How to Store a Graph?

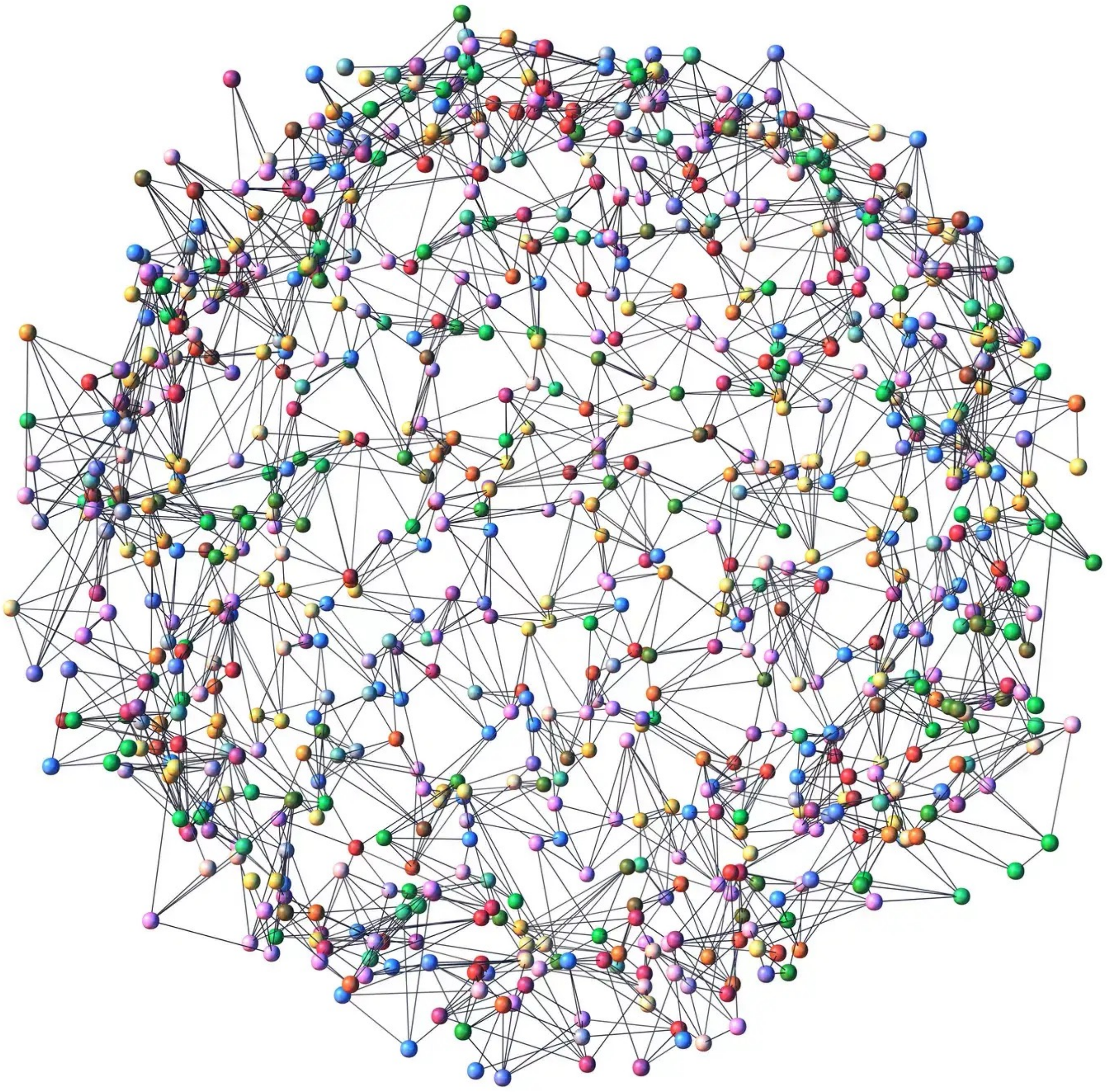
6 June, 2023

Henning Fernau
University of Trier, Germany

Kshitij Gajjar
Indian Institute of Technology Jodhpur

Large Graphs

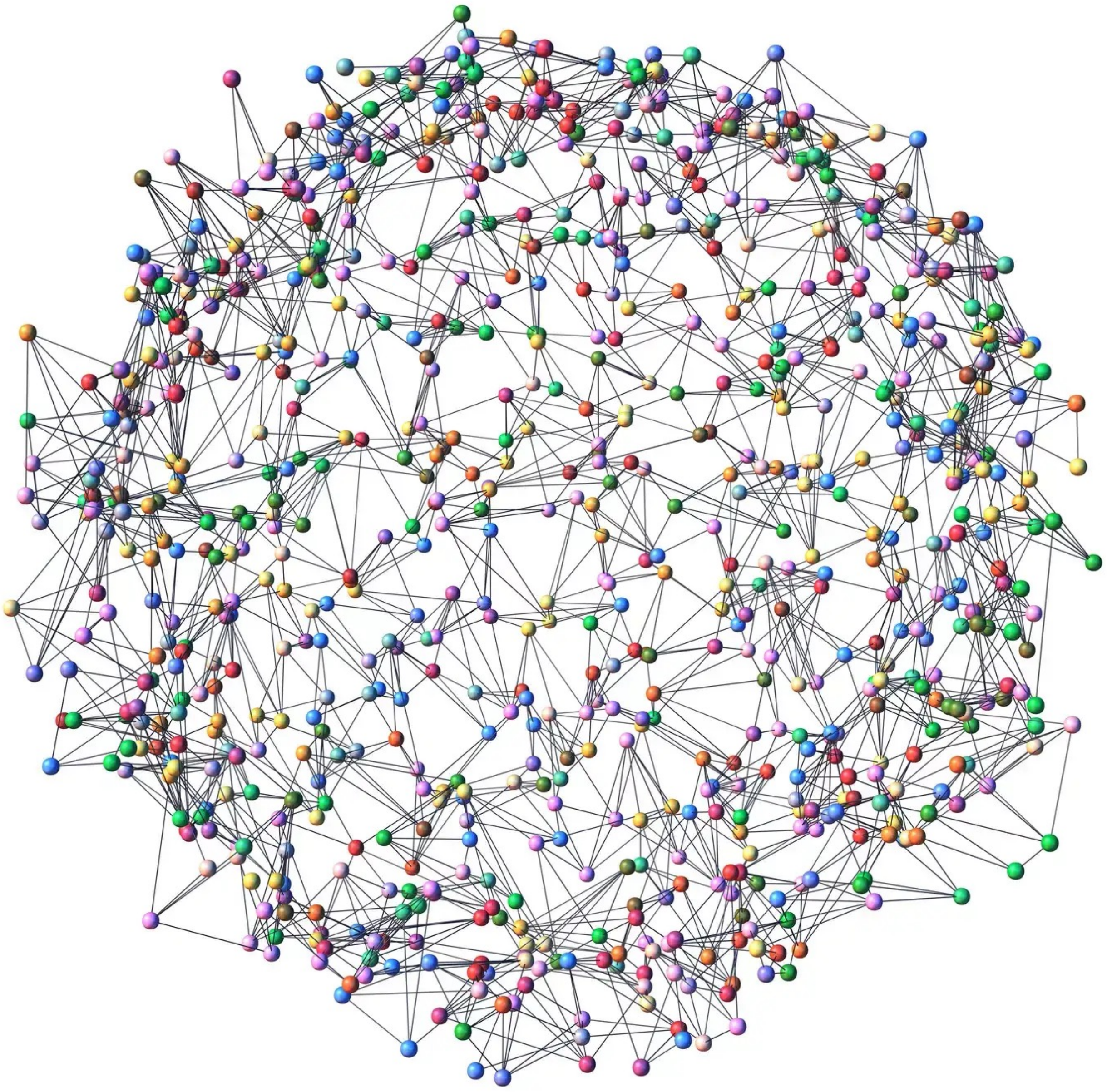
- Social network:
  Vertices: people;
  Edges: friendship.
- Communication network:
  Vertices: computers;
  Edges: wires.
- Traffic network:
  Vertices: junctions;
  Edges: roads.
- Human brain:
  Vertices: neurons;
  Edges: synapses.



https://anushkabhave.medium.com/graph-theory-algorithms-a816640610e3

A Random Graph

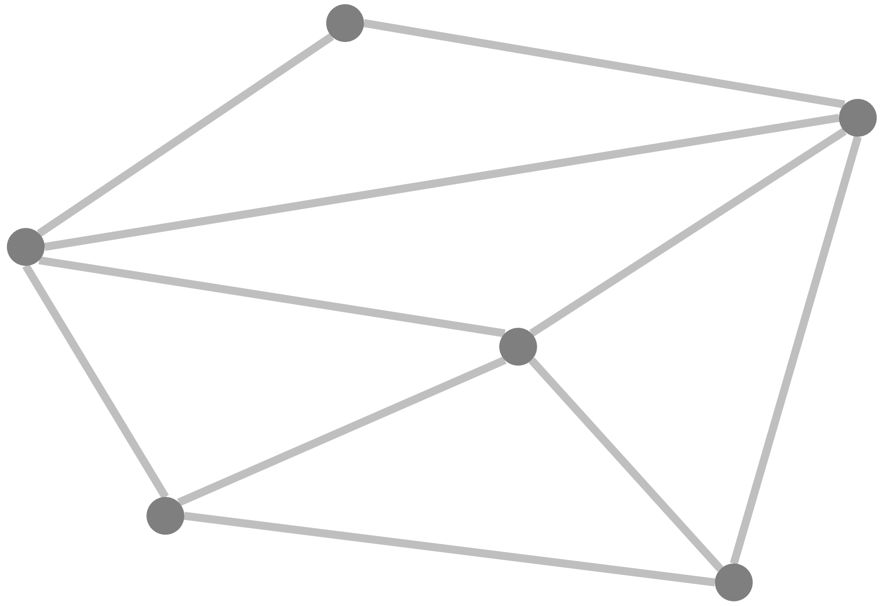
- Vertex set:
$$\{1,2,\dots,n\}.$$
- Consider a graph on $n = 120000$ vertices.
- Randomly put an edge between two vertices.
- For each pair of vertices, toss an unbased coin. If HEADS, then put an edge. If TAILS, then don't.
- The expected/average number of edges is $m \approx 3.6\times 10^9$.



https://anushkabhave.medium.com/graph-theory-algorithms-a816640610e3

How to Store a Graph?

- Vertex set:
  $\{1,2,\ldots,n\}$.
- Edge set:
  Incidence matrix: $O(mn)$.
  Adjacency matrix: $O(n^2)$.
  Adjacency list: $O(m+n)$.
- What if the vertex set could be a set of n positive integers (not necessarily $\{1,2,\ldots,n\}$), such that the numbers themselves encode the edge set of the graph?
- Then, we can eliminate the edge set entirely!



Has Anyone Ever Thought of this Before?

- Graph labelling.

- [Gallian, 2021] 576-page dynamic survey (cites over 3000 papers).

A Dynamic Survey of Graph Labeling

Joseph A. Gallian
Department of Mathematics and Statistics
University of Minnesota Duluth
Duluth, Minnesota 55812, U.S.A.
`jgallian@d.umn.edu`

Submitted: September 1, 1996; Accepted: November 14, 1997
Twenty-fourth edition, December 9, 2021
Mathematics Subject Classifications: 05C78

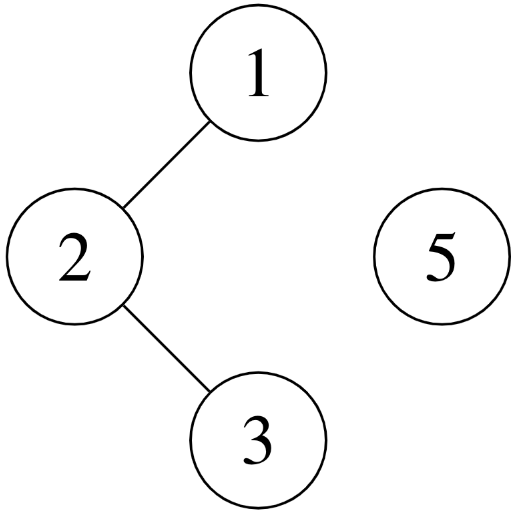
- Yes, people have thought of this before.

Sum Labelling

Definition [Harary, 1990] A graph G is called a sum graph if there is a one-to-one function $\lambda: V(G) \to \mathbb{N}$ such that for all vertices $v_1 \in V(G), v_2 \in V(G)$,

$$(v_1, v_2) \in E(G)$$
$$\Longleftrightarrow$$
$$\exists\, v_3 \in V(G) \text{ such that } \lambda(v_3) = \lambda(v_1) + \lambda(v_2)$$

Then we say that λ is a sum labelling of (the vertices of) G.



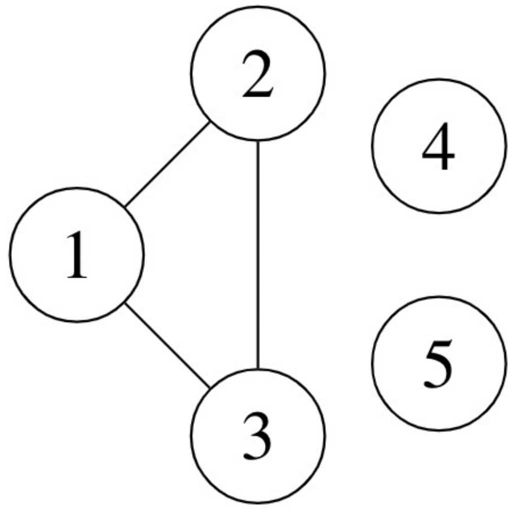
$(1, 2, 3, 5)$

A Puzzle

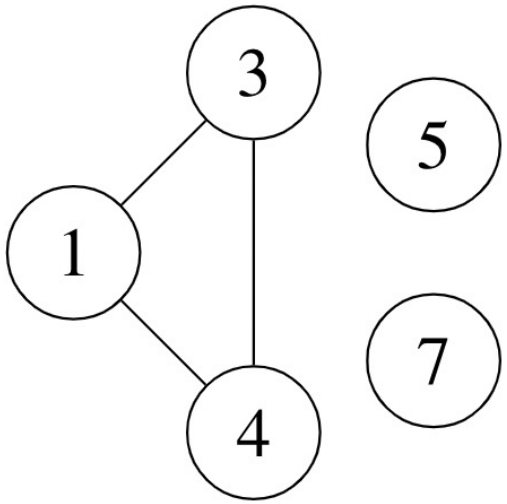
Q: Is this a valid sum labelling?

A: No.

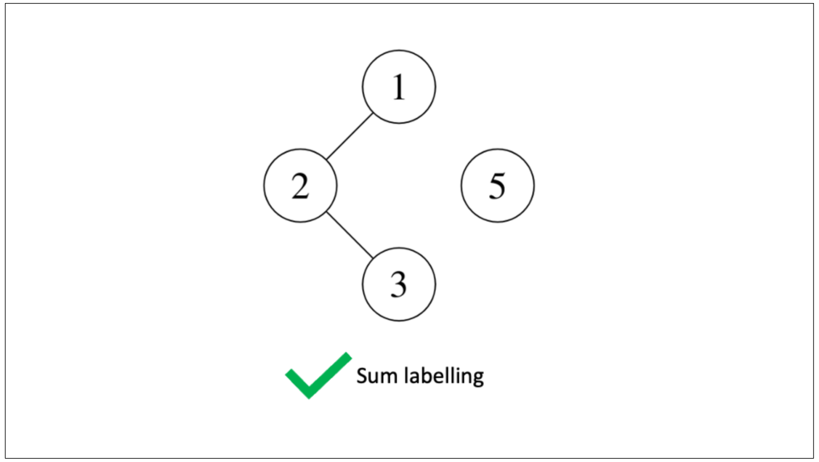
The vertices labelled 1 and 4 are non-adjacent, yet there is a vertex labelled $1 + 4 = 5$.

