Lab06-NP-Completeness

CS2308-Algorithm and Complexity, Xiaofeng Gao, Spring 2023

* Please upload your assignment to website. Contact Jiasen Li for any questions.
* Name: 张祥东 Student ID: 521030910206 Email: zhangxiangdong@sjtu.edu.cn

1. (Turing Machine) Design a three-tape TM M that computes the function $f(x,y) = x \cdot y$, where both x and y belong to the natural number set $\mathbb{N}$. The alphabet is $\{1, \Box, \rhd, \lhd\}$, where the input on the first tape is $x+1$ "1"'s and $y+1$ "1"'s (to distinguish "0" from "$\emptyset$") with a "$\Box$" as the separation. Below is the initial configurations for input (x,y). The result is the number of "1"'s on the output tape with the pattern of $\rhd 111\cdots111\lhd$. First describe your design and then write the specifications of M in the form like $\langle q_s, \rhd, \rhd, \rhd\rangle \to \langle q_1, \rhd, \rhd, R, R, R\rangle$ and explain the transition functions in detail (especially the meaning of each state).

Initial Configurations

Tape 1:	$\rhd$	1	1	$\cdots$	1	1	$\Box$	1	1	$\cdots$	1	1	$\lhd$
	$\uparrow$	$\leftarrow x+1$ squares $\rightarrow$						$\leftarrow y+1$ squares $\rightarrow$					
Tape 2:	$\rhd$	$\Box$	$\Box$	$\cdots$	$\cdots$	$\cdots$					$\Box$	$\Box$	$\Box$
	$\uparrow$												
Tape 3:	$\rhd$	$\Box$	$\Box$	$\cdots$	$\cdots$	$\cdots$					$\Box$	$\Box$	$\Box$
	$\uparrow$												

My design of this problem:First I copy $x+1$ "1"'s from tape 1 to tape 2.And start doing calculation process next. Then because the initial configurations for input (x,y) is $x+1$ "1"'s and $y+1$ "1"'s,so for convenience,I calculate $(x+1)\cdot(y+1)$ first,and then add 1,minus (y + 1) and minus (x + 1) sequentially. Because $x \cdot y = (x+1)\cdot(y+1)+1-(x+1)-(y+1)$,so my design can get correct answer at last.

In detail,each time I traverse $x+1$ "1" in tape 2,I move the reading head in the tape 1 with 1 step right.After $y+1$ "1" being used out,the $x+1$ "1" has been traversed for $y+1$ times.Each time the reading head in tape 2 move in the legal number area,the reading head in tape 3 write 1 and move right so we get $(x+1)\cdot(y+1)$.Then we add a 1 in tape 3.Then we tranverse $x+1$ "1" and $y+1$ "1",we do minus operation.Finally,add a 1 in tape 3 and write $\lhd$ in tape 3.

The specifications of M are as follows.

(a) $\langle q_s, \rhd, \rhd, \rhd\rangle \to \langle q_c, \rhd, \rhd, R, R, R\rangle$

(b) $\langle q_c, 1, \Box, \Box\rangle \to \langle q_c, 1, \Box, R, R, S\rangle$

(c) $\langle q_c, \Box, \Box, \Box\rangle \to \langle q_{ml}, \lhd, \Box, R, L, S\rangle$

(d) $\langle q_{ml}, 1, 1, \Box\rangle \to \langle q_{ml}, 1, 1, S, L, R\rangle$

(e) $\langle q_{ml}, 1, \rhd, \Box\rangle \to \langle q_{mr}, \rhd, \Box, R, R, S\rangle$

(f) $\langle q_{mr}, 1, 1, \Box\rangle \to \langle q_{mr}, 1, 1, S, R, R\rangle$

(g) $\langle q_{mr}, 1, \lhd, \Box\rangle \to \langle q_{ml}, \lhd, \Box, R, L, S\rangle$

(h) $\langle q_{ml}, \lhd, 1, \Box, \rangle \to \langle q_{minusl}, 1, 1, S, S, R\rangle$

(i) $\langle q_{mr}, \lhd, 1, \Box\rangle \to \langle q_{minusr}, 1, 1, S, S, R\rangle$

(j) $\langle q_{minusl}, \lhd, 1, \Box\rangle \to \langle q_{minusl}, 1, \Box, S, L, L\rangle$

(k) $\langle q_{minusl}, \lhd, 1, 1\rangle \to \langle q_{minusl}, 1, \Box, S, L, L\rangle$

(l) $\langle q_{minusl}, \lhd, \rhd, 1\rangle \to \langle q_{minus}, \rhd, 1, L, R, S\rangle$

