Exacting Eccentricity for Small-World Networks

Wentao Li[1], Miao Qiao[2], Lu Qin[1], Ying Zhang[1], Lijun Chang[3], and Xuemin Lin[4]

[1]CAI, FEIT, University of Technology Sydney, Australia

[2]Massey University, New Zealand

[3]The University of Sydney, Australia

[4]The University of New South Wales, Australia

ICDE PARIS 2018

Outline

- Background
- Existing Solutions
- Approach
- Performance Studies
- Conclusion

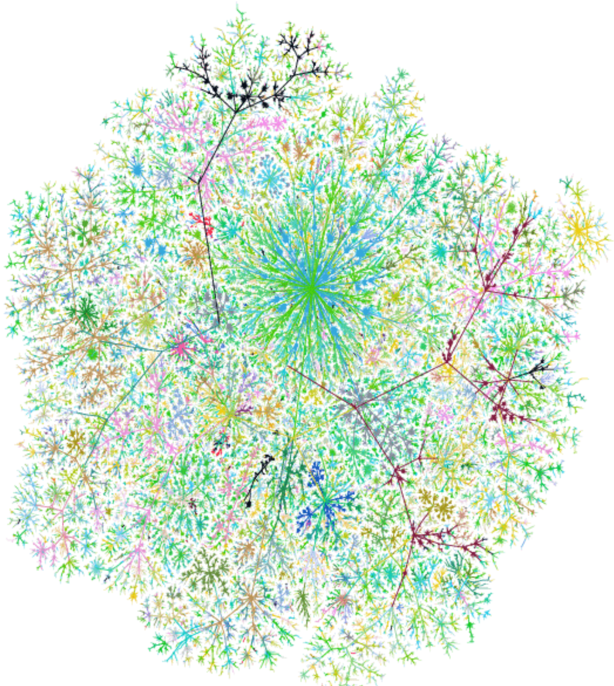
Background

- Exacting Eccentricity for Small-World Networks

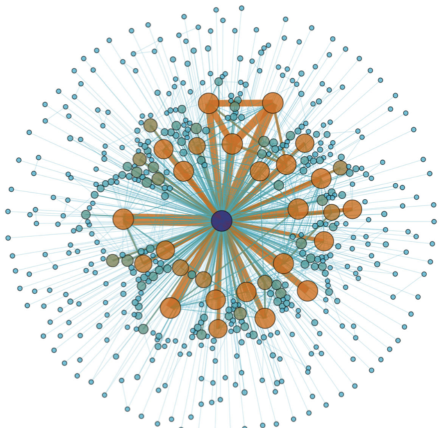
Small-World Networks

- Graphs that feature the highly clustered topology, low average pairwise distances, etc.

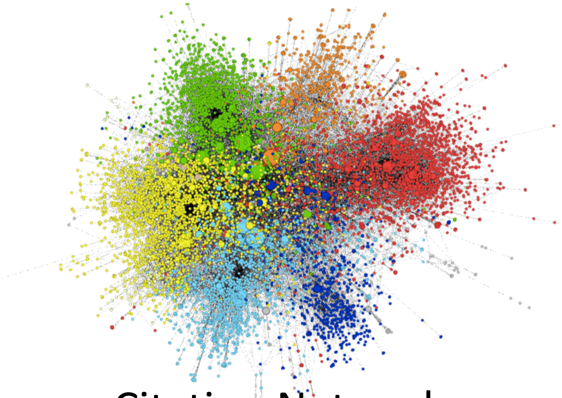
Social Network



Web Graph



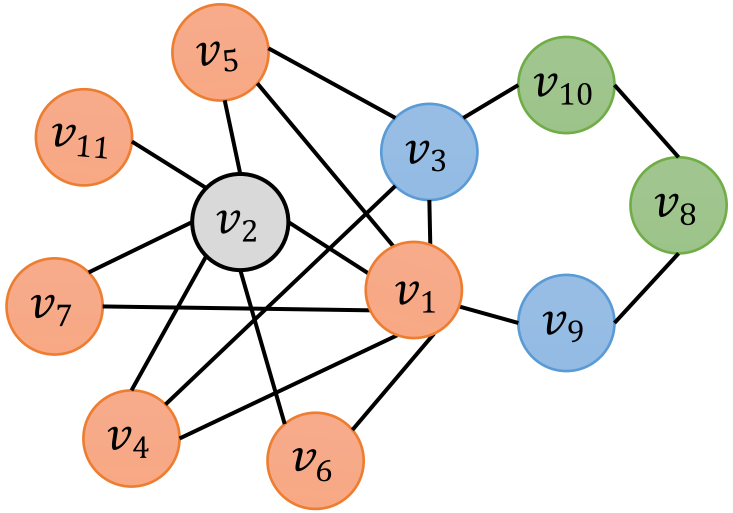
Collaboration Network



Citation Network

Eccentricity

- Given a small world network $G(V, E)$, the **distance** $d(u, v)$ is the length of the shortest path between u and v in G.



