



# ΗΥ - 280 Θεωρία Υπολογισμού

## Διάλεξη 20

Κωνσταντίνος Βάρσος

Τμήμα Επιστήμης Υπολογιστών – Πανεπιστήμιο Κρήτης

Χειμερινό εξάμηνο 2025 - 26

# Πολυπλοκότητα

## Τι είδαμε στην Διάλεξη 19

- Πολυωνυμικός επαληθευτής.
- Κλάση **NP**.
- **P** vs **NP**.
- Πολυωνυμική αναγωγή.

## 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]  $\text{SAT} \in \mathbf{NP}$ -πλήρης.

## NP-πληρότητα

**Θεώρημα.** Αν  $B$  είναι **NP**-πλήρης,  $C \in \mathbf{NP}$ , και  $B \leq_P C$ , τότε και η  $C$  είναι **NP**-πλήρης.

**Απόδειξη.**  $C \in \mathbf{NP}$ , άρα αρκεί να δείξουμε ότι κάθε  $A \in \mathbf{NP}$  ανάγεται στην  $C$  σε πολυωνυμικό χρόνο.

Εφόσον, η  $B$  είναι **NP**-πλήρης, κάθε  $A \in \mathbf{NP}$  ανάγεται στην  $B$  σε πολυωνυμικό χρόνο, η οποία με την σειρά της ανάγεται στην  $C$  σε πολυωνυμικό χρόνο.

Η σύνθεση αναγωγών πολυωνυμικού χρόνου δίνει αναγωγή πολυωνυμικού χρόνου. □

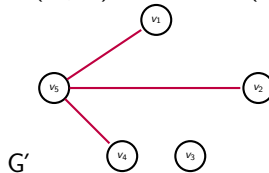
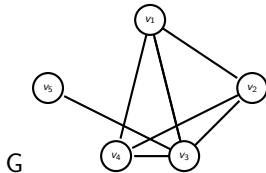
## NP-πληρότητα

■ Έστω η γλώσσα

$$INDSET = \{ \langle G, k \rangle \mid G \text{ γράφος και } k \text{ το πλήθος κορυφών που δεν συνδέονται} \}$$

■ Θα δείξουμε ότι  $INDSET \leq_P CLIQUE$ .

Έστω  $G \in INDSET$ . Κατασκευάζουμε νέο γράφο  $G'$  τ.ω.  $(u', v') \in E' \iff (u, v) \notin E$ .



## Αναγωγή σε Πολυωνυμικό Χρόνο

### Μερικά σχόλια

- 1** Αν ήδη γνωρίζουμε έναν αλγόριθμο για το  $A$  και  $B \leq_P A$ , τότε μπορούμε να χρησιμοποιήσουμε τη αναγωγή για να κατασκευάσουμε έναν αλγόριθμο για το  $B$ . Αυτό είναι ένα ευρέως χρησιμοποιούμενο εργαλείο στον σχεδιασμό αλγορίθμων.
- 2** Αν έχουμε αποδείξει (ή έχουμε ενδείξεις) ότι δεν υπάρχει αλγόριθμος πολυωνυμικού χρόνου για το  $B$  και  $B \leq_P A$ , τότε η ύπαρξη αυτής της αναγωγής μας επιτρέπει να συμπεράνουμε ότι δεν υπάρχει αλγόριθμος πολυωνυμικού χρόνου για το  $A$ .

## NP-πληρότητα

■ Ένα πρόβλημα  $B$  λέμε ότι είναι **NP**-πλήρες αν

1  $A \in \mathbf{NP}$

2 Για κάθε  $A \in \mathbf{NP}$ ,  $A \leq_P B$ .

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

■ Αν για ένα πρόβλημα  $A$  βρούμε ότι είναι **NP**-πλήρες, τότε είναι ένδειξη ότι οι υπολογισμοί για το  $A$  είναι δύσχρηστοι (intractable).

■ Για να δείξουμε ότι μια γλώσσα  $C$  είναι **NP**-πλήρης, αρκεί να δείξουμε ότι υπάρχει  $\leq_P$  σε κάποια άλλη **NP**-πλήρη γλώσσα.

■ Άρα, αν αποδείξουμε την **NP**-πληρότητα ενός προβλήματος τότε χρησιμοποιώντας πολυωνυμική αναγωγή μπορούμε να προσδιορίσουμε την **NP**-πληρότητα και άλλων προβλημάτων.

## NP-πληρότητα

**Παράδειγμα.** Θεωρήστε την γλώσσα

$$\text{Μακρά Διαδρομή} = \left\{ \langle G, a, b, k \rangle \mid \begin{array}{l} \text{το } G \text{ περιέχει απλή διαδρομή μήκους} \\ \text{το λιγότερο } k \text{ από το } a \text{ στο } b \end{array} \right\}$$

Να δείξετε ότι το πρόβλημα Μακρά Διαδρομή  $\in$  **NP**-πλήρης.

## NP-πληρότητα

**Παράδειγμα.** Θεωρήστε την γλώσσα

$$\text{Μακρά Διαδρομή} = \left\{ \langle G, a, b, k \rangle \mid \begin{array}{l} \text{το } G \text{ περιέχει απλή διαδρομή μήκους} \\ \text{το λιγότερο } k \text{ από το } a \text{ στο } b \end{array} \right\}$$

Να δείξετε ότι το πρόβλημα Μακρά Διαδρομή  $\in$  **NP**-πλήρης.

