# Αλγόριθμοι Προσέγγισης για NP-Δύσκολα Προβλήματα

Διδάσκοντες: **Ε. Ζάχος, Α. Παγουρτζής Δ. Φωτάκης**
Επιμέλεια διαφανειών: **Δ. Φωτάκης**

Σχολή Ηλεκτρολόγων Μηχανικών
και Μηχανικών Υπολογιστών

Εθνικό Μετσόβιο Πολυτεχνείο

## Υπολογιστική Πολυπλοκότητα

- Γιατί μερικά προβλήματα είναι δύσκολο να λυθούν από υπολογιστικές μηχανές.
- Αντικείμενο:
  - Επιλύσιμα προβλήματα: υπολογιστικοί πόροι.
    - Ευεπίλυτα: spanning tree, shortest paths, max flow, min-cost flow, linear programming, …
    - Δυσεπίλυτα: TSP, vertex cover, knapsack, scheduling, …
  - Επίδραση υπολογιστικού μοντέλου.



## Προβλήματα και Αλγόριθμοι

- Αλγόριθμος: λεπτομερής περιγραφή μεθόδου επίλυσης προβλήματος από υπολογιστική μηχανή.
- Πρόβλημα: άπειρο σύνολο στιγμιοτύπων.
  - Αποτελεί αντικείμενο μελέτης.
- Στιγμιότυπο: μαθηματικό αντικείμενο για το οποίο κάνουμε ερώτηση και περιμένουμε απάντηση.
- Προβλήματα:
  - Βελτιστοποίησης: λύση με βέλτιστη αντικειμενική τιμή.
  - Απόφασης: απάντηση ΝΑΙ ή ΌΧΙ.



## Προβλήματα Βελτιστοποίησης

- Πρόβλημα βελτιστοποίησης $\Pi$:
  - Σύνολο στιγμιότυπων $\Sigma_\Pi$
  - Σύνολο αποδεκτών λύσεων: $\forall \sigma \in \Sigma_\Pi\,,\ \Lambda_\Pi(\sigma)$
  - Αντικειμενική συνάρτηση: $\forall \sigma \in \Sigma_\Pi\,,\ f_\sigma : \Lambda_\Pi(\sigma) \mapsto \mathbb{R}$
- Δεδομένου στιγμιότυπου σ, ζητείται $\lambda^*_\sigma \in \Lambda_\Pi(\sigma)$:

$\forall \lambda \in \Lambda_\Pi(\sigma), \quad f_\sigma(\lambda^*_\sigma) \geq f_\sigma(\lambda)$ **πρόβλημα μεγιστοποίησης**

$\forall \lambda \in \Lambda_\Pi(\sigma), \quad f_\sigma(\lambda^*_\sigma) \leq f_\sigma(\lambda)$ **πρόβλημα ελαχιστοποίησης**

$\lambda^*_\sigma$ **βέλτιστη λύση και** $f_\sigma(\lambda^*_\sigma)$ **βέλτιστη αντικειμενική τιμή**

- Συνδυαστικής βελτιστοποίησης: πεπερασμένο σύνολο αποδεκτών λύσεων που περιλαμβάνει βέλτιστη.

## Προβλήματα Απόφασης

- Πρόβλημα απόφασης $\Pi$ :
  - Σύνολο στιγμιότυπων $\Sigma_\Pi$
  - Σύνολο (αποδεκτών) λύσεων: $\forall \sigma \in \Sigma_\Pi$ , $\Lambda_\Pi(\sigma)$
  - Δεδομένου $\sigma \in \Sigma_\Pi$ , $\Lambda_\Pi(\sigma) \neq \emptyset$;
- Επιδέχεται μόνο δύο απαντήσεων: ΝΑΙ ή ΌΧΙ.
- Παραδείγματα προβλημάτων απόφασης:
  - Συνεκτικότητα, κύκλος Euler, κύκλος Hamilton.
  - Ικανοποιησιμότητα λογικών προτάσεων.
  - Κάθε πρόβλημα βελτιστοποίησης: έχει λύση με αντικειμενική τιμή καλύτερη από Β;



## Αποδοτική Επίλυση: Κλάση P

- Μέγεθος στιγμιότυπου *n* : αριθμός bits για αναπαράστασή του.
- Αλγόριθμος πολυωνυμικού χρόνου λύνει κάθε στιγμιότυπο σε χρόνο $O(n^d)$, *d* σταθερά.
- **Κλάση P** : προβλήματα απόφασης που επιλύονται από αλγόριθμους πολυωνυμικού χρόνου.
  - Shortest paths, MST, max flow, min cut, min-cost flow, maximum matching, linear programming, …
- **Αξίωμα Cook-Karp** : κλάση ευεπίλυτων προβλημάτων ταυτίζεται με κλάση P.