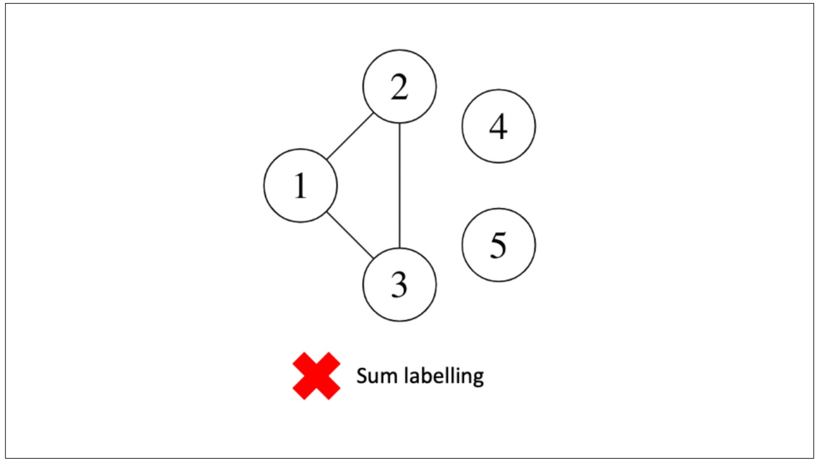


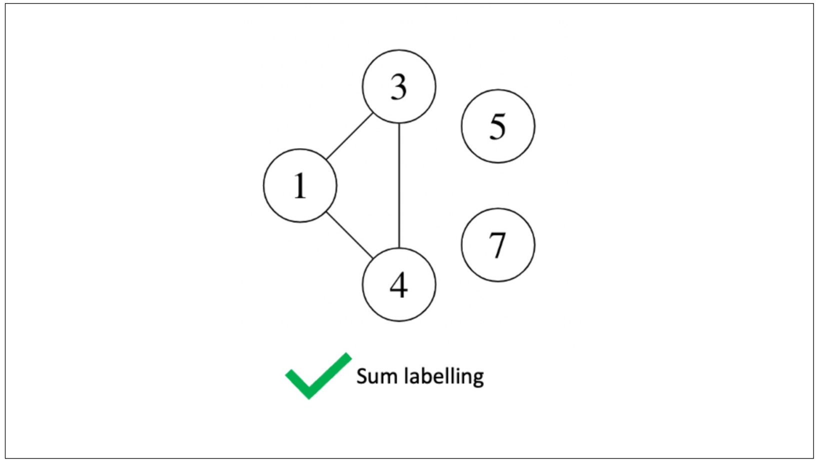
This is a valid sum labelling of the same graph.

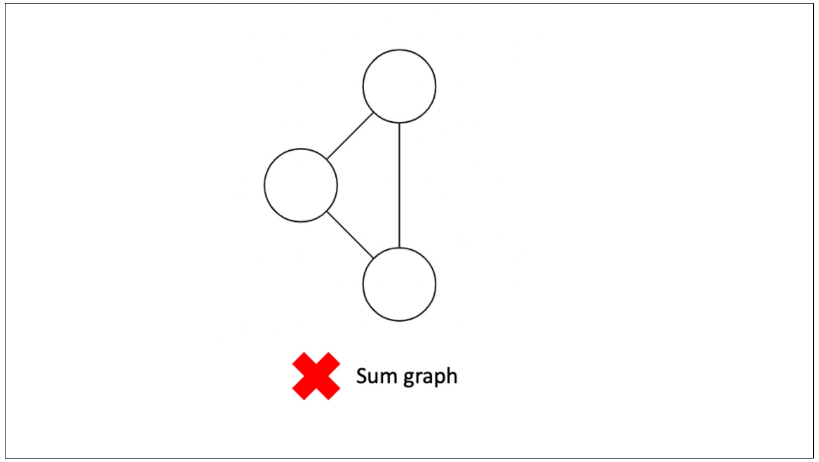


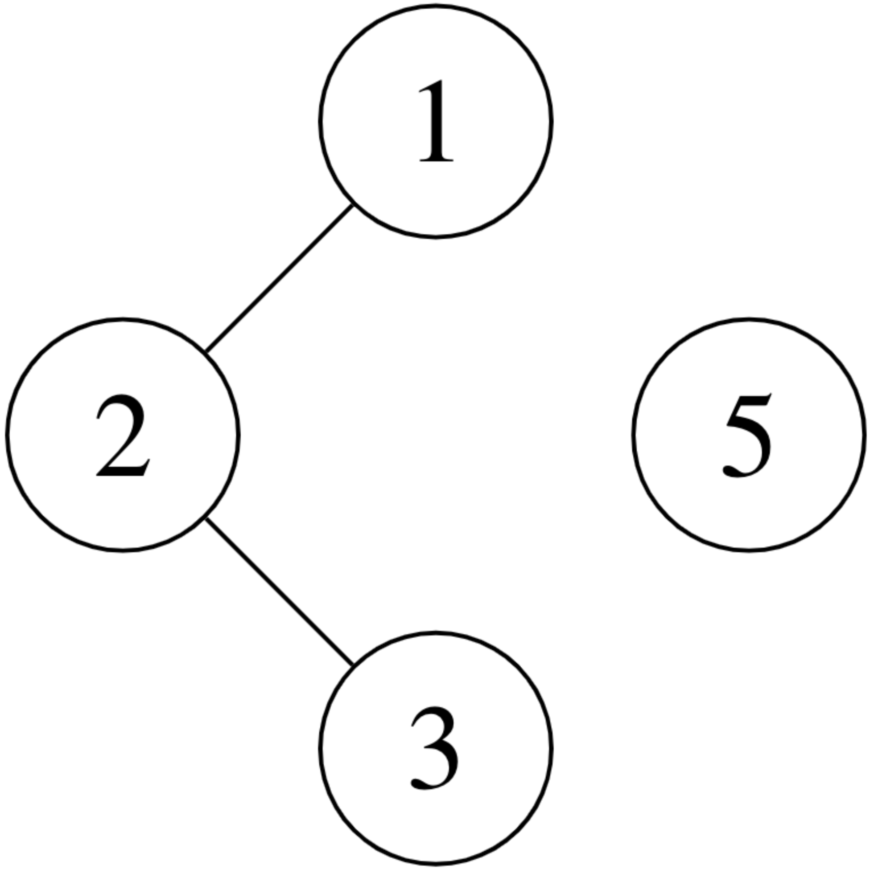
Examples


1
2
5
3
Sum labelling

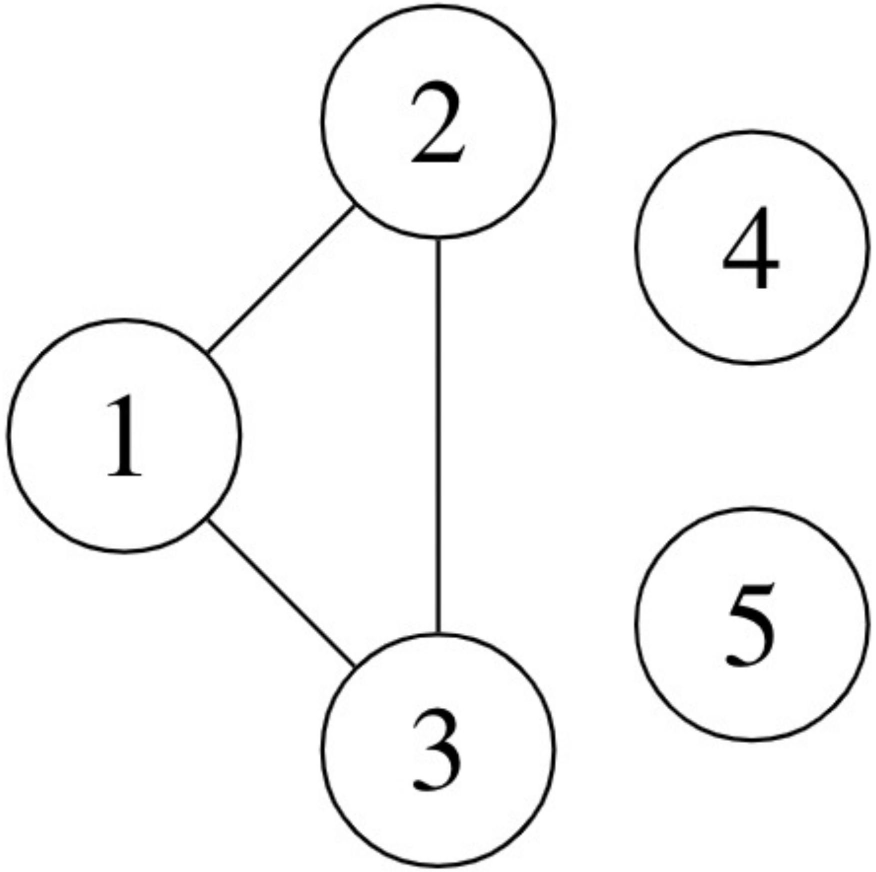

2
4
1
5
3
Sum labelling


3
5
1
7
4
Sum labelling

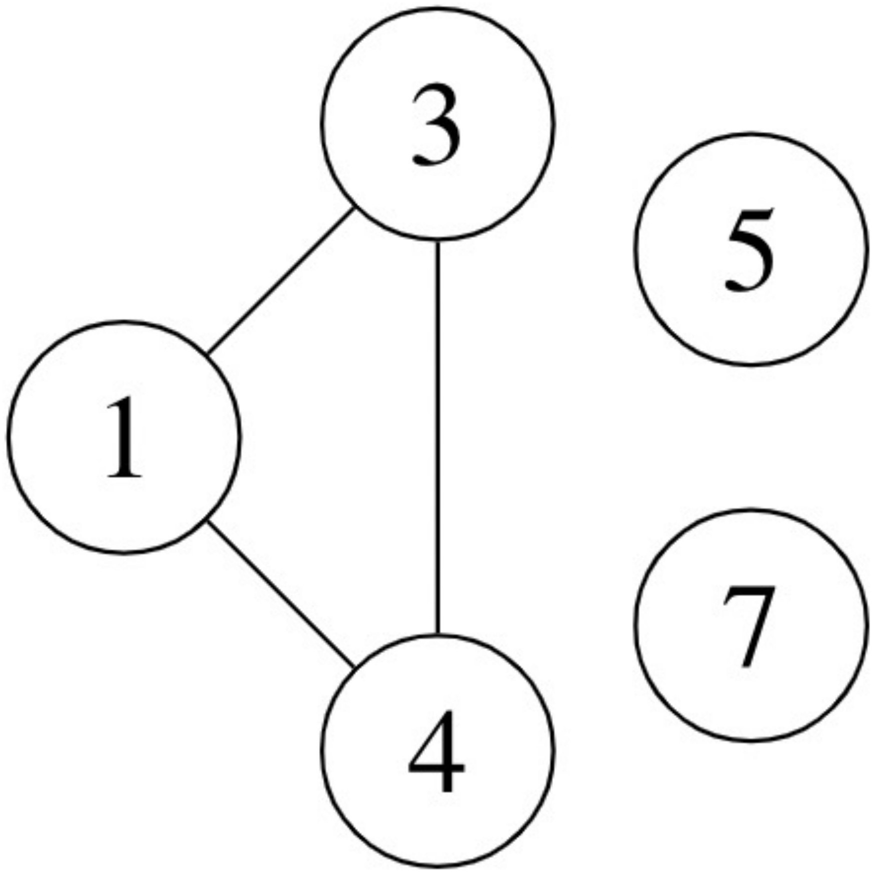

Sum graph



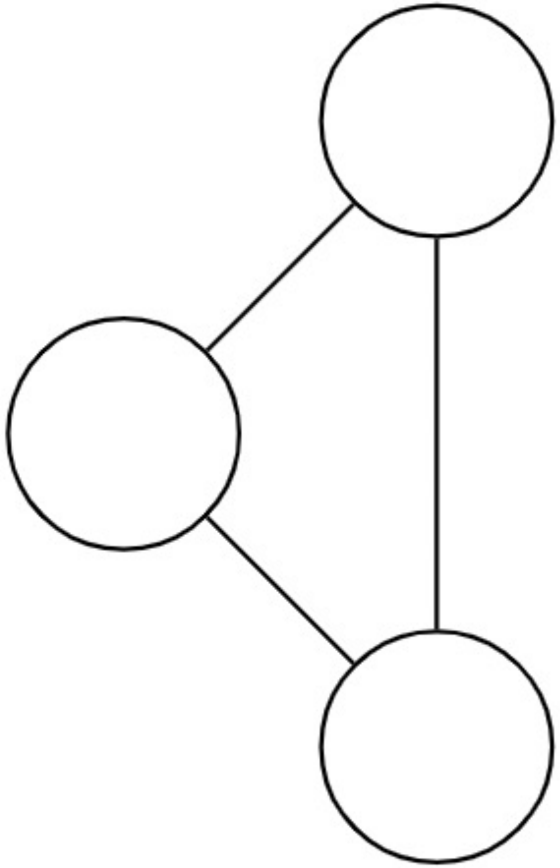
Sum labelling



Sum labelling



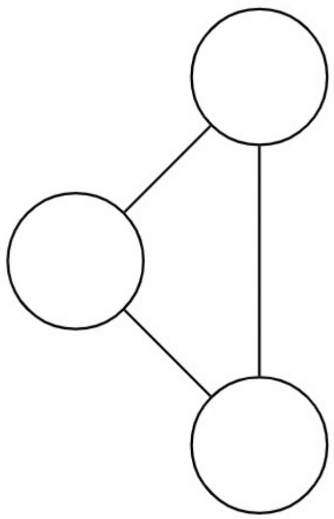
Sum labelling




Sum graph

Another Puzzle

Q: Is this a valid sum graph?

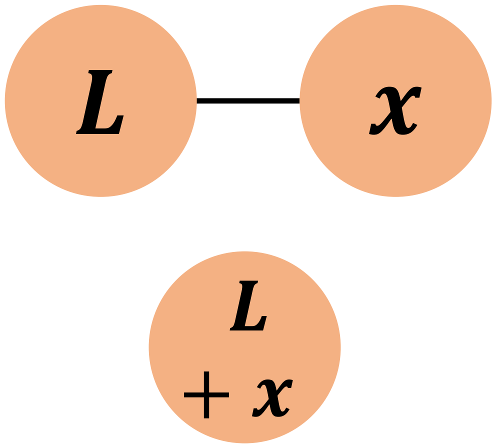


A: No. It does not have an isolated vertex.

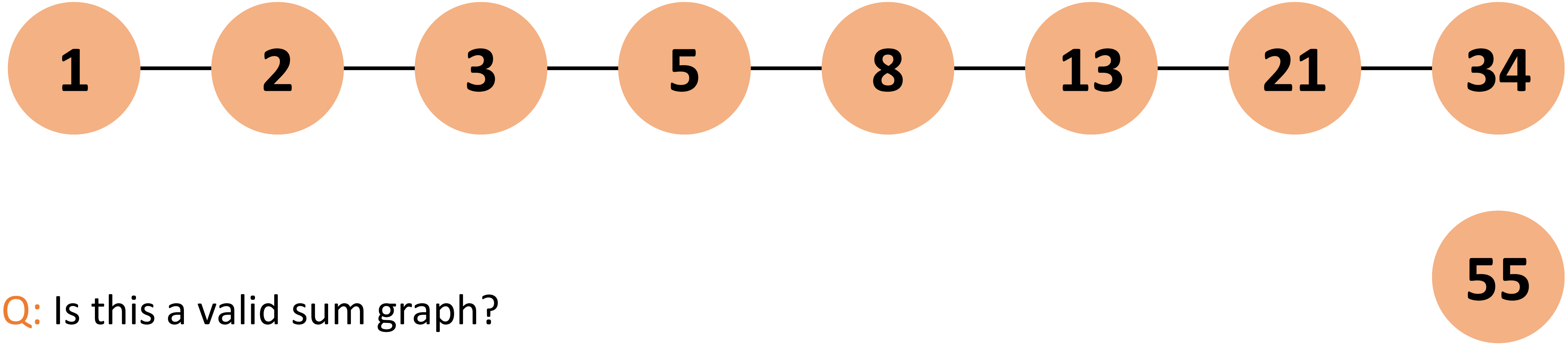
Fact [Harary, 1990] Every sum graph has at least one isolated vertex (a vertex with no neighbours).