**Λύση.** Πρώτα δείχνουμε Μακρά Διαδρομή  $\in$  **NP**. Οι γλώσσες στην **NP** έχουν επαληθευτές πολυωνυμικού χρόνου. Κατασκευάζουμε έναν επαληθευτή  $V$  ως εξής:

$V =$  “ Στην είσοδο  $\langle G, a, b, k, c \rangle$ , με  $c$  μονοπάτι:

1. Ελέγχει αν  $c$  είναι μη-επενалаμβανόμενη ακολουθία κορυφών του  $G$ .
2. Ελέγχει αν  $c$  τερματίζει στην  $b$  κορυφή του  $G$ .
3. Ελέγχει αν το μήκος της  $c$  μεγαλύτερο  $k$ .
4. Αν ικανοποιεί τα προηγούμενα βήματα, **αποδέχεται**. Αλλιώς, **απορρίπτει**.”

Ο επαληθευτής χρειάζεται  $O(|c|)$  επαναλήψεις στο βήμα 3. και από  $O(|V|)$  στα βήματα 1. και 2.. Συνεπώς, Μακρά Διαδρομή  $\in$  **NP**. ...

## NP-πληρότητα

...

Έπειτα δείχνουμε  $HAMPATH \leq_P \text{Μακρά Διαδρομή}$ .

Έστω στιγμιότυπο  $\langle G, a, b \rangle$  του  $HAMPATH$ , όπου  $G = (V, E)$  είναι ένας γράφος που η κορυφή εκκίνησης είναι η  $a$  και τερματισμού η  $b$ .

Η απεικόνιση παίρνει το  $\langle G, a, b \rangle$ , θέτει  $k = |V| - 1$ , τότε  $\langle G, a, b, k \rangle \in \text{Μακρά Διαδρομή}$ , το οποίο μπορεί να γίνει σε χρόνο  $O(|V| + |E|)$ . Θα πρέπει να δείξουμε ότι

$$\langle G, a, b \rangle \in HAMPATH \iff \langle G, a, b, k \rangle \in \text{Μακρά Διαδρομή}$$

...

## NP-πληρότητα

...

Αν το  $\langle G, a, b \rangle \in \text{HAMPATH}$ , τότε το  $G$  έχει απλό μονοπάτι Hamilton από την  $a$  στην  $b$ . Το οποίο συνεπάγεται ότι  $|V| - 1 = k$ . Άρα,  $\langle G, a, b, k \rangle \in \text{Μακρά Διαδρομή}$ .

Αν  $\langle G, a, b, k \rangle \in \text{Μακρά Διαδρομή}$ , τότε υπάρχει απλό μονοπάτι από την  $a$  στην  $b$  μήκους  $k = |V| - 1$ .

Καθώς το  $G$  έχει πλήθος κορυφών  $k + 1$  θα πρέπει το μονοπάτι να διατρέχει τις κορυφές του  $G$  ακριβώς μια φορά. Άρα το απλό μονοπάτι είναι μονοπάτι Hamilton. Συνεπώς,  $\langle G, a, b, k \rangle \in \text{Μακρά Διαδρομή}$ .

Εφόσον αποδείξαμε και τα δύο κριτήρια, Μακρά Διαδρομή  $\in$  **NP**-πλήρης. □

## NP-πληρότητα

- Εάν ένα πρόβλημα είναι **NP**-πλήρες, τότε υπό την υπόθεση ότι  $P \neq NP$ , δεν μπορεί να υπάρχει ένας αποτελεσματικός αλγόριθμος για αυτό.
- Κατά μία έννοια, τα **NP**-πλήρη προβλήματα είναι τα πιο δύσκολα προβλήματα στην κλάση **NP**.
- Όλα τα γνωστά **NP**-πλήρη προβλήματα είναι εξαιρετικά δύσκολο να λυθούν.
- Όλοι οι γνωστοί αλγόριθμοι για **NP**-πλήρη προβλήματα εκτελούνται σε εκθετικό χρόνο (στην ανάλυση χειρότερης περίπτωσης).
- Οι περισσότεροι αλγόριθμοι για **NP**-πλήρη προβλήματα είναι μη-εφικτοί για είσοδο εύλογου μεγέθους.

## NP-πληρότητα

- Τα **NP**-πλήρη προβλήματα προσφέρουν μια πολλά υποσχόμενη προσέγγιση για την επίλυση  $P \stackrel{?}{=} NP$ .
- Εάν οποιοδήποτε **NP**-πλήρες πρόβλημα βρίσκεται στο **P**, τότε  $P = NP$ .
- Εάν οποιοδήποτε **NP**-πλήρες πρόβλημα δεν βρίσκεται στο **P**, τότε  $P \neq NP$ .
- Ωστόσο, δεν έχουμε δείξει ότι κάποια προβλήματα είναι **NP**-πλήρη εξ αρχής! Πώς ξέρουμε καν ότι υπάρχουν;

## Ικανοποιησιμότητα

- Ένας προτασιακός λογικός τύπος  $\phi$  ονομάζεται ικανοποιήσιμος εάν υπάρχει κάποια ανάθεση στις μεταβλητές του που τον κάνει να αξιολογηθεί ως αληθής.
- Μια ανάθεση αληθούς και ψευδούς στις μεταβλητές του  $\phi$  που τον κάνει να αξιολογηθεί ως αληθής ονομάζεται ανάθεση ικανοποιησιμότητας.

