



ΗΥ - 280 Θεωρία Υπολογισμού

Διάλεξη 19

Κωνσταντίνος Βάρσος

Τμήμα Επιστήμης Υπολογιστών – Πανεπιστήμιο Κρήτης

Χειμερινό εξάμηνο 2025 - 26

Πολυπλοκότητα

Τι είδαμε στην Διάλεξη 18

- Κλάση P .
- Ιδιότητες κλάσης P .

Κλάση P

■ Ορίζουμε την κλάση προβλημάτων $\Sigma\Gamma N\Theta E\Gamma O\Sigma$, ως εξής

$$\Sigma\Gamma N\Theta E\Gamma O\Sigma := \{p \mid p \text{ σύνθετος ακέραιος αριθμός}\}$$

■ Είναι η $\Sigma\Gamma N\Theta E\Gamma O\Sigma \in \mathbf{P}$ αν χρησιμοποιήσουμε δυαδική αναπαράσταση;

Κατασκευάζουμε την εξής TM:

$M =$ “Για είσοδο $\langle p, q \rangle$, όπου p, q ακέραιοι αριθμοί:

1. Ελέγχει αν $1 < q < p$. Αν όχι, **απορρίπτει**.
2. Ελέγχει αν q διαιρεί το p . Αν ναι, **αποδέχεται**, αλλιώς **απορρίπτει**.”

(Ορθότητα) Ο p είναι σύνθετος ακέραιος αριθμός $\iff$ υπάρχει αριθμός $q \in (1, p)$ τ.ω. να διαιρεί το p . □

Επαλήθευση

■ Είδαμε ότι σε πολλά προβλήματα μπορούμε να αποφύγουμε την εξαντλητική αναζήτηση, χρησιμοποιώντας πολυωνυμικού χρόνου αλγορίθμους.

■ Όχι όμως όλα! Που οφείλεται αυτή η αποτυχία;

■ Ορισμένα προβλήματα έχουν το χαρακτηριστικό της **πολυωνυμικής επαληθευσιμότητας** (polynomial verification). Παρότι εξακολουθούμε να μην έχουμε κάποιον πολυωνυμικό αλγόριθμο για την επίλυση τους, μπορούμε να βεβαιωθούμε γι'αυτήν απλώς παραθέτοντας την.

■ Δηλαδή η επαλήθευση ίσως να είναι ευκολότερη από την εξακρίβωση.

π.χ., στο πρόβλημα $\Sigma\Gamma\text{ΝΘΕΤΟΣ} := \{p \mid p \text{ σύνθετος ακέραιος αριθμός}\}$ μπορούμε να επαληθεύσουμε πολύ εύκολα αν ένας αριθμός είναι σύνθετος.

Επαλήθευση

■ Και πάλι όμως, ορισμένα προβλήματα ενδεχομένως να μην είναι πολυωνυμικώς επαληθεύσιμα.

π.χ., στο πρόβλημα

$$HAMPATH := \left\{ \langle G, s, t \rangle \mid \begin{array}{l} \langle G, s, t \rangle \in PATH \text{ και το μονοπάτι} \\ \text{περνάει από κάθε κορυφή του } G \\ \text{μια φορά.} \end{array} \right\}.$$

ακόμα και να μπορούμε να εξακριβώσουμε το $\overline{HAMPATH}$, δεν μπορούμε να το επαληθεύσουμε σε πολυωνυμικό χρόνο.

Επαλήθευση

Ορισμός. Ονομάζουμε **επαληθευτή** μιας γλώσσας A οποιονδήποτε αλγόριθμος V τ.ω.

$$A := \{w \mid \text{ο } V \text{ αποδέχεται την λέξη } \langle w, c \rangle, \text{ όπου } c \text{ κάποια συμβολειρά}\}.$$

■ Ο χρόνος εκτέλεσης του επαληθευτή υπολογίζεται μόνο συναρτήσει του μήκους της συμβολοσειράς w . Άρα, ένας επαληθευτής πολυωνυμικού χρόνου έχει πολυωνυμικό χρόνο εκτέλεσης ως προς το μήκος της w .

■ Μια γλώσσα A ονομάζεται πολυωνυμικά επαληθεύσιμη εάν υπάρχει για αυτήν επαληθευτής πολυωνυμικού χρόνου.

■ Η c ονομάζεται πιστοποιητικό (certificate) ή απόδειξη της συμμετοχής της w στην A .

■ Στους πολυωνυμικούς επαληθευτές το πιστοποιητικό πρέπει να έχει πολυωνυμικό μήκος.

Επαλήθευση

Μερικά παραδείγματα πιστοποιητικών.

