# FPTAS για το Subset Sum

## Συνδυαστική Βελτιστοποίηση

Βασίλης Ζησιμόπουλος

Θεωρητική Πληροφορική
Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών

# Πρόβλημα Αθροίσματος Υποσυνόλων (Subset Sum)

**Ορισμός**

Δίνεται ένα σύνολο $S \subseteq \mathbb{N}$ και μια τιμή $t \in \mathbb{N}$. Υπάρχει σύνολο $S' \subseteq S$ με άθροισμα στοιχείων ίσο με $t$.

# Πρόβλημα Αθροίσματος Υποσυνόλων (Subset Sum)

## Ορισμός

Δίνεται ένα σύνολο $S \subseteq \mathbb{N}$ και μια τιμή $t \in \mathbb{N}$. Υπάρχει σύνολο $S' \subseteq S$ με άθροισμα στοιχείων ίσο με $t$.

Παράδειγμα: $S = \{1, 4, 16, 64, 256, 1040, 1041, 1093, 1284, 1344\}$
και $t = 3754 \Rightarrow S' = \{1, 16, 64, 256, 1040, 1093, 1284\}$

# Πρόβλημα Αθροίσματος Υποσυνόλων (Subset Sum)

## Ορισμός

Δίνεται ένα σύνολο $S \subseteq \mathbb{N}$ και μια τιμή $t \in \mathbb{N}$. Υπάρχει σύνολο $S^{'} \subseteq S$ με άθροισμα στοιχείων ίσο με $t$.

Παράδειγμα: $S = \{1, 4, 16, 64, 256, 1040, 1041, 1093, 1284, 1344\}$ και $t = 3754 \Rightarrow S^{'} = \{1, 16, 64, 256, 1040, 1093, 1284\}$

Το πρόβλημα αυτό είναι NP-complete.

# Πρόβλημα Αθροίσματος Υποσυνόλων (Subset Sum)

## Ορισμός

Δίνεται ένα σύνολο $S \subseteq \mathbb{N}$ και μια τιμή $t \in \mathbb{N}$. Υπάρχει σύνολο $S' \subseteq S$ με άθροισμα στοιχείων ίσο με $t$.

Παράδειγμα: $S = \{1, 4, 16, 64, 256, 1040, 1041, 1093, 1284, 1344\}$ και $t = 3754 \Rightarrow S' = \{1, 16, 64, 256, 1040, 1093, 1284\}$

Το πρόβλημα αυτό είναι NP-complete.

Subset Sum (optimization problem): Να βρεθεί σύνολο $S' \subseteq S$ με μέγιστο άθροισμα στοιχείων μικρότερο ίσο με $t$.

# Merging two sorted Lists

- $L$: λίστα από ακέραιους, $x$ ακέραιος
- $L + x = \{l_i + x | l_i \in L\}$
- Π.χ. $L = \{1, 2, 3, 5, 9\}$ και $x = 2$ τότε $L + x = \{3, 4, 5, 7, 11\}$
- Εάν $L$ και $L^{'}$ ταξινομημένες, τότε σε χρόνο $O(|L| + |L^{'}|)$ μπορούμε να τις συγχωνεύσουμε παίρνοντας την ένωσή τους $L^{''}$ ταξινομημένη.

# Αλγόριθμος για το Subset Sum

**Algorithm 1** Subset Sum $(S, t)$

$n = |S|$
$L_0 = \{0\}$
**for** $i = 1$ **to** n **do**
  $L_i = MergeLists(L_{i-1}, L_{i-1} + x_i)$
  $L_i = L_i - \{x | x > t\}$
**επέστρεψε** το μεγαλύτερο στοιχείο της $L_n$

$$S_i = \{x_1, \dots, x_i\}$$
$$S \subseteq S_i$$
$$P_i = \{x | x = \textstyle\sum_{s \in S} s\}$$

# Παράδειγμα

- $L = \{1, 4, 5\}$
- $S_0 = \emptyset, P_0 = \{0\}$
- $S_1 = \{1\}$, $\mathcal{P}(S_1) = \{\emptyset, \{1\}\}, P_1 = \{0, 1\}$
- $S_2 = \{1, 4\}$, $\mathcal{P}(S_2) = \{\emptyset, \{1\}, \{4\}, \{1, 4\}\}, P_2 = \{0, 1, 4, 5\}$
- $S_3 = \{1, 4, 5\}$, $\mathcal{P}(S_3) = \{\emptyset, \{1\}, \{4\}, \{5\}, \{1, 4\}, \{1, 5\}, \{4, 5\}, \{1, 4, 5\}\}, P_3 = \{0, 1, 4, 5, 6, 9, 10\}$
- $P_i = P_{i-1} \cup (P_{i-1} + x_i)$ (άσκηση)
- $L_i$ (ordered $\leq t$)

