
CSC373H1 Summer 2014 Assignment 3

Names: John Armstrong, Henry Ku

SNs\CDF username: 993114492\g2jarmst, 998551348\g2kuhenr

Question #	Score
1	
2	
3	
4	
5	
6	
7	
8	
Total	

Acknowledgements:

"We have used lecture notes, the CLRS textbook, and wikipedia (specifically for polytope, gradient and normal definitions). Otherwise, we declare that we have not used any outside help in completing this assignment."

Name: John Armstrong, Henry Ku

Date: July 25, 2014

Q1. Construction:

To construct a family of graphs where $G(V, E)$ has exponential number of minimum cuts between source and terminal, we start with n as the number of vertices that appear between the source and the terminal, we will also assume that the capacity on each edge is 1. If $n = 0$ then there are no vertices between the source and the terminal and thus $V = \{s, t\}$ and $E = \emptyset$. Clearly, a single minimum-cut exists since we can only have $s \in S$ and $t \in T$, no edges exist to place a capacity on, so indeed $2^{|V|-2} = 2^0 = 1$.

For arbitrary n , $n \geq 1$, and $n = |V| - 2$, then $G(V, E)$ would be constructed such that $V = \{s, t, v_1, v_2, \dots, v_n\}$ and $E = \{(s, v_1), (v_1, t), (s, v_2), (v_2, t), \dots, (s, v_n), (v_n, t)\}$ (i.e. the only edges connected to a vertex are those that have s as their startpoint, or t as their endpoint). This arrangement is chosen because every cut in $G(V, E)$ will be a minimum cut. By the arrangement of $G(V, E)$, and since all capacities are the same. Whether a vertex v is or is not in the set S will not affect the summation of outgoing edge capacities, either the capacity from (v, t) or from (s, v) , respectively, will be used in computing the minimum cut. Therefore we have a maximum number of minimum cuts in G .

Proof of exponential minimum cuts:

From the description above, in our construction of $G(V, E)$, for any $v_i \in V$, v_i may or may not be contained within the set that describes a minimum cut in G . If we think of each minimum-cut as being described by a binary string of size n , $(q_{v_1}, q_{v_2}, \dots, q_{v_n})$, then we define each binary digit as:

$$q_{v_i} = \begin{cases} 1 & \text{if } v_i \in \text{minimum cut set} \\ 0 & \text{if } v_i \notin \text{minimum cut set} \end{cases}$$

Base: Let $n = 1$, and let the B_1 be the set of binary strings describing all possible minimum cuts in $G(V, E)$. Then B_1 contains two binary strings such that $B_1 = \{0, 1\}$. Indeed, $|B_1| = 2 = 2^n = 2^{|V|-2}$.

IH: Let $n = i$, $i > 1$, and let B_i be the set of binary strings describing the minimum cuts in $G(V, E)$, such that $V = \{s, t, v_1, v_2, \dots, v_i\}$ and $E = \{(s, v_1), (v_1, t), (s, v_2), (v_2, t), \dots, (s, v_i), (v_i, t)\}$. Suppose that $|B_i| = 2^i = 2^{|V|-2}$.

IS: Let $n = i + 1$, it follows from the construction description that $V' = \{s, t, v_1, v_2, \dots, v_i, v_{i+1}\}$ and $E' = \{(s, v_1), (v_1, t), (s, v_2), (v_2, t), \dots, (s, v_i), (v_i, t), (s, v_{i+1}), (v_{i+1}, t)\}$. Clearly, the addition of v_{i+1} extends $G(V, E)$ to $G(V', E')$. The minimum cuts in $G(V', E')$ are composed from the minimum cuts in $G(V, E)$ except that v_{i+1} is either inside or outside these previous cuts. Thus, to compute the set of minimum cuts as binary strings, B_{i+1} , simply take each binary string in B_i and append a 0 to be a minimum cut where v_{i+1} is outside the set, and then again take each binary string in B_i and append a 1 to be a minimum cut where v_{i+1} is inside the set. Thus, $|B_{i+1}| = 2 * |B_i| = 2 * 2^i = 2^{i+1}$.

