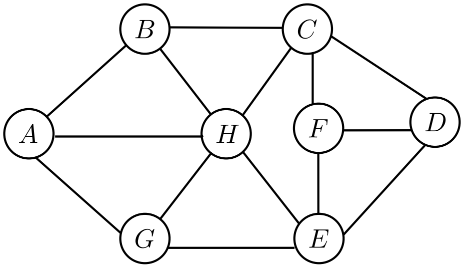
Efficient Algorithm for "Ear Decomposition"

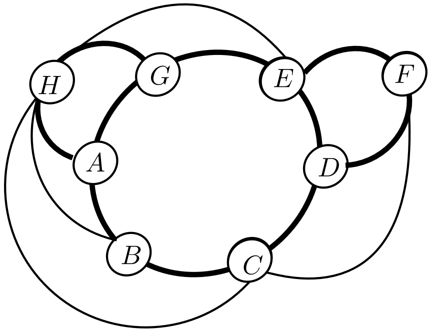
Debarshi Dutta, Kishore Kothaplli

IIIT Hyderabad

G.Ramakrishna, Sai Charan.R and Sai Harsh.T

IIIT Chittoor, Sri City

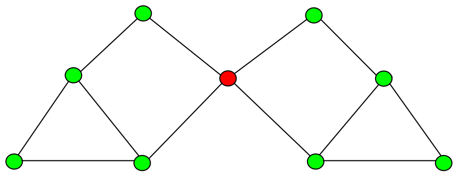




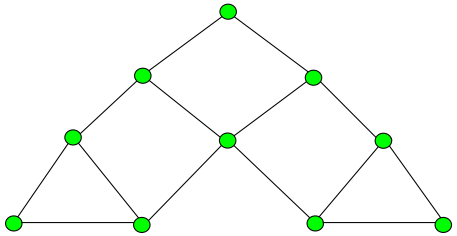
Overview

- Terminology
- Ear Decomposition Problem
- Past Work
- Our Contribution
- Experiments

Bi-Connected Graph



1-Node Connected Graph



2-Node Connected Graph

(Bi-Connected Graph)

Ear Decomposition Definition

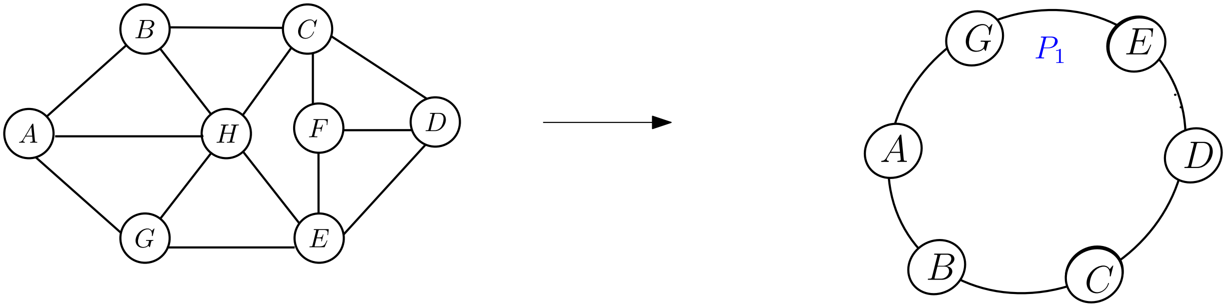
An ear decomposition of a graph G = (V, E) is partition of E into a sequence $(P_0, P_1, , P_k)$ such that

(i) P_0 is a cycle

(ii) for each $i \geq 1$, P_i is an ear of $P_0 \cup ... \cup P_i$.

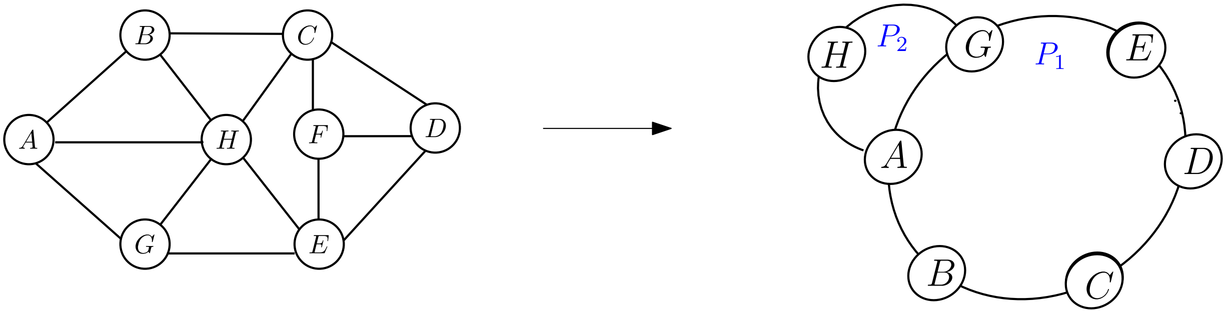
(iii) end points of all the ears P_i , $i \geq 1$, are distinct.