■ Για το πρόβλημα $\langle G, s, t \rangle \in \text{HAMPATH}$, ένα πιστοποιητικό είναι μια χαμιλτονιανή διαδρομή από τον s στον t .

■ Για το πρόβλημα $\langle x \rangle \in \Sigma\Gamma\text{ΝΘΕΤΟΣ}$, ένα πιστοποιητικό είναι ένας οποιοσδήποτε διαιρέτης του.

■ Για το πρόβλημα $\langle G, k \rangle \in \text{Περιοδεύον Πωλητής}$, ένα πιστοποιητικό είναι ένα δοσμένο μονοπάτι όπου πληρούνται οι προϋποθέσεις του k .

■ Για το πρόβλημα $\langle A, b, c \rangle \in \text{Γραμμικός Προγραμματισμός}$, ένα πιστοποιητικό είναι η απόδειξη της λύσης.

■ Για το πρόβλημα $\langle A, b, c \rangle \in \text{0-1 Γραμμικός Προγραμματισμός}$, ένα πιστοποιητικό είναι η απόδειξη της λύσης $x \in \{0, 1\}^d$.

Κλάση NP

■ Ορίζουμε ως **NP** την κλάση όλων των γλωσσών που επιδέχονται επαληθευτή πολυωνυμικού χρόνου.

Ορισμός. Έστω $t : \mathbb{N} \rightarrow \mathbb{R}_{\geq 0}$ οποιαδήποτε συνάρτηση. Ορίζουμε την **κλάση χρονικής πολυπλοκότητας** $\text{NTIME}(t(n))$ τη συλλογή όλων των γλωσσών που μπορούν να διαγνωστούν από μια μη-ντετερμινιστική TM χρόνου $O(t(n))$.

■ Για $t : \mathbb{N} \rightarrow \mathbb{R}_{\geq 0}$:

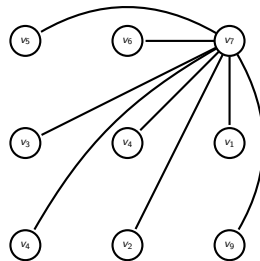
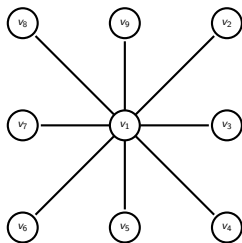
$$\text{NTIME}(t(n)) = \left\{ L \mid \begin{array}{l} L = L(M) \text{ για μη-ντετερμινιστική TM } M \\ \text{με } O(t(n)) \text{ χρονική πολυπλοκότητα} \end{array} \right\}$$

Κλάση NP

■ Έστω οι γράφοι $G_1 = (V_1, E_1)$, $G_2 = (V_2, E_2)$. Λέμε ότι ο G_1 είναι ισομορφικός με τον G_2 ($G_1 \cong G_2$) αν υπάρχει αμφιμονοσήμαντη συνάρτηση $f : V_1 \rightarrow V_2$ τ.ω.,

$$\forall u, v \in V_1, \quad (u, v) \in E_1 \iff (f(u), f(v)) \in E_2$$

π.χ.,



Κλάση NP

■ Ορίζουμε την γλώσσα $ISOG := \{\langle G_1, G_2 \rangle \mid G_1 \cong G_2\}$.

■ Θα δείξουμε ότι $ISOG \in \mathbf{NP}$.

Για πιστοποιητικό παίρνουμε την συνάρτηση $f : V_1 \rightarrow V_2$.

Κατασκευάζουμε την TM του επαληθευτή, έστω R ως εξής:

$R =$ “Για είσοδο $\langle G_1, G_2, f \rangle$:

1. Ελέγχει αν η f είναι αμφιμονοσήμαντη. Αν όχι, **απορρίπτει**.
2. $\forall u, v \in V_1$ ελέγχει αν ισχύει $(u, v) \in E_1 \iff (f(u), f(v)) \in E_2$.
3. Αν ισχύει για κάθε ζευγάρι κορυφών, **αποδέχεται**. Αλλιώς, **απορρίπτει**.”

□

Κλάση NP

Θεώρημα. Μια γλώσσα $A \in \mathbf{NP}$ ανν $\exists$ μη-ντετερμινιστική TM πολυωνυμικού χρόνου που να την διαγιγνώσκει.

Απόδειξη. (Ιδέα) Θα μετατρέψουμε έναν πολυωνυμικό επαληθευτή V σε έναν ισοδύναμο μη-ντετερμινιστικό διαγνώστη N , και αντιστρόφως.

[Ευθύ] Εάν ο V έχει χρόνο εκτέλεσης n^k , για κάποιο ακέραιο k , τότε κατασκευάζουμε τον N ως εξής:

$N =$ “Για είσοδο w μήκους n :

1. Επιλέγει με μη-ντετερμινιστικό τρόπο μια συμβολοσειρά c , με $|c| \leq n^k$.
2. Εκτελούμε τον V για είσοδο $\langle w, c \rangle$.
3. Αν V αποδέχεται, **αποδέχεται**.