### Το Πρόβλημα Ικανοποιησιμότητας (SAT)

Δοθείσας ενός λογικού προτασιακού τύπου  $\phi$ ,

είναι ο  $\phi$  ικανοποιήσιμος ;

- Ορίζουμε την γλώσσα

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

## Θεώρημα Cook-Levin

■ Ένας λογικός τύπος είναι ικανοποιήσιμος αν μπορούμε να εκχωρήσουμε τιμές στα  $x_1, \dots, x_n$  έτσι ώστε να γίνει αληθής ο τύπος  $\phi(x_1, \dots, x_n)$ .

■ Ένας λογικός τύπος  $\phi$  βρίσκεται σε CNF αν ισχύει:

$$\phi(x_1, \dots, x_n) = \bigwedge_{i=1}^m \phi_i$$

όπου κάθε  $\phi_i$  είναι μία διάζευξη literals (μεταβλητή  $x$  ή το συμπλήρωμά της).

**Θεώρημα.** Το πρόβλημα SAT είναι **NP**-πλήρες.

## Απόδειξη – Οι Κύριες Ιδέες

- $SAT \in \mathbf{NP}$  καθώς, δίνοντας μία ανάθεση τιμών στα  $x_1, \dots, x_n$ , μπορούμε να επαληθεύσουμε αν  $\phi(x_1, \dots, x_n) = 1$  σε πολυωνυμικό χρόνο.
- Τώρα πρέπει να δείξουμε ότι υπάρχει μία πολυωνυμική αναγωγή  $A \leq_P SAT$  για κάθε  $A \in \mathbf{NP}$ .
- $A \in \mathbf{NP}$  σημαίνει ότι υπάρχει μία μη-ντετερμινιστική ΤΜ  $N$  που τρέχει σε  $O(n^k)$  χρόνο και διαγιγνώσκει το  $A$ , για κάποιο  $k \in \mathbb{N}$ .
- Η  $N$  έχει  $K$  σύνολο καταστάσεων και  $\Gamma$  αλφάβητο ταινίας.
- Θα κατασκευάσουμε έναν λογικό τύπο  $\phi$  που είναι ικανοποιήσιμος αν και μόνο αν κάποιο κλάδος του υπολογισμού της  $N$  αποδέχεται μία είσοδο  $w$ .

## Απόδειξη – Περιγραφή της αναγωγής

■ Ονομάζουμε μητρώο της  $N$  για κάποια συμβολοσειρά  $w$  οποιονδήποτε πίνακα  $n^k \times n^k$  του οποίου οι γραμμές περιέχουν τις διάφορες φάσεις του υπολογισμού της  $N$  για  $w$ .

■ Κάθε αποδεκτικό στοιχείο της  $N$  επί της  $w$  αντιστοιχεί σε κάποιον αποδεκτικό κλάδο υπολογισμού της  $N$  επί της  $w$ . Άρα για να προσδιοριστεί αν η  $N$  αποδέχεται την  $w$ , αρκεί να υπάρχει αποδεκτικό μητρώο.

π.χ.,

|   |       |       |       |       |     |       |          |     |          |   |
|---|-------|-------|-------|-------|-----|-------|----------|-----|----------|---|
| # | $q_0$ | $w_1$ | $w_2$ | $w_3$ | ... | $w_n$ | $\sqcup$ | ... | $\sqcup$ | # |
| # | $a$   | $q_7$ | $w_2$ |       | ... |       |          |     |          | # |
| ⋮ |       |       |       |       |     |       |          |     |          | ⋮ |
| # |       |       |       |       |     |       |          |     |          | # |

I) Κάθε φάση αρχίζει και τελειώνει με #. II) Η 1η γραμμή περιέχει την εναρκτήρια φάση του υπολογισμού. III) Κάθε επόμενη φάση έπεται της προηγούμενης βάσει του πίνακα μεταβάσεων της  $N$ .

## Απόδειξη – Περιγραφή της αναγωγής

■ Εάν οποιαδήποτε γραμμή του πίνακα περιέχει αποδεκτική φάση, λέμε ότι το μητρώο είναι αποδεκτικό.

**Αναγωγή.** Για είσοδο  $w$ , η  $f$  επιστρέφει έναν τύπο  $\phi$ . Έστω  $C = \{K\} \cup \{\Gamma\} \cup \{\#\}$ .

■  $\forall i, j = 1, \dots, n^k$  και  $\forall s \in C$  φτιάχνουμε την μεταβλητή  $x_{i,j,s}$ .

■ Με τις  $x_{i,j,s}$  θα αναπαραστήσουμε τα περιεχόμενα του μητρώου.

■ Το μητρώο έχει  $(n^k)^2$  θέσεις, καθεμία εκ των οποίων περιέχει ένα σύμβολο από το  $C$ .

■ Εάν  $x_{i,j,s} = 1$ , τότε η θέση  $i, j$  του μητρώου περιέχει το σύμβολο  $s$ .

## Απόδειξη – Κατασκευή $\phi$

**Στόχος.** Κάθε ικανοποιήσιμη αποτίμηση στις μεταβλητές να αντιστοιχεί σε κάποιο αποδεκτικό μητρώο της  $A$  για την  $w$ .

■ Θέτουμε  $\phi = \phi_{\text{cell}} \wedge \phi_{\text{start}} \wedge \phi_{\text{move}} \wedge \phi_{\text{acc}}$ .

■ Ένα καλώς ορισμένο μητρώο σημαίνει ότι σε κάθε θέση υπάρχει ακριβώς ένα στοιχείο.

