ADA HW4

和李長諺、石容居、鄭天盛討論

Problem 5

(1)

取$K = max(|T - \Sigma_{i=1}^{n}|a_i|| + 1, |a_i|)$
接著把S中所有元素都加上K並在S中加入n個K形成$S^{'}$，然後把T加上nK
因此$S^{'}$中所有數都不會是負的，符合$JJBAP_{+}$的條件

- 當$JJBAP_{+}$為True時，代表可以從$S^{'}$中取出k個數字$a^{'}_{b_1} ... a^{'}_{b_k}$
  使得$\Sigma_{i=1}^{k} a^{'}_{b_i} = T + nK$ (b_i為一數列)
  已知$K > \Sigma_{i=1}^{n}|a_i| - T,\ K > T - \Sigma_{i=1}^{n}|a_i|$
  - 假設$k = n - d\ (1 \le d < n, d \in N)$，因為$a^{'}_{i} = a_i + K$所以可以得到
    $\Sigma_{i=1}^{m} a_{b_i} + (n-d)K = T + nK\ (0 < m \le n - d)\ \rightarrow K = \frac{\Sigma_{i=1}^{m} a_{b_i} - T}{d}$且$K$為正

    取$d = 1$使得K最大則$K = \Sigma_{i=1}^{m} a_{b_i} - T \le \Sigma_{i=1}^{n}|a_i| - T$
    又$K > \Sigma_{i=1}^{n}|a_i| - T$ 因此矛盾，所以k不小於n
  - 假設$k = n + d\ (d \ge 1, d \in N)$，因為$a^{'}_{i} = a_i + K$所以可以得到
    $\Sigma_{i=1}^{m} a_{b_i} + (n+d)K = T + nK\ (0 < m \le n + d)\ \rightarrow K = \frac{T - \Sigma_{i=1}^{m} a_{b_i}}{d}$且$K$為正

    取$d = 1$使得K最大則$K = \Sigma_{i=1}^{m} a_{b_i} - T \le \Sigma_{i=1}^{n}|a_i| - T$
    又$K > \Sigma_{i=1}^{n}|a_i| - T$ 因此矛盾，所以k不大於n

  因此k一定會等於n，因此可以對兩邊同減去nK，$\Sigma_{i=1}^{k}(a^{'}_{b_i} - K) = T$
  而$a^{'}_{i}$減去K會是0或是a_i中的某個元素，因此$\Sigma_{i=1}^{k}(a^{'}_{b_i} - K)$會是一些$a_i$的元素和0的和
  因此可以得到某些a_i的和等於T，因此$JJBAP$為True
- 當$JJBAP$為True時，代表可以從S中取出一些數字$a_{b_1} ... a_{b_k}$
  使得$\Sigma_{i=1}^{m} a_{b_i} = T$ (b_i為一數列)
  可以對兩邊同加上nK，$\Sigma_{i=1}^{m}(a^{'}_{b_i} + K) + (n - m)K = T + nK$
  而a_i加上K和K會是$S^{'}$中的某個元素，且$m - n \le n$所以多加的K數量必不大於n
  因此可以得到$\Sigma_{i=1}^{n} a^{'}_{b_i} = T + nK$，所以$JJBAP_{+}$也為True
- 而轉換需要計算出K並把S中元素都加上K，並且把T加上nK
  因此時間複雜度會是$O(n)$，為polynomial

由此可知，$JJBAP \le_p JJBAP_{+}$

(2)

(3)

定義$DDBP$是給定一數列且元素的weight都在$[0,1]$之間，要確定這串數列是否可以被分成k個bins使得bins的weight不超過1公斤

取$DDBP$是分成兩個bins
先把S中的元素都看過，如果有任何一個超過$\frac{U}{2}$則直接輸出false
若都沒有則把S中的數字都除以$\frac{U}{2}$，得到$S^{'}$
因為有先確定過S中元素都不大於$\frac{U}{2}$，因此$\{0 \le a \le 1 | \forall a \in S^{'}\}$符合$DDBP$的條件且此時$S^{'}$的sum=2

