το μήκος τους), κάτι το οποίο με τη σειρά του υπονοεί ότι θα αφορά μια ιδιαίτερα πολύπλοκη ιδιότητα (υπερεκθετικής

δυσκολίας ως προς το μήκος των εισόδων) και της οποίας η περιπλοκότητα ιδεάζει αισθητά μεγαλύτερη από αυτή των παραδοσιακών τεχνικών.

Σε κάθε περίπτωση, παρότι τα κυκλώματα φάνηκαν να είναι ένα θεμελιωμένα ισχυρό εργαλείο για την απόδειξη διαφόρων ιδιοτήτων και συσχετίσεων των παραδοσιακών υπολογιστικών κλάσεων, εν τούτοις δε φαίνεται να έχουν τη δυνατότητα με ένα απλό ή/και μη εξειδικευμένο επιχείρημα να απαντήσουν στα μεγάλα ερωτήματα της Θεωρητικής Πληροφορικής, όπως το αν $P \neq NP$, καθώς δε φαίνεται να υπάρχει καν ένας εύκολος τρόπος χαρακτηρισμού των πολυωνυμικά συμπυκνώσιμων συναρτήσεων. Η δημοσίευση αυτού του αποτελέσματος έκανε ξεκάθαρο ότι χρειάζεται πλέον μια νέα προσέγγιση ώστε μια εφικτή απόδειξη τέτοιων Θεωρημάτων να είναι πιθανή και σίγουρα για πάνω από μια δεκαετία είχε απομακρύνει από την επιστημονική κοινότητα τις ελπίδες ότι αυτή θα προερχόταν από τον χώρο των Κυκλωμάτων. Την άποψη αυτή φαίνεται σταδιακά να άρουν ορισμένα πρόσφατα αποτελέσματα διαχωρισμού υπολογιστικών και κυκλωματικών κλάσεων, τα οποία ανήκουν στη λεγόμενη τρίτη γενιά αποδείξεων, την οποία και εξερευνούμε λεπτομερέστερα στο ακόλουθο κεφάλαιο.

Κεφάλαιο 5

Τρίτη Γενιά Αποδείξεων

Στο σημείο αυτό φαίνεται να έχουμε μία ακόμη «προειδοποίηση» της μαθηματικής πραγματικότητας ότι μία απλή τεχνική δεν είναι ικανή να απαντήσει σε μεγάλα ερωτήματα που αφορούν διαχωρισμό κλάσεων όπως αυτό του $P \neq NP$. Πράγματι η σχετικοποίηση μας έδειξε ότι απλά επιχειρήματα διαγωνιοποίησης που δεν εξειδικεύουν στο τρόπο που εκτελείται η μηχανή, αλλά απλά στηρίζονται στη σημασιολογική τους φύση δεν είναι ικανά να απαντήσουν στη σχέση του P με το NP αλλά και στη σχέση πολλών άλλων ζευγών κλάσεων που δεν έχουν τετριμμένο διαχωρισμό. Θα πρέπει, λοιπόν, σίγουρα να εξερευνήσουμε περαιτέρω τις επί μέρους πράξεις που κάνει ο υπολογισμός και να χρησιμοποιήσουμε περιοριστικές ιδιότητες της πεπερασμένης φύσης τους. Το προφανές βήμα που έγινε προς την κατεύθυνση αυτή, ήταν η εμβάθυνση σε αυτές τις ιδιότητες και ο «ευκολότερος» τρόπος για αυτό φάνηκε να είναι η μελέτη της διάταξης των στοιχειωδών πράξεων (πυλών) στη μορφή του ισοδύναμου κυκλώματος (δεδομένου ότι ένα από τα μεγαλύτερα πλεονεκτήματα, άλλωστε, των κυκλωμάτων είναι η ευκολία τυποποίησης (και σε πολλές περιπτώσεις απόδειξης) συνδυαστικών επιχειρημάτων). Ακόμη κι έτσι όμως, φάνηκε από τα προηγούμενα ότι ένα σχετικά απλό (παρότι αρκετά εκφραστικό) συνδυαστικό επιχείρημα που δεν εκμεταλλεύεται τις ιδιότητες και τη φύση της κλάσης που σκοπεύει να διαχωρίσει, παρά μόνο στηρίζεται σε κυκλωματικές παραμέτρους μεγέθους, δεν είναι πιθανό καν να υπάρχει κάτω από εύλογες κρυπτογραφικές υποθέσεις.

Επομένως μια καθαρά σημασιολογική ή μια καθαρά συντακτική προσέγγιση μάλλον δεν δύναται να δώσει ισχυρά αποτελέσματα κάτω φραγμάτων ή διαχωρισμού κλάσεων κι άρα το επιχείρημα που θα δώσει την τελική απάντηση (αν ποτέ δοθεί) θα περιέχει σχεδόν σίγουρα αυξημένη αποδεικτική πολυπλοκότητα. Τον νέο αυτό δρόμο φαίνεται να χαράζει μια νέα τεχνική (που ανήκει στη λεγόμενη τρίτη γενιά αποδείξεων) η οποία συνδυάζει τις δύο αυτές μεθόδους (με εγγυημένα αυξημένη πολυπλοκότητα) παίρνοντας στοιχεία από ένα μεγάλο μέρος του φάσματος της Θεωρίας Πολυπλοκότητας αλλά και πολλών συγγενών κλάδων.

5.1 Σημασιολογία και Σύνταξη

5.1.1 Γενικό Πλαίσιο

Είδαμε, σε προηγούμενα, ότι οι σχετικιστικές αποδείξεις προσπαθούν να φτιάξουν μια λογική αντίφαση στηριζόμενοι στις σημασιολογικές ιδιότητες μιας υπολογιστικής κατασκευής (η οποία αρκεί να μπορεί να προσομοιώνει τη λειτουργία μιας άλλης σε συγκρίσιμο χρόνο) και δεν εξαρτώνται καθόλου από τα περαιτέρω χαρακτηριστικά ή δυνατότητες του υπολογιστικού συστήματος. Πυρήνας όλων αυτών των αποδείξεων είναι το άτοπο που προκύπτει από την προς κατάρριψη υπόθεση μέσω ενός διαγωνιοποιητικού επιχειρήματος, το οποίο βασίζεται στη κατασκευή μιας μη αποφεύξιμης αντιφατικής αυτοαναφοράς.

Στις φυσικές αποδείξεις, απ΄ την άλλη, είχαμε προσπάθεια καθορισμού των δύσκολων συναρτήσεων από τον πίνακα αληθείας τους με ένα επιχείρημα που έπρεπε να αποφασίζει τη δυσκολία τους σε πολυωνυμικό (ως προς το μήκος του TT) χρόνο (κατασκευασιμότητα), να ορίζει ως δύσκολες ένα μη αμελητέο ποσοστό όλων των συναρτήσεων (πληθωριστικότητα) και προφανώς η δυσκολία να αντιστοιχεί σε μη ύπαρξη πολυωνυμικού κυκλώματος (χρησιμότητα). Δεδομένων κάποιων εύλογων κρυπτογραφικών υποθέσεων, όπως είδαμε, δεν υπάρχει κάποιο τέτοιο επιχείρημα. Ένα χαρακτηριστικό της σχετικής απόδειξης του Θεωρήματος 4.8, όμως, είναι ότι εν τέλει οι εμπλεκόμενες αρχικές δομές είναι κατά βάση ανομοιόμορφα κυκλώματα, και καταλήξαμε στο συμπέρασμα ανυπαρξίας της αντίστοιχης συνδυαστικής ιδιότητας χωρίς να αναφερθούν πουθενά οι υπολογιστικές κλάσεις τις οποίες θέλαμε να αποκλείσουμε από το $P_{/poly}$.

Αρχίζει λοιπόν να διαφαίνεται ένα κοινό μοτίβο στις δύο παραπάνω αποδεικτικές κατηγορίες. Κάθε μία εστιάζει είτε μόνο στη σημασιολογική, είτε μόνο στη συντακτική φύση των εμπλεκόμενων υπολογιστικών δομών. Είναι εύλογο, λοιπόν, να δοκιμαστεί η περίπτωση όπου έχουμε συνδυασμό αμφότερων σε μία ενιαία απόδειξη. Συγκεκριμένα, αντί να προσπαθούμε να δείξουμε την αδυναμία έκφρασης δύσκολων συναρτήσεων από κυκλώματα πολυωνυμικού μεγέθους μέσω συνδυαστικών επιχειρημάτων που στηρίζονται στο σχετικά μικρό τους μέγεθος, προσπαθούμε να εξάγουμε μια περιορισμένη εκφραστικότητα για την αντίστοιχη κυκλωματική οικογένεια, βασιζόμενοι στην ύπαρξη απαγορευτικά εύκολων αλγορίθμων για κάποια ιδιότητα που τους αφορά. Χρησιμοποιούμε, δηλαδή, την ύπαρξη ενός αλγορίθμου για κυκλώματα, ο οποίος υπολογίζει μια συγκεκριμένη ιδιότητα τους (π.χ. την ικανοποιησιμότητα τους) σε χρόνο για τον οποίο γνωρίζουμε ότι είναι μη εφικτός για το αντίστοιχο πρόβλημα μιας μεγαλύτερης κλάσης (διαχωρίζοντας έτσι τη κλάση αυτή από τέτοιες οικογένειες κυκλωμάτων). Το καίριο σημείο είναι ότι πλέον η ιδιότητα δεν είναι καθαρά συνδυαστική και δε στηρίζεται σε μέτρα μεγέθους, αλλά αντιθέτως εξαρτάται από έναν αλγόριθμο, δένοντας έτσι τη συντακτική με τη σημασιολογική πλευρά (ή αλλιώς την ανομοιομορφία των κυκλωμάτων με την ομοιομορφία ενός αλγόριθμου). Δηλαδή χρησιμοποιούμε τη σύνταξη για την παρασκευή ενός

πιο εύκολου αλγορίθμου συγκεκριμένης σημασιολογίας και μετά δημιουργούμε το άτοπο όχι πάνω στη συνδυαστική ιδιότητα, αλλά πάνω στην ύπαρξη ενός ομοιόμορφου αλγορίθμου για αυτήν (με άλλα λόγια εισάγουμε κρίσιμα στοιχεία διαγωνιοποίησης για ομοιόμορφους αλγορίθμους που προκύπτουν, όμως, από τεχνικά συνδυαστικά επιχειρήματα ανομοιόμορφης φύσης, συνδυάζοντας έτσι αμφότερα τα πεδία).

5.1.2 Αλγόριθμοι για Ιδιότητες Κυκλωμάτων

Η γενική μορφή αυτών των αλγορίθμων μπορεί να τυποποιηθεί ως ακολούθως (βλ. [Wil14]):

Έστω μια ιδιότητα $\mathcal{P}$ που αναφέρεται σε κυκλώματα.
Η είσοδος είναι η κωδικοποίηση ενός κυκλώματος C μιας κλάσης $\mathcal{C}$.
Ο αλγόριθμος αποδέχεται αν και μόνο αν το C έχει την ιδιότητα $\mathcal{P}$.

Ορισμένα χαρακτηριστικά παραδείγματα ιδιοτήτων είναι τα εξής (στα οποία θεωρούμε ότι η είσοδος είναι πάντα η κωδικοποίηση ενός κυκλώματος C της κλάσης $\mathcal{C}$):

- $\mathcal{C} - SAT$ (Satisfiability): Υπάρχει κάποια είσοδος x για το C, τέτοια ώστε $C(x) = 1$;
- $\mathcal{C} - CAPP$ (Circuit Approximation Probability Problem): Με επιπλέον είσοδο έναν αριθμό $b \in [0, 1]$ (ή θέτοντας το ως σταθερά), ισχύει ότι $|\Pr[C(\boldsymbol{x}) = 1] - b| < \epsilon$ (για κάποια σταθερά $\epsilon < 1/2$);
- $\mathcal{C} - MCSP$ (Minimum Circuit Size Problem): Με επιπλέον είσοδο έναν αριθμό S, υπάρχει ισοδύναμο του C κύκλωμα μεγέθους το πολύ S;

Γενικά, μας ενδιαφέρουν και πάλι οι περιπτώσεις όπου η ιδιότητα αναφέρεται στον πίνακα αληθείας της συνάρτησης του κυκλώματος και δεν εξαρτάται από δομικά στοιχεία του ίδιου του κυκλώματος εισόδου. Αυτό είναι αναμενόμενο, αλλά και αναγκαίο, εφόσον θέλοντας να διαχωρίσουμε τη κλάση $\mathcal{C}$ από μια άλλη (συνήθως υπολογιστική) κλάση, θα πρέπει να κάνουμε αναφορά στο μοναδικό κοινό τους στοιχείο, δηλαδή το TT.

Ένα άλλο τεχνικό σημείο είναι η αναγνώριση ότι το κύκλωμα ανήκει πράγματι στη κλάση $\mathcal{C}$. Αυτό ασφαλώς δεν είναι ζήτημα, όταν τα χαρακτηριστικά της κλάσης είναι ακριβείς (υπολογίσιμες) αριθμητικές συναρτήσεις π.χ. του πλήθους των δυφίων εισόδου του κυκλώματος. Το ζήτημα προκύπτει για τις γενικότερες κλάσεις που μελετάμε, όπως τις AC^0, ACC^0, NC_{non-u}, $P_{/poly}$, $subEXPSIZE$ κ.ο.κ. Και πάλι, ωστόσο, όλα τα συμπεράσματα που ακολουθούν συνδέουν τέτοιες γενικές κλάσεις (με την έννοια ότι π.χ. στο TC^0 έχουμε οικογένειες κυκλωμάτων με πύλες πλειοψηφίας σταθερού (αλλά διαφορετικού για κάθε οικογένεια) βάθους). Ακόμη και τότε, όμως, η μέθοδος που θα ακολουθούμε είναι ότι θα παίρνουμε δύο φραγμένα στιγμιότυπα των δύο κλάσεων

προς διαχωρισμό και θα έχουμε φροντίσει το γενικό αποδεικτικό επιχείρημα που θα έχουμε κατασκευάσει να είναι τέτοιο ώστε να λειτουργεί για κάθε συνδυασμό παραμέτρων, οδηγώντας εν τέλει στο γενικό αποκλεισμό μέσω της ύπαρξης ενός διαχωριστικού επιχειρήματος για κάθε ζεύγος υποκλάσεων.

Επίσης, όλες οι παραπάνω ιδιότητες έχουν έναν τετριμμένο αλγόριθμο εξαντλητικής αναζήτησης εκθετικού χρόνου. Το ενδιαφέρον προκύπτει όταν για κάποια από αυτές υπάρχει αλγόριθμος που να εκτελείται σε έστω και λίγο λιγότερο χρόνο. Τότε με μια κατάλληλη χρήση αυτών των κυκλωμάτων μπορούμε να χρησιμοποιήσουμε έναν αλγόριθμο που να επιλύει πιο γρήγορα ένα πρόβλημα, για το οποίο γνωρίζουμε ότι δεν μπορεί να λυθεί σε λιγότερο χρόνο για τα πλήρη στιγμιότυπα της κλάσης που θέλουμε να διαχωρίσουμε. Δηλαδή, πλέον το τελικό διαχωριστικό στοιχείο είναι οι άνισες σημασιολογικές ικανότητες (που σε παραδείγματα που ακολουθούν έχουν προκύψει από παραδοσιακή διαγωνιοποίηση), όπου όμως οι ικανότητες αυτές έχουν με τη σειρά τους προκύψει από συνδυαστικά επιχειρήματα συντακτικής φύσεως των κλάσεων επί των οποίων υποθέτουμε ότι μπορούν να μοντελοποιηθούν οι αρχικές, προς διαχωρισμό, κλάσεις.

Συνολικά έχουμε ότι το παραπάνω μοντέλο αποδείξεων, γλιτώνει το παράδειγμα των σχετικιστικών αποδείξεων καθώς ο εν λόγω αλγόριθμος στηρίζεται σε μια τεχνική και συνδυαστική ιδιότητα μιας συγκεκριμένης κυκλωματικής κλάσης (ιδιότητες οι οποίες κατά κόρον δεν σχετικοποιούνται) και επιπλέον φαίνεται να γλιτώνει και το παράδειγμα των φυσικών αποδείξεων καθώς καταλήγει στο τελικό συμπέρασμα κάνοντας χρήση ενός παραδοσιακού διαγώνιου επιχειρήματος (το οποίο επίσης δε συμβαδίζει με τις φυσικές αποδείξεις λόγω πληθωριστικότητας) Ακόμη περισσότερο, κάποιες από τις παραπάνω ιδιότητες δε φαίνεται να έχουν καν κάποιον αποδοτικό αλγόριθμο ως προς το αντίστοιχο μήκος του TT (εν τούτοις δεν είναι ακόμη γνωστό αν φυσικοποιούνται ορισμένες από αυτές τις αποδείξεις, καθώς όπως είπαμε τα φυσικά επιχειρήματα είναι συχνά περίπλοκα κωδικοποιημένα μέσα τους). Τα παραπάνω φαίνεται να αποτελούν ένα πολύ καλό στοιχείο ότι η νέα αυτή τεχνική δεν έχει να φοβηθεί κάτι από τα φαντάσματα που στοιχειώνουν τις δύο προηγούμενες μεγάλες κατηγορίες, ενώ ταυτόχρονα έχει τη δυνατότητα να συνδυάσει τα οφέλη και τα εργαλεία αμφότερων. Στις ενότητες που ακολουθούν θα δούμε ορισμένες συγκεκριμένες αποδείξεις αυτού του νέου παραδείγματος ενισχύοντας την παραπάνω άποψη.

