DeJuan Anderson May 7, 2014
**6.046 Problem 5-1**
Collaborators: *Jake Barnwell, Anthony Adams, Michelle Johnson, Kang Zhang)* Recitation: Kang Zhang

*Heavily based on the document "How to Write a Proof of NP-completeness".*

**Given an undirected graph G = (V,E) and a positive integer L, does there exist a subset S of L vertices in V such that every vertex in V is either in S or is connected by an edge to some vertex in S? Prove that this problem is NP complete using a reduction from 3SAT.**

# 1 Proof that SFSP is in the class NP

First, we show that SFSP is in NP by showing that if the answer to a decision version of this problem is "yes", that we can verify the answer in polynomial time.

This problem was given in decision format, so no additional phrasing or seperate example is necessary. If the answer to the given instance of the problem is "yes", that such a subset exists, then a certificate that demonstrates this possibility would have L vertices in the subset S, with the remaining vertices, if any, connected to vertices in the subset S.

We can verify this in polynomial time by first checking that only vertices actually in the graph are used, then checking that the subset S contains L vertices in O(L) time, and lastly checking that the remaining —V— - —L— vertices are connected to some vertex in L. This last check could involve examining all edges in the graph to find the connections, and also depends on the number of edges, so it would take time polynomial in the number of edges E.

Therefore, given a "yes" instance, we can verify the certificate in time polynomial in L + E.

# 2 Reduction Phase: Proving that all NP-complete problems are polynomial time reducible to SFSP

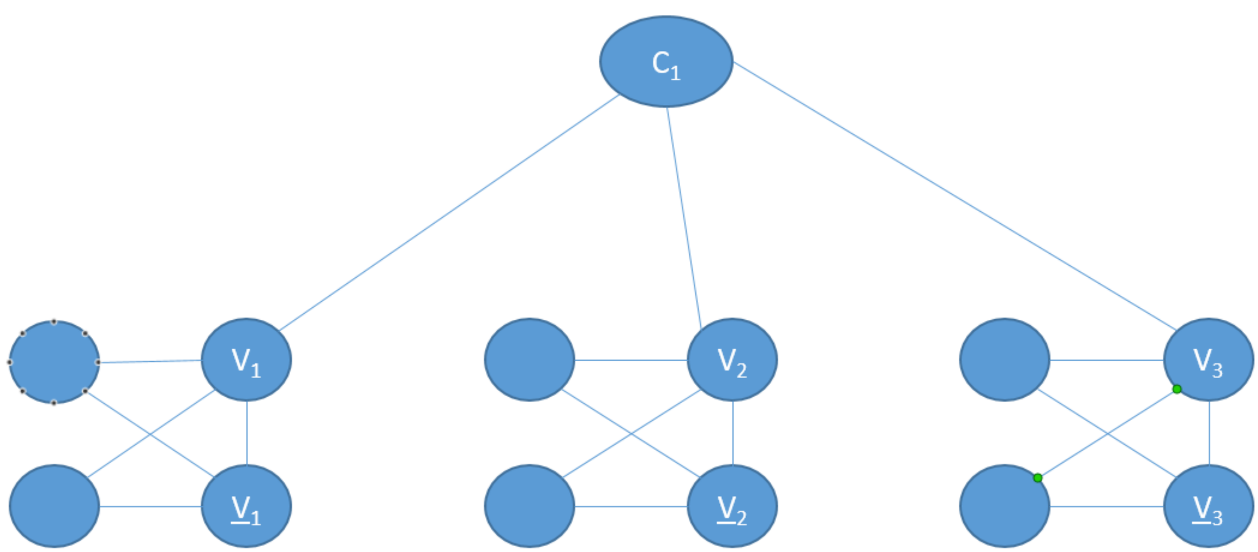
## 2.1 What is the reduction?

For this problem, we are required to reduce 3SAT to SFSP. So, consider an instance of 3SAT with v variables and L clauses. Construct a graph G from this instance as follows:

1) For each of the v variables, construct a pair of vertices $v_1$ and $\overline{v_1}$ and connect them with an edge. Now, construct two extra "dummy" vertices, and connect each to both $v_1$ and $\overline{v_1}$, but not to each other. Let each such construction be known as a "variable gadget".

2) For each clause in the 3SAT instance, construct a vertex. Connect this vertex to the three variable gadgets that make up the 3SAT clause; in particular, make this connection by connecting a clause $C_i$ with the three $v_i$ vertices in the three gadgets.

An image of a small part of the construction (only one clause is shown) is given below. In this representation, $\underline{V_i}$ represents $\overline{V_i}$.




To formulate this as an SFSP problem, we must ask the decision problem version of this question, where we ask if an SFSP set of size K can be found. We define this K as V for our reduction. This completes the description of the reduction.

## 2.2 Is it polynomial time?

This reduction is polynomial time in C and V.

For the C aspect, we construct one vertex per clause, and spend constant time connecting it to its three potentially satisfying vertices. Thus, we spend constant time per vertex and there will be V vertices.

As for the V aspect, it takes constant time to construct each variable gadget. We create four vertices for each gadget and use 5 edges to finish the construction, as shown in the image on the previous page. Thus, we spend constant time constructing each variable gadget as well, and there are V such gadgets.

This completes the argument that this reduction can be carried out in polynomial time.

## 2.3 Is the reduction correct? Step 1: If "yes" to 3SAT, "yes" to SFSP!

The reduction is indeed correct. Consider first an arbitrary 3SAT instance, X, and the corresponding SFSP version X'.

Assume that the answer to X is Yes. Then there exists a satisfactory assignment of variables for X. For each variable V in X, if it is assigned "true", then in the corresponding variable gadget, we would include $V_i$ in our SFSP set. Similarly, if it is assigned "false" then we include $\overline{V_i}$ version in our set.

Either way, we will include one of the two vertices. This uses one vertex per gadget. Note that since we need at least one vertex per gadget, there are V gadgets, and we have to choose exactly V vertices, we can never choose two vertices in one gadget.

We also can never choose one of the two unlabeled vertices, as these do not correspond to an assignment of a value in the 3SAT clause.

When we are done, we will have constructed an SFSP set of size V from our 3SAT instance.

For every vertex in a variable gadget, choosing either $V_i$ or $\overline{V_i}$ satisfies all vertices in the gadget, by construction. However, to satisfy the clause vertices, at least one of the $V_i$ it is connected to must have been included in our SFSP set, which will be true in all clauses since we know our 3SAT instance was satisfied and thus there was a true evaluation in at least one variable for every clause.

Thus, all clauses are satisfied and all variable gadgets are satisfied, meaning we have a valid SFSP cover.

This completes the proof that if we start with a 3SAT instance whose answer is "yes", then the answer to the corresponding constructed instance is "yes".

## 2.4 Is the reduction correct? Step 2: If "yes" to SFSP, "yes" to 3SAT!

Lastly, assume that the answer to our constructed SFSP graph X' is yes; that is, there exists an SFSP cover of size V.

In this case, notice that as stated earlier in the previous subsection, for the SFSP cover to be valid, by the construction of the graph, for every clause gadget, at least one of its three connections must connect to a variable $V_i$ that was included in our set.

Since $V_i$ corresponds to a variable having an evaluation of true, this means that by our construction, we've ensured that at least one variable in every 3SAT clause has an evaluation of true. This is sufficient to prove that the 3SAT instance is satisfied, since all clauses have at least one satisfying variable $V_i$, which evaluates to "true" in the corresponding 3SAT instance.

Q.E.D.