From now on: Advanced FPT techniques.

31. Fast Subset Convolution

Goal: · Solve problems that are formulated as computing all solutions of a certain shape
↳ Say all k-colorable induced subgraphs.

· Introduce operations on functions to the computation of which these problems can be reduced.
↳ Cover product and subset convolution.

Idea: · Generalize the principle of inclusion and exclusion.

31.1 Fast Zeta and Möbius Transforms

Goal: · Reformulate inclusion-exclusion algorithms via transforms between functions on the subset lattice.

· These transforms take some $f: \mathcal{P}(V) \to \mathbb{Z}$ and return $g: \mathcal{P}(V) \to \mathbb{Z}$.

· Besides coming up with actual algorithms, our goal is to study properties of these transforms.

Setting: · Let $|V| = n$.

· Consider functions on the powerset lattice $(\mathcal{P}(V), \subseteq)$.
More precisely,
$f, g: \mathcal{P}(V) \to \mathbb{Z}$,
but $\mathbb{Z}$ can be replaced by any ring in most cases.

· Write $\alpha \circ \beta$ as $\alpha\beta$, with $\alpha\beta(X) = \alpha(\beta(X))$.

Definition: Let $f : \mathbb{P}(V) \to \mathbb{Z}$.

- The **zeta transform** is $\zeta f : \mathbb{P}(V) \to \mathbb{Z}$ with
$$(\zeta f)(X) := \sum_{Y \subseteq X} f(Y).$$
One can see this as a relaxation of f.

- The **Möbius transform** is $\mu f : \mathbb{P}(V) \to \mathbb{Z}$ with
$$(\mu f)(X) := \sum_{Y \subseteq X} (-1)^{|X \setminus Y|} \cdot f(Y).$$

- The **odd-negation transform** is $\sigma f : \mathbb{P}(V) \to \mathbb{Z}$ with
$$(\sigma f)(X) := (-1)^{|X|} \cdot f(X).$$

Proposition (Homework):

1. $\sigma\sigma = id$ 2. $\zeta = \sigma\mu\sigma$ 3. $\mu = \sigma\zeta\sigma$.

The following formula will be the counterpart of the inclusion-exclusion principle.

Theorem (Inversion formula):

$\mu\zeta = \zeta\mu = id$, which means $\mu\zeta f = \zeta\mu f = f$ for all $f : \mathbb{P}(V) \to \mathbb{Z}$, which means $(\mu\zeta f)(X) = (\zeta\mu f)(X) = f(X)$ for all $X \subseteq V$.

Proof:
$$(\mu\zeta f)(X) \overset{(\text{Prop. }3)}{=} (\sigma\zeta\sigma\zeta f)(X)$$
$$\overset{(\text{Def } \sigma, \zeta)}{=} (-1)^{|X|} \cdot \sum_{Y \subseteq X} (\sigma\zeta f)(Y)$$
$$\overset{(\text{Def } \sigma, \zeta)}{=} (-1)^{|X|} \cdot \sum_{Y \subseteq X} (-1)^{|Y|} \cdot \sum_{Z \subseteq Y} f(Z)$$
$$\overset{(\text{Think about it})}{=} (-1)^{|X|} \cdot \sum_{Z \subseteq X} f(Z) \cdot \sum_{Z \subseteq Y \subseteq X} (-1)^{|Y|}$$

$$(\text{bolake } Z = X) = (-1)^{|X|} \cdot f(X) \cdot (-1)^{|X|}$$

$$+ (-1)^{|X|} \cdot \sum_{Z \subsetneq X} f(Z) \cdot \sum_{Z \subseteq Y \subseteq X} (-1)^{|Y|}$$

$$= f(X) + (-1)^{|X|} \cdot \sum_{Z \subseteq X} f(Z) \cdot \underbrace{\sum_{Z \subseteq Y \subseteq X} (-1)^{|Y|}}_{=0}$$

$$\left(\sum_{Z \subseteq Y \subseteq X} (-1)^{|Y|} = 0\right)^{(*)} = f(X).$$

To see $(*)$, note that every set A has the same number of odd and even sized subsets.

Let $a \in A$.

We partition all subsets into pairs:

For every $S \subseteq A \setminus \{a\}$, pair with $S \cup \{a\}$.

Each pair has a set with even and another set with odd cardinality.

