Graph Decomposition

Md. Saidur Rahman

Department of Computer Science and Engineering,
Bangladesh University of Engineering and Technology, Dhaka

June 2016

Outline

1. Recursively Decomposable Graph

2. Clique Separator Decompostion

Recursively Decomposable Graph

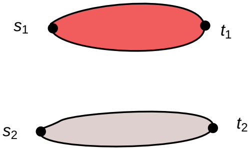
- Graphs are composed or decomposed in such a way that interactions between the graph is localized to a small subset of vertices, which are called *terminals*. A *k-terminal graph* is a graph in which at most k vertices are designated as terminals.

Recursively Decomposable Graph

- **Composition operation:** Combine k-terminal graphs according to one of a fixed set of rules. The composition rules may combine the graphs by identifying terminals of one graph with terminals of the other graph, and choosing a new set of terminals from the set of terminals of the two subgraphs. Edges in each of the original subgraphs are preserved in the new graph. A graph class is defined by allowing all graphs which can be built up using these rules from a finite set of base graphs.

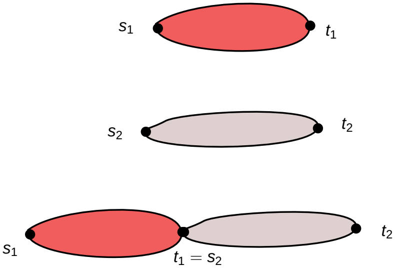
Recursively Decomposable Graph

Composition Rules for Series-Parallel Graphs



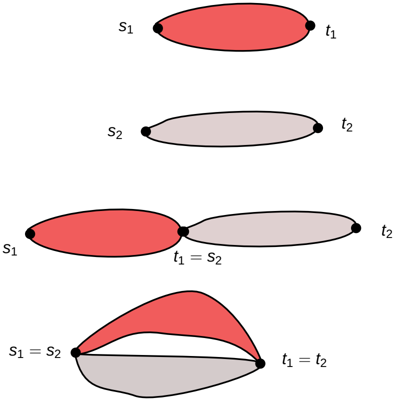
Recursively Decomposable Graph

Composition Rules for Series-Parallel Graphs



Recursively Decomposable Graph

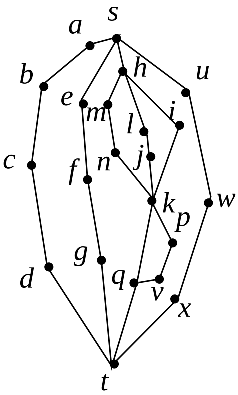
Composition Rules for Series-Parallel Graphs



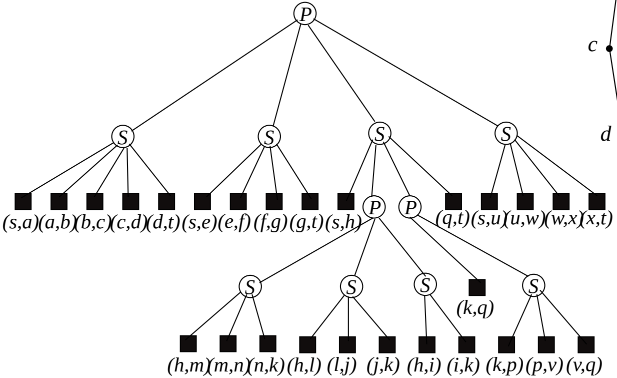
Recursively Decomposable Graph

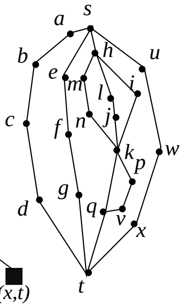
Decomposition: A graph G to be decomposed into G_1 and G_2 if G_1 and G_2 can be combined to get G using a composite rule. We can get a natural decomposition tree. Internal nodes indicate type of rules and all leaves are base graphs.

Decomposition Tree of a Series-Parallel Graph



Decomposition Tree of a Series-Parallel Graph





Application of Recursive Decompostion

Clique separator decomposition is used to devise dynamic programming or devide and conqure technique to deal with optimization problems which are hard for general graphs.