5.2 Συμπυκνώσιμα Ερωτήματα στο $NEXP$

Προτού δούμε αυτές τις αποδείξεις, είναι πολύ χρήσιμο να αφιερώσουμε λίγο χρόνο να μελετήσουμε κάποια κρίσιμα χαρακτηριστικά της κλάσης $NEXP$, την οποία και αφορούν τα περισσότερα αποτελέσματα. Αν και θα φανεί λεπτομερέστερα παρακάτω, ο ένας λόγος που συμβαίνει αυτό είναι επειδή η $NEXP$ εμπεριέχει την EXP κι άρα σίγουρα όλες τις ενδιαφέρουσες κλάσεις που έχου-

με συναντήσει (όπως τις $subEXP$, NP, $PSPACE = IP$ κ.ο.κ.) και αποτελεί «καθαρή» κλάση με την έννοια ότι χαρακτηρίζεται έντονα από το χρονικό στοιχείο και έχει πλήρη αντιπρόσωπο μπορώντας έτσι να συμμετάσχει σε διαγωνιοποιητικά επιχειρήματα (σε αντίθεση π.χ. με την PH). Ο δεύτερος λόγος είναι ότι η $NEXP$ είναι εν τέλει η ελάχιστη κλάση που πληροί τα παραπάνω κριτήρια και ταυτόχρονα περιέχει μη ντετερμινισμό, καθώς οι ακόλουθοι αλγόριθμοι είναι κατά κύριο λόγο αλγόριθμοι ελέγχου «δυσκολίας» και όχι παραγωγής και στους οποίους τα εμπλεκόμενα κυκλώματα πρέπει να μαντεύονται.

Ένα πολύ χρήσιμο λήμμα που εφαρμόζεται σε ένα μεγάλο ποσοστό αυτών των αποδείξεων, είναι, λοιπόν, το ακόλουθο που μας εγγυάται ότι σε περίπτωση που το $NEXP$ επιδέχεται μοντελοποίηση από οικογένεια κυκλωμάτων της κλάσης $\mathcal{C}$, τότε της ίδιας δυσκολίας είναι και (τουλάχιστον ένα από) τα πιστοποιητικά που επαληθεύουν τα ικανοποιήσιμα στιγμιότυπα του προβλήματος. Να σημειωθεί ότι, στα ακόλουθα, εκτός κι αν αναφέρεται ρητά αλλιώς, θεωρούμε ότι η κλάση $\mathcal{C}$ αντιστοιχεί σε κάποια από τις γενικές κλάσεις $\mathcal{C} \subseteq P_{/poly}$ που έχουμε ορίσει στα αντίστοιχα κεφάλαια (π.χ. ACC^0, TC^0, NC^1_{non-u}, NC_{non-u}, κ.ο.κ. (γενικά κλάσεις άνω του AC^0, το οποίο άλλωστε το έχουμε ήδη αποκλείσει από το P με φυσικό επιχείρημα)). Πολλά από τα αποτελέσματα ασφαλώς ισχύουν και για άλλες ενδιάμεσες κλάσεις, όπως θα γίνει προφανές.

Λήμμα 5.0.1. *Έστω ότι $NEXP \subseteq \mathcal{C}$. Τότε για κάθε πρόβλημα στο $L \in NEXP$ και για κάθε ακολουθία ικανοποιήσιμων στιγμιότυπων του L υπάρχει ακολουθία πιστοποιητικών για τα αντίστοιχα στιγμιότυπα, τα οποία να έχουν $\mathcal{C}$-συμπύκνωση (δηλαδή τα TT τους να παράγονται από μια οικογένεια κυκλωμάτων που να ανήκει στο $\mathcal{C}$).*

Απόδειξη. Εφόσον $NEXP \subseteq \mathcal{C}$, έχουμε και $NEXP \subseteq P_{/poly}$ κι άρα από Θεώρημα 4.5 έχουμε ότι το παραπάνω ισχύει τετριμμένα για $\mathcal{C} = P_{/poly}$. Ωστόσο όπως είδαμε, προκύπτει τότε ότι $NEXP = EXP$ καθώς απλώς ελέγχουμε τετριμμένα όλα τα κυκλώματα μέχρι κάποιο φραγμένο πολυωνυμικό μέγεθος μαζί με τα TT τους (προφανώς σε $2^{poly(n)}$ χρόνο) και σταματάμε στο πρώτο κύκλωμα-πιστοποιητικό που ικανοποιεί το στιγμιότυπο. Έτσι το i-οστό στοιχείο αυτού του πρώτου πιστοποιητικού υπολογίζεται σε εκθετικό (ως προς την αντίστοιχη είσοδο $\langle x, i \rangle$) χρόνο κι άρα αφού $EXP \subseteq \mathcal{C}$ ισχύει προφανώς ότι το TT (σταθεροποιώντας το x) μπορεί να περιγραφεί από ένα τέτοιο κύκλωμα, από το οποίο εύκολα προκύπτει το αποδεικτέο. ■

Για να εκμεταλλευτούμε στο έπακρο, ωστόσο, την ύπαρξη μη τετριμμένων αλγορίθμων για μία συγκεκριμένη κυκλωματική κλάση $\mathcal{C}$, δεν αρκεί να έχουμε $\mathcal{C}$-απεικόνιση για τα πιστοποιητικά, αλλά χρειαζόμαστε και $\mathcal{C}$-απεικόνιση της ίδιας της ντετερμινιστικής εκτέλεσης ελέγχου του πιστοποιητικού. Σε συμμόρφωση με το NP-πλήρες πρόβλημα $3-SAT$ όπου η εκτέλεση μεταφράζεται σε μια $3-CNF$ φόρμουλα και το πιστοποιητικό αντιστοιχεί στην απονομή που την ικανοποιεί, έχουμε ότι η εκτέλεση ελέγχου (δηλαδή η αντίστοιχη $3-CNF$

φόρμουλα) μπορεί να παραχθεί από ένα $\mathcal{C}$ κύκλωμα. Για αρχή μπορούμε να ορίσουμε το ακόλουθο πρόβλημα:

Ορισμός 5.2.1. *(succinct-SAT)* Ορίζουμε ως $succinct - SAT$ το εξής πρόβλημα: Δοθείσης της περιγραφής ενός κυκλώματος C με n δυφία εισόδου (συμβολίζονται με x) και 3 εξόδους των $n+1$ δυφίων (συμβολίζονται με a, b, c αντίστοιχα) να ελεγχθεί αν υπάρχει απονομή που να ικανοποιεί την φόρμουλα 2^n μεταβλητών και 2^n όρων που ορίζει το C ως εξής:
Για είσοδο x οι έξοδοι $a(x), b(x), c(x)$ αντιστοιχούν στο δείκτη των 3 μεταβλητών που συμμετέχουν στον x-οστό όρο της φόρμουλας (συνολικά n δυφία αρκούν για 2^n δυνατές μεταβλητές) μαζί με ένα έξτρα δυφίο $p_a(x), p_b(x), p_c(x)$ που συμβολίζει το αν παίρνουμε την κατάφαση ή την άρνηση της μεταβλητής.

Το παραπάνω πρόβλημα προσομοιάζει το $NEXP$, με την έννοια ότι έτσι όπως στο $succinct - SAT$ δεχόμαστε λογικά ερωτήματα (δηλαδή λογικές φόρμουλες) τα οποία να είναι σχετικά ομοιόμορφα, με την έννοια ότι όλοι οι όροι παράγονται από ένα κοινό μικρό κύκλωμα, έτσι και στο $NEXP$ ζητάμε μια μεγάλη απονομή που να ικανοποιεί μια εκτέλεση η οποία καθορίζεται από μια μικρή είσοδο (λογαριθμικού μεγέθους ως προς το μέγεθος της εκτέλεσης). Συγκεκριμένα, έχουμε το ακόλουθο αποτέλεσμα:

Θεώρημα 5.1 ([BLT92])**.** *Το $succinct - SAT$ είναι $NEXP$-πλήρες.*

Απόδειξη. (Σχέδιο) Έχουμε ήδη δει πως για κάθε $NEXP$ πρόβλημα υπάρχει μια αντίστοιχη $3 - SAT$ φόρμουλα μήκους πολυωνυμικά συγκρίσιμου με το χρόνο εκτέλεσης της αντίστοιχης μηχανής και η οποία είναι ικανοποιήσιμη αν και μόνο αν η αντίστοιχη μηχανή οδηγεί σε αποδοχή, από το Θεώρημα 2.3. Επομένως το μόνο που μένει να επιχειρηματολογήσουμε είναι ότι η όλη κατασκευή μπορεί να αναπαρασταθεί από μικρά κυκλώματα. Αποφεύγοντας τις τεχνικές λεπτομέρειες, θα στηριχθούμε στα ακόλουθα. Δοθέντος ενός δείκτη i μπορούμε πολύ εύκολα να επιστρέψουμε το i-οστό στοιχείο της αρχικής διαμόρφωσης της μηχανής (κι άρα και του αντίστοιχου κομματιού του ισοδύναμου $3 - SAT$) με ένα πολυωνυμικό κύκλωμα αφού όλες οι θέσεις είναι $\sqcup$ πέραν της εισόδου στην αρχή. Στη συνέχεια για το i-οστό στοιχείο των επόμενων διαμορφώσεων έχουμε ότι κάθε σημείο απλώς εξαρτάται από την i-οστή μεταβολή της μηχανής και ορισμένες τοπικές μεταβλητές (που αντιστοιχούν στις λίγες γειτονικές θέσεις της ταινίας). Επειδή όμως το προηγούμενο ισχύει για κάθε i είναι εύκολο να δούμε ότι το αντίστοιχο κομμάτι της λογικής φόρμουλας μπορεί να παραχθεί από ένα κατασκευάσιμο κύκλωμα πολυωνυμικού μεγέθους, εφόσον το γενικό μοτίβο παραμένει ίδιο και μόνο οι δείκτες των αντίστοιχων μεταβλητών μεταβάλλονται (οι οποίοι είναι εκθετικοί στο πλήθος και άρα αρκούν δείκτες πολυωνυμικού μεγέθους για να συμβολιστούν). Τέλος αυτή η τοπικότητα δεν είναι μόνο χωρική αλλά και χρονική, δηλαδή όπως ακριβώς ισχύουν οι ίδιοι κανόνες για όλα τα στοιχεία της διαμόρφωσης για μια χρονική στιγμή, ισχύουν οι ίδιοι κανόνες μετάβασης και για κάθε χρονική στιγμή (και αυτό επεκτείνεται

ασφαλώς και στη δομή της ισοδύναμης λογικής φόρμουλας). Έτσι μπορούμε να ενοποιήσουμε όλα αυτά τα κυκλώματα σε ένα ενιαίο (επίσης πολυωνυμικού μεγέθους ακριβώς χάρη σε αυτή την ομοιομορφία όλων αυτών) και μάλιστα με κατασκευάσιμο τρόπο (όπως εύκολα μπορούμε να δούμε). Εν τέλει, έχουμε την προφανή αναγωγή όπου από κάθε στιγμιότυπο ενός $NEXP$ προβλήματος κατασκευάζουμε σε πολυωνυμικό χρόνο το παραπάνω κύκλωμα που παράγει την ισοδύναμη $3-SAT$ και έτσι έχουμε τελειώσει την απόδειξη. ■

Σημείωση. Το κύκλωμα που κατασκευάζεται με την παραπάνω αναγωγή έχει εν τέλει τόσα δυφία εισόδου όσα και ο λογάριθμος του μήκους του αντίστοιχου $3-SAT$. Παρότι το σχέδιο της απόδειξης βασίστηκε στην περίπτωση όπου το ισοδύναμο $3-SAT$ έχει μήκος $poly(T(n))$ (όπου $T(n)$ ο χρόνος εκτέλεσης της μη ντετερμινιστικής μηχανής), η παραπάνω πολυωνυμική κατασκευή μπορεί να επεκταθεί και για την κωδικοποίηση όπου το ισοδύναμο $3-SAT$ έχει μήκος $O(T(n)logT(n))$ (βλ. Παράρτημα Α′.1) κι άρα εν προκειμένω θα θεωρούμε ότι για $T(n)=2^{n^c}$ το αντίστοιχο κατασκευάσιμο κύκλωμα έχει μόλις $n^c+klogn$ δυφία εισόδου.

Για τους γνωστούς λόγους που έχουμε δει πολλάκις, οι φόρμουλες μήκους 2^n που παράγονται με τον παραπάνω τρόπο (για πολυωνυμικό μέγεθος του C) είναι ασφαλώς υποσύνολο όλων των δυνατών φορμουλών μήκους 2^n. Θα μπορούσε λοιπόν αυτός ο εκφραστικός περιορισμός των αντίστοιχων ερωτημάτων να αντιστοιχεί και σε εκφραστικό περιορισμό των πιστοποιητικών που μπορούν να το επαληθεύσουν; Δηλαδή δεδομένου ότι το παραπάνω πρόβλημα είναι $NEXP$-πλήρες κι άρα κάθε $NEXP$ ερώτημα μπορεί πράγματι να παραχθεί από ομοιόμορφα πολυωνυμικά κυκλώματα, το ισοδύναμο ερώτημα είναι: Ισχύει ότι $NEXP \subseteq P_{/poly}$; Προς το παρόν, έχουμε μόνο την ακόλουθη πρόταση που δείχνει τη στενή σχέση μεταξύ των δύο.

Πρόταση 5.2.1. *Το $succinct-SAT$ έχει $\mathcal{C}$-συμπυκνώσιμα πιστοποιητικά αν και μόνο αν $NEXP \subseteq \mathcal{C}$.*

Απόδειξη. Πράγματι αν $NEXP \subseteq \mathcal{C}$, έχουμε ήδη δει σε συνδυασμό με το προηγούμενο Θεώρημα και Λήμμα ότι το $succinct-SAT$ έχει και $\mathcal{C}$-συμπυκνώσιμα πιστοποιητικά. Για την αντίστροφη πορεία, θα κατασκευάσουμε ένα ερώτημα (με τρόπο που εύκολα προκύπτει ότι είναι $\mathcal{C}$-συμπυκνώσιμος) που θα εμπεριέχει όλα τα πιθανά ερωτήματα ενός μήκους N κι από το $\mathcal{C}$ πιστοποιητικό που εγγυάται η υπόθεση, θα κατασκευάσουμε εν τέλει ένα $\mathcal{C}$ κύκλωμα για όλες τις εισόδους αυτού του μήκους. Θα το δείξουμε για το $NEXP$-πλήρες $succinct-SAT$ και επεκτείνεται φυσικά σε όλα τα υπόλοιπα. Έστω, λοιπόν, το ερώτημα που για κάθε είσοδο μήκους N (δηλαδή κύκλωμα μεγέθους N που παράγει μία φόρμουλα μέσω του TT της) και χρησιμοποιώντας ανεξάρτητες μεταβλητές για κάθε είσοδο, ζητάει αν υπάρχει απονομή όλων αυτών των πολύ 2^N*2^N μεταβλητών που να ικανοποιεί τουλάχιστον S όρους (το οποίο S δίνεται επίσης ως είσοδος). Επειδή το παραπάνω είναι στο $NEXP$, μπορεί να μοντελοποιηθεί με $succinct-SAT$ ερώτημα, και αν σταθεροποιήσουμε το S στη τιμή Z

για την οποία ικανοποιείται το μέγιστο δυνατό πλήθος όρων (κι άρα σίγουρα όλοι οι όροι των ικανοποιήσιμων υποφόρμουλων και μόνο μερικοί από τις μη ικανοποιήσιμες), έχουμε από υπόθεση ότι υπάρχει ένα κύκλωμα D^* πολυωνυμικού μεγέθους $2N$ εισόδων που να παράγει όλη αυτή τη μεγάλη ικανοποιητική απονομή. Το κύκλωμα αυτό έχει πολυωνυμικό μέγεθος ως προς τις αρχικές παραμέτρους του προβλήματος κι άρα το μόνο που μένει είναι να ορίσουμε το εξής πρόβλημα: Με είσοδο ένα κύκλωμα D που περιγράφει μια απονομή (στη θέση του οποίου σκοπεύουμε να δώσουμε το D^*) και μια είσοδο x να ελεγχθεί αν το κομμάτι της παραγόμενης απονομής που αντιστοιχεί στο στιγμιότυπο της εισόδου x (δηλαδή οι μεταβλητές με δείκτες μεταξύ των $x0^N$ και $x1^N$) ικανοποιεί το $succinct - SAT$ που παράγεται από το x. Όλο αυτό ασφαλώς μπορεί να μοντελοποιηθεί με ένα $succinct - SAT$ ερώτημα, που απλά διατρέχει συμμετρικά όλους τους όρους (με κάποιο πολυωνυμικό υπέρβαρο), κι άρα υπάρχει κύκλωμα $F \in \mathcal{C}$ το οποίο να δέχεται δύο εισόδους D και x και να επιστρέφει 1 αν και μόνο αν ισχύει το παραπάνω. Σταθεροποιώντας το D ίσο με D^*, έχουμε ότι το παραπάνω συμβαίνει αν και μόνο αν υπάρχει απονομή που να ικανοποιεί το εκθετικού μήκους SAT που παράγεται από την είσοδο x, δηλαδή το F τότε πράγματι μοντελοποιεί ορθά το $succinct-SAT$ για όλες τις εισόδους x μήκους N και αφού είναι τύπου $\mathcal{C}$, η απόδειξη ολοκληρώθηκε. ■

Σε γενικές γραμμές, έχει γίνει φανερή πλέον η σχέση του $NEXP$ με την παραγωγή συμβολοσειρών από κυκλώματα τόσο στο κομμάτι της παραγωγής ερωτημάτων, όσο και σε αυτό της πιθανής παραγωγής πιστοποιητικών. Η παραπάνω σύνδεση μάλιστα είναι τόσο έντονη έτσι ώστε όποτε υποθέτουμε ότι $NEXP \subseteq \mathcal{C}$, οι αντίστοιχες συμβολοσειρές να μπορούν μάλιστα να αναπαρασταθούν επίσης από κυκλώματα της εκάστοτε κλάσης $\mathcal{C}$, μια ιδιότητα που θα σταθεί καίριας σημασίας στα Θεωρήματα που ακολουθούν.

5.3 Εφαρμογές

Ακολουθούν ορισμένες χαρακτηριστικές περιπτώσεις όπου η παραπάνω μέθοδος λειτουργεί με αποδεδειγμένα αποτελέσματα. Συγκεκριμένα παίρνοντας μια κυκλωματική ιδιότητα (που αναφέρεται στο TT της εισόδου-κυκλώματος) για μια συγκεκριμένη κλάση $\mathcal{C}$ και θεωρώντας ότι υπάρχει μη τετριμμένος αλγόριθμος που την επιλύει σε χρόνο συντομότερο αυτού της εξαντλητικής αναζήτησης, καταλήγουμε σε ένα διαγωνιοποιητικό άτοπο και σε μια αντίστοιχη σχέση διαχωρισμού κάποιας υπολογιστικής κλάσης από τη $\mathcal{C}$.

