Exercise VII, Sublinear Algorithms for Big Data Analysis 2024-2025

These exercises are for your own benefit. Feel free to collaborate and share your answers with other students, and solve as many problems as you can. Problems marked (*) are more difficult, but also more rewarding. These problems have been taken from various sources on the Internet, too numerous to cite individually.

1 Design an algorithm for testing whether a graph $G = (V, E), |V| = n$, given as a stream of edge updates (insertions and deletions) is bipartite. Your algorithm should use $n \log^{O(1)} n$ space and succeed with probability at least $1 - 1/n$.

Solution. We design a streaming algorithm to determine if a graph $G = (V, E)$, with $|V| = n$, given as a stream of edge updates, is bipartite, using $n \log^{O(1)} n$ space and succeeding with probability at least $1 - 1/n$. Given $G = (V, E)$, define $G' = (V', E')$ as follows: $V' = \{v^+, v^- \mid v \in V\}$, where each vertex $v \in V$ is split into v^+ and v^-, so $|V'| = 2n$. For each edge $(u, v) \in E$, add edges (u^-, v^+) and (u^+, v^-) to E'.

Our main claim will be that G is bipartite if and only if G' has exactly two connected components. First we argue that G bipartite $\implies$ G' has 2 components: Suppose G is bipartite with partition $V = A \cup B$, $A \cap B = \emptyset$, and all edges in E between A and B. In G': For $v \in A$, v^+ and v^- are not directly connected. For an edge (u, v) with $u \in A$, $v \in B$: $(u^-, v^+) \in E'$ connects u^- to v^+ and $(u^+, v^-) \in E'$ connects u^+ to v^-. Define:

$$C_1 = \{v^+ \mid v \in A\} \cup \{v^- \mid v \in B\},$$
$$C_2 = \{v^- \mid v \in A\} \cup \{v^+ \mid v \in B\}.$$

C_1 is connected: For $u \in A$, $v \in B$, the path from u to v in G connects $u^- \in C_1$ to $v^+ \in C_1$. Similarly C_2 is connected. No edges between C_1 and C_2: E' contains only (u^-, v^+) or (u^+, v^-), staying within C_1 or C_2. Thus, G' has exactly two components: C_1 and C_2.

Now we show that if G' has 2 components $\implies$ G is bipartite: Suppose G' has two components, D_1 and D_2. For each $v \in V$, v^+ and v^- are in different components (no direct edge in G'). Define $A = \{v \in V \mid v^+ \in D_1\}$, $B = \{v \in V \mid v^+ \in D_2\}$. Then:

- If $v \in A$, $v^+ \in D_1$, $v^- \in D_2$.
- If $v \in B$, $v^+ \in D_2$, $v^- \in D_1$.

For $(u, v) \in E$:

- $(u^-, v^+) \in E'$: If $u \in A$, $u^- \in D_2$, $v^+ \in D_1$, so $v \in B$.
- $(u^+, v^-) \in E'$ is consistent.
- Thus, (u, v) is between A and B.

No edges within A or B, so G is bipartite with partition A and B.

Now in the case G is not bipartite, then it has an odd cycle. Let the odd cycle in G be $v_1, v_2, \ldots, v_{2k+1}, v_1$: In G': $v_1^- \to v_2^+ \to v_2^- \to v_3^+ \to \cdots \to v_{2k+1}^- \to v_1^+$. This connects v_1^- to v_1^+, merging components. All v_i^+ and v_i^- become connected since there will be a path from each of them to some vertex on the cycle, reducing G' to one component. Thus If G is not bipartite, G' has 1 components.

Thus we can use a connectivity algorithm to test whether the graph G' has two components or not. The algorithm correctly tests bipartiteness using $O(n \log^{O(1)} n)$ space and probability $\geq 1 - 1/n$, as required. □

2 Suppose that that a graph $G = (V, E), |V| = n$, is given as a stream of edges, followed by an assignment of potentials $x \in \mathbb{R}^V$ to vertices of G. Design an algorithm that, given a precision parameter $\epsilon \in (0, 1/2)$, reads the stream of edges followed by the vector of potentials x, and outputs a $(1 \pm \epsilon)$-approximation to

$$\sum_{\{u,v\} \in E} (x_u - x_v)^2$$

using $\frac{1}{\epsilon^2} n \log^{O(1)} n$ space and succeeds with probability at least $1 - 1/n$.

Solution. Deferred to a later exercise set. □