- Independent set problem
- Coloring
- Ranking
- Graph Drawing

Outline

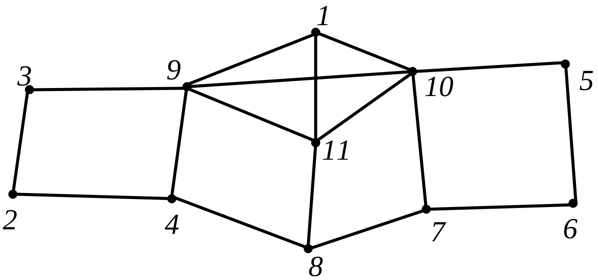
1 Recursively Decomposable Graph

2 Clique Separator Decompostion

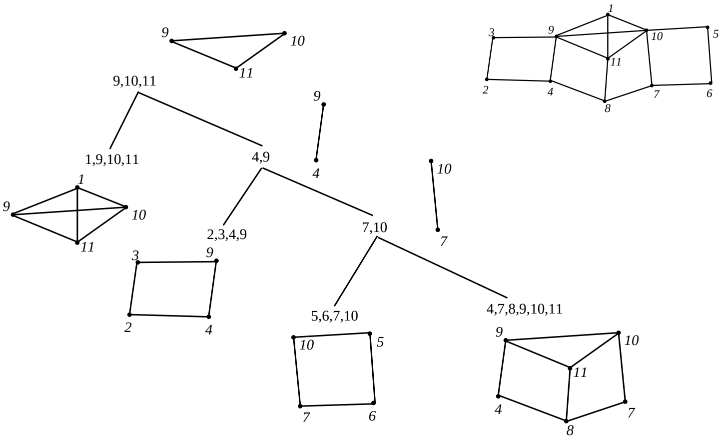
Clique Separator Decompostion

A *clique separator* is a clique C such that the subgraph induced by $G - C$ is disconnected.
The clique separator decomposition takes any clique separator C of G, splits G into subgraphs consisting of C plus each connected component of $G - C$, and decomposes each subgraph recursively.