Ear Decomposition



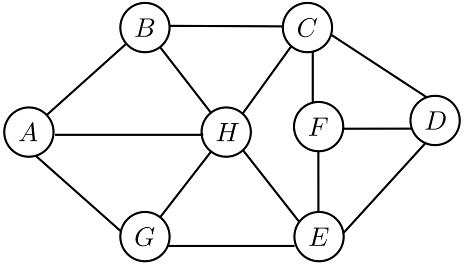
(P_1,

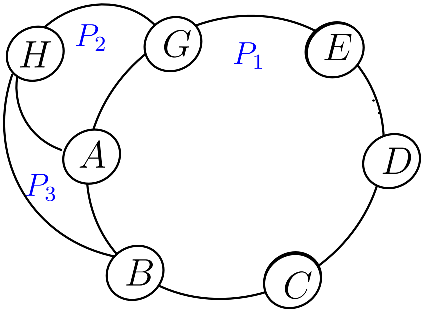
Ear Decomposition



$(P_1, P_2,$

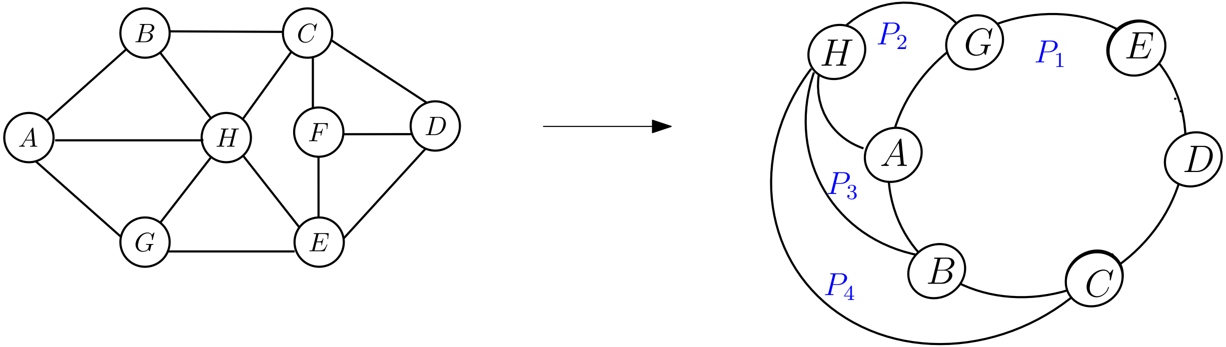
Ear Decomposition





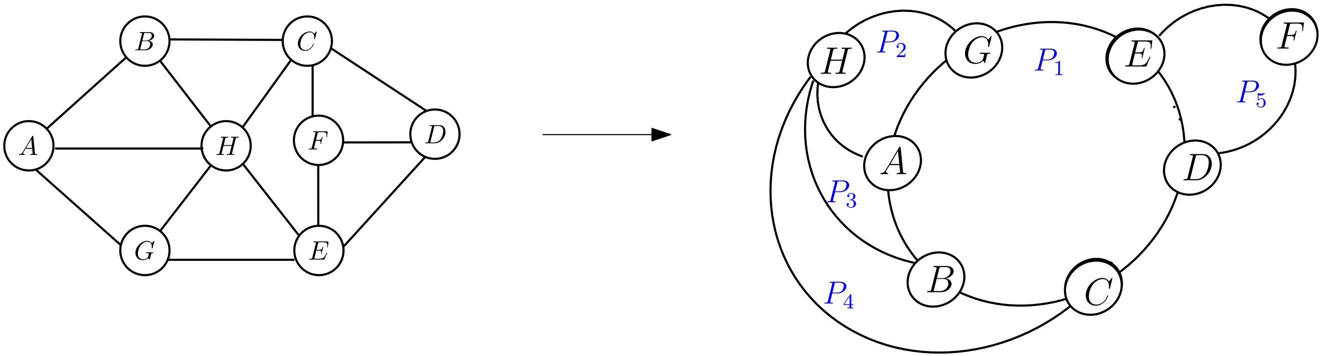
$(P_1,\ P_2,\ P_3,$

Ear Decomposition



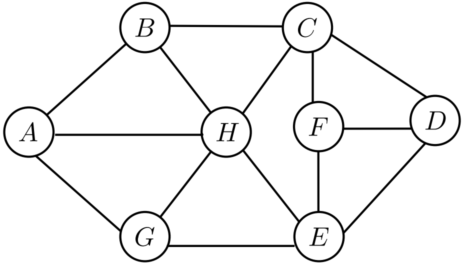
(P_1, P_2, P_3, P_4,

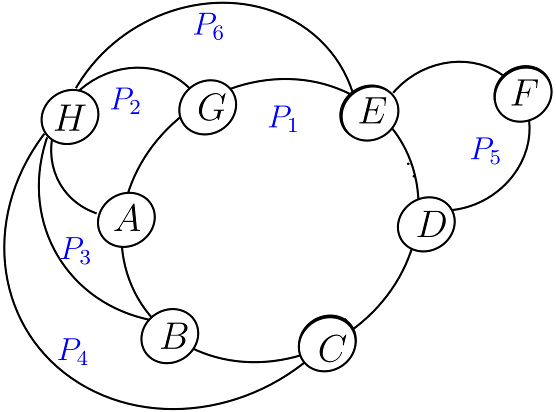
Ear Decomposition



$(\ P_1, \ P_2, \ P_3, \ P_4, \ P_5,$

Ear Decomposition





$(\ P_1,\ P_2,\ P_3,\ P_4,\ P_5,\ P_6\ )$

Ear Decomposition Problem

Input: An unweighted bi-connected Graph G

Question: Obtain Ear Decomposition of G ?

Efficiently

Past Work

Lovász's Algorithm

L. Lovász. Computing ears and branchings in parallel. *Found. of Comp. Sci*, 464-467, 1985.

Schmidt's Algorithm

J. M. Schmidt. A simple test on 2-vertex- and 2-edge-connectivity. *Info. Proc. Lett.*, 113, 241-244, 2013.

Our Contribution

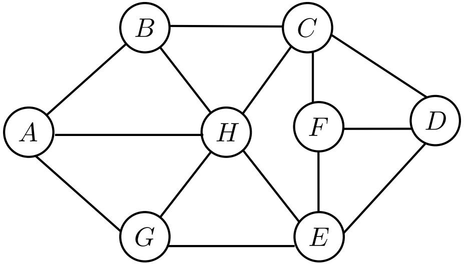
In Theory

An Algorithm that is Asymptotically equivalent to Schmidt's Algorithm

In Practice

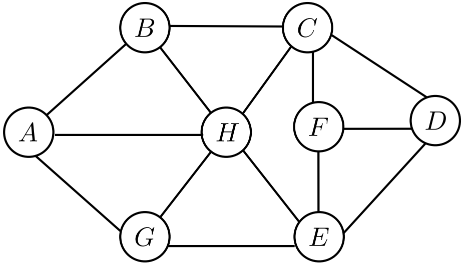
An Algorithm with 2X SpeedUp w.r.t Schmidt's Algorithm