5.3.1 $MCSP$

Παρότι παραπάνω ορίσαμε το πρόβλημα $\mathcal{C} - MCSP$, είναι εύκολο να δούμε ότι (περιορίζοντας την είσοδο σε κλάσεις τύπου $\mathcal{C} \subseteq P_{/poly}$) πρόκειται για NP-πλήρες πρόβλημα (αφού το σταθερό κύκλωμα έχει το ελάχιστο μέγεθος), οπότε από αυτή τη σκοπιά αρκεί η μελέτη για το $\mathcal{C}-SAT$ που ακολουθεί ύστερα. Κατ'

εξαίρεση λοιπόν θεωρούμε ότι το κύκλωμα εισόδου μπορεί να είναι ό,τι μεγέθους (ως προς τα δυφία εισόδου του) θέλουμε και δεδομένου ότι μπορούμε πάντα να αναπαραστήσουμε με τετριμμένο κύκλωμα εκθετικού μεγέθους οποιοδήποτε TT, παίρνουμε για απλότητα την περίπτωση όπου η είσοδος είναι το ίδιο το TT.

Συγκεκριμένα ορίζουμε ως $MCSP$ το πρόβλημα, όπου δοθέντος ενός TT και ενός αριθμού S υπάρχει αποδοχή αν και μόνο αν υπάρχει κύκλωμα μεγέθους S που να το παράγει. Υπάρχει τετριμμένος εκθετικός αλγόριθμος αφού το ελάχιστο ισοδύναμο κύκλωμα είναι το πολύ όσο το μέγεθος του TT κι άρα ελέγχοντας τετριμμένα κάθε κύκλωμα μέχρι το $min\{S, TT\}$ (και για κάθε κύκλωμα ελέγχοντας την ισοδυναμία σε χρόνο εκθετικό ως προς το πλήθος δυφίων εισόδου (κι άρα σίγουρα γραμμικό ως προς όλη την είσοδο)) μπορούμε να το αποφασίσουμε. Για την αντίστοιχη ιδιότητα (και σταθεροποιώντας την είσοδο S σε κάτι υπερπολυωνυμικό) έχουμε κατά τα γνωστά ότι υπάρχει πληθωριστικότητα, αλλά δε γνωρίζουμε για κατασκευασιμότητα μέχρι στιγμής (και πιστεύεται ότι δεν υπάρχει καθώς αν υπήρχε πολυωνυμικός αλγόριθμος για αυτό το πρόβλημα, τότε θα είχαμε την περίπτωση ύπαρξης ενός φυσικού επιχειρήματος κατά του $P_{/poly}$ (βλ. [KC00])). Εν τούτοις έχουμε το ακόλουθο Θεώρημα, που μας δίνει μια πρώτη γεύση της ισχυρής σύνδεσης εύκολων αλγορίθμων για κυκλώματα και την ύπαρξη κάτω φραγμάτων για υψηλότερες κλάσεις εξαιτίας αυτών.

Θεώρημα 5.2. *Αν $MCSP \in DTIME[f(n)] \subseteq subEXP$, τότε $NEXP \nsubseteq P_{/poly}$ (όπου f κατασκευάσιμη και τέτοια ώστε να υπάρχει (πολυωνυμικά) κατασκευάσιμη $h(n) = n^{\omega(1)}$ με $DTIME[f(h(n))] \subseteq E$).*

Απόδειξη. Πράγματι έστω ότι $MCSP \in DTIME[f(n)] \subseteq subEXP$ και $NEXP \subseteq P_{/poly}$. Τότε $NE \subseteq SIZE[n^c]$ για κάποια σταθερά $c > 0$. Έστω το ακόλουθο πρόγραμμα που ανήκει στο NE:
Με είσοδο $\langle x, i\rangle$ μάντεψε μια συμβολοσειρά Y της μορφής $y0^{2^n - h(n)}$, όπου το y είναι μήκους $h(n)$, και το y να έχει κυκλωματική δυσκολία τουλάχιστον $\sqrt{h(n)}$.[1] Επαλήθευσε με τον «σύντομο» αλγόριθμο του $MCSP$ ότι πράγματι ισχύει αυτή η δυσκολία για το y (σε χρόνο προφανώς επίσης $f(h(n)) \leq 2^{O(n)}$) και αποδέξου αντίστοιχα.

Όμως επειδή πάντα υπάρχει ένα τέτοιο δύσκολο y (από Κυκλωματική Ιεραρχία) κι άρα πάντα υπάρχει κι ένα τέτοιο Y και επειδή υποθέσαμε $NEXP \subseteq P_{/poly}$, σημαίνει ότι υπάρχει τουλάχιστον μια άπειρη οικογένεια τέτοιων συμβολοσειρών Y που να ανήκουν στο $P_{/poly}$ (δηλαδή να παράγονται ως TT μιας οικογένειας κυκλωμάτων πολυωνυμικού μεγέθους), το οποίο είναι άτοπο, εφόσον υποθέσαμε ότι $h(n) = n^{\omega(1)}$. ■

Να σημειωθεί ότι παρότι έχουμε ένα αντίστοιχο αποτέλεσμα για το SAT (δηλαδή ότι αν $NP \subseteq subEXP$, τότε $NEXP \nsubseteq P_{/poly}$, καθώς αλλιώς κα-

[1] Προκύπτει εύκολα τότε ότι και το Y είναι της ίδιας τάξης κυκλωματικής δυσκολίας (καθώς αν γινόταν να γραφτεί με π.χ. n^c κυκλώματα, το ίδιο θα ίσχυε, με σταθεροποίηση κάποιων από τα δυφία εισόδου, και για το y οδηγώντας σε αντίφαση).

ταλήγουμε κατά τα γνωστά ότι $NEXP \subseteq \Sigma_2^P \subseteq subNEXP$), εν τούτοις είδαμε ότι μπορούμε να καταλήξουμε στο ίδιο συμπέρασμα ακόμη κι αν χρησιμοποιήσουμε μόλις το $MCSP$ για το οποίο δεν έχουμε απόδειξη ότι είναι NP-πλήρες (και για διάφορους λόγους πιστεύεται από μεγάλη μερίδα της ερευνητικής κοινότητας ότι δεν είναι ή ότι αν είναι, η απόδειξη οφείλει να είναι αρκετά μη τετριμμένη, εν μέρει επειδή θα οδηγούσε στην απόδειξη κυκλωματικών κάτω φραγμάτων [KC00]).

5.3.2 $\mathcal{C}$-SAT

Για τη γενική μορφή του SAT έχουμε, όπως είπαμε, ήδη αντίστοιχα συμπεράσματα, όμως περισσότερο ενδιαφέρον υπάρχει για τη περίπτωση του $\mathcal{C}$-SAT. Συγκεκριμένα προκύπτει ότι αν μία κλάση $\mathcal{C}$ έχει ιδιότητες τέτοιες ώστε να μπορεί να ελεγχθεί σε (έστω και ελάχιστα) μη τετριμμένο χρόνο το αν έχει είσοδο που να το οδηγεί σε έξοδο 1, τότε δεν είναι αρκετά ισχυρό ώστε να μπορεί να μοντελοποιήσει το $NEXP$. Ειδικότερα έχουμε τα ακόλουθα Θεωρήματα, τα οποία αποδείχτηκαν πρόσφατα κατά κύριο λόγο από τον Ryan Williams.

Θεώρημα 5.3 ([Wil13, Wil11])**.** *Έστω ότι το $\mathcal{C}$-SAT λύνεται σε χρόνο $2^N/N^k$ για κάθε $k > 0$, όπου το $\mathcal{C}$ χαρακτηρίζεται από τα κυκλώματα N δυφίων εισόδου, βάθους $O(d(N))$ και μεγέθους $poly(N)$ (πάνω σε κάποια πλήρη βάση πυλών). Τότε $NEXP \nsubseteq \mathcal{C}$.*[2]

Απόδειξη. Πράγματι έστω ένα στιγμιότυπο ενός $NEXP$ προβλήματος, το οποίο τρέχει σε 2^{n^a} χρόνο. Όπως είδαμε από το Θεώρημα 5.1, η αντίστοιχη ισοδύναμη λογική φόρμουλα μπορεί να αναπαρασταθεί ως το TT ενός κατασκευάσιμου πολυωνυμικού κυκλώματος Q. Από την άλλη, αν $NEXP$ $subseteq\mathcal{C}$, τότε είδαμε από τη Πρόταση 5.2.1 ότι υπάρχει αντίστοιχο πιστοποιητικό που παράγεται από $\mathcal{C}$ κυκλώματα V. Υπάρχει πλέον τετριμμένος τρόπος ελέγχου του πιστοποιητικού που παράγεται από το κύκλωμα V μέσω των αντίστοιχων κυκλωμάτων: Θεωρώντας ότι οι έξοδοι του Q είναι τέτοιοι ώστε να αντιστοιχούν στους δείκτες των αντίστοιχων μεταβλητών κάθε όρου (δηλαδή 3 έξοδοι a, b, c των $poly(n)$ δυφίων όπως στον Ορισμό 5.2.1 μαζί με 3 δυφία προσήμου (άρνηση ή κατάφαση) p_a, p_b, p_c όπου αν $p = 1$, τότε έχουμε άρνηση και κατάφαση αλλιώς), τότε έχουμε ότι δίνοντας κάθε τέτοια έξοδο ως είσοδο στο V, το V θα επιστρέφει 1 αν και μόνο αν στην απονομή που ορίζει, η αντίστοιχη μεταβλητή έχει την τιμή T. Επομένως επειδή κάθε όρος πρέπει να έχει τουλάχιστον ένα από τα τρία l_i αληθές για να είναι ικανοποιήσιμος, φτιάχνουμε το εξής κύκλωμα S: Με είσοδο το αντίστοιχο δείκτη i για το Q, οδηγούμε κάθε μία από τις τρεις εξόδους μήκους $poly(n)$ του Q σε ένα ξεχωριστό αντίτυπο του V (τρία στο σύνολο) και αν v_a, v_b, v_c οι αντίστοιχες τρεις έξοδοι, τότε δίνουμε ως

[2] Αυτό ερμηνεύεται πιο λεπτομερώς ως εξής: Αν για κάθε οικογένεια κυκλωμάτων βάθους $b * d(N)$ και μεγέθους N^c υπάρχει αλγόριθμος χρόνου $2^N/N^k$ που να επιλύει το SAT τους, για κάθε $k > 0$, τότε το $NEXP$ δε μπορεί να εκφραστεί με καμία οικογένεια κυκλωμάτων βάθους $b'd(N)$ και μεγέθους $N^{c'}$.

έξοδο στο συνολικό κύκλωμα το $o_S = (v_a \oplus p_a) \vee (v_b \oplus p_b) \vee (v_c \oplus p_c)$. Εύκολα βλέπουμε ότι η έξοδος του παραπάνω κυκλώματος S με είσοδο i γίνεται 1 αν και μόνο αν ο i-οστός όρος είναι ικανοποιήσιμος. Άρα το V είναι πράγματι ένα έγκυρο πιστοποιητικό για το συγκεκριμένο στιγμιότυπο αν και μόνο αν το S έχει έξοδο 1 για κάθε είσοδο i. Αυτό ασφαλώς δε σημαίνει ότι αυτό μπορεί να γίνει σε λιγότερο από εκθετικό χρόνο, αφού στη γενική περίπτωση αποτελεί ένα $coNP$-πλήρες πρόβλημα, το στιγμιότυπο του οποίου, μάλιστα, σχηματίζεται αφότου έχουμε μαντέψει το αντίστοιχο V (δηλαδή μέχρι στιγμής έχουμε απλά μια Σ_2^P απεικόνιση του $NEXP$ η οποία είναι αναμενόμενο να υπάρχει λόγω της υπόθεσης ότι $NEXP \subseteq \mathcal{C} \subseteq P_{/poly}$).

Για να μπορέσουμε, όμως να χρησιμοποιήσουμε την υπόθεση, θα πρέπει να κατασκευάσουμε ένα καθ' ολοκληρίαν $\mathcal{C}$ κύκλωμα για το οποίο να αρκεί ο έλεγχος ύπαρξης μηδενικού στο TT του. Για το παραπάνω κύκλωμα S, έχουμε ότι το πιστοποιητικό V μπορεί να γραφτεί με $\mathcal{C}$ κυκλώματα (τα οποία μαντεύουμε), αλλά για το κομμάτι του Q έχουμε κατασκευάσιμο τρόπο μόνο για τη περίπτωση του $P_{/poly}$. Θα εκμεταλλευτούμε όμως το γεγονός ότι $NEXP \subseteq \mathcal{C}$ και δεδομένου ότι το Q είναι κατασκευάσιμο σε πολυωνυμικό χρόνο, το ίδιο θα ισχύει και για την «υποσυνάρτηση» κάθε πύλης του. Άρα για κάθε πύλη του υπάρχει ένα $\mathcal{C}$ κύκλωμα που υπολογίζει το αποτέλεσμα της για κάθε είσοδο (όπως ασφαλώς και ένα που να υπολογίζει την ίδια τη συνάρτηση του Q). Ο λόγος που χρειαζόμαστε τις παραπάνω υποσυναρτήσεις είναι επειδή δεν αρκεί να μαντέψουμε απλά ένα κύκλωμα τύπου $\mathcal{C}$ ισοδύναμο με το Q· θα πρέπει να μπορούμε να αποδείξουμε αυτή την ισοδυναμία. Αυτό θα το κάνουμε βηματικά επαληθεύοντας εν παραλληλία την ισοδυναμία όλων των πυλών καταλήγοντας αναδρομικά στο Q. Μαντεύουμε, λοιπόν, ένα τέτοιο $\mathcal{C}$ κύκλωμα $\mathrm{G_i}$ για κάθε πύλη g_i του Q και μένει να βρούμε τρόπο να επαληθεύσουμε την ορθότητα του (δηλαδή ότι κάθε $\mathrm{G_i}$ είναι ισοδύναμο με τη g_i). Για τις πύλες g_{inp} που αντιστοιχούν στις εισόδους είναι εύκολο να επαληθεύσουμε την ορθότητα (απλώς ελέγχοντας αν τα αντίστοιχα κυκλώματα δίνουν κατ' ευθείαν ως έξοδο την αντίστοιχη είσοδο). Από κει και πέρα για κάθε επόμενο κύκλωμα-πύλη $\mathrm{G_i}$ με $g_i = op(g_j, g_k)$ (όπου op μια συνάρτηση-πύλη, π.χ. AND, OR, NOT κ.ο.κ.) αρκεί να φτιάξουμε τον όρο $\mathrm{O_i} = \neg(\mathrm{G_i} \oplus op(\mathrm{G_j}, \mathrm{G_k}))$, ο οποίος γίνεται 1 αν και μόνο αν για την αντίστοιχη είσοδο x, ισχύει $\mathrm{G_i}(x) = op(\mathrm{G_j}(x), \mathrm{G_k}(x))$. Έχουμε πλέον ότι όλα αυτά τα κυκλώματα είναι πράγματι ισοδύναμα με τις αντίστοιχες πύλες g_i αν και μόνο αν η έξοδος $\mathrm{O_i}$ είναι 1 για κάθε είσοδο. Επίσης αν $\mathrm{G_Q}$ το αντίστοιχο κύκλωμα της πύλης εξόδου g_Q του Q, μπορούμε να προωθήσουμε τις εξόδους του στο V όπως περιγράφηκε προηγούμενως. Έχουμε λοιπόν ότι πρέπει ταυτόχρονα να βγαίνει 1 για κάθε είσοδο τόσο στον όρο $\mathrm{O_S}$ όσο και σε κάθε $\mathrm{O_i}$. Ενώνουμε όλες αυτές τις εξόδους σε μία τελική πύλη AND φτιάχνοντας έτσι ένα τελικό κύκλωμα T το οποίο επιστρέφει 1 για όλες τις εισόδους, αν και μόνο αν το αρχικό στιγμιότυπο είναι ικανοποιήσιμο. Επιπλέον έχουμε ότι το T είναι στο $\mathcal{C}$. Πράγματι το μέγιστο βάθος είναι $d_{O_i} + d_{O_V} + O(1) = O(d(n))$ και το μέγεθος παραμένει πολυωνυμικό (ως γινόμενο πολυωνύμων εφόσον το Q έχει πολυωνυμικό πλήθος πυλών).

Έχουμε εν τέλει ότι πρέπει να μαντέψουμε όλα τα κυκλώματα G_i μαζί με το V (όπου όλα κωδικοποιούνται σε μια ενιαία συμβολοσειρά πολυωνυμικού μήκους) και μετά κατασκευάζοντας το T σε πολυωνυμικό χρόνο, να ελέγξουμε αν το T έχει έξοδο 1 για κάθε είσοδο. Επειδή όμως το T είναι κύκλωμα $\mathcal{C}$, μπορούμε να χρησιμοποιήσουμε τον αλγόριθμο χρόνου $2^N/N^k$, όπου το N είναι το πλήθος δυφίων εισόδου του Q. Επειδή όπως είπαμε υπάρχει κωδικοποίηση μήκους $O(T(n)logT(n)) = O(2^{n^a}n^a)$, το πλήθος των δυφίων είναι $n^a + blogn$. Επομένως διαλέγοντας τον αντίστοιχο αλγόριθμο για κάποιο $k > b/a$ (υπάρχει από υπόθεση), έχουμε ότι ο αλγόριθμος τρέχει σε $\frac{2^{n^a+blogn}}{(n^a+blogn)^k} \leq 2^{n^a}n^{-(ak-b)}$. Άρα ο όλος αλγόριθμος μπορεί να τρέξει με μία μη ντετερμινιστική μηχανή χρόνου $2^{n^a}/n^e$ για κάποιο $e > 0$ και καταλήγουμε ότι $NTIME[2^{n^a}] \subseteq NTIME[2^{n^a}/n^e]$ παραβιάζοντας τη μη ντετερμινιστική χρονική Ιεραρχία κι άρα οδηγώντας σε άτοπο. Επομένως αν ισχύει η υπόθεση, τότε πράγματι $NEXP \nsubseteq \mathcal{C}$. ■

Το προηγούμενο Θεώρημα μπορεί να επεκταθεί και για την περίπτωση του EXP^{NP} αλλά τώρα με ακόμη πιο ισχυρά αποτελέσματα, καθώς πλέον μπορούμε να βγάλουμε αντίστοιχο συμπέρασμα ακόμη κι αν το $\mathcal{C}$ περιλαμβάνει κυκλώματα (σχεδόν) εκθετικού μεγέθους. Ένας από τους κυριότερους λόγους για αυτό, είναι ότι ύπαρξη σύντομου πιστοποιητικού έχουμε (μέχρι στιγμής) μόνο όταν υποθέτουμε ότι το $NEXP$ εκφράζεται από κυκλώματα (υπερ-)πολυωνυμικού, αλλά πάντοτε αισθητά υποεκθετικού, μεγέθους. Έτσι, τώρα η προηγούμενη απόδειξη μπορεί να γίνει λίγο πιο στενή (ως προς το μέγεθος του $\mathcal{C}$) καθώς πλέον μπορούμε να βρούμε το ελάχιστο κύκλωμα-πιστοποιητικό με δυαδική αναζήτηση και η εξασφάλιση της αναφοράς σε ένα κοινό πιστοποιητικό επιτρέπει να προσπεράσουμε τους φραγμούς που συναντήσαμε κατά την απόδειξη του Θεωρήματος 4.5.