## Αποδοτική Επίλυση: Κλάση P

- Υπέρ αξιώματος Cook-Karp:
  - Κλάση P ανεξάρτητη υπολογιστικού μοντέλου.
  - Συνήθως πολυώνυμα μικρού βαθμού (π.χ. $n$, $n^2$, $n^3$).
  - Διπλασιασμός υπολογιστικής ισχύος: σημαντική αύξηση στο μέγεθος στιγμιότυπων που επιλύουμε.
- Κριτική στο αξίωμα Cook-Karp:
  - Ακραίες περιπτώσεις: πρακτικό το $n^{100}$ αλλά όχι το $2^{n/100}$ !
  - Γραμμικός Προγραμματισμός:
    - Simplex εκθετικού χρόνου στη χειρότερη περίπτωση αλλά πολύ γρήγορος στην πράξη.
    - Ελλειψοειδές πολυωνυμικού χρόνου αλλά όχι πρακτικός.



## «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. Καλό αλλά δύσκολο!
  - **Κανένας** δεν μπορεί να βρει αποδοτικό αλγόριθμο (και όλοι πιστεύουν ότι δεν υπάρχει).
- Θεωρία **NP-πληρότητας.**
  - NP-πλήρη: κλάση εξαιρετικά σημαντικών προβλημάτων που είτε όλα επιλύονται σε πολυωνυμικό χρόνο είτε κανένα.

# Η Κλάση NP

- …περιλαμβάνει προβλήματα απόφασης:
  - Για κάθε ΝΑΙ-στιγμιότυπο, υπάρχει «πιστοποιητικό» **εύκολο να ελεγχθεί** (ελέγχεται σε πολυωνυμικό χρόνο).
    - Συνοπτικό πιστοποιητικό (succinct certificate).
  - Πιστοποιητικό μπορεί να είναι **δύσκολο να υπολογισθεί.** Αν δοθεί, ελέγχεται εύκολα!
  - Δεν απαιτείται συνοπτικό πιστοποιητικό για ΌΧΙ-στιγμιότυπα.
- Κλάση co-NP περιλαμβάνει προβλήματα απόφασης με συνοπτικό πιστοποιητικό για ΌΧΙ-στιγμιότυπα.
- Προβλήματα P ανήκουν NP
- Προβλήματα P ανήκουν co-NP

$$\Rightarrow \mathcal{P} \subseteq \mathcal{NP} \cap \text{co-}\mathcal{NP}$$



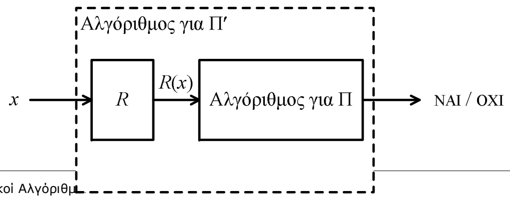
# Η Κλάση NP

- Για σημαντικότερα προβλήματα βελτιστοποίησης, αντίστοιχα προβλήματα απόφασης ανήκουν στο NP.
  - Συντομότερο s – t μονοπάτι μήκους ≤ B? Πιστοποιητικό μονοπάτι μήκους ≤ B.
  - Ελάχιστο Επικαλύπτον Δέντρο βάρους ≤ B? Πιστοποιητικό επικαλύπτον δέντρο βάρους ≤ B.
  - Υπάρχει κύκλος Hamilton?
  - Περιοδεία Περιοδεύοντος Πωλητή μήκους ≤ B? Πιστοποιητικό περιοδεία μήκους ≤ B.
  - Ελάχιστο κάλυμμα κορυφών με #κορυφών ≤ B? Πιστοποιητικό κάλυμμα κορυφών με ≤ B κορυφές.