By proof of induction this construction will have exponential minimum cuts between the source and the terminal. ■

Q2. Claim:

Function f , such that $f(S)$ is the number of edges (u, v) with $u \in S$, $v \in V \setminus S$, is submodular.

Proof:

Using the fact provided in the question, all that is necessary to show that f is submodular is to show that for any two subsets $A, B \subseteq V$, $f(A) + f(B) \geq f(A \cup B) + f(A \cap B)$.

Consider the set of edges in $G(V, E)$, $E = \{e_1, e_2, \dots, e_n\}$. If we think of $f(S)$ as an iterative algorithm that goes through each $e_i \in E$ and checks whether the startpoint of $e_i \in S$ and whether the endpoint of $e_i \in V \setminus S$. Then we have $f_{e_i}(S)$:

$$f_{e_i}(S) = \begin{cases} 1 & \text{if startpoint of } e_i \text{ in } S \text{ and endpoint of } e_i \text{ in } V \setminus S \\ 0 & \text{otherwise} \end{cases}$$

And so, $f(S) = \sum_{i=1}^n f_{e_i}(S)$. Suppose (u, v) is an arbitrary edge in E , also suppose A, B are some arbitrary subsets of V . We wish to show in all cases, however vertices u and v appear or do not appear in A and/or B , that $f_{(u,v)}(A) + f_{(u,v)}(B) \geq f_{(u,v)}(A \cup B) + f_{(u,v)}(A \cap B)$:

Case 1: $u \notin A$ and $u \notin B$: Clearly, $u \notin (A \cup B)$ and $u \notin (A \cap B)$, thus, $f_{(u,v)}(A) = f_{(u,v)}(B) = f_{(u,v)}(A \cup B) = f_{(u,v)}(A \cap B) = 0$. So, $f_{(u,v)}(A) + f_{(u,v)}(B) \geq f_{(u,v)}(A \cup B) + f_{(u,v)}(A \cap B)$ holds.

Case 2: $u \in A$ and $u \notin B$ (The case for $u \in B$ and $u \notin A$ is symmetric):

Case 2.1: $v \in A$: Clearly, $u, v \in (A \cup B)$ and $u \notin (A \cap B)$, it follows that $f_{(u,v)}(A) = f_{(u,v)}(B) = f_{(u,v)}(A \cup B) = f_{(u,v)}(A \cap B) = 0$. So, $f_{(u,v)}(A) + f_{(u,v)}(B) \geq f_{(u,v)}(A \cup B) + f_{(u,v)}(A \cap B)$ holds.

Case 2.2: $v \notin A$ and $v \in B$: So, $f_{(u,v)}(A) = 1$ and $f_{(u,v)}(B) = 0$, but since $u, v \in (A \cup B)$ and $u \notin (A \cap B)$, then $f_{(u,v)}(A \cup B) = f_{(u,v)}(A \cap B) = 0$. So, $f_{(u,v)}(A) + f_{(u,v)}(B) \geq f_{(u,v)}(A \cup B) + f_{(u,v)}(A \cap B)$ holds.

Case 2.3: $v \notin A$ and $v \notin B$: Again, $f_{(u,v)}(A) = 1$ and $f_{(u,v)}(B) = 0$. Also, $u \in (A \cup B)$ and $v \notin (A \cup B)$, then $f_{(u,v)}(A \cup B) = 1$. Again, $u \notin (A \cap B)$, then $f_{(u,v)}(A \cap B) = 0$. So, $f_{(u,v)}(A) + f_{(u,v)}(B) \geq f_{(u,v)}(A \cup B) + f_{(u,v)}(A \cap B)$ holds.

Case 3: $u \in A$ and $u \in B$:

Case 3.1: $v \in A$ and $v \in B$: Since $u, v \in A$ and B , then $f_{(u,v)}(A) = f_{(u,v)}(B) = 0$. Also, $u, v \in (A \cup B)$ and $(A \cap B)$, then $f_{(u,v)}(A \cup B) = f_{(u,v)}(A \cap B) = 0$. So, $f_{(u,v)}(A) + f_{(u,v)}(B) \geq f_{(u,v)}(A \cup B) + f_{(u,v)}(A \cap B)$ holds.

