Υπολογιστική Πολυπλοκότητα

Πολλά προβλήματα, αν και επιλύσιμα, δεν φαίνεται να μπορούν να επιλυθούν υπό καμμία πρακτική έννοια από υπολογιστές εξαιτίας των υπερβολικών χρονικών απαιτήσεων τους.

Πρόβλημα του Περιοδεύοντος Πωλητή: Δίνεται ένα σύνολο $\{c_1, \ldots, c_n\}$ από n πόλεις καθώς και ένας $n \times n$ πίνακας d από μη αρνητικούς ακεραίους, όπου d_{ij} είναι η απόσταση ανάμεσα στην πόλη c_i και την πόλη c_j. Μια λύση στο πρόβλημα είναι μια διάταξη $\langle c_{\pi(1)}, \ldots, c_{\pi(n)} \rangle$ η οποία ελαχιστοποιεί την ποσότητα: $cost(\pi) = d_{\pi(1)\pi(2)} + d_{\pi(2)\pi(3)} + \cdots + d_{\pi(n-1)\pi(n)} + d_{\pi(n)\pi(1)}$.

Ένας αλγόριθμος που εξετάζει ένα προς ένα τα πιθανά δρομολόγια χρειάζεται χρόνο της τάξης του $(n-1)!$. Όταν το n είναι μεγάλο, είναι πρακτικά αδύνατο να λύσουμε το πρόβλημα αυτό.

Αποδοτικοί Αλγόριθμοι

Πως μπορεί να εκφραστεί η έννοια του «αποδοτικού (efficient) αλγορίθμου»;

Ένας αλγόριθμος που απαιτεί $n!$ ή 2^n βήματα, είναι προφανώς μη ρεαλιστικός.

Αντίθετα, ένας αλγόριθμος που απαιτεί πολυωνυμικό χρόνο, είναι πιο ελκυστικός.

(Ένας πολυωνυμικός αλγόριθμος με χρόνο εκτέλεσης n^{100} δεν ακούγεται ιδιαίτερα ελκυστικός. Όμως όταν ένα πρόβλημα μπορεί να λυθεί σε πολυωνυμικό χρόνο, συνήθως το αντίστοιχο πολυώνυμο είναι μικρού βαθμού.)

Η Κλάση P

Ορισμός: Μια μηχανή Turing ονομάζεται πολυωνυμικά φραγμένη αν υπάρχει πολυώνυμο $p(n)$ τέτοιο ώστε η μηχανή να τερματίζει πάντα μετά από $p(n)$ το πολύ βήματα, όπου n είναι το μήκος της εισόδου.

Ορισμός: Μια γλώσσα ονομάζεται πολυωνυμικά αποφασίσιμη αν υπάρχει μια πολυωνυμικά φραγμένη μηχανή Turing που την αποφασίζει.

Η κλάση των πολυωνυμικά αποφασίσιμων γλωσσών συμβολίζεται ως P.

Η Κλάση P, συνέχεια

Μπορούμε τώρα να εκφράσουμε μια εκλέπτυνση της θέσης των Church-Turing:

«Οι πολυωνυμικά φραγμένες μηχανές Turing και η κλάση P εκφράζουν ικανοποιητικά τις διαισθητικές έννοιες, αντίστοιχα, των αποδοτικών αλγορίθμων και των ρεαλιστικά επιλύσιμων προβλημάτων.»

Θεώρημα: Η κλάση P είναι κλειστή ως προς το συμπλήρωμα.

Απόδειξη: Εύκολη (γιατί;)

Παραδείγματα

Πολλά γνωστά προβλήματα αντιστοιχούν σε γλώσσες που ανήκουν στο P.