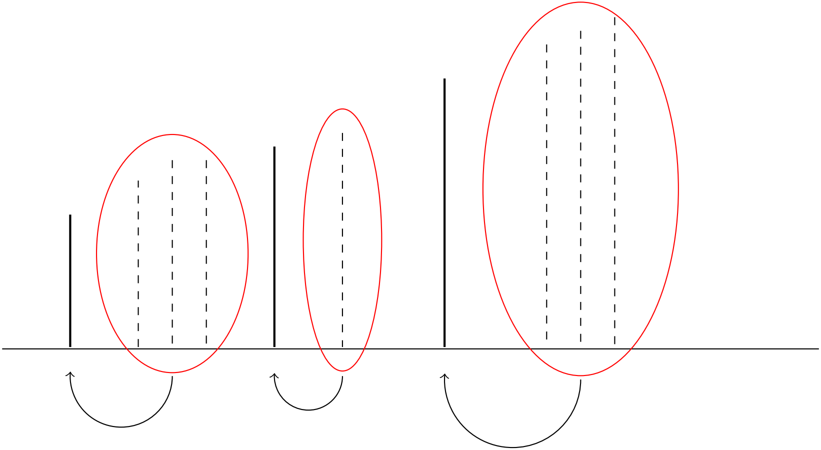
# Πολυπλοκότητα

- Εκθετική: Το μήκος της $L_i$ αυξάνει εκθετικά ως προς το $i$ ($|L_i| = 2^i$).
- Πολυωνυμική: αν το $t$ είναι πολυωνυμικό ως προς $|S|$.
- Πολυωνυμική: αν $\forall x \in S$, το $x$ φράσσεται από ένα πολυώνυμο του $|S|$.

# Ένα FPTAS για το Subset Sum

- Trimming μιας λίστας $L$.
- Παράμετρο $\delta \in (0,1)$
- $L^{'} = L - \{y\}$ (trimming)
- Διαγράφουμε $y \in L$ αν $\exists z \in L^{'} \mid \dfrac{y-z}{y} \leq \delta$ (δηλαδή $(1-\delta)y \leq z \leq y$)

# Trimming



Σχήμα: Διαγράφονται τα στοιχεία που παριστάνονται με διακεκομένη γραμμή, αφού εκπροσωπούνται από αυτά με συνεχή γραμμή.

# Εκπρόσωποι (Παράδειγμα)

- $\delta = 0.1$

# Εκπρόσωποι (Παράδειγμα)

- $\delta = 0.1$
- $L = \{10, 11, 12, 15, 20, 21, 22, 23, 24, 29\}$

# Εκπρόσωποι (Παράδειγμα)

- $\delta = 0.1$
- $L = \{10, 11, 12, 15, 20, 21, 22, 23, 24, 29\}$
- Το $11$ μπορεί να εκπροσωπηθεί από το $10$, το $21$ και το $22$ από το $20$, και το $24$ από το $23$

# Εκπρόσωποι (Παράδειγμα)

- $\delta = 0.1$
- $L = \{10, 11, 12, 15, 20, 21, 22, 23, 24, 29\}$
- Το $11$ μπορεί να εκπροσωπηθεί από το $10$, το $21$ και το $22$ από το $20$, και το $24$ από το $23$
- $L^{'} = \{10, 12, 15, 20, 23, 29\}$

## Εκπρόσωποι (Παράδειγμα)

- $\delta = 0.1$
- $L = \{10, 11, 12, 15, 20, 21, 22, 23, 24, 29\}$
- Το $11$ μπορεί να εκπροσωπηθεί από το $10$, το $21$ και το $22$ από το $20$, και το $24$ από το $23$
- $L^{'} = \{10, 12, 15, 20, 23, 29\}$
- $L^{'} \subseteq L$ και η $L^{'}$ υπολογίζεται σε πολυωνυμικό χρόνο.

# Αλγόριθμος Trimming

**Algorithm 2** Trim$(L, \delta)$

$m \leftarrow |L|$
$L^{'} \leftarrow \{y_1\}$
last $\leftarrow \{y_1\}$
**for** $i = 2$ **to** $m$ **do**
  **if** last $< (1 - \delta) y_i$ **then**
    προσάρτησε την $y_i$ στο τέλος της $L^{'}$
  last $\leftarrow \{y_i\}$
**επέστρεψε** την $L^{'}$

Η $L$ είναι ταξινομημένη, άρα η $L^{'}$ που προκύπτει σε χρόνο $\Theta(m)$ είναι επίσης ταξινομημένη.