Proof.

- Let L be the largest label. Claim: L is an isolated vertex.
- If not, then let its neighbour be x.
- Since (L, x) is an edge there is a vertex with label $L + x$.
- Contradicts the assumption that L is the largest label.



Yet Another Puzzle



Q: Is this a valid sum graph?

A: Yes.

Edges: $F_i + F_{i+1} = F_{i+2}$

Non-edges: $F_j < F_i + F_j < F_{j+1}$, when $i + 2 \leq j$

Some Graphs are Sum Graphs, Some are Not

- [Sutton, 2000] The sum number of a graph is the minimum number of isolated vertices that need to be added to the graph to make it a sum graph.
- Sum graphs have sum number zero.
- [Gould & Rodl, 1991] Sum number of every graph is at most n^2.
- The sum graph can be expressed as a sorted list of n positive integers.
- Edge queries can be answered in $O(\log n)$ time.

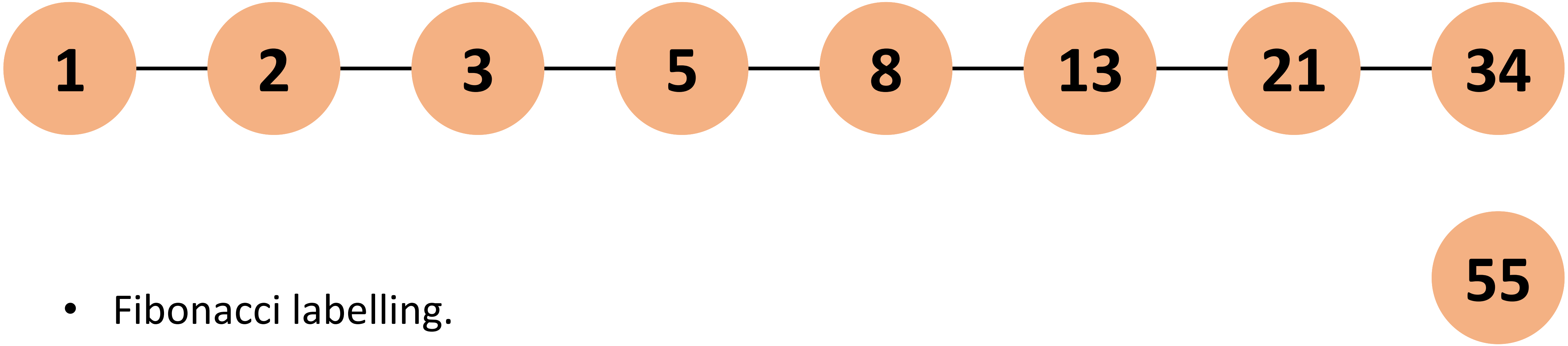
Earlier Work

- [Ellingham, 1990] Sum number of trees is always 1.
- [Harary, 1990] Sum number of cycles is 2, unless it is a 4-cycle, in which case it is 3.
- [Bergstrand *et al.*, 1989] Sum number of complete graphs is $2n - 3$.
- [Miller, Ryan, Slamin, Smyth, 1998] Sum number of wheel graphs is $\Theta(n)$.
- [Wang, Liu, 2001] Sum number of complete bipartite graphs is $\Theta(n^2)$.
- [Fernau, Ryan, Sugeng, 2008] Sum number of flowers is always 2.

Does this Help in Storing the Graph Better?

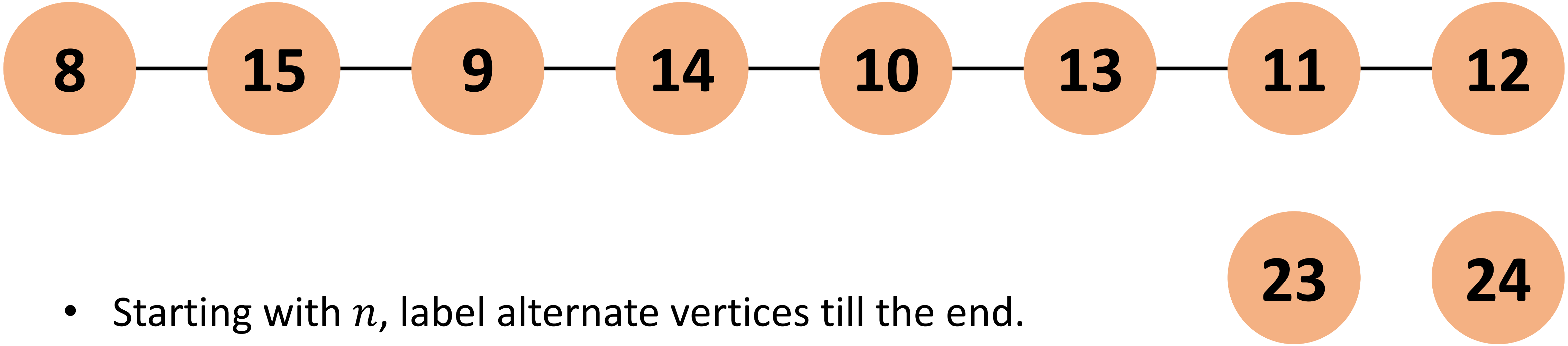
- Almost all earlier works attempt to optimize the number of isolated vertices required.
- From a computational (space) complexity, we should also be optimizing the number of bits required for each label.
- Does having extra isolated vertices reduce the space complexity of storing the graph?

Does this Help in Storing the Graph Better?



- Fibonacci labelling.
- n^{th} term of Fibonacci series is exponential in n.
- $\Omega(n)$ bits to store the largest label.

Does this Help in Storing the Graph Better?



- Starting with n, label alternate vertices till the end.
- After reaching the end, continue labelling in reverse.
- Size of the largest isolate is $3n$: can be stored using $O(\log n)$ bits.

Our Result

Theorem [Kratochvil, Miller, Nguyen, 2001] Every n-vertex sum graph has a sum labelling in which the size of each label is at most 4^n.

Pros	Cons
Label size can be upper-bounded in terms of number of vertices	Size of labels is exponential (requires linear numer of bits)
Number of isolated vertices needed is minimum possible	Proof is existential, not constructive

Theorem [Fernau, G., 2021] Every graph on n vertices and m edges can be made a sum graph by adding at most m isolated vertices to it such that the size of each label is at most $12n^3$.

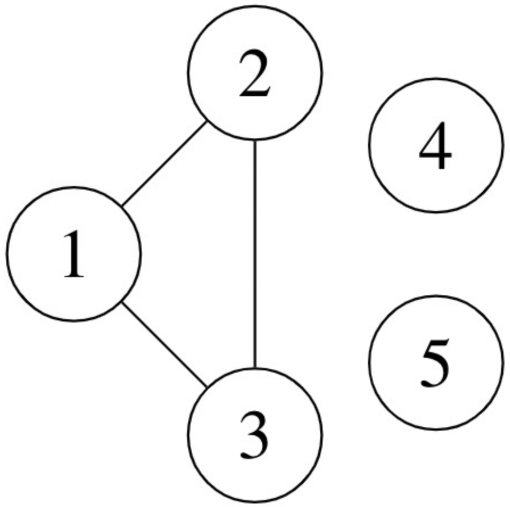
Pros	Cons
Size of labels is polynomial (requires logarithmic bits)	Does not perform well for dense graphs
Labelling can be constructed in polynomial time	
Works optimally for sparse graphs	

Proof Idea

- Definition Three vertices with labels (a, b, c) in a graph G form a conflicting triple if

$$(a, b) \notin E(G) \text{ and } a + b = c.$$

- For example, here $(1,4,5)$ is a conflicting triple.



Proof Idea

- Definition Three vertices with labels (a, b, c) in a graph G form a conflicting triple if

$$(a, b) \notin E(G) \text{ and } a + b = c.$$

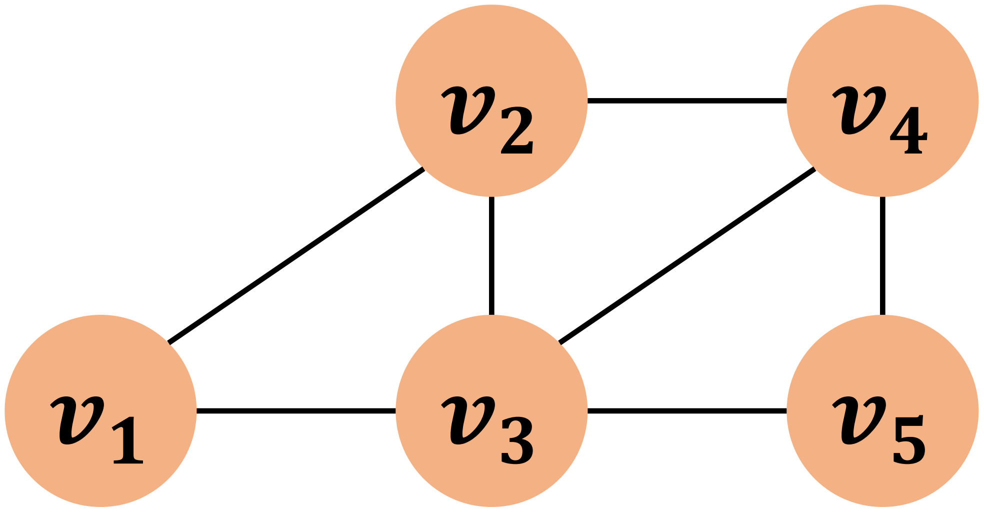
- Intuition: given a graph on n vertices, pick a set of size n out of a set of (say) the first n^{20} positive integers at random.

- Edges: for every edge (a, b), add isolated vertex with label $a + b$.

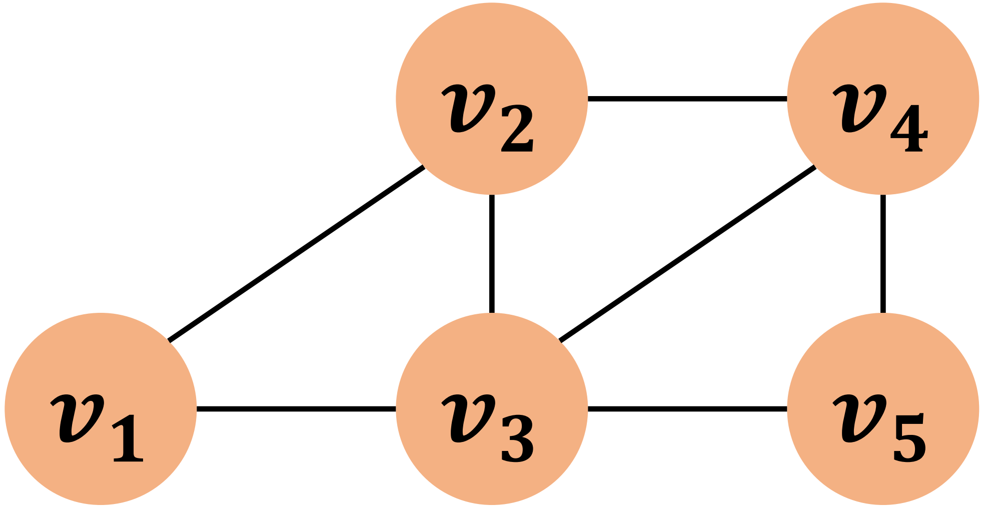
- Non-edges: difficult to end up with a conflicting triple, since

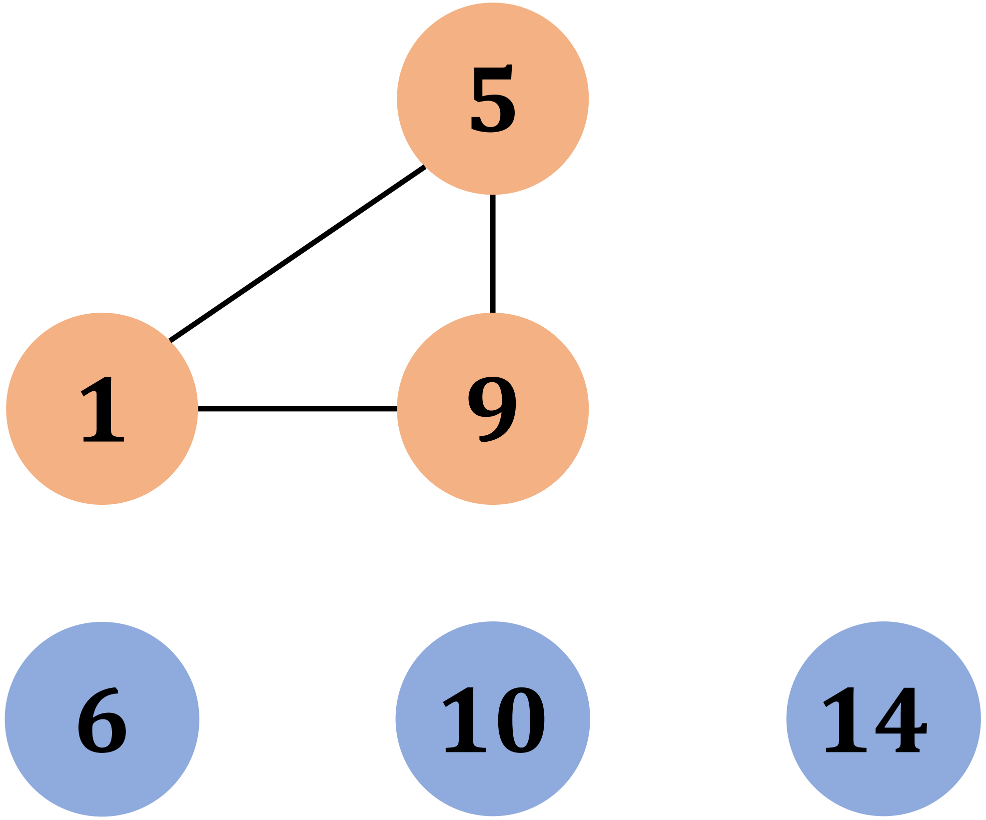
  1. there are only $c - 1$ ways for two numbers to add up to c;
  2. there are at most $n^2 \times n^2 \times n^2 = n^6$ conflicting triples, and $n^6 \ll n^{20}$.