Αλλιώς, **απορρίπτει**.”

...

Κλάση NP

...

Απόδειξη. [Αντιστρόφως] Τώρα A διαγνώσιμη από μια μη-ντετερμινιστική TM N πολυωνυμικού χρόνου. Θα κατασκευάσουμε έναν πολυωνυμικό επαληθευτή V ως εξής:

$V =$ “Για είσοδο $\langle w, c \rangle$:

1. Προσομοιώνουμε την N για είσοδο w , χρησιμοποιώντας κάθε σύμβολο της c ως μια περιγραφή της μη-ντετερμινιστικής επιλογής.
2. Αν ο συγκεκριμένος κλάδος υπολογισμού αποδέχεται, **αποδέχεται**.
Αλλιώς, **απορρίπτει**.”



Κλάση NP

■ NP είναι η **κλάση όλων των διαγνώσιμων σε μη-ντετερμινιστικό πολυωνυμικό χρόνο γλωσσών**,

$$NP = \bigcup_k NTIME(n^k)$$

- Η **NP** είναι αναλλοίωτη από την επιλογή του μη-ντετερμινιστικού μοντέλου [πληρώνουμε μόνο κάποιο πολυωνυμικό κόστος]
- Κάθε βήμα του υπολογισμού θα πρέπει να έχει μια “προφανή” υλοποίηση πολυωνυμικού χρόνου σε κάποιο μη-ντετερμινιστικό μοντέλο.
- Το πλήθος των σταδίων που εκτελούνται σε κάθε κλάδο του υπολογισμού θα πρέπει να είναι το πολύ πολυωνυμικό.
- Είναι κλειστή ως προς την παράθεση.

Κλάση NP

Μερικά σχόλια:

■ Έστω $L \in \mathbf{NP}$, αν υπάρχουν σταθερές $a > 0$ και $b \geq 0$, και πολυωνυμικός επαληθευτής V , τ.ω. $\forall w \in \Sigma^*$

$$w \in L \iff \exists c \in \Sigma^* \text{ τ.ω. } |c| \leq a \cdot |w|^b \text{ και } V(w, c) = 1$$

- Η συμβολοσειρά c είναι το πιστοποιητικό.
- Αν $w \in L$, τότε υπάρχει κάποιο πιστοποιητικό που κάνει τον επαληθευτή να αποδέχεται.
- Αν $w \notin L$, τότε κάθε πιστοποιητικό πρέπει να κάνει τον επαληθευτή να απορρίπτει.
- Το πιστοποιητικό μπορεί να είναι οποιοδήποτε συμβολοσειρά αρκεί να βοηθά τον επαληθευτή να πάρει τη σωστή απόφαση.
- Γενικά, θα χρησιμοποιήσουμε αυτήν την διατύπωση για να δείξουμε ότι μια γλώσσα ανήκει στο **NP**.

Κλάση NP

Μερικά σχόλια:

■ Έστω $L \in \mathbf{NP}$, αν υπάρχουν σταθερές $a > 0$ και $b \geq 0$, και πολυωνυμικός επαληθευτής V , τ.ω. $\forall w \in \Sigma^*$

$$w \in L \iff \exists c \in \Sigma^* \text{ τ.ω. } |c| \leq a \cdot |w|^b \text{ και } V(w, c) = 1$$

- Η συμβολοσειρά c είναι το πιστοποιητικό.
- Αν $w \in L$, τότε υπάρχει κάποιο πιστοποιητικό που κάνει τον επαληθευτή να αποδέχεται.
- Αν $w \notin L$, τότε κάθε πιστοποιητικό πρέπει να κάνει τον επαληθευτή να απορρίπτει.
- Το πιστοποιητικό μπορεί να είναι οποιοδήποτε συμβολοσειρά αρκεί να βοηθά τον επαληθευτή να πάρει τη σωστή απόφαση.
- Γενικά, θα χρησιμοποιήσουμε αυτήν την διατύπωση για να δείξουμε ότι μια γλώσσα ανήκει στο **NP**.

■ Άρα, $\Sigma\Gamma\mathbf{N}\Theta\mathbf{E}\mathbf{T}\mathbf{O}\Sigma \in \mathbf{NP}$ και $\mathbf{H}\mathbf{A}\mathbf{M}\mathbf{P}\mathbf{A}\mathbf{T}\mathbf{H} \in \mathbf{NP}$.

