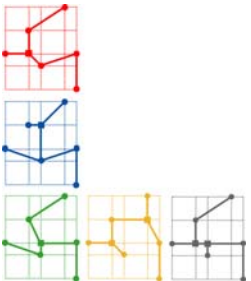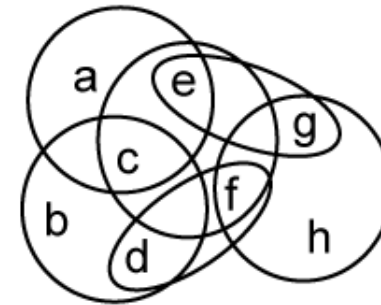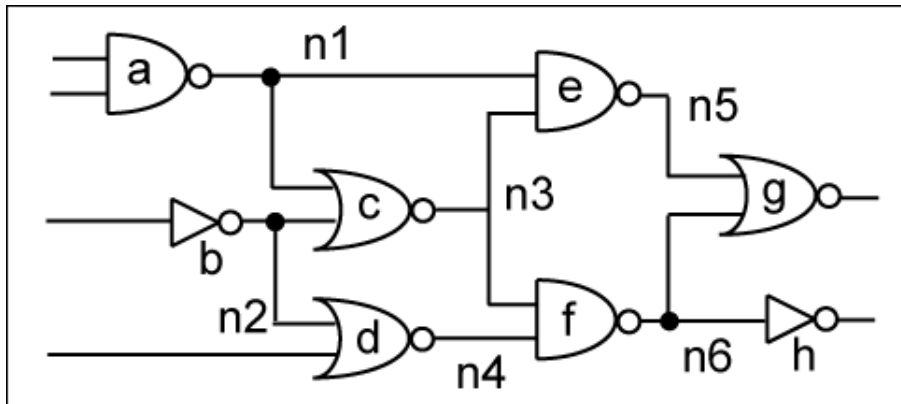# Multi-level Coarsening Algorithm

- Perform Edge Coarsening (EC)
  - Visit nodes and break ties in alphabetical order
  - Explicit clique-based graph model is not necessary
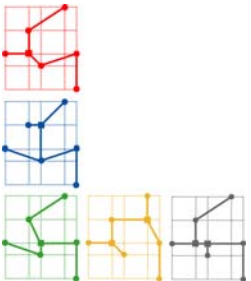
Edge Coarsening

(a) visit a: Note that a is contained in n_1 only. So, $neighbor(a) = \{c, e\}$. The weight of $(a, c) = 1/(|n_1| - 1) = 0.5$. The weight of $(a, e) = 1/(|n_1| - 1) = 0.5$. Thus, we break the tie based on alphabetical order. So, a merges with c. We form $C_1 = \{a, c\}$ and mark a and c.

(b) visit b: Note that b is contained in n_2 only. So, $neighbor(b) = \{c, d\}$. Since c is already marked, b merges with d. We form $C_2 = \{b, d\}$ and mark b and d.

(c) since c and d are marked, we skip them.
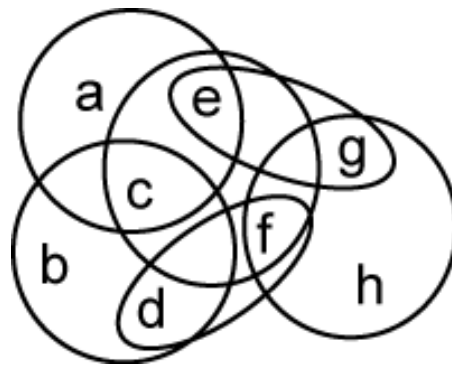
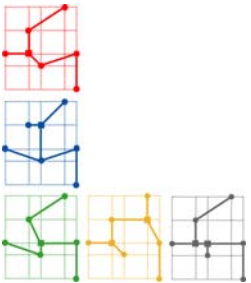| cluster	nodes
C_1	$\{a, c\}$
C_2	$\{b, d\}$
C_3	$\{e, g\}$
C_4	$\{f, h\}$

Edge Coarsening (cont)

(d) visit e: the unmarked neighbors of e are g and f. We see that $w(e, g) = 1$ and $w(e, f) = 0.5$. So, e merges with g. We form $C_3 = \{e, g\}$ and mark e and g.