Algorithm



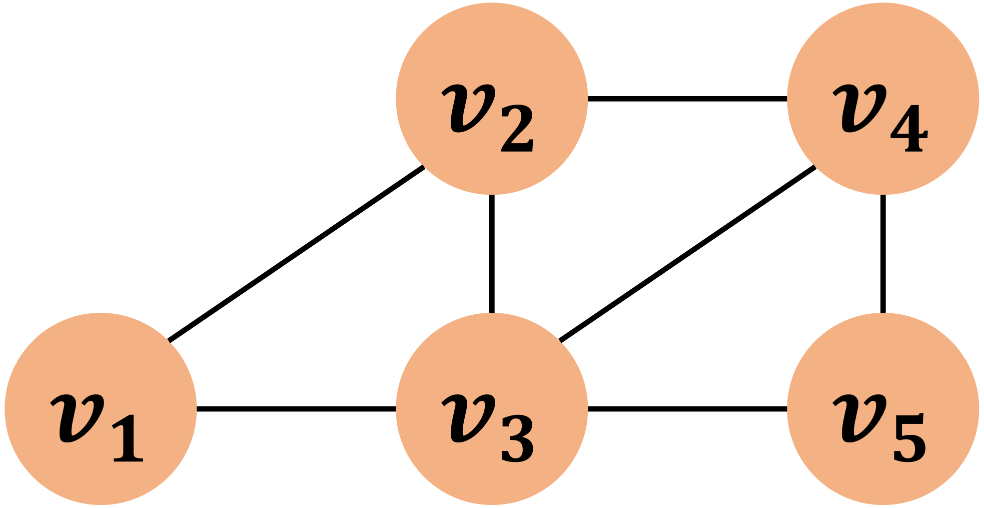
Algorithm





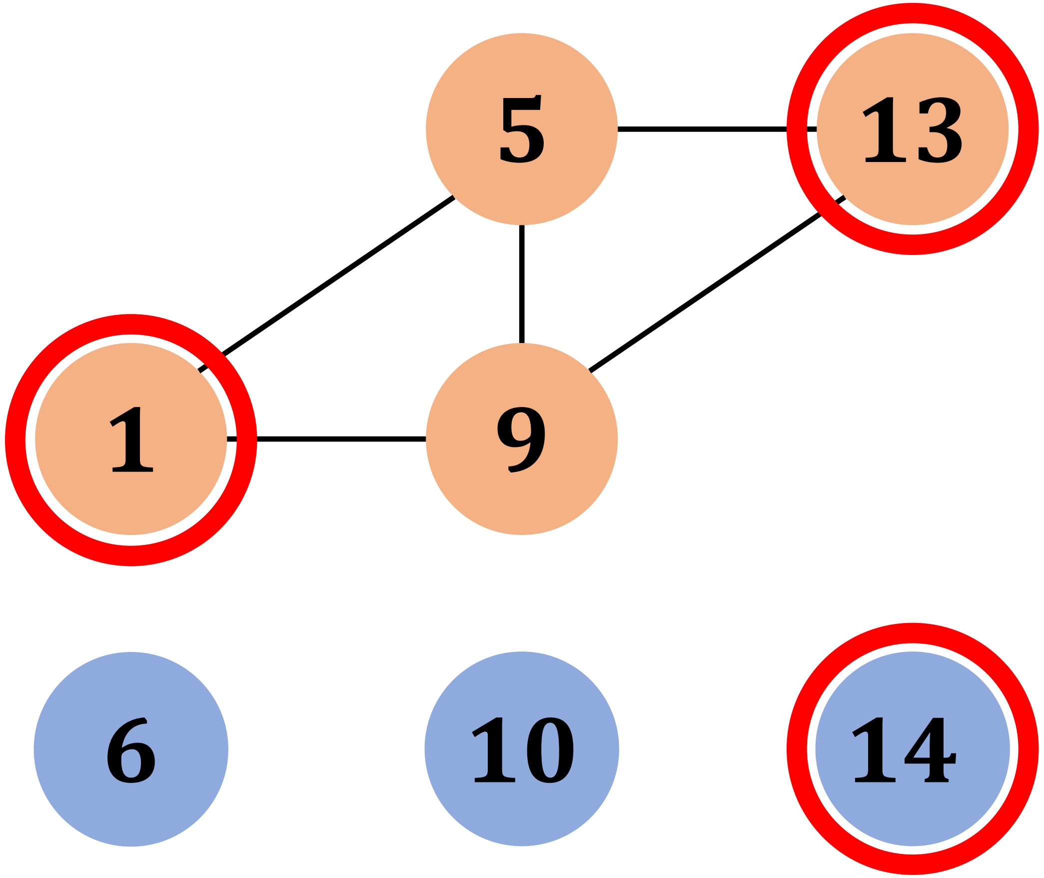
- All vertices of original graph are labelled $1 \bmod 4$.
- All isolated vertices are labelled $2 \bmod 4$.

Algorithm

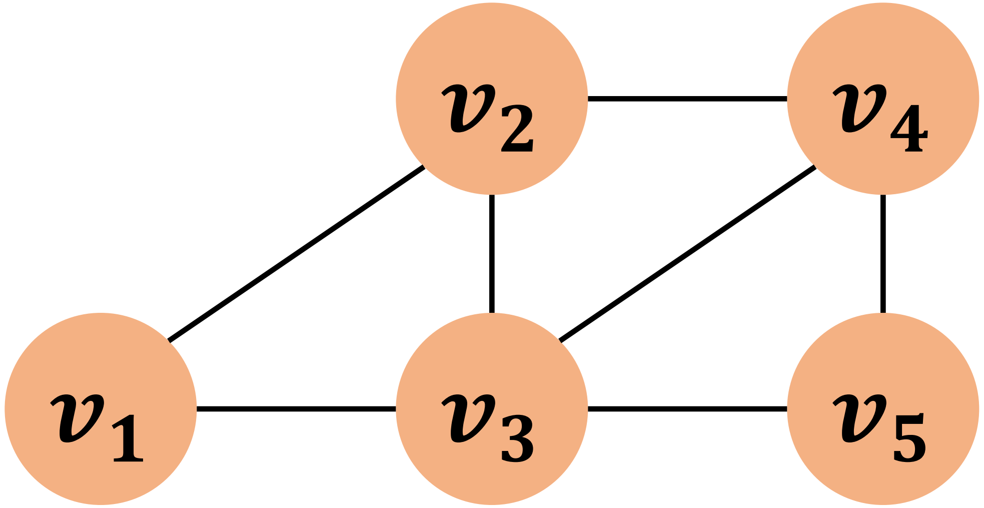


- All vertices of original graph are labelled $1 \bmod 4$.

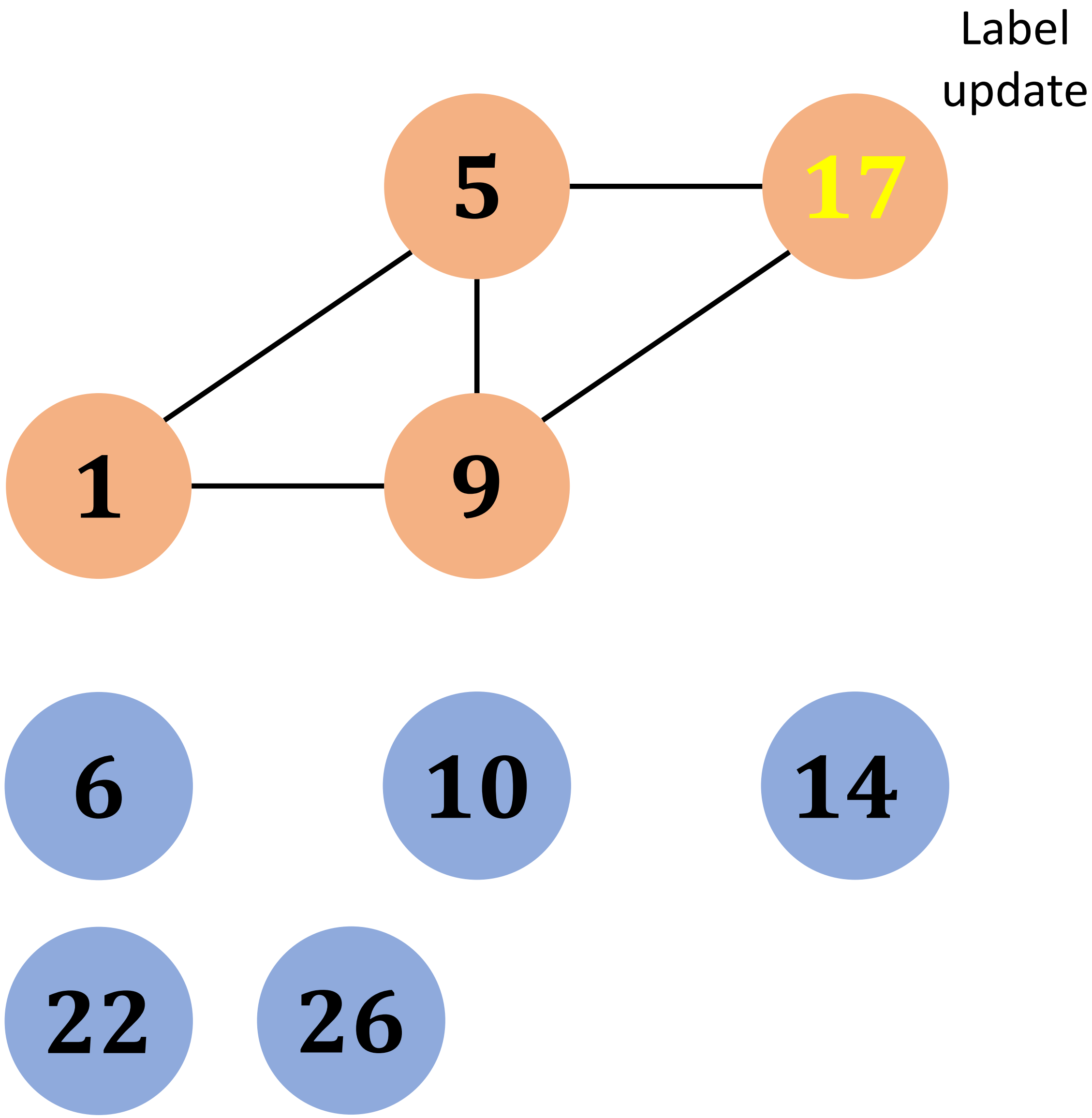
- All isolated vertices are labelled $2 \bmod 4$.



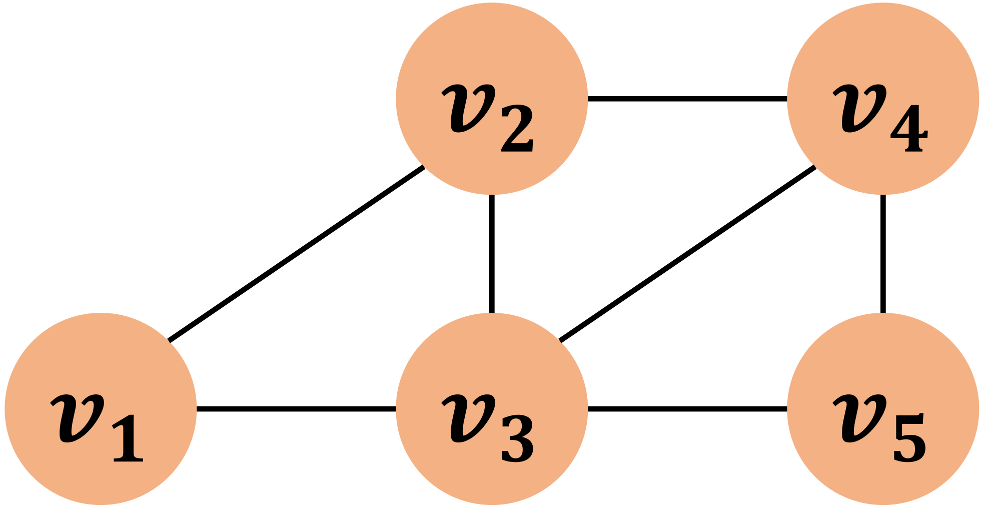
Algorithm



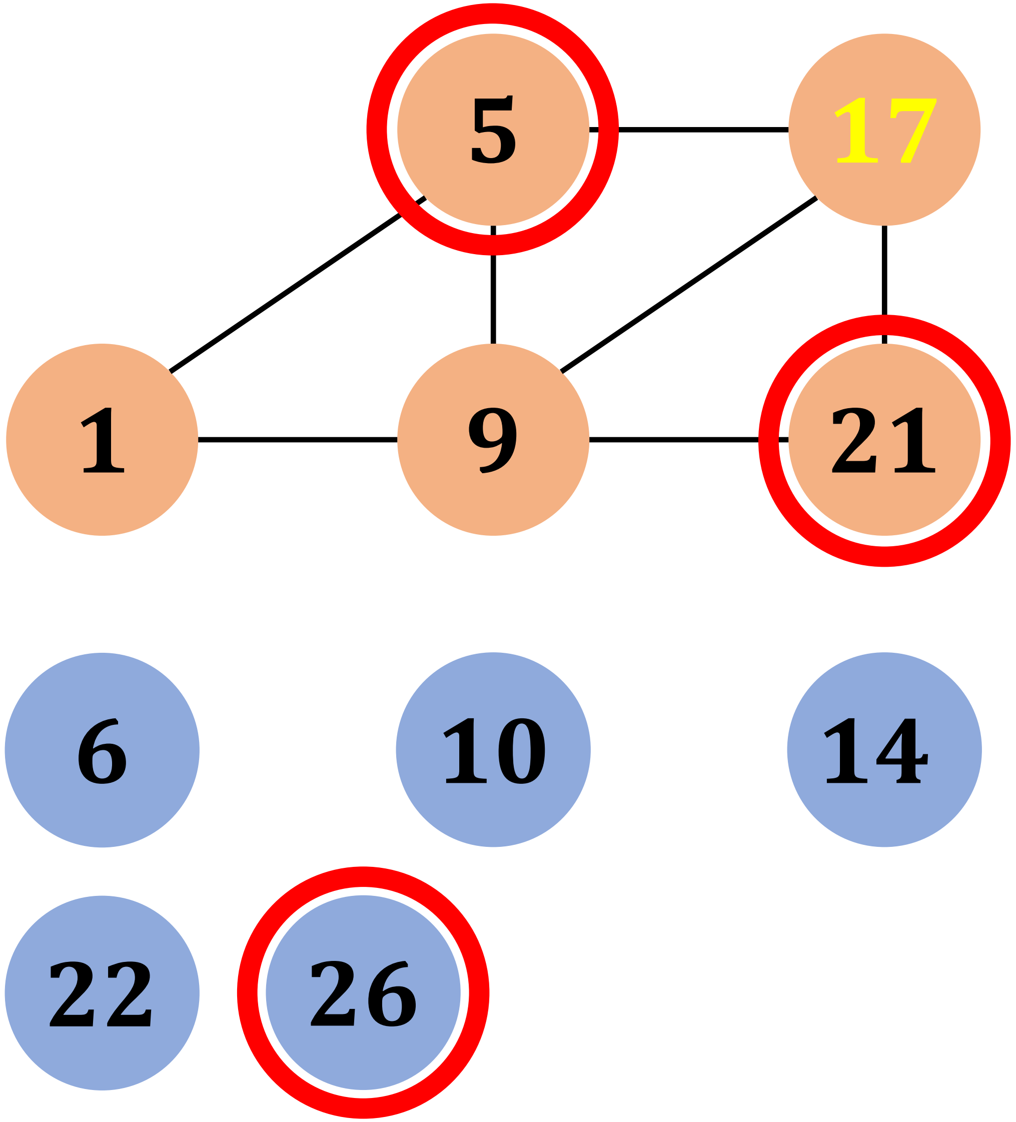
- All vertices of original graph are labelled $1 \bmod 4$.
- All isolated vertices are labelled $2 \bmod 4$.