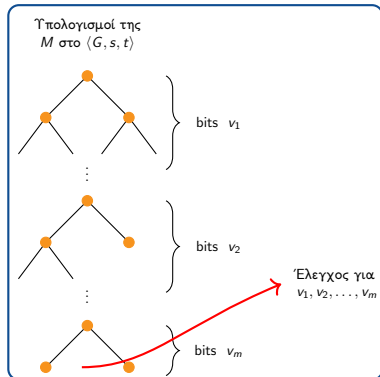
Μονοπάτι Hamilton

Θεώρημα. $HAMPATH \in NP$.

Απόδειξη. Έστω ότι το G έχει m κορυφές. Η ΤΜ M σε είσοδο $\langle G, s, t \rangle$ εκτελεί τις ακόλουθες ενέργειες:

$T =$ “Για είσοδο $\langle G, s, t \rangle$:

1. Επιλέγει με μη-ντετερμινιστικό τρόπο μια ακολουθία $(v_1, v_2, \dots, v_m)$ κορυφών.
2. **Αποδέχεται** αν:
3. $(v_1 = s, v_m = t) \wedge$
4. $((v_i, v_{i+1}) \in G, \forall i \in [m]) \wedge$
5. (οι v_i δεν επαναλαμβάνονται).
6. **Απορρίπτει**, αν οποιαδήποτε συνθήκη παύει να ισχύει.” $\square$



Σύνθετοι ακέραιοι

Θεώρημα. $\Sigma\Gamma\text{N}\Theta\text{E}\text{T}\text{O}\Sigma \in \text{NP}$.

Απόδειξη. Κατασκευάζουμε TM M σε είσοδο $\langle p \rangle$ η οποία εκτελεί τις ακόλουθες ενέργειες.

$M =$ “ Για είσοδο $\langle p \rangle$:

1. Επιλέγει με μη-ντετερμινιστικό τρόπο q , τ.ω. $1 \leq q \leq p$.

1. **Αποδέχεται** αν q διαιρεί τέλεια το p . Αλλιώς, **απορρίπτει**.” $\square$

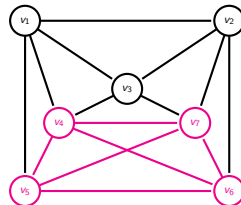
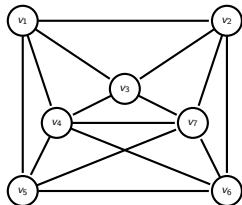
■ Αν αντί για δυαδική αναπαράσταση είχαμε δεκαδική αναπαράσταση, δεν θα άλλαζε κατά πολύ την ανάλυση καθώς μπορούμε να μετασχηματίσουμε τις αναπαραστάσεις σε πολυωνυμικό χρόνο.

■ Όμως δεν θα ήταν πρακτικό να γράψουμε τους αριθμούς σε μοναδιαία αναπαράσταση.
[θα καθιστούσε τους αριθμούς εκθετικά μεγαλύτερους!]

■ Το 2002 αποδείχθηκε ότι $\Sigma\Gamma\text{N}\Theta\text{E}\text{T}\text{O}\Sigma \in \text{P}$. [Δείτε [εδώ](#)]

Κλίκα

■ Έστω μη-κατευθυνόμενο γράφημα $G = (V, E)$. Κλίκα είναι ένα υπογράφημα όπου οποιηδήποτε δύο κόμβοι συνδέονται απευθείας με ακμή.



■ Μια κλίκα με k κόμβους ονομάζεται k -κλίκα.

■ Ορίζουμε την γλώσσα

$$CLIQUE := \{ \langle G, k \rangle \mid G \text{ μη-κατευθυνόμενο γράφημα που περιέχει } k\text{-κλίκα} \}.$$

Κλίκα

Θεώρημα. $CLIQUE \in NP$.

Απόδειξη. Χρησιμοποιούμε ως πιστοποιητικό ένα υπογράφημα $U \subseteq V(G)$.

Ο αλγόριθμος του επαληθευτή V είναι ο ακόλουθος:

$V =$ “ Για είσοδο $\langle G, k, U \rangle$:

1. Ελέγχει αν $U \subseteq V(G)$. Αν όχι, **απορρίπτει**.
2. Ελέγχει αν $|U| = k$. Αν όχι, **απορρίπτει**.
3. $\forall u, v \in U$ ελέγχει αν $(u, v) \in E$. Αν ισχύει για κάθε ζεύγος κορυφών, **αποδέχεται**.
Αλλιώς, **απορρίπτει**. ” □

Κλίκα

Θεώρημα. $CLIQUE \in NP$.