(m) $\langle q_{minusr}, \lhd, 1, \Box\rangle \to \langle q_{minusr}, 1, \Box, S, R, L\rangle$

(n) $\langle q_{minusr}, \lhd, 1, 1\rangle \to \langle q_{minusr}, 1, \Box, S, R, L\rangle$

(o) $\langle q_{minusr}, \lhd, \lhd, 1\rangle \to \langle q_{minus}, \lhd, 1, L, L, S\rangle$

(p) $\langle q_{minus}, 1, 1, 1\rangle \to \langle q_{minus}, 1, 1, L, S, L\rangle$

(q) $\langle q_{minus}, \Box, 1, 1\rangle \to \langle q_{over}, 1, \lhd, S, S, S\rangle$

Here comes the explaination of the transition functions.(a)-(q) below is symbolic of transition functions above.

- q_s: start status.
  - (a) aims to move all reading head one step right and get into next status q_c.
- q_c: copy stage (copy x + 1 "1" to the tape 2).
  - (b) means when meeting 1 in tape 1,write 1 to tape 2 and move right.
  - (c) means when the x + 1 "1" is used out,write $\lhd$ to the tape 2 and move tape 1 right(becoming 1),move tape left(becoming 1) and then get into q_{ml}.
- q_{ml}: multiply(calculating $(x+1)\cdot(y+1)$) towards the left direction which means reading head in tape 2 move left.
  - (d) means when tape 2 is 1 and tape 1 is 1,write 1 to the tape 3(namely add 1) and make tape 1 stay,move tape 2 left and move tape 3 right.
  - (e) means when tape 2 is $\rhd$ which means (x + 1) "1" is used out at this iteration,we write nothing at this step,move reading head in the tape 1 right to get into next iteration and next step we enter q_{mr} to do multiply towards the right direction.
  - (h) means when tape 1 is used out which means $(x+1)\cdot(y+1)$ has completed.We get into status q_{minusl} which aims to minus (x + 1) towards the left direction.
- q_{mr}: multiply(calculating $(x+1)\cdot(y+1)$) towards the right direction which means reading head in tape 2 move right.
  - (f) means when tape 2 is 1 and tape 1 is 1,write 1 to the tape 3(namely add 1) and make tape 1 stay,move tape 2 right and move tape 3 right.
  - (g) means when tape 2 is $\lhd$ which means (x + 1) "1" is used out at this iteration,we write nothing at this step,move reading head in the tape 1 right to get into next iteration and next step we enter q_{ml} to do multiply towards the left direction.
  - (i) means when tape 1 is used out which means $(x+1)\cdot(y+1)$ has completed.We get into status q_{minusr} which aims to minus (x + 1) towards the right direction.

  note that in (h) and (i),I write 1 to the tape 3 and move reading head of tape 3 right which aims to calculate $(x+1)\cdot(y+1)+1$ at this step.
- q_{minusl}: minus (x + 1) towards the left direction.
  - (j) means when tape 2 is 1,minus 1 to the tape 3,namely doing (S,L,L) to the three tape.
  - (k) means when tape 2 is 1,minus 1 to the tape 3,namely doing (S,L,L) to the three tape which is just the same as (j).(j) is just set for taking special condition into consideration.
  - (l) means when minus(x + 1) over,stay reading head 3 and move reading head 2 right(make it be 1) to cut down next state q_{minus}'s consideration conditions.
- q_{minusr}: minus (x + 1) towards the right direction.

- (m) means when tape 2 is 1,minus 1 to the tape 3,namely doing (S,R,L) to the three tape.
- (n) means when tape 2 is 1,minus 1 to the tape 3,namely doing (S,R,L) to the three tape which is just the same as (m).(m) is just set for taking special condition into consideration.
- (o) means when minus(x + 1) over,stay reading head 3 and move reading head 2 left(make it be 1) to cut down next state q_{minus}'s consideration conditions.

- q_{minus} minus (y + 1) towards the left direction.(it's easy to figure out there only exists one direction for minus (y + 1) from my above operation)
  - (p) means when tape 1 is 1,minus 1 to the tape 3.
  - (q) means when minus (x + 1) completes,write $\triangleleft$ to the tape 3 and get into q_{over}.
- q_{over}: all operations complete.Above all,we get $(x+1)\cdot(y+1)+1-(x+1)-(y+1) = x\cdot y$ and $x \cdot y$ "1" has been written to the tape 3.

2. (NP-Problems) What is the "certificate" and "certifier" for the following problems?

(a) *STEINER TREE*: Given a graph $G = (V, E)$, a weight $w(e) \in \mathbb{Z}^+$ for each $e \in E$, a subset $R \subset V$, and a parameter k, is there a subtree of G that includes all the vertices of R and its sum of the edge weights is no more than k?