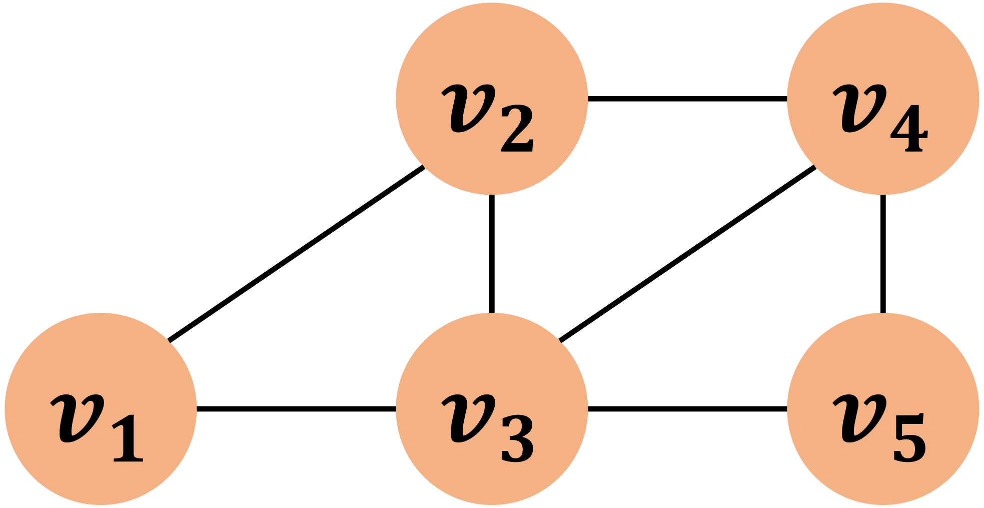
Algorithm

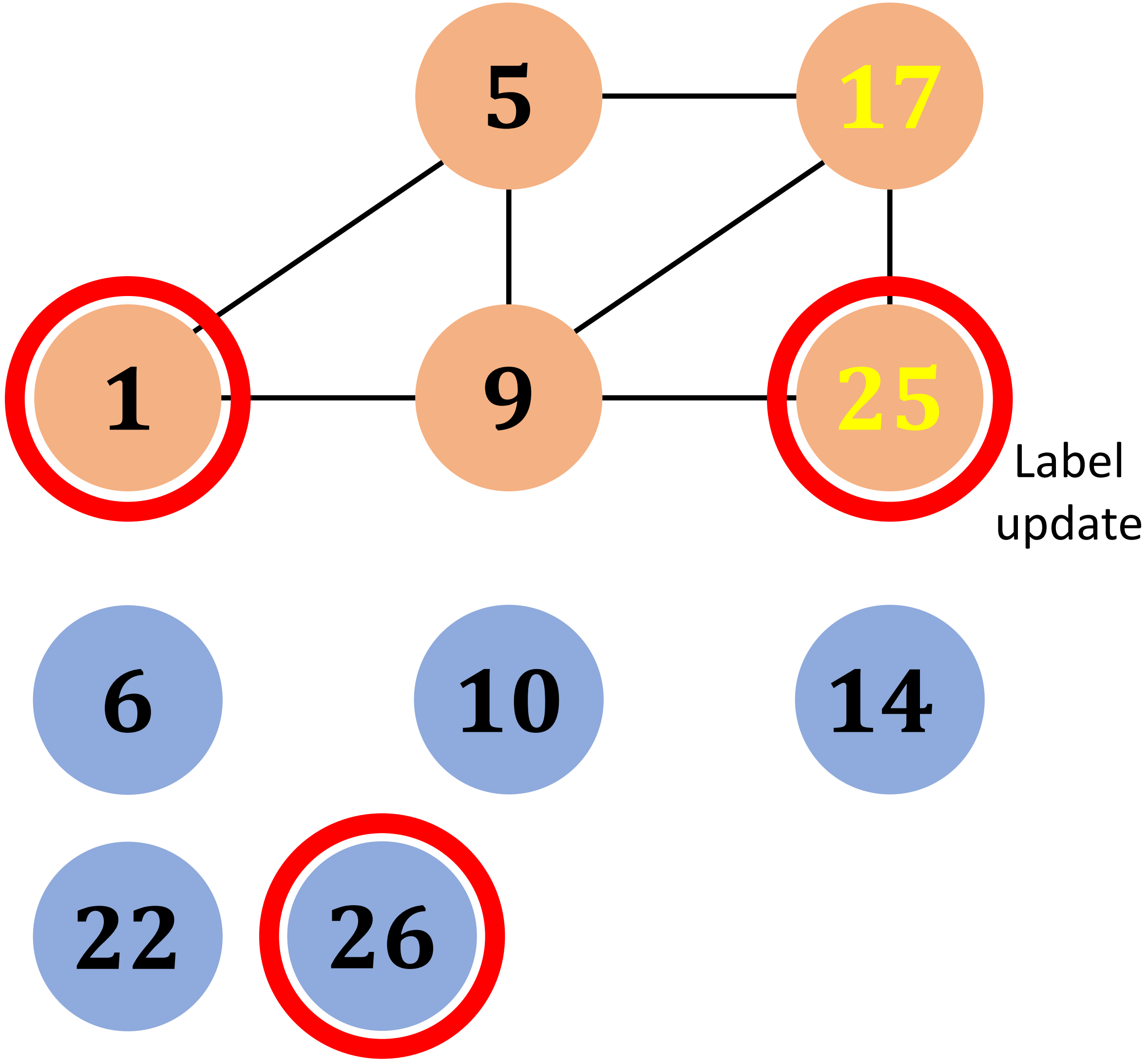


- All vertices of original graph are labelled $1 \bmod 4$.
- All isolated vertices are labelled $2 \bmod 4$.



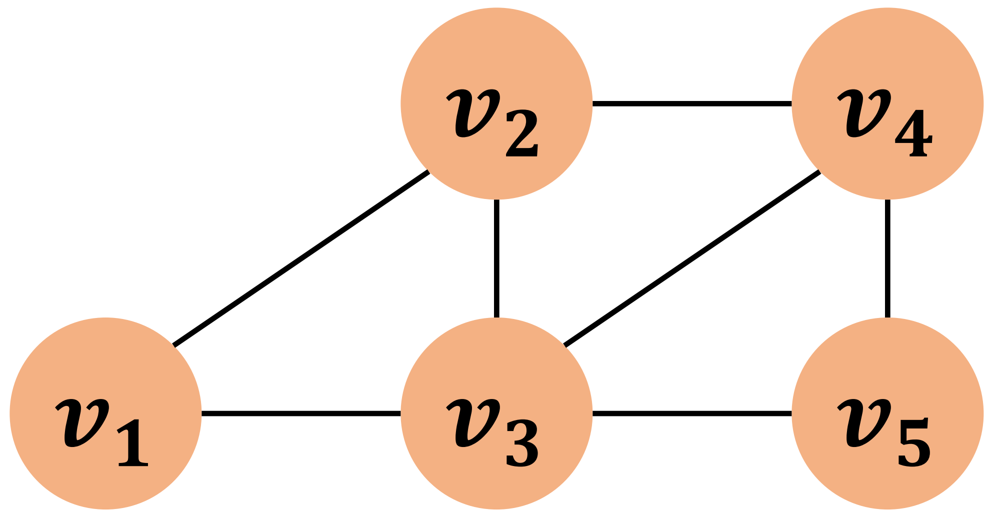
Algorithm



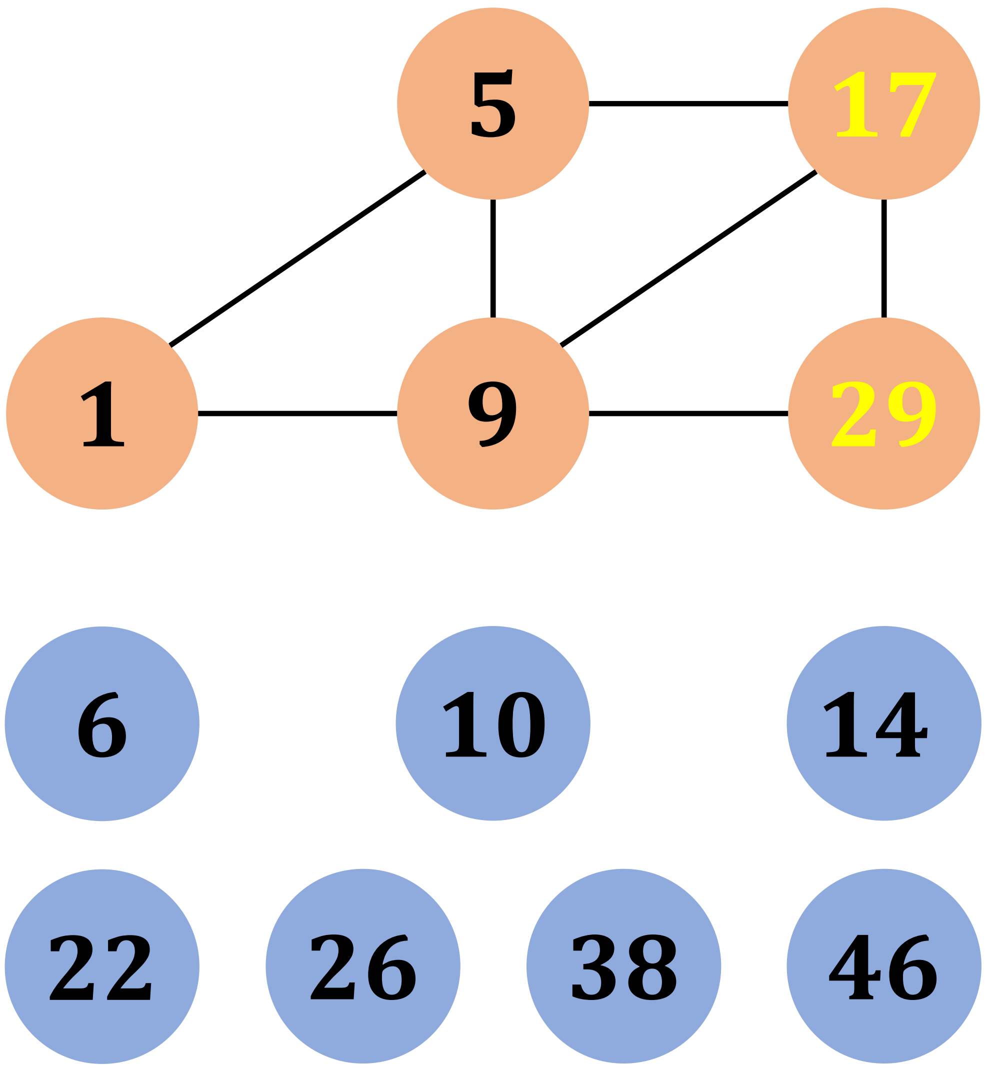


- All vertices of original graph are labelled $1 \bmod 4$.
- All isolated vertices are labelled $2 \bmod 4$.

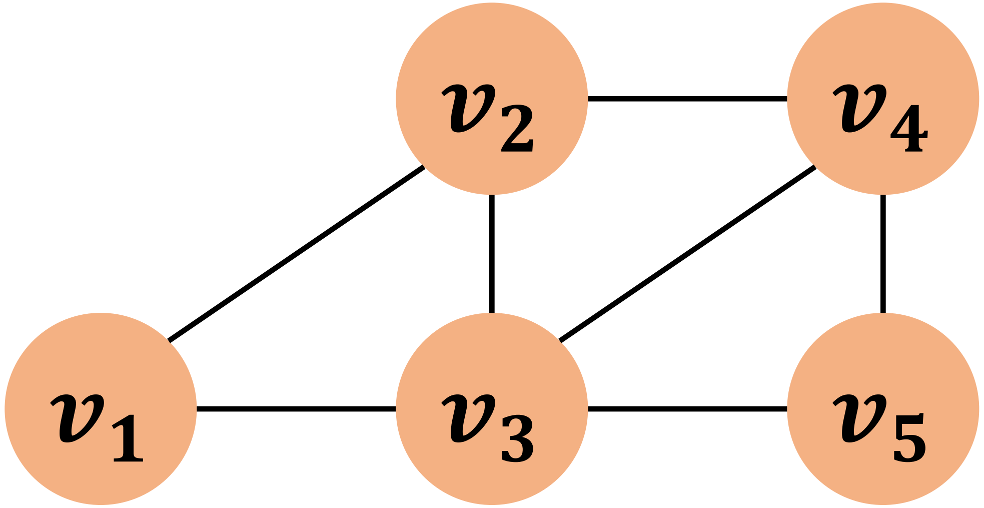
Algorithm



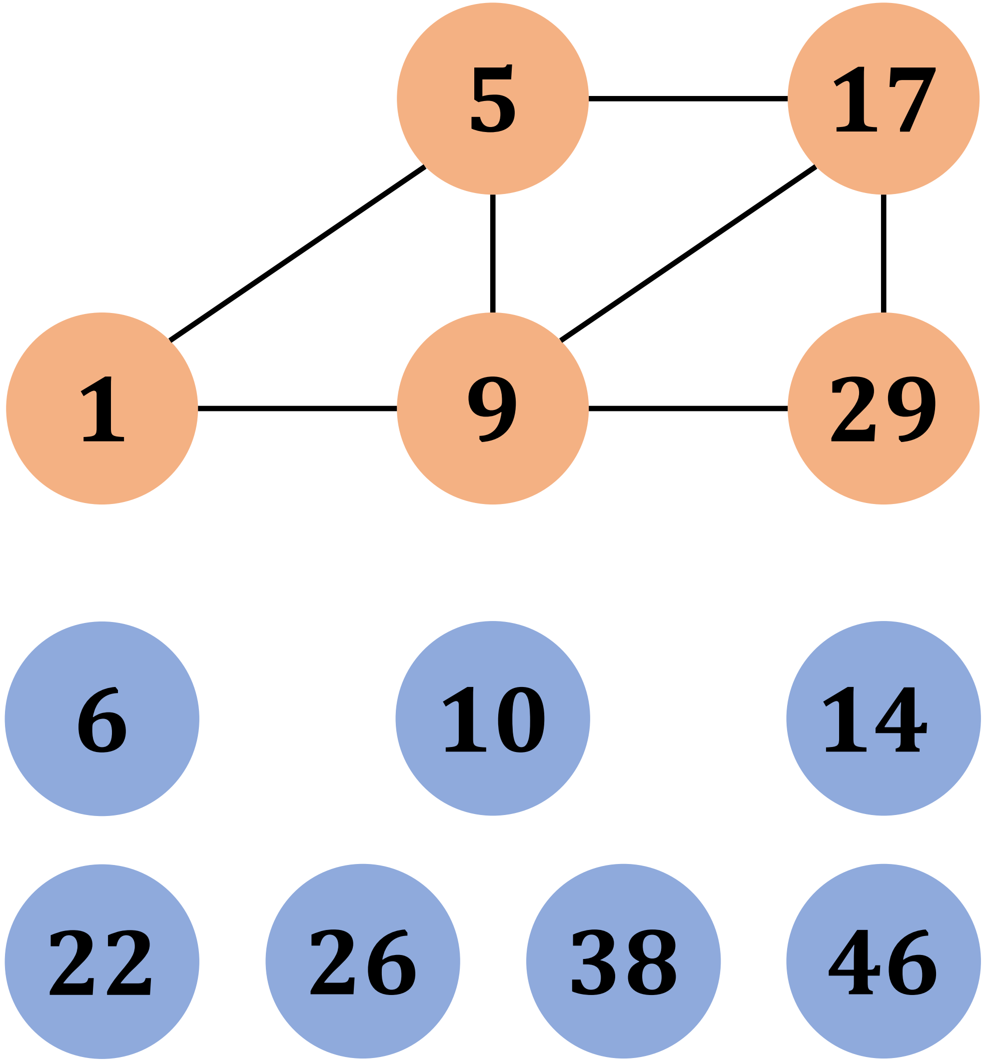
- All vertices of original graph are labelled $1 \bmod 4$.
- All isolated vertices are labelled $2 \bmod 4$.