# NP-Πληρότητα

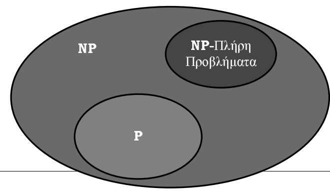
- **NP-δύσκολο** πρόβλημα Π αν κάθε πρόβλημα Π′ στο NP ανάγεται σε πολυωνυμικό χρόνο στο Π.
  - Αναγωγή: αλγόριθμος $R : \Sigma_{\Pi'} \mapsto \Sigma_{\Pi}$ πολυωνυμικού χρόνου με ιδιότητα *x* ΝΑΙ-στιγμ. ανν *R*(*x*) ΝΑΙ-στιγμ.
  - Αν Π λύνεται σε πολυωνυμικό χρόνο, Π′ επίσης!
- **NP-πλήρες** πρόβλημα κάθε NP-δύσκολο που ανήκει NP.






# P και NP

- Εξ′ ορισμού $\mathcal{P} \subseteq \mathcal{NP}$
- Ερώτημα: $\mathcal{P} = \mathcal{NP}$ ή $\mathcal{P} \neq \mathcal{NP}$
- Αν $\mathcal{P} = \mathcal{NP}$, πολλά σημαντικά προβλήματα **ευεπίλυτα!**
- Αν $\mathcal{P} \neq \mathcal{NP}$ (όπως όλοι πιστεύουν), κάποια προβλήματα στο NP δεν λύνονται σε πολυωνυμικό χρόνο!
  - Εξ′ ορισμού, τα NP-πλήρη πρέπει να είναι σε αυτά.



## Αντιμετώπιση NP-Δυσκολίας

- Αν $\mathcal{P} \neq \mathcal{NP}$, όχι αλγόριθμος που για όλα τα στιγμιότυπα υπολογίζει βέλτιστη λύση σε πολυωνυμικό χρόνο.
- Ευρετικές τεχνικές: συχνά γρήγορα βέλτιστη λύση αλλά και δύσκολα στιγμιότυπα (αργά ή / και όχι βέλτιστη λύση).
  - Τοπική αναζήτηση.
  - Simulated annealing.
  - Γενετικοί αλγόριθμοι.
  - Branch-and-Bound, Branch-and-Cut.
  - …



## Αντιμετώπιση NP-Δυσκολίας

- Ανάλυση μέσης περίπτωσης / πιθανοτική ανάλυση.
  - Γρήγοροι σε στιγμιότυπα που εμφανίζονται συχνότερα (αργοί μόνο για στιγμιότυπα με μικρή πιθανότητα).
  - Διαφορά από ευρετικές τεχνικές: θεωρητική ανάλυση.
    - Γνωρίζουμε πιθανότητα και πότε καλή / κακή απόδοση.
- «Εύκολες» περιπτώσεις.
- Αλγόριθμοι προσέγγισης [Johnson, Sahni and Gonzalez, …, 70's]
  - Αλγόριθμοι πολυωνυμικού χρόνου (χ.π.).
  - Όχι (πάντα) βέλτιστη λύση.
  - Ανάλυση χειρότερης περίπτωσης ως προς ποιότητα λύσης.



## Αλγόριθμοι Προσέγγισης

- Απόδοση χειρότερης περίπτωσης γνωστών ευρετικών αλγόριθμων (αρχικά κυρίως άπληστων).
- Σχεδιασμός poly-time αλγόριθμων που συμπεριφέρονται **αποδεδειγμένα καλά** για κάθε στιγμιότυπο.
- **Λόγος προσέγγισης**
  - Αλγόριθμου A για πρόβλημα Π (πάντα ≥ 1):

$$\text{Μεγιστοποίηση:}\quad \gamma_\Pi(A) = \max_{\sigma \in S_\Pi} \frac{f_\sigma(\lambda^*(\sigma))}{f_\sigma(\lambda_A(\sigma))}$$

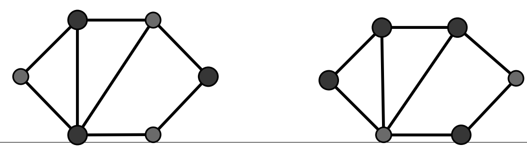
$$\text{Ελαχιστοποίηση:}\quad \gamma_\Pi(A) = \max_{\sigma \in S_\Pi} \frac{f_\sigma(\lambda_A(\sigma))}{f_\sigma(\lambda^*(\sigma))}$$

  - Προβλήματος Π: $\gamma_\Pi = \min_{A \text{ poly-time alg}} \{\gamma_\Pi(A)\}$



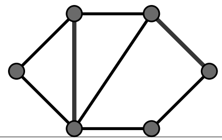
## Ελάχιστο Κάλυμμα Κορυφών

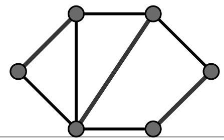
- Είσοδος: γράφημα $G(V, E)$
- Εφικτή λύση: υποσύνολο κορυφών $C \subseteq V$: κάθε ακμή τουλάχιστον ένα άκρο στο $C$
  - $C$ είναι κάλυμμα κορυφών (vertex cover).
- Στόχος: κάλυμμα κορυφών με ελάχιστο #κορυφών.
- **NP-δύσκολο** πρόβλημα.