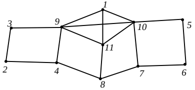
Clique Separator Decompostion



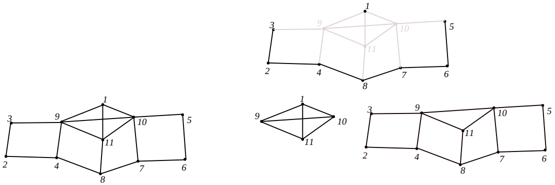
Clique Separator Decompostion



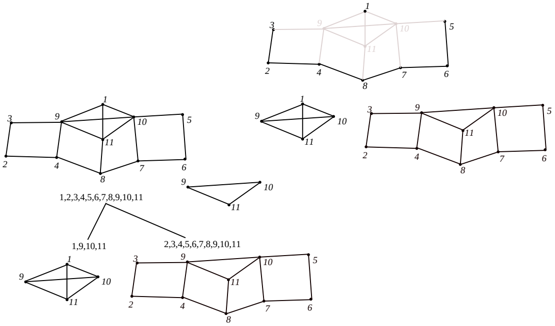
Clique Separator Decompostion



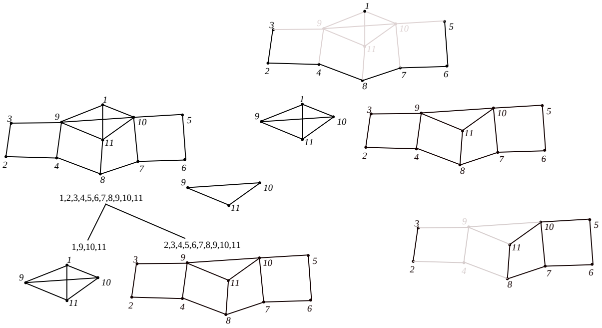
Clique Separator Decompostion



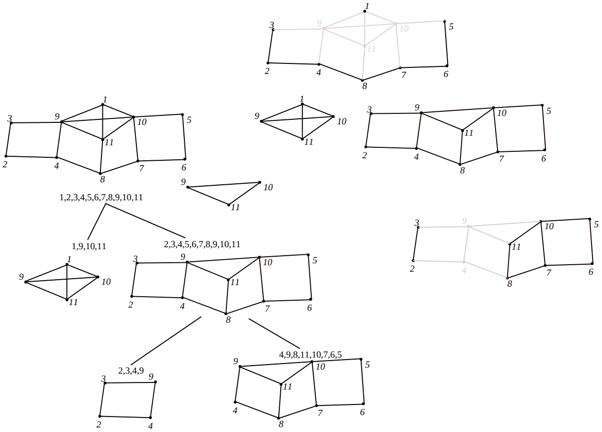
Clique Separator Decompostion



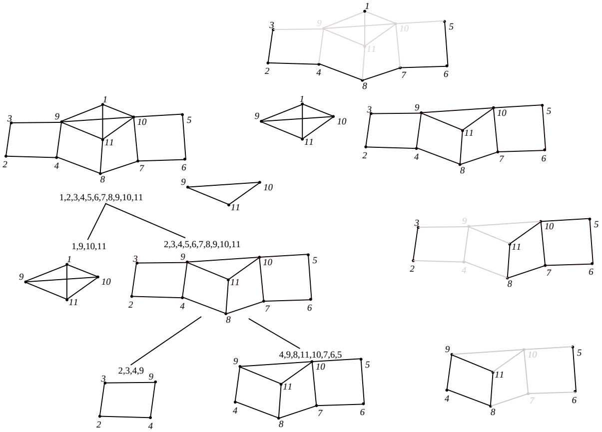
Clique Separator Decompostion



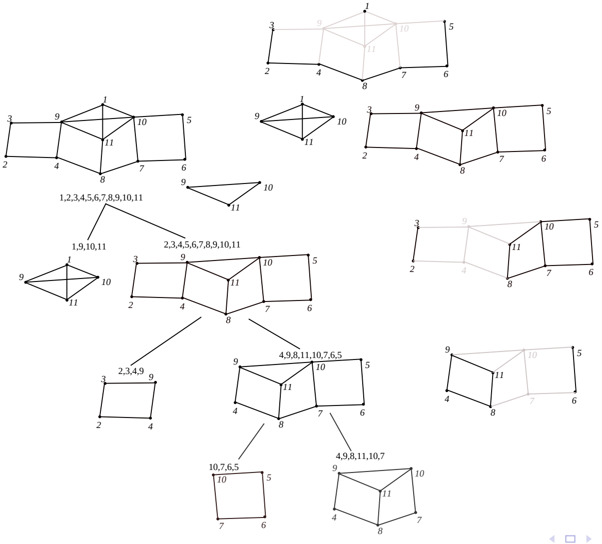
Clique Separator Decompostion



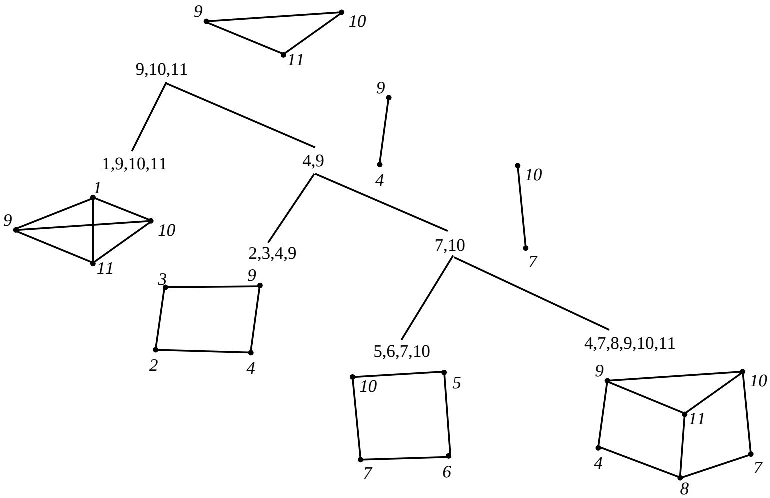
Clique Separator Decompostion



Clique Separator Decompostion



Clique Separator Decompostion



Clique Separator Decompostion

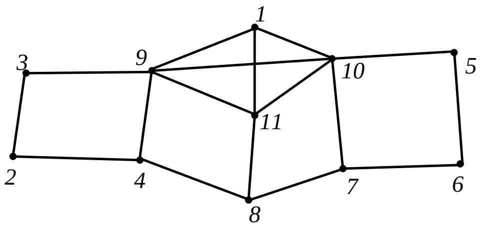
Fill Edges Technique

Given an ordering of the vertices, one computes fill by stepping through vertices in order, and adding edges between all neighbors of the current vertex which occure after the vertex in the ordering.

The problem of minimum fill asks for an ordering such that the fill created by this ordering is minimum.