Απόδειξη. (Εναλλακτική) Μπορούμε να κατασκευάσουμε μια μη-ντετερμινιστική TM N πολυωνυμικού χρόνου ως εξής:

$N =$ “ Για είσοδο $\langle G, k \rangle$:

1. Επιλέγει μη-ντετερμινιστικά ένα σύνολο k κόμβων, έστω c .
2. Ελέγχει αν το G περιέχει όλες τις ακμές οι οποίες συνδέουν κόμβους του c .
3. Αν ναι **αποδέχεται**. Αλλιώς, **απορρίπτει**. □

Άθροισμα Υπακολουθίας

■ Έστω ακολουθία $x := \{x_i\}_{i \in [n]}$ και αριθμός $t \in \mathbb{N}$. Σκοπός είναι να βρούμε (εάν υπάρχει) υπακολουθία της x , έστω y , τ.ω. το άθροισμα των στοιχείων της να κάνει ακριβώς t .

π.χ., Αν $x = \{4, 4, 11, 17, 27\}$ και $t = 25$, τότε υπάρχει υπακολουθία και αυτή είναι η $x_k = \{x_1, x_2, x_4\}$.

■ Ορίζουμε την γλώσσα

$$SUMSUB := \left\{ \langle S, t \rangle \mid \begin{array}{l} S = \langle x_1, x_2, \dots, x_n \rangle \text{ και υπάρχει υπακολουθία} \\ x_{k_1}, \dots, x_{k_\ell} \text{ με } \sum_i x_{k_i} = t \end{array} \right\}.$$

Άθροισμα Υπακολουθίας

Θεώρημα. $SUMSUB \in NP$.

Απόδειξη. Χρησιμοποιούμε ως πιστοποιητικό την υπακολουθία.

Ο αλγόριθμος του επαληθευτή V είναι ο ακόλουθος:

$V =$ “ Για είσοδο $\langle\langle S, t \rangle, c\rangle$:

1. Ελέγχει αν c μια ακολουθία αριθμών με άθροισμα t .
2. Ελέγχει αν η c είναι υπακολουθία της S .
3. Εάν οι δύο έλεγχοι δώσουν θετικό αποτέλεσμα, **αποδέχεται**.

Αλλιώς, **απορρίπτει**.”



Κλάση co-NP

- Είναι η κλάση των προβλημάτων των οποίων το συμπλήρωμα είναι στη **NP**.

$$\text{co-NP} = \{L \mid \bar{L} \in \text{NP}\}$$

- Το να αποδείξουμε *δεν υπάρχει* φαίνεται να είναι δυσκολότερο από το να αποδείξουμε ότι *υπάρχει*.

π.χ., το συμπληρωματικό πρόβλημα του ισομορφισμού γράφων.

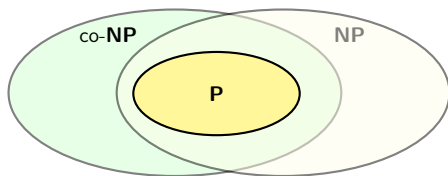
- Ένα ανοιχτό ερώτημα, **NP** = co-**NP**;

P vs NP

★ **P**, όλες οι γλώσσες για τις οποίες διαγιγνώσκουμε *γρήγορα* τα μέλη τους.

★ **NP**, όλες οι γλώσσες για τις οποίες επαληθεύουμε *γρήγορα* τα μέλη τους.

■ Αν μπορούμε να διαγνώσουμε *γρήγορα*, τότε μπορούμε να πιστοποιήσουμε *γρήγορα*.
Άρα, $P \subseteq NP$. **Ερώτηση**: Ισχύει ότι $P = NP$; [Cook '71].



★ **Εικασία**. $P \neq NP$ ★

■ Η ταχύτερη ντετερμινιστική μέθοδος για επίλυση προβλημάτων της **NP** απαιτεί χρόνο $NP \subseteq \bigcup_k (TIME(2^{n^k})), (EXP)$.

P vs NP

■ Έχουμε την γλώσσα $A_{CFG} := \{\langle G, w \rangle \mid G \text{ είναι CFG και } w \in L(G)\}$.

Θεώρημα. A_{CFG} είναι διαγνώσιμη.

Απόδειξη. Κατασκευάζουμε TM M το οποίο σε είσοδο $\langle G, w \rangle$ εκτελεί τις ακόλουθες ενέργειες:

$M =$ “ Για είσοδο $\langle G, w \rangle$:

1. Μετατρέπει την G σε CNF.
2. Δοκιμάζει όλα τα παραγόμενα μεγέθους $2|w| - 1$.
3. **Αποδέχεται** αν κάποια από τις παραγωγές δημιουργεί την w .

Αλλιώς **απορρίπτει**.”

□

P vs NP

Θεώρημα. $A_{CFG} \in NP$.

Απόδειξη. Κατασκευάζουμε TM M το οποίο σε είσοδο $\langle G, w \rangle$ η οποία εκτελεί τις ακόλουθες ενέργειες.