Schmidt's Algorithm

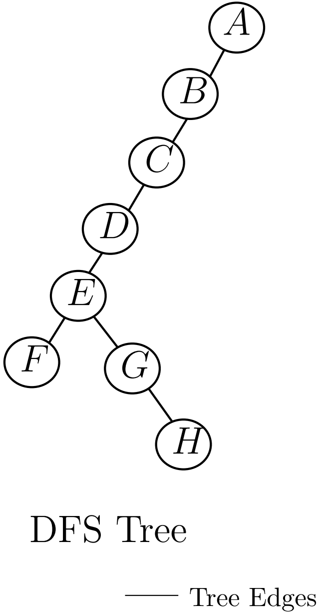


Graph

Schmidt's Algorithm

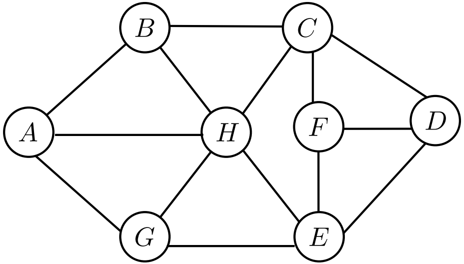


Graph

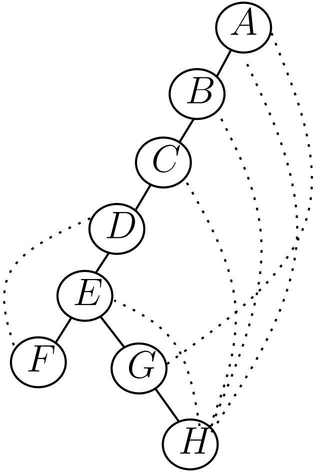


DFS Tree

Schmidt's Algorithm



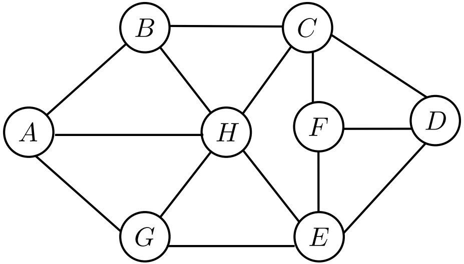
Graph



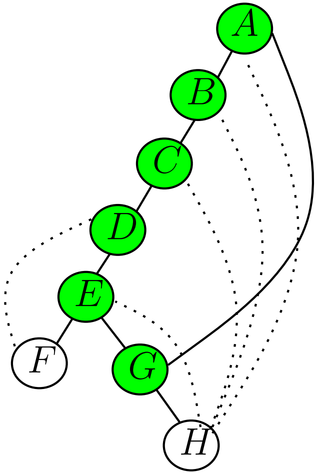
DFS Tree

Schmidt's Algorithm

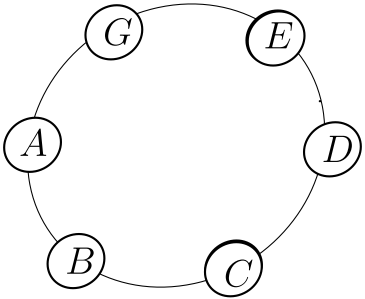
(P_1,



Graph



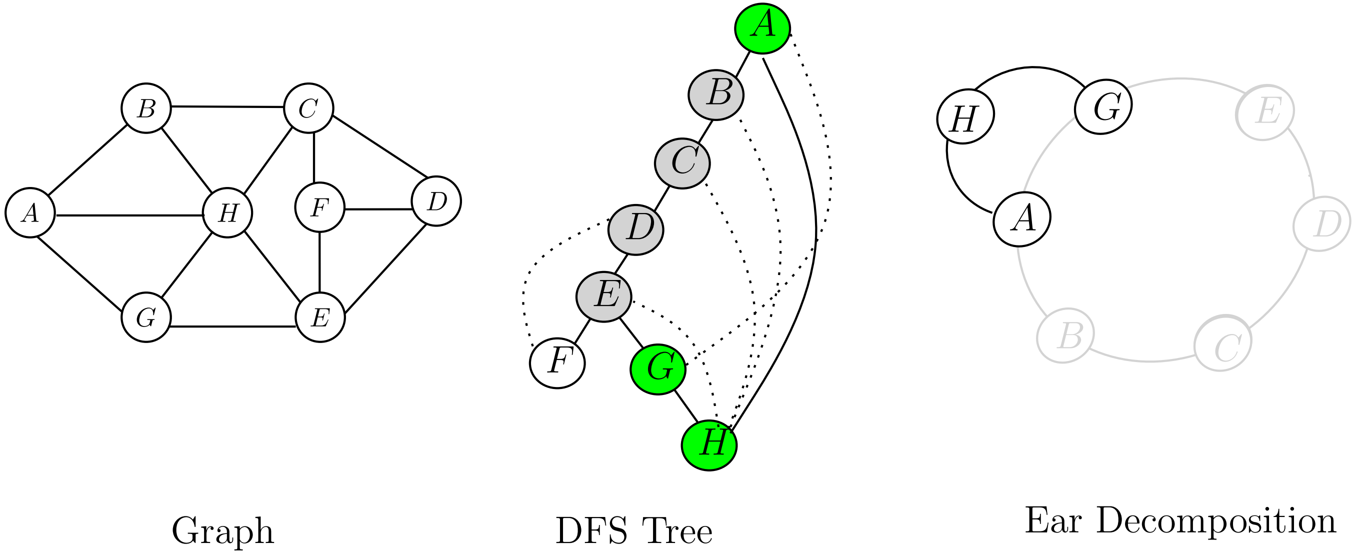
DFS Tree



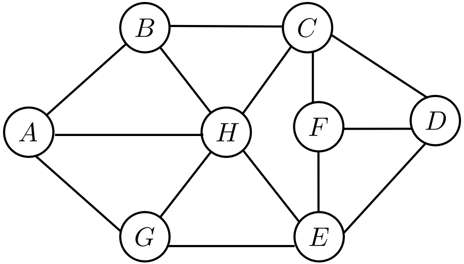
Ear Decomposition

Schmidt's Algorithm

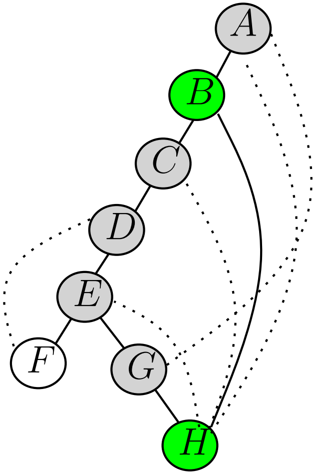
(P_1, P_2,



Schmidt's Algorithm

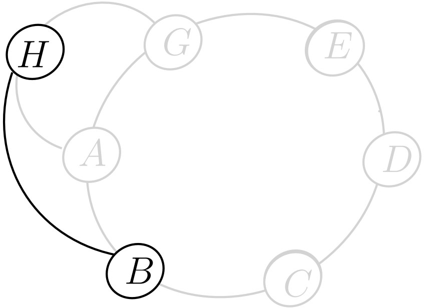


Graph



DFS Tree

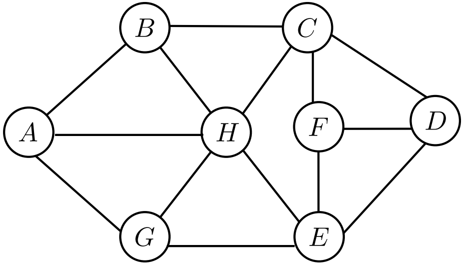
(P_1, P_2, P_3,



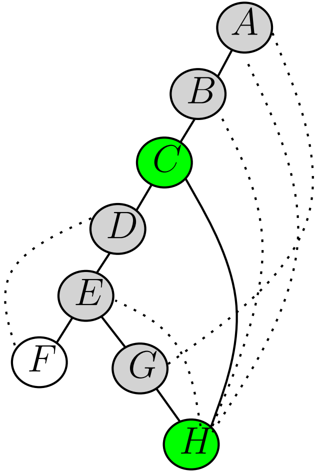
Ear Decomposition

Schmidt's Algorithm

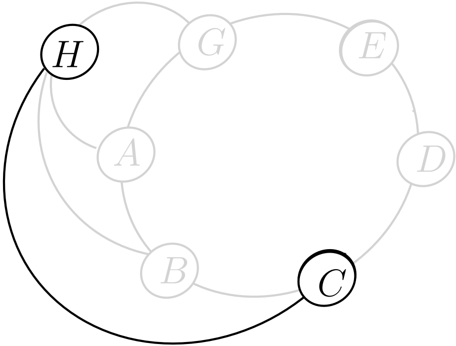
(P_1, P_2, P_3, P_4,



Graph



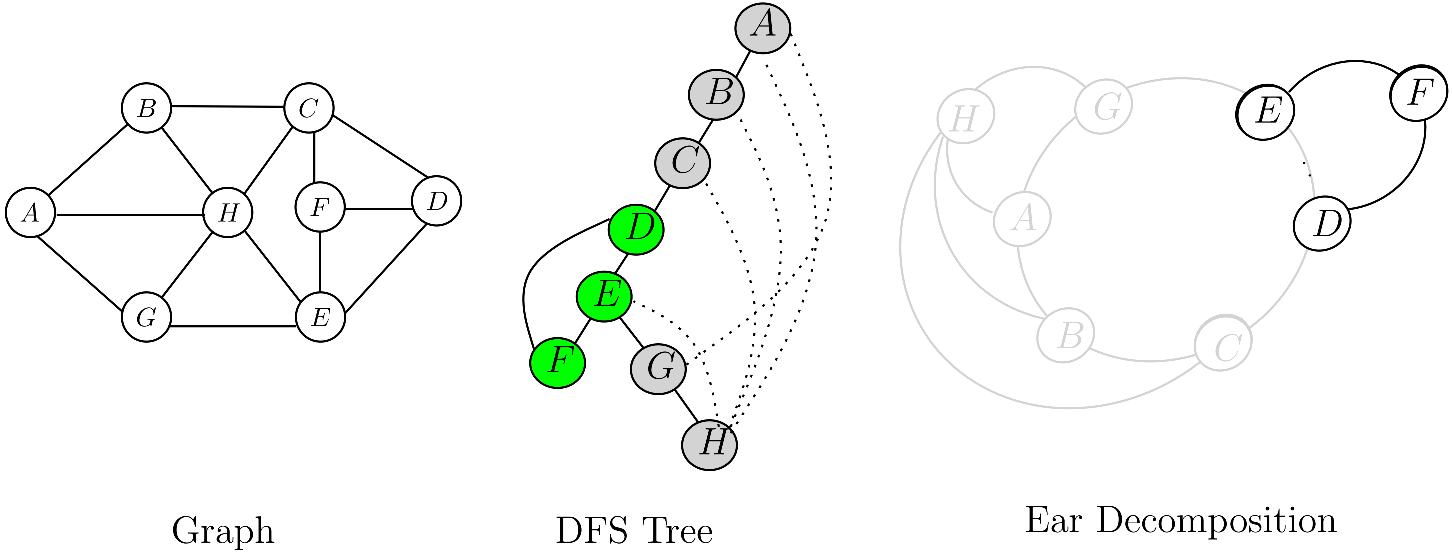
DFS Tree



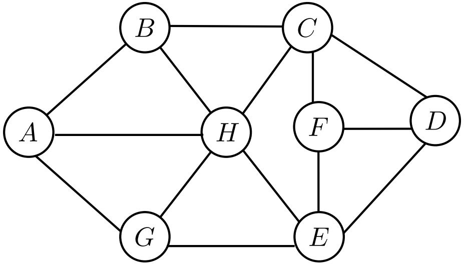
Ear Decomposition

Schmidt's Algorithm

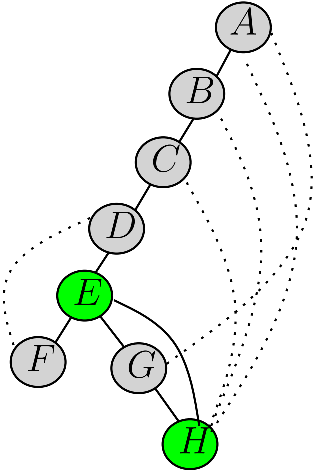
$(P_1, P_2, P_3, P_4, P_5,$



Schmidt's Algorithm

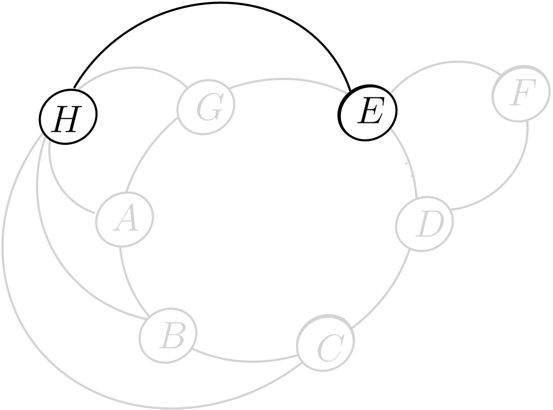


Graph



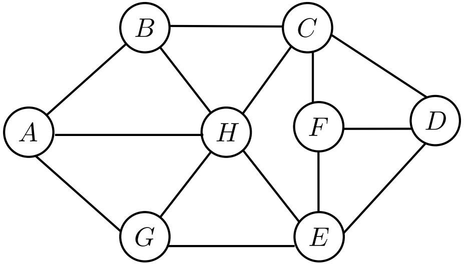
DFS Tree

$(P_1, P_2, P_3, P_4, P_5, P_6)$

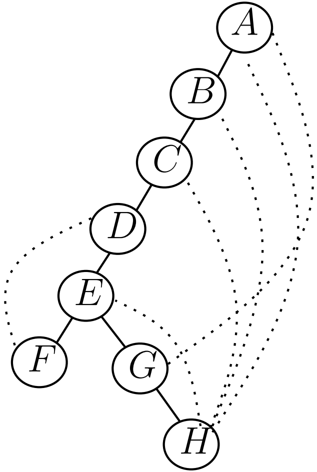


Ear Decomposition

Schmidt's Algorithm

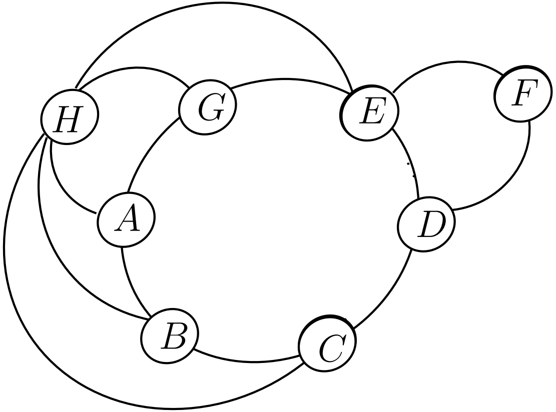


Graph



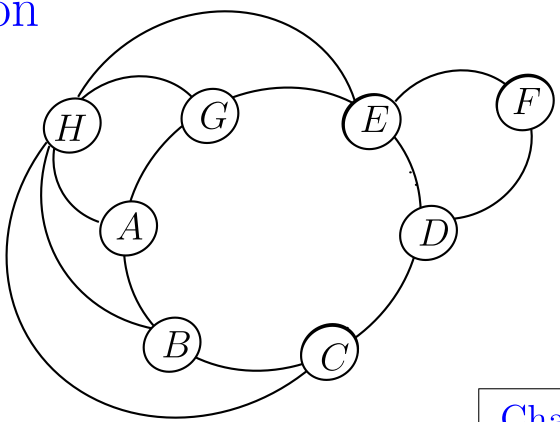
DFS Tree

$(P_1, P_2, P_3, P_4, P_5, P_6)$



Ear Decomposition

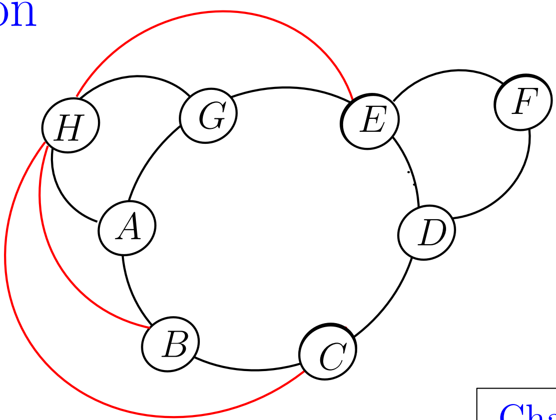