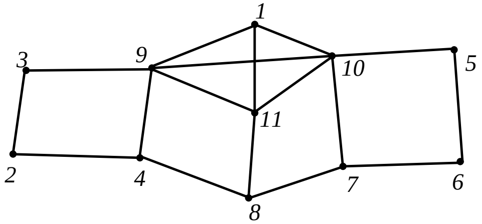
The problem of minimal fill asks for an ordering such that the fill created by this ordering does not properly contain the fill created by any other ordering.

Clique Separator Decompostion

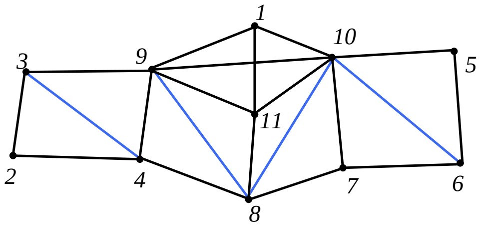


Ordering: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11

Clique Separator Decompostion



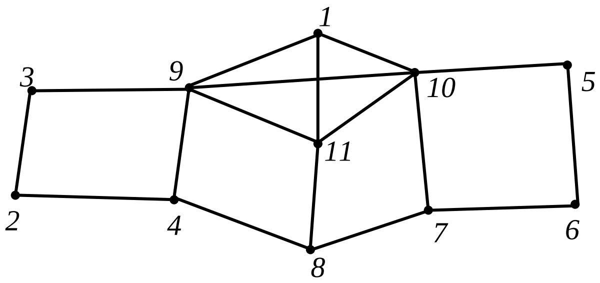
Ordering: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11



Algorithm for finding an ordering for minimal fill

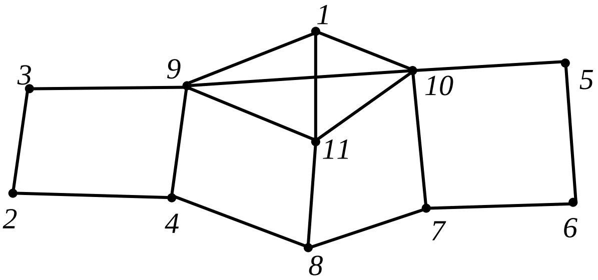
- Start with all vertices in a single set. Repeatedly remove a vertex v from the last set, and place it in the output list.
- Let knownfilled(v) be the set of vertices w such that there is a path from v to w in G using only intermediate vertices which come from sets which preced w in the current ordering; note that all neighbors of v are included in this group.
- Divide each of the current set S into $S - knownfilled(v)$ and $knownfilled(v) \cap S$. and place the former set before the latter.