- For $\varrho\mu = id$, note that

$$\varrho\mu \overset{(\text{Prop. 3})}{=} \varrho\sigma\varrho\sigma \overset{(\text{Prop. 1})}{=} \sigma\sigma\varrho\sigma\varrho\sigma$$

$$\overset{(\text{Prop. 3})}{=} \sigma\mu\varrho\sigma$$

$$\overset{(\text{Just shown})}{=} \sigma\sigma \overset{(\text{Prop. 1})}{=} id.$$

$\square$

- Recall <u>CHROMATIC NUMBER</u>:

<u>Given</u>: Graph G, $k \in \mathbb{N}$.

<u>Question</u>: Does G have a <u>k-coloring</u> $c: V(G) \to [1,k]$ (so that $c(u) \neq c(v)$ whenever u, v adjacent).

- Our goal is to solve CHROMATIC NUMBER using the transforms.

For $X \subseteq V(G)$, let

$f(X) :=$ the number of covers of X by k- independent sets
$=$ the number of k-tuples $(I_1, \dots, I_k)$ of independent sets in G with $\bigcup_{j=1}^{k} I_j = X$.

- Then $(\zeta f)(X) = \sum_{Y \subseteq X} f(Y) =$ number of k-tuples $(I_1, \dots, I_k)$ with $\bigcup_{j=1}^{k} I_j \subseteq X$.

With the previous notation,

$$(\zeta f)(X) = s(X)^k.$$

With Section 27, this means $(\zeta f)(X)$ can be computed

↳ after a dynamic programming preprocessing of $2^n \cdot n^{O(1)}$-time

↳ in polynomial time (just exponentiation).

- With the inversion formula,

$$\boxed{f(V(G)) = \mu(\zeta f(V(G))) = (\mu \zeta f)(V(G)).}$$

Hence, we can compute the values of f from ζf using μ.

Applying μ also takes $2^n \cdot n^{O(1)}$-time.

- Note that G is k-colorable iff $f(V(G)) \neq 0$. $\square$

- It will sometimes be useful to compute the zeta and Möbius transforms for <u>all</u> $X \subseteq V$.

- The algorithm that computes these values separately for all subsets needs

$$O\Big(\sum_{X \subseteq V} 2^{|X|}\Big) = O\Big(\sum_{i=0}^{n} \binom{n}{i} 2^i\Big) \overset{\substack{\text{(Binom.}\\ \text{theorem)}}}{=} O((2+1)^n) = O(3^n)$$

arithmetic operations and evaluations of the transformed function f.

This can be done better!

<u>Theorem (Fast Zeta and Möbius Transforms):</u>

Given all the 2^n values of f on the input all the 2^n values of ζf and of μf can be computed using $O(2^n \cdot n)$ arithmetic operations. (Note that this is polynomial in the size of the input.)

<u>Proof:</u>

Consider the zeta transform.

Let $V = \{1, \dots, n\}$.

We interpret f as a function on bit-vectors, i.e.

$$f(x_1, \dots, x_n) = f(X), \text{ where } i \in X \text{ iff } x_i = 1$$

// Characteristic vector.

With this,

$$(\zeta f)(x_1, \dots, x_n) = \sum_{y_1, \dots, y_n \in \{0,1\}} [y_1 \leq x_1, \dots, y_n \leq x_n] \cdot f(y_1, \dots, y_n).$$

Here, $[\dots]$ is 1, if the expression in the brackets holds,
0, otherwise.

- Let ζ_j fix the last $n-j$ bits, which means

$$\zeta_j(x_1, \dots, x_n) := \sum_{y_1, \dots, y_j \in \{0,1\}} [y_1 \leq x_1, \dots, y_j \leq x_j] \cdot f(y_1, \dots, y_j, \underbrace{x_{j+1}, \dots, x_n}_{\text{fixed.}})$$

Denote $\zeta_0(x_1, \dots, x_n) := f(x_1, \dots, x_n)$.

Note that $\boxed{\zeta_n(x_1, \dots, x_n) = (\zeta f)(x_1, \dots, x_n).}$

- The values of ζ_j can be found with the following dynamic programming:

$$\zeta_j(x_1, \dots, x_n) = \begin{cases} \zeta_{j-1}(x_1, \dots, x_n), & \text{if } x_j = 0 \\ \zeta_{j-1}(x_1, \dots, x_{j-1}, 0, x_{j+1}, \dots, x_n), & \text{if } x_j = 1 \\ + \zeta_{j-1}(x_1, \dots, x_{j-1}, 1, x_{j+1}, \dots, x_n) & \end{cases}$$

- The computation takes $O(\underbrace{2^n}_{\text{vectors}} \cdot \underbrace{n}_{\text{iterations}})$ arithmetic operations.