■ Στην Προτασιακή Λογική αυτό μεταφράζεται στην φόρμουλα,

$$\phi_{ij} = \left( \bigvee_{s \in C} x_{i,j,s} \right) \wedge \left( \bigwedge_{s,t \in C, s \neq t} (\overline{x_{i,j,s}} \vee \overline{x_{i,j,t}}) \right).$$

Επομένως, θέτουμε

$$\phi_{\text{cell}} = \bigwedge_{1 \leq i,j \leq n^k} \phi_{i,j}.$$

## Απόδειξη – Κατασκευή $\phi$

$$\phi_{cell} = \bigwedge_{1 \leq i, j \leq n^k} \left( \bigvee_{s \in C} x_{i,j,s} \right) \wedge \left( \bigwedge_{s, t \in C, s \neq t} (\overline{x_{i,j,s}} \vee \overline{x_{i,j,t}}) \right).$$

- Μια τουλάχιστον από τις μεταβλητές της κάθε θέσης είναι ενεργοποιημένη.
- Σε κάθε ζεύγος κορυφών για την κάθε θέση, τουλάχιστον μια είναι απενεργοποιημένη.

Άρα κάθε αποτίμηση η οποία καθιστά την  $\phi$  αληθή ενεργοποιεί σε κάθε θέση μία και μόνο μια από τις αντίστοιχες μεταβλητές.

■ Άρα κάθε τέτοια αποτίμηση ορίζει ένα σύμβολο σε κάθε θέση του μητρώου.

## Απόδειξη – Κατασκευή $\phi$

Ο τύπος  $\phi_{\text{start}}$  εξασφαλίζει ότι η 1η γραμμή του μητρώου περιέχει την εναρκτήρια φάση της  $N$  για είσοδο  $w$ ,

|   |       |       |       |       |         |       |          |         |          |   |
|---|-------|-------|-------|-------|---------|-------|----------|---------|----------|---|
| # | $q_0$ | $w_1$ | $w_2$ | $w_3$ | $\dots$ | $w_n$ | $\sqcup$ | $\dots$ | $\sqcup$ | # |
|---|-------|-------|-------|-------|---------|-------|----------|---------|----------|---|

$$\begin{aligned}
 \phi_{\text{start}} = & x_{1,1,\#} \wedge x_{1,2,q_0} \\
 & \wedge x_{1,3,w_1} \wedge x_{1,4,w_2} \wedge \dots \wedge x_{1,n+2,w_n} \\
 & \wedge x_{1,n+3,\sqcup} \wedge x_{1,n+4,\sqcup} \wedge \dots \wedge x_{1,n^k-1,\sqcup} \wedge x_{1,n^k,\#}
 \end{aligned}$$

## Απόδειξη – Κατασκευή $\phi$

Ο τύπος  $\phi_{acc}$  εξασφαλίζει ότι το μητρώο περιέχει κάποια αποδεκτή φάση. Δηλαδή, το  $q_{acc}$  εμφανίζεται σε κάποια από τις θέσεις του μητρώου.

Αυτό με την σειρά του επιβάλλει την ενεργοποίηση κάποιας από τις αντίστοιχες μεταβλητές.

$$\phi_{acc} = \bigvee_{1 \leq i, j \leq n^k} x_{i,j,q_{acc}}$$

## Απόδειξη – Κατασκευή $\phi$

Ο τύπος  $\phi_{\text{move}}$  εξασφαλίζει ότι η φάση που περιέχεται σε κάθε γραμμή του μητρώου έπεται από τη φάση της προηγούμενης γραμμής, σύμφωνα με τους κανόνες της  $N$ .

Κάθε παράθυρο ( $2 \times 3$  υποπίνακας) είναι έγκυρος,

|       |       |       |
|-------|-------|-------|
| $a_1$ | $a_2$ | $a_3$ |
| $a_4$ | $a_5$ | $a_6$ |

Ένα παράθυρο είναι έγκυρο αν δεν παραβιάζει τη συνάρτηση μετάβασης της  $N$ .

π.χ., Έστω TM  $N$  με μετάβαση  $\delta(q_1, b) = \{(q_2, c, L), (q_2, a, R)\}$  είναι έγκυρα τα

|       |       |     |
|-------|-------|-----|
| $a$   | $q_1$ | $b$ |
| $q_2$ | $a$   | $c$ |

|     |       |       |
|-----|-------|-------|
| $a$ | $q_1$ | $b$   |
| $a$ | $a$   | $q_2$ |

|     |     |       |
|-----|-----|-------|
| $a$ | $c$ | $q_1$ |
| $a$ | $c$ | $a$   |

|     |     |     |
|-----|-----|-----|
| $a$ | $b$ | $a$ |
| $a$ | $b$ | $a$ |

όχι όμως τα

|       |       |     |
|-------|-------|-----|
| $a$   | $q_1$ | $b$ |
| $q_1$ | $b$   | $b$ |

, 

|       |       |     |
|-------|-------|-----|
| $a$   | $q_1$ | $b$ |
| $q_1$ | $a$   | $a$ |

, 

|       |       |       |
|-------|-------|-------|
| $b$   | $q_1$ | $b$   |
| $q_1$ | $b$   | $q_2$ |

.

## Απόδειξη – Κατασκευή $\phi$

**Παρατήρηση.** Κάθε γραμμή στο μητρώο είναι μια φάση που έπεται από την προηγούμενη γραμμή σύμφωνα με την  $N$  ανν κάθε παράθυρο είναι έγκυρο.