$M =$ “ Για είσοδο $\langle G, w \rangle$:

1. Επιλέγει με μη-ντετερμινιστικό τρόπο κάποιο παραγόμενο μεγέθους $2|w| - 1$.
2. **Αποδέχεται** αν δημιουργεί την w .
3. Αλλιώς **απορρίπτει**.” □

P vs NP

Θεώρημα. $A_{CFG} \in P$.

Απόδειξη. Χρησιμοποιούμε έναν επαναληπτικό αλγόριθμο C τ.ω. ελέγχει αν η G δημιουργεί την w ξεκινώντας από οποιαδήποτε μεταβλητή A . Ο C εκτελεί τις ακόλουθες ενέργειες:

$C =$ “ Για είσοδο $\langle G, w, A \rangle$:

1. Για κάθε δυνατή διαίρεση του w σε x και y , και για κάθε κανόνα $A \rightarrow ST$.
2. Χρησιμοποίησε τον C για να ελέγξεις τις $\langle G, x, S \rangle$ και $\langle G, y, T \rangle$.
3. **Αποδέχεται** αν κ οι δύο αποδέχονται.
4. **Απορρίπτει** αν δεν υπάρχει καμία αποδοχή.”

Έπειτα, διαγιγνώσκει A_{CFG} ξεκινώντας από την αρχική μεταβλητή της G . □

■ Ο C είναι ορθός αλγόριθμος, αλλά απαιτεί μη-πολυωνυμικό χρόνο. [κάθε επανάληψη χρειάζεται $O(n)$ διαιρέσεις και το ύψος είναι περίπου $\log n$]

P vs NP

■ Η 1η απόπειρα να δείξουμε ότι $A_{CFG} \in P$ απέτυχε.

■ Μπορούμε να χρησιμοποιήσουμε επανάληψη (όπως πριν) και κάποιου είδους μνήμης
Δυναμικός Προγραμματισμός.

■ Μια συμβολοσειρά w έχει το πολύ $O(n^2)$ υποσυμβολοσειρές, επομένως υπάρχουν μονάχα $O(n^2)$ υπό-προβλήματα $\langle G, x, S \rangle$.

Θεώρημα. $A_{CFG} \in P$.

(Διαισθητικά) Ο αλγόριθμος απομνημονεύει προηγούμενα αποτελέσματα. Εν συνεχεία, όπως στην προηγούμενη απόδειξη.

P vs NP

Θεώρημα. $A_{CFG} \in P$.

Απόδειξη. Χρησιμοποιούμε έναν επαναληπτικό αλγόριθμο D τ.ω. ελέγχει αν η G δημιουργεί την w ξεκινώντας από οποιαδήποτε μεταβλητή A . Ο D εκτελεί τις ακόλουθες ενέργειες:

$D =$ “ Για είσοδο $\langle G, w, A \rangle$:

0. Αν το $\langle G, w, A \rangle$ έχει επιλυθεί ήδη, δώσε την ίδια απάντηση, αλλιώς συνέχισε.
1. Για κάθε δυνατή διαίρεση του w σε x και y , και για κάθε κανόνα $A \rightarrow ST$.
2. Χρησιμοποίησε τον C για να ελέγξεις τις $\langle G, x, S \rangle$ και $\langle G, y, T \rangle$.
3. **Αποδέχεται** αν κ οι δύο αποδέχονται.
4. **Απορρίπτει** αν δεν υπάρχει καμία αποδοχή.”

Έπειτα, διαγιγνώσκει A_{CFG} ξεκινώντας από την αρχική μεταβλητή της G . $\square$

■ Το βήμα 0. λέγεται memoization.

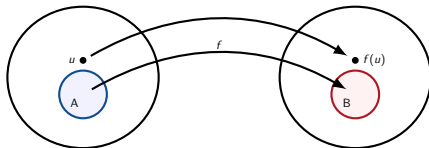
Αναγωγή σε Πολυωνυμικό Χρόνο

Ορισμός. Μια συνάρτηση $f : \Sigma^* \rightarrow \Sigma^*$ είναι **υπολογίσιμη σε πολυωνυμικό χρόνο** εάν $\exists$ TM M πολυωνυμικού χρόνου η οποία $\forall$ είσοδο w , τερματίζει έχοντας στην ταινία της μόνο της συμβολοσειρά $f(w)$.

■ Λέμε ότι η f είναι μια συνάρτηση πολυωνυμικού χρόνου.

Ορισμός. Έστω γλώσσες A και B . Λέμε ότι η A είναι **ανάγεται σε πολυωνυμικό χρόνο** στη B , $A \leq_p B$, αν υπάρχει συνάρτηση πολυωνυμικού χρόνου $f : \Sigma^* \rightarrow \Sigma^*$ τ.ω. $\forall w$ να ισχύει

$$w \in A \iff f(w) \in B$$



■ Λέμε ότι η f είναι η πολυωνυμικού χρόνου αναγωγή της A στην B .