Algorithm



- All vertices of original graph are labelled $1 \bmod 4$.
- All isolated vertices are labelled $2 \bmod 4$.

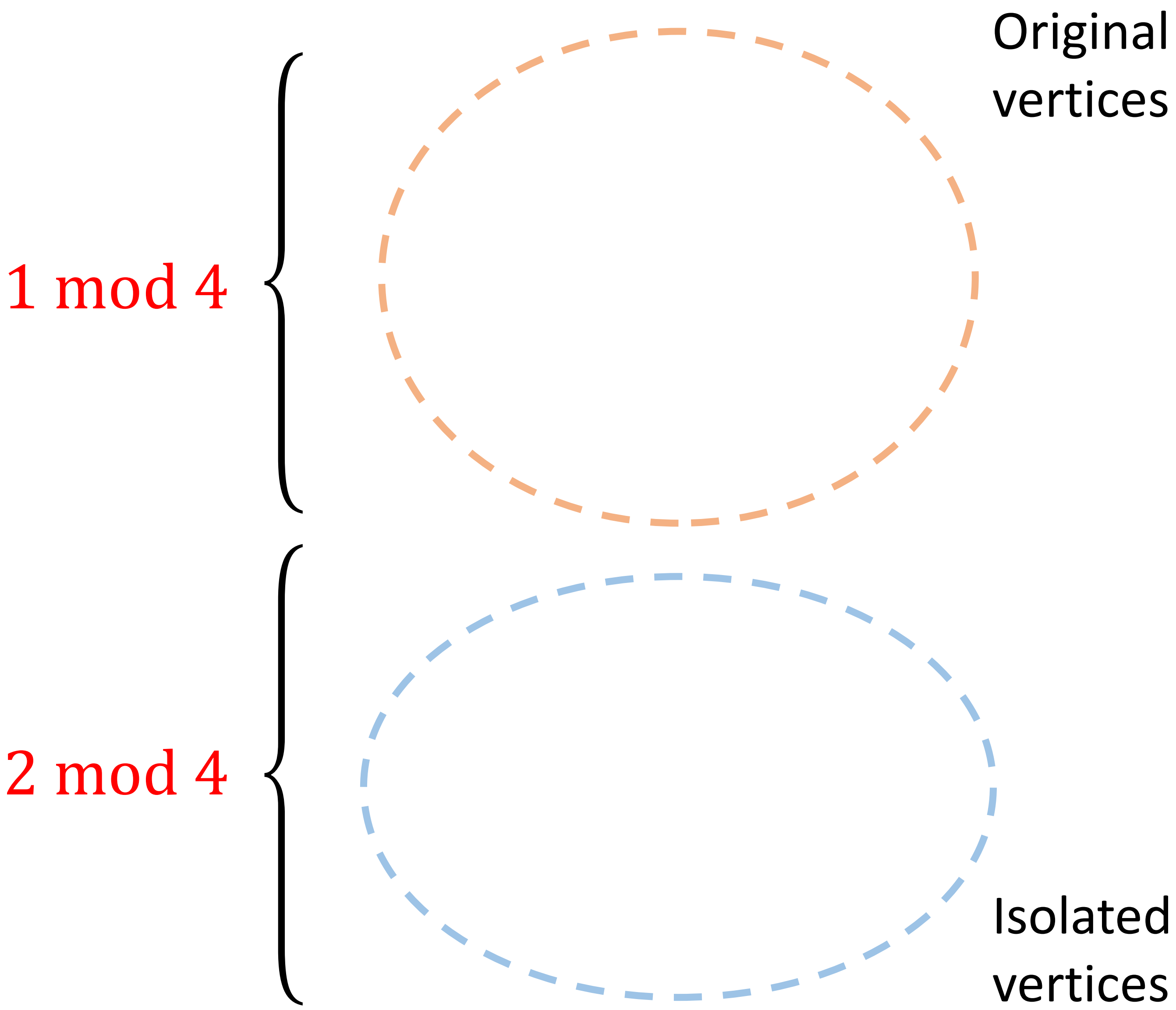


Analysis

Claim 1: Conflicting triples (a, b, c) are possible only when a and b are from the original graph and c is an isolated vertex.

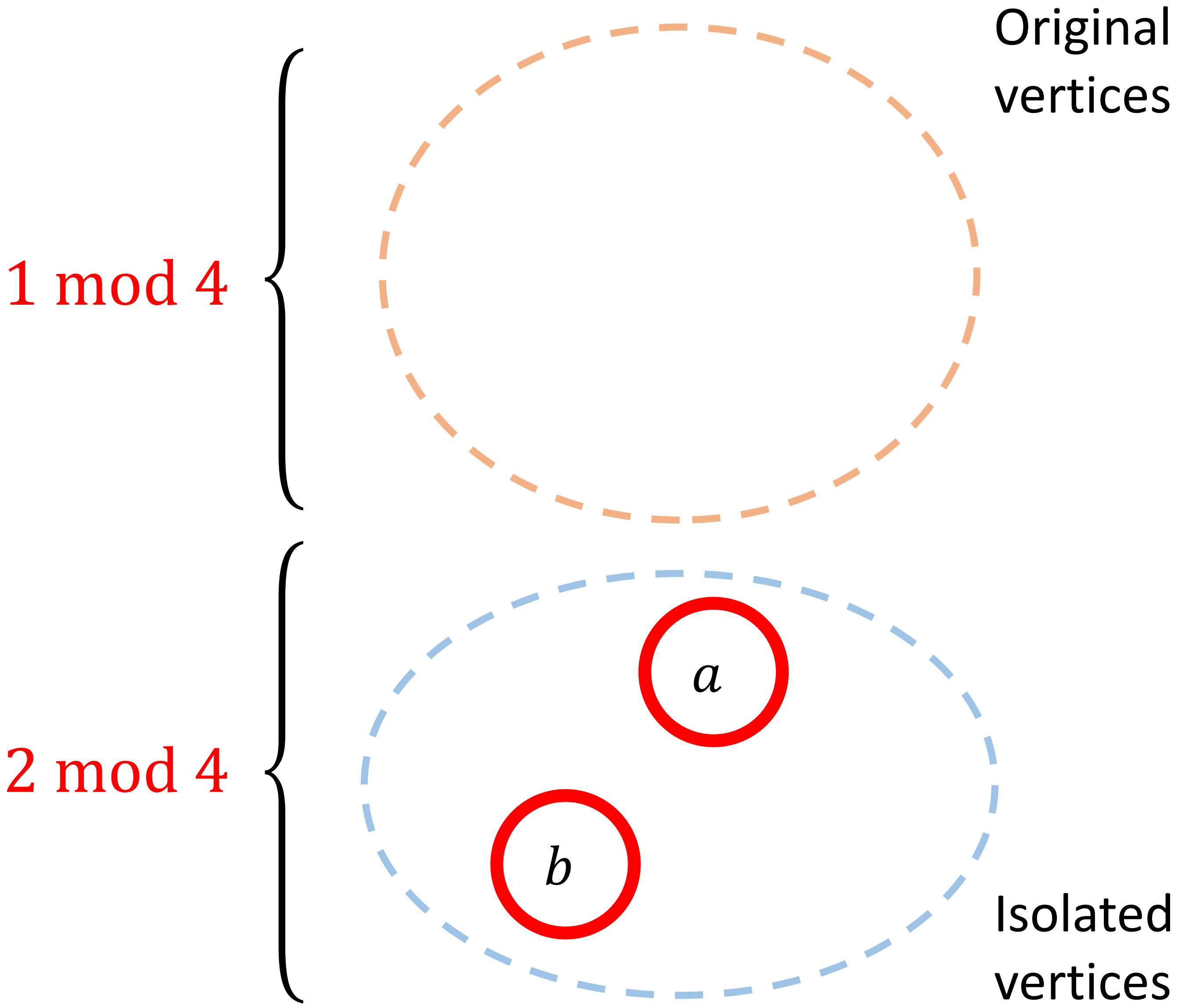
Analysis

Claim 1: Conflicting triples (a, b, c) are possible only when a and b are from the original graph and c is an isolated vertex.



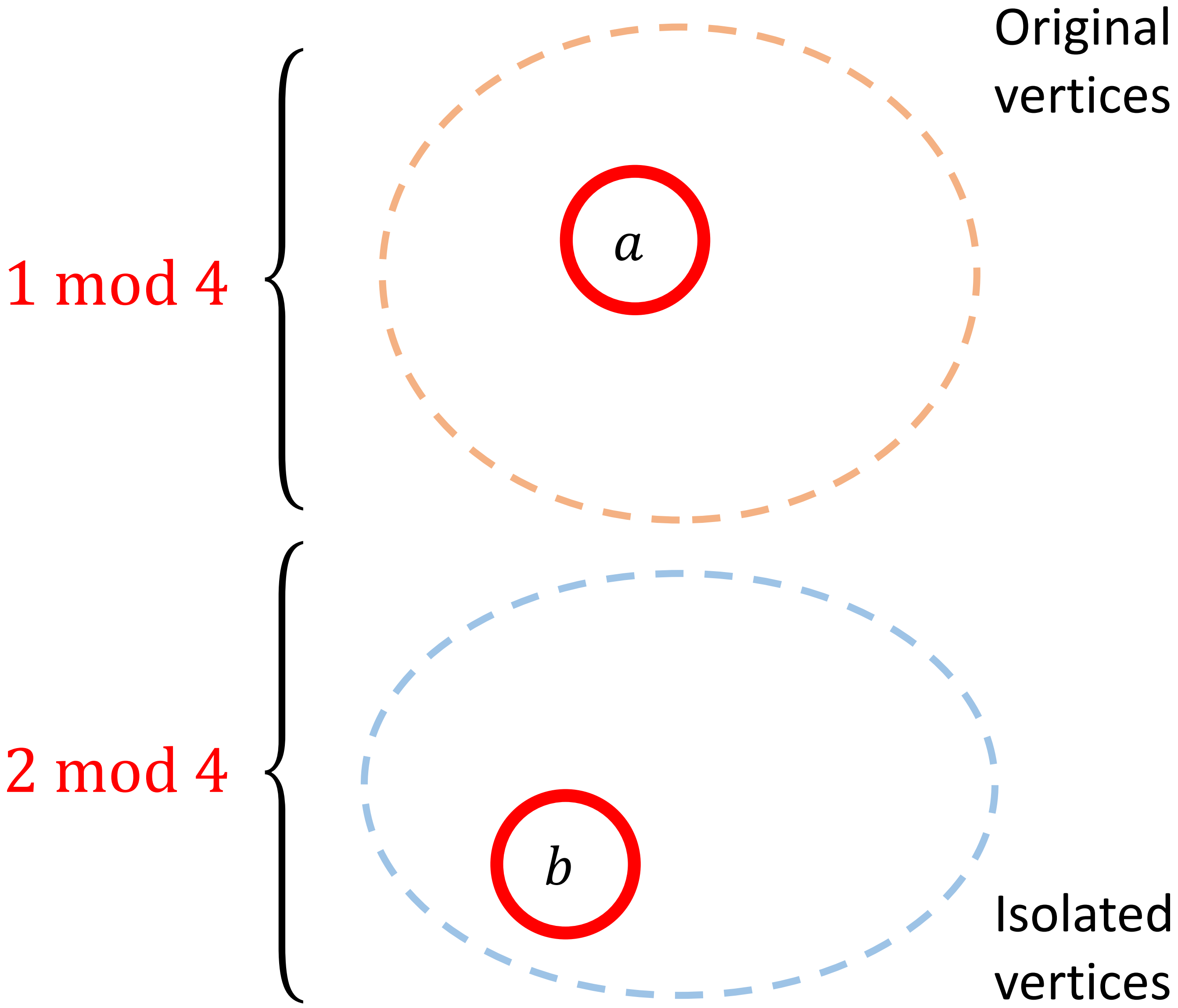
Analysis

Claim 1: Conflicting triples (a, b, c) are possible only when a and b are from the original graph and c is an isolated vertex.



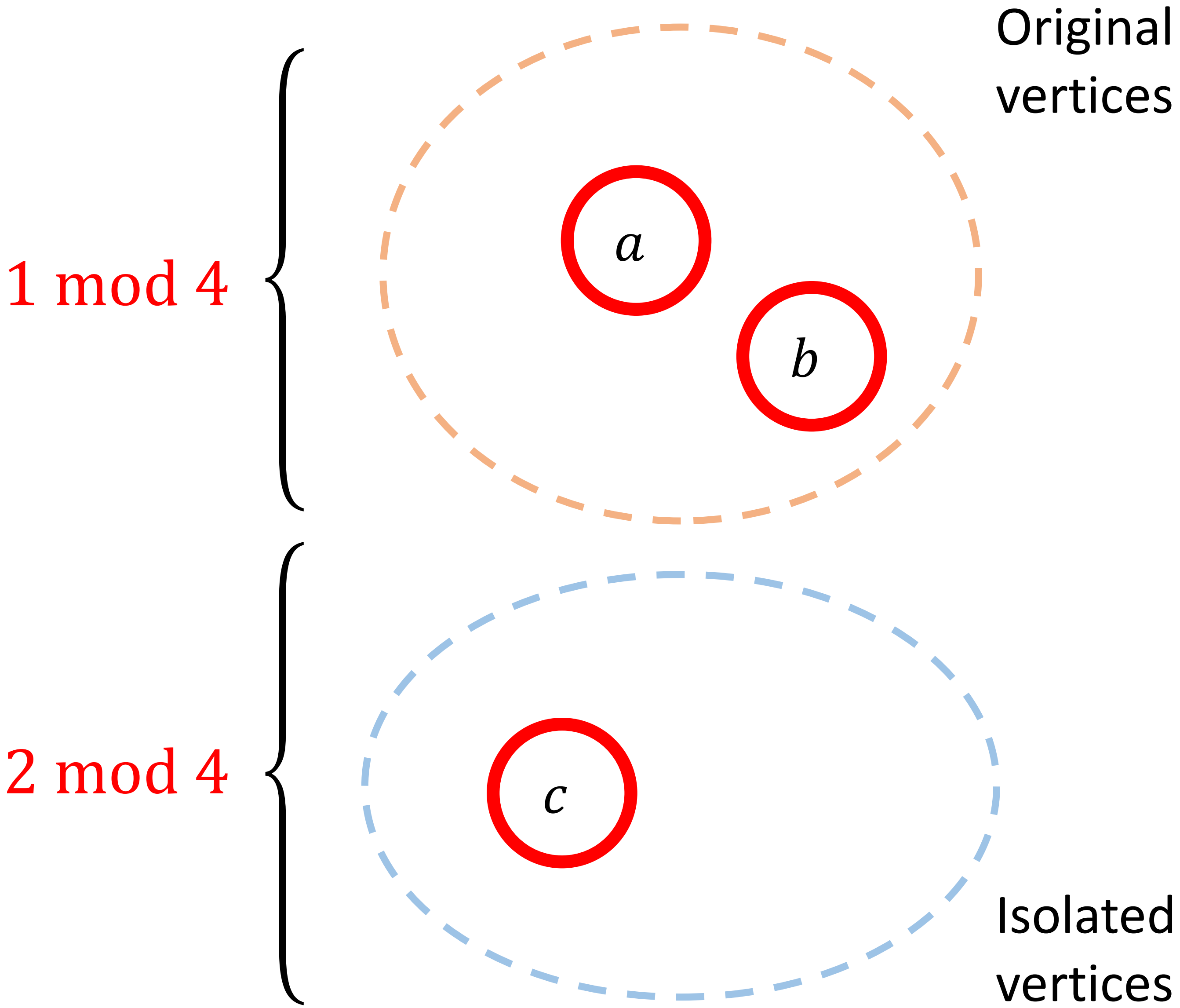
Analysis

Claim 1: Conflicting triples (a, b, c) are possible only when a and b are from the original graph and c is an isolated vertex.



