

Υπολογιστική Πολυπλοκότητα

Δημήτρης Φωτάκης

Σχολή Ηλεκτρολόγων Μηχανικών
και Μηχανικών Υπολογιστών

Εθνικό Μετσόβιο Πολυτεχνείο



Υπολογιστική Πολυπλοκότητα

- Γιατί κάποια (επιλύσιμα) **προβλήματα** είναι δύσκολο να λυθούν από **υπολογιστικές μηχανές**.
 - Ποια επιλύσιμα προβλήματα είναι **εύκολα** και ποια **δύσκολα**;
- Αντικείμενο: ελάχιστοι **υπολογιστικοί πόροι** για επιλύσιμα προβλήματα.
 - Εύλογοι υπολογιστικοί πόροι: **ευεπίλυτα προβλήματα**.
 - Fractional knapsack, minimum spanning tree, shortest paths, max-flow, min-cut, linear programming, ...
 - Διαφορετικά, **δισεπίλυτα προβλήματα**.
 - TSP, discrete knapsack, vertex cover, independent set, set cover, scheduling, ...
 - Επίδραση **υπολογιστικού μοντέλου**.

Προσέγγιση

- Κλάσεις προβλημάτων (**complexity classes**) με παρόμοια υπολογιστική «δυσκολία» (**computational complexity**).
- Με (κατάλληλη) **αναγωγή** ορίζουμε «διάταξη» προβλημάτων σε κάθε κλάση (με βάση δυσκολία).
 - Δυσκολότερα προβλήματα: **πλήρη** για την κλάση, συνοψίζουν **δυσκολία κλάσης**.
 - Πλήρες πρόβλημα «εύκολο»: όλη η κλάση «εύκολη».
 - **Αρνητικό** αποτέλεσμα: όλα τα πλήρη προβλήματα «δύσκολα».
 - Έτσι (προσπαθούμε να) καθορίσουμε **επαρκείς υπολογιστικούς πόρους** για επίλυση **δύσκολων προβλημάτων**.
- Διαλεκτική σχέση **αλγόριθμων** και **πολυπλοκότητας**.

Χρονική Πολυπλοκότητα

- Χρονική πολυπλοκότητα DTM M :
 - Αύξουσα συνάρτηση $t : \mathbb{N} \rightarrow \mathbb{N}$ ώστε για κάθε x , $|x| = n$, $M(x)$ τερματίζει σε $\leq t(n)$ βήματα.
 - Χρονική πολυπλοκότητα προβλήματος Π :
 - Χρονική πολυπλοκότητα «ταχύτερης» DTM που λύνει Π .
 - Κλάση $\mathbf{DTIME}[t(n)] \equiv \{\Pi : \Pi \text{ λύνεται σε χρόνο } O(t(n))\}$
 - Ιεραρχία κλάσεων χρονικής πολυπλοκότητας:
 $\mathbf{DTIME}[t(n)] \subset \mathbf{DTIME}[\omega(t(n) \log t(n))]$
 $\mathbf{DTIME}[n] \subset \mathbf{DTIME}[n^2] \subset \mathbf{DTIME}[n^3] \subset \dots$
 - Πολυωνυμικός χρόνος: $\mathbf{P} \equiv \bigcup_{k \geq 0} \mathbf{DTIME}[n^k]$
 - Εκθετικός χρόνος: $\mathbf{EXP} \equiv \bigcup_{k \geq 0} \mathbf{DTIME}[2^{n^k}]$
- $\mathbf{P} \subset \mathbf{EXP}$

Ευεπίλυτα Προβλήματα

- **Κλάση P** : προβλήματα απόφασης που λύνονται σε πολυωνυμικό χρόνο.
- **Θέση Cook – Karp** : κλάση ευεπίλυτων προβλημάτων ταυτίζεται με κλάση P.