(e) visit f: Node f is contained in n_3, n_4, and n_6. So, $neighbor(f) = \{c, d, e, g, h\}$. But, the only unmarked neighbor is h. So, f merges with h. We form $C_4 = \{f, h\}$ and mark f and h.
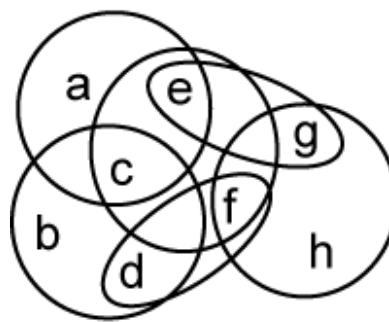
(f) since g and h are marked, we skip them.
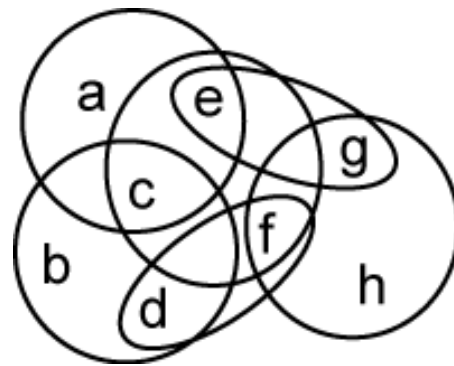
cluster	nodes
C_1	$\{a, c\}$
C_2	$\{b, d\}$
C_3	$\{e, g\}$
C_4	$\{f, h\}$

Obtaining Clustered-level Netlist

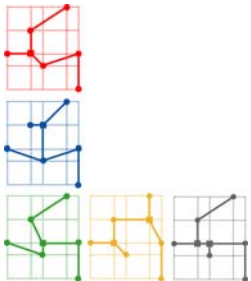- # of nodes/hyperedges reduced: 4 nodes, 5 hyperedges

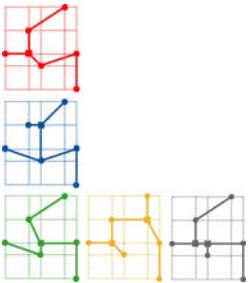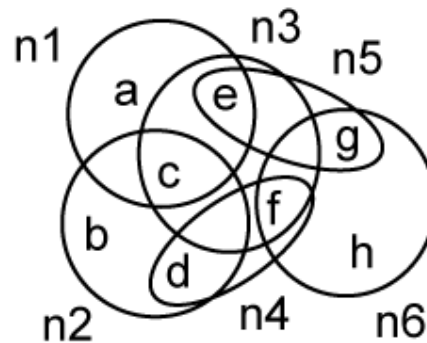| net	gate-level	cluster-level	final	cluster	nodes
n_1	$\{a, c, e\}$	$\{C_1, C_1, C_3\}$	$\{C_1, C_3\}$	C_1	$\{a, c\}$
n_2	$\{b, c, d\}$	$\{C_2, C_1, C_2\}$	$\{C_1, C_2\}$	C_2	$\{b, d\}$
n_3	$\{c, e, f\}$	$\{C_1, C_3, C_4\}$	$\{C_1, C_3, C_4\}$	C_3	$\{e, g\}$
n_4	$\{d, f\}$	$\{C_2, C_4\}$	$\{C_2, C_4\}$	C_4	$\{f, h\}$
n_5	$\{e, g\}$	$\{C_3, C_3\}$	$\emptyset$		
n_6	$\{f, g, h\}$	$\{C_4, C_3, C_4\}$	$\{C_3, C_4\}$		

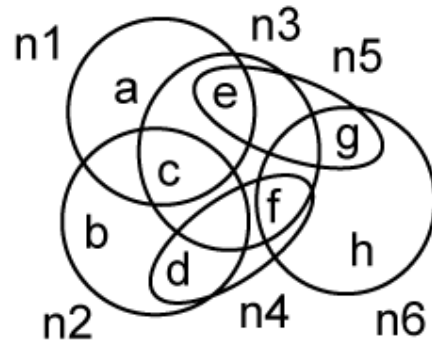(a)   (b)

Hyperedge Coarsening

- ## Initial setup

  - Sort hyper-edges in increasing size: n_4, n_5, n_1, n_2, n_3, n_6
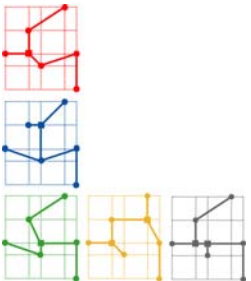 - Unmark all nodes

Hyperedge Coarsening

(a) visit $n_4 = \{d, f\}$: since d and f are not marked yet, we form $C_1 = \{d, f\}$ and mark d and f.

