CS 599—Graph Analytics Assignment

Zora Che

November 2021

Triangle Packing
Input: G(V,E), integer k
Output: Yes if G contains k node disjoint triangles, no otherwise.

1. Let $|V| = 3n$. Prove that when $k = n$ the problem is NP-complete.
   We would like to prove that the problem is NP-complete for the above condition. We will show that the problem is in NP, and that SAT can be reduced to our given problem in polynomial time, proving that our problem is NP-hard.
   Let our problem be $|V| = 3K$ disjoint triangle packing, we will describe a poly-time verifier to solve the problem. A certificate is a set of triangles using node indices of the graph: $(x_1, x_2, x_3), (x_4, x_5, x_6), ...(x_{3n-2}, x_{3n-1}, x_{3n})$. On such an input, the verier first checks for validity, such that the total number of nodes indices listed is 3n, and that the number of triangle tuples is n. We can polynomial check if the giving assignment satisfies $|V| = 3K$ disjoint triangle packing. For each tuple of node indices, check if the nodes are connected with each other in G. If false, there cannot be K disjoint triangle in total since, the verifier returns false. If true, the verifier marks the three indices as visited, and moves on to the next tuple. Repeating the checking of if the tuple of nodes are connected, while also checking if the nodes are already visited. If the nodes are visited, then the disjoint condition is violated and the verifier returns false. If the tuple is both all connected to each other and not visited, the verifier continues the same checking for the rest of the tuples until it exhausts the list. The verification will take polynomial time, since for each tuple, checking of edges existing for the three nodes is polynomial time, checking to see if each node is visited already with an updating array is polynomial time as well. The total running time would be O(poly(k)), since the triangle tuple list is length k and each tuple takes poly-time. We have proved that there exists a polynomial time verifier for our problem $|V| = 3K$ disjoint triangle packing. Then, our problem is in NP.

   Now, we would like to reduce 3-SAT to our given problem in polynomial time. Consider a boolean formula with 3n variables and n clauses with 3 literals each.

   For each 3-SAT instance, let the conjunction denote that the three nodes are marked as a triangle. The clause is satisfied if and only if the three nodes are connected and that they are connected to no other nodes in the set of all nodes V. Then, from the boolean formula we constructed a graph G with 3n nodes corresponding to 3n variables making up n possible disjoint triangles. Suppose that there exists a satisfying assignment of the 3-SAT boolean, then by construction there must exist k node disjoint triangles, for each triangle is valid and disjoint and there are k triangles created by the k clauses. On the other hand, suppose there does not exist a satisfying assignment of the 3-SAT boolean problem, then there cannot be k triangles, since at least one triangle would be invalid for the conjunction of the k clauses to be invalid. Since total nodes is 3n and we desire k node disjoint triangles. We have shown that 3-SAT reduces to $|V| = 3K$ disjoint triangle packing. The construction of the graph takes polynomial time, and since 3-SAT is NP-complete and reduces to $|V| = 3K$ disjoint triangle packing, we know that $|V| = 3K$ disjoint triangle packing is NP-hard.
   The given problem is both in NP and NP-hard, we know that it is NP-complete.