Υπέρ θέσης Cook – Karp:

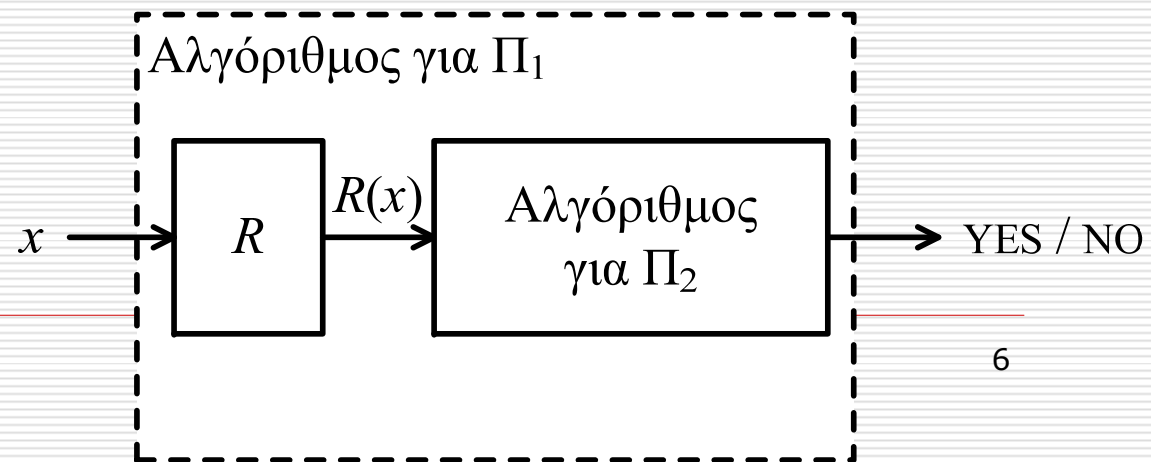
- Συνήθως πολυώνυμα μικρού βαθμού (π.χ. n , n^2 , n^3).
- Κλειστότητα κλάσης.
- Διπλασιασμός υπολογιστικής ισχύος: σημαντική αύξηση στο μέγεθος στιγμιότυπων που λύνουμε.

Εναντίον θέσης Cook – Karp:

- Ακραίες περιπτώσεις: πρακτικό το n^{100} αλλά όχι το $(1.001)^n$!
- Γραμμικός Προγραμματισμός: **Simplex** εκθετικού χρόνου αλλά πολύ γρήγορος στην πράξη. **Ελλειψοειδές** πολυωνυμικού χρόνου αλλά καθόλου πρακτικός!

(Πολυωνυμική) Αναγωγή

- Π_1 **ανάγεται** πολυωνυμικά σε Π_2 ($\Pi_1 \leq_p \Pi_2$):
 - Υπάρχει πολυωνυμικά υπολογίσιμη συνάρτηση $R: \Sigma^* \rightarrow \Sigma^*$ ώστε $\forall x \in \Sigma^*, x \in \Pi_1 \Leftrightarrow R(x) \in \Pi_2$.
 - R καλείται **πολυωνυμική αναγωγή**.
 - $\Pi_1 \leq_p \Pi_2$: Π_2 είναι τουλ. τόσο δύσκολο όσο το Π_1 (για τον υπολογισμό σε πολυωνυμικό χρόνο).
 - Αν $\Pi_2 \in \mathbf{P}$, τότε και $\Pi_1 \in \mathbf{P}$.
 - Αν $\Pi_1 \notin \mathbf{P}$, τότε και $\Pi_2 \notin \mathbf{P}$.



Πληρότητα

- Έστω \mathbf{C} μια κλάση προβλημάτων.
 - Π είναι **C-δύσκολο** (**C-hard**) ως προς αναγωγή R αν κάθε πρόβλημα Π' στην \mathbf{C} ανάγεται κατά R στο Π .
 $\forall \Pi' \in \mathbf{C}, \Pi' \leq_R \Pi$
 - Αν $\Pi \in \mathbf{C}$ και Π είναι **C-δύσκολο** ως προς αναγωγή R , τότε Π είναι **C-πλήρες** (**C-complete**) ως προς R .
 $\forall \Pi' \in \mathbf{C}, \Pi' \leq_R \Pi$
και $\Pi \in \mathbf{C}$
- **Πλήρη** προβλήματα (ως προς κατάλληλη αναγωγή) συνοψίζουν υπολογιστική δυσκολία κλάσης \mathbf{C} .
 - Αναγωγή πρέπει να είναι «λίγο ευκολότερη» από «δυσκολότερα» προβλήματα στην κλάση \mathbf{C} .
- Κλάση \mathbf{C} **κλειστή** ως προς αναγωγή R αν
 $\forall \Pi_1, \Pi_2, \Pi_1 \leq_R \Pi_2$ και $\Pi_2 \in \mathbf{C} \Rightarrow \Pi_1 \in \mathbf{C}$