(b) visit $n_5 = \{e, g\}$: since e and g are not marked yet, we form $C_2 = \{e, g\}$ and mark e and g.

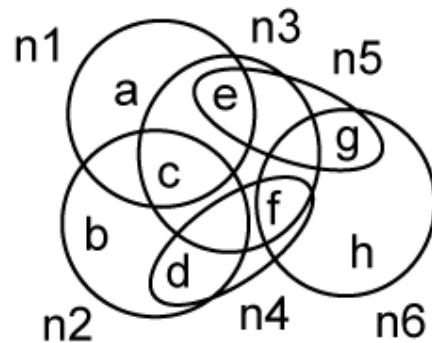(c) visit $n_1 = \{a, c, e\}$: since e is already marked, we skip n_1.

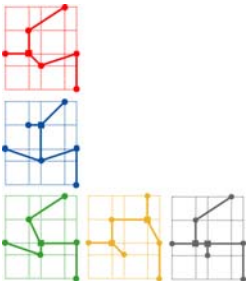| cluster	nodes
C_1	$\{d, f\}$
C_2	$\{e, g\}$
C_3	$\{a\}$
C_4	$\{b\}$
C_5	$\{c\}$
C_6	$\{h\}$

Hyperedge Coarsening

(d) visit $n_2 = \{b, c, d\}$: since d is already marked, we skip n_2.

(e) visit $n_3 = \{c, e, f\}$: since e and f are already marked, we skip n_3.

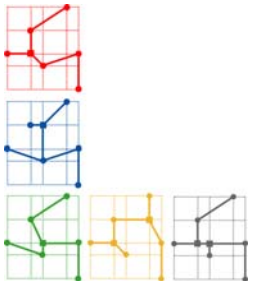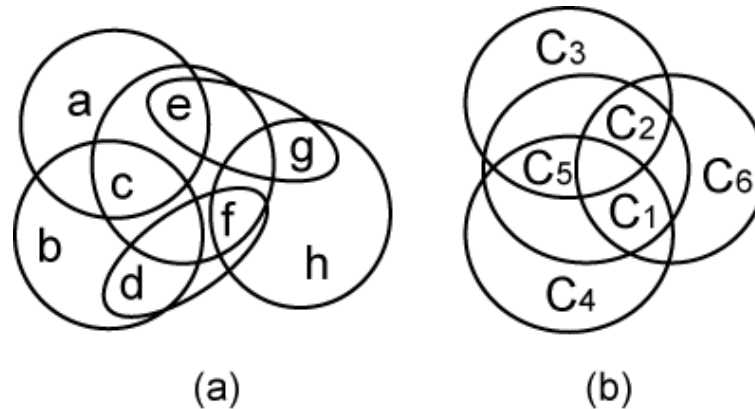(f) visit $n_6 = \{f, g, h\}$: since f and g are already marked, we skip n_6.

cluster	nodes
C_1	$\{d, f\}$
C_2	$\{e, g\}$
C_3	$\{a\}$
C_4	$\{b\}$
C_5	$\{c\}$
C_6	$\{h\}$

Obtaining Clustered-level Netlist

- # of nodes/hyperedges reduced: 6 nodes, 4 hyperedges

net	gate-level	cluster-level	final
n_1	$\{a, c, e\}$	$\{C_3, C_5, C_2\}$	$\{C_3, C_5, C_2\}$
n_2	$\{b, c, d\}$	$\{C_4, C_5, C_1\}$	$\{C_4, C_5, C_1\}$
n_3	$\{c, e, f\}$	$\{C_5, C_2, C_1\}$	$\{C_5, C_2, C_1\}$
n_4	$\{d, f\}$	$\{C_1, C_1\}$	$\emptyset$
n_5	$\{e, g\}$	$\{C_2, C_2\}$	$\emptyset$
n_6	$\{f, g, h\}$	$\{C_1, C_2, C_6\}$	$\{C_1, C_2, C_6\}$