Analysis

Claim 1: Conflicting triples (a, b, c) are possible only when a and b are from the original graph and c is an isolated vertex.



Analysis

Claim 2: For each conflicting triple (a, b, c), at most one side of the equality

$$a + b = c$$

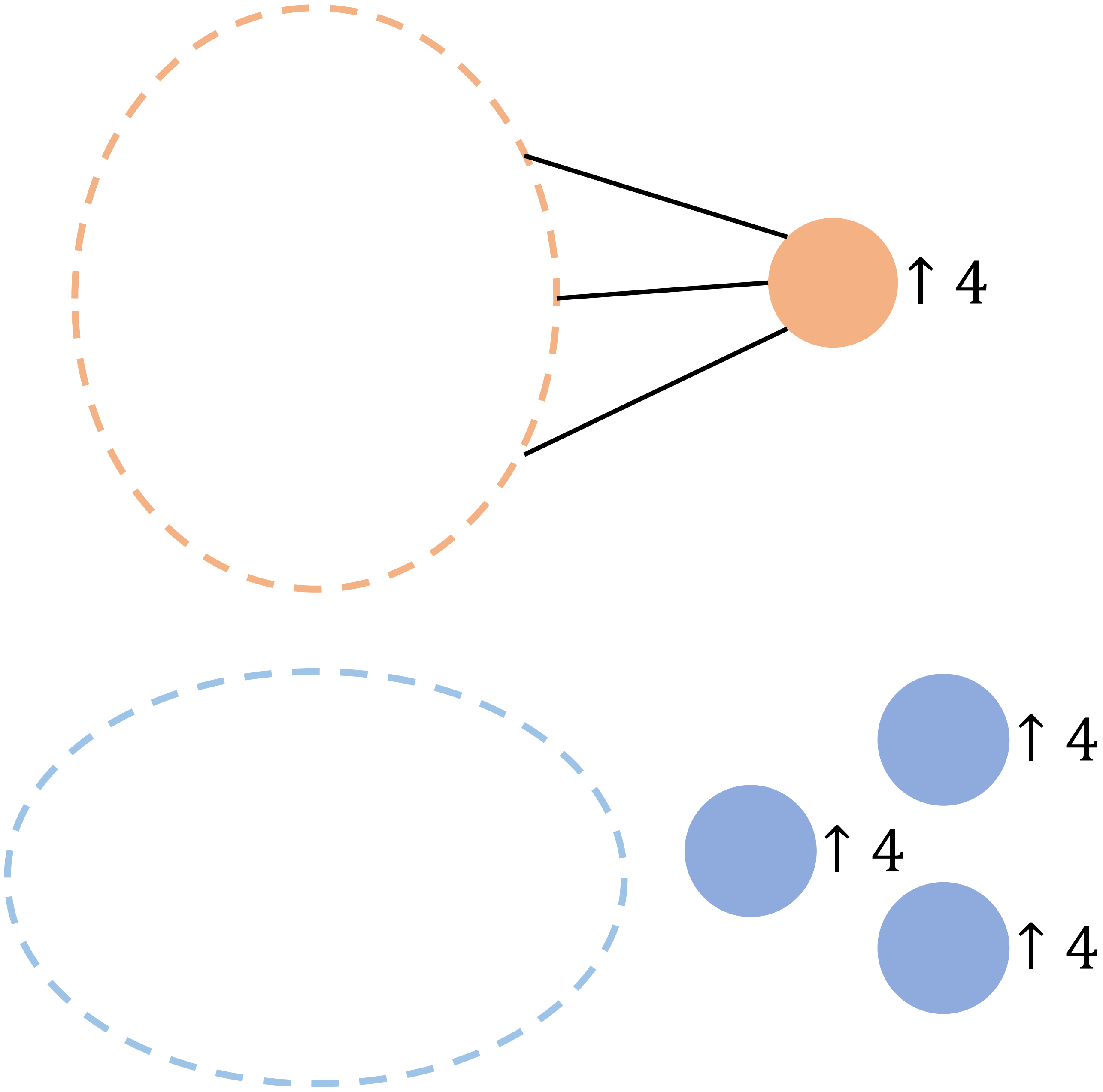
changes (increments by 4) in the label update step.

Analysis

Claim 2: For each conflicting triple (a, b, c), at most one side of the equality

$$a + b = c$$

changes (increments by 4) in the label update step.

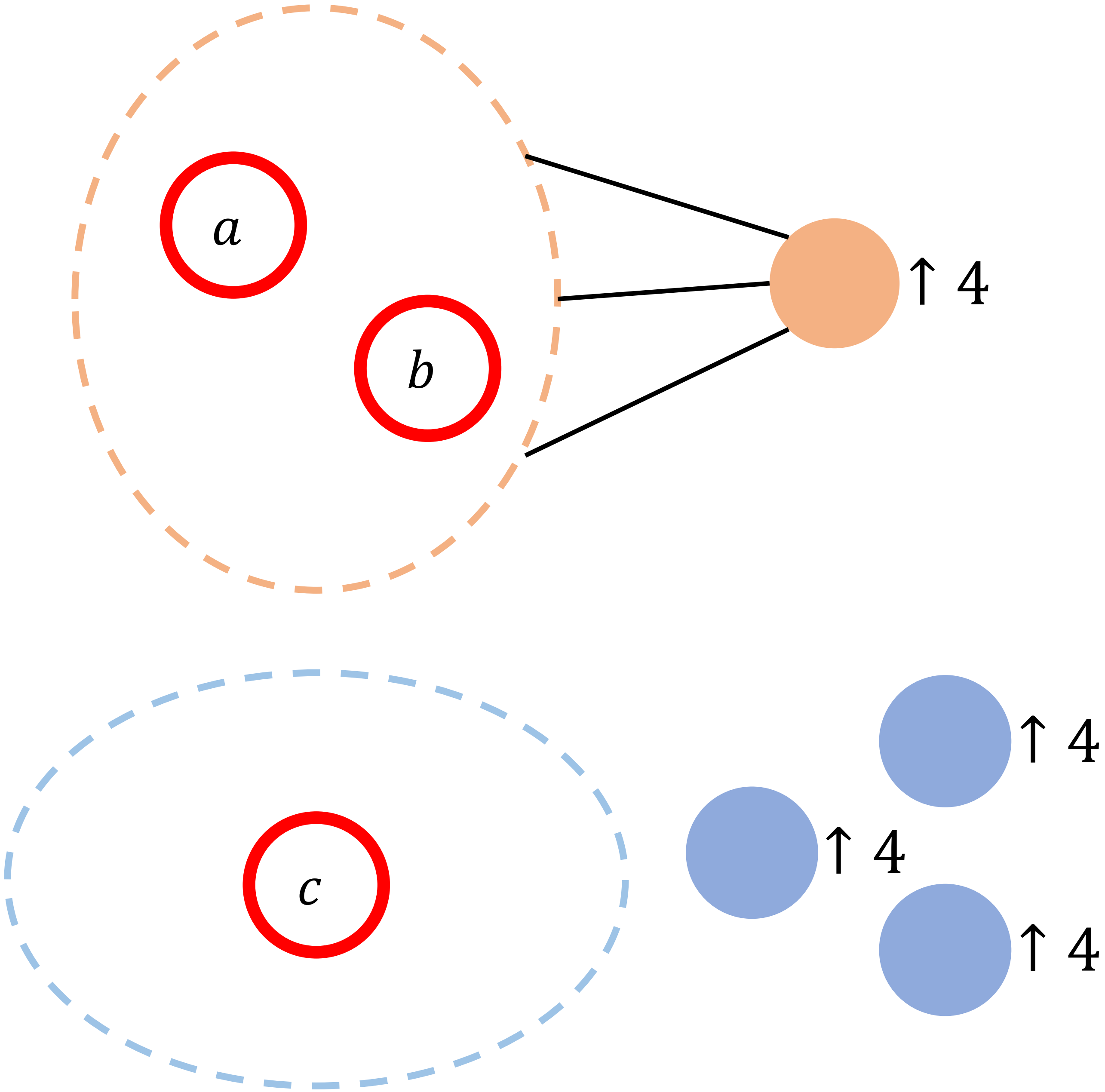


Analysis

Claim 2: For each conflicting triple (a, b, c), at most one side of the equality

$$a + b = c$$

changes (increments by 4) in the label update step.

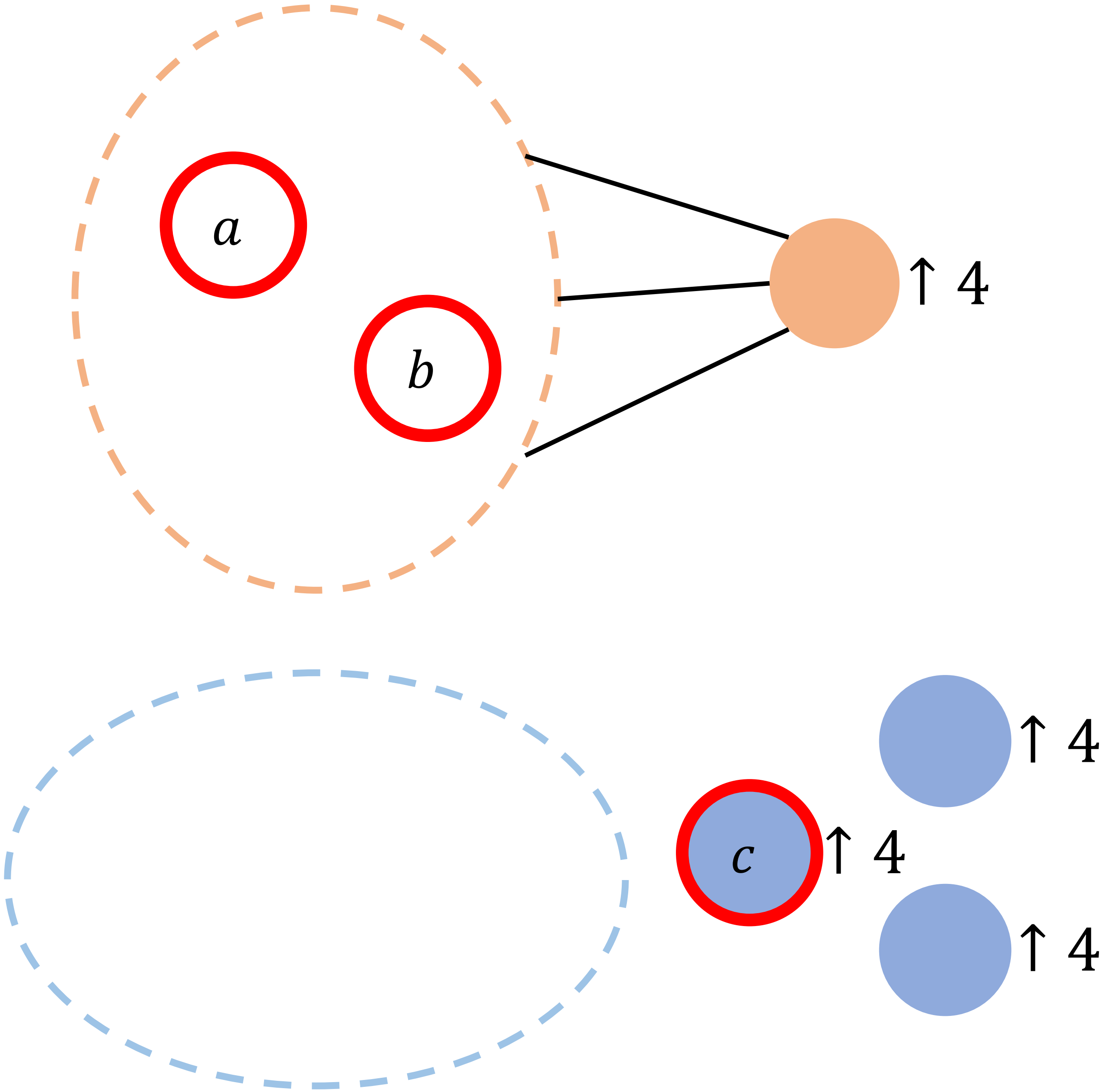


Analysis

Claim 2: For each conflicting triple (a, b, c), at most one side of the equality

$$a + b = c$$

changes (increments by 4) in the label update step.

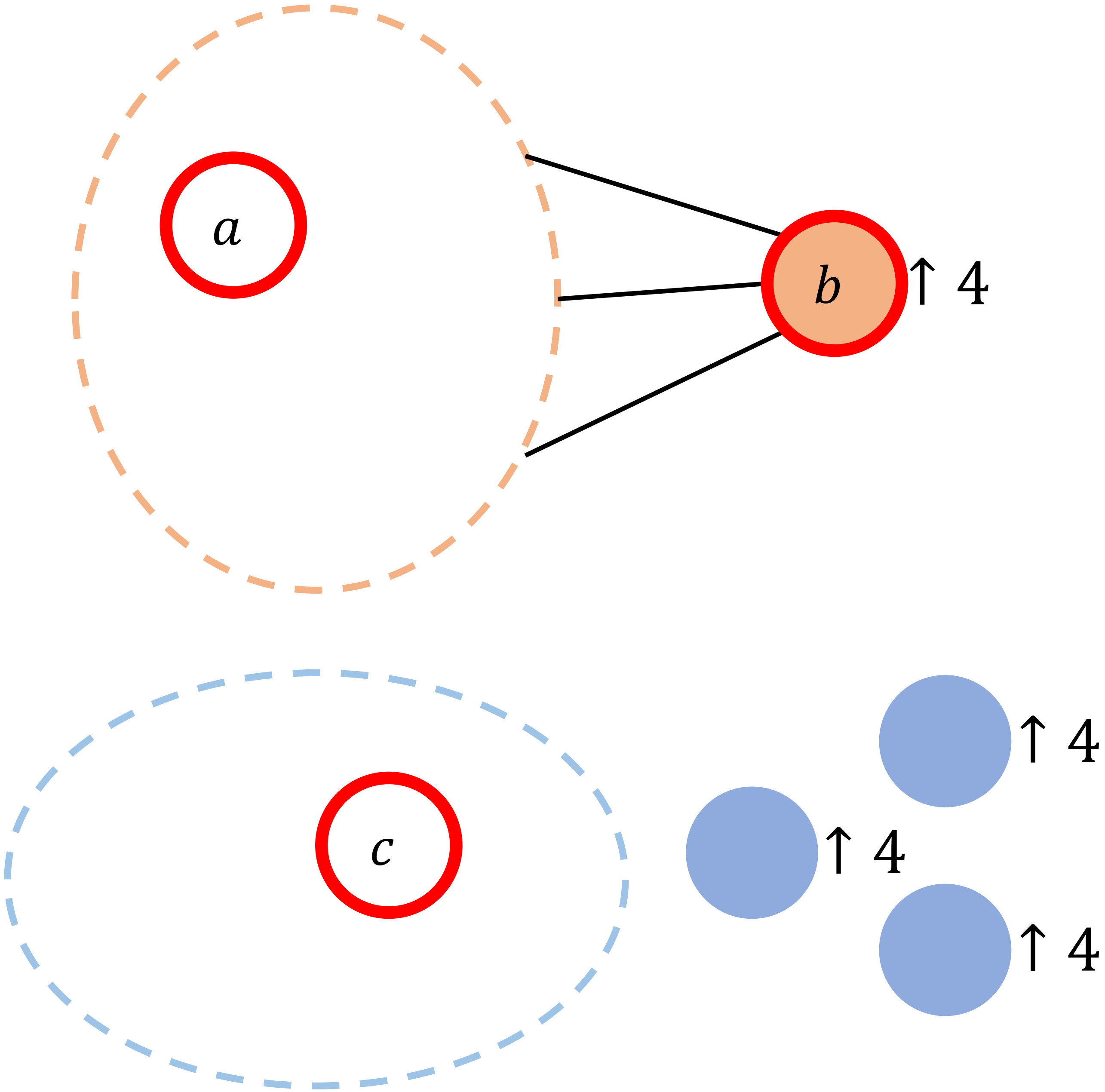


Analysis

Claim 2: For each conflicting triple (a, b, c), at most one side of the equality

$$a + b = c$$

changes (increments by 4) in the label update step.