**Απόδειξη.** Για κάθε γραμμή  $i$ , η φάση την γραμμή  $i + 1$  μπορεί να διαφέρει από την γραμμή  $i$  το πολύ σε 3 διαδοχικές θέσεις.

Επομένως, ελέγχοντας όλα τα έγκυρα παράθυρα είναι το ίδιο με το να ελέγξουμε ότι το μητρώο είναι έγκυρο σύμφωνα με το  $N$ . □

**Παρατήρηση.** Ο αριθμός των έγκυρων παραθύρων είναι πεπερασμένος ( $\leq |C|^6$ ).

## Απόδειξη – Κατασκευή $\phi$

Άρα, η συνθήκη ότι κάθε γραμμή έπεται από την προηγούμενη σύμφωνα με το  $N$  εκφράζεται ως εξής:

$$\phi_{move} = \bigvee_{1 \leq i < n^k, 1 < j < n^k} \phi_{window,i,j},$$

όπου

$$\phi_{window,i,j} = \bigvee_{(a_1, \dots, a_6) \text{ έγκυρο}} (x_{i,j-1,a_1} \wedge x_{i,j,a_2} \wedge x_{i,j+1,a_3} \wedge x_{i+1,j-1,a_4} \wedge x_{i+1,j,a_5} \wedge x_{i+1,j+1,a_6}).$$

## Απόδειξη – Σύνθεση

■ Δοθείσας μιας μη-ντετερμινιστικής TM  $N$  και εισόδου  $w$  δείξαμε ότι υπάρχει τύπος  $\phi$  που ορίζεται ως

$$\phi_{N,w} = \phi_{\text{cell}} \wedge \phi_{\text{start}} \wedge \phi_{\text{move}} \wedge \phi_{\text{acc}}$$

ο οποίος είναι ικανοποιήσιμος όταν η  $N$  αποδέχεται την  $w$ .

■ Οι υπο-φόρμουλες που κωδικοποιούν τις 4 συνθήκες θα πρέπει να έχουν ένα αποδεκτικό μητρώο για τον υπολογισμό της  $N$  στην  $w$ .

■ Μένει να δείξουμε ότι η αναγωγή είναι υπολογίσιμη σε πολυωνυμικό χρόνο.

## Απόδειξη – Πολυωνυμική αναγωγή

- Έστω ότι η  $N$  τρέχει σε  $O(n^k)$  για είσοδο μεγέθους  $n$ , άρα το μητρώο έχει  $O(n^k)$  γραμμές και  $O(n^k)$  στήλες.
- Η  $\phi$  που κατασκευάσαμε από την αναγωγή έχει  $O(n^{2k})$  literals, καθώς υπάρχει σταθερού μεγέθους φόρμουλα για κάθε θέση του μητρώου.
- Η  $\phi$  μπορεί να κατασκευαστεί αποδοτικά από την περιγραφή της μη-ντετερμινιστικής TM  $N$ .
- Καταλήξαμε σε αναγωγή με πολυωνυμικό χρόνο στο μέγεθος της εισόδου.
- Συνεπώς,  $A \leq_P SAT$ . Μπορούμε να κατασκευάσουμε έναν τύπο  $\phi_{N,w}$  σε πολυωνυμικό χρόνο, ο οποίο να είναι ικανοποιήσιμος ανν  $w \in A$ . □
- Παρατηρείστε ότι δεν αλλάζει κάτι στην ανάλυση αν αντί για  $n^k$  πάρουμε  $O(n^k)$ . [που είναι και το ορθότερο]

## NP-πληρότητα

■ Ένας προτασιακός τύπος είναι σε 3-*CNF* μορφή αν

- Είναι σε κανονική διαζευκτική μορφή.
- Κάθε όρος του έχει ακριβώς τρεις ατομικές προτάσεις (literals).

π.χ.,  $(x \vee y \vee z) \wedge (\neg x \vee \neg y \vee z), (x_1 \vee x_2 \vee x_3) \wedge (y_1 \vee \neg y_2 \vee y_3) \wedge (\neg z_1 \vee z_2 \vee z_3)$ , κτλ.

■ Ορίζουμε την γλώσσα 3-*SAT*

$$3 - SAT := \{ \langle \varphi \rangle \mid \text{ο } \varphi \text{ είναι ένας ικανοποιήσιμος 3-} CNF \text{ λογικός τύπος} \}$$

## NP-πληρότητα

**Θεώρημα.**  $3\text{-SAT} \in \text{NP-πλήρες}$ .

**Απόδειξη.** Θα χρησιμοποιήσουμε την αναγωγή  $\text{SAT} \leq_P 3\text{-SAT}$ . Αρχικά μερικές παρατηρήσεις

■ Πώς διαφέρει το  $\text{SAT}$  από το  $3\text{-SAT}$ ;

- Στο  $\text{SAT}$  οι προτάσεις μπορεί να έχουν οποιοδήποτε μήκος:  $1, 2, 3, \dots$  μεταβλητές.
- Στο  $3\text{-SAT}$  κάθε πρόταση πρέπει να έχει ακριβώς 3 διαφορετικά κυριολεκτικά.

■ Βασική ιδέα αναγωγής από  $\text{SAT}$  σε  $3\text{-SAT}$ :

- 1 Επέκταση σύντομων προτάσεων ώστε να έχουν 3 κυριολεκτικά.
- 2 Διαχωρισμός εκτενών προτάσεων σε μικρότερες μικρότερες.
- 3 Επανάληψη της διαδικασίας μέχρι να επιτευχθεί 3-CNF.