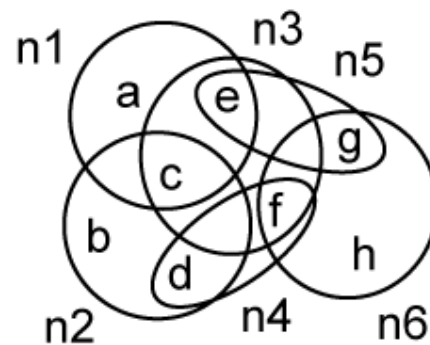| cluster	nodes
C_1	$\{d, f\}$
C_2	$\{e, g\}$
C_3	$\{a\}$
C_4	$\{b\}$
C_5	$\{c\}$
C_6	$\{h\}$

(a)   (b)

Modified Hyperedge Coarsening

- Revisit skipped nets during hyperedge coarsening
  - We skipped n_1, n_2, n_3, n_6
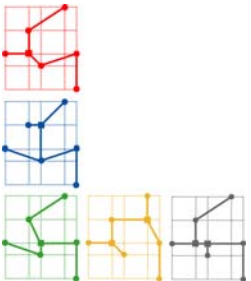 - Coarsen un-coarsened nodes in each net
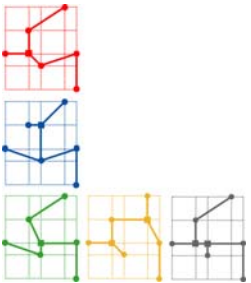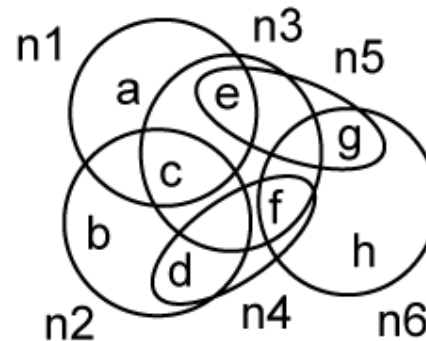
Modified Hyperedge Coarsening

(a) visit $n_1 = \{a, c, e\}$: since e is already marked during HEC, we group the remaining unmarked nodes a and c. We form $C_3 = \{a, c\}$ and mark a and c.

(b) visit $n_2 = \{b, c, d\}$: since d is marked during HEC and c during MHEC as above, we form $C_4 = \{b\}$ and mark b.

(c) visit $n_3 = \{c, e, f\}$: all nodes are already marked, so we skip n_3.

(d) visit $n_6 = \{f, g, h\}$: since f and g are already marked, we form $C_5 = \{h\}$ and mark h.
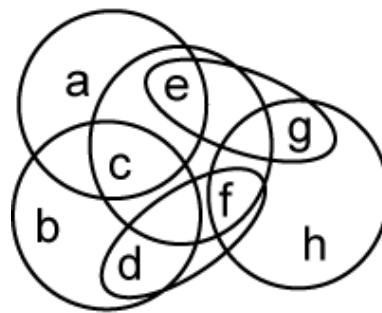

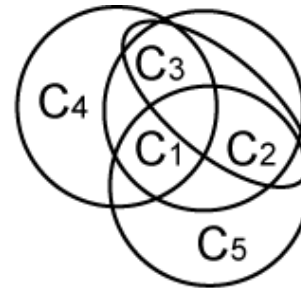| cluster	nodes
C_1	$\{d, f\}$
C_2	$\{e, g\}$
C_3	$\{a, c\}$
C_4	$\{b\}$
C_5	$\{h\}$

Obtaining Clustered-level Netlist

- # of nodes/hyperedges reduced: 5 nodes, 4 hyperedges

net	gate-level	cluster-level	final
n_1	$\{a, c, e\}$	$\{C_3, C_3, C_2\}$	$\{C_3, C_2\}$
n_2	$\{b, c, d\}$	$\{C_4, C_3, C_1\}$	$\{C_4, C_3, C_1\}$
n_3	$\{c, e, f\}$	$\{C_3, C_2, C_1\}$	$\{C_3, C_2, C_1\}$
n_4	$\{d, f\}$	$\{C_1, C_1\}$	$\emptyset$
n_5	$\{e, g\}$	$\{C_2, C_2\}$	$\emptyset$
n_6	$\{f, g, h\}$	$\{C_1, C_2, C_5\}$	$\{C_1, C_2, C_5\}$

cluster	nodes
C_1	$\{d, f\}$
C_2	$\{e, g\}$
C_3	$\{a, c\}$
C_4	$\{b\}$
C_5	$\{h\}$

(a)   (b)