# CS 5003: Parameterized Algorithms

## Lectures 20-21

Krithika Ramaswamy

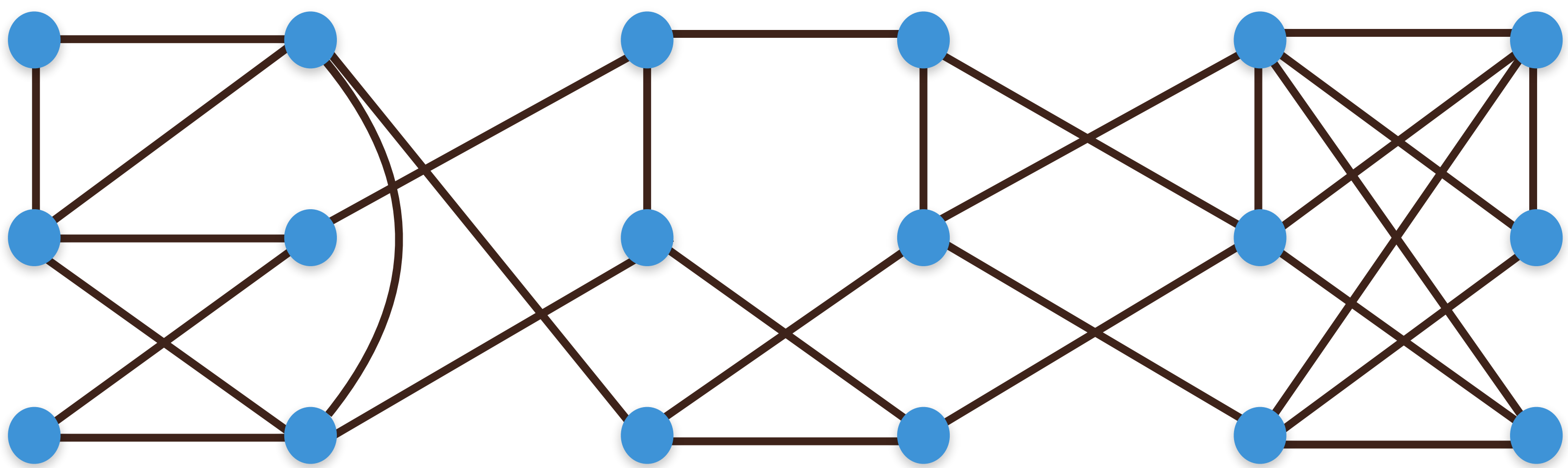
IIT Palakkad

References: Parameterized Algorithms by Cygan et al. and Kernelization by Fomin et al.

# Longest Path

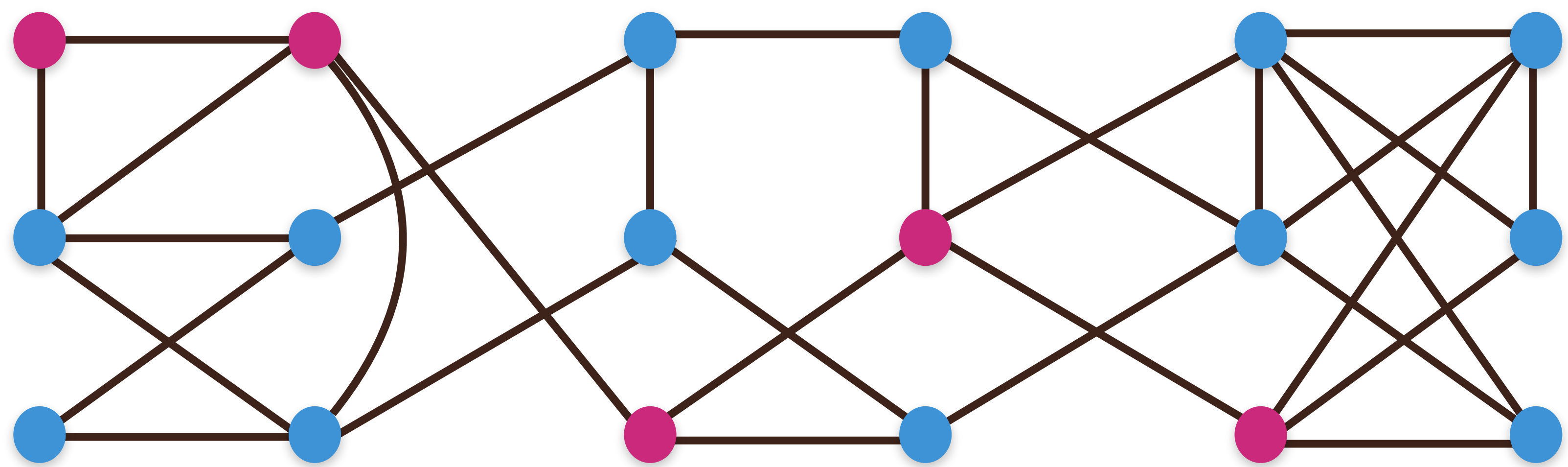
simple path -> vertices should not repeat length of path -> no. of vertices in a path. path = simp