- For the Möbius transform, use this result plus Prop. 3 above.

Note: We stated the bound in terms of the number of arithmetic operations. If an arithmetic operation can be carried out in polynomial time, we obtain

$$2^n \cdot n^{O(1)}$$

as an upper bound on the runtime.

Application:

- Using the fast Möbius transform in the last step of the previous coloring algorithm, instead of computing

$$(\mu \zeta f)(V(G)),$$

we can compute

$$(\mu \zeta f)(X) \quad \text{for } \underline{\underline{\text{all}}} \ X \subseteq V(G).$$

- Note that for $X \subseteq V(G)$ we have

$$G[X] \text{ is } k\text{-colorable} \quad \text{iff} \quad f(X) \neq 0.$$

Hence, in time

$$2^n \cdot n^{O(1)}$$

we found **all** k-colorable induced subgraphs of G.

31.2 Fast Subset Convolution and Cover Product

Goal:
- Define and study two binary operations on functions on the subset lattice.
- These operations turn out particularly handy in that
  - ↳ many problems can be reduced to them and
  - ↳ they occur in many algorithms.

- Particularly well-suited for problems that reason over partitions of the input.
- Technically: Compute sums over partitions of the input (functions applied to these partitions).

Definition:

Consider $f, g : \mathbb{P}(V) \to \mathbb{Z}$.

- The subset convolution is the function $f * g : \mathbb{P}(V) \to \mathbb{Z}$ with

$$(f * g)(X) := \sum_{Y \subseteq X} f(X)\, g(Y \setminus X),$$

or equivalently,

$$(f * g)(X) := \sum_{\substack{A \cup B = X \\ A \cap B = \emptyset}} f(A)\, g(B)$$

- The cover product is $f *_c g : \mathbb{P}(V) \to \mathbb{Z}$ with

$$(f *_c g)(X) := \sum_{A \cup B = X} f(A) \cdot g(B).$$

- The pointwise product is $f \cdot g : \mathbb{P}(V) \to \mathbb{Z}$ with

$$(f \cdot g)(X) := f(X) \cdot g(X).$$

Fact: The pointwise product can be computed in $\mathcal{O}(2^{|V|})$ arithmetic operations.

Technical goal:
- Compute all the $2^{|V|}$ values of cover product and subset convolution.
- Will show that this works in $2^n \cdot n^{\mathcal{O}(1)}$ arithmetic operations $(n = |V|)$

Note that this is polynomial in the input size.

Note:
- The naive algorithms for subset convolution and cover product need

  $O(3^n)$ and $O(4^n)$

  arithmetic operations.
- Indeed, for subset convolution, the number is proportional to the pairs $Y \subseteq X \subseteq V$,

  thus $\sum_{k=0}^{n} \binom{n}{k} 2^k = 3^n$.
- For cover product, we have all triples (A, B, Y) with $A, B \subseteq Y \subseteq V$.

  There are $2^n \cdot 2^n = 4^n$ such triples (every choice of A, B determines one).

Lemma:
The zeta transform of the cover product is the pointwise product of the zeta-transformed arguments:

$$\zeta(f *_c g) = (\zeta f)(\zeta g).$$

Proof:
$$\zeta(f *_c g)(X) \overset{(\text{Def. } \zeta, *_c)}{=} \sum_{Y \subseteq X} \sum_{A \cup B = Y} f(A) g(B)$$

$$= \sum_{A \cup B \subseteq X} f(A) g(B)$$

$$= \Big(\sum_{A \subseteq X} f(A)\Big)\Big(\sum_{B \subseteq X} g(B)\Big) = [(\zeta f)(\zeta g)](X). \quad \square$$

By the inversion formula, we obtain the following consequence.

Corollary: $f *_c g = \mu((\zeta f)\cdot(\zeta g))$.

Using this corollary and the fast zeta and Möbius transforms from the main result in 31.1, we now have a fast algorithm for the cover product.