Distance Information of v_2

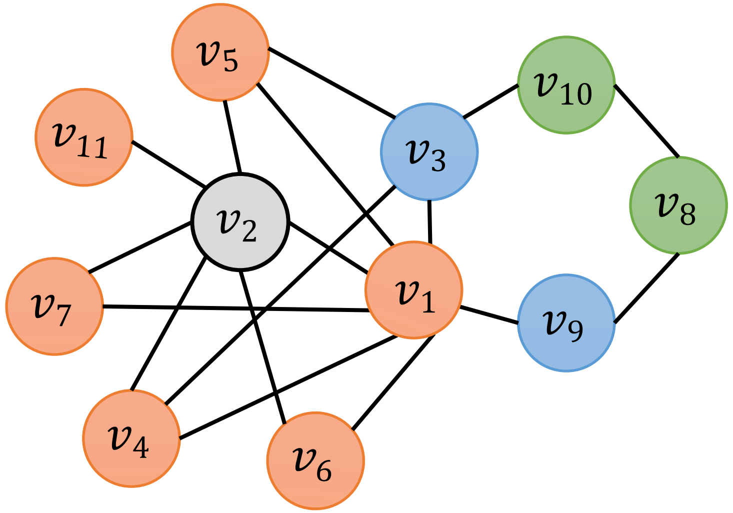
$d = 1: \{v_1, v_4, v_5, v_6, v_7, v_{11}\}$

$d = 2: \{v_3, v_9\}$

$d = 3: \{v_8, v_{10}\}$

Eccentricity

- Given a small world network $G(V, E)$, the **distance** $d(u, v)$ is the length of the shortest path between u and v in G.
- The **eccentricity** $ecc(v)$ is the ***largest distance*** from the node v to any other reachable nodes.



Distance Information of v_2

$d = 1$: $\{v_1, v_4, v_5, v_6, v_7, v_{11}\}$

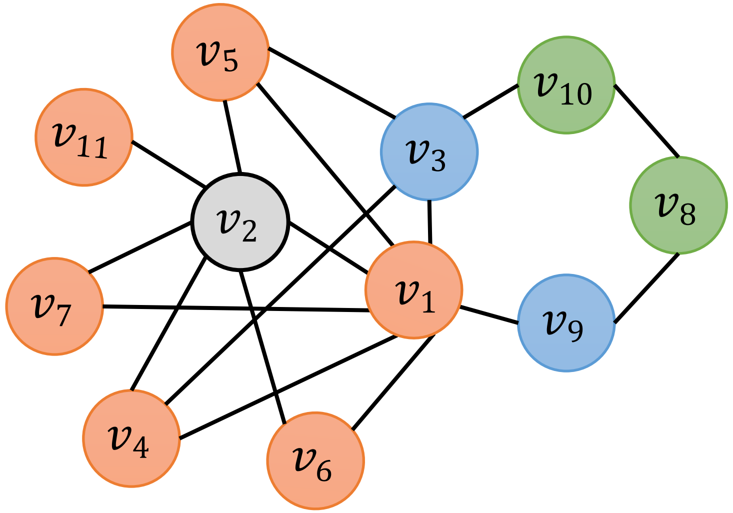
$d = 2$: $\{v_3, v_9\}$

$d = 3$: $\{v_8, v_{10}\}$

$ecc(v_2) = 3$

Eccentricity

- Given a small world network $G(V,E)$, the **distance** $d(u,v)$ is the length of the shortest path between u and v in G.
- The **eccentricity** $ecc(v)$ is the ***largest distance*** from the node v to any other reachable nodes.
- **Problem Statement**
    - Given a graph $G(V,E)$, compute the eccentricity-distribution, namely, the eccentricity $ecc(u)$ for all the nodes $u \in V$.



Distance Information of v_2

$d = 1: \{v_1, v_4, v_5, v_6, v_7, v_{11}\}$

$d = 2: \{v_3, v_9\}$

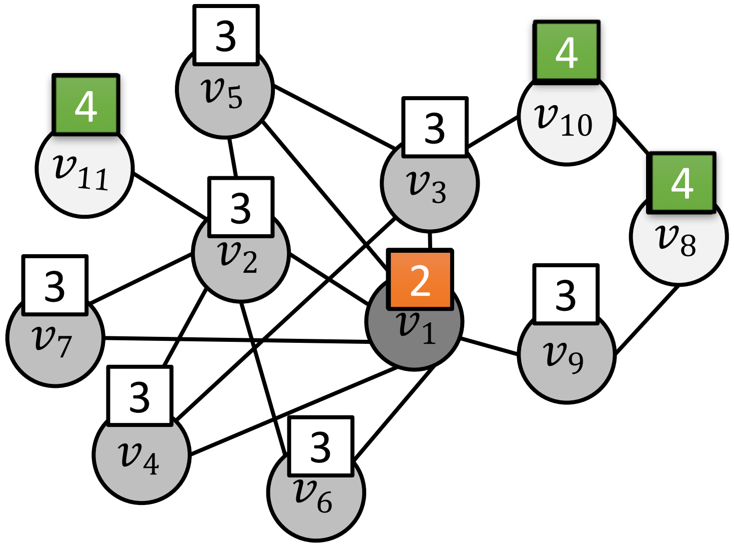
$d = 3: \{v_8, v_{10}\}$

$ecc(v_2) = 3$

Unweighted, Undirected Graphs

Applications

- Vital node identifying
  - Influential people in a social network
  - Critical nodes in an epidemic contact network
  - Important sites in a web graph
- Computing radius, diameter
  - Radius = $min_{v \in V} ecc(v)$
  - Diameter = $max_{v \in V} ecc(v)$