Instance: An undirected graph G and an integer k
Question: Does there exist a (simple) path consisting of at least k vertices?
Parameter: k

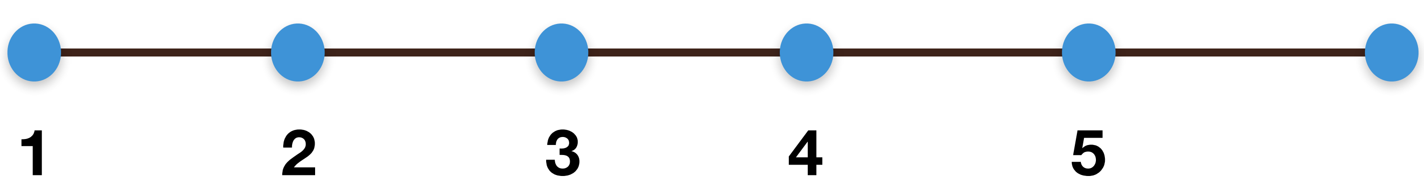


* NP-hard as Hamiltonian Path is a special case of Longest Path

# Longest Path



* Suppose we know that there is a 5-path.
* How to determine if there is a 6-path?
    * Look at the 5-path's last vertex and check if there is an "unused-neighbour"




$\in N(5) \backslash \{1,2,3,4\}$

# An Exponential Time Algorithm

* Define $\Gamma(v, X) = 1$ iff G has |X|-path using vertices in X and ending at v
  * G has a k-path iff $\Gamma(v, Z) = 1$ for some v and Z s.t |Z|=k
* Compute $\Gamma(v, X) = 1$ for all v and X such that |X| =1
  * For every v and every X with |X|=1, $\Gamma(v, X) = 1$ = iff $X = \{v\}$
* For each v, for each X with |X|>=2 and $v \in X$,
  * $\Gamma(v, X) = 1$ iff $\exists w \in N(v)$ s.t $\Gamma(w, X \backslash \{v\}) = 1$

$O(n^k n^2)$ algorithm

# Color Coding Algorithm

* Let Z denote {1,2, . . ., k}.
* Randomly color the vertices of G using colours from Z. Let $\chi$ denote this coloring.
* Focus on finding a colorful k-path: a path in which no 2 vertices have same colour
* Define $\Gamma(v, C) = 1$ iff G has colorful |C|-path using colours in C and ending at v
  * G has a colorful k-path iff $\Gamma(v, Z) = 1$ for some v in V(G)
* Compute $\Gamma(v, C) = 1$ for all v and C such that |C| =1
  * For every v and every i, $\Gamma(v, i) = 1$ = iff $\chi(v) = i$
* For each v, for each C with |C|>=2 and $\chi(v) \in C$,
  * $\Gamma(v, C) = 1$ iff $\exists w \in N(v)$ s.t $\Gamma(w, X \backslash \{\chi(v)\}) = 1$

