21/9/20

1

Kousshik Raj
17CS30022

1\. Let the parameterised problem be (I, k).

For the forward direction, let us assume that (I, k) admits a kernel whose output is (I', k'). We know that ~~$|I'|+k'$~~ the kernel runs in polynomial time (by definition of kernel) and $|I'| + k' \leq g(k)$.

Now we can try an exhaustive search of the instance I' for a solution, which will be a computable function, $f(|I'|) \leq f(g(k))$.

Hence, we have an algorithm that runs in $O(f(k) + poly(n))$.

For the other direction, assume A solves (I, k) in $O(f(k) + poly(n)) \leq O(f(k) \cdot poly(n))$ $= O(f(k) \cdot n^c)$

Run the ~~steps~~ algo for exactly ~~[illegible]~~ n^{c+1} steps. If we have an answer by then output it. Else output the instance as it is. This is because $f(k) \cdot n^c \geq n^{c+1}$,

$f(k) \geq n \Rightarrow |I'| + k \leq f(k)$.

Thus we have a kernel for the problem (I, k) if there exists an algorithm running in $O(f(k) + poly(n))$

~~Thus~~.

Scanned with CamScanner

②

Kousshik Raj
17CS30022

2. a) For the forward direction, it is obvious because if there is a directed triangle, then the triangle itself is a considered a directed cycle.

For the other direction, ~~assume~~ let there be a directed cycle ~~~~ in the tournament T. Assume that the length of the shortest cycle is $l > 3$. ~~~~ As, T is a tournament, every ~~node~~ ~~~~ pair of nodes has a directed edge. Then there will be a chord for the circle, whose shortest length is l, as $l > 3$. ~~~~ Thus there will be ~~an~~ ~~an~~ shorter cycle of length $r (< l)$, which contradicts that l is the length of the shortest cycle. Hence, our assumption that the length of shortest cycle is greater than 3 is wrong.

∴ In tournaments, there ~~~~ is a directed cycle if and only if there is a directed triangle

b) ~~Since~~ ~~~~ Let the given instance be (T, k). First, we try to find a directed triangle ~~~~ in the tournament T, say C. If there is no such ~~~~ C, return that it is an YES-instance. Else, if $k \leq 0$, return that it is a NO-instance. Otherwise, ~~choose the~~

i) For FVST :-

~~Branch~~ Let x, y, z be the vertices in the directed triangle. Branch on $(T - \{x\}, k-1)$, $(T - \{y\}, k-1)$, $(T - \{z\}, k-1)$. If any 1 of

We try to remove each vertex & check.

Scanned with CamScanner

Kousshik Raj
17CS30022

them returns an YES-instance, add the corresponding vertex to the solution & return that it is an YES-instance. If all of them return NO, return that (T, k) is a NO-instance.

ii) For FAST :-
Let $x \to y \to z$ be the directed triangle.
Branch on $(T - \{(x,y)\} \cup \{(y,x)\}, k-1)$, $(T - \{(y,z)\} \cup \{(z,y)\}, k-1)$, ~~(T - ...)~~ $(T - \{(z,x)\} \cup \{(x,z)\}, k-1)$. That is we try reversing all possible edges in the directed triangle. If any of them return YES, add the corresponding edge to the solution and return YES. Else return that it is a NO.

As the branching factor is 3, and the maximum depth is k, we have an $O(3^k)$ algo, for both FAST & FVST.

3. a) Longest common subsequence of 2 permutations of length n, can be found in $O(n^2)$ with dynamic programming. Let P_1 & P_2 be the permutation. The following pseudo code exemplifies the idea. Here we find the length, but this can be backtracked to find the actual sequence.

```
ALGO:-  for i in 0.....n:
            dp[i,0] = 0
            dp[0,i] = 0

        for i in 1.....n:
        for j in 1.....n:
            if P1[i] = P2[j]:
                dp[i,j] = dp[i-1, j-1] + 1
            else:
                dp[i,j] = max(dp[i-1,j], dp[i,j-1])
```

$O(n^2) \leftarrow$ polynomial

④

Kousshik Raj
17CS30022

b) Since, $T-\{v\}$ is a DAG & v does not take part in any directed triangles, T is also a DAG. Let I be the topological ordering of $T-\{v\}$, and M be topological ordering of T. Since both are DAG & tournaments, a unique ordering exists.

Hence after inserting v in I, the ordering becomes M. Hence the position of v in I is the same as the position of v in M.

c) We can use an iterative compression algorithm for solving FVST.

<u>DISJOINT FVST:-</u>

Given tournament T, FVS W, k, we need to find FVS X such that $|X| \leq k$ & $X \cap W = \phi$. Let $A = V(T) - W$. Since W is a feedback vertex set $T[A]$ is acyclic. Let's reduce the instance.

i) If $T[W]$ has a directed triangle, there is no solution and hence return that this is a no instance.

ii) if a vertex $v \in A$ does not take part in any of the directed triangles in T, remove v.

iii) if there is a directed triangle with exactly 1 vertex in A, then add that vertex to the solution X, and decrease k by 1.

5

Kowshik Raj
17CS30022

Now we have vertices $v \in A$ such that $T[W \cup \{v\}]$ is DAG, by reduction rules.
Let F be the topological ordering of $T[W]$ & $\boxed{A_1}$ be the topological ordering of $T[A]$.

Now define a ordering ~~of~~ on A, say $\underline{A_2}$ where $u < v$ in A_2 if, ~~position of~~ $p[u] < p[v]$ or if $p[u] = p[v]$, then u occurs before v in A_1. Here $p[v]$ denotes the position of v in the topological ordering of $T[W \cup \{v\}]$.
As it is a DAG, such an ordering exists.

Let X' be the vertices belonging to the longest common subsequence of A_1 & A_2. The X' can be calculated in polynomial time. $X = A - X'$ is a minimum feedback vertex set. Hence, if $|X| > k$, return NO, else return X as a solution. Since this algorithm runs in $O(n^{O(1)})$, the compression algorithm runs in $O(2^k n^{O(1)})$ and the total iterative compression algorithm also runs in $O(2^k n^{O(1)})$.