Existing solutions

Naïve Algorithms

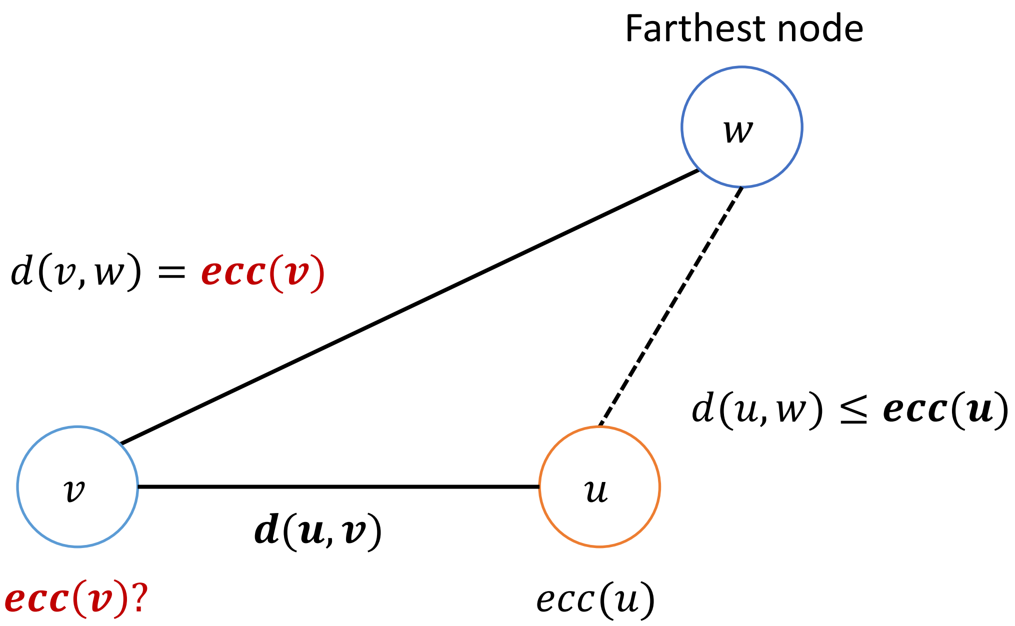
- The **eccentricity** $ecc(v)$ is the ***largest distance*** from the node v to any other reachable nodes.
- Exact
  - Apply the all-pairs shortest path (APSP) algorithm to find the eccentricity of all nodes.
  - Time complexity $O(mn)$: the size of n and m causes an efficiency issue.
- Approximation
  - There are some works on approximate the eccentricity [stoc 2013, soda 2014].
  - Approximation algorithms may lead to undesirable errors for networks with small diameter.

The state-of-the-art: BoundEcc

- Idea
  - Associates each node v with an upper and a lower bound on its eccentricity;
    - Lower bound $\leq$ eccentricity $\leq$ upper bound
  - For each node v
    - if the upper and lower bounds of v do not meet, compute the eccentricity $ecc(v)$ using a Breadth-First-Search (BFS)
    - and then update the bounds globally for all other nodes.

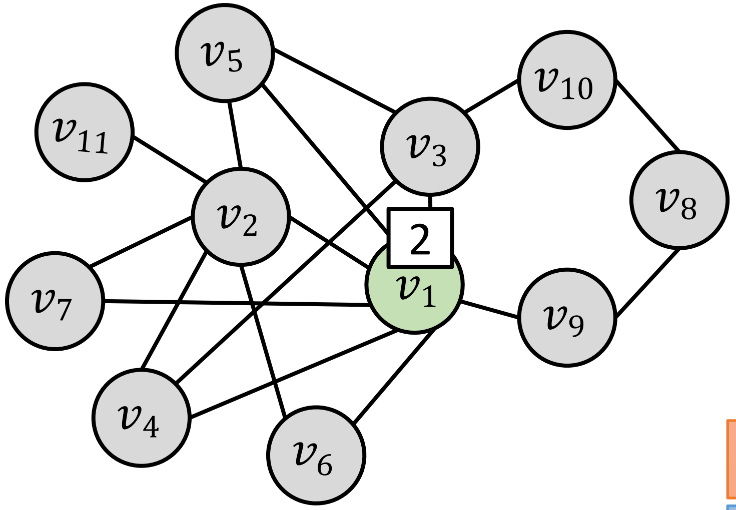
Bound Update Rule

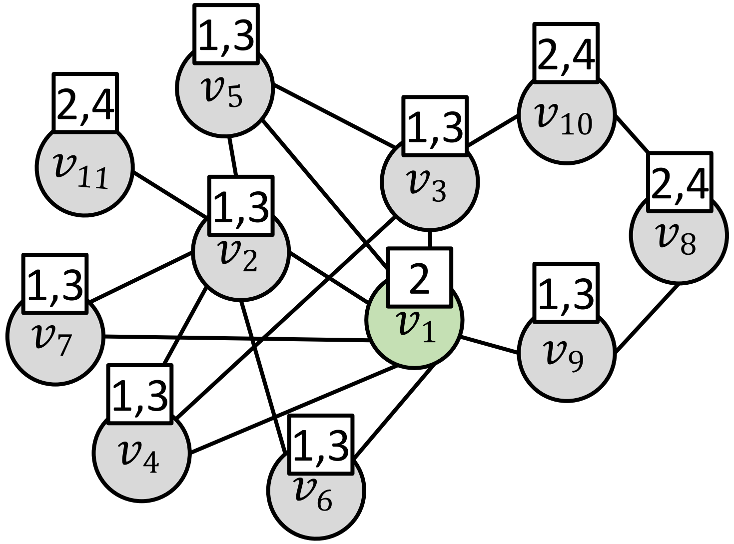
- Lemma (Eccentricity bound)
  - Let u be a node of Graph G with known eccentricity $ecc(u)$. Given a node v and its distance $d(u,v)$ to u:
    - $ecc(v) \leq ecc(u) + d(u,v)$
    - $ecc(v) \geq ecc(u) - d(u,v)$
    - $ecc(v) \geq d(u,v)$



Why it works?