Αναγωγή σε Πολυωνυμικό Χρόνο

■ Η αναγωγή πολυωνυμικού χρόνου είναι μια ταχεία απεικονιστική αναγωγή.

■ Μετατρέπουμε τον έλεγχο συμμετοχής στην A στον έλεγχο συμμετοχής στην B . [για να ελέγξουμε αν $w \in A$, μετατρέπουμε w σε $f(w)$ και ελέγχουμε $f(w) \in B$]

Θεώρημα. Αν $A \leq_p B$ και $B \in \mathbf{P}$, τότε $A \in \mathbf{P}$.

Απόδειξη. Έστω M αλγόριθμος πολυωνυμικού χρόνου που διαγιγνώσκει την B , και f πολυωνυμικού χρόνου αναγωγή της A στην B . Ένας αλγόριθμος που διαγιγνώσκει την A σε πολυωνυμικό χρόνο είναι ο εξής:

$N =$ “ Για είσοδο $\langle w \rangle$:

1. Υπολογίζουμε την λέξη $f(w)$.
2. Εκτελούμε τον M για είσοδο $f(w)$. Αποδίδουμε, ότι αποδίδει στην έξοδο του.”

...

Αναγωγή σε Πολυωνυμικό Χρόνο

Απόδειξη. (συνέχεια)

$N =$ “ Για είσοδο $\langle w \rangle$:

1. Υπολογίζουμε την λέξη $f(w)$.
2. Εκτελούμε τον M για είσοδο $f(w)$. Αποδίδουμε, ότι αποδίδει στην έξοδο του.”

Εφόσον, η f είναι η αναγωγή από την A στη B , γνωρίζουμε ότι $w \in A \iff f(w) \in B$.

Συνεπώς, ο M αποδέχεται την $f(w)$ ανν ο N αποδέχεται την w . Άρα ο N είναι ορθός.

Καθένα από τα στάδια του N εκτελείται σε πολυωνυμικό χρόνο, άρα και ο N χρειάζεται πολυωνυμικό χρόνο. □

Ικανοποιησιμότητα

■ Ένας λογικός τύπος (boolean formula) φ έχει λογικές μεταβλητές (p, q, r) και λογικούς τελεστές ($\vee, \wedge, \neg$). **π.χ.** $(p \wedge \neg q) \vee (r \vee q)$. [Για συντομία θα γράφουμε $\bar{x}$ αντί για $\neg x$.]

■ Λογικός τύπος είναι οποιαδήποτε έκφραση η οποία περιλαμβάνει λογικές μεταβλητές και πράξεις.

■ Λέμε ότι η φ είναι **ικανοποιήσιμη** αν αποτιμάται ως Αληθής για κάποια ανάθεση τιμών στις μεταβλητές.

π.χ. Ο λογικός τύπος $\varphi = (p \vee q) \wedge (\bar{p} \vee \bar{q})$ είναι ικανοποιήσιμος για $p = \text{Αληθές}$, $q = \text{Ψευδές}$.

■ Ορίζουμε την γλώσσα

$$SAT := \{ \langle \varphi \rangle \mid \text{η } \varphi \text{ είναι ένας ικανοποιήσιμος λογικός τύπος} \}$$

Θεώρημα. [Cook-Levin] $SAT \in P \iff P = NP$.

Ικανοποιησιμότητα

■ Θα δουλέψουμε με συγκεκριμένης μορφής τύπους

$$(q_1 \vee \dots \vee q_k) \wedge (p_1 \vee \dots \vee p_\ell) \wedge \dots \wedge (r_1 \vee \dots \vee r_m)$$

Δηλαδή, πρόκειται για συζεύξεις ($\wedge$) διαζεύξεων ($\vee$) μεταξύ literals (είτε μεταβλητή είτε άρνηση μεταβλητής). [Θα χρησιμοποιούμε κ την γραφή $\varphi_1 \wedge \varphi_2 \wedge \dots \wedge \varphi_k$, όπου φ_i πρόταση]

π.χ., $(x_1 \vee \bar{x}_2 \vee x_3) \wedge (y_1 \vee y_3) \wedge (\bar{z}_1 \vee z_2 \vee \bar{z}_3 \vee z_4)$

■ Η μορφή αυτή είναι η Συζευκτική Κανονική Μορφή (ΣΚΠ) (ή Conjunctive Normal Form (CNF)) [Προσοχή! μη την συγχέετε με την Chomsky Normal Form (CNF)].