Θεώρημα 5.4 ([Wil13, Wil11]). *Έστω ότι το $\mathcal{C}$-SAT λύνεται σε χρόνο $2^N/N^k$ για κάθε $k > 0$, όπου το $\mathcal{C}$ περιέχει τα κυκλώματα N δυφίων εισόδου, βάθους $O(d(N))$ και μεγέθους $2^{N^{o(1)}}$. Τότε $E^{NP} \nsubseteq \mathcal{C}$.*

Απόδειξη. Έστω ότι $E^{NP} \subseteq \mathcal{C}$ και $L \in NTIME[2^{n/2}]$ κι έστω το πρόβλημα $L' = \{\langle x, i\rangle |$ Το i-οστό δυφίο του ελάχιστου (αλφαριθμητικά) πιστοποιητικού που ικανοποιεί το στιγμιότυπο για την L με είσοδο x είναι $1\}$. Έχουμε ότι το L' είναι υπολογίσιμο στο E^{NP} εφόσον απλώς επαναλαμβάνουμε το ερώτημα, μήκους $2^{n/2}poly(n)$ για το αν υπάρχει πιστοποιητικό για την L (με είσοδο x) αλφαριθμητικά μικρότερο του S, για $2^{n/2}$ φορές (με δυαδική αναζήτηση στο S) μέχρις ότου να εντοπιστεί το ελάχιστο πιστοποιητικό και στη συνέχεια αποδεχόμαστε ανάλογα με το αν το i-οστό δυφίο του είναι 1. Επομένως έχουμε ότι υπάρχει κύκλωμα τύπου $\mathcal{C}$ το οποίο να αποτελεί πιστοποιητικό για το παραπάνω πρόβλημα L. Από κει και πέρα επαναλαμβάνουμε την προηγούμενη απόδειξη όπου όμως όλα τα αντίστοιχα κυκλώματα είναι τύπου $\mathcal{C}$ κι άρα το τελικό κύκλωμα T είναι μεγέθους $poly(n) * 2^{n^{o(1)}}$, οπότε εφαρμόζοντας τον

υποτιθέμενο αλγόριθμο για τη συγκεκριμένη έκδοση του $\mathcal{C}$-SAT καταλήγουμε πάλι ότι $L \in NTIME[2^{\frac{n}{2}}/n^{\Theta(1)}]$ οπότε και πάλι σε άτοπο. ■

Έχει γίνει ξεκάθαρο πλέον ότι έστω και μία μικρή βελτίωση στη χρονική πολυπλοκότητα του $\mathcal{C}-SAT$ οδηγεί σε αποκλεισμό της αντίστοιχης κυκλωματικής κλάσης από τις ψηλές μη ντετερμινιστικές ομοιόμορφες εκθετικές κλάσεις. Παρότι τα παραπάνω αποτελέσματα έχουν αξία, έστω και σε αυτή την υπό συνθήκες μορφή (ως γεννήτριες αποδείξεων), στέρεα αποτελέσματα διαχωρισμού προκύπτουν μόνο όταν έχουμε βρει αλγόριθμο (ή έστω έχουμε δείξει την ύπαρξη ενός) που να πληροί τις υποθέσεις. Το ευτυχές, είναι ότι έχουμε ήδη δει έναν τέτοιον αλγόριθμο για το ACC^0 οδηγώντας στα ακόλουθα συμπεράσματα.

Θεώρημα 5.5. $NEXP \not\subseteq ACC^0$

Απόδειξη. Από το Θεώρημα 3.7 έχουμε ότι για κάθε βάθος d, υπάρχει ένα $\delta > 0$ τέτοιο ώστε για κάθε κύκλωμα ACC^0 μεγέθους 2^{n^δ} (κι άρα για κάθε κύκλωμα μεγέθους $poly(n)$) το αντίστοιχο SAT να λύνεται σε χρόνο $2^{n-n^\delta} \leq 2^n/n^k$ για κάθε $k > 0$. Εφόσον λοιπόν ικανοποιούνται οι συνθήκες του Θεωρήματος 5.3 προκύπτει ότι πράγματι $NEXP \not\subseteq ACC^0$. ■

Αντίστοιχα έχουμε:

Θεώρημα 5.6. *Το E^{NP} δεν επιδέχεται οικογένειες κυκλωμάτων σταθερού βάθους, μεγέθους $2^{n^{o(1)}}$ και με πύλες MOD_m (πέραν των βασικών). Ακόμη περισσότερο για κάθε βάθος d, υπάρχει ένα $\delta > 0$ τέτοιο ώστε το E^{NP} να μην επιδέχεται οικογένειες κυκλωμάτων πυλών MOD_m (πέραν των βασικών), βάθους d και μεγέθους 2^{n^δ}.*

Απόδειξη. Η απόδειξη προκύπτει όπως και προηγουμένως με συνδυασμό των Θεωρημάτων 3.7 και 5.4. Συγκεκριμένα, μελετώντας με περισσότερη στενότητα και λεπτομέρεια την απόδειξη του Θεωρήματος 5.4 μπορούμε τετριμμένα να καταλήξουμε στο αποδεικτέο ισοζύγιο. ■

Οι παραπάνω αποδείξεις είναι οι πρώτες, μετά από μια μεγάλη περίοδο «ξηρασίας», που αποδεικνύουν πλήρως ανομοιόμορφα κάτω κυκλωματικά φράγματα για μία ομοιόμορφη υπολογιστική κλάση και αποτελούν προφανώς ενός είδους ορόσημο στην ιστορία των σχετικών αποδείξεων. Το σημαντικότερο, αποτελούν τη πρώτη άνευ όρων «επιτυχία» των υποθετικών αυτών Θεωρημάτων, τα οποία απαιτούν την ύπαρξη ενός μη τετριμμένου αλγορίθμου για να οδηγήσουν χωρίς συνθήκες στο αντίστοιχο κάτω φράγμα.

5.3.3 $\mathcal{C}$-$CAPP$

Είδαμε ότι μια σχετικά μικρή βελτίωση στον τετριμμένο αλγόριθμο του $\mathcal{C}-SAT$ μπορεί να οδηγήσει σε κάτω φράγματα και επίσης είδαμε ότι η κατασκευή τέτοιων αλγορίθμων δεν είναι απίθανη. Εν τούτοις μία από τις δυσκολίες που

υπάρχουν σε αυτούς, είναι η στενή τους σύνδεση με NP-πλήρη προβλήματα και η γενικότερη αδυναμία μας μέχρι στιγμής να ξεχωρίσουμε τέτοια στιγμιότυπα από ένα μαύρο κουτί στο οποίο η απονομή ικανοποίησης να είναι μια τυχαία τιμή (κι άρα να απαιτείται η εξαντλητική αναζήτηση για την εύρεση ενός άσσου μέσα σε εκθετικό πλήθος μηδενικών). Ένα πιο ελπιδοφόρο πρόβλημα φαίνεται να είναι το $\mathcal{C}$-$CAPP$, στο οποίο η είσοδος είναι ένα κύκλωμα τύπου $\mathcal{C}$ και ένας αριθμός $b \in [0,1]$ και ζητείται να βρεθεί αν $|\Pr[C(\boldsymbol{x}) = 1] - b| < \epsilon$ για κάποια θετική σταθερά $\epsilon < 1/2$. Είναι προφανές (δεδομένης και της αυθαιρεσίας των σταθερών) ότι αρκεί να περιοριστούμε σε μία μόνο εκδοχή του προβλήματος, π.χ. αυτή στην οποία $\epsilon = 0.1$ και όπου τότε το b δεν έχει νόημα να παίρνει διακριτές τιμές μικρότερης απόστασης από $\epsilon = 0.1$ (οπότε έχουμε άμεσα μια διακριτοποίηση των πιθανών τιμών εισόδου). Επιπλέον είναι αρκετά προφανές ότι δε θα τεθεί ουσιώδες ζήτημα ισοδυναμίας αλγορίθμων (από χρονικής πλευράς) αν αντί για την παραπάνω συνθήκη, θέσουμε ως συνθήκη την $\Pr[C(\boldsymbol{x}) = 1] > b$ με σφάλμα το πολύ ϵ κ.ο.κ.

Διαισθητικά το παραπάνω πρόβλημα φαίνεται να αποτελεί απλούστερη περίπτωση από το $\mathcal{C}$-SAT αφού το αποτέλεσμα του δεν αλλάζει από την ύπαρξη μιας μόνο τιμής 1 ανάμεσα σε πολλές άλλες διαφορετικές, αλλά από την ύπαρξη ενός συνήθως συντριπτικά μεγάλου μέρους αυτών ανάμεσα στις υπόλοιπες.[3] Προκύπτει ωστόσο, ότι μία βελτιωμένη έκδοση του αντίστοιχου τετριμμένου αλγορίθμου για το $\mathcal{C}$-$CAPP$ είναι επίσης επαρκής για την απόδειξη παρόμοιων κάτω φραγμάτων.

Θεώρημα 5.7 ([BSV14, Wil13]). *Αν το $\mathcal{C}$-$CAPP$ έχει (μη ντετερμινιστικό) αλγόριθμο χρόνου $2^{N^{o(1)}}$ για κάθε $k > 0$, όπου $\mathcal{C} \subseteq P_{/poly}$ μια κυκλωματική κλάση με παραμέτρους βάθους και μεγέθους τέτοιες ώστε να είναι κλειστή ως προς τις πολυωνυμικές μεγεθύνσεις*[4] *τότε $NEXP \not\subseteq \mathcal{C}$.*

Απόδειξη. (Σχέδιο) Όπως έχουμε δει, αν $NEXP \subseteq \mathcal{C} \subseteq P_{/poly}$, τότε $NEXP = MA$ όπου απλώς δίνοντας στον Ελεγκτή (πολυωνυμικού χρόνου) το κύκλωμα που περιγράφει τη δράση του Αποδείκτη, ο πρώτος έχει μόνο να ελέγξει την ορθότητα χρησιμοποιώντας τυχαία δυφία (με ζεύγος ποσοδεικτών, ισούται με το $(\exists\exists^+, \forall\exists^+)$). Προκύπτει ωστόσο [BSV14] ότι δοθέντων των τυχαίων δυφίων, ο Ελεγκτής μπορεί να αντιστοιχιστεί σε μια $3-CNF$ (με είσοδο ασφαλώς και το κύκλωμα του Αποδείκτη) κι άρα σίγουρα μπορεί να μοντελοποιηθεί από ένα $\mathcal{C}$ κύκλωμα (εφόσον υποθέσαμε $AC^0 \subseteq \mathcal{C}$) κι επομένως η όλη διαλογική συζήτηση προκύπτει ότι μοντελοποιείται εν τέλει από ένα $\mathcal{C}$ κύκλωμα T, εισόδων

[3] Εν τούτοις, δεδομένου ότι το συγκεκριμένο πρόβλημα, όπως προκύπτει τετριμμένα, (για $\mathcal{C} = P_{/poly}$) είναι $promiseBPP$-πλήρες και ότι δεν είναι γνωστή η ακριβής σχέση αυτής της κλάσης με το NP, με τη τωρινή γνώση θα μπορούσε να προκύψει ασφαλώς μέχρι και ότι είναι ισοδύναμο ή και δυσκολότερο από το αντίστοιχο SAT.

[4] Δηλαδή για κάθε οικογένεια κυκλωμάτων στο $\mathcal{C}$ με βάθος $\Omega(d(n))$ και μέγεθος $\Omega(S(n))$, υπάρχει αντίστοιχη οικογένεια επίσης στο $\mathcal{C}$ με βάθος $\Omega(d(n^k))$ και μέγεθος $\Omega(S(n^k))$ για κάθε $k > 0$. Όλες οι κλάσεις (υποσύνολα του $P_{/poly}$) που έχουμε δει μέχρι στιγμής έχουν αυτή την ιδιότητα.

$\langle x, r\rangle$ (όπου x η αρχική είσοδος, και r τα τυχαία δυφία). Σταθεροποιώντας λοιπόν την είσοδο x, μένει να εφαρμόσουμε τον αλγόριθμο χρόνου $2^{N^{o(1)}}$ επί του αντίστοιχου κυκλώματος T (πολυωνυμικού ως προς το n μεγέθους) – όπου ακόμη κι αν είναι μη ντετερμινιστικός, μαντεύουμε το αντίστοιχου μήκους πιστοποιητικό πάλι στον ίδιο υποεκθετικό χρόνο. Ανάλογα το αποτέλεσμα (αν δηλαδή προκύψει ότι $\Pr[T(\boldsymbol{r}) = 1] > 2/3$ αποδεχόμαστε) έχουμε επιλύσει το αρχικό πρόβλημα σε υποεκθετικό μη ντετερμινιστικό χρόνο, οδηγώντας στο συμπέρασμα ότι $NEXP \subseteq subNEXP$ που είναι άτοπο από μη ντετερμινιστική Χρονική Ιεραρχία. ■

Έχει γίνει πλέον φανερός ο τρόπος, με τον οποίο, σχεδόν οποιαδήποτε κλασική ιδιότητα (που αναφέρεται στη σημασιολογική συμπεριφορά ενός κυκλώματος), η οποία επιδέχεται αλγόριθμο μη τετριμμένου χρόνου, οδηγεί σε ισχυρά (ως προς την ανομοιομορφία) κάτω φράγματα για εκθετικές υπολογιστικές κλάσεις. Με άλλα λόγια, διαφαίνεται ένα ισοζύγιο στο οποίο η ύπαρξη ένας εύκολου αλγορίθμου για μια οικογένεια κυκλωμάτων καθιστά τη δεύτερη αρκετά απλή, ώστε να μη μπορεί να εκφράσει συναρτήσεις υψηλών κλάσεων· αλλά και αντίθετα, μια ομοιόμορφη υπολογιστική κλάση που αποδεδειγμένα επιδέχεται μοντελοποίηση από οικογένειες κυκλωμάτων, για τις οποίες υπάρχουν εύκολοι αλγόριθμοι, δεν μπορεί να συμπεριλαμβάνει ορισμένες υψηλές ομοιόμορφες κλάσεις. Δηλαδή αν με κάποιο επιχείρημα δείχναμε ότι $BPP \subseteq ACC^0$ ή $NEXP \nsubseteq P_{/poly}$ θα είχαμε άμεσο διαχωρισμό των BPP και $NEXP$ (ένα αποτέλεσμα που θα αφορούσε καθαρά ομοιόμορφες υπολογιστικές κλάσεις, αλλά θα είχε δειχθεί με τη βοήθεια ανομοιόμορφων κυκλωματικών κάτω φραγμάτων).

Κεφάλαιο 6

Συμπεράσματα και Επεκτάσεις

6.1 Εποπτεία

Είδαμε τα δύο μοντέλα υπολογισμού μαζί με τις αντίστοιχες ιδιότητες κάθε ενός. Συγκεκριμένα ο ομοιόμορφος (ρεαλιστικός) υπολογισμός, με αντιπρόσωπο τις μηχανές Turing, ή με κάποιο υπολογιστικά ισοδύναμο μοντέλο (δηλαδή μηχανές με τυχαιοκρατία, μη ντετερμινισμό, υπολογίσιμο μαντείο κ.ο.κ.) ήταν το κατάλληλο πεδίο για την εφαρμογή των μεθόδων διαγωνιοποίησης, εφόσον στηριζόταν σε πολύ απλές ιδιότητες μιας μηχανής, όπως ότι η λειτουργία της είναι πεπερασμένα κωδικοποιήσιμη και ότι μπορεί να προσομοιώνει άλλες (κάτι που ασφαλώς μπορεί να εφαρμοστεί σε οποιοδήποτε ρεαλιστικό και όσο το δυνατόν πιο πλήρες μοντέλο υπολογισμού, αφού απλώς αντιστοιχεί σε εντολοδοσία και εκτέλεση). Στο πεδίο αυτό δεν χρειάστηκε να εισέλθουμε σε λεπτομέρειες των μηχανών για να καταλήξουμε στο ζητούμενο, παρά μόνο στη σημασιολογική-λογική πλευρά της λειτουργίας τους. Εν τέλει αποδείχτηκε ότι αυτή η «υψηλή» θεώρηση δεν ήταν επαρκής για την απάντηση μεγάλων ανοιχτών ερωτημάτων όπως το $P =^{?} NP$ και ότι απαιτείται μία πιο λεπτομερής ενδοσκόπηση στις επί μέρους πράξεις μιας μηχανής καθώς και μία επιβολή μέτρων σε αυτά που μπορούν να υπολογίσουν με περιορισμένους πόρους (χώρο, χρόνο, μη ντετερμινισμό κ.ο.κ.).