# Προσεγγιστικό Άθροισμα Υποσυνόλων

**Algorithm 3** Προσεγγιστικό Σχήμα για Άθροισμα Υποσυνόλων$(S, t, \varepsilon)$

$n \leftarrow |S|$
$L_0 \leftarrow \{0\}$
**for** $i = 1$ **to** $n$ **do**
    $L_i = MergeLists(L_{i-1}, L_{i-1} + x_i)$
    $L_i = Trim(L_i, \frac{\varepsilon}{i})$
    $L_i = L_i - \bigcup_{1 \leq j \leq i} \{x_j \in L_i \mid x_j > t\}$
**επέστρεψε** το μεγαλύτερο στοιχείο $(z^*)$ της $L_n$

# Παράδειγμα

- $L = \{104, 102, 201, 101\}, t = 308, \varepsilon = 0.2$

# Παράδειγμα

- $L = \{104, 102, 201, 101\}, t = 308, \varepsilon = 0.2$
- $\delta = \frac{\varepsilon}{4} = 0.05$

# Παράδειγμα

- $L = \{104, 102, 201, 101\}, t = 308, \varepsilon = 0.2$
- $\delta = \frac{\varepsilon}{4} = 0.05$
- $L_0 = \{0\}$

# Παράδειγμα

- $L = \{104, 102, 201, 101\}, t = 308, \varepsilon = 0.2$
- $\delta = \frac{\varepsilon}{4} = 0.05$
- $L_0 = \{0\}$
- $i = 1, L_1 = \{0, 104\}$
  - $L_1 = \{0, 104\}$ (όχι trimming)
  - $L_1 = \{0, 104\}$ (καμία αποκοπή)

# Παράδειγμα

- $L = \{104, 102, 201, 101\}, t = 308, \varepsilon = 0.2$
- $\delta = \frac{\varepsilon}{4} = 0.05$
- $L_0 = \{0\}$
- $i = 1, L_1 = \{0, 104\}$
  - $L_1 = \{0, 104\}$ (όχι trimming)
  - $L_1 = \{0, 104\}$ (καμία αποκοπή)
- $i = 2, L_2 = \{0, 102, 104, 206\}$
  - $L_2 = \{0, 102, 206\}$ (trimming)
  - $L_2 = \{0, 102, 206\}$ (καμία αποκοπή)

# Παράδειγμα

- $i = 3, L_3 = \{0, 102, 201, 206, 303, 407\}$
  - $L_3 = \{0, 102, 201, 303, 407\}$ (trimming)
  - $L_3 = \{0, 102, 201, 303\}$ (αποκοπή, $t < 407$)

# Παράδειγμα

- $i = 3, L_3 = \{0, 102, 201, 206, 303, 407\}$
  - $L_3 = \{0, 102, 201, 303, 407\}$ (trimming)
  - $L_3 = \{0, 102, 201, 303\}$ (αποκοπή, $t < 407$)
- $i = 4, L_4 = \{0, 101, 102, 201, 203, 302, 303, 404\}$
  - $L_4 = \{0, 101, 201, 302, 404\}$ (trimming)
  - $L_4 = \{0, 101, 201, 302\}$ (αποκοπή, $t < 404$)

# Παράδειγμα

- $i = 3$, $L_3 = \{0, 102, 201, 206, 303, 407\}$
  - $L_3 = \{0, 102, 201, 303, 407\}$ (trimming)
  - $L_3 = \{0, 102, 201, 303\}$ (αποκοπή, $t < 407$)
- $i = 4$, $L_4 = \{0, 101, 102, 201, 203, 302, 303, 404\}$
  - $L_4 = \{0, 101, 201, 302, 404\}$ (trimming)
  - $L_4 = \{0, 101, 201, 302\}$ (αποκοπή, $t < 404$)

$A = 302$
$OPT = 104 + 102 + 101 = 307$
Για $\varepsilon = 20\%$ παίρνουμε λύση λιγότερο από $2\%$ χειρότερη από τη βέλτιστη.