## NP-πληρότητα

**Απόδειξη.** (συνέχεια) Δίνεται μια  $\phi \in SAT$ , δημιουργούμε μια  $\phi' \in 3-SAT$ , ως εξής:

- 1 Η φόρμουλα  $\phi$  είναι ικανοποιήσιμη αν και μόνο αν η φόρμουλα  $\phi'$  είναι ικανοποιήσιμη.
- 2 Η φόρμουλα  $\phi'$  μπορεί να κατασκευαστεί από την  $\phi$  σε πολυωνυμικό χρόνο ως προς το μέγεθος της  $\phi$ .

Θα κάνουμε το εξής: Αν μια πρόταση της  $\phi$  δεν έχει μήκος 3, την αντικαθιστούμε με αρκετές προτάσεις μήκους ακριβώς 3.

Θα χρησιμοποιήσουμε το εξής αποτέλεσμα: Έστω  $X, Y$  λογικοί τύποι και  $z$  μεταβλητή.  $X \vee Y$  ικανοποιήσιμος τύπος ανν  $(X \vee z) \wedge (Y \vee \bar{z})$  είναι ικανοποιήσιμος τύπος.

## NP-πληρότητα

**Περίπτωση 1:** πρόταση με 2 literals  $c = \ell_1 \vee \ell_2$ . Έστω  $u$  μια νέα μεταβλητή. Η νέα πρόταση είναι:

$$c' = (\ell_1 \vee \ell_2 \vee u) \wedge (\ell_1 \vee \ell_2 \vee \bar{u})$$

Έστω,  $\phi = \psi \wedge c$ , τότε η  $\phi' = \psi \wedge c'$  είναι ικανοποιήσιμη αν και μόνο αν η  $\phi$  είναι ικανοποιήσιμη.

## NP-πληρότητα

**Απόδειξη.** (συνέχεια)

**Περίπτωση 2:** πρόταση με 1 literal  $c = \ell$ . Έστω  $u, v$  νέες μεταβλητές. Η νέα πρόταση είναι:

$$c' = (\ell \vee u \vee v) \wedge (\ell \vee u \vee \bar{v}) \wedge (\ell \vee \bar{u} \vee v) \wedge (\ell \vee \bar{u} \vee \bar{v})$$

Έστω,  $\phi = \psi \wedge c$ , τότε η  $\phi' = \psi \wedge c'$  είναι ικανοποιήσιμη αν και μόνο αν η  $\phi$  είναι ικανοποιήσιμη.

**Περίπτωση 3:** πρόταση με  $|\text{literal}| = k > 3$ , π.χ.  $c = \ell_1 \vee \ell_2 \vee \dots \vee \ell_{k-1} \vee \ell_k$ . Έστω  $u$  μια νέα μεταβλητή. Η νέα πρόταση είναι:

$$c' = (\ell_1 \vee \ell_2 \vee \ell_{k-2} \vee u) \wedge (\ell_{k-1} \vee \ell_k \vee \bar{u})$$

Εξακολουθητικά έως ότι να μην υπάρχει πρόταση με  $|\text{literal}| = k > 3$ . [Θα χρειαστούμε  $O(k)$ ]

Έστω,  $\phi = \psi \wedge c$ , τότε η  $\phi' = \psi \wedge c'$  είναι ικανοποιήσιμη αν και μόνο αν η  $\phi$  είναι ικανοποιήσιμη.

## NP-πληρότητα

π.χ., Έστω ο τύπος  $\in SAT$ :

$$\begin{aligned}\phi = & (\bar{x}_1 \vee \bar{x}_4) \wedge (x_1 \vee \bar{x}_2 \vee \bar{x}_3) \\ & \wedge (\bar{x}_2 \vee \bar{x}_3 \vee x_4 \vee x_1) \wedge (x_1)\end{aligned}$$

Ο ισοδύναμος τύπος  $\in 3-SAT$  είναι:

$$\begin{aligned}\phi' = & (\bar{x}_1 \vee \bar{x}_4 \vee z) \wedge (\bar{x}_1 \vee \bar{x}_4 \vee \bar{z}) \\ & \wedge (x_1 \vee \bar{x}_2 \vee \bar{x}_3) \\ & \wedge (\bar{x}_2 \vee \bar{x}_3 \vee y_1) \wedge (x_4 \vee x_1 \vee \bar{y}_1) \\ & \wedge (x_1 \vee u \vee v) \wedge (x_1 \vee u \vee \bar{v}) \wedge (x_1 \vee \bar{u} \vee v) \wedge (x_1 \vee \bar{u} \vee \bar{v})\end{aligned}$$



## NP-πληρότητα

**Απόδειξη.** (συνέχεια) Για κάθε πρόταση  $c$  της  $\phi$ :

- Αν η  $c$  δεν έχει ακριβώς 3 κυριολεκτικά, κατασκευάστε την  $c'$  όπως πριν.
- Αλλιώς,  $c' = c$ .

Το  $\psi$  είναι η σύζευξη όλων των  $c'$  που κατασκευάστηκαν.

Επιστρέφουμε το  $\psi$  ως λύση του προβλήματος 3-SAT.

■ **Ορθότητα:** Η  $\phi$  είναι ικανοποιήσιμη αν και μόνο αν η  $\psi$  είναι ικανοποιήσιμη.



