

1 Problem statement

Given a weighted undirected graph $G = (V, E)$ and two special vertices $s, t \in V, s \neq t$, find a subset $E' \subseteq E$ of minimum weight such that for all $v \in V \setminus \{s, t\}$ there exists a path from v to s in $(V \setminus \{t\}, E')$ and another path from v to t in $(V \setminus \{s\}, E')$ and return its weight. It is guaranteed that the solution exists.

2 Wrong approaches

This problem looks like a textbook MST with just a few modifications, since we could think of E' as an MST for both $G_s := (V \setminus \{t\}, E)$ and $G_t := (V \setminus \{s\}, E)$. One could think of E' as the union of an MST for G_s and another MST for G_t . However, this approach does not work, as after doing the union of both MST's there are some redundant edges that we could take away and still fulfill the requirements of the problem statement. We could try to prune these edges, but since there could be many different MST's for the same graph and choosing one or another could make the solution different, this approach does not work.

3 Matroid definitions

We will provide some basic matroid definitions so that we can tackle this problem using the matroid intersection optimization technique.

Definition 1. A set system $(E, \mathcal{F}), \mathcal{F} \subseteq 2^E$ is an **independence system** if

1. (M1) $\emptyset \in \mathcal{F}$
2. (M2) If $X \subseteq Y \in \mathcal{F}$, then $X \in \mathcal{F}$

The elements of \mathcal{F} are called **independent** and the elements of $2^E \setminus \mathcal{F}$ **dependent**. Minimal dependent sets are called **circuits** and maximal independent sets are called **bases**.

Definition 2. An independence system is a **matroid** if

1. (M3) If $X, Y \in \mathcal{F}$ and $|X| > |Y|$, then there is an $x \in X \setminus Y$ with $Y \cup \{x\} \in \mathcal{F}$.

Definition 3. A **graphic matroid** is a matroid (E, \mathcal{F}) where E is the set of edges of some undirected graph $G = (V, E)$ and

$$\mathcal{F} := \{F \subseteq E : (V, F) \text{ is a forest}\}$$

It can be proved that this independence system is indeed a matroid. Note that the bases of the graphic matroids are any spanning forest of G . In the case of connected graphs, they are just the spanning trees of G .

Definition 4. Let (E, \mathcal{F}) be an independence system. We define the **dual** of (E, \mathcal{F}) by (E, \mathcal{F}^*) , where

$$\mathcal{F}^* := \{F \subseteq E : \text{there is a basis } B \text{ of } (E, \mathcal{F}) \text{ such that } F \cap B = \emptyset\}$$

Proposition 5. *If (E, \mathcal{F}) is a matroid, then (E, \mathcal{F}^*) is a matroid as well.*

The duals of graphic matroids are called **bond matroids**. If (E, \mathcal{F}) is a bond matroid and G is connected, we can think of an independent set $X \in \mathcal{F}$ as a subset of edges such that, after removing X from E , there is still a spanning tree in the graph. In other words, X is independent if and only if $G' := (V, E \setminus X)$ is still connected.

4 Solving the problem

It is well known in matroid theory that, given two matroids $(E, \mathcal{F}_1), (E, \mathcal{F}_2)$ over the same ground set, we can calculate the **(weighted) matroid intersection** in polynomial time with Edmonds' algorithm, that is, the subset $X \in \mathcal{F}_1 \cap \mathcal{F}_2$ of greatest (weight) cardinality. The algorithm for finding the weighted matroid intersection is given in `weighted_matroid_isect.pdf`.

We can describe this problem as the weighted matroid intersection of the matroid of G_s and the bond matroid of G_t , since finding the subset of edges of least weight that keeps G_s and G_t connected is equivalent to finding the subset of edges of maximum weight that we can remove and keep both G_s and G_t connected.

To implement such a solution, we usually need an **independence oracle** that can answer the question $X \in \mathcal{F}_i$ for any X , as well as inserting and removing elements from X in an efficient manner.

For this problem regarding bond matroids, the best approach is to use a dynamic connectivity data structure (we can use Top Cut Trees or Euler Tour Trees) that, given a graph G , can do the following operations in $\mathcal{O}(\log^2 n)$:

1. Remove an edge (u, v) from the graph (cut).
2. Add an edge (u, v) to a graph (link).
3. Give the size of the connected component in which a vertex u belongs.