## Ταίριασμα

- Ταίριασμα : υποσύνολο ακμών $M \subseteq E$ χωρίς κοινά άκρα.
- Για κάθε ταίριασμα M, τουλάχιστον ένα από τα άκρα ακμών M ανήκει σε κάθε κάλυμμα κορυφών:
  - $\forall$ταίριασμα M, ελάχιστο κάλυμμα κορυφών $|C^*| \geq |M|$

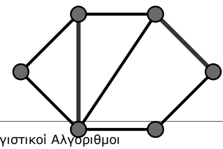


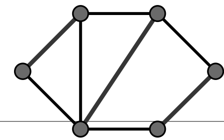




## Μεγιστοτικό Ταίριασμα

- Μεγιστοτικό ταίριασμα : ταίριασμα που αν προσθέσουμε ακμή παύει να είναι ταίριασμα.
  - Μεγιστοτικό ταίριασμα M : κάθε ακμή εκτός M έχει κοινό άκρο με ακμή του M.
  - Άκρα ακμών μεγιστοτικού ταιριάσματος M συγκροτούν κάλυμμα κορυφών C.
  - $|C| = 2|M| \leq 2|C^*|$







## Αλγόριθμος ΜΜ

- Υπολογισμός μεγιστοτικού ταιριάσματος M.
  - Προσθήκη ακμών ενόσω υπάρχουν ακμές που προσθήκη τους δίνει ταίριασμα.
- κάλυμμα κορυφών C : όλα τα άκρα ακμών M.
- Πολυωνυμικός χρόνος.
- **Ορθότητα** : ιδιότητα μεγιστοτικού ταιριάσματος.
- **Λόγος προσέγγισης = 2.**
  - $|C| = 2|M| \leq 2|C^*|$ (πάνω φράγμα).
  - Παραδείγματα όπου κόστος ΜΜ διπλάσιο βέλτιστου.

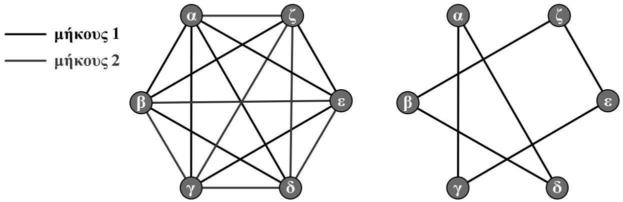


## Βασική Ιδέα (ελαχιστοποίηση)

- Ξεκινώ από κάτω φράγμα στο κόστος βέλτιστης λύσης.
  - Για κάθε ταίριασμα M, $|C^*| \geq |M|$.
  - Κάτω φράγμα εκφράζεται σαν συνάρτηση κάποιων άλλων παραμέτρων του στιγμιότυπου εισόδου.
  - Πολλές φορές κάτω φράγμα προκύπτει από **δυϊκότητα.**
- (Πολυωνυμικός) αλγόριθμος: εφικτή λύση με κόστος = συνάρτηση των παραμέτρων στο κάτω φράγμα.
  - Μεγιστοτικό ταίριασμα M: κάλυμμα κορυφών C, $|C| = 2\,|M|$.
- Σύγκριση δίνει άνω φράγμα στο λόγο προσέγγισης.
  - $|C| = 2\,|M| \leq 2\,|C^*|$.