Στη συνέχεια μελετήσαμε το μη ομοιόμορφο (και μη ρεαλιστικό) μοντέλο υπολογισμού, με αντιπρόσωπο τα Κυκλώματα. Το συγκεκριμένο πεδίο φάνηκε να είναι το πιο εύφορο για τη μελέτη αυτών που δε μπορούσε το προηγούμενο, των επί μέρους βημάτων δηλαδή μιας υπολογιστικής διαδικασίας και της θέσπισης ορίων στις δυνατότητες αυτών υπό διάφορους περιορισμούς. Μεγάλο προσόν του συγκεκριμένου μοντέλου είναι ότι όλος ο υπολογισμός βρίσκεται ενώπιον του μελετητή (σε μορφή λογικών πυλών) εν αντιθέσει με τις παραδοσιακές μηχανές, όπου είναι κωδικοποιημένος σε πεπερασμένη μορφή για κάθε είσοδο. Εν τούτοις, από ό,τι φάνηκε, ούτε αυτό το πεδίο είχε ελπίδες να παράξει

σημαντικά αποτελέσματα, καθώς μια ιδιότητα που θα ξεχώριζε τα πολυωνυμικά κυκλώματα από μεγαλύτερα (με τρόπο που να αντιστοιχεί τουλάχιστον σε κάποιο εύλογο κι εφικτό αποδεικτικό επιχείρημα) θα οδηγούσε στην κατάρριψη ευρέως αποδεκτών κρυπτογραφικών εικασιών.

Η λύση φάνηκε να έρχεται συνδυάζοντας αμφότερες τις μεθόδους σε μία. Πράγματι, έτσι, οι ιδιότητες και τα προτερήματα της μίας άρχισαν να συμπληρώνουν τα κενά και τις αδυναμίες της άλλης. Συγκεκριμένα η δημιουργία μίας απόδειξης που στηρίζεται τόσο σε ένα διαγώνιο όσο και σε ένα τεχνικό επιχείρημα, όπως αυτές που είδαμε προηγουμένως, είχαν τα εξής οφέλη:
Αφ΄ ενός ο τεχνικό επιχείρημα που εμπλεκόταν απέφευγε τους σκοπέλους της σχετικοποίησης, καθώς οι κυκλωματικές ιδιότητες είναι κατά κόρον μη φυσικοποιήσιμες. Ο λόγος που συμβαίνει αυτό είναι η έλλειψη ομοιομορφίας μεταξύ των κυκλωμάτων, το οποίο οδηγεί στο ότι η περιγραφή ενός κυκλώματος μπορεί να είναι είσοδος παρά μόνο σε ένα κύκλωμα με μεγαλύτερο πλήθος δυφίων εισόδου κι άρα σε ένα πρακτικά ανεξάρτητο κύκλωμα, με το οποίο προφανώς δε μπορούμε να εφαρμόσουμε κάποια αυτοαναφορά.
Από την άλλη, το περιεχόμενο διαγώνιο επιχείρημα, φαίνεται ότι μπορεί να αποφεύγει τους κινδύνους της φυσικοποίησης, καθώς η διαγωνιοποίηση είναι εγγενώς μη φυσικοποίησιμη ιδιότητα (κυρίως λόγω της έλλειψης πληθωριστικότητας, για τους λόγους που έχουμε ήδη δει παραπάνω). Ακόμη περισσότερο, μεγάλο παράγοντα, υπέρ αυτού, αποτελεί η ύπαρξη ενός ομοιόμορφου αλγορίθμου για ένα κύκλωμα, δένοντας έτσι εξ αρχής τις φύσεις της μη ομοιομορφίας με αυτήν της ομοιομορφίας κι επιτρέποντας την εφαρμογή του ύστερου αυτοαναφορικού επιχειρήματος.

Ο συνδυασμός λοιπόν και των δύο τεχνικών σε μία ενιαία, φαίνεται να αποφεύγει τις αδυναμίες αμφότερων. Ο ακριβής τρόπος που αυτό συμβαίνει είναι παρασκευάζοντας μη τετριμμένους αλγόριθμους για κυκλώματα (όπου υπεισέρχεται το τεχνικό - μη ομοιόμορφο κομμάτι) και στη συνέχεια από το γεγονός ότι είναι μη τετριμμένοι να καταλήγουμε, με μία κατάλληλη οργάνωση αυτών των υποθετικών κυκλωμάτων για άλλους αλγόριθμους (όπου υπεισέρχεται το διαγώνιο - ομοιόμορφο κομμάτι), σε άτοπο. Ο συνδυασμός δηλαδή αλγορίθμων για κυκλώματα και κυκλωμάτων για αλγορίθμους είναι αυτός που οδήγησε στα παραπάνω συμπεράσματα κάτω φραγμάτων.

Το ευτυχές είναι ότι όπως είδαμε στο προηγούμενο Κεφάλαιο, υπάρχει έτοιμη μια μεγάλη πληθώρα Θεωρημάτων, έτοιμα να εκμεταλλευτούν αλγορίθμους για κυκλώματα και να οδηγήσουν σε διαχωρισμό (κυρίως) του $NEXP$ από τυχούσες χαμηλές κυκλωματικές κλάσεις. Το δύσκολο κομμάτι που απομένει, ως αναμενόμενο, είναι το τεχνικό, δηλαδή η εκμετάλλευση συγκεκριμένων αδυναμιών μιας κλάσης κυκλωμάτων και η εξαγωγή από αυτές μη εξαντλητικών αλγορίθμων που τις εκμεταλλεύονται με τρόπο τέτοιον ώστε να υπολογίζουν διάφορες ιδιότητες τους σε μη τετριμμένο χρόνο. Ασφαλώς η ύπαρξη αυξημένης δυσκολίας σε αυτή την ενοποιητική τεχνική ήταν αναμενόμενη, αν την αναλογιστούμε ως το φυσικό ισοζύγιο στην εύρεση μιας λύσης που αποφεύγει τα σφάλματα των δύο προηγούμενων τεχνικών, συνδυάζοντας τις· εν τούτοις

το γεγονός ότι έχουμε μια επιτυχημένη εφαρμογή αυτού του παραδείγματος με το διαχωρισμό $NEXP \not\subseteq ACC^0$ (μετά από αρκετό καιρό, μάλιστα, που δεν υπήρχαν αντίστοιχα αποτελέσματα), δε μπορεί παρά να είναι σημάδι ότι πρόκειται για μια ουσιώδη και καρπερή μέθοδο.

6.2 Μελλοντικές Κατευθύνσεις

Υπάρχουν ορισμένες προφανείς κατευθύνσεις στις οποίες μπορούν να επεκταθούν τα παραπάνω αποτελέσματα. Πρώτα από όλα, να βρεθούν τεχνικοί αλγόριθμοι για υψηλότερες κυκλωματικές κλάσεις. Το Θεώρημα 5.3 μας εγγυάται ότι για οποιαδήποτε κλάση (γενικά υποσύνολο του $P_{/poly}$ - αν και υπάρχουν επεκτάσεις και για ψηλότερες) βρούμε έστω και λίγο πιο αποδοτικό αλγόριθμο, τότε έχουμε ήδη αποκλείσει το $NEXP$ από αυτήν. Επόμενο ευκολότερο βήμα φαίνεται να είναι το TC^0. Εν τούτοις οι τεχνικές που εφαρμόστηκαν για τον αλγόριθμο του ACC^0 δε φαίνεται να μπορούν να εφαρμοστούν άμεσα στο TC^0, εν μέρει επειδή οι πρώτες εκμεταλλεύονται το γεγονός ότι οι πύλες MOD_m αφορούν μόνο κάποια φραγμένη σταθερά m, ενώ στο TC^0 μπορούν να υπολογιστούν μέχρι και MOD_N συναρτήσεις με $N = poly(n)$. Ενδεικτικό του πόσο δύσκολα είναι συνήθως τέτοια τεχνικά επιχειρήματα ακόμη και για τόσο μικρές κλάσεις, είναι το γεγονός ότι για πολλά χρόνια πριν το κεντρικό αποτέλεσμα του Κεφαλαίου 5, θα μπορούσε με την τότε τρέχουσα γνώση να ισχύει ότι το E^{NP} μπορεί να γραφτεί με πολυωνυμικά κυκλώματα βάθους 3 στο $ACC^0(6)$ [Wil11, AB09, p.298] (κάτι που εξηγεί γιατί θεωρείται τόσο σπουδαίο το σχετικό αποτέλεσμα, αφού απέρριψε όλη την κλάση ACC^0 άνευ τυχόν περιορισμών). Εν τούτοις διάδοχος των προηγούμενων «εξωφρενικών» υποθέσεων είναι πλέον το TC^0 και το αντίστοιχο τεχνικό επιχείρημα φαίνεται να πρέπει να είναι αισθητά πιο δύσκολο (πόσο μάλλον για υψηλότερες κλάσεις όπως οι NC^1_{non-u}, ACC^1, NC_{non-u} κ.ο.κ.). Ενδιαφέρον φαίνεται να παρουσιάζει και η περίπτωση όπου οι υποτιθέμενοι τεχνικοί αλγόριθμοι γεννιούνται εξ αιτίας της αρχικής υπόθεσης ότι το $NEXP$ μοντελοποιείται από μικρά κυκλώματα. Σε κάθε περίπτωση, όμως, για λόγους που αναφέραμε, η προηγούμενη προσέγγιση δεν αναμένεται να αποφέρει καρπούς αν δε μπολιαστεί με κάποιο τεχνικό επιχείρημα, καθώς αλλιώς θα επρόκειτο για μία κατά βάση σχετικιστική απόδειξη.

Μια άλλη πιθανή προσέγγιση φαίνεται να είναι το να βρεθούν ισχυρότερα αποτελέσματα για το $NEXP$ ή παρόμοια αποτελέσματα για χαμηλότερες κλάσεις. Να σημειωθεί αρχικά ότι με διάφορες τεχνικές βελτιώσεις και με στενότερη μελέτη του προβλήματος, έχουν ήδη βρεθεί κάποια πιο στενά αποτελέσματα ως προς το πρώτο ζήτημα (συγκεκριμένα έχουν βρεθεί αντίστοιχα αποτελέσματα για ACC κυκλώματα με βάθος έως $o(\frac{logn}{loglogn})$ [CP16] ή για μέγεθος έως μία αριθμητική συνάρτηση, η οποία αν εφαρμοστεί τρεις φορές στον εαυτό της με πολυωνυμική μορφή εισόδου δίνει μια υποεκθετική συνάρτηση [Wil11] (περιλαμβάνει τις $n^{log^i n}$ και κάποιες ψηλότερες, αλλά επ ουδενί εκθετικές ή ημιεκθετικές συναρτήσεις)). Παραμένει ασφαλώς το ζήτημα αν μπορούν

να γίνουν ακόμη πιο στενά τα όρια, π.χ. αντίστοιχα του αποτελέσματος για το E^{NP}. Το μεγάλο εμπόδιο φαίνεται να είναι η έλλειψη κάποιου σχετικού αποτελέσματος για ύπαρξη συμπυκνώσιμων πιστοποιητικών για το $NEXP$, απλά ως απόρροια της μοντελοποίησης τους από μικρά κυκλώματα κάποιας κλάσης. Συγκεκριμένα θα θέλαμε μια βελτίωση της απόδειξης του Θεωρήματος 4.5, τέτοια ώστε να μπορεί να εφαρμοστεί με πιο στενές παραμέτρους (κάτι που πιθανόν να οδηγούσε και σε πιο στενά χαμηλά όρια για τη κλάση Σ_2^{EXP}, η οποία προς το παρόν πάσχει από το ίδιο πρόβλημα και γνωρίζουμε μόνο ότι δε μοντελοποιείται από κυκλώματα ημιεκθετικού μεγέθους). Ως προς την άλλη κατεύθυνση, προκύπτουν παρόμοια ζητήματα, αφού ανάγεται στην προηγούμενη με κάποιο επιχείρημα παραγεμίσματος.

Θα μπορούσε κανείς να αναρωτηθεί τι γίνεται με επίσης εκθετικές κλάσεις, αλλά χωρίς τον μη ντετερμινισμό, για παράδειγμα κάτω κυκλωματικά φράγματα για το EXP. Το μεγάλο πρόβλημα που προκύπτει, τότε, είναι ότι δε μπορούμε να ξέρουμε ποια είναι τα αντίστοιχα κυκλώματα που μοντελοποιούν το EXP και εφόσον δεν μπορούμε να τα μαντέψουμε, ο καλύτερος αλγόριθμος που έχουμε τώρα είναι να τα διασχίσουμε όλα (απαιτώντας επίσης εκθετικό χρόνο και απορρίπτοντας έτσι τη δυνατότητα υποθετικής κατάρριψης της χρονικής ιεραρχίας). Βλέπουμε, δηλαδή, ότι στην εν λόγω προσέγγιση τόσο ο μη ντετερμινισμός, όσο και ο εκθετικός χρόνος είναι προς το παρόν απαραίτητα στοιχεία, αλλά και το ότι να ανέβουμε σε πιο «εξελιγμένη» κλάση με πιο δυνατές πύλες είναι ένα μάλλον δύσκολο ζήτημα. Σε κάθε περίπτωση αυτό δε σημαίνει ότι κάποια από αυτές τις κατευθύνσεις είναι κλειστή (τουλάχιστον σίγουρα όχι με τον τρόπο που ήταν οι προηγούμενες δύο ατυχείς τεχνικές).

Από εκεί και πέρα έντονο ενδιαφέρον παρουσιάζει η περίπτωση του $MCSP$ που φαίνεται να είναι ένα κεντρικού ρόλου μεταπρόβλημα με πολλαπλές επιρροές και στενές συνδέσεις με τα κάτω κυκλωματικά φράγματα (οι οποίες εν μέρει είναι αναμενόμενες, ακριβώς εξαιτίας της φύσης του προβλήματος). Εν τούτοις, λόγω του υπαρξιακού τελεστή στην εκφώνηση του, κατά τη μελέτη του συγκεκριμένου προβλήματος ερχόμαστε συχνά αντιμέτωποι με εμπόδια ίδιας φύσεως με αυτά του SAT (αν και, όπως έχουμε πει, δεν υπάρχει μέχρι στιγμής απόδειξη για το αν είναι NP-πλήρες και υπάρχουν κάποιοι λόγοι που συνηγορούν υπέρ του ότι δεν είναι ή ότι έστω δεν υπάρχει εύκολη σχετική απόδειξη για αυτό). Σε κάθε περίπτωση, για τη γενική μορφή αυτού του προβλήματος, η βέλτιστη αντιμετώπιση που έχουμε μέχρι στιγμής είναι, δυστυχώς επίσης, παρόμοια αυτής που έχουμε απέναντι σε ένα μαύρο κουτί, όπου πρέπει να ψάξουμε ενδελεχώς κάθε πιθανότητα.

Ένα πιο ελπιδοφόρο πρόβλημα φαίνεται να είναι η περίπτωση του $\mathcal{C}-CAPP$. Για αρχή να σημειώσουμε ότι, όπως έχουμε δει, ισχύει πως $P = BPP$ εκτός κι αν το E μπορεί να γραφτεί με κυκλώματα μεγέθους $2^{o(n)}$ κι ακόμη περισσότερο $BPP \subseteq QuasiP$ εκτός κι αν το EXP γράφεται με κυκλώματα μεγέθους $2^{n^{o(1)}}$. Ένα αρκετά μεγάλο μέρος της επιστημονικής κοινότητας πιστεύει ήδη την πρώτη εικασία, αλλά ένα ακόμη μεγαλύτερο πιστεύει τη δεύτερη (δηλαδή

ότι το EXP δε γράφεται με υποεκθετικά κυκλώματα). Αυτό από μόνο του αποτελεί μια καλή ένδειξη της γενικής πεποίθησης ότι τα προβλήματα εκτίμησης του ποσοστού των άσσων (όπως το $CAPP$) είναι αρκετά πιο εύκολα από αυτά του υπολογισμού της ύπαρξης έστω ενός (όπως το SAT). Δεδομένης της ελαστικότητας του Θεωρήματος 5.7, όπου αρκεί να βρούμε αλγόριθμο για το $\mathcal{C}$ που να τρέχει σε απλώς υποεκθετικό χρόνο (ακόμη κι αν επιτρέψουμε μη ντετερμινισμό), έτσι ώστε να αποκλείσουμε το $NEXP$ από το $\mathcal{C}$, κάποιος που ενδιαφέρεται να προσεγγίσει το πρόβλημα από την τεχνική του πλευρά (αλγόριθμους για κυκλώματα), ίσως βρει πιο πρόσφορο έδαφος σε αυτό το κομμάτι εύλογα· αν όχι για τις γενικές κλάσεις (για τις οποίες άλλωστε, από τα παραπάνω, φαίνεται το $P_{/poly}$-$CAPP$ να έχει καλές πιθανότητες να είναι στο P), τότε έστω αρχικά για τις πιο χαμηλές κλάσεις, όπως π.χ. αυτές με υπογραμμικό (πολυλογαριθμικό) βάθος όπως την NC_{non-u}, ή ακόμη και για αυτές με σταθερό μόλις βάθος όπως την TC^0.

Σε κάθε περίπτωση, πέραν των παραπάνω κατευθύνσεων γύρω από την κεντρική δομή των Θεωρημάτων που παρουσιάστηκαν προηγουμένως, ελπιδοφόρα φαίνεται η γενική φιλοσοφία που περιβάλλει αυτά τα Θεωρήματα, η οποία δεν είναι άλλη από το συνδυασμό τεχνικών και λογικών επιχειρημάτων. Πράγματι, αν υπάρχει ένα ηθικό δίδαγμα από τη μέχρι τώρα πορεία μας, είναι ότι απλές αποδείξεις οι οποίες προσεγγίζουν μονοδιάστατα ένα πρόβλημα δεν επαρκούν για τα μεγάλα ανοιχτά ερωτήματα, αλλά και ότι ο συνδυασμός αυτών δε λύνει μόνο σε θεωρητικό-υποθετικό επίπεδο τα προβλήματα των προηγούμενων «απλοϊκών» προσεγγίσεων, αλλά ότι έχει ήδη αρχίσει να παράγει αποτελέσματα! Το ότι θα πρέπει να εμπιστευτούμε αυτή τη νέα προσέγγιση, ενισχύεται άλλωστε από το γεγονός ότι η ιδέα της είναι τόσο γενική (δεδομένου ότι απλώς συνδυάζει δύο από τις γενικότερες δυνατές προσεγγίσεις ενός θεωρητικού προβλήματος Πολυπλοκότητας με αυθαίρετη αναλογία μεταξύ τους), έτσι ώστε να είναι μάλλον ιδιαίτερα δύσκολο να μπορεί να βρεθεί ένα γενικό μοτίβο που να περιλαμβάνει ικανοποιητικά κάθε παραλλαγή της κι έτσι ώστε να μπορέσει να εξάγει από αυτό μια συγκεκριμένη αδυναμία (όπως συνέβη με τα σχετικιστικά και φυσικά επιχειρήματα που αργότερα ενσωματώθηκαν σε δύο αντίστοιχα γενικά μοτίβα, τα οποία αποδείχθηκαν ότι ήταν μη επαρκώς εκφραστικά). Επομένως το τελικό συμπέρασμα που θα έπρεπε να κρατήσουμε, είναι ότι ανεξάρτητα του ακριβούς περιβάλλοντος, όταν πρόκειται για αντιμετώπιση μεγάλων ερωτημάτων που αφορούν την ακριβή σχέση μεταξύ ανομοιόμορφων και ομοιόμορφων κλάσεων υπολογισμού, θα πρέπει αναπόφευκτα να αναμιγνύουμε στοιχεία και εργαλεία στις αποδείξεις που να αναφέρονται και να είναι κατάλληλα για τους επί μέρους διαχωρισμούς των υποκλάσεων κάθε μίας.