因此當$DDBP$為True時，代表存在可以把$S^{'}$分成兩堆且兩堆的和都不大於1
因為$S^{'}$的和為2，所以若有一堆的和小於1，則另一堆的和必大於1
因此可以得到當$DDBP$為True時，代表可以把$S^{'}$分成和為1的兩堆
假設其中一堆為$b_1 ... b_m$，則有$\Sigma_{i=1}^{m} b_i = 1$
兩邊同乘$\frac{U}{2}$，$\Sigma_{i=1}^{m}(\frac{U}{2}b_i) = \frac{U}{2}$
而$S^{'}$又是S除以$\frac{U}{2}$而來，所以$\frac{U}{2}b_i \in S$
因此QQP也會得到True

當QQP為True時，代表可以把S分成大小相等的兩堆
把S中的元素都除以$\frac{U}{2}$後，因為所有元素必小於$\frac{U}{2}$
所以除完後所有元素必在[0,1]間，而等式就會變成有兩群元素各自的和是1，因此$DDBP$為True

而把QQP轉成$DDBP$要把S中所有元素都除以$\frac{U}{2}$，因此需要$O(n)$
由以上可以得出$QQP \le_p DDBP$，因此$QQP \le_p DBP$

(4)

令S分成兩組b_1, b_2是QQP的一組certificate，把S的總和算出來為$O(N)$
接著把b_1, b_2各自的和算出來也是$O(N)$，最後把b_1, b_2以及S的和做比較
因此QQP可以在polynomial time被verify，所以QQP是NP

令S分成n組$k_1 ... k_n$是$DDBP$的一組certificate
只要把$k_1 ... k_n$中每個元素加起來確認是否小於等於1就能verify $k_1 ... k_n$是一組解
因此$DDBP$是NP

(5)

假設存在polynomial time $G-approximation$ for DBP $(G<\frac{3}{2},G\in R)$
由QQP可知在$k=2$時為NPC問題
當正確答案為2時，因為$G<\frac{3}{2}$所以$2G<3$且為正整數 $(bin$數量必是整數$)$
因此由$G-approximation$可以找出答案為2，但由題可知$P\neq NP$
所以必不可能在polynomial time找出答案為2的解，因此矛盾
所以不存在polynomial time $G-approximation$ for DBP $(G<\frac{3}{2})$
也就是polynomial time $(\frac{3}{2}-\epsilon)-approximation$ for DBP $(\epsilon>0)$

(6)

$\Sigma_{i=1}^{\frac{1}{c}}H_i^m$
因為元素皆大於c所以取出的組合中最多會有$\frac{1}{c}$個元素，因此每組裡可能有1,2...$\frac{1}{c}$個元素
而這每一種可能的數量都是從m種元素中挑選出來的
令每種元素拿了$x_1...x_m$個，目前的組合中有k個元素$\rightarrow\Sigma_{i=1}^m x_i=k$
因此x_i的組合數最多會有H_k^m個，而全部加起來就是所求upper bound

(7)

可以視為在最多M種放法中，對於每種可能取出$x_1,x_2...x_M$
而由題目可知要取出k個放法，因此可以得到$\Sigma_{i=1}^M x_i=k$，所以x_i的組合會是H_k^M
$H_k^M=C_k^{M+k-1}=C_{M-1}^{M+k-1}$
而C_{M-1}^{M+k-1}中共有$M-1$項，且對於所有項而言$M+k-1$都會大於他們
因此可以得到$C_{M-1}^{M+k-1}<(M+k-1)^{M-1}$
又M是常數，取$c=M^{M-1}$,$ck^{M-1}=(MK)^{M-1}$
$(MK)^{M-1}-(M+k-1)^{M-1}=(MK-M-k+1)^{M-1}$
$=((M-1)(K-1))^{M-1}>0$，因此可以得到
$(M+k-1)^{M-1}=O(k^{M-1})=O(k^M)$

(8)

由(6)可知求出M的時間複雜度可以是$O(\frac{1}{c})=O(1)$
由(7)可知當有k個bins時，放的方法數會是$O(k^M)$
先算出每種球各有幾個，並算出M種取法各用到哪些球各幾個
接著從$k=1$開始看過所有放法中有沒有valid的，檢查是否valid需要$O(n)$
如果valid則當前的k即為所求
如此會跑過$O(1^M+2^M+...+k^M)$
假設複雜度為$d(1^M+...+k^M),d\in N$
令$c=d$，則對於所有$k>1$
$ck^{M+1}=c(k^M+k^M+...+k^M)$ (共k個)
$\rightarrow d(1^M+...+k^M)=c(1^M+...+k^M)<c(k^M+k^M+...+k^M)$