Intusion



Ear Decomposition

Chain	Length
P_1	6
P_2	2
P_3	1
P_4	1
P_5	2
P_6	1

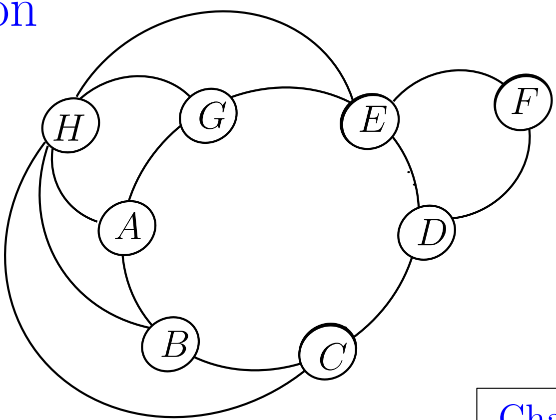
Intusion



Ear Decomposition

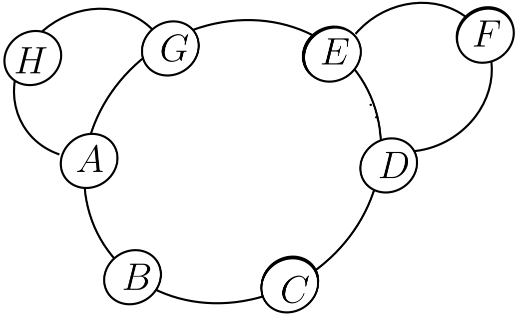
Chain	Length
P_1	6
P_2	2
P_3	1
P_4	1
P_5	2
P_6	1

Intusion



Ear Decomposition
with Trivial Ears

Chain	Length
P_1	6
P_2	2
P_3	1
P_4	1
P_5	2
P_6	1



Ear Decomposition
without Trivial Ears

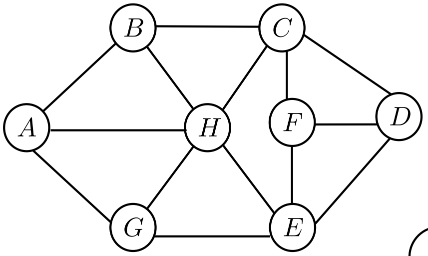
Lemma

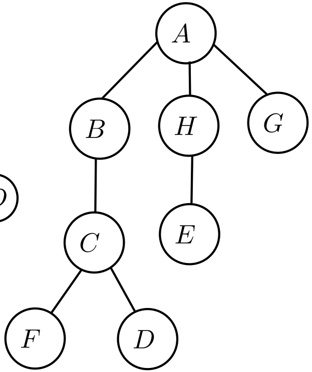
$G : Graph$
$T : Spanning\ Tree\ of\ G$
$F : Spanning\ Forest\ of\ G - T$

$$|BiconnectedComponents(G)| \ = \ |BiconnectedComponents(T \cup F)|$$

G. Cong and D. A. Bader. An experimental study of parallel biconnected components algorithms on symmetric multiprocessors. *Inter. Par. and Dist. Proc. Symp*, 2005.

Key Approach





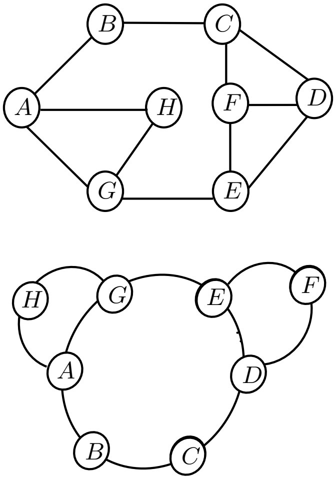
BFS Tree

$(T$ From $G)$

Construct a Spanning Forest

$(F$ From $G - T)$

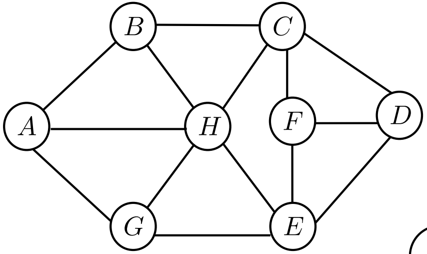
$EarDecomposition(T \cup F)$

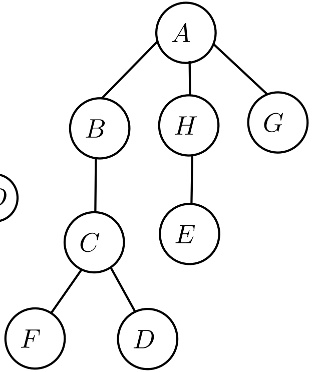


Experiment-1

Breadth First or
Depth First Traversals

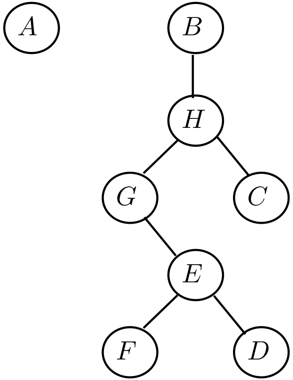
$EarDecomposition(T \cup F)$





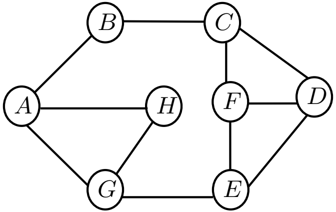
BFS Tree

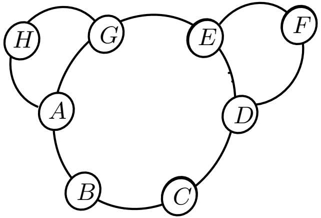
(T From G)



Spanning Forest

(F From $G - T$)





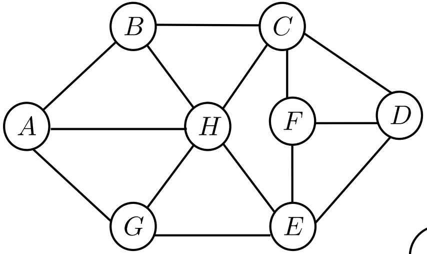
$\boxed{4m + O(n)}$

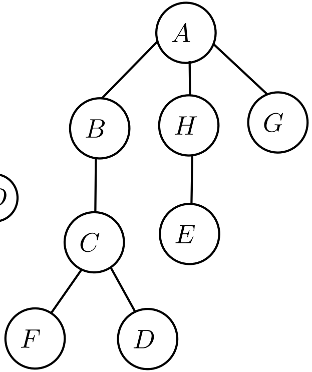
$2m + O(n)$

$2m + O(n)$

$O(n)$

Experiment-2



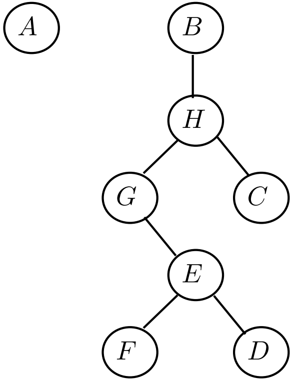


BFS Tree

(T From G)