# Περιοδεύων Πωλητής

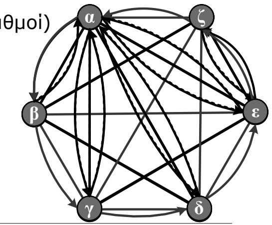
- Είσοδος: *n* σημεία με (συμμετρικές) αποστάσεις τους.
  - Αποστάσεις ικανοποιούν τριγωνική ανισότητα (metric space).
- Αποδεκτές λύσεις: περιοδείες (μεταθέσεις) *n* σημείων.
- Στόχος: **περιοδεία ελάχιστου συνολικού μήκους.**






# Κάτω φράγμα

- Ελάχιστο Επικαλύπτον Δέντρο (ΕΕΔ).
  - Κάθε περιοδεία έχει μήκος ≥ βάρος ΕΕΔ.
  - Περιοδεία – ακμή: επικαλύπτον δέντρο.
- Αλγόριθμος:
  - $T^*$ ΕΕΔ βάρους $w(T^*)$
  - «Διπλασιασμός» ακμών $T^*$ (άρτιοι βαθμοί)
  - Κύκλος Euler στο διπλασιασμένο $T^*$
  - Αποφυγή διπλών εμφανίσεων «κόβοντας» δρόμο.
- Μήκος ≤ 2 $w(T^*)$ λόγω τριγωνικής ανισότητας
- Λόγος προσέγγισης ≤ 2 (tight).






# Καλύτερος Αλγόριθμος

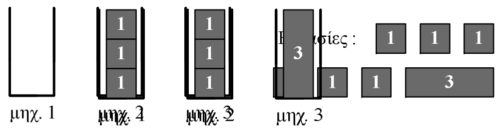
- Αλγόριθμος Χριστοφίδη (1976)
  - Ελάχιστο Επικαλύπτον Δέντρο.
  - Ταίριασμα ελάχιστου βάρους μεταξύ κορυφών ΕΕΔ με περιττό βαθμό.
  - Κύκλος Euler.
  - Περιοδεία μήκους ≤ βάρος ΕΕΔ + βάρος ταιριάσματος.
  - Λόγος προσέγγισης **= 3/2.**
- Μετά 30 χρόνια, καλύτερος γνωστός αλγόριθμος.
- Υπάρχουν καλύτεροι αλγόριθμοι για ειδικές περιπτώσεις (π.χ. TSP(1, 2), planar TSP, …).



# Δρομολόγηση Εργασιών

- Είσοδος:
  - m ίδιες μηχανές (σύνολο M).
  - n εργασίες μεγέθους $w_1, w_2, \ldots, w_n$ (σύνολο J).
- Αποδεκτές λύσεις: κάθε δρομολόγηση φ
- Στόχος: **ελαχιστοποίηση μέγιστου φορτίου μηχανής:**

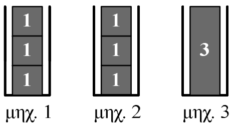
$$\forall \phi: J \mapsto M,\ S(\phi) = \max_{i \in M}\left\{\sum_{j:\phi(j)=i} w_j\right\}$$
$$\phi^* = \arg\min_{\phi: J \mapsto M}\{S(\phi)\}$$



# Κάτω φράγμα

- $S^* \geq \sum_{j=1}^{n} w_j / m = W_{tot}/m$
- $S^* \geq \max_{1 \leq j \leq n}\{w_j\} = w_{max}$

$$\left.\begin{array}{l} S^* \geq \sum_{j=1}^{n} w_j / m = W_{tot}/m \\ S^* \geq \max_{1 \leq j \leq n}\{w_j\} = w_{max} \end{array}\right\} \Rightarrow S^* \geq \max\{W_{tot}/m, w_{max}\}$$






# Αλγόριθμος Graham (1966)

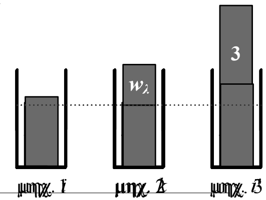
- Εργασίες μία – μία με σειρά που δίνονται **(online)**.
- Νέα εργασία σε μηχανή με ελάχιστο φορτίο **(greedy)**.
- Άνω φράγμα στο μέγιστο φορτίο:
  - Φορτίο μηχανής i πριν δρομολογηθεί εργασία j : $S_j^{(i)}$
  - k μηχανή με μεγαλύτερο φορτίο
  - $w_\lambda$ τελευταία εργασία στην μηχανή k

$$\forall i \in M\ ,\ S_\lambda^{(k)} \leq S_\lambda^{(i)}$$

$$\Rightarrow m S_\lambda^{(k)} \leq \sum_{i=1}^{m} S_\lambda^{(i)} \leq W_{tot} - w_\lambda$$

$$\Rightarrow S_\lambda^{(k)} \leq (W_{tot} - w_\lambda)/m$$

$$S_\lambda^{(k)} + w_\lambda \leq W_{tot}/m + (1 - \tfrac{1}{m}) w_\lambda \leq (2 - \tfrac{1}{m}) S^*$$