Case 3.2: $v \notin A$ and $v \in B$ (The case for $v \notin B$ and $v \in A$ is symmetric): It follows that $f_{(u,v)}(A) = 1$ and $f_{(u,v)}(B) = 0$. Since $u, v \in (A \cup B)$ then $f_{(u,v)}(A \cup B) = 0$, but $u \in (A \cap B)$ and $v \notin (A \cap B)$, thus $f_{(u,v)}(A \cap B) = 1$. Again, $f_{(u,v)}(A) + f_{(u,v)}(B) \geq f_{(u,v)}(A \cup B) + f_{(u,v)}(A \cap B)$ holds.

Case 3.3: $v \notin A$ and $v \notin B$: Clearly, $f_{(u,v)}(A) = 1$ and $f_{(u,v)}(B) = 1$. Also, $u \in (A \cup B)$ and $(A \cap B)$, but $v \notin (A \cup B)$ and $(A \cap B)$, then $f_{(u,v)}(A \cup B) = f_{(u,v)}(A \cap B) = 1$. Then, $f_{(u,v)}(A) + f_{(u,v)}(B) \geq f_{(u,v)}(A \cup B) + f_{(u,v)}(A \cap B)$ holds.

In all cases in which the vertices of some edge $(u, v) \in E$ might appear in $A, B \subseteq V$ it has been shown that $f_{(u,v)}(A) + f_{(u,v)}(B) \geq f_{(u,v)}(A \cup B) + f_{(u,v)}(A \cap B)$ holds. It must follow then that $\sum_{i=1}^n f_{e_i}(A) + \sum_{i=1}^n f_{e_i}(B) \geq \sum_{i=1}^n f_{e_i}(A \cup B) + \sum_{i=1}^n f_{e_i}(A \cap B) \rightarrow f(A) + f(B) \geq f(A \cup B) + f(A \cap B)$. Clearly, function f is submodular by the equivalence of submodular functions. ■

Q3. Consider the restructuring of LP formulation of maximum flow into standard form:

maximize: $\vec{c} \cdot \vec{x}$

subject to: $A\vec{x} \leq \vec{b}, \vec{x} \geq 0$.

Suppose this LP operates on $G(V, E)$ such that paths $P = \{p_1, \dots, p_n\}$ and $E = \{e_1, \dots, e_m\}$.

Let $\vec{c} = [1, \dots, 1]$, such that $|\vec{c}| = n$.

Let $\vec{x} = [x_1, \dots, x_n]$, such that $x_i = f_{p_i}$, maximum flow along path p_i .

Let A be an $m \times n$ matrix and let A be defined as follows:

$$A[i, j] = \begin{cases} 1 & \text{edge } e_i \text{ appears on path } p_j \\ 0 & \text{edge } e_i \text{ does not appear on path } p_j \end{cases}$$

Let $\vec{b} = [c(e_1), \dots, c(e_m)]$, such that $|\vec{b}| = m$.

Using the restructured LP, we construct the dual as follows:

minimize: $\vec{b} \cdot \vec{y}$

subject to: $A^t \vec{y} \geq \vec{c}, \vec{y} \geq 0$.

Let $\vec{y} = [y_1, \dots, y_m]$, y_i is a value associated with e_i .

Again, we define \vec{c} and \vec{b} as the same in the original construction.

Now, A^t is an $n \times m$ matrix, and so A^t is defined as follows:

$$A^t[i, j] = \begin{cases} 1 & \text{in path } p_i \text{ the edge } e_j \text{ appears} \\ 0 & \text{in path } p_i \text{ the edge } e_j \text{ does not appear} \end{cases}$$

The conclusion for the dual LP is that the objective function and constraints represent a construction of a max-flow min-cut, such that when the dual LP is complete its solution will equal the max-flow calculated by the original LP. It can now be stated:

$$y_i = \begin{cases} 1 & \text{edge } e_i \text{ connects sets } S \text{ and } T^* \\ 0 & \text{edge } e_i \text{ does not connect sets } S \text{ and } T^* \end{cases}$$

*clearly S and T represent the set of vertices separated by the minimum cut of $G(V, E)$.