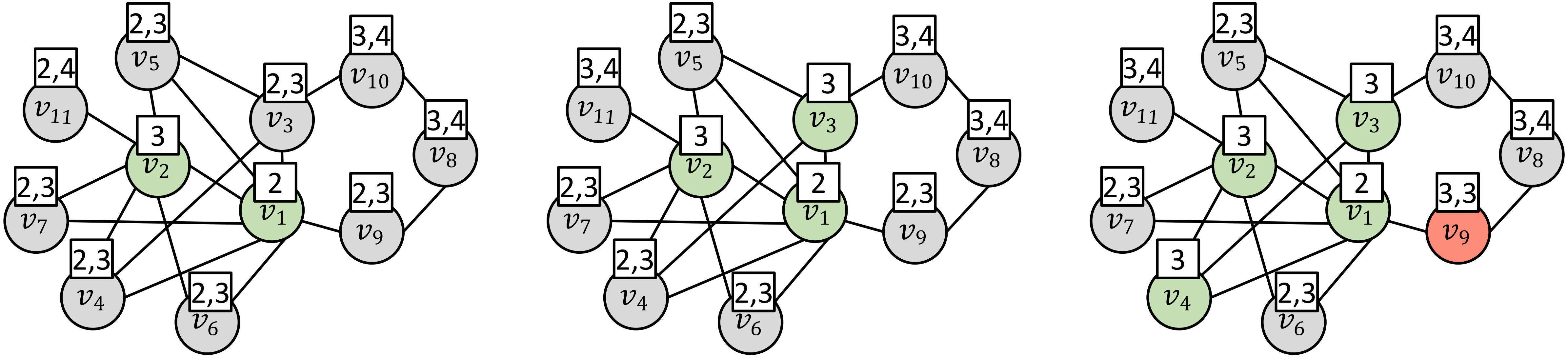
- **Round 1**
  - v_1





$d = 1$: $\{v_2, v_3, v_4, v_5, v_6, v_7, v_9\}$

$d = 2$: $\{v_8, v_{10}, v_{11}\}$



v_2 v_3 v_4

√ BoundEcc prunes the node with equal bounds.

✕ BoundEcc still needs lot of BFS.

Approach

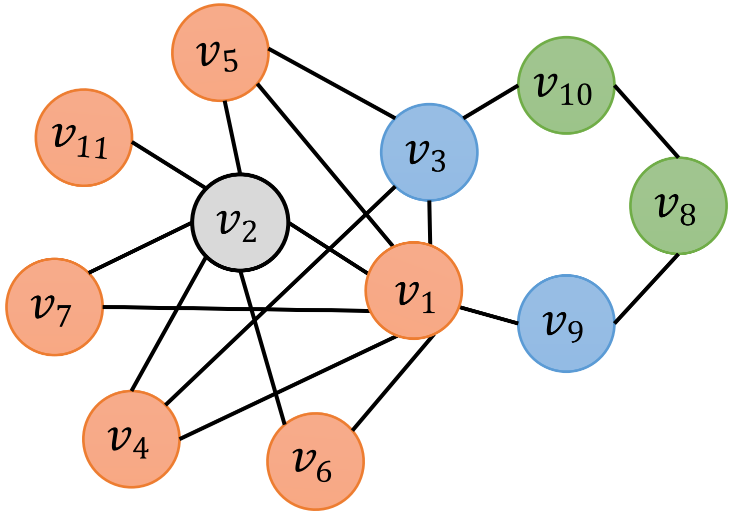
Motivation

- Drawback

BFS has to traverse $\boldsymbol{n}$ nodes in G to determine $ecc(v)$.

- Question

Can we traverse $\ll \boldsymbol{n}$ nodes in G to determine $ecc(v)$?



Idea 1: 2-hop Distance Labeling

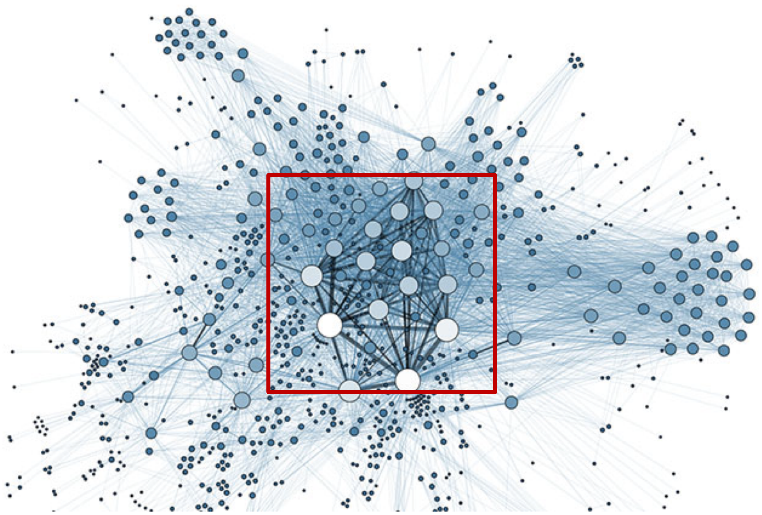
- **To answer Pair-Wise Shortest Distance query faster than BFS.**
- 2-hop distance labeling methods
  - Each node is assigned with the labels.
  - The pair-wise distance between two nodes can be obtained by using their labels.

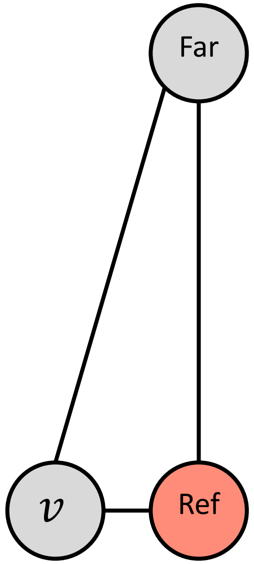
Method	Paper
Greedy	[Cohen, E.,2003]
HC-L	[Jing, R., 2012]
IS-L	[Fu, A.W., 2013]
PLL	[Akiba, T., 2013]
Hop-D	[Jiang, M., 2014]
PHL	[Akiba, T., 2014]

Idea 2: Reference Nodes

- **To find the farthest nodes without traveling the entire graph.**
- Observation 1: There exists some ***reference nodes*** that are near to all the nodes in a small-world network.
- Observation 2: If a node v is near a reference node, then the ***farthest nodes*** of the reference node will not be too close to v.

Let the reference node to guide the visiting of nodes for v.