$disjoint - set - forest$

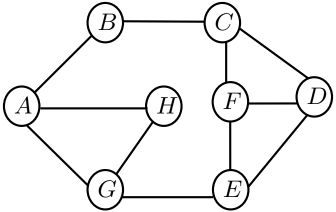
union-by-rank and path-compression

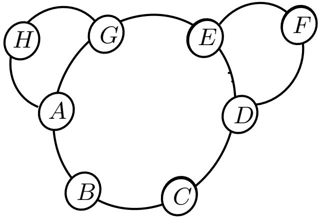


Spanning Forest

(F From $G - T$)

$EarDecomposition(T \cup F)$





$\boxed{2m + \alpha(n)m + O(n)}$

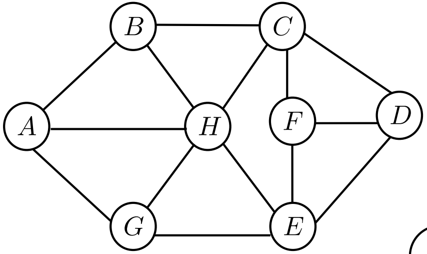
$2m + O(n)$

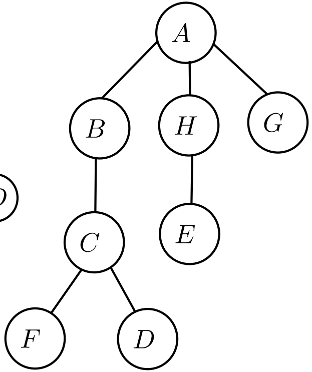
$m\alpha(n) + O(n)$

$O(n)$

Experiment-3

Final Approach





BFS Tree

$(T \text{ From } G)$

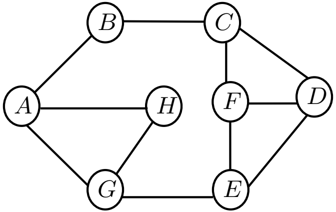
Random Sampling

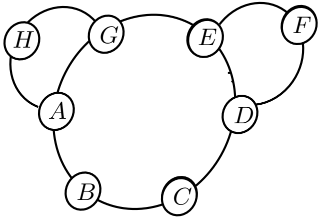
Random Sampling

Spanning Forest

$(F \text{ From } G - T)$

$Ear Decomposition(T \cup F)$





$\boxed{2m + O(n \log n)}$ $2m + O(n)$ $O(n \log n)$ $O(n)$

Random Sampling

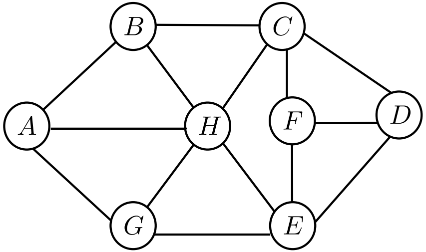
Theorem

$$n : number\ of\ vertices$$
$$p : \frac{\ln n}{n}$$

$A\ random\ graph\ G(n,p)\ is\ connected$
with very high probability

P. Erds and A Rnyi. On the evolution of random graphs. *Publ. Math. Inst. Hungar. Acad. Sci*, *pages* 17-61, 1960.

Random Sampling



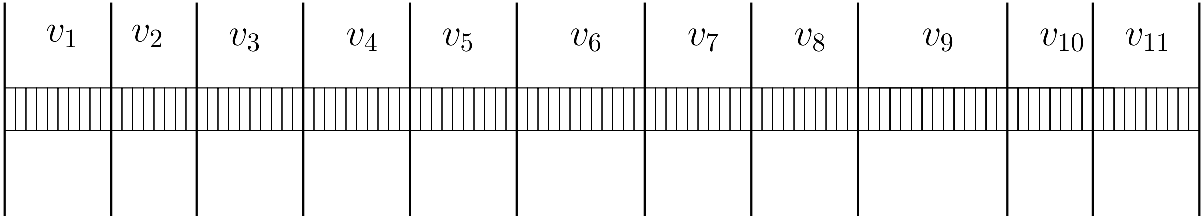
Graph (G)

CSR Format of a Graph (G)

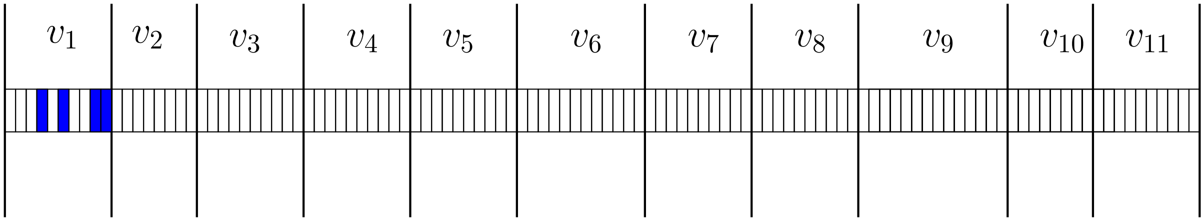
A	B	C	D	E	F	G	H
B G H	A C H	B D F H	C E F	D F G H	C D E	A E H	A B C E G

Random Sampling

CSR Format of a Dense Graph



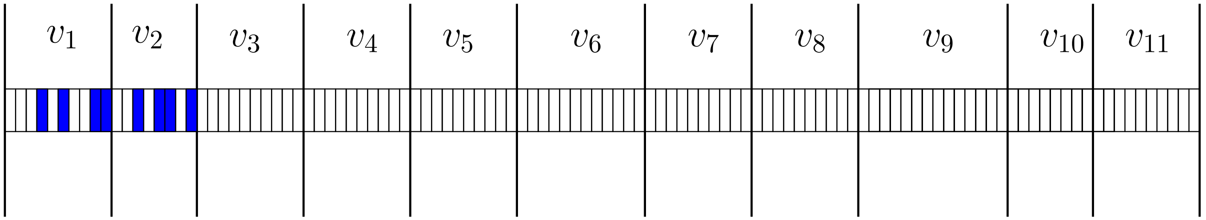
Random Sampling



Here $\log n = 4$

Moving $\log n$ edges of each vertex from G to $T \cup F$

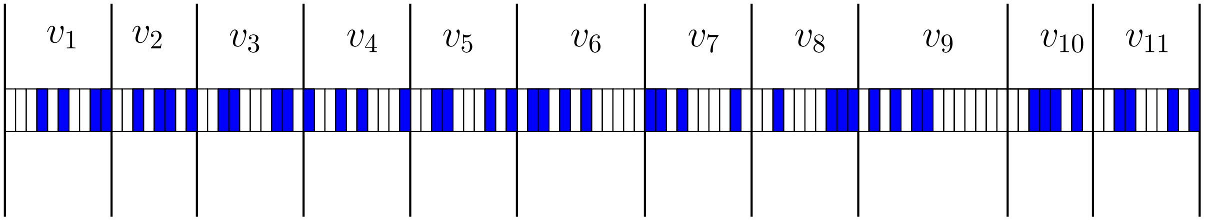
Random Sampling



Here $\log n = 4$

Moving $\log n$ edges of each vertex from G to $T \cup F$

Random Sampling

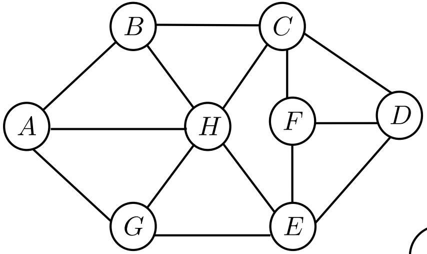


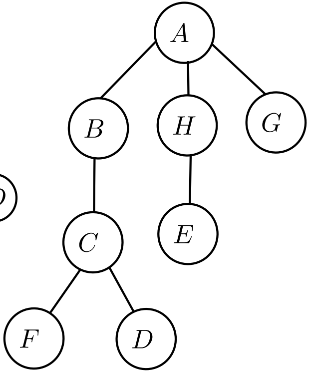
Here $\log n = 4$

Moving $\log n$ edges of each vertex from G to $T \cup F$

Experiment-3

Final Approach





BFS Tree

(T From G)

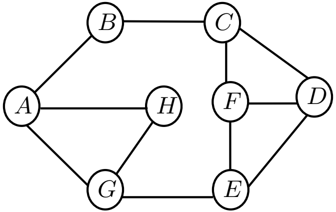
Random Sampling

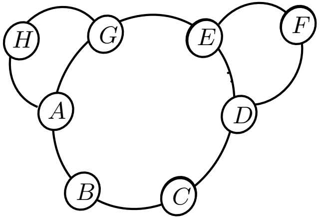
Random Sampling

Spanning Forest

(F From $G - T$)