Theorem (Fast Cover Product):

For $f, g : \mathcal{P}(V) \to \mathbb{Z}$ given via all 2^n values for the inputs, we can compute all 2^n values of $f *_c g$ in $O(2^n \cdot n)$ arithmetic operations.

Proof:

We proceed like in the application to coloring.
The first observation is that

$$\zeta(f *_c g) = (\zeta f)(\zeta g) \text{ is easy to compute.}$$

Indeed,

ζf and ζg work in $O(2^n \cdot n)$ arithmetic operations by the previous main result.

Multiplying the results also takes $O(2^n)$ arithmetic operations by the above fact.

By the inversion formula, we obtain

$$f *_c g \text{ as } \mu\zeta(f *_c g).$$

Computing the fast Möbius transform again takes $O(2^n n)$ arithmetic operations. $\square$

- We now use the algorithm for cover product to compute the subset convolution.

- Intuitively, the observation is that cardinalities are enough to control disjointness. The idea is to consider, for every $0 \le i \le k$, the contribution of those terms in $f * g$ where $|A| = i$ and $|B| = |X| - i$.

Definition:

Let $f : \mathcal{P}(V) \to \mathbb{Z}$.

Then $f_i : \mathcal{P}(V) \to \mathbb{Z}$

with $f_i(X) := f(X) \cdot [\,|X| = i\,]$.

Theorem (Fast Subset Convolution):

Consider $f, g : \mathcal{P}(V) \to \mathbb{Z}$ given via their 2^n values. We can compute all 2^n values of $f * g$ in $O(2^n \cdot n^3)$ arithmetic operations.

Proof:

$$(f * g)(X) = \sum_{\substack{A \cup B = X \\ A \cap B = \emptyset}} f(A) \cdot g(B)$$

$$\overset{(\text{"Key trick"})}{=} \sum_{i=0}^{|X|} \sum_{\substack{A \cup B = X \\ |A| = i \\ |B| = |X| - i}} f(A)\, g(B)$$

$$\overset{(\text{Definition})}{=} \sum_{i=0}^{|X|} \sum_{A \cup B = X} f_i(A) \cdot g_{|X|-i}(B)$$

$$= \sum_{i=0}^{|X|} (f_i *_c g_{|X|-i})(X).$$

With this, the algorithm is as follows.

- For all $i, j \in \{0, \dots, n\}$ and $X \in \mathbb{P}(V)$, we compute and store

$$(f_i *_c g_j)(X).$$

- With the fast cover product from above, this takes

$$n^2 \cdot O(2^n \cdot n) = O(2^n \cdot n^3) \text{ operations.}$$

- With the above equations, we can then compute

$$(f * g)(X) \qquad \text{for all } X \in \mathbb{P}(V)$$

with another $O(\underset{\substack{\uparrow \\ \text{sets}}}{2^n} \cdot \underset{\substack{\uparrow \\ \text{sum over} \\ \text{the size}}}{n})$ operations. $\square$

31.3 Counting Colorings via (Iterated) Fast Subset Convolution:

Goal: - Devise an algorithm that counts the number of k-colorings in every induced subgraph of the input graph.

- By a k-coloring, we mean a function $c: \mathbb{P}(V(G)) \longrightarrow [1, k]$, and we count the number of distinct functions that are valid k-colorings.
- Consider colorings as distinct, if they partition the vertices into color classes the same way but the colors for the classes are different.

Algorithm: - Define $s: \mathbb{P}(V(G)) \longrightarrow \{0, 1\}$ such that

$$s(X) = 1 \text{ iff } X \text{ is an independent set.}$$

- Then $\underbrace{(s * \dots * s)}_{k\text{-times}}(X)$ is the number of k-colorings of $G[X]$.

Note that a k-coloring need not use all colors,
which happens when a set is empty.

- The value $(s * \dots * s)(V(G))$ can be found in time and space $2^n \cdot n^{O(1)}$ by applying fast subset convolution $k-1$ times.
  Note that, due to associativity of $*$ (exercise), we can also do this with $\log k$ fast subset convolutions by iterated squaring:
  $$s, \quad s^2 = s * s, \quad s^4 = (s^2)^2 = s^2 * s^2, \quad s^8 = (s^4)^2 = s^4 * s^4 \text{ etc.}$$

<u>Theorem</u>:
The number of k-colorings in every induced subgraph of the input graph G can be found in time and space $2^n \cdot n^{O(1)}$.