Then the objective function represents the summation of the capacity of those edges where $y_i = 1$, and the summation will be the maximum flow in the network $G(V, E)$. The n path constraints, $A^t \vec{y} \geq \vec{c}$, represent the fact that on each path there exists at least one edge that connects set S to set T . Thus, for path p_i , $\sum_{j=1}^m a_{ij} y_j \geq 1$.

Q4. Polytope P is represented by $Ax \leq b$. Let $x \in \mathbb{R}^n$, so x is a point within P and so $Ax \leq b$ holds. More specifically, let A be an $m \times n$ matrix, and b a series of m constants, such that $a_{i1}x_1 + \dots + a_{in}x_n \leq b_i$ represents a closed half-space for each facet of the polytope, so that when all constraints are combined they represent all internal and boundary points of P .*

The LP must be designed so that we find a centre point y of a ball in P , such that all points on the surface of the ball that intersect the plane where P exists are also in P . Suppose some point x exists at the border of P , then for some i that represents the facet on which x exists, we have $a_{i1}x_1 + \dots + a_{in}x_n = b_i$. So, if the ball is the largest possible ball to fit in P the surface points must be as close to the boundaries as can be.

Suppose that $\vec{a}_i = [a_{i1}, \dots, a_{in}]$ is a unit vector for some facet i , such that $\|\vec{a}_i\| = 1$. Indeed, \vec{a}_i is the gradient unit vector to the half-space represented by $a_{i1}x_1 + \dots + a_{in}x_n \leq b_i$, its direction is also perpendicular to the surface of the half-space, and therefore the facet face. Also, as a gradient/normal vector, \vec{a}_i points in the direction outside the closed half-space. To discover if the ball at center y with radius $r \in \mathbb{R}$ can fit in P , we check for each facet i if the surface point $z_i = (\vec{a}_i^*r + y)$ is within P . To do this we use the constraint $a_{i1}z_{i1} + \dots + a_{in}z_{in} \leq b_i$. To explain z_i further, \vec{a}_i^*r is an r length vector that when added to y , projects a point to the surface or boundary of the ball and is the point on the ball closest to facet i when the ball is within P . In order to find the largest such ball, the LP must be designed to calculate both the ball's maximum radius r and its optimal centre point y . This is done by establishing z_i 's such that for all i , given constraints $a_{i1}z_{i1} + \dots + a_{in}z_{in} \leq b_i$, we want $a_{i1}z_{i1} + \dots + a_{in}z_{in}$ to be as close as possible to b_i for some maximum r and optimal centre point y .

Thus, the LP is:

maximize:

r

subject to:

$$\vec{a}_i \cdot (\vec{a}_i^*r + y) \leq b_i, \forall i, 1 \leq i \leq m$$

$$r \geq 0.$$

*explanation of polytope taken from wikipedia, http://en.wikipedia.org/wiki/Convex_polytope.

Q5. Suppose we have primal LP's for P_1 and P_2 respectively, with constraints that place x within the specific polyhedron:

maximize: c_1x

subject to: $A_1x \leq b_1$

maximize: c_2x

subject to: $A_2x \leq b_2$

Separately, their dual LP's are:

minimize: sb_1

subject to: $sA_1 = c_1, s \geq 0$.

minimize: tb_2

subject to: $tA_2 = c_2, t \geq 0$.

Suppose that P_1 and P_2 were to intersect and $c_1 + c_2 = c$, a primal LP that would represent this would be a combination of the two previous primal LP's:

maximize: cx

subject to: $A_1x + A_2x \leq b_1 + b_2$

Clearly the dual LP is a combination of the two previous dual LP's:

minimize: $sb_1 + tb_2$

subject to: $sA_1 + tA_2 = c, s, t \geq 0$.

