1. Motivation

(a) Interval graphs: each vertex corresponds to an interval on the number line, two vertices connected if their intervals overlap.
(b) ‘Real’ instance in PL: variable conflict graphs in a program.
(c) Key problem in register allocation: color this graph with minimum number of colors.
(d) Optimum: greedy backwards in order of right endpoints of intervals.
(e) Let $N^+(u)$ be neighbors of $u$ that occur before it in the ordering. Pick color that’s not present in $N^+(u)$ and add it to $u$.
(f) Proof: need no more colors than $1 + |N^+(u)|$. This is also a lower bound since $\{u\} \cup N^+(u)$ is a clique.
(g) Reverse construction: given just the graph $G$, find this interval embedding.
(h) Key property used: ordering of vertices $v_1 \ldots v_n$ such that all ‘right neighbors’ of a vertex form a clique.
(i) Perfect elimination ordering (PEO): $N^+(u)$ is a clique for all $u$.
(j) Chordal graph: graphs with perfect elimination orderings.

2. Lexicographical BFS

(a) Due to Rose, Tarjan, and Lueker ’76, presentation based on survey by Corneil ’04.
(b) Assign labels $label(u)$ to all vertices.
(c) Initialize labels of all vertices to $\Lambda$.
(d) For $i = 1 \ldots n$
   i. Find vertex $u$ with least (lexicographical) label among remaining vertices. ($\Lambda$ is larger than all other entries)
   ii. Place it at position $i$ in ordering
   iii. Append $i$ to label of $N^+(u)$ before the $\Lambda$.
(e) Adjacent to earlier vertex guarantees earlier in list. Order generated is a BFS ordering.
(f) $O(mn^2)$ time naively.
(g) Faster algorithm: group vertices with same labels together.
(h) We never append same label twice, so more helpful to think about labels being $n$-bit masks.
(i) Final ordering a more refined version of what we start with.
(j) Track all equivalence classes in a linked list in sorted order, slices.
(k) Need to break up each slice based on $N^+(u)$.
(l) Can break all slices in $O(|N(u)|)$ time.
(m) Slices with non-empty intersection with $N(u)$ can be rearranged by one deletion and two insertion into linked list.
3. Claim: given a chordal graph, the reversal of the ordering generated by LBFS is a PEO.

(a) Many many structural properties of a LBFS.
(b) Certificate of non-chordal: chordless cycle $v_1 \ldots v_l$ such that no edge between $v_i, v_j$ with $|i - j| \neq 1$.
(c) Proof: suppose both a PEO and such a cycle exist. WLOG (by cyclic symmetry) $v_l$ is the last vertex in the PEO. Then by definition of PEO $v_l v_1 \in E$, contradiction.
(d) Use $u \preceq v$ to denote that $u$ occurs before $v$ in the order produced by LBFS.
(e) Lemma: if $c \preceq b \preceq a, ca \in E, cb \notin E$. Then there exist $d \preceq c$ such that $db \in E, da \notin E$.
(f) Proof: consider tie breaking when $b$ is chosen over $a$.
(g) Assume minimum counter example. LBFS produces $a_1 \ldots a_n$. Let $a_n = w$. Then $a_1 \ldots a_{n-1}$ is a PEO for $G \setminus \{w\}$.
(h) Let $u_1 \preceq v_1$ be the earliest pair of vertices s.t. $wu_1 \in E, wv_1 \in E, u_1 v_1 \notin E$. ‘Earliest’ defined by location of $u_1$ in ordering, with ties broken by $v_1$.
(i) Inductively show that if there is no chordless cycle, there exist $u_i \preceq v_i \preceq \ldots \preceq u_1 \preceq v_1 \preceq u_0 = w$ such that the edges between them are exactly $u_{i+1} u_i, v_{i+1} v_i$ and $u_1 w, v_1 w$.
(j) Base case follows from assumption.
(k) Inductive case:
   i. $v_i$ came before $u_{i-1}$ although $u_i u_{i-1} \in E, u_i v_i \notin E$. Exist $v \preceq u_i, v v_i \in E, v v_{i-1} \notin E$. Let $v_{i+1}$ be the earliest such $v$.
   A. Base case given by choice of $v_{i+1}$.
   B. Inductive case: $u_{i+1} u_{i+1} \in E, v_{i+1} u_{i+1} \in E$ by assumption of $a_1 \ldots a_{n-1}$ being PEO implies $v_{i+1} u_{i+1} \in E$. Contradiction.
   iii. For all $i' \prec i, v_{i+1} u_{i'} \notin E$. Otherwise, we would choose $v_{i+1}$ in place of $v_{i+1}$.
   iv. $v_{i+1} w \notin E$: when $i = 1$, follows from $w = u_0$. Otherwise, $v_{i+1}, v_{i+1}, \ldots, v_i, w$ gives a chordless cycle.
   v. $v_{i+1} u_{i} \notin E$: otherwise, $v_{i+1} v_i, \ldots, v_1, w, u_1, \ldots, u_i$ gives a chordless cycle.
   vi. Now needs to exhibit $u_{i+1}$. Almost a mirror proof of the existence of $v_{i+1}$.

4. Extensions / other uses

(a) Tarjan, Yannakakis ‘84: maximum cardinality search (add vertex with most visited neighbors) also generates an ordering whose reversal is a PEO for chordal graphs.
(b) LBFS can be viewed as an iterative partition/refinement routine, has more structure.
(c) $S_{uv}$: minimal slice that contain both $u$ and $v$.
(d) prior path lemma: let $t$ be first vertex in $S_{uv}$. Exist path from $t$ to $u$ that misses $v$ (does not go through $N(v)$). Furthermore, all vertices on this path occur before $u$.
(e) Proof idea: $t$ partitions $u$ and $v$ into different slices.
(f) Can also start with an initial ordering of vertices for extra tie breaking.
(g) Can recover unit interval graphs and interval graphs using 3 and 5 sweeps respectively.
(h) Can also recognize cographs (PS2, problem 8) in 3 sweeps.
(i) Approximate diameter?