# Απόδειξη για φράξιμο λάθους

- $L_i \subseteq P_i$

# Απόδειξη για φράξιμο λάθους

- $L_i \subseteq P_i$
- $z^* = \sum_{s \in S'} s, \quad S' \subseteq S$ (output of algorithm)

# Απόδειξη για φράξιμο λάθους

- $L_i \subseteq P_i$
- $z^* = \sum_{s \in S'} s, \quad S' \subseteq S$ (output of algorithm)
- $\dfrac{|C - C^*|}{C^*} \leq \varepsilon(n) \Leftrightarrow C^*(1-\varepsilon) < C$

# Απόδειξη για φράξιμο λάθους

- $(1-\delta)y \leq z \leq y$

# Απόδειξη για φράξιμο λάθους

- $(1-\delta)y \leq z \leq y$
- $\forall y \in P_i \mid y \leq t \Rightarrow \exists z \in L_i^{'}$ τέτοιο ώστε: $(1-\frac{\varepsilon}{i})^i y \leq z \leq y$ (άσκηση) από επαγωγή στο $i$.
  $i = 1 : (1-\frac{\varepsilon}{1})^1 y \leq z \leq y$ αληθές
  $i = 2 : (1-\frac{\varepsilon}{2})^2 y \leq z \leq y$
  $i = k : (1-\frac{\varepsilon}{k})^k y \leq z \leq y \Rightarrow i = k+1 : (1-\frac{\varepsilon}{k+1})^{k+1} y \leq z \leq y$

# Απόδειξη για φράξιμο λάθους

- $(1-\delta)y \leq z \leq y$
- $\forall y \in P_i \mid y \leq t \Rightarrow \exists z \in L_i'$ τέτοιο ώστε: $(1-\frac{\varepsilon}{i})^i y \leq z \leq y$
  (άσκηση) από επαγωγή στο $i$.
  $i = 1 : (1-\frac{\varepsilon}{1})^1 y \leq z \leq y$ αληθές
  $i = 2 : (1-\frac{\varepsilon}{2})^2 y \leq z \leq y$
  $i = k : (1-\frac{\varepsilon}{k})^k y \leq z \leq y \Rightarrow i = k+1 : (1-\frac{\varepsilon}{k+1})^{k+1} y \leq z \leq y$
- Άν $y^* \in P_n \Rightarrow \exists z \in L_n'$ τέτοιο ώστε: $(1-\frac{\varepsilon}{n})^n y^* \leq z^* \leq y^*$

# Απόδειξη για φράξιμο λάθους

- $\frac{d}{dn}(1 - \frac{\varepsilon}{n})^n > 0$

# Απόδειξη για φράξιμο λάθους

- $\frac{d}{dn}(1 - \frac{\varepsilon}{n})^n > 0$
- $(1 - \frac{\varepsilon}{n})^n$ αύξουσα

# Απόδειξη για φράξιμο λάθους

- $\frac{d}{dn}(1 - \frac{\varepsilon}{n})^n > 0$
- $(1 - \frac{\varepsilon}{n})^n$ αύξουσα
- $(1 - \varepsilon) < (1 - \frac{\varepsilon}{n})^n \Rightarrow (1 - \varepsilon) y^* \leq z^*$

# Απόδειξη για φράξιμο χρόνου

- Μετά το trimming, για δύο διαδοχικά στοιχεία $z, z^{'}$ της $L_i$ με $z > z^{'}$ ισχύει ότι: $\dfrac{z}{z^{'}} > \dfrac{1}{1 - \dfrac{\varepsilon}{n}} (z^{'} < (1-\delta)z)$

# Απόδειξη για φράξιμο χρόνου

- Μετά το trimming, για δύο διαδοχικά στοιχεία $z, z^{'}$ της $L_i$ με $z > z^{'}$ ισχύει ότι: $\dfrac{z}{z^{'}} > \dfrac{1}{1-\dfrac{\varepsilon}{n}}(z^{'} < (1-\delta)z)$

- $|L_i| \leq \log_{\frac{1}{1-\frac{\varepsilon}{n}}} t = \dfrac{\ln t}{-\ln(1-\dfrac{\varepsilon}{n})} \leq \dfrac{n \ln t}{\varepsilon}$

# Απόδειξη για φράξιμο χρόνου

- Μετά το trimming, για δύο διαδοχικά στοιχεία $z, z^{'}$ της $L_i$ με $z > z^{'}$ ισχύει ότι: $\dfrac{z}{z^{'}} > \dfrac{1}{1-\dfrac{\varepsilon}{n}} (z^{'} < (1-\delta)z)$

- $|L_i| \leq \log_{\frac{1}{1-\frac{\varepsilon}{n}}} t = \dfrac{\ln t}{-\ln(1-\dfrac{\varepsilon}{n})} \leq \dfrac{n \ln t}{\varepsilon}$

- $\dfrac{x}{1+x} \leq \ln(1+x) \leq x, \;\; x > -1$