Πρόβλημα της Προσβασιμότητας (Reachability) : Δεδομένων ενός κατευθυνόμενου γραφήματος $G \subseteq V \times V$, όπου $V = \{v_1, \ldots, v_n\}$ είναι ένα πεπερασμένο σύνολο, και δύο κόμβων $v_i, v_j \in V$, υπάρχει μονοπάτι από τον v_i στον v_j;

Το πρόβλημα της προσβασιμότητας μπορεί να αναπαρασταθεί από τη γλώσσα:

$$R = \{k(G)b(i)b(j) : \text{υπάρχει μονοπάτι στο γράφημα } G \text{ από τον } v_i \text{ στον } v_j\}$$

όπου τα $b(i), b(j)$ είναι οι δυαδικές αναπαραστάσεις των i, j και $k(G)$ είναι κάποιος λογικός τρόπος κωδικοποίησης γραφημάτων ως συμβολοσειρές.

Είναι εύκολο να δει κανείς ότι το πρόβλημα της προσβασιμότητας είναι στο P.

Παραδείγματα

Κύκλος Euler: Δεδομένου ενός κατευθυνόμενου γραφήματος G, υπάρχει κλειστό μονοπάτι στο G το οποίο χρησιμοποιεί κάθε ακμή ακριβώς μία φορά; Ένα γράφημα που περιέχει ένα τέτοιο μονοπάτι ονομάζεται γράφημα Euler.

Η γλώσσα $L = \{k(G) : \text{το } G \text{ είναι γράφημα Euler}\}$ ανήκει στο P γιατί τα γραφήματα Euler χαρακτηρίζονται από το εξής θεώρημα:

Θεώρημα [Euler] Ένα γράφημα G είναι γράφημα Euler αν και μόνο αν έχει τις παρακάτω δύο ιδιότητες:

1. Για κάθε ζεύγος κόμβων u, v του G κανένας από τους οποίους δεν είναι απομονωμένος, υπάρχει μονοπάτι από τον u στον v.
2. Όλοι οι κόμβοι έχουν ίσο αριθμό εισερχόμενων και εξερχόμενων ακμών.

Παραδείγματα

Κύκλος Hamilton: Δεδομένου ενός μη κατευθυνόμενου γραφήματος G, υπάρχει κύκλος που περνάει από κάθε κόμβο του G ακριβώς μία φορά; Ένας τέτοιος κύκλος ονομάζεται κύκλος γράφημα Hamilton.

Κανείς μέχρι σήμερα δεν έχει καταφέρει να ανακαλύψει πολυωνυμικό αλγόριθμο για το πρόβλημα του κύκλου Hamilton.

Το πρόβλημα βέβαια είναι προφανώς επιλύσιμο (πώς;)

Παραδείγματα

Το πρόβλημα του Περιοδεύοντος Πωλητή που ήδη συζητήσαμε, είναι ένα πρόβλημα βελτιστοποίησης: αναζητούμε την καλύτερη από όλες τις λύσεις με βάση κάποια συνάρτηση κόστους.

Πώς μπορούμε να μετατρέψουμε τα προβλήματα βελτιστοποίησης σε γλώσσες, ώστε να μελετήσουμε την πολυπλοκότητα τους με κομψό τρόπο;

Βασική Ιδέα: Εφοδιάζουμε κάθε είσοδο με έναν περιορισμό στη συνάρτηση κόστους.

Προβλήματα

Πρόβλημα του Περιοδεύοντος Πωλητή με Προϋπολογισμό: Δίνεται ένα σύνολο $\{c_1, \ldots, c_n\}$ από n πόλεις καθώς και ένας $n \times n$ πίνακας d από μη αρνητικούς ακεραίους, όπου d_{ij} είναι η απόσταση ανάμεσα στην πόλη c_i και την πόλη c_j. Δίνεται επίσης ένας ακέραιος $B \geq 0$. Υπάρχει διάταξη π των n πόλεων τέτοια ώστε $cost(\pi) \leq B$;

Αν μπορούσαμε να λύσουμε το αρχικό πρόβλημα σε πολυωνυμικό χρόνο, τότε θα μπορούσαμε να λύσουμε και το παραπάνω πρόβλημα σε πολυωνυμικό χρόνο: υπολογίζουμε το κόστος της φθηνότερης διαδρομής και το συγκρίνουμε με το B.

Επομένως, κάθε αρνητικό αποτέλεσμα σχετικό με την πολυπλοκότητα του παραπάνω προβλήματος, θα έχει άμεσες αρνητικές συνέπειες και για το αρχικό πρόβλημα.