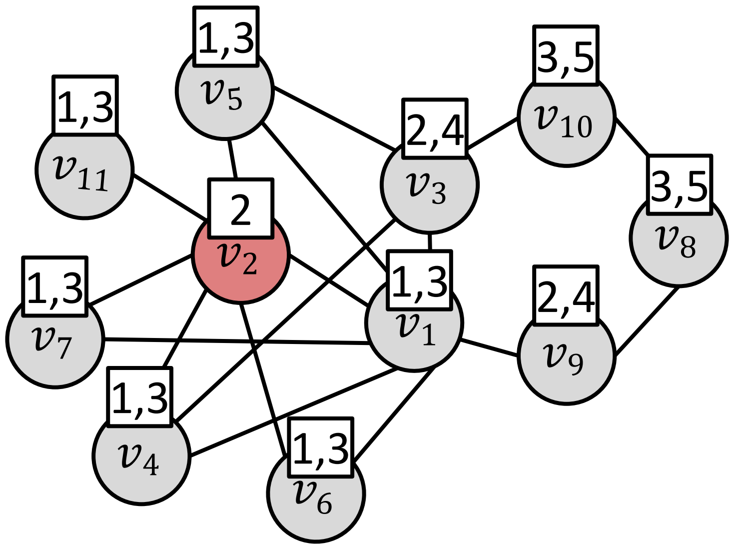
One Reference Node

- The reference node v_2
  - *How to select reference nodes will be introduced later.*

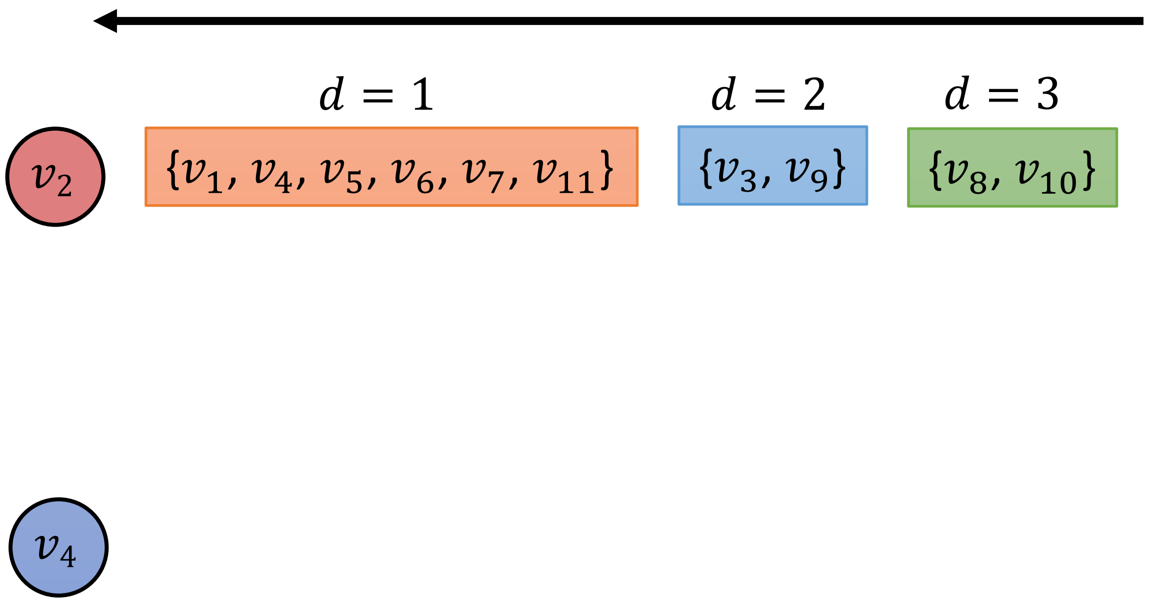
The farthest nodes of v_2 are more likely to be far to other nodes.

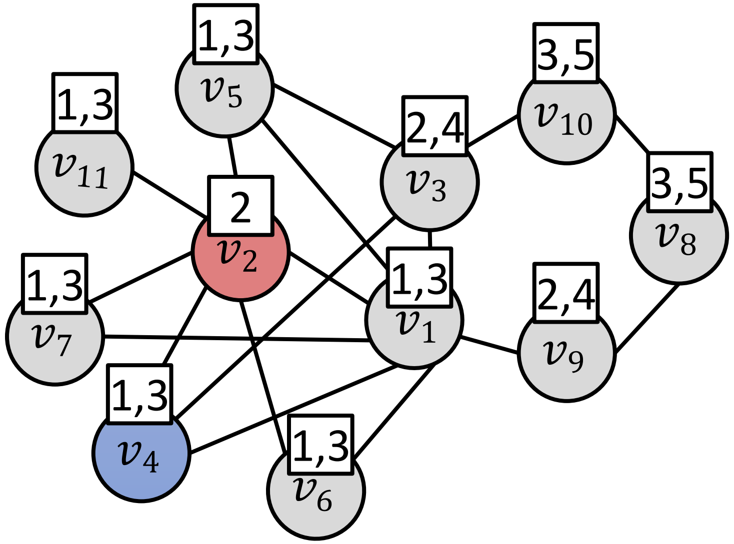
	$d = 1$	$d = 2$	$d = 3$
v_2	$\{v_1, v_4, v_5, v_6, v_7, v_{11}\}$	$\{v_3, v_9\}$	$\{v_8, v_{10}\}$

Guide the visiting of nodes from farthest to nearest.