Algorithm for finding an ordering for minimal fill



1,2,3,4,5,6,7,8,9,10,11

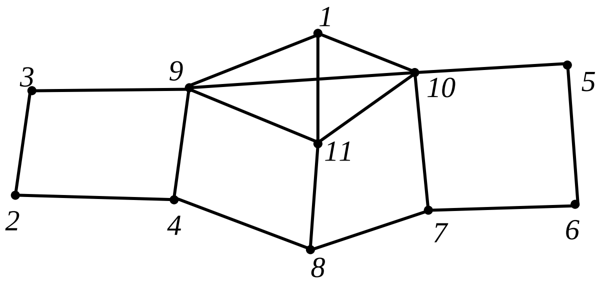
Algorithm for finding an ordering for minimal fill



1,2,3,4,5,6,7,8,9,10,11

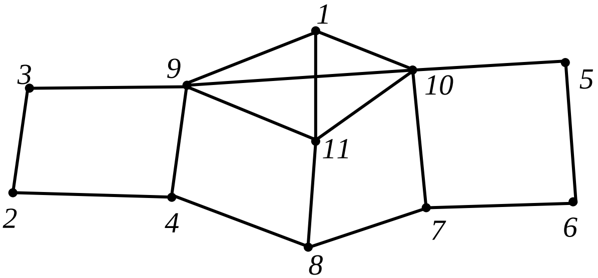
(2,3,4,5,6,7)(1,8,9,10) 11

Algorithm for finding an ordering for minimal fill



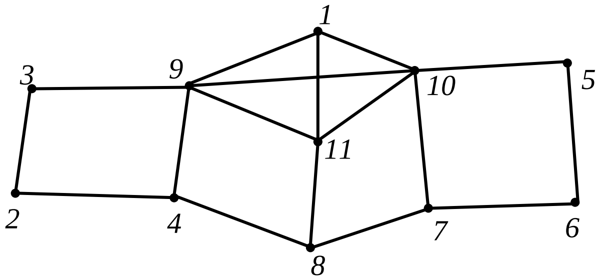
1,2,3,4,5,6,7,8,9,10,11
(2,3,4,5,6,7)(1,8,9,10) 11
(2,3,4,6)(5,7)(1,8,9) 10 11

Algorithm for finding an ordering for minimal fill



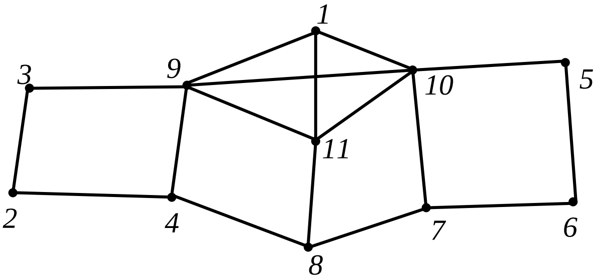
1,2,3,4,5,6,7,8,9,10,11
(2,3,4,5,6,7)(1,8,9,10) 11
(2,3,4,6)(5,7)(1,8,9) 10 11
(2,6)(3,4)(5,7)(1,8) 9 10 11

Algorithm for finding an ordering for minimal fill



1,2,3,4,5,6,7,8,9,10,11
(2,3,4,5,6,7)(1,8,9,10) 11
(2,3,4,6)(5,7)(1,8,9) 10 11
(2,6)(3,4)(5,7)(1,8) 9 10 11
(2,6)(3)(4)(5)(7)(1) 8 9 10 11

Algorithm for finding an ordering for minimal fill



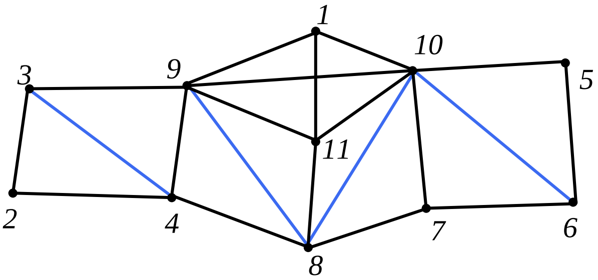
1,2,3,4,5,6,7,8,9,10,11
(2,3,4,5,6,7)(1,8,9,10) 11
(2,3,4,6)(5,7)(1,8,9) 10 11
(2,6)(3,4)(5,7)(1,8) 9 10 11
(2,6)(3)(4)(5)(7)(1) 8 9 10 11
2 6 3 4 5 7 1 8 9 10 11

Finding Clique Separators

- Let L be the minimal fill ordering
- For Each v, let $C(v)$ be the set of vertices w such that w comes after v in L, and (v, w) is either an edge in G or part of the fill caused by this ordering.
- Let A be the connected component of the graph induced by $v - C(V)$ which contains v, and let $B = V - (C(v) \cup A)$.
- If $C(v)$ is a clique in G and B is nonempty, then we have found a successful decomposition into $A \cup C(v)$ and $B \cup C(v)$.