$EarDecomposition(T \cup F)$



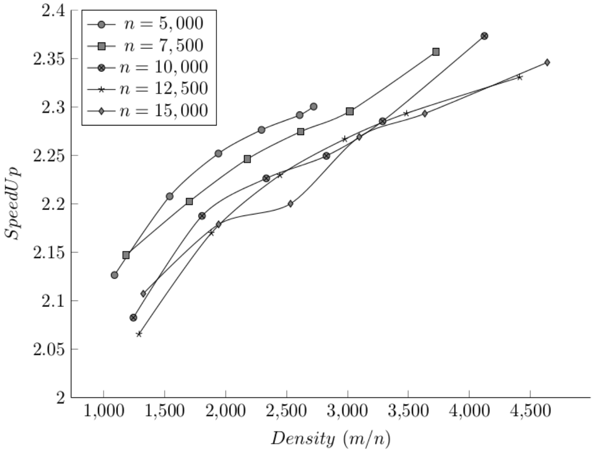


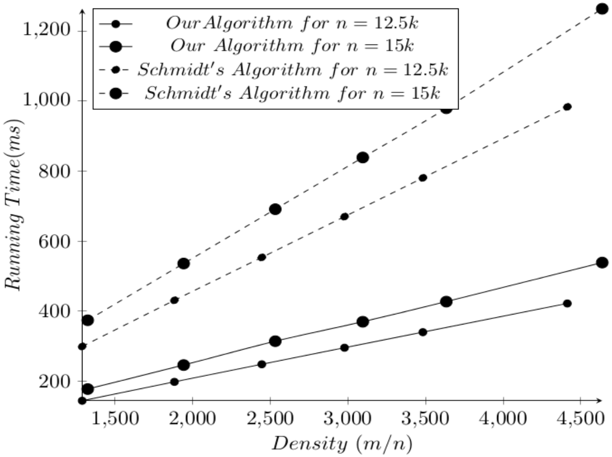
$\boxed{2m + O(n \log n)}$ $\quad 2m + O(n) \quad O(n \log n) \quad O(n)$

Running Time Comparision

$Schmidt's\ Algorithm:$ 4m+O(n)				
	BFS (T)	Spanning Forest (F)	$Ear Decomposition (T \cup F)$	Total
Breadth First or Depth First Traversals	$2m + O(n)$	$2m + O(n)$	$O(n)$	$4m + O(n)$
$disjoint - set - forest$ union-by-rank and path compression	$2m + O(n)$	$m\alpha(n) + O(n)$	$O(n)$	$2m + \alpha(n)m + O(n)$
Random Sampling	$2m + O(n)$	$O(n \log n)$	$O(n)$	$2m + O(n \log n)$

Results



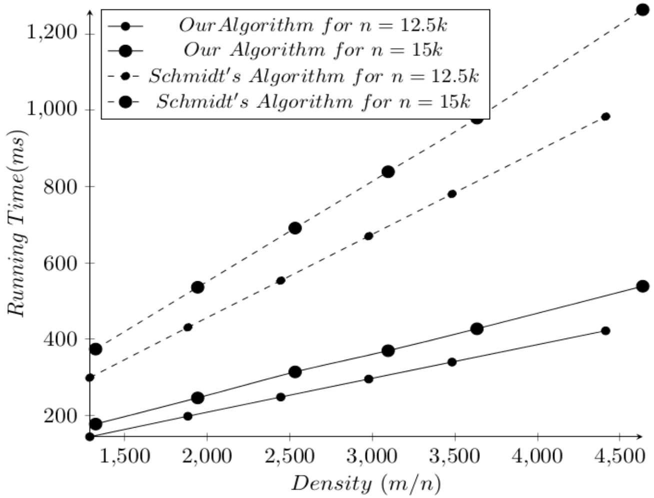


2X Seed Up

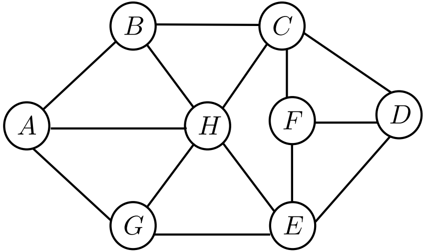
Thank You..!!

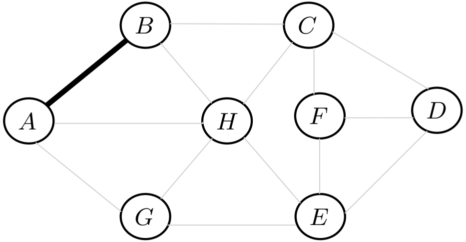
Questions?

Results

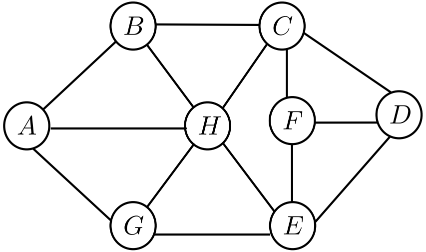


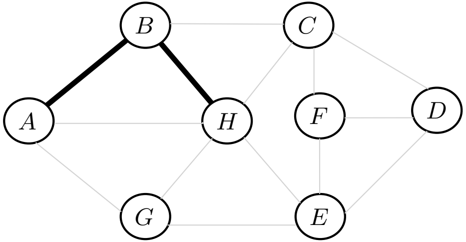
$Our Algorithm\ for\ n = 12.5k$
$Our\ Algorithm\ for\ n = 15k$
$Schmidt's\ Algorithm\ for\ n = 12.5k$
$Schmidt's\ Algorithm\ for\ n = 15k$
Running Time(ms)
1,200
1,000
800
600
400
200
1,500
2,000
2,500
3,000
3,500
4,000
4,500
Density (m/n)



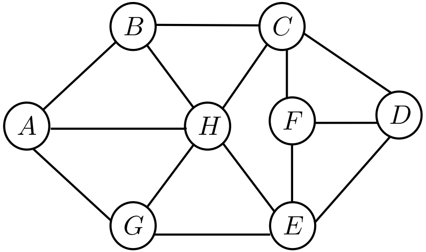


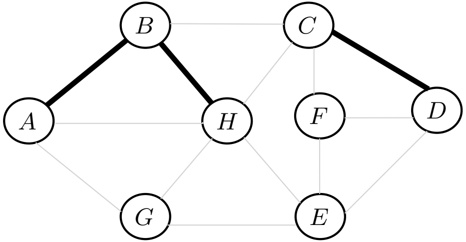
<table>
<tr><td colspan="3">A</td><td colspan="3">B</td><td colspan="4">C</td><td colspan="3">D</td><td colspan="4">E</td><td colspan="3">F</td><td colspan="3">G</td><td colspan="5">H</td></tr>
<tr><td>B</td><td>G</td><td>H</td><td>A</td><td>C</td><td>H</td><td>B</td><td>D</td><td>F</td><td>H</td><td>C</td><td>E</td><td>F</td><td>D</td><td>F</td><td>G</td><td>H</td><td>C</td><td>D</td><td>E</td><td>A</td><td>E</td><td>H</td><td>A</td><td>B</td><td>C</td><td>E</td><td>G</td></tr>
</table>



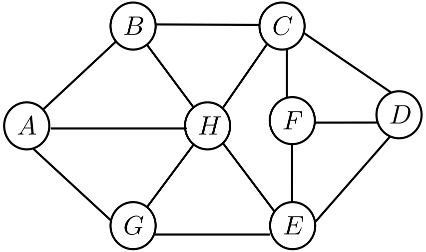


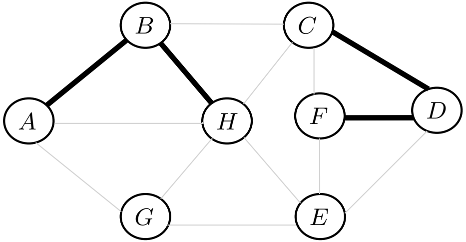
A			B			C				D			E				F			G			H				
B	G	H	A	C	H	B	D	F	H	C	E	F	D	F	G	H	C	D	E	A	E	H	A	B	C	E	G