Ιδιότητες Αναγωγής

- Κλάση \mathbf{P} είναι κλειστή ως προς πολυωνυμική αναγωγή.
 - Αν $\Pi_2 \in \mathbf{P}$, τότε και $\Pi_1 \in \mathbf{P}$.
- Πολυωνυμική αναγωγή είναι μεταβατική.
 - Σύνθεση πολυωνυμικών αναγωγών αποτελεί πολυωνυμική αναγωγή.
- Αν $\Pi_1 \leq_P \Pi_2$ και $\Pi_2 \leq_P \Pi_1$, τότε Π_1 και Π_2 πολυωνυμικά ισοδύναμα, $\Pi_1 \equiv_P \Pi_2$.
- Κλάσεις κλειστές ως προς αναγωγή R με κοινό πλήρες πρόβλημα ως προς αναγωγή R **ταυτίζονται**.
 - Έστω κλάσεις $\mathbf{C}_1, \mathbf{C}_2$ κλειστές ως προς αναγωγή R .
 - Αν $\mathbf{C}_1, \mathbf{C}_2$ έχουν κοινό πλήρες πρόβλημα Π ως προς αναγωγή R , τότε $\mathbf{C}_1 = \mathbf{C}_2$.

(Απλά) Παραδείγματα Αναγωγών

- Κύκλος Hamilton \leq_p TSP με αποστάσεις 1 και 2 – TSP(1, 2).
 - Δίνεται γράφημα $G(V, E)$. Έχει G κύκλο Hamilton;
 - Από G , κατασκευάζουμε στιγμιότυπο I_G του TSP(1, 2):
 - Μία «πόλη» u για κάθε κορυφή $u \in V$.
 - Συμμετρικές αποστάσεις: $d(u, v) = \begin{cases} 1 & \text{αν } \{u, v\} \in E \\ 2 & \text{αν } \{u, v\} \notin E \end{cases}$
 - G έχει κύκλο Hamilton αν I_G έχει περιοδεία μήκους $\leq |V|$.
- TSP(1, 2) \leq_p Metric TSP.
 - 1^ο ειδική περίπτωση 2^{ου} : αποστάσεις 1 και 2 ικανοποιούν τριγωνική ανισότητα.

(Απλά) Παραδείγματα Αναγωγών

- Min Vertex Cover \equiv_p Max Independent Set \equiv_p Max Clique.
 - Vertex cover C σε γράφημα $G(V, E)$ ανν
independent set $V \setminus C$ σε γράφημα G ανν
clique $V \setminus C$ σε συμπληρωματικό γράφημα \bar{G} .
- Έστω μη κατευθυνόμενο γράφημα $G(V, E)$, $|V| = n$.
Τα παρακάτω είναι ισοδύναμα:
 - Το G έχει vertex cover $\leq k$.
 - Το G έχει independent set $\geq n - k$.
 - Το συμπληρωματικό \bar{G} έχει clique $\geq n - k$.