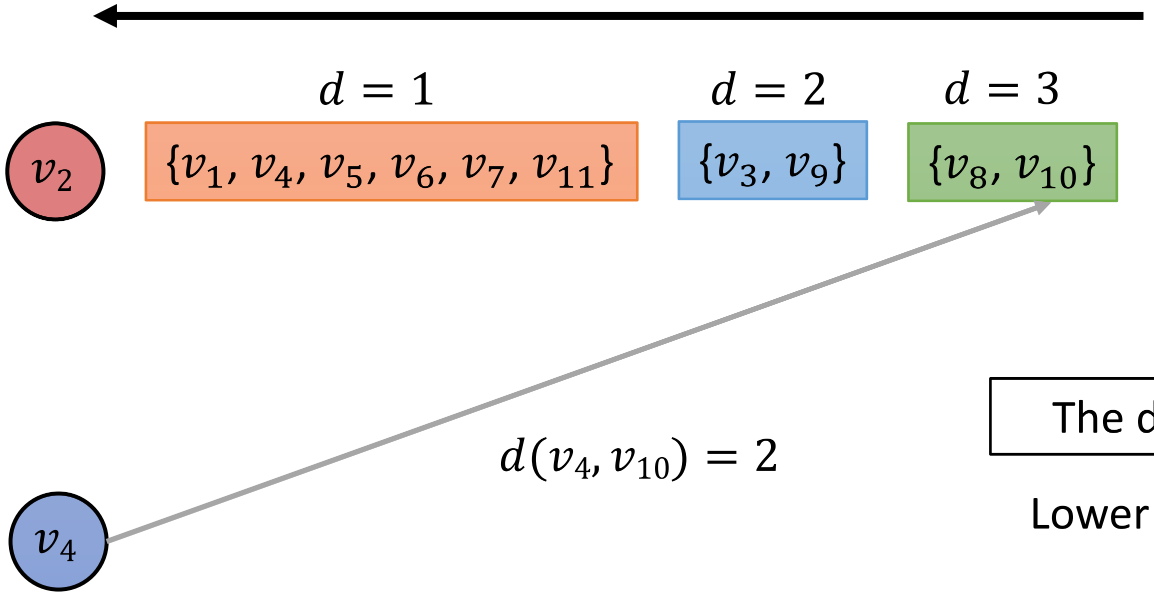


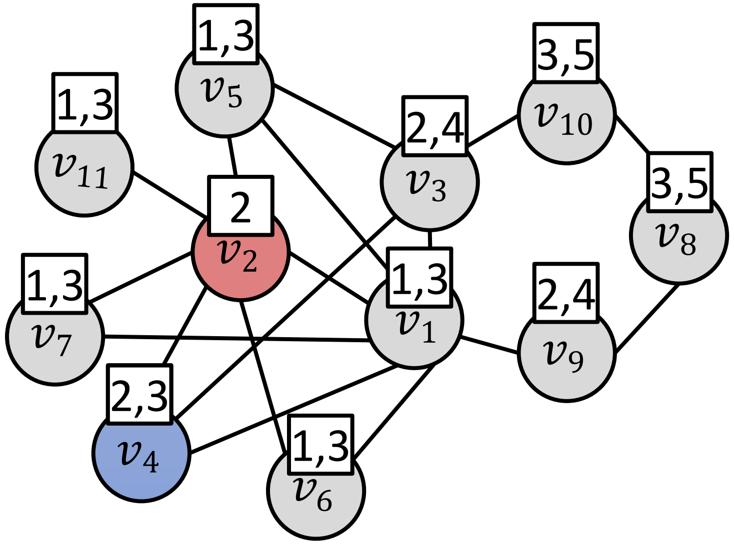
- **Example: Node v_4**





- **Example: Node v_4**
  - **Round 1**



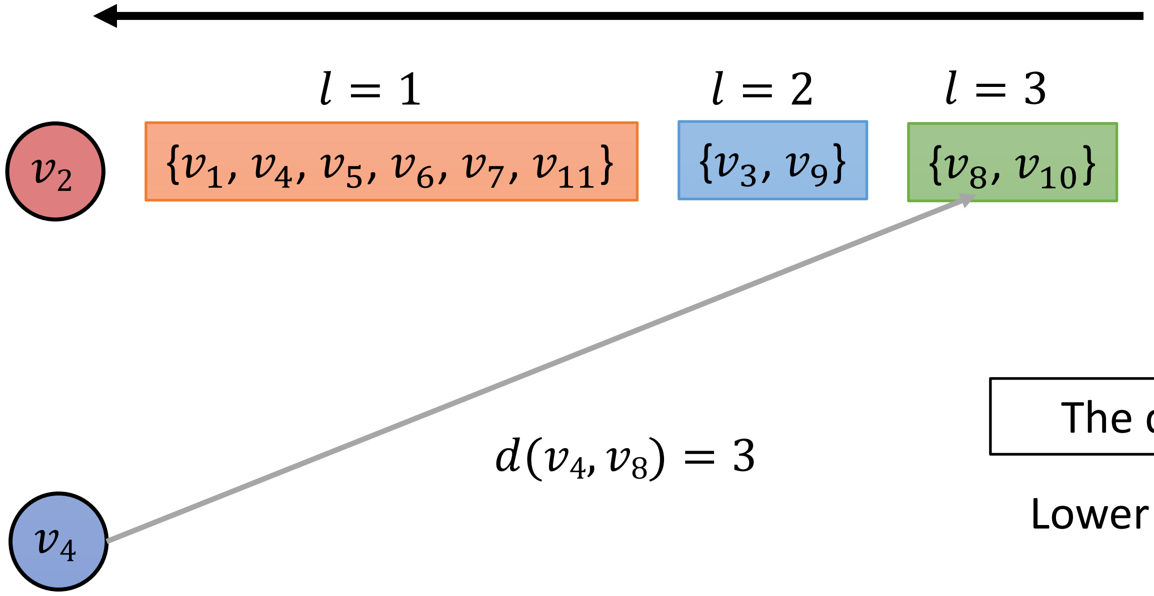


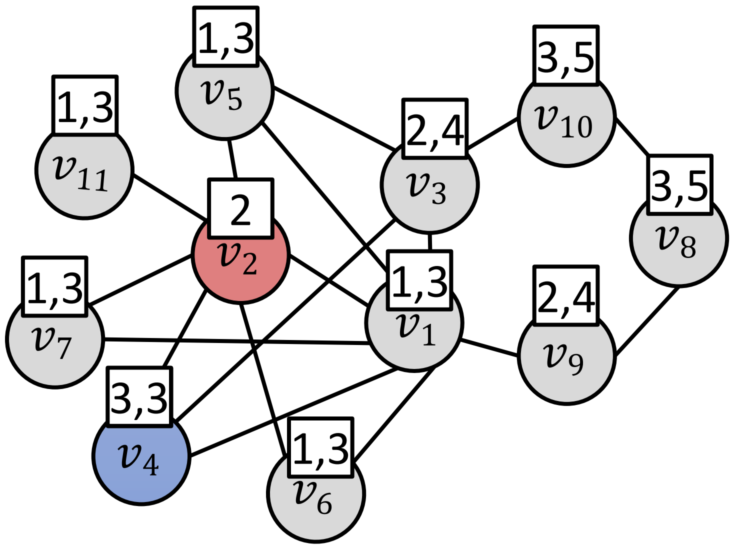
The distance of v_4 to the visited node v_{10}

Lower bound: max(**1**,**2**)=2

Eccentricity is the largest distance of a node

- **Example: Node v_4**
  - **Round 2**





The distance of v_4 to the visited node v_8

Lower bound: max(**2**,**3**)=3

The distance of visited nodes (in the reversing order of v_2) provides a good lower bound.