Προβλήματα από τη Λογική Boole

Πολλά σημαντικά αλγοριθμικά προβλήματα προκύπτουν από τη μελέτη της λογικής Boole.

Έστω $X = \{x_1, x_2, \ldots, x_n\}$ ένα σύνολο από μεταβλητές Boole, και έστω $\overline{X} = \{\overline{x_1}, \overline{x_2}, \ldots, \overline{x_n}\}$ το σύνολο των αρνήσεων των μεταβλητών του X.

Τα μέλη του $X \cup \overline{X}$ θα τα ονομάζουμε στοιχεία (literals). Τα μέλη του X θα λέγονται θετικά στοιχεία και αυτά του $\overline{X}$ αρνητικά στοιχεία.

Μια φράση (clause) C είναι ένα μια διάζευξη $k \geq 1$ στοιχείων.

Ένας τύπος Boole σε ΣΥΖΕΥΚΤΙΚΗ κανονική μορφή (Conjunctive Normal Form) είναι μία σύζευξη φράσεων.

Προβλήματα από τη Λογική Boole, συνέχεια

Παράδειγμα: Έστω $X = \{x_1, x_2, x_3\}$ και $\overline{X} = \{\overline{x_1}, \overline{x_2}, \overline{x_3}\}$.
Η ακόλουθη είναι μία φράση:

$$(\overline{x_1} \vee x_2 \vee \overline{x_3})$$

Ένας τύπος Boole σε συζευκτική κανονική μορφή που αποτελείται απο τρεις φράσεις:

$$(x_1 \vee \overline{x_2}) \wedge (\overline{x_3}) \wedge (x_1 \vee x_3)$$

Πώς μπορούμε να δώσουμε νόημα στους τύπους Boole;

Προβλήματα από τη Λογική Boole, συνέχεια

Ορισμός: Έστω F ένας τύπος Boole σε συζευκτική κανονική μορφή. Μια ανάθεση αλήθειας για τον F είναι μια απεικόνιση από το σύνολο X των μεταβλητών του F στο σύνολο $\{\bot, \top\}$ (διαισθητικά, στο $\{\mathsf{false}, \mathsf{true}\}$).

Ορισμός: Έστω F ένας τύπος Boole σε συζευκτική κανονική μορφή και T μια ανάθεση αλήθειας για τον F. Θα λέμε ότι η T ικανοποιεί τον F αν κάθε φράση C του F αληθεύει, δηλ. αν κάθε φράση C του F περιέχει τουλάχιστον ένα στοιχείο που παίρνει την τιμή $\top$ στην T.

Ένας τύπος F ονομάζεται ικανοποιήσιμος αν υπάρχει ανάθεση αλήθειας που να τον ικανοποιεί.

Παράδειγμα: Έστω ο τύπος Boole

$$F = (x_1 \vee \overline{x_2} \vee x_3) \wedge (\overline{x_1}) \wedge (x_2 \vee \overline{x_2})$$

Μια απόδοση τιμών που ικανοποιεί τον F είναι η $T(x_1) = \bot$, $T(x_2) = \top$, και $T(x_3) = \top$.

Παράδειγμα: Έστω ο τύπος Boole

$$F = (x_1 \vee x_2 \vee x_3) \wedge (\overline{x_1} \vee x_2) \wedge (\overline{x_2} \vee x_3) \wedge (\overline{x_3} \vee x_1) \wedge (\overline{x_1} \vee \overline{x_2} \vee \overline{x_3})$$

Μπορεί να δειχτεί ότι ο παραπάνω τύπος δεν είναι ικανοποιήσιμος (πώς;).

Προβλήματα από τη Λογική Boole, συνέχεια

Πρόβλημα της Ικανοποιησιμότητας (SATISFIABILITY): Δεδομένου ενός τύπου F σε συζευκτική κανονική μορφή, είναι ο F ικανοποιήσιμος;

Για το πρόβλημα αυτό δεν έχει βρεθεί μέχρι σήμερα πολυωνυμικός αλγόριθμος (και η πλέον αποδεκτή άποψη είναι ότι τέτοιος αλγόριθμος δεν υπάρχει).