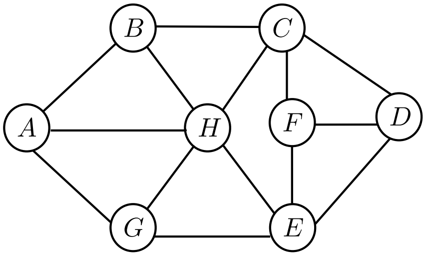


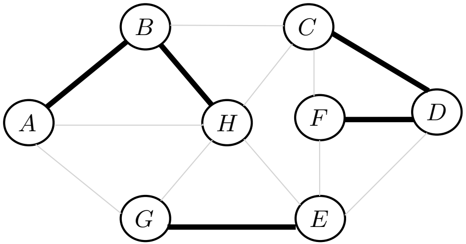
A	B	C	D	E	F	G	H
B G H	A C H	B D F H	C E F	D F G H	C D E	A E H	A B C E G



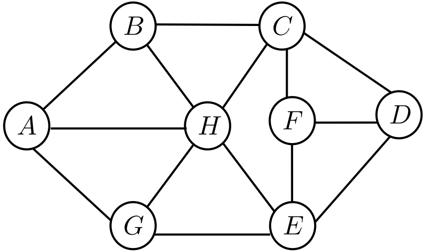


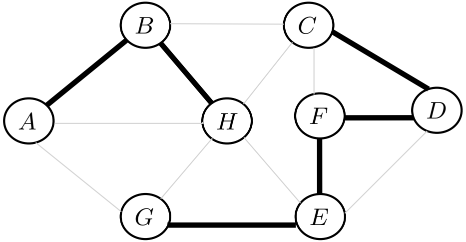
A	B	C	D	E	F	G	H
B G H	A C H	B D F H	C E F	D F G H	C D E	A E H	A B C E G



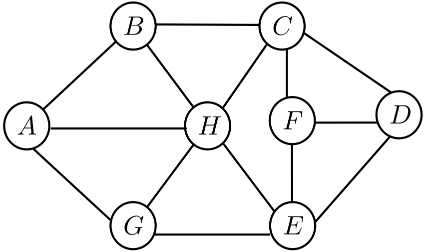


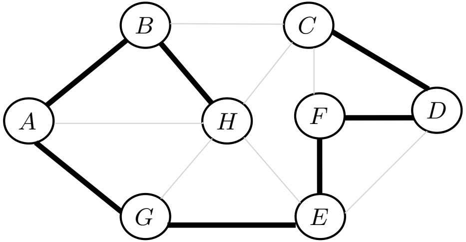
A			B			C				D			E				F			G			H				
B	G	H	A	C	H	B	D	F	H	C	E	F	D	F	G	H	C	D	E	A	E	H	A	B	C	E	G



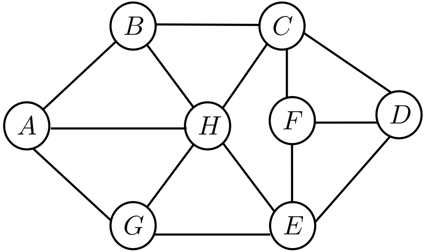


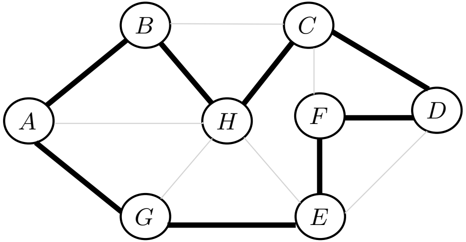
<table>
<tr><th colspan="3">A</th><th colspan="3">B</th><th colspan="4">C</th><th colspan="3">D</th><th colspan="4">E</th><th colspan="3">F</th><th colspan="3">G</th><th colspan="5">H</th></tr>
<tr><td>B</td><td>G</td><td>H</td><td>A</td><td>C</td><td>H</td><td>B</td><td>D</td><td>F</td><td>H</td><td>C</td><td>E</td><td>F</td><td>D</td><td>F</td><td>G</td><td>H</td><td>C</td><td>D</td><td>E</td><td>A</td><td>E</td><td>H</td><td>A</td><td>B</td><td>C</td><td>E</td><td>G</td></tr>
</table>



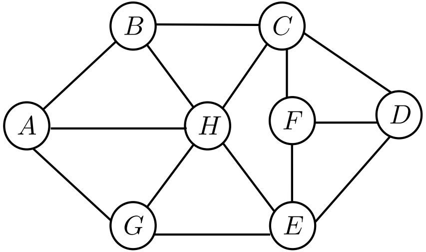


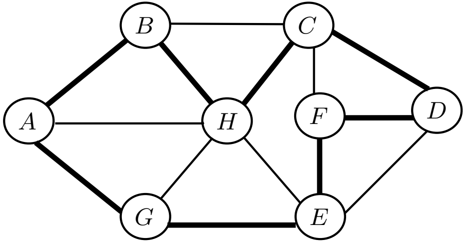
A	B	C	D	E	F	G	H
B G H	A C H	B D F H	C E F	D F G H	C D E	A E H	A B C E G



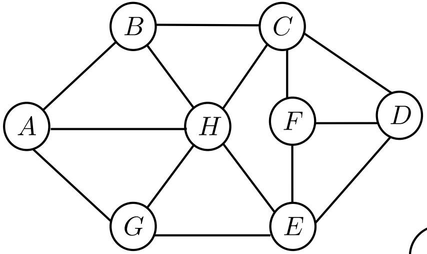


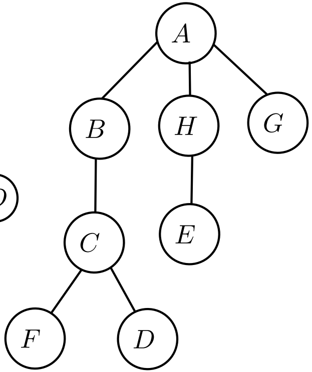
A	B	C	D	E	F	G	H
B G H	A C H	B D F H	C E F	D F G H	C D E	A E H	A B C E G





A			B			C				D			E				F			G			H				
B	G	H	A	C	H	B	D	F	H	C	E	F	D	F	G	H	C	D	E	A	E	H	A	B	C	E	G



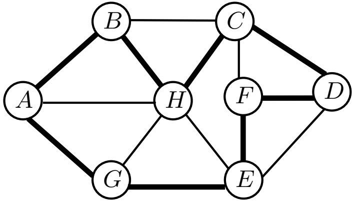


BFS Tree

(T From G)

$2m + O(n)$

Random Sampling

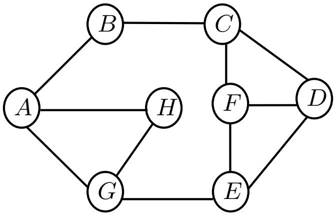


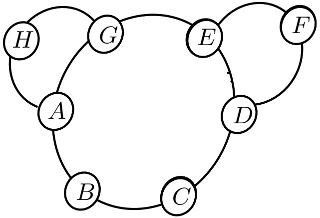
Spanning Forest

(F From $G - T$)

$m + O(n \log n)$

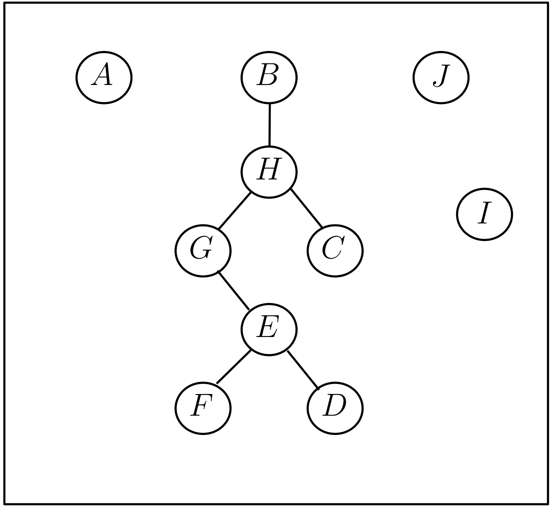
$EarDecomosition(T \cup F)$

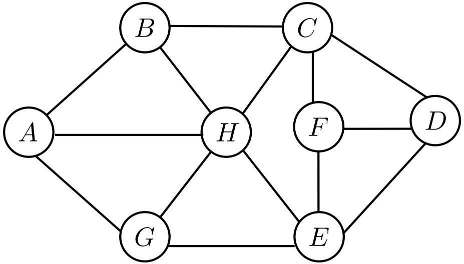




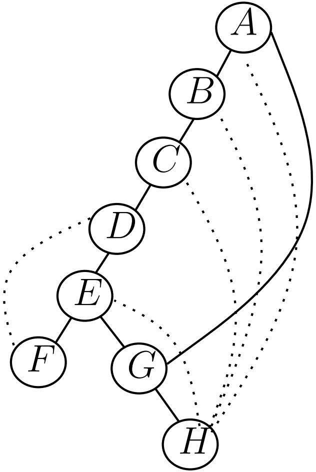
$O(n)$

TUF is connected graph with some isolated vertices.