## NP-πληρότητα

### Χρήση Θεωρήματος Cook-Levine

■ Στην διαγνωσιμότητα, χρησιμοποιήσαμε το γεγονός ότι  $A_{TM}$  δεν είναι διαγνώσιμη ως σημείο εκκίνησης για την εύρεση άλλων μη-διαγνώσιμων γλωσσών. [**Ιδέα**: Ανάγουμε την  $A_{TM}$  σε κάποια άλλη γλώσσα]

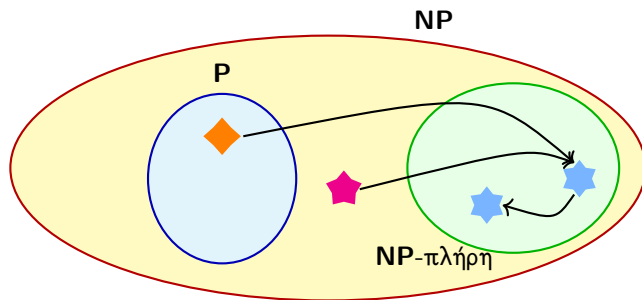
■ Όταν εξετάζουμε την **NP**-πληρότητα, θα χρησιμοποιήσουμε το γεγονός ότι  $3 - SAT \in \mathbf{NP}$ -πλήρες ως σημείο εκκίνησης για την εύρεση άλλων **NP**-πλήρων γλωσσών.

**Ιδέα**: Ανάγουμε την  $3 - SAT$  σε κάποια άλλη γλώσσα.

## NP-πληρότητα

■ Για να αποδείξουμε ότι  $L \in \mathbf{NP}$ -πλήρης, *i)*  $L \in \mathbf{NP}$  και, *ii)* ανάγουμε κάποιο γνωστό  $\mathbf{NP}$ -πλήρες πρόβλημα στην  $L$ .

! Μην ανάγετε την  $L$  σε ένα γνωστό  $\mathbf{NP}$ -πλήρες πρόβλημα.



## NP-πληρότητα

### Μερικά παραδείγματα.

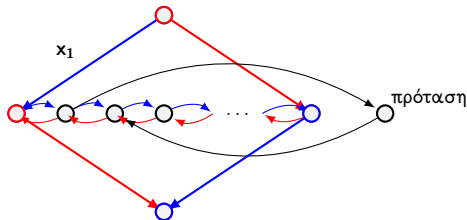
- $INDSET := \left\{ \langle G, k \rangle \mid \begin{array}{l} G \text{ μη-κατευθυνόμενος γράφος με ανεξάρτητο} \\ \text{σύνολο μεγέθους τουλάχιστον } k \end{array} \right\}.$
- $3 - COLOR := \left\{ \langle G \rangle \mid \begin{array}{l} G \text{ μη-κατευθυνόμενος γράφος, υπάρχει} \\ 3 - coloring \text{ των κορυφών του} \end{array} \right\}.$
- $HAMPATH := \left\{ \langle G, s, t \rangle \mid \begin{array}{l} \langle G, s, t \rangle \in PATH \text{ και το μονοπάτι} \\ \text{περνάει από κάθε κορυφή του } G \end{array} \right\}.$

## NP-πληρότητα

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

**Θεώρημα.** Το πρόβλημα *HAMPATH* είναι **NP**-πλήρες.

**Απόδειξη.** Θα δείξουμε ότι  $3\text{-SAT} \leq_p \text{HAMPATH}$ . Θα προσομοιώσουμε τις μεταβλητές και τις προτάσεις με *gadgets*.



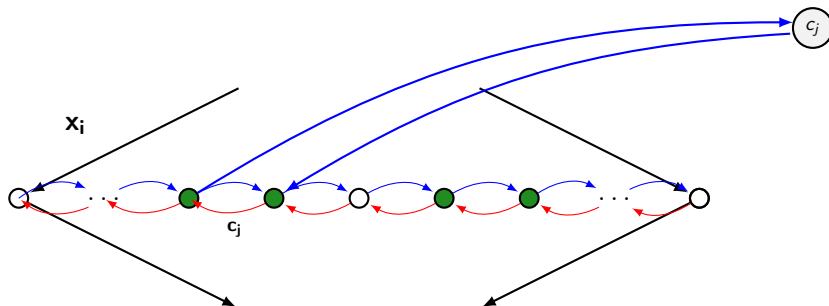
$$\varphi = (x_1 \vee \bar{x}_2 \vee x_3) \wedge \dots \wedge (\bar{x}_k \vee x_l \vee x_m)$$

— μονοπάτια,  $x_1 = \text{Αληθής}$ .

— μονοπάτια,  $x_1 = \text{Ψευδής}$ .

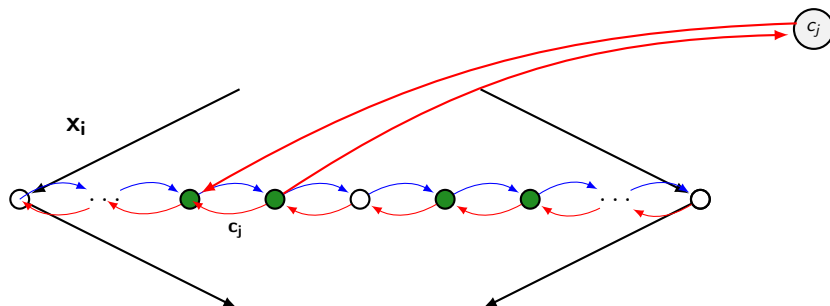
## NP-πληρότητα

**Απόδειξη.** (συνέχεια) Άν η  $x_i$  περιέχεται στην πρόταση  $c_j$  τότε κατασκευάζουμε την εξής παράκαμψη: Αν  $x_i = \text{Αληθής}$ .