O($2^k$ $n^2$) randomized algorithm

# Analysis

Identity1: $k! > (k/e)^k$

* Running Time: $O(2^k n^2)$ time
* Correctness:
  * If (G,k) is a no-instance then Algorithm is correct
  * If (G,k) is a yes-instance
    * The random colouring need not color the vertices of any k-path with distinct colours
    * Success probability >= $(k!\ \ k^{n-k})/k^n = k!/k^k > (k/e)^k/k^k = e^{-k}$

**Theorem:** Longest Path can be solved in randomized $O(2^k n^2)$ time, with success probability at least $e^{-k}$.

# Color Coding Algorithm

Given (G,k), run the following algorithm $e^k$ times. If one of the executions return yes, then declare that (G,k) is a yes-instance. Else, declare that (G,k) is a no-instance.

* Randomly color the vertices of G using colours from Z. Let $\chi$ denote this coloring.
* Define $\Gamma(v, C) = 1$ iff G has colorful |C|-path using colours in C and ending at v
  * G has a colorful k-path iff $\Gamma(v, Z) = 1$ for some v in V(G)
* Compute $\Gamma(v, C) = 1$ for all v and C such that |C| =1
  * For every v and every i, $\Gamma(v, i) = 1$ = iff $\chi(v) = i$
* For each v, for each C with |C|>=2 and $\chi(v) \in C$,
  * $\Gamma(v, C) = 1$ iff $\exists w \in N(v)$ s.t $\Gamma(w, X \backslash \{\chi(v)\}) = 1$

O((2e)$^k$ n$^2$) randomized algorithm

# Analysis

Identity1: $k! > (k/e)^k$

Identity2: $(1-p)^t <= (e^{-p})^t$

* Running Time: $O((2e)^k n^2)$ time
* Correctness:
  * If (G,k) is a no-instance then Algorithm is correct
  * If (G,k) is a yes-instance
    * The random colouring (of an execution) need not color the vertices of any k-path with distinct colours
    * Success probability >= $(k! \,.\, k^{n-k})/k^n = k!/k^k > (k/e)^k/k^k = e^{-k}$
    * Pr(No colorful k-path is found in all runs) <= $(1-e^{-k})$^$e^k$ <= $1/e$
    * Success probability >= $1-1/e > 1/2$

**Theorem:** Longest Path can be solved in randomized $O((2e)^k n^2)$ time, with constant success probability.

# Derandomization

**Definition:** An (n,k,r)-splitter F is a family of functions from [n] to [r] such that for every set $S \subseteq [n]$ of size k, there is a function f in F that splits S evenly. That is, for each pair $i, j \in [r]$, $|f^{-1}(i) \cap S|$ and $|f^{-1}(j) \cap S|$ differ by <=1.

**Definition:** An (n,k,k)-splitter is called an (n,k)-perfect hash family.

**Theorem:** For any n,k>=1, there is a construction of an (n,k,$k^2$)-splitter of size $k^{O(1)} \log n$ in time $k^{O(1)} n \log n$.

**Theorem:** For any n,k>=1, there is a construction of an (n,k)-perfect hash family of size $e^k k^{O(\log k)} \log n$ in time $e^k k^{O(\log k)} n \log n$.

# Color Coding Algorithm

Given (G,k), run the following algorithm for each coloring function f in F. If one of the executions return yes, then declare that (G,k) is a no-instance. Else, declare that (G,k) is a yes-instance.

* Color the vertices of G using f. Let $\chi$ denote this coloring.
* Define $\Gamma(v, C) = 1$ iff G has colorful |C|-path using colours in C and ending at v
  * G has a colorful k-path iff $\Gamma(v, Z) = 1$ for some v in V(G)
* Compute $\Gamma(v, C) = 1$ for all v and C such that |C| =1
  * For every v and every i, $\Gamma(v, i) = 1$ = iff $\chi(v) = i$
* For each v, for each C with |C|>=2 and $\chi(v) \in C$,
  * $\Gamma(v, C) = 1$ iff $\exists w \in N(v)$ s.t $\Gamma(w, X \setminus \{\chi(v)\}) = 1$

**Theorem:** Longest Path can be solved in $(2e)^k k^{O(\log k)} n^{O(1)}$ time.