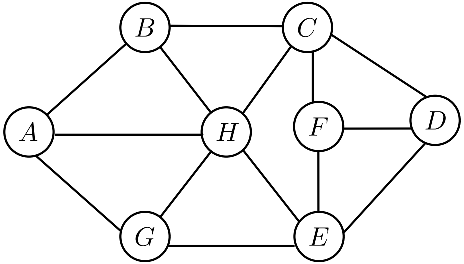


Graph

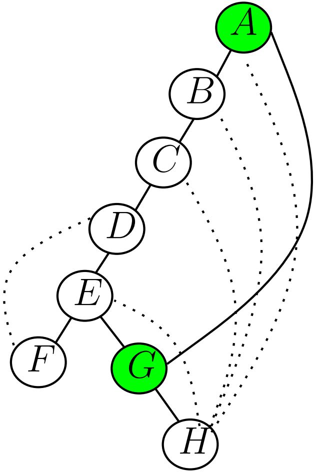


DFS Tree

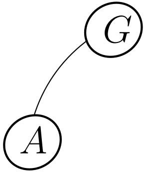
Decomposed Graph



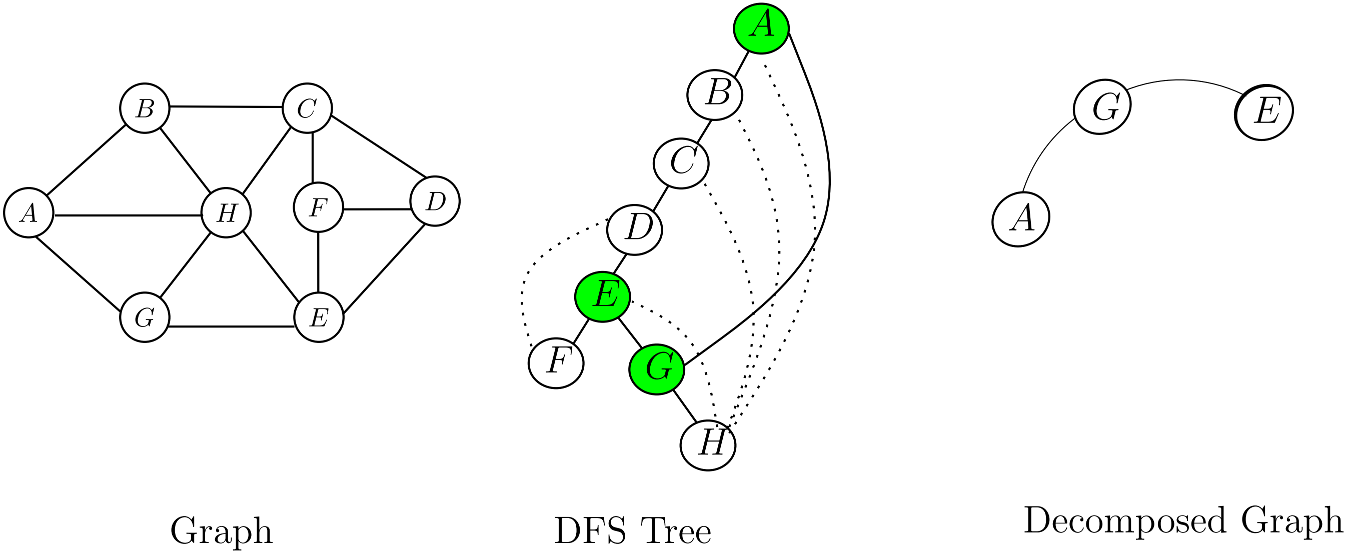
Graph

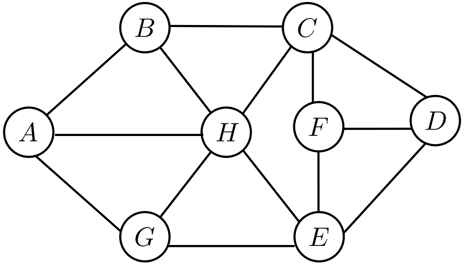


DFS Tree

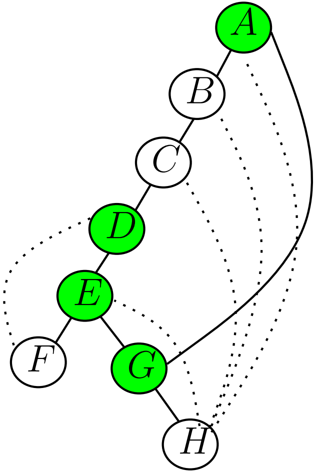


Decomposed Graph

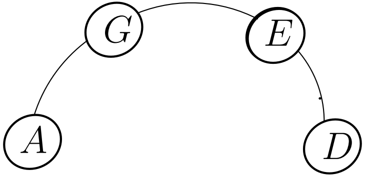

A
B
C
D
E
F
G
H
Graph
DFS Tree
Decomposed Graph



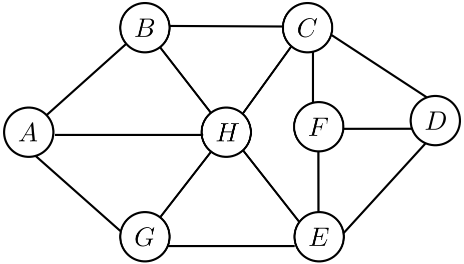
Graph



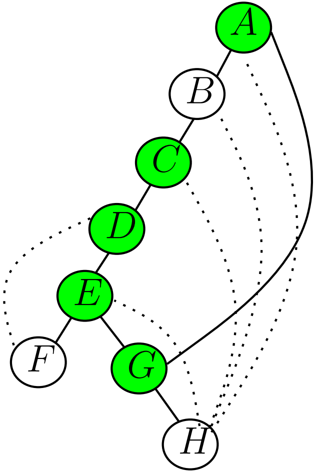
DFS Tree



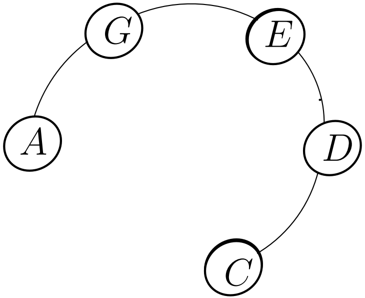
Decomposed Graph



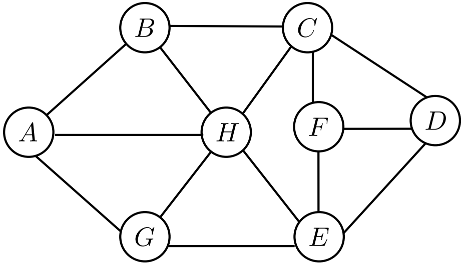
Graph



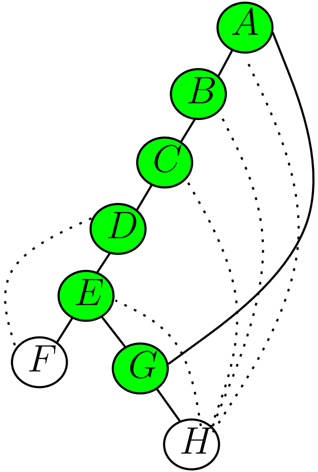
DFS Tree



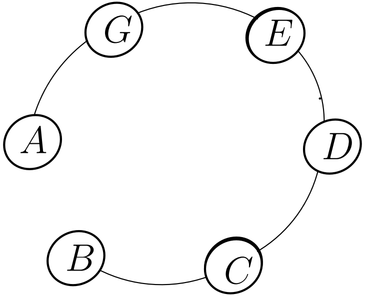
Decomposed Graph



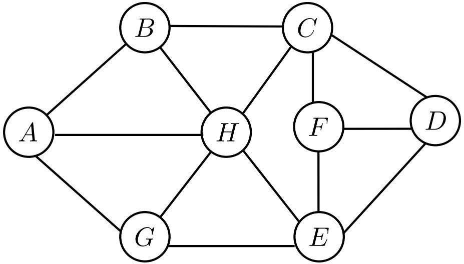
Graph



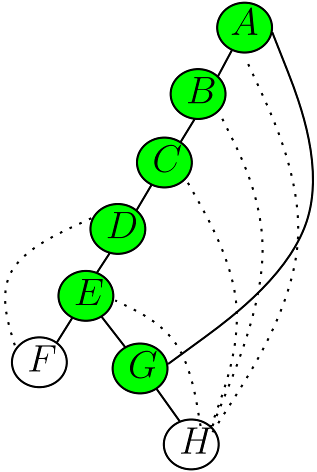
DFS Tree



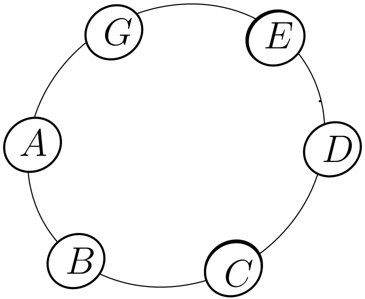
Decomposed Graph



Graph



DFS Tree



Decomposed Graph