However, we know that P_1 and P_2 do not intersect, so in the primal LP no x will be found with the provided constraint. The primal LP is infeasible. In the dual LP suppose we let c be the zero vector, then clearly there exists an s and t such that $sA_1 + tA_2 = \vec{0}$. By the theorem of duality, where the primal LP has an optimal solution if and only if the dual has an optimal solution, it follows that our dual LP cannot have an optimal solution. If the dual had a feasible solution it would contradict the infeasibility of the primal LP.

If both LP's had optimal solutions then by weak duality we would have, $(sA_1 + tA_2)x = cx \leq sb_1 + tb_2$, implying that, when $sA_1 + tA_2 = \vec{0}$ then $sb_1 + tb_2 \geq 0$. Indeed, in the feasible case, the result of the dual LP would be an upperbound to the result of the primal LP. It should be noted that the inequality $sb_1 + tb_2 \geq 0$ suggests that the dual has a minimum value, which if it did, then the result of the dual LP would be feasible, but this cannot be. In the situation where P_1 and P_2 do not intersect, the objective function of the dual, $sb_1 + tb_2$, is unbounded. It follows that there exists some $s, t \geq 0$ where $sA_1 + tA_2 = \vec{0}$ but $sb_1 + tb_2 < 0$. ■

Q6. Claim: Indeed the objective function, $\sum c(u, v)f(u, v)$, when minimized will produce the mean of the minimum mean cycle $C(V', E')$ in $G(V, E, c)$, which will equal $\frac{\sum_{e \in E'} c(e)}{|E'|}$.

Once the LP is complete the minimum mean cycle can be identified by the series of edges (u, v) where $f(u, v) > 0$.

By the first constraint, any vertex i in G , which has only outward pointing edges (i, j) , then $f(i, j)$ must equal 0. This is also the same for a vertex i' that has only inward pointing edges, so for any edge (j, i') , then $f(j, i')$ must equal 0. Indeed, if there were no cycles in G then for all edges (u, v) , $f(u, v) = 0$. However, if a cycle exists in G any edge (u', v') in the cycle is such that $f(u', v') \geq 0$, and for any two edges in the same cycle, say, (u'_1, v'_1) and (u'_2, v'_2) , then $f(u'_1, v'_1) = f(u'_2, v'_2)$, and this maintains the first constraint.

The second constraint, gives away the real meaning of what the f values represent. It was established that only with edges from a cycle will the f value be greater than or equal to zero. Also, all edges in the same cycle have the same f values. Thus, $\sum f(u, v) = 1$, such that when the LP is complete and that $f(u, v) > 0$ represent an edge in cycle $C(V', E')$, then $f(u, v) = \frac{1}{|E'|}$. Indeed, since $f(u, v) = \frac{1}{|E'|}$ and all other (u, v) not in $C(V', E')$, $f(u, v) = 0$, then the objective function indeed returns the mean of the minimum mean cycle, $\frac{\sum_{e \in E'} c(e)}{|E'|}$. ■

Converting the primal LP into standard form:

minimize: $c \cdot x$

subject to: $Ax = b, x \geq 0$.

Where $|V| = n$, and $|E| = m$. Then c and x are m length vectors where $c = [c(e_1), \dots, c(e_m)]$ and $x = [f(e_1), \dots, f(e_m)]$.

A is an $(n + 1) \times m$ matrix, such that:

$$A[i, j] = \begin{cases} 1 & \text{vertex } i \text{ is a startpoint in } e_j \\ -1 & \text{vertex } i \text{ is an endpoint in } e_j \\ 1 & i = n + 1 \end{cases}$$

Also, b is an $n + 1$ length vector such that the first n entries are zero and the entry $n + 1$ is 1. The final clause in A and the final element in b deal with the second constraint, $\sum f(u, v) = 1$.

Converting to the dual LP:

maximize: $y \cdot b$

subject to: $yA \leq c, y \geq 0$.