Another Example

- **Another Example: Node v_1**

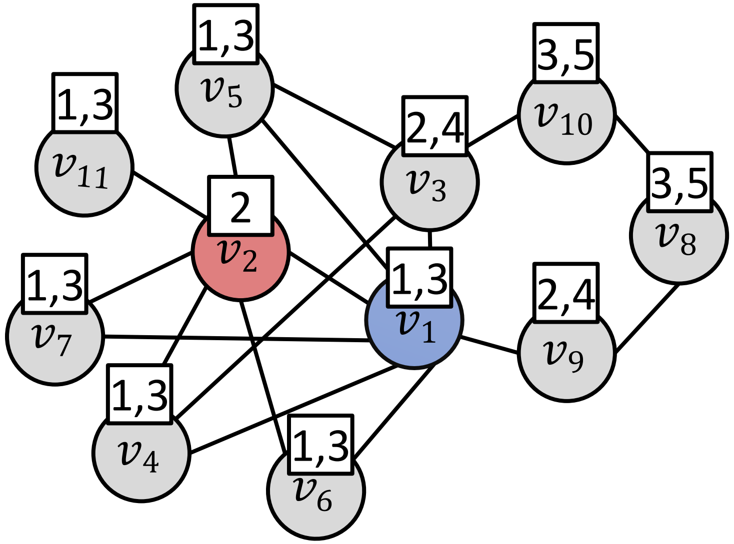
v_2

$d = 1$: $\{v_1, v_4, v_5, v_6, v_7, v_{11}\}$

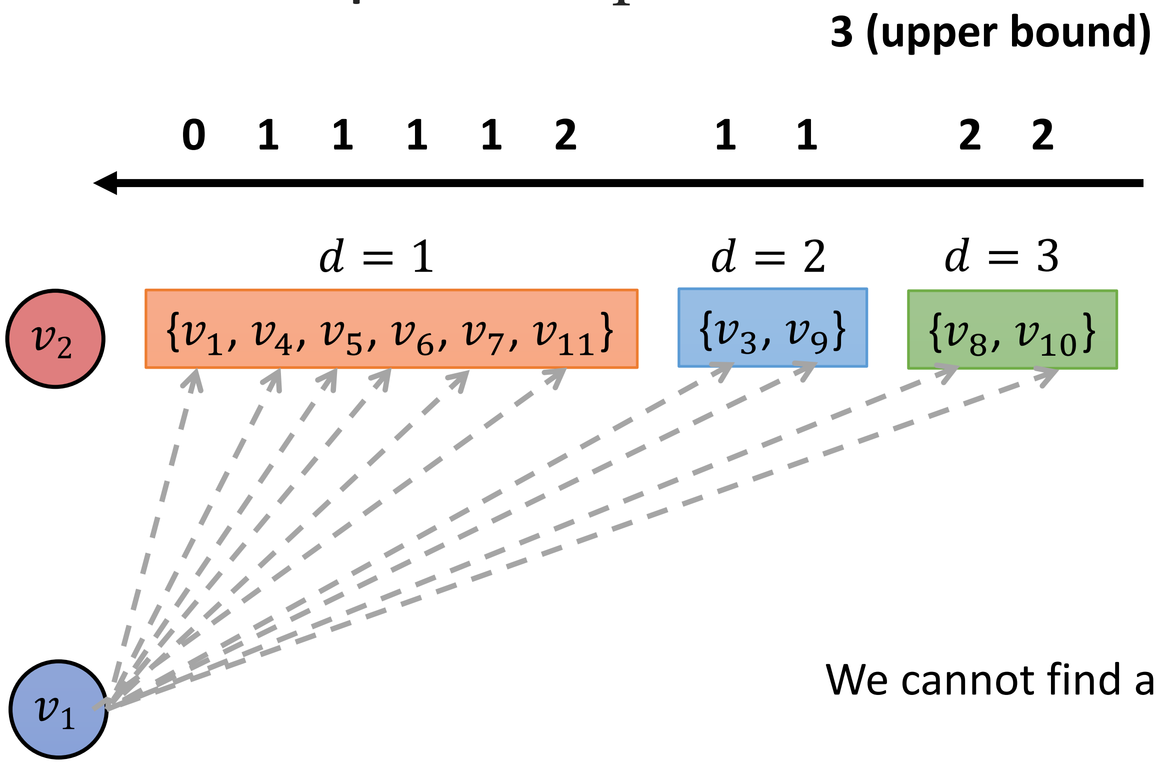
$d = 2$: $\{v_3, v_9\}$

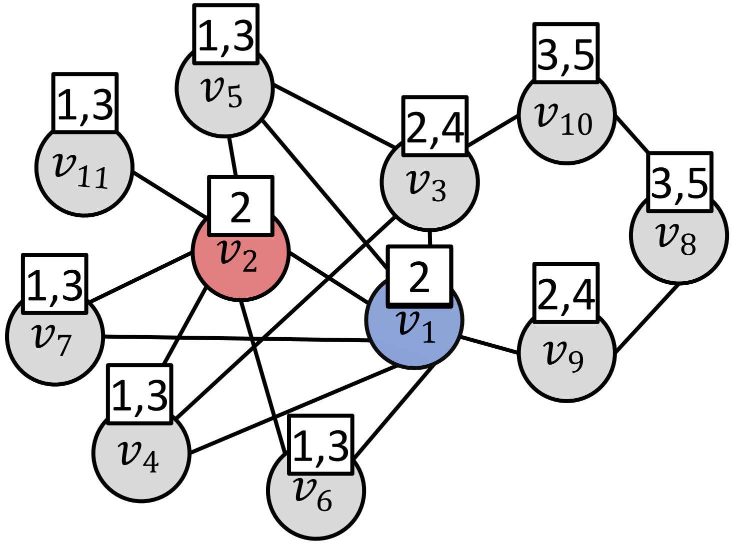
$d = 3$: $\{v_8, v_{10}\}$

v_1