Βιβλιογραφία

[Aar05] Scott Aaronson. *Complexity Zoo*, 2005. Available at http://complexityzoo.uwaterloo.ca/Complexity_Zoo:L#l.2Fpoly.

[AB09] Sanjeev Arora and Boaz Barak. *Computational Complexity: A Modern Approach*. Cambridge University Press, New York, NY, USA, 1st edition, 2009.

[Adl78] Leonard Adleman. Two theorems on random polynomial time. In *Proceedings of the 19th Annual Symposium on Foundations of Computer Science*, SFCS '78, pages 75–83, Washington, DC, USA, 1978. IEEE Computer Society.

[AG91] Eric Allender and Vivek Gore. On strong separations from AC^0. In *Fundamentals of Computation Theory, 8th International Symposium, FCT '91, Gosen, Germany, September 9-13, 1991, Proceedings*, pages 1–15, 1991.

[Ajt83] M. Ajtai. $\sum_1^1$-formulae on finite structures. *Annals of Pure and Applied Logic*, 24(1):1–48, 1983.

[BBR94] David A. Mix Barrington, Richard Beigel, and Steven Rudich. Representing boolean functions as polynomials modulo composite numbers. *Computational Complexity*, 4(4):367–382, 1994.

[BBS86] Jose L. Balcázar, Ronald V. Book, and Uwe Schöning. The polynomial-time hierarchy and sparse oracles. *J. ACM*, 33(3):603–617, May 1986.

[BFL91] László Babai, Lance Fortnow, and Carsten Lund. Nondeterministic exponential time has two-prover interactive protocols. *Computational Complexity*, 1(1):3–40, 1991.

[BFNW93] László Babai, Lance Fortnow, Noam Nisan, and Avi Wigderson. Bpp has subexponential time simulations unless exptime has publishable proofs. *Comput. Complex.*, 3(4):307–318, October 1993.

[BFT98] Harry Buhrman, Lance Fortnow, and Thomas Thierauf. Nonrelativizing separations. In *Proceedings of the 13th Annual IEEE Conference on Computational Complexity, Buffalo, New York, USA, June 15-18, 1998*, pages 8–12, 1998.

[BGS75] Theodore Baker, John Gill, and Robert Solovay. Relativizations of the $\mathcal{P} =?\mathcal{NP}$ question. *SIAM Journal on Computing*, 4(4):431–442, 1975.

[BLT92] José L. Balcázar, Antoni Lozano, and Jacobo Torán. *The Complexity of Algorithmic Problems on Succinct Instances*, pages 351–377. Springer US, Boston, MA, 1992.

[BM84] Manuel Blum and Silvio Micali. How to generate cryptographically strong sequences of pseudo-random bits. *SIAM J. Comput.*, 13(4):850–864, November 1984.

[BSV14] Eli Ben-Sasson and Emanuele Viola. *Short PCPs with Projection Queries*, pages 163–173. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014.

[BT94] Richard Beigel and Jun Tarui. On ACC. *Comput. Complex.*, 4(4):350–366, October 1994.

[CCJK06] David A. Cohen, Martin C. Cooper, Peter G. Jeavons, and Andrei A. Krokhin. The complexity of soft constraint satisfaction. *Artificial Intelligence*, 170(11):983 – 1016, 2006.

[Chu85] Alonzo Church. *The Calculi of Lambda Conversion. (AM-6) (Annals of Mathematics Studies)*. Princeton University Press, Princeton, NJ, USA, 1985.

[CK02] Peter Clote and Evangelos Kranakis. *Boolean Functions and Computation Models*. Texts in Theoretical Computer Science. An EATCS Series. 2002.

[CLL+05] M. Charikar, E. Lehman, Ding Liu, R. Panigrahy, M. Prabhakaran, A. Sahai, and A. Shelat. The smallest grammar problem. *IEEE Transactions on Information Theory*, 51(7):2554–2576, July 2005.

[Coo71] Stephen A. Cook. The complexity of theorem-proving procedures. In *Proceedings of the Third Annual ACM Symposium on Theory of Computing*, STOC '71, pages 151–158, New York, NY, USA, 1971. ACM.

[Coo72] Stephen A. Cook. A hierarchy for nondeterministic time complexity. In *Proceedings of the Fourth Annual ACM Symposium*

on Theory of Computing, STOC '72, pages 187–192, New York, NY, USA, 1972. ACM.

[CP16] S. Chen and P. A. Papakonstantinou. Depth-reduction for composites. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 99–108, Oct 2016.

[FM05] Gudmund Skovbjerg Frandsen and Peter Bro Miltersen. Reviewing bounds on the circuit size of the hardest functions. *Information Processing Letters*, 95(2):354–357, 2005.

[For94] Lance Fortnow. The role of relativization in complexity theory. *Bulletin of the European Association for Theoretical Computer Science*, 52:52–229, 1994.

[FSS84] Merrick Furst, James B. Saxe, and Michael Sipser. Parity, circuits, and the polynomial-time hierarchy. *Mathematical systems theory*, 17(1):13–27, 1984. Preliminary version FOCS '81.

[Für82] Martin Fürer. The tight deterministic time hierarchy. In *Proceedings of the Fourteenth Annual ACM Symposium on Theory of Computing*, STOC '82, pages 8–16, New York, NY, USA, 1982. ACM.

[GGM86] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. *J. ACM*, 33(4):792–807, August 1986.

[Goo12] R.L. Goodstein. *Boolean Algebra*. Dover Books on Mathematics. Dover Publications, 2012.

[Has86] J Hastad. Almost optimal lower bounds for small depth circuits. In *Proceedings of the Eighteenth Annual ACM Symposium on Theory of Computing*, STOC '86, pages 6–20, New York, NY, USA, 1986. ACM.

[HCC+92] J. Hartmanis, R. Chang, S. Chari, D. Ranjan, and P. Rohatgi. Relativization: A revisionistic retrospective. *Bulletin of the European Association for Theoretical Computer Science*, 47:144–153, 1992.

[HS65] J. Hartmanis and R. E. Stearns. On the computational complexity of algorithms. *Transactions of the American Mathematical Society*, 117:285–306, 1965.

[HS66] F. C. Hennie and R. E. Stearns. Two-tape simulation of multitape turing machines. *J. ACM*, 13(4):533–546, October 1966.

[IKW02] Russell Impagliazzo, Valentine Kabanets, and Avi Wigderson. In search of an easy witness: Exponential time vs. probabilistic polynomial time. *J. Comput. Syst. Sci.*, 65(4):672–694, December 2002.

[IM02] Kazuo Iwama and Hiroki Morizumi. An explicit lower bound of 5n - o(n) for boolean circuits. In *Proceedings of the 27th International Symposium on Mathematical Foundations of Computer Science*, MFCS '02, pages 353–364, London, UK, UK, 2002. Springer-Verlag.

[IW97] Russell Impagliazzo and Avi Wigderson. P = bpp if e requires exponential circuits: Derandomizing the xor lemma. In *Proceedings of the Twenty-ninth Annual ACM Symposium on Theory of Computing*, STOC '97, pages 220–229, New York, NY, USA, 1997. ACM.

[Kab03] Valentine Kabanets. Cmpt 710 - complexity theory: Lecture 16, October 2003.

[Kan82] Ravi Kannan. Circuit-size lower bounds and non-reducibility to sparse sets. *Information and Control*, 55:40–56, 1982.

[KC00] Valentine Kabanets and Jin-Yi Cai. Circuit minimization problem. In *Proceedings of the Thirty-second Annual ACM Symposium on Theory of Computing*, STOC '00, pages 73–79, New York, NY, USA, 2000. ACM.

[KI03] Valentine Kabanets and Russell Impagliazzo. Derandomizing polynomial identity tests means proving circuit lower bounds. In *Proceedings of the Thirty-fifth Annual ACM Symposium on Theory of Computing*, STOC '03, pages 355–364, New York, NY, USA, 2003. ACM.

[KL80] Richard M. Karp and Richard J. Lipton. Some connections between nonuniform and uniform complexity classes. In *Proceedings of the Twelfth Annual ACM Symposium on Theory of Computing*, STOC '80, pages 302–309, New York, NY, USA, 1980. ACM.

[KL82] R. Karp and R. Lipton. Turing machines that take advice. *L'Enseignement Mathématique*, 28(2):191–209, 1982. A preliminary version appeared in STOC 1980.

[Kle37] S. C. Kleene. General recursive functions of natural numbers. *Journal of Symbolic Logic*, 2(1):38–38, 1937.

[Kol63] AN Kolmogorov. Sankhyâ. *Series A*, 25:369–376, 1963.

[KV87] Marek Karpinski and Rutger Verbeek. On the monte carlo space constructible functions and separation results for probabilistic complexity classes. *Information and Computation*, 75(2):178 – 189, 1987.

[KY00] J. C. Kieffer and En-Hui Yang. Grammar-based codes: a new class of universal lossless source codes. *IEEE Transactions on Information Theory*, 46(3):737–754, May 2000.

[Lad75] Richard E. Ladner. On the structure of polynomial time reducibility. *J. ACM*, 22(1):155–171, January 1975.

[Lau83] Clemens Lautemann. BPP and the polynomial hierarchy. *Inf. Process. Lett.*, 17(4):215–217, 1983.

[Lip94] Richard J. Lipton. Some consequences of our failure to prove non-linear lower bounds on explicit functions. In *Proceedings of the Ninth Annual Structure in Complexity Theory Conference, Amsterdam, The Netherlands, June 28 - July 1, 1994*, pages 79–87, 1994.

[MM11] Cristopher Moore and Stephan Mertens. *The Nature of Computation*. Oxford University Press, Inc., New York, NY, USA, 2011.

[NW94] Noam Nisan and Avi Wigderson. Hardness vs randomness. *Journal of Computer and System Sciences*, 49(2):149 – 167, 1994.

[Pap94] Christos H. Papadimitriou. *Computational Complexity*. Addison Wesley Pub. Co., Massachussetts, 1994.

[Par14] William R. Parks. *Introduction to Boolean Algebra and Switching Circuits - Volume 4*. William R. Parks, 1st edition, 2014.

[Raz87] A. A. Razborov. Lower bounds on the size of bounded depth circuits over a complete basis with logical addition. *Mathematical notes of the Academy of Sciences of the USSR*, 41(4):333–338, 1987.

[RR97] Alexander A Razborov and Steven Rudich. Natural proofs. *Journal of Computer and System Sciences*, 55(1):24 – 35, 1997.

[Sel12] A.L. Selman. *Complexity Theory Retrospective: In Honor of Juris Hartmanis on the Occasion of His Sixtieth Birthday, July 5, 1988*. Springer New York, 2012.

[SFM78] Joel I. Seiferas, Michael J. Fischer, and Albert R. Meyer. Separating nondeterministic time complexity classes. *J. ACM*, 25(1):146–167, January 1978.

[Sha49a] C. E. Shannon. Communication theory of secrecy systems. *The Bell System Technical Journal*, 28(4):656–715, Oct 1949.

[Sha49b] Claude. E. Shannon. The synthesis of two-terminal switching circuits. *Bell System Technical Journal*, 28(1):59–98, 1949.

[Sha92] Adi Shamir. IP = PSPACE. *J. ACM*, 39(4):869–877, October 1992.

[Sip78] Michael Sipser. Halting space-bounded computations. In *Proceedings of the 19th Annual Symposium on Foundations of Computer Science*, SFCS '78, pages 73–74, Washington, DC, USA, 1978. IEEE Computer Society.

[Sip83] Michael Sipser. A complexity theoretic approach to randomness. In *Proceedings of the Fifteenth Annual ACM Symposium on Theory of Computing*, STOC '83, pages 330–335, New York, NY, USA, 1983. ACM.

[Smo87] R. Smolensky. Algebraic methods in the theory of lower bounds for boolean circuit complexity. In *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing*, STOC '87, pages 77–82, New York, NY, USA, 1987. ACM.

[Sto76] Larry J. Stockmeyer. The polynomial-time hierarchy. *Theoretical Computer Science*, 3(1):1 – 22, 1976.

[STV01] Madhu Sudan, Luca Trevisan, and Salil Vadhan. Pseudorandom generators without the xor lemma. *Journal of Computer and System Sciences*, 62(2):236 – 266, 2001.

[Tod91] Seinosuke Toda. Pp is as hard as the polynomial-time hierarchy. *SIAM J. Comput.*, 20(5):865–877, October 1991.

[Tse68] G. S. Tseitin. On the complexity of derivation in propositional calculus. *Studies in constructive mathematics and mathematical logic*, 2(115-125):10–13, 1968.

[Tur36] Alan Turing. On computable numbers, with an application to the entscheidungsproblem. *Proceedings of the London Mathematical Society*, 42(1):230–265, 1936.

[Uma01] Ch. Umans. The minimum equivalent DNF problem and shortest implicants. *Journal of Computer and System Sciences*, 63(4):597–611, 2001.

[Uma03] Christopher Umans. Pseudo-random generators for all hardnesses. *Journal of Computer and System Sciences*, 67(2):419 – 440, 2003. Special Issue on {STOC} 2002.

[VV86] L. G. Valiant and V. V. Vazirani. Np is as easy as detecting unique solutions. *Theor. Comput. Sci.*, 47(1):85–93, November 1986.

[Wil11] R. Williams. Non-uniform acc circuit lower bounds. In *2011 IEEE 26th Annual Conference on Computational Complexity*, pages 115–125, June 2011.

[Wil13] Ryan Williams. Improving exhaustive search implies superpolynomial lower bounds. *SIAM Journal on Computing*, 42(3):1218–1244, 2013.

[Wil14] R. Williams. Algorithms for circuits and circuits for algorithms. In *2014 IEEE 29th Conference on Computational Complexity (CCC)*, pages 248–261, June 2014.

[Yao82] Andrew C. Yao. Theory and application of trapdoor functions. In *Proceedings of the 23rd Annual Symposium on Foundations of Computer Science*, SFCS '82, pages 80–91, Washington, DC, USA, 1982. IEEE Computer Society.

[Yao90] A.C.-C. Yao. On ACC and threshold circuits. *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, 00:619–627 vol.2, 1990.

[Zac86] Stathis Zachos. *Probabilistic quantifiers, adversaries, and complexity classes : An overview*, pages 383–400. Springer Berlin Heidelberg, Berlin, Heidelberg, 1986.

[ZP03] S. Zachos and A. Pagourtzis. Combinatory complexity: Operators on complexity classes. In *Proceedings of 4th Panhellenic Logic Symposium*, PLS 2003, July 2003.

Παράρτημα Α΄

Απόδειξη Τεχνικών Λημμάτων

Α΄.1 Προσομοίωση Εκτέλεσης Μηχανής Turing σε γραμμολογαριθμικό χρόνο.

Ένα από τους κυριότερους λόγους της πλεονάζουσας καθυστέρησης κατά τη προσομοίωση μιας μηχανής Turing (από μία άλλη) είναι η επαναλαμβανόμενη αντιγραφή του συνόλου της διαμόρφωσης της για κάθε βήμα, ενώ οι ουσιώδεις αλλαγές μεταξύ δύο διαδοχικών στιγμών είναι πεπερασμένου μήκους. Θα παρουσιάσουμε, λοιπόν, αρχικά έναν γενικό τρόπο αποθήκευσης του περιεχόμενου της ταινίας, με τον οποίο μπορούμε να μη μετακινούμε κάθε φορά όλα τα κελιά για κάθε ολίσθηση, αλλά μόνο όσα γειτονικά κελιά χρειάζονται με τρόπο τέτοιο ώστε να αρκεί συνολικός χρόνος $T(n)logT(n)$ και στη συνέχεια αναφέρουμε ορισμένα κεντρικά πεδία στα οποία έχει εφαρμογή αυτή η ιδέα δίνοντας στενότερους χρόνους προσομοίωσης σε αυτά.

Πράγματι, έστω ότι εισάγουμε ένα νέο σύμβολο $\boxtimes$ που δεν ανήκει στο αλφάβητο της μηχανής υπό προσομοίωση και το οποίο χρησιμοποιείται απλά ως παραγέμισμα. Για να το ξεχωρίζουμε από τα υπόλοιπα κελιά, θα το ονομάζουμε ανούσιο και όλα τα υπόλοιπα που περιέχουν σύμβολα του αρχικού αλφαβήτου θα τα αποκαλούμε ουσιώδη. Θα χωρίσουμε τώρα τα κελιά της ταινίας εισόδου εκατέρωθεν του κεντρικού κελιού, όπου θεωρούμε ότι βρίσκεται μόνιμα η κεφαλή, σε νοητές ζώνες οι οποίες συμβολίζονται με L_i, R_i για $i \in \{0, 1, ...logT\}$, όπου ο δείκτης i συμβαδίζει με την απόσταση που βρίσκεται η εκάστοτε ζώνη (δηλαδή η ταινία κάθε στιγμή περιγράφεται από τη συμβολοσειρά $L_{logT} \circ ... \circ L_1 \circ L_0 \circ H \circ R_0 \circ R_1 \circ ... \circ R_{logT}$, όπου H το κελί της κεφαλής (μήκους 1)). Για τις ζώνες αυτές ισχύουν οι ακόλουθες αναλλοίωτες συνθήκες, καθ΄ όλη τη διάρκεια εκτέλεσης:

1. Αν αφαιρέσουμε τα ανούσια σύμβολα από τη ταινία προσομοίωσης, η συμβολοσειρά που απομένει είναι αυτή που θα είχε η μηχανή την εκάστοτε

χρονική στιγμή (με τη κεφαλή στο κεντρικό H).

