# The Algorithm

We construct by taking and adding a new node on every edge. For example, say we have edge from , and we add a new node on it for , it would become two edges and .

So is constructed as such.

is .

is (all the original nodes/people from ).

# Proof of Correctness

Claim: if has an independent set of size then has a very independent subset of

the set of size .

Claim: if has no independent set of size k then also doesn’t have a very

independent subset of the set of size .

# Runtime Analysis

The reduction algorithm (computing , , and ) is linear in the number of edges because we go to every edge in and just add a new node, which effectively is deleting that old edge and adding two new edges. Getting and is constant because we’re just equating them to old inputs. If we let , this algorithm is .