Theorem 1. *Assuming* ETH*, there is* $\epsilon > 0$ *such that for every large enough constant* k *the problem* k-CLIQUE *can't be solved in time less than* $n^{\epsilon \cdot k}$.

We will rely on the fact that ETH implies exponential hardness not only for 3SAT, but also for other problems that are reducible from 3SAT in linear-time; for example:

Proposition 2. *Assuming* ETH*, there is* $\epsilon' > 0$ *such that* 3-COLOR *on* n *vertices can't be solved in time less than* $2^{\epsilon' \cdot n}$.

Proof. Recall that ETH implies that 3SAT on n variables and $m = C \cdot n$ clauses requires time $2^{\epsilon \cdot n}$, for some $C > 1$. We saw in the tutorials a reduction of 3-SAT to 3-COLOR that maps such formulas to graphs with $m' = c \cdot (n + m) \leq \bar{C} \cdot n$ vertices. Hence, if we could solve 3-COLOR on m' vertices in time $2^{\epsilon' \cdot m'}$, where $\epsilon' < \epsilon / \bar{C}$, we could solve 3SAT in time $2^{\epsilon' \cdot m'} = 2^{\epsilon' \cdot \bar{C} \cdot n} < 2^{\epsilon \cdot n}$, a contradiction. ■

Proof of Theorem 1. The key idea in the proof is:

> We reduce 3-COLOR on an n-vertex graph G_0 to a k-CLIQUE problem on a graph G of size $N \approx 2^{n/k} \ll 2^{\epsilon \cdot n}$. Therefore any algorithm for k-CLIQUE with runtime $N^{\epsilon \cdot k}$ implies an algorithm for 3-COLOR with runtime $2^{\epsilon \cdot n}$.

In more detail, assuming ETH there is $\epsilon' > 0$ such that 3-COLOR can't be solved in time less than $2^{\epsilon' \cdot n}$. Let k be any integer larger than $100/\epsilon'$.

We reduce 3-COLOR to k-CLIQUE. Given a graph G_0 over n vertices, we partition the vertices into k sets $S_1, ..., S_k$ each of size at most n/k, and construct the following graph $G = (V_G, E_G)$ as an input for k-CLIQUE:

> For each $i \in [k]$ and each valid 3-coloring of S_i, we create a vertex $v \in V_G$ labeled by i and by the 3-coloring.
>
> For each two vertices $u, v \in V_G$ corresponding to different S_i and S_j, if their 3-colorings are consistent,[1] we add an edge (u, v).

Analysis. For correctness, observe that there is a valid 3-coloring of G_0 if and only if there is a k-clique in G. (Proof left as an exercise.)

Now, assume that for $\epsilon < \epsilon'$ there is an algorithm A that solves k-CLIQUE in time $N^{\epsilon \cdot k}$ on graphs with N vertices. Then, we can solve 3-COLOR on G_0 by constructing G and running $A(G)$. To bound the runtime, note that $N \stackrel{\text{def}}{=} |V_G| \leq k \cdot 3^{n/k}$, and hence the description size of G is at most $O(N^2) = O(k^2 \cdot 3^{2n/k})$, and we can construct G in time N^3. [2] Thus, the total runtime of the reduction and of A is

$$O(N^3) + N^{\epsilon \cdot k} \leq O(N^{\epsilon \cdot k}) = O(k \cdot 3^{\epsilon \cdot (n/k) \cdot k}) < 2^{\epsilon' \cdot n} \,,$$

relying on $k \geq 100/\epsilon'$ and on $\epsilon < \epsilon'$. This contradicts ETH. ■

---

[1] That is, for u, v labeled by 3-colorings of S_i, S_j, we want the resulting 3-coloring of $S_i \cup S_j$ to be valid.

[2] To see the N^3 runtime bound, consider going over each coloring of each S_i to check if it's valid, in time $O(k \cdot 3^{n/k} \cdot \text{poly}(n)) < N^2$; and then checking, for each pair S_i, S_j and each valid coloring of them, whether the partial colorings are consistent, in time $O(k^2 \cdot 3^{2n/k} \cdot \text{poly}(n)) < N^3$.