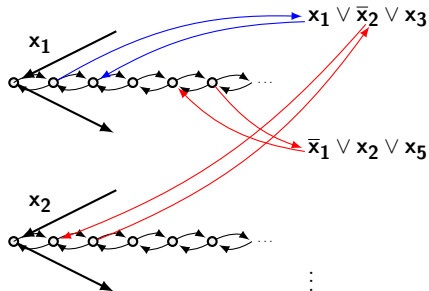
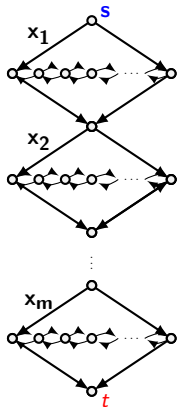
## NP-πληρότητα

**Απόδειξη.** (συνέχεια) Άν η  $x_i$  περιέχεται στην πρόταση  $c_j$  τότε κατασκευάζουμε την εξής παράκαμψη: Αν  $x_i = \text{Ψευδής}$ .



## NP-πληρότητα

**Απόδειξη.** (συνέχεια)  $\varphi = (x_1 \vee \bar{x}_2 \vee x_3) \wedge \dots \wedge (\bar{x}_k \vee x_l \vee x_m)$ . Υποθέτουμε ότι  $c_1 = x_1 \vee \bar{x}_2 \vee x_3$ ,  $c_2 = \bar{x}_1 \vee \bar{x}_2 \vee x_5$ , κτλ. Κατασκευάζουμε το γράφημα  $G$  ως εξής:



## NP-πληρότητα

### Απόδειξη. (συνέχεια)

**Προσοχή.** Στην προηγούμενη διαφάνεια, προφανώς, δεν θα συνδεθούν όλοι οι ρόμβοι με όλους τους κόμβους των  $c$  όπου υπάρχουν μεταβλητές τους. [γιατί;]

Αν ο  $\phi$  είναι ικανοποιήσιμος τύπος, τότε αρκεί ένα από τα literals των  $c_i$  να έχει αποτίμηση Αληθές.

Επομένως, αρκεί να χρησιμοποιήσουμε μια από τις παρακάμψεις [π.χ., στην προηγούμενη διαφάνεια, στο αρκεί είτε η μπλε παράκαμψη είτε η κόκκινη για την  $c_1$ .]

## NP-πληρότητα

**Απόδειξη.** (συνέχεια) Η  $\varphi$  είναι ικανοποιήσιμη αν το γράφημα  $G$  έχει μονοπάτι Hamilton.

[Ευθύ] Για οποιαδήποτε αποτίμηση της που ικανοποιεί την  $\varphi$ , κατασκευάζουμε όλα τα μπλέ και κόκκινα μονοπάτια στο  $G$ , ξεκινώντας από το  $s$  και καταλήγοντας στο  $t$ . Στην συνέχεια προσπελάνουμε τον γράφο ώστε να επισκεφθούμε τους κόμβους που ανταποκρίνονται στην πρόταση  $c_i$ .

[Αντίστροφο] Για οποιοδήποτε μονοπάτι Hamilton από το  $s$  στο  $t$ . Δείχνουμε ότι υπάρχουν μπλέ και κόκκινα μονοπάτια τ.ω. μπορούμε να επισκεφθούμε τους κόμβους που ανταποκρίνονται σε κάθε πρόταση. Αυτό κατασκευάζει μια αποτίμηση αλήθειας για την  $\varphi$ . Επομένως η  $\varphi$  είναι ικανοποιήσιμη.  $\square$

■ Η αναγωγή  $f$  είναι υπολογίσιμη σε πολυωνυμικό χρόνο.

## NP-πληρότητα

**Απόδειξη.** (συνέχεια) Η  $\varphi$  είναι ικανοποιήσιμη αν το γράφημα  $G$  έχει μονοπάτι Hamilton.

[Ευθύ] Για οποιαδήποτε αποτίμηση της που ικανοποιεί την  $\varphi$ , κατασκευάζουμε όλα τα μπλέ και κόκκινα μονοπάτια στο  $G$ , ξεκινώντας από το  $s$  και καταλήγοντας στο  $t$ . Στην συνέχεια προσπελαύνουμε τον γράφο ώστε να επισκεφθούμε τους κόμβους που ανταποκρίνονται στην πρόταση  $c_i$ .

[Αντίστροφο] Για οποιοδήποτε μονοπάτι Hamilton από το  $s$  στο  $t$ . Δείχνουμε ότι υπάρχουν μπλέ και κόκκινα μονοπάτια τ.ω. μπορούμε να επισκεφθούμε τους κόμβους που ανταποκρίνονται σε κάθε πρόταση. Αυτό κατασκευάζει μια αποτίμηση αλήθειας για την  $\varphi$ . Επομένως η  $\varphi$  είναι ικανοποιήσιμη.  $\square$

■ Η αναγωγή  $f$  είναι υπολογίσιμη σε πολυωνυμικό χρόνο.

□ Η κατασκευή που χρησιμοποιήσαμε για την απόδειξη δουλεύει και για μη-κατευθυνόμενο γράφημα;