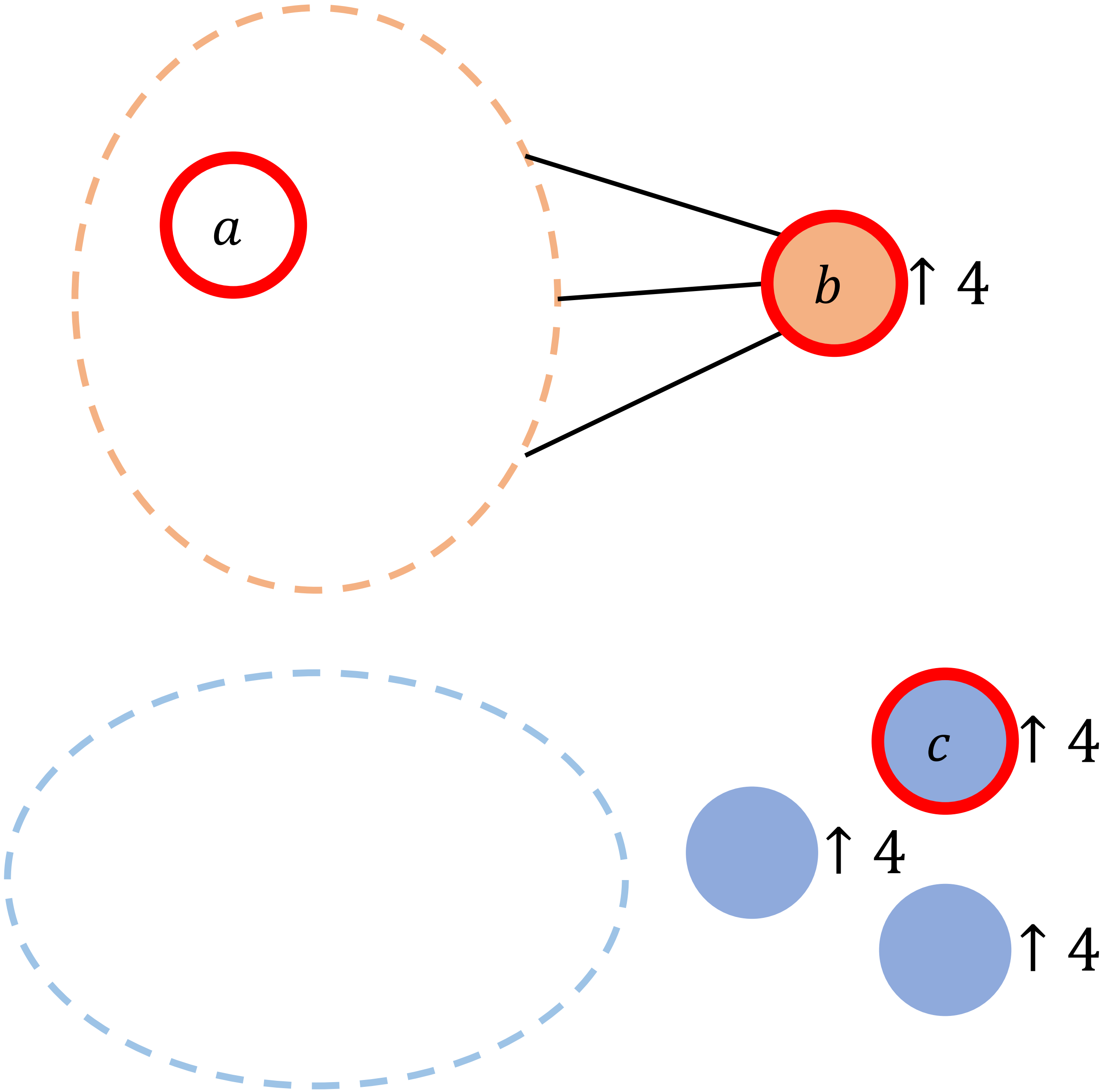


Analysis

Claim 2: For each conflicting triple (a, b, c), at most one side of the equality

$$a + b = c$$

changes (increments by 4) in the label update step.

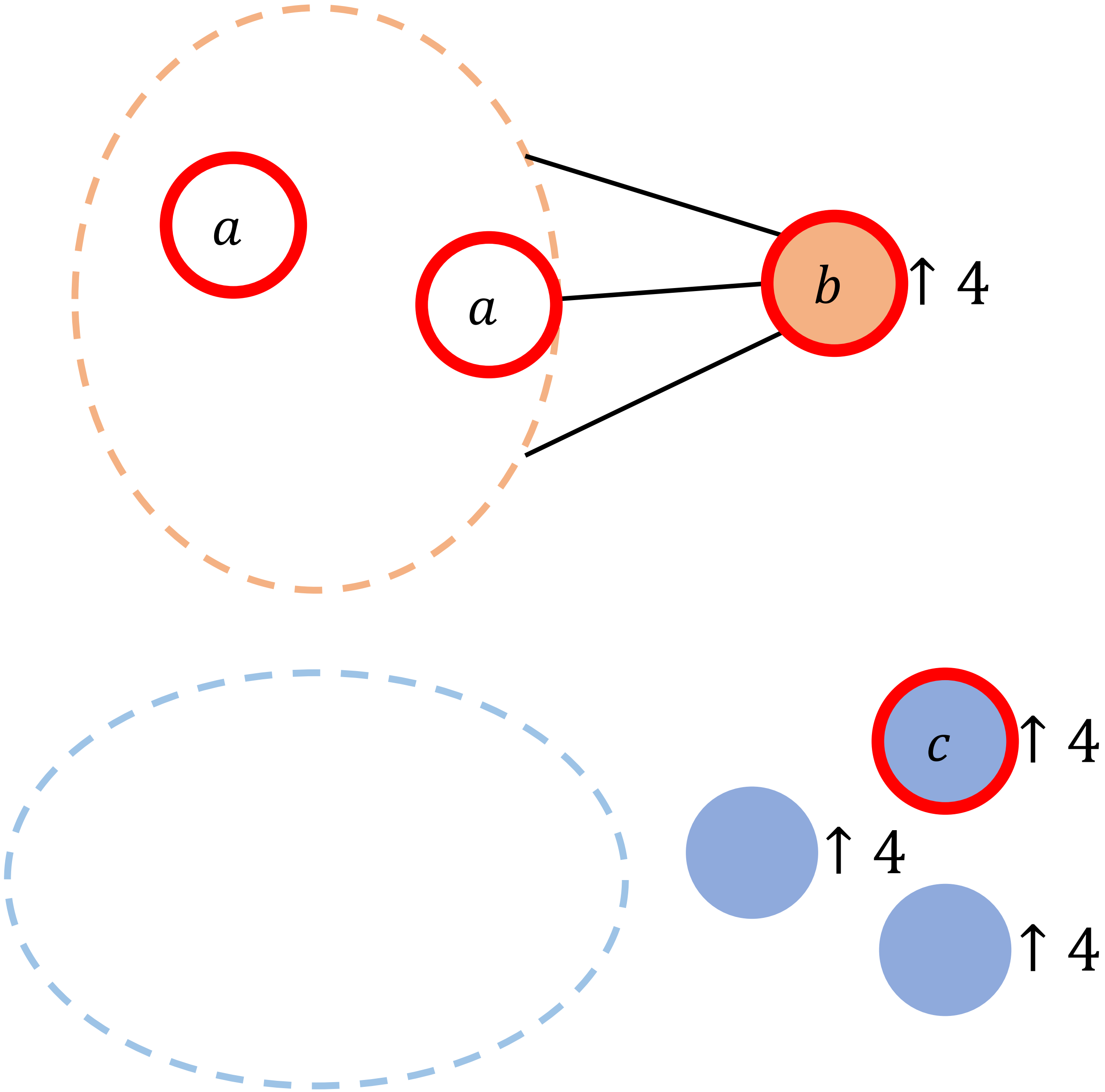


Analysis

Claim 2: For each conflicting triple (a, b, c), at most one side of the equality

$$a + b = c$$

changes (increments by 4) in the label update step.



Improving the Upper Bound

Definition A graph is called d-degenerate if every subgraph of the graph has a vertex of degree at most d. The degeneracy of a graph is the minimum d for which it is d-degenerate.

Fact Every forest is 1-degenerate.

Proof. Every forest has at least one tree, and every tree has at least one leaf.

Fact Every planar graph is 5-degenerate.

Proof. Every planar graph on n vertices has at most $3n - 6$ edges.

Our Result

Theorem [Fernau, G., 2021] Every n-vertex, m-edge, d-degenerate graph can be made a sum graph by adding at most m isolated vertices to it such that:

- The label of each vertex of the original graph is at most $6dn^2$.
- The label of each isolated vertex is at most $12dn^2$.

Implications

- Graphs with $O(n)$ edges: can be stored with $O(n \log n)$ bits, matching trivial lower bound of $\Omega(n \log n)$.

- Complete graphs: optimal labelling.

- Universal graphs:

  [Dujmovic *et al.*, 2020] For every positive integer n, there is a universal graph U_n on $n^{1+o(1)}$ vertices such that every n-vertex planar graph is an induced subgraph of U_n.

  [Fernau, G., 2021] Every n-vertex planar graph can be represented by a subset of the first $60n^2$ positive integers, $\{1,2,\ldots,60n^2\}$.

A New Variant of Sum Labelling!

Definition [Fernau, G., 2021] H is a supersum graph of G if H is a sum graph and G is an induced subgraph of H.

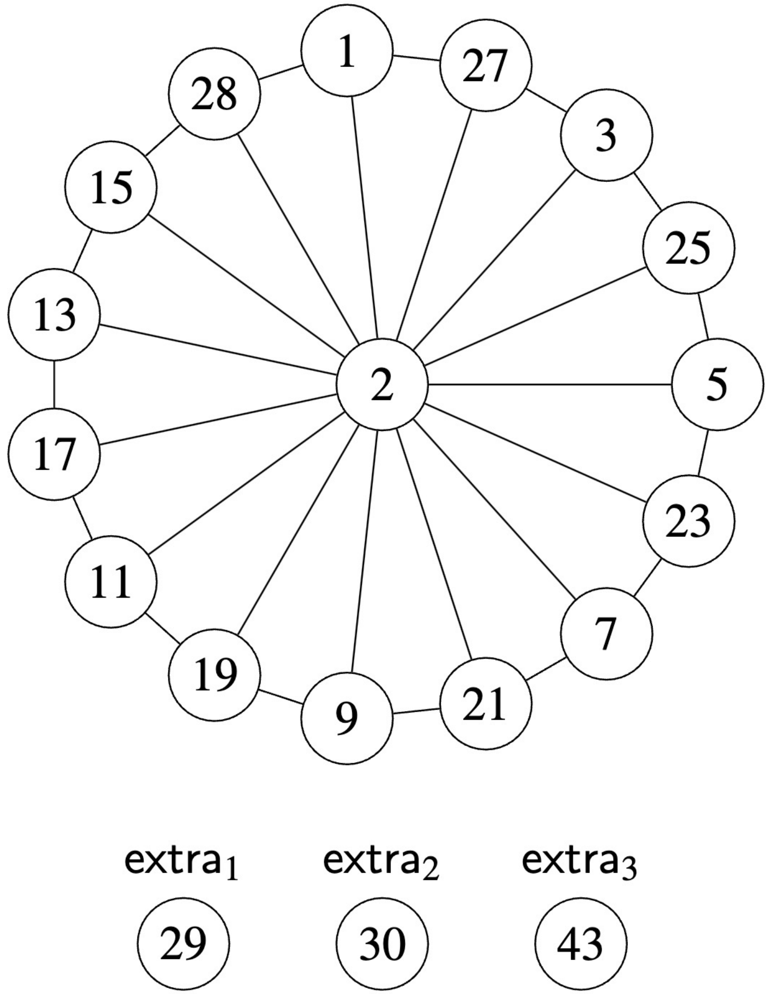
Given a graph G, let $H_{\min}$ be a supersum graph of G with the minimum number of vertices. Then the supersum number of G is

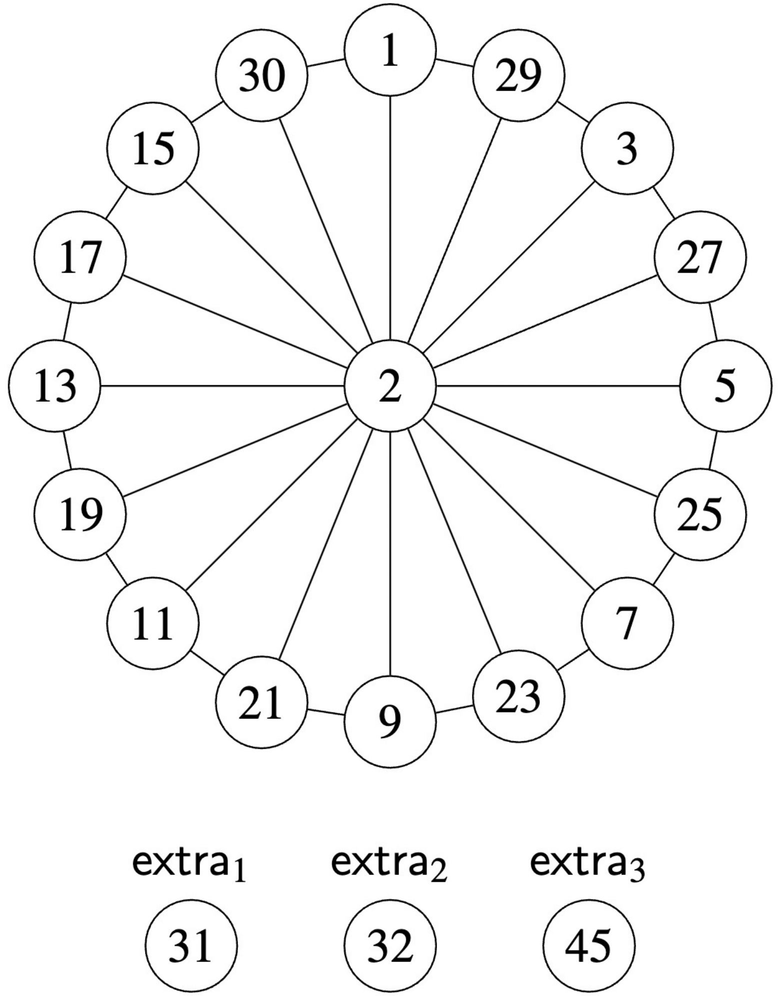
$$|V(H_{\min})| - |V(G)|.$$

Wheel Graphs

Theorem [Miller, Ryan, Slamin, Smyth, 1998] The sum number of the n-vertex wheel graph is $\Theta(n)$.

Theorem [Fernau, G., 2021] The supersum number of the n-vertex wheel graph is 3.





Conclusion

Are there polynomial-time algorithms for (any of) the following problems?

- Determine the sum number of a given graph.
- Given a sum graph, find a sum labelling for it.
- Perform graph operations/algorithms without actually constructing the edges of a given sum graph, by looking at the list of its vertex labels.
- Given a directed graph with a sum labelling of its underlying undirected graph, relabel it so as to also preserve the orientations of the edges.

Thank You!