因此對於所有$k>1$，都能找到一個$c=d$使得$ck^{M+1}>d(1^M+2^M...+k^M)$
所以時間複雜度會是$O(k^{M+1})$
而k最多不會超過n，因此$O(k^{M+1})=O(n^{M+1})$
而計算$(c,m)-Kuan\ DDBP$時只要將$(c,m)-Kuan\ DBP$的答案和k比較即可，因此$(c,m)-Kuan\ DDBP$也可在多項式時間做出來，所以是P問題

Problem 6

(1)

G中每個頂點的degree都是偶數且G是connected

(2)

當樹只有一個點時，degree為奇數的點是0個因此符合
假設當任意一棵樹在有n個vertex的時候degree為奇數的點有偶數個
則當樹上有$n+1$個vertex時，存在某棵vertex數為n的樹加上葉子L後會變成這棵樹
當L接在一個degree為奇數的vertex K上的話，則K的degree會變成偶數
但L的degree=1也是奇數，因此這棵樹degree為奇數的點一樣有偶數個
當L接在一個degree為偶數的vertex K上的話，則K的degree會變成奇數
L的degree=1也是奇數，所以這棵樹degree為奇數的點會多兩個，因此一樣是偶數
由數學歸納法得出，不論樹的大小為多少，樹上degree為奇數的點必有偶數個

(3)

已知OPT的路徑會形成一個環$v_1...v_k,v_1$，且圖滿足三角不等式
將V'中的點沿著OPT路徑的順序排列後再將這些點照順序連接起來，會形成一個多邊形的環K且總$cost$必小於OPT，可以得到$cost(K)\leq OPT$
而取得perfect matching的方式可以交錯取出K中的邊，取法會有兩種恰將K分成兩堆路徑
因此必有一種perfect matching E使得
$cost(E)\leq\frac{cost(K)}{2}\Rightarrow cost(E)\leq\frac{cost(K)}{2}\leq OPT/2$

而minimum cost perfect matching的cost(M)必會小於等於E的cost
所以得到$cost(M)\leq cost(E)\leq OPT/2$
因此可以得到$cost(M)\leq OPT/2$

(4)

- 先建構出圖上的MST，令其為T
- 接著挑出T上degree為奇數的頂點集，令其為O且由(2)可知O中有偶數個點
- 然後利用$Oracle$找出O在原圖上的perfect matching，令其為M

- 接著把M中的邊加到T裡面，令修改完的圖為J (相同的邊加入時視為兩條)
- 如此一來可以得到J中的點degree都是偶數且connected，因此J中存在歐拉環
  - 因為O是T中degree為奇數的頂點集，且perfect matching會為每個點加上一條邊
    因此加回T後所有點的degree都會是偶數
- 接著找出圖上的歐拉環
  - 找歐拉環時，先選出一條沒走過的邊，接著一直走沒走過的邊，這樣走完所有點後會形成許多個環，接著把有交集的環的展開後連接起來，把所有點都做過後就能得到一個歐拉環
- 最後再把這個歐拉環改造成哈密頓cycle就是答案，改造的方式只要跳過歐拉環中重複的點就好

Prove:

假設最優解路徑為I，長度為$L(I)$、上述方法算出的路徑為H，路徑長為$L(H)$、最小生成樹的路徑為T，路徑長為$L(T)$、E為上面作法最後提到的歐拉環、degree為奇數的點集合為O，M為O上的minimum perfect matching，路徑長為$L(M)$

- $L(E) = L(T) + L(M)$且因為存在三角不等式所以$L(H) < L(E)$
- 又因為去掉I中任意一條路徑後，I就會變成一棵生成樹，所以$L(T) \leq L(I)$
- 令O的optimal TSP路徑為P，由(3)可知M一定會小於P的cost的一半
  如此可以得到$L(M) \leq \frac{L(P)}{2}$
- 又因為O只包含原圖中的部分點，因此
  $L(P) \leq L(I) \rightarrow L(K) \leq 0.5L(P) \leq 0.5L(I)$
- 總和以上可知，
  $L(H) < L(E) = L(T) + L(M) \leq L(I) + 0.5L(I) = 1.5L(I)$

Time complexity:

建構MST需要$O(ElogV)$
選出degree為奇數的頂點需要$O(V^2)$
構造minimum perfect matching是多項式時間
把M的邊加入T需要$O(E^2)$
找出並改造歐拉環都是多項式時間
因此全部的動作都能在polynomial time完成