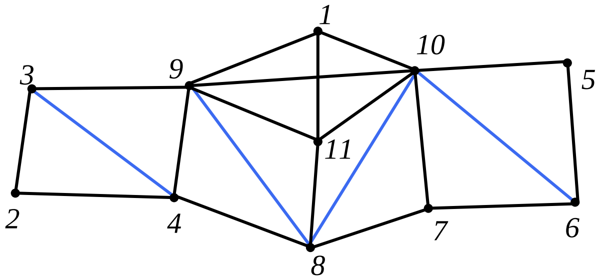
If there is a clique separator of G, some clique seperator is of the form $C(v)$ [Tarjan] $O(nm)$ time algorithm for finding a clique separator.

Finding Cliques



Ordering: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11

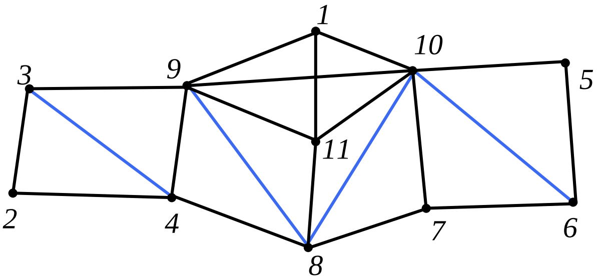
Finding Cliques



Ordering: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11

- $C(1) = \{9, 10, 11\}$ is a clique.

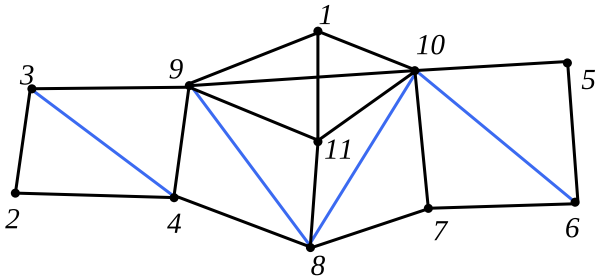
Finding Cliques



Ordering: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11

- $C(1) = \{9, 10, 11\}$ is a clique.
- $C(2) = \{3, 4\}$ is not a clique.

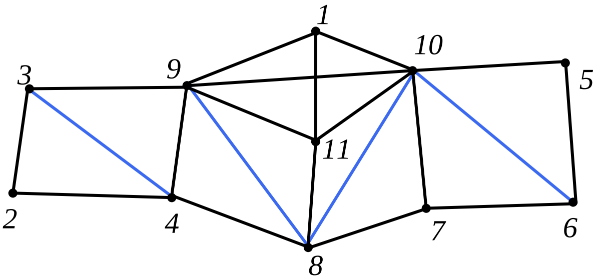
Finding Cliques



Ordering: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11

- $C(1) = \{9, 10, 11\}$ is a clique.
- $C(2) = \{3, 4\}$ is not a clique.
- $C(3) = \{4, 9\}$ is a clique; separates 3,2 from the rest of the component.

Finding Cliques



Ordering: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11

- $C(1) = \{9, 10, 11\}$ is a clique.
- $C(2) = \{3, 4\}$ is not a clique.
- $C(3) = \{4, 9\}$ is a clique; separates 3,2 from the rest of the component.
- $C(6) = \{7, 10\}$ is a clique.

Application of Clique Separator Decompostion

Clique separator decomposition is used to devise devide and conqure technique to deal with the following problems.

- Minimal and minimum triangulation
- Computing treewidth
- Coloring
- Maximum clique