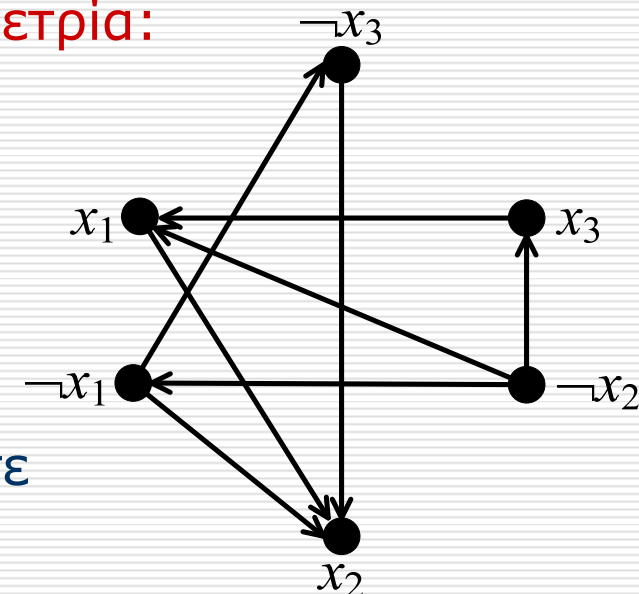
k -Ικανοποιησιμότητα

- Λογική πρόταση φ σε k -Συζευκτική Κανονική Μορφή, k -CNF:
 $\varphi \equiv c_1 \wedge \dots \wedge c_m$, όπου $c_i = l_{i_1} \vee \dots \vee l_{i_k}$, με $l_{i_j} \in \{x_1, \neg x_1, \dots, x_n, \neg x_n\}$
 - c_j : όροι. l_{i_j} : literals. #literals σε κάθε όρο $\leq k$.
Π.χ. για $k = 2$: $(x_1 \vee x_2) \wedge (x_1 \vee \neg x_3) \wedge (\neg x_1 \vee x_2) \wedge (x_2 \vee x_3)$
- k -Ικανοποιησιμότητα:
 - Δίνεται φ σε k -CNF. Είναι φ ικανοποιήσιμη;

2-Ικανοποιησιμότητα

□ 2-Ικανοποιησιμότητα $\in \mathbf{P}$.

- Παρατηρούμε ότι $l_i \vee l_j \equiv (\neg l_i \rightarrow l_j) \wedge (\neg l_j \rightarrow l_i)$
- Κατασκευάζουμε κατευθυν. γράφημα G_φ με «συνεπαγωγές» φ .
 G_φ έχει κορυφές $\{x_1, \dots, x_n\} \cup \{\neg x_1, \dots, \neg x_n\}$
- Για κάθε όρο $l_i \vee l_j$, ακμές G_φ $(\neg l_i, l_j)$ και $(\neg l_j, l_i)$
- Ακμές και μονοπάτια G_φ εμφανίζουν συμμετρία:
ακμή $(l_i, l_j) \Leftrightarrow$ ακμή $(\neg l_j, \neg l_i)$
 $l_i - l_j$ μονοπάτι $\Leftrightarrow \neg l_j - \neg l_i$ μονοπάτι
- Όμως $l_i \rightarrow l_j$ ψευδής $\Leftrightarrow l_i = 1$ και $l_j = 0$
- φ μη ικανοποιήσιμη αν υπάρχουν $x_i - \neg x_i$ και $\neg x_i - x_i$ μονοπάτια.
- Λόγω αυτών, καμία αποτίμηση x_i και $\neg x_i$ σε συμπληρωματικές τιμές δεν ικανοποιεί φ .



«Δύσκολα» Προβλήματα

- Τι κάνουμε όταν ένα πρόβλημα φαίνεται «δύσκολο»;
 - «Δύσκολο»: μετά από μεγάλη προσπάθεια, δεν βρίσκουμε αποδοτικό αλγόριθμο (πολυωνυμικού χρόνου).
- Πάμε στο αφεντικό και λέμε:
 - Δεν **μπορώ** να βρω αποδοτικό αλγόριθμο. Απόλυση!
 - Δεν **υπάρχει** αποδοτικός αλγόριθμος. Too good to be true!
 - **Κανένας** δεν μπορεί να βρει αποδοτικό αλγόριθμο:
 - **Ανάγουμε** πολυωνυμικά κάποιο γνωστό **NP-πλήρες** πρόβλημα στο «δικό μας».
- Θεωρία **NP-πληρότητας**.
 - **NP-πλήρη**: κλάση εξαιρετικά **σημαντικών προβλημάτων** που ανάγονται πολυωνυμικά το ένα στο άλλο.
 - **Είτε όλα** λύνονται σε πολυωνυμικό χρόνο **είτε κανένα**.
 - Έχουν **μελετηθεί τόσο πολύ**, ώστε όλοι πιστεύουν ότι **κανένα!**