Solution. Take n = $|V|$,m = $|E|$.The input size is $O(n + m + n) = O(n + m)$.

certificate: a subtree of G is of size $O(n + m)$ is of polynomial relationship with input size $O(n + m)$.

certifier: check that the subtree of G includes all vertices of R and the sum of this subtree's edge weights is no more than k.It's $O(n^2+m)$ which is of polynomial relationship with input size $O(n + m)$.

Above all,the question is NP problem for its certificate and certifier is of polynomial relationship with input size and the certificate and certifier are listed above. □

(b) *BIN PACKING*: Given a finite set U of items, a size $s(u) \in \mathbb{Z}^+$ for each $u \in U$, a positive integer bin capacity B, and a positive integer k, is there a partition of U into disjoint sets $U_1, U_2, \cdots, U_k$ such that the sum of the sizes of the items in each U_i is B or less?

Solution. Take n = $|U|$.The input size is $O(n)$.

certificate: a partition of U into disjoint sets $U_1, U_2, \cdots, U_k$.The overall size of certificate is O(n) which is of polynomial relationship with input size $O(n)$.

certifier: check that the sum of the sizes of the items in each U_i is B or less.The time complexity for this operation is $O(n)$ for checking each Union's sum which is of polynomial relationship with input size $O(n)$.

Above all,the question is NP problem for its certificate and certifier is of polynomial relationship with input size and the certificate and certifier are listed above. □

(c) *DOMINATING SET*: Given a graph $G = (V, E)$ and a parameter k, is there a subset $D \subseteq V$ with no more than k vertices, such that every vertex not in D is adjacent to at least one vertex in D?

Solution. Take n = $|V|$,m = $|E|$.The input size is $O(n + m)$.

certificate: a subset D,$D \subseteq V$ with no more than k vertices.The size is O(n) which is of polynomial relationship with input size $O(n + m)$.

certifier: check that every vertex not in D is adjacent to at least one vertex in D.The time complexity for the certifier is $O(n^2 + mn)$ which is of polynomial relationship with input size $O(n + m)$.

Above all,the question is NP problem for its certificate and certifier is of polynomial relationship with input size and the certificate and certifier are listed above. $\square$

3. (Reduction) Please complete the following polynomial time (many-to-one) reduction.

   (a) *3-SAT* $\leq_p$ *3D-MATCHING*. Please convert an instance $\Phi = (x_1 \vee \overline{x_2} \vee \overline{x_3})$ of 3-SAT into an instance of 3D-MATCHING. (You can directly write X, Y, Z, and T without drawing a picture)

   Background knowledge:from the ppt in class,we know X,Y,Z are disjoint sets and the $T \subseteq X\text{x}Y\text{x}Z$.

   For this question,we have

$$X = \{g_1^1, g_1^2, g_1^3, g_1^R, g_1^C, g_2^C\}$$
$$Y = \{b_1^1, b_1^2, b_1^3, b_1^R, b_1^C, b_2^C\}$$
$$Z = \{r_1^1, r_2^1, r_1^2, r_2^2, r_1^3, r_2^3\}$$

$$\begin{aligned} T =\{&\{g_1^1, b_1^1, r_1^1\}, \{g_1^1, b_1^1, r_2^1\}, \{g_1^2, b_1^2, r_1^2\}, \{g_1^2, b_1^2, r_2^2\}, \{g_1^3, b_1^3, r_1^3\}, \{g_1^3, b_1^3, r_2^3\}, \\ &\{g_1^R, b_1^R, r_2^1\}, \{g_1^R, b_1^R, r_1^2\}, \{g_1^R, b_1^R, r_1^3\}, \\ &\{g_1^C, b_1^C, r_1^1\}, \{g_1^C, b_1^C, r_2^1\}, \{g_1^C, b_1^C, r_1^2\}, \{g_1^C, b_1^C, r_2^2\}, \{g_1^C, b_1^C, r_1^3\}, \{g_1^C, b_1^C, r_2^3\}, \\ &\{g_2^C, b_2^C, r_1^1\}, \{g_2^C, b_2^C, r_2^1\}, \{g_2^C, b_2^C, r_1^2\}, \{g_2^C, b_2^C, r_2^2\}, \{g_2^C, b_2^C, r_1^3\}, \{g_2^C, b_2^C, r_2^3\}\} \end{aligned} \tag{0.1}$$

   The instance $\Phi = (x_1 \vee \overline{x_2} \vee \overline{x_3})$ can be converted into a *3D-MATCHING* problem by above X,Y,Z,T.By finding a proper match in the T above,we can solve the previous *3-SAT* problem.