We do not know what y represents other than the fact that it is an $n + 1$ length vector. Indeed the first n elements in y must represent some function on each of the n vertices in $G(V, E)$, we will then take y_i to represent some value associated with vertex i . Additionally, let q be the last element of y . There are m constraints given that A is an $(n + 1) \times m$ matrix. Such that each constraint is based on each edge in E . It would follow that for each (u, v) the constraint would be $y_u - y_v + q \leq c(u, v)$. As defined in A , $y_u - y_v$, because u is a startpoint and v an endpoint, and the addition of q due to the row of 1's in the $(n + 1)^{th}$ row. Additionally, the objective function $y \cdot b = q$. Rewriting the dual:

maximize: q

subject to: $y_u - y_v + q \leq c(u, v), \forall (u, v)$.

Claim: Indeed the objective function, q , when maximized will produce the mean of the minimum mean cycle

$C(V', E')$ in $G(V, E, c)$, which will equal $\frac{\sum_{e \in E'} c(e)}{|E'|}$.

By the duality theorem, when the dual LP is complete, q should represent the mean of the minimum mean cycle. Suppose we rearrange the constraints such that $c(u, v) + y_v - y_u \geq q, \forall (u, v)$. Also, suppose we know some cycle $C^*(V^*, E^*)$ in $G(V, E, c)$. Now if we take only the constraints for the edges of that cycle and sum them together we have:

$$\begin{aligned} \sum_{(u,v) \in E^*} (c(u, v) + y_v - y_u) &\geq \sum_{(u,v) \in E^*} q \rightarrow \\ \sum_{(u,v) \in E^*} c(u, v) + \sum_{(u,v) \in E^*} (y_v - y_u) &\geq |E^*| * q \rightarrow \\ (\sum_{(u,v) \in E^*} c(u, v) + \sum_{(u,v) \in E^*} (y_v - y_u)) / |E^*| &\geq q \rightarrow \end{aligned}$$

As stated previously, the optimal value of q is the mean of the minimum mean cycle, a tight bound only occurs when $q = \frac{\sum_{e \in E'} c(e)}{|E'|}$, for the minimum mean cycle $C(V', E')$. In order to establish a tight bound on q , it follows for the edges of any arbitrary cycle that $\sum_{(u,v) \in E^*} (y_v - y_u) = 0$, then we have $q \leq \frac{\sum_{e \in E^*} c(e)}{|E^*|}$. Indeed if $C^*(V^*, E^*)$ is the minimum mean cycle then $q = \frac{\sum_{e \in E^*} c(e)}{|E^*|}$. ■

Q7. Let conventional milk be x_1 , Let organic milk be x_2

Profit from selling x_1 = selling price - buying price = \$2 - \$1 = \$1

Profit from selling x_2 = selling price - buying price = \$3 - \$1.50 = \$1.50

Objective function:

Maximize Profit = $x_1 + 1.5 x_2$

Constraints:

$$x_1 + x_2 \leq 3000$$

$$2x_2 \leq x_1 \rightarrow 2x_2 - x_1 \leq 0$$

$$x_1, x_2 \geq 0$$

First constraint shows how the combination of conventional milk and organic milk Alice sells in a week must be at least 3000 litre.

Second constraint shows that for every 1 litre of organic milk Alice buys, she needs to buy at least 2 litre of conventional milk.

Third line of constraint shows that there cannot be negative amount of conventional or organic milk.

Standard LP Form:

Maximize

$$\vec{c} = \begin{bmatrix} 1 \\ 1.5 \end{bmatrix} \bullet \vec{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Subject to:

$$\vec{A}\vec{x} = \begin{bmatrix} 1 & 1 \\ -1 & 2 \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \leq \begin{bmatrix} 3000 \\ 0 \end{bmatrix} = \vec{b}$$

$$\vec{x} \geq 0$$

Q8. To prove our question "q8" to be NP-Complete we need to prove that it is:

- 1) NP
- 2) NP Hard

NP:

Given the result set X:

Check size of set $X \leq k - O(1)$

Check that X has at least 1 element that coincides with every $S_i - O(n * m)$

These together take polynomial time

Therefore $q8 \in NP$ is satisfied

NP-Hard:

