

# CPSC 320 2024W1: NP-Completeness Tutorial Problems

## 1 Strategically Placed Krispy Kremes

UBC Rec student leaders are planning their next fundraiser, and are seeking your help in identifying strategic locations to set up their stands of Krispy Kremes. They have a map showing  $n$  locations of buildings and outdoor spots on campus. Their  $k$  stands need to be set up in the outdoor spots. They want you to select  $k$  spots such that the maximum distance from any of the  $n$  locations to a stand is as small as possible. Formally an instance of the Strategically Placed Krispy Kremes optimization problem is a set  $V$  of size  $n$  (the locations), a subset  $S$  of  $V$  (the outdoor locations), and an integer  $k$ ,  $1 \leq k \leq |S|$ , as well as a symmetric matrix  $d[1..n][1..n]$  of distances between each pair of locations. The *optimization* problem is to find a subset  $S' \subseteq S$  of size  $k$ , so as to minimize

$$\max_{v \in V} \min_{s \in S'} \{d(s, v) \mid s \in S'\}.$$

The corresponding Strategically Placed Krispy Kremes *decision* problem, which we'll call SPKK, is as follows. An instance of SPKK has a set  $V$  of size  $n$ , a subset  $S$  of  $V$ , an integer  $k$ ,  $1 \leq k \leq |S|$ , and a symmetric matrix  $d[1..n][1..n]$ , plus an additional nonnegative integer  $b$ . The problem is to determine if there is a subset  $S' \subseteq S$  of size  $k$ , such that

$$\max_{v \in V} \min_{s \in S'} \{d[v][s] \mid s \in S'\} \leq b. \quad (1)$$

1. Show that the SPKK problem is in NP.

Let  $I = (V, S, d[1..n][1..n], k, b)$  be a (valid) instance of SPKK, and let  $S'$  be a potential solution for  $I$ , which we'll define simply to be a subset of  $V$ . Our certification algorithm works as follows. First, check that  $S'$  has size at most  $k$  and is a subset of  $S$ . Then, for each  $v \in V$ , check that  $\min_{s \in S'} \{d[v][s] \mid s \in S'\} \leq b$ . If both checks pass, output "Yes", and otherwise output "No".

The first check can be done in  $O(n \log n)$  time by sorting the subsets  $S$  and  $S'$ , and scanning the sorted lists to check that every entry in  $S'$  is also in  $S$ . The second check can be done in  $O(n^2)$  time, by enumerating all  $v$  in  $V$  and  $s$  in  $S'$ , keeping track of the quantity  $\min_{s \in S'} \{d[v][s] \mid s \in S'\}$  for each  $v$ , and comparing this quantity with  $b$ . So the algorithm runs in polynomial time.

The algorithm is correct because, if indeed  $I$  is a "Yes"-instance of SPKK, there exists a potential solution that passes the checks, namely the set  $S'$  that satisfies (2). Also, if  $I$  is a "No"-instance of SPKK, then there is no potential solution that passes the checks.

2. Suppose that, in specifying the decision problem, we replace the " $\leq$ " in (2) with "=", i.e.,

$$\max_{v \in V} \min_{s \in S'} \{d[v][s] \mid s \in S'\} = b. \quad (2)$$

Call this new decision problem variant SPKK'. Can we easily adapt our certification algorithm from part 1 to SPKK'?

Yes, we could also check that, for our chosen  $S'$ ,  $\max_{v \in V} \min_{s \in S'} \{d[v][s] \mid s \in S'\} = b$ .

---

## 2 Reductions: Vertex Cover and Dominating Sets

Your task is to find a reduction from Vertex Cover to Dominating Sets, two famous graph problems:

- *Vertex Cover*: An instance is a graph  $G = (V, E)$  and an integer  $K$ . The problem asks: Is there a vertex cover with at most  $K$  vertices in  $G$ ? Here, a vertex cover is a subset  $W$  of  $V$  such that  $|W| \leq K$ , such that every edge in  $E$  has at least one endpoint in  $W$ .
  - *Dominating Set*: An instance is a graph  $G = (V, E)$  and an integer  $K$ . The problem asks: Is there a dominating set with at most  $K$  vertices in  $G$ ? Here, a dominating set is a subset  $W$  of  $V$  such that  $|W| \leq K$ , such that every element of  $V - W$  is joined by an edge to an element of  $W$ .
1. Give a reduction from Vertex Cover to Dominating Set. Explain why your reduction is correct, and runs in polynomial time.

Given an instance  $(G, K)$  of Vertex Cover, construct an instance  $(G', K')$  of Dominating Set by adding one node  $v_e$  for each edge  $e \in E$ , and connecting this node to the endpoints of  $e$ . That is, if  $e = \{x, y\}$  then we add the edges  $\{v_e, x\}$  and  $\{v_e, y\}$ .

Choose  $K'$  to be  $K$  plus the number of isolated nodes of  $G$ .

The time to generate the new nodes and edges is  $O(m)$ .

Suppose that  $G$  has a vertex cover  $W$  of size at most  $K$ . Since  $W$  is a vertex cover of  $G$ , (i) all non-isolated nodes in  $V - W$  are joined by an edge to the nodes of  $W$  and (ii) all nodes  $v_e$  are also joined by an edge to the nodes of  $W$ . So the set  $W'$  which contains all nodes of  $W$ , plus the isolated nodes of  $G$ , is a dominating set of  $G'$  and has size at most  $K'$ .

Conversely, let  $W'$  be a dominating set of  $G'$ . Note that  $W'$  must contain all isolated nodes of  $G'$ . Let  $W$  be obtained from  $W'$  by discarding isolated nodes, and replacing any  $v_e$  node in  $W'$  by one of  $e$ 's endpoints. Then the nodes of  $W$  are all nodes of  $G$ , and have an edge to all of the  $v_e$  nodes. This means that every edge of  $G$  has at least one endpoint in  $W$ , and so  $W$  must be a vertex cover of  $G$ . [Note: Without the extra  $v_e$  nodes, this part of the reduction correctness would not hold.]