■ Εάν ένας τύπος σε CNF περιέχει n αριθμός συμβόλων σε κάθε “υποομάδα εσωτερικών” διαζεύξεων λέγεται n -CNF.

π.χ., ο $(x_1 \vee \bar{x}_2) \wedge (y_1 \vee y_3) \wedge (\bar{z}_3 \vee z_4)$ είναι 2-CNF.

Ικανοποιησιμότητα

■ Θα αναφερόμαστε ως k -SAT για το πρόβλημα ικανοποιησιμότητας των k -CNF.

Θεώρημα. $3\text{-SAT} \leq_P \text{CLIQUE}$.

Απόδειξη. Έστω $\phi = (a_1 \vee b_1 \vee c_1) \wedge (a_2 \vee b_2 \vee c_2) \wedge \dots \wedge (a_n \vee b_n \vee c_n)$. Η αναγωγή θα πρέπει να παράγει την συμβολοσειρά $\langle G, k \rangle$, με G ένα μη-κατευθυνόμενο γράφημα:

κόμβοι, τα literals [χωρισμένα σε τριάδες, όπου κάθε τριάδα αντιστοιχεί σε μια C]

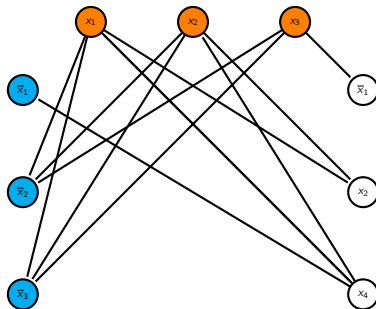
ακμές, μεταξύ κόμβων διαφορετικών τριάδων, εκτός από αντίθετα literals.

...

Ικανοποιησιμότητα

Απόδειξη. (συνέχεια)

π.χ., για $\phi = (x_1 \vee x_2 \vee x_3) \wedge (\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_3) \wedge (\bar{x}_1 \vee x_2 \vee x_4)$ ο γράφος G γίνεται,



[Προσοχή, ο γράφος δεν είναι ολοκληρωμένος!]

...

Ικανοποιησιμότητα

Απόδειξη. (συνέχεια) **Ισχυρισμός.** ϕ είναι ικανοποιήσιμος ανν ο G περιέχει k -κλίκα.

$[\Rightarrow]$ Έστω μια ανάθεση που ικανοποιεί το ϕ και για φ_i επιλέγουμε ένα literal ℓ_i^* που είναι αληθές σύμφωνα με αυτή την ανάθεση.

Στο σύνολο $L = \{\ell_i^* \mid i = 1, \dots, k\}$ παίρνουμε ένα literal ανά φ_i . Άρα, κάθε δύο επιλεγμένοι κόμβοι από το L δεν ανήκουν στο ίδιο φ_i .

Επίσης το L δεν περιέχει κανένα ζεύγος συμπληρωματικών literals.

Άρα το G περιέχει μια k -κλίκα.

Ικανοποιησιμότητα

Απόδειξη. (συνέχεια)

[$\Leftarrow$] Έστω L οι κορυφές μιας κλίκας μεγέθους k στο G .

Δεδομένου ότι καμία κλίκα του G δεν μπορεί να περιέχει δύο literals από την ίδια φ_i

Άρα υπάρχουν ακριβώς k φ , καθώς το L περιέχει ακριβώς ένα literal ℓ_i^* από κάθε φ_i .

Επιπλέον, αυτό το σύνολο literals δεν μπορεί να περιέχει κανένα ζεύγος συμπληρωματικών literals.

Μπορούμε να βρούμε μια ανάθεση για το ϕ ορίζοντας τις μεταβλητές σύμφωνα με τις τιμές των literals στο L . Οι “υπόλοιπες” μεταβλητές (αν υπάρχουν) που δεν αντιστοιχούν σε κανένα literal στο L μπορούν να οριστούν αυθαίρετα. $\square$

NP-πλήρης

■ Αν το *CLIQUE* επιλύεται σε πολυωνυμικό χρόνο, τότε και το 3-*SAT* θα επιλύεται σε πολυωνυμικό χρόνο.

■ Μια γλώσσα B είναι **NP**-πλήρης εάν ικανοποιεί τις εξής συνθήκες:

1 $B \in \mathbf{NP}$.

2 Κάθε γλώσσα $A \in \mathbf{NP}$ ανάγεται σε πολυωνυμικό χρόνο στην B . [**NP-hard**]

■ Αν B είναι **NP**-πλήρης και $B \in \mathbf{P}$, τότε $\mathbf{P} = \mathbf{NP}$.

■ [Cook-Levin] $\mathbf{SAT} \in \mathbf{NP}$ -πλήρης.

Μια Ιεραρχία