We show that q8 is NP Hard by reduction from the known NP-Complete SAT to our question q8

$SAT \leq_p q8$

Given SAT in CNF:

$$\phi = (z_1 \vee \overline{z_2} \vee \dots) \wedge (z_2 \vee \dots) \wedge \dots (\dots \vee \dots \vee \dots)$$

$$\phi = c_1 \wedge c_2 \wedge \dots \wedge c_p$$

Properties of the reduction:

1. Let the set of variables $Z = \{z_1, \dots, z_q, \overline{z_1}, \dots, \overline{z_q}\}$ such that $|Z| = 2q$, be converted to the set of marbles $M = \{x_1, \dots, x_q, x_{q+1}, \dots, x_n\}$, such that $|M| = n = 2q$.
2. For each clause c_i above, they represent the set $S_i \subseteq M$
3. When z_j (or $\overline{z_j}$) is in clause c_i , it means that marble x_j (or x_{j+q}) is in the set S_i , and if z_j (or $\overline{z_j}$) is not present in c_i , then x_j (or x_{j+q}) is not in the set S_i .
4. Set X, defined in the question, is constructed as follows. Each z_j (or $\overline{z_j}$) in a clause can have the value 0 or 1. If z_j (or $\overline{z_j}$) has the value 1, that means that marble x_j (or x_{j+q}) can be placed in the set X, 0 otherwise.
5. Finally, let $k = |X|$.

Claim: Reduction is polynomial

1. $O(q)$, this involves converting each variable to a marble once.
2. $O(m)$, suppose there are m clauses, then each clause is converted to a marble set once.
3. $O(mq)$, in the worst case each set contains all the variables possible.
4. $O(mq)$, this involves going through each clause, finding a variable that is assigned a true value, and then finding its corresponding marble and inserting it into X if it is not there already.
5. $O(q)$, in the worst case all marbles are in X.

Thus the reduction will be a polynomial algorithm.

Let's call the result of the reduction on ϕ , Y.

An example of Y:

Given:

$$\phi = (x_1 \vee x_2) \wedge (x_2 \vee x_3) \wedge (x_4)$$

$x_1 = 0, x_2 = 1, x_3 = 0$, and $x_4 = 1$

Then:

$M = [1,2,3,4], S_1 = [1,2], S_2 = [2,3], S_3 = [4]$

Now $X = [2,4]$ by the definition that $X \cap S_i \neq \emptyset$

Also, let $k = |X| = 2$.

In this example ϕ satisfies SAT and Y satisfies q8.

Claim: ϕ is satisfiable iff Y is satisfied by our question q8

First we show if ϕ is satisfiable \rightarrow Y is satisfiable:

If ϕ is satisfiable, it means that our CNF will output the value of 1

This means that all of our clauses in Y will have to output 1 due to \cap property, and each clause will have at least 1 element x_i that has the value 1, due to the \cup property.

Since each clause C_i represents the set S_i from our question, and since each z_j (or $\overline{z_j}$) in C_i with value 1 is a marble x_j (or x_{j+q}) in X, then that means each S_i will have a least one marble that exists in the set X.

Therefore $X \cap S_i \neq \emptyset$. Also, because the reduction sets $k = |X|$ it follows that q8 will be satisfied.

Now we show if Y is satisfiable $\rightarrow \phi$ is satisfiable:

If Y is satisfiable, it means that X has all the correct elements x_i such that $X \cap S_i \neq \emptyset$ for all S_i , and $k \geq |X|$.

Since X has all the elements to represent each clauses, it means that in each clause of C_i there will be at least one element with value 1. (By the properties of Y)

Because the clause has at least one element with the value 1, a clause taken separately would output the value 1 because all operators are or operators. Since all clauses output the value 1, with the and operators connecting all clause then the CNF will output 1, thus the SAT is satisfied.

Since the claim is proven, we have shown that $\text{SAT} \leq_p \text{q8}$ and that q8 is NP-Hard.

Since it is NP and NP-Hard, we have shown that q8 is NP-Complete. ■