Υπάρχουν ειδικές περιπτώσεις του προβλήματος για τις οποίες υπάρχει πολυωνυμικός αλγόριθμος (πχ. το $2SAT$ στο οποίο κάθε φράση έχει το πολύ δύο στοιχεία).

Μια καινούργια κλάση προβλημάτων

Τί κοινό έχουν τα τρία προβλήματα απόφασης που παρουσιάσαμε; (Κύκλος **Hamilton**, Πρόβλημα Περιοδεύοντος Πωλητή με Προϋπολογισμό, Ικανοποιησιμότητα).

Για ποιο λόγο τα θεωρούμε υπολογιστικά δύσβατα (**intractable**);

Θα δείξουμε πως και τα τρία, μαζί με πολλά άλλα «ομοειδή» προβλήματα, είναι εξίσου δύσκολα. Αν μπορούσαμε να λύσουμε ένα από αυτά σε πολυωνυμικό χρόνο τότε υπάρχει πολυωνυμικός αλγόριθμος για όλα τα άλλα.

Μια καινούργια κλάση προβλημάτων, συνέχεια

Θέλουμε να χαρακτηρίσουμε με ακρίβεια μια κλάση προβλημάτων για τα οποία πιστεύουμε πως δεν υπάρχει πολυωνυμικός αλγόριθμος.

Τί γνωρίζουμε για αυτά τα προβλήματα, ώστε να ορίσουμε τελικά την κλάση που θέλουμε;

Γνωρίζουμε πως λύνονται από πολυωνυμικούς, μη ντετερμινιστικούς αλγορίθμους.

Η κλάση NP

Ορισμός: Μία μη ντετερμινιστική μηχανή Turing ονομάζεται πολυωνυμικά φραγμένη αν υπάρχει πολυώνυμο $p(n)$ τέτοιο ώστε για κάθε είσοδο x, όλοι οι υπολογισμοί (ενσαρκώσεις) της μηχανής τερματίζουν το πολύ σε $p(|x|)$ βήματα.

Ορίζουμε ως NP (Nondeterministic Polynomial) την κλάση όλων των γλωσσών που αποφασίζονται από πολυωνυμικά φραγμένες μη ντετερμινιστικές μηχανές Turing.

[Υπενθυμίζουμε ότι μια μη ντετερμινιστική μηχανή αποφασίζει μια γλώσσα L αν: για κάθε είσοδο που δεν ανήκει στην L, όλοι οι υπολογισμοί απορρίπτουν την είσοδο, και για κάθε είσοδο που ανήκει στην L, υπάρχει *ένας τουλάχιστον* υπολογισμός που δέχεται την είσοδο.]

Προβλήματα στην κλάση NP

Είδαμε τρία παραδείγματα (Περιοδεύων Πωλητής, Κύκλος Hamilton, Ικανοποιησιμότητα), για τα οποία υπάρχει ευρέως η πεποίθηση ότι δεν ανήκουν στο P. Είναι εύκολο να δει κανείς ότι τα προβλήματα αυτά ανήκουν στο NP.

Παράδειγμα: Για το πρόβλημα της Ικανοποιησιμότητας, είναι εύκολο να κατασκευάσει κανείς μια μη ντετερμινιστική πολυωνυμικά φραγμένη μηχανή Turing M που αποφασίζει τη γλώσσα που περιλαμβάνει όλες τις κωδικοποιήσεις των ικανοποιήσιμων τύπων Boole.

Δεδομένου ενός τύπου F, η M μαντεύει μια απόδοση τιμών αληθείας για το δεδομένο τύπο. Κατόπιν ελέγχει ντετερμινιστικά αν η ανάθεση αυτή ικανοποιεί τον τύπο.

Προβλήματα στην κλάση NP, συνέχεια

Για το πρόβλημα του περιοδεύοντος πωλητή με προϋπολογισμό, μπορούμε επίσης να κατασκευάσουμε μια μη ντετερμινιστική πολυωνυμικά φραγμένη μηχανή Turing M που αποφασίζει την αντίστοιχη γλώσσα.