   (b) *3-SAT* $\leq_p$ *SUBSET SUM*. Please convert an instance $\Phi = (x_1 \vee \overline{x_2} \vee \overline{x_3}) \wedge (\overline{x_1} \vee \overline{x_2} \vee x_3)$ of 3-SAT into an instance of SUBSET SUM. (You can directly write the set S and the integer W without writing a table)

   Background knowledge:with a *3-SAT* problem with two clauses,we can tranform 3 variables in this question to 6 number which represent $x_1, \overline{x_1}, x_2, \overline{x_2}, x_3, \overline{x_3}$.In addtion to 6 number,add 2x2 = 4 number.Then,a *3-SAT* problem can be converted to *SUBSET SUM* problem.

$$\text{set S} = \{10010, 10001, 1000, 1011, 101, 110, 10, 20, 1, 2\}$$

$$W = 11144$$

   The instance $\Phi = (x_1 \vee \overline{x_2} \vee \overline{x_3}) \wedge (\overline{x_1} \vee \overline{x_2} \vee x_3)$ can be converted into a *SUBSET SUM* problem with the above set and W.We can solve the previous problem by choosing subset of S whose sum is exactly W.

4. (Reduction) Given a collection $\mathcal{C}$ of subsets of a finite set S, positive integer $k \leq |S|$, the *HITTING SET* problem asks whether there exist a subset $S' \subset S$ with $|S'| \leq k$ such that S' contains at least one element from each subset in $\mathcal{C}$.

Prove that: *SET COVER* $\leq_p$ *HITTING SET*.

Given *SET COVER* instance $U = \{u_1, u_2, \ldots, u_n\}$,$C = \{S_1, S_2, \ldots, S_m\}$,$S_i \subseteq U$,k, we construct a *HITTING SET* instance.

***note that my notation is different from the question statement.**

The construction instance of *HITTING SET* from the *SET COVER* instance is shown as follows.

$S' = \{S_1, S_2, \ldots, S_m\}$,$\mathcal{C} = \{U_1, U_2, \ldots, U_n\}$,$U_i = \{S_j \mid u_i \in S_j, \forall j \in \{1, \ldots, m\}\}$,$k' = k$.

X is a certificate of *SET COVER*.$X \in C$.$|X| \leq k$,$X = \{S_{x_1}, S_{x_2}, \ldots, S_{x_q}\}$.

$\Rightarrow$Y is a certificate of *HITTING SET*.$Y \in S'$,$|Y| \leq k'$.$Y = \{S_{x_1}, S_{x_2}, \ldots, S_{x_q}\}$

$\Rightarrow' s$ proof is as follows.

Proof. X is yes $\Rightarrow$ Y is yes.

From the certifier of *SET COVER* and X is yes,we know $|X| \leq k$ and $X \in C$ and

$$\bigcap X = U$$

We know,X=Y.Because $\bigcap X = U$,for any u_i,there exist at least one Set $S_y \in X$ that satisfies $u_i \in S_y$.

Then,in $\mathcal{C}$ of *HITTING SET* instance,for the same i,the U_i contains S_y for U_i satisfies $U_i = \{S_j \mid u_i \in S_j, \forall j \in \{1, \ldots, m\}\}$.$(U_i \in \mathcal{C})$

That's to say,for any element U_i in $\mathcal{C}$,there exist a Set S_y that satisfies $S_y \in U_i$ and $S_y \in Y$.So Y constains at least one element from each subset in $\mathcal{C}$.

So X is yes $\Rightarrow$ Y is yes. $\square$

$\Leftarrow' s$ proof is as follows.

Proof. Y is yes $\Rightarrow$ X is yes.

From the certifier of *HITTING SET* and Y is yes,we know $|Y| \leq k$ and $Y \in \mathcal{C}$. Besides,for every $i \in \{1, 2, \ldots, n\}$,there exists at least one element belonging to U_i,$U_i \in \mathcal{C}$ that belongs to Y.Call any of such elements as S_y

From the construction principle above $U_i = \{S_j \mid u_i \in S_j, \forall j \in \{1, \ldots, m\}\}$,we know $u_i \in S_y$.

Then from the statement above,for every u_i,there is at least one $S_y(S_y \in Y$ and $S_y \in U_i)$ that satisfies $u_i \in S_y$.So we have $\bigcap Y = U$.Due to X = Y,we have $\bigcap X = U$.Namely,X is yes(satisfies *SET COVER*'s certifier).

So Y is yes $\Rightarrow$ X is yes. $\square$

Above all,the *SET COVER* $\leq_p$ *HITTING SET* is true.Thus,the proof completes.