# Κάλυμμα Συνόλων (Set Cover)

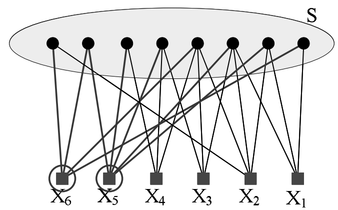
- Σύνολο στοιχείων $S = \{1, \ldots, n\}$
- Μη-κενά υποσύνολα του $S$ : $X_1, \ldots, X_m$ , $\bigcup_{i=1}^{m} X_i = S$
- Κόστος υποσυνόλων: $w_1, \ldots, w_m$
- Ζητούμενο: κάλυμμα *S* με ελάχιστο κόστος.
  - Ελάχιστου κόστους συλλογή υποσυνόλων $\mathcal{C}$ : $\bigcup_{i \in \mathcal{C}} X_i = S$

$$\begin{array}{lll} \min \sum_{j=1}^{m} x_j w_j & & \\ \text{s.t. } \sum_{j: i \in X_j} x_j \geq 1 & & \forall i \in S \\ \quad x_j \in \{0,1\} & & \forall j \in [m] \end{array}$$

- **NP**-δύσκολο πρόβλημα.
- **Απληστία**: καλύτερος προσεγγιστικός αλγόριθμος.



# Παράδειγμα




- S = {1, 2, 3, 4, 5, 6, 7, 8 }
- $X_1$ = {1, 2, 3}, $X_2$ = {2, 3, 4, 8}, $X_3$ = {3, 4, 5}
  $X_4$ = {4, 5, 6}, $X_5$ = {2, 3, 5, 6, 7}, $X_6$ = {1, 4, 7, 8}
- Βέλτιστη λύση: $X_5$ , $X_6$

# Άπληστος Αλγόριθμος

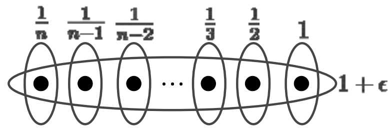
- Σύνολο U ακάλυπτων στοιχείων (αρχικά U = S).
- Επιλογή υποσυνόλου που ελαχιστοποιεί **κόστος ανά ακάλυπτο στοιχείο** που καλύπτει: $w_i / |X_i \cap U|$
- Ενημέρωση U και συνέχεια ενόσω U δεν είναι κενό.

$$
\begin{array}{l}
\mathbf{greedySetCover}(S, (X_1, w_1), \ldots, (X_m, w_m)) \\
\quad U \leftarrow S; \quad C \leftarrow \emptyset; \\
\quad \text{while } U \neq \emptyset \text{ do} \\
\quad\quad j \leftarrow \arg\min_{i \in [m]} \{w_i / |X_i \cap U|\}; \\
\quad\quad C \leftarrow C \cup \{j\}; \quad U \leftarrow U \setminus X_j; \\
\quad \text{return}(C, \sum_{i \in C} w_i);
\end{array}
$$



# Αντιπαράδειγμα

- Δεν πλησιάζει τη βέλτιστη λύση!
  - Βέλτιστη λύση έχει κόστος 1+ε.
  - Κόστος άπληστου αλγόριθμου: $H_n = \sum_{i=1}^{n} 1/n \approx \ln n$
  - Παράδειγμα: χειρότερη περίπτωση άπληστου αλγόριθμου.






# Ανάλυση

- Έστω OPT κόστος βέλτιστης λύσης.
- Αρχή *i*-οστής επανάλ.: ακάλυπτα στοιχεία $n_i \leq n - i + 1$ (κάθε προηγούμενη επανάληψη καλύπτει ≥ 1 στοιχείο).
- Βέλτιστη καλύπτει στοιχεία με μέσο κόστος $\mathrm{OPT}/n_i$
- Άπληστη επιλογή έχει κόστος / στοιχείο $\leq \mathrm{OPT}/n_i$
- Αθροίζοντας για ≤ *n* επαναλήψεις, κόστος άπληστου αλγ.

$$\leq \mathrm{OPT} \sum_{i=n}^{1} 1/i = \mathrm{OPT}\, H_n \approx \mathrm{OPT} \ln n$$

- **Λόγος προσέγγισης** $\approx \ln n$
- Αποδεικνύεται ότι **δεν** υπάρχει αλγόριθμος πολυωνυμικού χρόνου με καλύτερο λόγο προσέγγισης.