Η M επιλέγει μη ντετερμινιστικά μια διαδρομή από πόλεις. Κατόπιν υπολογίζει το κόστος της διαδρομής αυτής και το συγκρίνει με τον προϋπολογισμό B.

Άρα και το πρόβλημα αυτό είναι στο NP.

Σχέση του P με το NP

Προφανώς $P \subseteq NP$.

Το πιο σημαντικό ερώτημα της θεωρίας πολυπλοκότητας:

Ισούται το P με το NP;

Υπάρχει ευρέως η πεποίθηση ότι $P \neq NP$, αλλά όλες οι προσπάθειες για απόδειξη έχουν πέσει στο κενό.

Η κλάση EXP

Ορισμός: Μία (ντετερμινιστική) μηχανή Turing ονομάζεται εκθετικά φραγμένη αν υπάρχει πολυωνυμο $p(n)$ τέτοιο ώστε για κάθε είσοδο x η μηχανή τερματίζει πάντα μετά από $2^{p(|x|)}$ το πολύ βήματα.

Ορίζουμε ως EXP την κλάση όλων των γλωσσών που αποφασίζονται από εκθετικά φραγμένες μηχανές Turing.

Ποιά είναι η σχέση του NP με το EXP;

Η κλάση EXP, συνέχεια

Θεώρημα: NP⊆EXP

Απόδειξη (Σκιαγράφηση): Έχουμε δει ότι κάθε μη ντετερμινιστική μηχανή Turing N μπορεί να προσομοιωθεί από μία ντετερμινιστική μηχανή Turing D.

Η βασική ιδέα στην προσομοίωση αυτή ήταν ότι η D προχωράει με breadth first search: επισκέπτεται κάθε φορά τους κόμβους (συνολικές καταστάσεις της N) που βρίσκονται στο ίδιο βάθος.

Επομένως, αν η N αποφασίζει μία γλώσσα L με χρονικό όριο $p(n)$, τότε και η D θα αποφασίζει την L σε χρόνο το πολύ $p(n)c^{p(n)}$, όπου c είναι μία σταθερά που εξαρτάται από το πλάτος του δέντρου της D.

Όμως, $p(n)c^{p(n)} = 2^{O(p(n))}$. Κατά συνέπεια, $L \in$ EXP.

Σχέση μεταξύ μη ντετερμινισμού και ντετερμινισμού

Αποδείξαμε και το ακόλουθο

Θεώρημα: Για κάθε μη ντετερμινιστική Μ. Τ. που τρέχει σε χρόνο

$$t(n)$$

υπάρχει ισοδύναμη ντετερμινιστική που τρέχει σε χρόνο

$$2^{O(t(n))}.$$

Αυτή η απλή προσομοίωση που δείξαμε (με ψάξιμο κατά πλάτος) είναι η καλύτερη, από άποψη πολυπλοκότητας χρόνου, που γνωρίζουμε.

Από μία άποψη, η ερώτηση $P \stackrel{?}{=} NP$ ρωτάει αν μπορεί να γίνει προσομοίωση από ντετερμινιστική μηχανή σε χρόνο $O\left([t(n)]^c\right)$, για κάποια σταθερά $c > 0$.

Η κλάση EXP, συνέχεια

Το προηγούμενο θεώρημα μας οδηγεί σε ένα ακόμη σημαντικό ανοιχτό ερώτημα της Θεωρίας Πολυπλοκότητας:

Ισούται το NP με το EXP;

Με βάση τα όσα έχουμε πει μέχρι τώρα:

$$\mathsf{P} \subseteq \mathsf{NP} \subseteq \mathsf{EXP}$$

Μπορεί επιπλέον να αποδειχτεί ότι $\mathsf{P} \subset \mathsf{EXP}$. Κατά συνέπεια, τουλάχιστον μία από τις σχέσεις υποσυνόλου στην παραπάνω σχέση, είναι γνήσια (αλλά δεν ξέρουμε ποιά ή ποιές!).