Tree decomposition techniques

venerdì 28 maggio 2021 14:54

HEAVY-LIGHT DECOMPOSITION

Problema. Abbiamo un albero su n nodi. Ogni arco ha un peso. Vogliamo:

- fare update del tipo: modifica il peso dell'arco $u-v$ in w;
- rispondere a query del tipo: qual è il peso minimo sul cammino semplice da x a y?

La soluzione naive impiega:

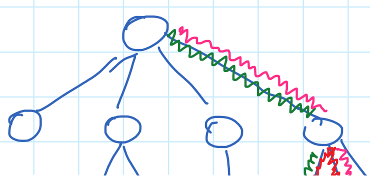
- $O(n)$ di preprocessing
- $O(1)$ per update
- $O(n)$ per query.

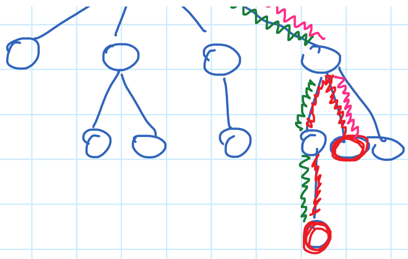
La HLD ci permette di passare alle seguenti complessità:

$O(n \log n)$ di preprocessing	$O(n \log f(n))$	in generale
$O(\log^2 n)$ per update	$O(\log n \, g(n))$	
$O(\log^2 n)$ per query	$O(\log n \, h(n))$	

L'idea è quella di costruire un'opportuna DS su alcuni cammini dell'albero.

Un possibile modo (non efficiente) di farlo è considerare i cammini dalla radice alle foglie. Su ognuno di questi cammini costruisco un segment tree.





Osservazioni: ogni cammino semplice è unione di (al più) due sottocammini dei cammini che ho considerato.

Questo mi permette di rispondere a query e update in $O(\log n)$

Issue: preprocessing e memoria quadratici

HLD suddivide l'albero in esattamente n cammini <u>disgiunti</u> (e che partizionano l'insieme dei nodi).

D'ora in poi, assumiamo che l'albero sia radicato.

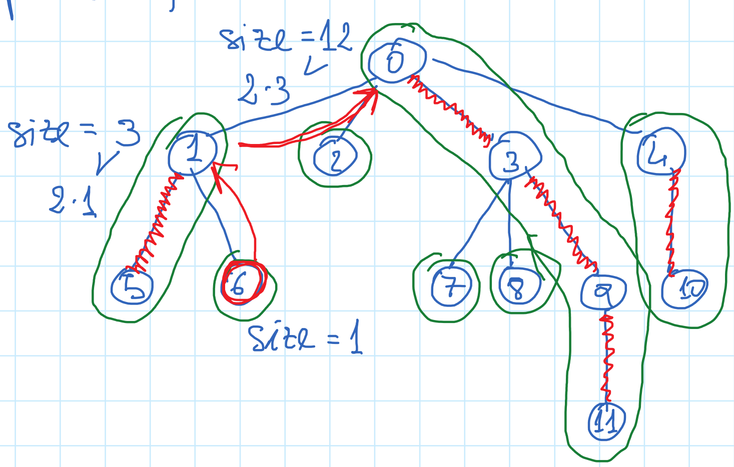
<u>Definizione</u>. Un arco si dice *pesante* se, detti u e v i suoi estremi, con u padre di v, si ha

$$\text{size}(\text{subtree}(v)) \geq \max_{w \in \text{children}(u)} \text{size}(\text{subtree}(w))$$

e, se $\text{size}(\text{subtree}(v)) = \text{size}(\text{subtree}(w))$, $v < w$.

Un arco non pesante si dice leggero.

I cammini che consideriamo sono quelli formati dagli archi pesanti, e li chiamiamo heavy paths.



Teorema. Sia v un nodo. Allora, nel cammino dalla radice a v ci sono al più $\log_2 n$ archi leggeri.

Dimostrazione. Risaliamo da v alla radice. Sia

$v = v_0 \to v_1 \to \dots \to v_k = r$ il cammino.

Supponiamo che $v_i - v_{i+1}$ sia un arco leggero.

Claim: $\text{size}(\text{subtree}(v_{i+1})) > 2 \cdot \text{size}(\text{subtree}(v_i))$.

Infatti $\exists\, u \neq v_i$ figlio di v_{i+1} tale che $v_{i+1} - u$ è un arco pesante. Allora

$$\text{size}(\text{subtree}(v_{i+1})) \geq 1 + \text{size}(st(v_i)) + \text{size}(st(u))$$
$$\geq 1 + 2 \cdot \text{size}(st(v_i))$$

Adesso è chiaro che non ci possono essere più di $\log_2 n$ archi leggeri (altrimenti $\underbrace{\text{size}(st(r))}_{= n} > 2^{\log_2 n}\, \underbrace{\text{size}(st(v))}_{\geq 1} \geq n$).

Questo teorema ci dice che è possibile decomporre ogni cammino semplice in $O(\log n)$ archi leggeri, alternati a $O(\log n)$ sottocammini di heavy paths.

Quindi basta costruire un segment tree su ciascun heavy path.

IMPLEMENTAZIONE

Linearizziamo l'albero. Facciamo partire una DFS dalla radice. Ogni volta che entriamo in un nodo, lo mettiamo in testa a una lista. In questo modo, la sequenza di nodi risultante ha la proprietà che ogni sottoalbero è un intervallo che inizia con v.

Ora, possiamo fare questa DFS visitando sempre per primo il figlio più pesante (quello corrispondente all'arco pesante). Questo fa sì che anche gli heavy path siano degli intervalli.

A questo punto, non è neppure necessario mantenere un segment tree diverso per ogni heavy path (ne basta uno solo).