2. Οι ζώνες L_i, R_i έχουν ακριβώς $2 * 2^i$ σύμβολα, για κάθε i.

3. Κάθε ζώνη είναι γεμάτη, μισή ή άδεια (δηλαδή δεν έχει κανένα, έχει ακριβώς τα μισά ή έχει μόνο ανούσια σύμβολα αντίστοιχα).

4. Το άθροισμα των ουσιωδών συμβόλων των L_i και R_i ισούται με $2 * 2^i$, για κάθε i.

Γίνεται εύκολα εμφανές ότι υπάρχουν κάθε στιγμή τουλάχιστον $2T-1$ ουσιώδη κελιά κι άρα ο χώρος είναι επαρκής για την αναπαράσταση της κατάστασης της ταινίας καθ' όλη τη διάρκεια εκτέλεσης της. Επίσης αρχικά φροντίζουμε στην αρχική διαμόρφωση όλες οι ζώνες να είναι μισές (γίνεται εύκολα σε χρόνο $O(T)$). Από εκεί και πέρα ακολουθούμε τους παρακάτω κανόνες:

- Αν η πράξη προς εκτέλεση είναι εγγραφή νέου στοιχείου, τότε αυτό γράφεται απλώς στο H σε $O(1)$ και οι αναλλοίωτες ασφαλώς διατηρούνται.

- Αν η πράξη προς εκτέλεση είναι μετακίνηση της κεφαλής προς τα αριστερά, τότε θα ολισθήσουμε την ταινία προς τα δεξιά ως εξής:
  Βρίσκουμε το πρώτο j για το οποίο η ζώνη L_j δεν είναι άδεια και μετακινούμε το δεξιότερο ουσιώδες της σύμβολο στο H και τα $2^j - 1$ επόμενα ουσιώδη σύμβολα της (έχει τουλάχιστον τόσα λόγω της αντίστοιχης αναλλοίωτης) στις j χαμηλότερες L_i (με $i < j$) με τη σειρά που ήταν, έτσι ώστε να γίνουν όλες μισές (πράγματι $2^0 + 2^1 + ... + 2^{j-1} = 2^j - 1$). Αντίστοιχα έχουμε, λόγω των αναλλοίωτων, ότι όλες οι R_i για $i < j$ ήταν γεμάτες και η R_j ήταν το πολύ μισή, άρα μαζί με το τέως στοιχείο του H μετακινούμε τα $2^j - 1 + 1$ ουσιώδη σύμβολα στις επόμενες θέσεις (με τη σειρά που ήταν) έτσι ώστε όλες οι R_i (με $i < j$) να είναι μισές και η R_j να αποκτήσει 2^j παραπάνω ουσιώδη σύμβολα. Εύκολα βλέπουμε ότι όλες οι αναλλοίωτες διατηρούνται και ότι η ολίσθηση μπορεί να γίνει σε χρόνο $O(2^j)$.

- Αν η πράξη προς εκτέλεση είναι μετακίνηση της κεφαλής προς τα αριστερά, τότε επαναλαμβάνουμε συμμετρικά και προς την αντίθετη κατεύθυνση τη παραπάνω διαδικασία.

Έχουμε ότι για τις εγγραφές, συνολικά δε θα χρειαστεί παραπάνω από $O(T(n))$ χρόνος. Όσον αφορά τις ολισθήσεις, έχουμε ότι όποτε ο δείκτης j είναι αυτός που βρίσκεται ως ο ελάχιστος μη κενός σε μια ολίσθηση, σημαίνει ότι στο επόμενο βήμα όλες οι χαμηλότερες ζώνες είναι μισές κι άρα θα πρέπει να γίνουν τουλάχιστον 2^j ολισθήσεις της ίδιας φοράς προτού επιλεγεί ξανά αυτός ο δείκτης. Άρα αυτό θα συμβεί το πολύ $T/2^j$ φορές, οπότε αθροίζοντας για κάθε

δυνατό j παίρνουμε ότι θα χρειαστεί συνολικός χρόνος

$$\sum_{j=0}^{logT} \frac{T}{2^j} * O(2^j) = O(TlogT)$$

κι άρα πράγματι όλη η προσομοίωση μπορεί να γίνει σε τόσο χρόνο.

Μια πρώτη εφαρμογή αυτής της μεθόδου είναι η προσομοίωση μιας μηχανής $k > 1$ ταινιών από μία μηχανή μίας ταινίας σε χρόνο $O(TlogT)$. Πράγματι τοποθετούμε το εκάστοτε κελί κάθε ταινίας υπό προσομοίωση διαδοχικά το ένα μετά το άλλο για κάθε ταινία, έτσι ώστε στο αντίστοιχο H να υπάρχουν τα αντίστοιχα κελιά των k κεφαλών της ταινίας (κι όμοια για τα κελιά των υπόλοιπων ζωνών), οπότε κάθε ολίσθηση μιας ταινίας κοστίζει πλέον το πολύ $O(k * 2^j) = O(2^j)$ κι άρα ο τελικός αλγόριθμος προσομοίωσης είναι πράγματι χρόνου $O(TlogT)$.

Εκμεταλλευόμενοι την παραπάνω παρατήρηση για το πότε θα ξαναεπιλεγεί το j ως ο ελάχιστος δείκτης μη άδειας ζώνης, μπορούμε να μεταφέρουμε τη παραπάνω προσομοίωση σε ένα κύκλωμα το οποίο αντί να αντιγράφει σε κάθε στήλη όλα τα κελιά της προηγούμενης διαμόρφωσης, να φέρνει μόνο τα κελιά της j-οστής ζώνης κάθε $T/2^j$ στήλες. Το συνολικό κύκλωμα είναι ασφαλώς $O(TlogT)$ μεγέθους και η ορθότητα του προκύπτει άμεσα από τα παραπάνω. Μεγάλης σημασίας αποτελεί επίσης το γεγονός ότι το κύκλωμα αυτό είναι ομοιόμορφο με την έννοια ότι για τη περίπτωση που αφορά τη προσομοίωση μιας μεγάλης κλάσης (π.χ. της EXP) τότε μπορούμε εύκολα με είσοδο το δείκτη μιας πύλης να επιστρέψουμε σε χρόνο πολυωνυμικό (ως προς το μήκος του δείκτη κι άρα και του στιγμιοτύπου) τις πληροφορίες αυτής της πύλης (τι τύπου είναι και τους δείκτες των πυλών με τις οποίες επικοινωνεί).

Τέλος, μπορούμε σε κάθε περίπτωση, εισάγοντας φρέσκιες μεταβλητές για κάθε σύρμα του παραπάνω ομοιόμορφου κυκλώματος (ακριβώς με το τρόπο που παρουσιάζεται στην Απόδειξη του Θεωρήματος 5.3), να παράξουμε μια (ομοιόμορφη) έκφραση SAT μήκους $TlogT$ η οποία να είναι ικανοποιήσιμη αν και μόνο αν η προσομοίωση οδηγεί σε αποδοχή.

Α΄.2 Απόδειξη του Λήμματος Εναλλαγής (3.5.1)

Γνωρίζουμε ότι για κάθε λογική συνάρτηση, μεγιστόρος είναι μία απονομή σε ένα υποσύνολο των μεταβλητών εισόδου, η οποία σταθεροποιεί τη προκύπτουσα συνάρτηση (μετά τη σταθεροποίηση αυτών των μεταβλητών σε αυτή την απονομή) σε 0, ενώ αντίστοιχα ελαχιστόρος όταν τη σταθεροποιεί σε 1. Είναι εύκολο να δείξουμε την ισχύ της ακόλουθης πρότασης:

Πρόταση Α΄.2.1. *Μία συνάρτηση γράφεται σε $s-DNF$ αν και μόνο αν κάθε ελαχιστικός ελαχιστόρος της είναι μεγέθους το πολύ s (όπου ελαχιστικός ελαχιστόρος είναι ο ελαχιστόρος που δεν έχει υποσύνολο κάποιον ελαχιστόρο και αντίστοιχα ορίζουμε τον ελαχιστικό μεγιστόρο – τούδε και στο εξής θα θεωρούμε ότι οι ελαχιστόροι και μεγιστόροι είναι ελαχιστικοί).*

Απόδειξη. Αν γράφεται σε $s-DNF$, τότε γράφεται και σε κανονικοποιημένη $s-DNF$ που σημαίνει ότι δεν μπορούμε να εφαρμόσουμε σε κάποιο σημείο προσεταιριστική ιδιότητα ώστε να ελαττώσουμε το μήκος της. Τότε για κάθε ελαχιστόρο έχουμε ότι είτε καθορίζει όλες τις τιμές των μεταβλητών ενός όρου AND (και επειδή είναι ελαχιστικός μόνο αυτές) κι άρα είναι πράγματι μεγέθους το πολύ s, είτε διέρχεται από πολλούς όρους AND και οι προκύπτοντες όροι ισούνται με ταυτότητα. Όμως τότε θα μπορούσαμε να εφαρμόσουμε προσεταιριστική ιδιότητα με κοινό παράγοντα τις μεταβλητές που αντιστοιχούν στην απονομή του ελαχιστόρου και να μειώσουμε το μέγεθος της $s-DNF$ που όπως είπαμε έχουμε φροντίσει να μη μπορεί να συμβεί.
Αντίστροφα αν μία λογική συνάρτηση έχει ελαχιστόρους μεγέθους το πολύ s, τότε ισούται με την $s-DNF$ που προκύπτει από την OR όλων αυτών των ελαχιστόρων (πράγματι όταν αληθεύει ένας ελαχιστόρος, αληθεύει και η συνάρτηση, αλλά και αντίστροφα όταν αληθεύει η συνάρτηση αληθεύει και τουλάχιστον ένας ελαχιστόρος, το οποίο πρέπει να συμβαίνει αλλιώς οι ελαχιστόροι δεν θα ήταν ελαχιστικοί). ■

Κλασικά η παραπάνω πρόταση μπορεί να αναχθεί στη δυϊκή της πλευρά για τις $s-CNF$ προτάσεις. Λόγω της ισοδυναμίας, έχουμε λοιπόν ότι μία λογική συνάρτηση δεν μπορεί να γραφτεί σε μορφή $s-CNF$ αν και μόνο αν έχει έναν μεγιστόρο μεγέθους τουλάχιστον $s+1$.

Θα πρέπει τώρα να βρούμε ένα άνω φράγμα στο πλήθος των απονομών της $k-DNF$ μεγέθους $t=(1-q)*n$ των οποίων η προκύπτουσα συνάρτηση έχει τουλάχιστον έναν μεγιστόρο μεγέθους $s+1$. Συμβολίζουμε για ευκολία με $f|P$ την προκύπτουσα συνάρτηση μετά την σταθεροποίηση της απονομής P στην f και με $f|PR$ το $(f|P)|R$ όπου P,R ξένες απονομές ασφαλώς. Έστω λοιπόν T η σταθεροποίηση κάποιων t μεταβλητών. Για να μην έχει η $f|t$, σημαίνει ότι η t δεν την κάνει 0 όλους τους όρους της (στη $k-DNF$ μορφή της) αλλά δεν κάνει και κανέναν 1 (αλλιώς θα προέκυπτε σταθερή συνάρτηση με προφανή $s-CNF$ απεικόνιση). Υπάρχει επίσης μια ελαχιστική απονομή

S μεγέθους s, τέτοια ώστε $f|RS \equiv 0$ αλλά $f|RS' \not\equiv 0$ για κάθε $S' \subsetneq S$. Θεωρώντας την f ταξινομημένη (π.χ. λεξικογραφικά) σχηματίζουμε την S μεγέθους $s+1$ ως εξής: Σε κάθε βήμα θεωρούμε την απονομή s_i η οποία θέτει τιμές στις μεταβλητές του πρώτου όρου που δεν έχει μηδενιστεί από το $TS_1S_2...S_{i-1}$ έτσι ώστε $S_1S_2...S_{i-1}S_i \subseteq S$ (με άλλα λόγια προχωράμε την κατασκευή της S ίσα κατά τις μεταβλητές του πρώτου όρου που δεν μηδενίστηκε (οπότε και τώρα μηδενίζεται)) και παράλληλα ορίζουμε τη μοναδική απονομή R_i που δίνει σε αυτές τις μεταβλητές του όρου τιμή τέτοια ώστε να μη μηδενίζεται εξαιτίας αυτών. Η κατασκευή αυτή κάποτε τελειώνει με τρόπο τέτοιο ώστε $S_1S_2...S_m \equiv S$ και τα $S_1, ..., S_m$ να έχουν δώσει απονομή σε ακριβώς $s+1$ μεταβλητές (οπότε το ίδιο θα έχουν κάνει και οι $R_1, ..., R_m$).

Κατά κάποιο τρόπο το $TR_1R_2...R_m$ αν δεν δινόταν σκέτο, αλλά είχε κάποιους δείκτες που ξεχωρίζουν τις επιμέρους απονομές μαρτυράει ποιες είναι οι απονομές T και S κι άρα καθορίζει μονοσήμαντα την T που πρέπει να προσμετρήσουμε. Για να το κάνουμε αυτό και επίσημα, θα πρέπει να προσθέσουμε και μια πληροφορία που μαρτυράει ακριβώς τα S_i, R_i. Η μέθοδος που ακολουθούμε έχει ως εξής: Εφαρμόζουμε στην f το $TR_1R_2...R_m$ και παίρνουμε τον πρώτο μη μηδενικό όρο. Εκεί εμπεριέχεται εκ κατασκευής το S_1 και για να ξέρουμε ακριβώς ποιο είναι, αποθηκεύουμε την θέση των μεταβλητών του S_1 μαζί με τις τιμές που θέτει σε αυτές (και θέτουμε κι ένα διαχωριστικό για την επόμενη πληροφορία). Για να το κάνουμε αυτό επειδή η f βρίσκεται σε $k-DNF$ μορφή αρκεί να χρησιμοποιήσουμε $(logk+1)*|S_1|$ δυφία. Έχουμε πλέον εντοπίσει τα R_1, S_1 και είμαστε σε θέση να εφαρμόσουμε στην f το $TS_1R_2...R_m$. Όμως τότε επαναλαμβάνοντας τη διαδικασία, ο πρώτος όρος που δε μηδενίζεται μας αρκεί ώστε σε συνδυασμό με τις αντίστοιχες πληροφορίες χώρου $(logk+1)*|S_2|$ να εντοπίσουμε και τα R_2, S_2 κ.ο.κ. Εν τέλει θα έχουμε «αναβιώσει» ολόκληρη την S και R και από αυτήν καθορίζεται προφανώς μονοσήμαντα το T. Το μόνο που χρειαστήκαμε ήταν το $TR_1R_2...R_m$ μαζί με μία πληροφορία συνολικού χώρου της τάξης του $(logk+1)*(|S_1|+|S_2|+...+|S_m|) = O(s*logk)$.

Ανακεφαλαιώνοντας έχουμε αμφιμονοσήμαντη αντιστοίχιση των απονομών T των οποίων η $f|T$ δεν επιδέχεται $s-CNF$ μορφή, με ένα υποσύνολο των απονομών $f|TR$ μαζί με μία πληροφορία μεγέθους $O(s*logk)$ όπου R απονομή μεγέθους s. Έχουμε ωστόσο ότι το πλήθος των τυχαίων απονομών μεγέθους l είναι γενικά

$$\binom{n}{l} * 2^l$$

οπότε εν τέλει το πλήθος των ζητούμενων απονομών φράσσεται από το

$$\binom{n}{t+s} * 2^{t+s} * 2^{C''slogk} = \binom{n}{t+s} * 2^t * k^{C's}$$

Η τελική πιθανότητα που ζητάμε λοιπόν προκύπτει αν διαιρέσουμε με το $\binom{n}{t}*2^t$,

οπότε και έχουμε ότι φράσσεται από το

$$\frac{\binom{n}{t+s}}{\binom{n}{t}} * k^{C's}$$

Για να ολοκληρώσουμε την απόδειξη, παρατηρούμε ότι επειδή παίρνουμε $t = n - n^{\epsilon}$ (με $\epsilon \geq 2$)έχουμε

$$\binom{n}{t+s} \leq \binom{n}{t} * \left(\frac{n-t}{t}\right)^{s} \leq \binom{n}{t} * \left(\frac{n^{\epsilon}}{n-n^{\epsilon}}\right)^{s} \leq \binom{n}{t} * \left(\frac{1}{n}\right)^{s/2} \leq \binom{n}{t} * q^{s/2}$$

οπότε πράγματι έχουμε εν τέλει ότι η εν λόγω πιθανότητα φράσσεται για κάποια σταθερά C από την ποσότητα $\left(q * k^{C}\right)^{s/2}$ και η απόδειξη ολοκληρώθηκε.

Α΄.3 Απόδειξη του Λήμματος Κατασκευής $SYM+$ Κυκλωμάτων για το ACC^0 (3.7.1)

Η απόδειξη που θα παρουσιάσουμε προσομοιάζει ιδιαίτερα τη μέθοδο δειγματοληψίας και αποτυχαιοποίησης που συναντάμε στην απόδειξη του Θεωρήματος του Toda [Tod91]. Θεωρούμε και πάλι ότι το κύκλωμα βρίσκεται σε μορφή δέντρου με ένα είδος πυλών ανά στρώμα κ.ο.κ. όπως είχαμε και στην απόδειξη του Θεωρήματος 3.5. Επίσης θεωρούμε για απλότητα ότι έχουμε μόνο πύλες τύπου mod_p για πρώτους p (αλλά θα φανεί κατά τη διάρκεια της απόδειξης ότι μπορούμε να το επεκτείνουμε φυσικά για κάθε αριθμό) και μόνο πύλες OR (οι NOT προκύπτουν ως $mod_p(1, ..., 1, x)$ (όπου έχουμε δώσει $p-1$ άσσους) και οι AND με κανόνα de Morgan από τις άλλες δύο).