# Μη-Προσεγγισιμότητα

- Προβλήματα στο NP που προσέγγιση είναι NP-δύσκολη!
  - Περιοδεύων Πωλητής χωρίς τριγωνική ανισότητα, μέγιστη κλίκα / σύνολο ανεξαρτησίας, χρωματικός αριθμός, …
- Πρόβλημα Περιοδεύοντος Πωλητή χωρίς Τριγωνική Ανισότητα (ΠΠΠ):
  - n σημεία και συμμετρικές αποστάσεις (αλλά όχι metric).
  - Ζητούμενο: περιοδεία ελάχιστου συνολικού μήκους.
- Για κάθε γ, γ-προσέγγιση ΠΠΠ είναι **NP-δύσκολη** [Sahni και Gonzalez, 1976].
  - Κάθε γ-προσεγγιστικός αλγόριθμος για ΠΠΠ λύνει πρόβλημα κύκλου Hamilton!

# Απόδειξη

- Γράφημα G(V, E): υπάρχει κύκλος Hamilton στο G;
- Αναγωγή σε γ-προσέγγιση ΠΠΠ (για οποιοδήποτε γ > 1):
  - Κορυφές ↔ σημεία.
  - Αποστάσεις: $d(u,v) = \begin{cases} 1 & \text{αν } \{u,v\} \in E \\ \gamma|V| & \text{αν } \{u,v\} \notin E \end{cases}$
  - Κύκλος Hamilton στο G ⇒ περιοδεία μήκους |V|
  - Όχι κύκλος Hamilton στο G ⇒ περιοδεία μήκους ≥ γ|V| + |V| – 1 > γ|V|
- γ-προσεγγιστικός αλγόριθμος για ΠΠΠ:
  - Κύκλος Hamilton στο G ⇔ περιοδεία μήκους ≤ γ|V|
  - Αποφασίζει (σωστά) αν υπάρχει κύκλος Hamilton στο G.



# Επισκόπηση Περιοχής

- Σχήματα προσέγγισης: λόγος (1+ε), για κάθε ε > 0.
  - Σακίδιο, δρομολόγηση εργασιών, γεωμετρικά προβλήματα, …
  - Δυναμικός προγραμματισμός και διακριτοποίηση.
- Σταθερός λόγος προσέγγισης.
  - MAX-SNP-δυσκολία: NP-δύσκολο να υπάρξει σχήμα
    - PCP Θεώρημα: NP = PCP(log *n* , 1).
  - Προβλήματα σε μετρικούς χώρους: ΠΠΠ-ΤΑ, facility location, δέντρο Steiner, …
  - Προβλήματα σε γραφήματα: κάλυμμα κορυφών, μέγιστη τομή, feedback vertex set, …
  - Προβλήματα ικανοποιησιμότητας: Max-k-SAT.



# Επισκόπηση Περιοχής

- Τεχνικές για σταθερό λόγο προσέγγισης:
  - Τοπική αναζήτηση – μέθοδος απληστίας.
  - Primal-dual μέθοδος.
  - Dual-fitting μέθοδος.
  - Relaxation του Ακέραιο Προγράμματος σε Γραμμικό Πρόγραμμα, επίλυση, και τυχαίο στρογγύλεμα μη-ακέραιων λύσεων.



# Επισκόπηση Περιοχής

- Λογαριθμικός λόγος προσέγγισης.
  - Ελάχιστο κάλυμμα συνόλων
    - Άπληστος αλγόριθμος (dual-fitting) καλύτερος δυνατός.
  - Αραιότερη τομή, γραμμικές διατάξεις, …
  - Εμβάπτιση μετρικών χώρων σε απλούστερους χώρους όπου προβλήματα λύνονται ευκολότερα.
- Πολυωνυμικός λόγος προσέγγισης.
  - Μέγιστη κλίκα / σύνολο ανεξαρτησίας, χρωματισμός γραφημάτων, …
  - PCP Θεώρημα: για κάθε ε > 0, προσέγγιση μέγιστης κλίκας σε λόγο $|V|^{1-\varepsilon}$ είναι NP-δύσκολο πρόβλημα!