Προτού προχωρήσουμε, επειδή όπως θα φανεί στη συνέχεια θα είναι κρίσιμο το να φράξουμε τον εισάριθμο των πυλών OR από μία ποσότητα της τάξης του $polylog(S)$, θα δείξουμε πρώτα την ακόλουθη πρόταση:

Πρόταση Α΄.3.1. *Για το $\epsilon\nu$ λόγω C, υπάρχει ένα ισοδύναμο κύκλωμα μεγέθους $S^{O(log^2 S)}$ και βάθους το πολύ $4d$, το οποίο χρησιμοποιεί την ίδια βάση με εξαίρεση μία πύλη MAJ στην κορυφή και με την ιδιότητα κάθε πύλη OR να έχει εισάριθμο το πολύ $O(log^2 S)$.*

Απόδειξη. Θα το κάνουμε αυτό, δείχνοντας ότι υπάρχει ένα σύνολο $\mathcal{D}$ μεγέθους $|\mathcal{D}| = O(S^{O(log^2 S)})$ από ACC^0 κυκλώματα μεγέθους το πολύ $O(S^2)$ και βάθους $3d + O(1)$ όπου χρησιμοποιούν την ίδια βάση πυλών, έχουν τον επιθυμητό εισάριθμο στις OR και για κάθε x τουλάχιστον τα μισά κυκλώματα του $\mathcal{D}$ να συμφωνούν με την έξοδο του C. Αν το δείξουμε αυτό, τότε στη συνέχεια απλά παίρνουμε τη MAJ όλων αυτών των κυκλωμάτων και έχουμε άμεσα το ζητούμενο.

Αποδεικνύεται ωστόσο [VV86] ότι αν διαλέξουμε τυχαία μια συνάρτηση κατακερματισμού από το $\{0,1\}^{2^k}$ (όπου $k = O(logS)$ αφού κάθε πύλη OR θα έχει το πολύ S εισόδους) στο $\{0,1\}$ από ένα κατάλληλο σύνολο $\mathcal{H}$ που περιέχει ανά δύο ανεξάρτητες συναρτήσεις αυτού του είδους (δηλαδή με την ιδιότητα για κάθε είσοδο x και $i \neq j$ να ισχύει $\Pr[h_i(x) = h_j(x) = 0] = \frac{1}{4}$ (κι όμοια για ισότητα με 1)), τότε με πιθανότητα τουλάχιστον $(\frac{1}{2k} + \epsilon)$ ισχύει $\sum_{i:h(i)=1} x_i \ (mod\ 2) \neq 0$ για κάθε $x \neq 0$. Άρα αν διαλέξουμε τυχαία $r = cklogS$ τέτοιες συναρτήσεις το $OR(x_1, ..., x_{2^k})$ θα ισούται (για κατάλληλη επιλογή του c) με πιθανότητα τουλάχιστον $1 - \frac{1}{2S} + \epsilon'$ με το OR των r αυτών αθροισμάτων. Επομένως η πιθανότητα να ισούνται όλες οι πύλες OR του κυκλώματος με τις αντίστοιχες που προκύπτουν από το παραπάνω τυχαία επιλεγμένο σύνολο συναρτήσεων κατακερματισμού είναι γνήσια μεγαλύτερο από $1 - S * \frac{1}{2S}$ δηλαδή τουλάχιστον $\frac{1}{2} + \epsilon''$.

Επειδή επιπλέον, όπως είπαμε, $r = cklogS = O(log^2 S)$ και το βάθος κάθε OR αυξήθηκε το πολύ κατά 3 (από τον αθροιστή, τη mod_2 και την τελική

OR) το μόνο που μένει για να ολοκληρωθεί η απόδειξη είναι να δείξουμε ότι τα δυνατά r υποσύνολα του $\mathcal{H}$ (τα οποία έκαστο εφαρμοσμένο στο αρχικό κύκλωμα αντιστοιχούν σε ένα κύκλωμα του $\mathcal{D}$) είναι της τάξης του $O(S^{log^2 S})$. Για καλή μας τύχη, όμως, αποδεικνύεται [VV86] ότι υπάρχει ένα τέτοιο σύνολο ανά δύο ανεξάρτητων συναρτήσεων κατακερματισμού $\{0,1\}^{2^k} \to \{0,1\}$ μεγέθους μόλις $O(2^{2k}) = O(S^2)$. Επομένως είναι πράγματι $|\mathcal{D}| = O(|\mathcal{H}|^r) = O(S^{log^2 S})$.

■

Για να μετατρέψουμε το παραπάνω κύκλωμα σε ένα $SYM+$ κύκλωμα θα πρέπει αρχικά να το επιτρέψουμε να κάνει πράξεις μεταξύ ακεραίων κι άρα να μετατρέψουμε όλες τις πύλες του σε πύλες ακεραίων (που παίρνουν ως είσοδο έναν ακέραιο και βγάζουν ως έξοδο επίσης έναν ακέραιο). Αντικαθιστούμε λοιπόν τις πύλες $OR(x_1, ..., x_k)$ με πύλες $1 - x_1 ... x_k$ (δηλαδή έναν πολλαπλασιαστή κι έναν αθροιστή) όπου ο εισάριθμος του πολλαπλασιαστή θα είναι όπως είπαμε $O(log^2 S)$. Αντικαθιστούμε και τις πύλες $mod_p(x_1, ...x_l)$ με πύλες της μορφής $(x_1 + ... + x_l)^{p-1} modp$ (δηλαδή που χρησιμοποιούν έναν αθροιστή, έναν πολλαπλασιαστή εισάριθμου $p-2 = O(1)$ και μια πύλη $modp$ που κάνει το προφανές) και που από το μικρό Θεώρημα του Fermat , βγάζουν έξοδο 0 ή 1 ανάλογα με την mod_p (εδώ είναι ένα σημείο που χρειάζεται μια ελαφρώς διαφορετική κατασκευή μέσω του γενικότερου Θεωρήματος Euler για την γενική περίπτωση των p). Επίσης μπορούμε να πάμε όλες τις πύλες πολλαπλασιασμού στο πιο χαμηλό στρώμα, εφαρμόζοντας τους κανόνες $(x_1 + x_2)(x_3 + x_4) = x_1x_3 + x_2x_3 + x_1x_4 + x_2x_4$ και το γεγονός ότι $(x_1 \ mod p)(x_2 \ mod p) = (x_1x_2) \ mod p$ (αυτό απλά αυξάνει τον εισάριθμο των πυλών σε $O(log^{O(d)} S)$). Επίσης η πύλη MAJ της εξόδου αντικαθίσταται από μια πύλη πρόσθεσης όπου θέτουμε τη συνάρτηση Θ να αποδέχεται μόνο όσους ακέραιους είναι μεγαλύτεροι από $\frac{|\mathcal{D}|}{2}$.

Έχουμε σχεδόν τελειώσει με τη διαφορά ότι το βάθος του κυκλώματος μας δεν είναι 2 (ώστε να αντιστοιχεί σε ένα πολυώνυμο) και ότι έχουν μείνει οι ενδιάμεσες πύλες mod_p. Έστω λοιπόν ότι το στρώμα των πυλών κάτω από την κορυφαία πύλη πρόσθεσης είναι mod_p (αν είναι πύλη πρόσθεσης τότε απλά τις συμπτύσσουμε με την κορυφαία, ενώ αν είναι πύλη πολλαπλασιασμού, σημαίνει ότι (αφού αυτές τις διατηρούμε πάντα μόνο στο πιο κάτω στρώμα) έχουμε τελειώσει). Η έξοδος του κυκλώματος, λοιπόν, είναι της μορφής $\sum_{i=1}^{k} z_i \ (mod \ p)$ όπου z_i η αντίστοιχη είσοδος της i-οστής mod_p πύλης του προτελευταίου στρώματος. Προκύπτει, λοιπόν, ότι υπάρχει κατασκευάσιμο πολυώνυμο $P_r(z)$ βαθμού $poly(r)$ (και συντελεστών $2^{O(r)}$) τέτοιο ώστε για κάθε z να ισχύει $P_r(z) \ (mod \ p^r) = z \ (mod \ p)$ [BT94]. Για $r > logk/logp + 1$ έχουμε λοιπόν

$$\sum_{i=1}^{k} (z_i \ (mod \ r)) = \sum_{i=1}^{k} (P_r(z_i) \ (mod \ p^r)) = \left(\sum_{i=1}^{k} P_r(z_i) \right) (mod \ p^r)$$

εφόσον $\sum_{i=1}^{k} (z_i) \ (mod \ r) \leq (r-1) * k < p^r$.
Επομένως, κατά κάποιο τρόπο, συνενώσαμε όλες τις mod_p πύλες αυτού του στρώματος με μία mod_{p^r} την οποία μπορούμε να εισάγουμε σε μία νέα Θ' τέτοια

ώστε $\Theta'(x) = \Theta(x \ (mod \ p^r))$. Επίσης σπρώχνουμε όλους τους νέους πολλαπλασιαστές (πλήθους $O(S^{polylog(S)})$) προς τα κάτω (αυξάνοντας τον εισάριθμο τους, το πολύ κατά $O(log^{O(d)}S)$). Επομένως σε κάθε τέτοια επανάληψη παίρνουμε ένα νέο κύκλωμα με μεγέθη ίδιας τάξης με εξαίρεση το βάθος που μειώνεται κατά 1. Επειδή το βάθος είναι σταθερό, η τελική τάξη του κυκλώματος θα είναι πράγματι της μορφής $O(S^{polylog(S)})$ και ο εισάριθμος των πολλαπλασιαστών θα είναι $O(polylog(S))$, οπότε θα έχουμε πράγματι ένα $SYM+$ ισοδύναμο κύκλωμα με τα ζητούμενα μεγέθη, το οποίο κατασκευάσαμε σε χρόνο $O(S^{polylog(S)})$ (εφόσον όλα τα βήματα της απόδειξης ήταν ντετερμινιστικά) και η απόδειξη ολοκληρώθηκε.

Σημείωση. Οι φορές που μεγάλωναν οι εκθέτες στα λογαριθμικά που χρησιμοποιήθηκαν παραπάνω εξαρτιόνταν μόνο από το πλήθος των επαναλήψεων των εκάστοτε βημάτων, το οποίο με τη σειρά του εξαρτιόνταν μόνο από το βάθος του κυκλώματος, οπότε πράγματι ο εκθέτης είναι της μορφής $D(d)$ όπως θέλουμε.

Α΄.4 Απόδειξη του Λήμματος Εξαγωγής Ψευδοτυχαίων Γεννητριών από Ψευδοτυχαίες Λογικές Συναρτήσεις

Θεωρούμε δοθείσα μια σχέση της μορφής

$$|\Pr[C_n(\boldsymbol{f_n}) = 1] - \Pr[C_n(g(\boldsymbol{x})) = 1]| > 2^{-O(n)}$$

και θέλουμε να καταλήξουμε σε μια σχέση της μορφής

$$|\Pr[C'(\boldsymbol{x_{2k}}) = 1] - \Pr[C'(G(\boldsymbol{x_k})) = 1]| > 2^{-O(n)}$$

(όπου κάθε συμβολισμός και μεταβλητή συμβαδίζει με αυτούς που χρησιμοποιήθηκαν στην απόδειξη του Θεωρήματος 4.8). Η τεχνική που θα ακολουθήσουμε παρουσιάστηκε στο [RR97] και χρησιμοποιεί ιδέες που έχουν αναφερθεί και στο [GGM86].

Έστω ένα πλήρες δυαδικό δέντρο με 2^n φύλλα και $v_1, v_2, ..., v_{2^n-1}$ οι εσωτερικοί κόμβοι διατεταγμένοι με τρόπο που να σέβεται τη διάταξη προγόνου-απογόνου (δηλαδή αν ο v είναι απόγονος του u, τότε ο v εμφανίζεται πρώτος στην παραπάνω διάταξη). Έστω F_i το δάσος που προκύπτει αν πάρουμε τα υποδέντρα που περιλαμβάνουν τους πρώτους i κόμβους της παραπάνω διάταξης μαζί με όλα τα 2^n φύλλα y. Εναποθέτουμε μια τυχαία ομοιόμορφα κατανεμημένη μεταβλητή $\boldsymbol{x_r}$ k τυχαίων δυφίων στην κορυφή κάθε ρίζας r στο δάσος F_i. Έστω επίσης $r_i(y)$ η ρίζα του φύλλου y στο δάσος F_i. Ορίζουμε τότε $G_{i,y} \equiv G_{y_n} \circ G_{y_{n-1}} ... \circ G_{y_{n-d(y,i)+1}}$ όπου $d(y,i)$ η απόσταση του φύλλου y από τη ρίζα $r_i(y)$. Σημειωτέον ότι αν $r_i(y) = y$ (δηλαδή το φύλλο y είναι μόνος κόμβος στο F_i), τότε $G_{i,y} \equiv id$. Έστω λοιπόν $\boldsymbol{g_i}(y) = \#1.G_{i,y}(\boldsymbol{x_{r_i(y)}})$ η αντίστοιχη τυχαία λογική ψευδοτυχαία γεννήτρια. Από τους ορισμούς έχουμε ότι $\boldsymbol{g_0} \equiv \boldsymbol{f_n}$ και $\boldsymbol{g_{2^n-1}} \equiv g(\boldsymbol{x})$.

Προκύπτει, λοιπόν, από την ανωτέρω υπόθεση, ότι ισχύει ισοδύναμα η ανισότητα $|\Pr[C_n(\boldsymbol{g_0}) = 1] - \Pr[C_n(\boldsymbol{g_{2^n-1}}) = 1]| > 2^{-O(n)}$ κι άρα κάποια από τις διαφορές $|\Pr[C_n(\boldsymbol{g_i}) = 1] - \Pr[C_n(\boldsymbol{g_{i+1}}) = 1]|$ οφείλει να είναι μεγαλύτερη από $2^{-O(n)}$ (εφόσον αν s η μέγιστη τιμή που παίρνουν, ισχύει ότι $2^n s > 2^{-O(n)}$ από το οποίο προκύπτει άμεσα ότι $s > 2^{-O(n)}$). Έστω τα παιδιά v_L, v_R του v_{i+1}, για αυτό το i. Σταθεροποιώντας τώρα τα υπόλοιπα $\boldsymbol{x}$, πέραν του $\boldsymbol{x_{v_{i+1}}}$, έτσι ώστε να διατηρείται το κάτω φράγμα $2^{-O(n)}$, προκύπτει από τα παραπάνω ένα κύκλωμα C' μεγέθους προφανώς επίσης $2^{O(n)}$ το οποίο ξεχωρίζει τα $< \boldsymbol{x_{v_L}}, \boldsymbol{x_{v_R}} >$ από το $< G_0(\boldsymbol{x_{v_{i+1}}}), G_1(\boldsymbol{x_{v_{i+1}}}) > \equiv G(\boldsymbol{x_{v_{i+1}}})$ με επίσης τουλάχιστον $2^{-O(n)}$ διαφορά στο μέτρο, ολοκληρώνοντας έτσι την απόδειξη.

Παράρτημα Β΄

Λεξιλόγιο Όρων

Αναγωγή	Reduction
Αναδρομική	Recursive
Αποδείκτης	Prover
Αποτυχαιοποίηση	Derandomization
Αρτιότητα	Parity
Βάθος	Depth
Γραμματική χωρίς συμφραζόμενα	Context-free grammar
Γραμμολογαριθμικό	Linearithmic
Διαγωνιοποίηση	Diagonalization
Διαλογικές Αποδείξεις	Interactive Proofs
Διαμόρφωση	Configuration
Δυαδική αναπαράσταση	Binary represantation
Δυσκολία	Hardness
Δυφία	Bits
Εισάριθμος	Fan-in
Ελαχιστόρος	Minterm
Ελεγκτής	Verifier
Εξάριθμος	Fan-out
Ημιεκθετικό	Half-exponential
Ικανοποιήσιμο	Satisfiable
Καταβόθρα	Sink
Κατασκευάσιμη συνάρτηση	Time-constructible function
Κατασκευασιμότητα	Constructibility
Καταστάσεις	States
Κάτω Φράγμα	Lower Bound
Κλειδί	Key
Κύκλωμα	Circuit
Κυκλωματική πολυπλοκότητα	Circuit Complexity
Λογαροχώρο-ομοιόμορφα	Logspace-uniform
Λογική συνάρτηση	Boolean function

Μαντείο	Oracle
Μέγεθος	Size
Μεγιστόρος	Maxterm
Μη ντετερμινιστικός	Non-deterministic
Μηχανή Turing	Turing Machine
Ντετερμινιστικός	Deterministic
Ομοιόμορφα	Uniform
Παραγέμισμα	Padding
Πίνακας Αληθείας	Truth Table
Πιστοποιητικό	Certificate
Πληθωριστικότητα	Largeness
Πλήρη	Complete
Πολυλογαριθμικό	Polylogarithmic
Πρόβλημα Τερματισμού	Halting Problem
Πρωταρχικές αναδρομικές	Primitive recursive
Σημασιολογικό	Semantic
Σταθεροποίηση	Hardwiring
Στιγμιότυπο	Instance
Συμβουλή	Advice
Συμπυκνώσιμο πιστοποιητικό	Succinct witness
Συνάρτηση κατακερματισμού	Hash function
Συντακτικό	Syntactic
Σχετικιστικές Αποδείξεις	Relativizing Proofs
Σχετικοποιείται	Relativizes
Τυχαιοκρατικός	Probabilistic
Τυχαίος	Random
Υπερεκθετικό	Super-exponential
Υπερπολυωνυμικό	Super-polynomial
Υπογραμμικό	Sublinear
Υποεκθετικές	Subexponential
Υπολογισιμότητα	Computability
Φυσικές Αποδείξεις	Natural Proofs
Φυσικοποιείται	Naturalizes
Χρησιμότητα	Usefulness
Χρονική Ιεραρχία	Time Hierarchy
Χωρική Ιεραρχία	Space Hierarchy
Ψευδοτυχαία Γεννήτρια	Pseudorandom Generator
Ψευδοτυχαιότητα	Pseudorandomness