- **Another Example: Node v_1**





We cannot find a good lower bound to meet the upper bound

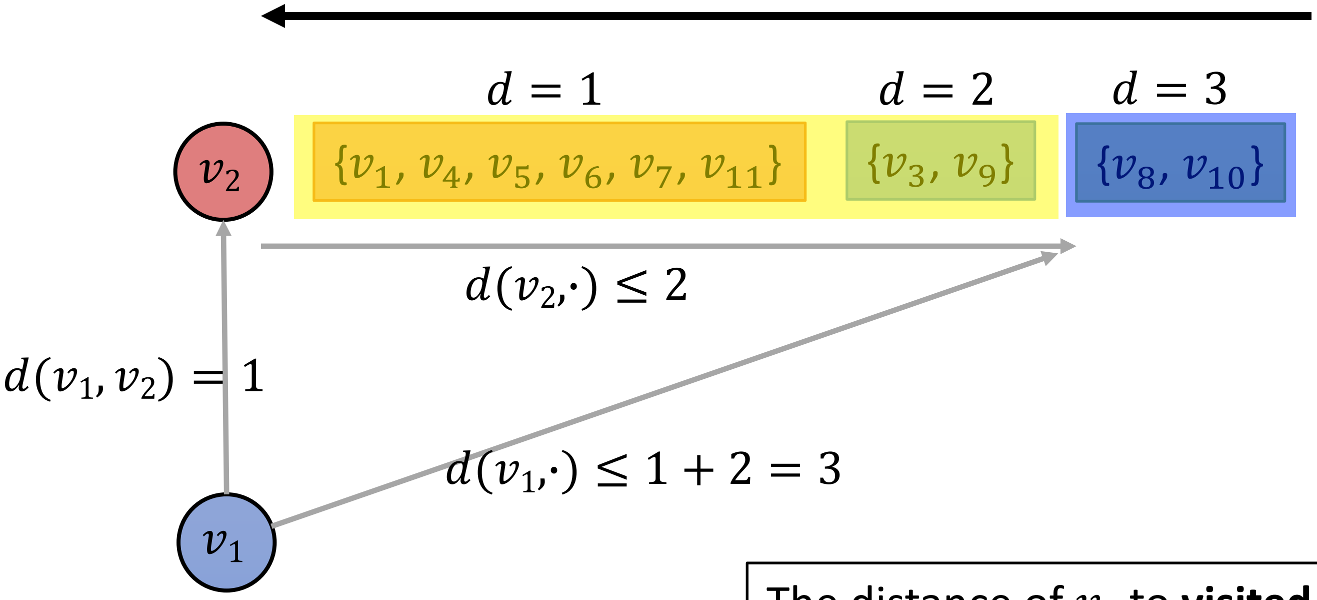
$ecc(v_1) = 2$ according to the definition

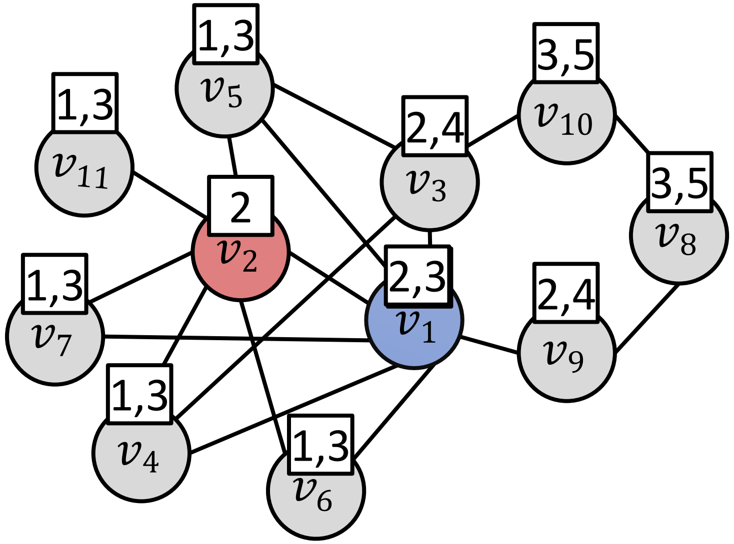
Lower bound is not enough.

A better upper bound is needed.

Design a Upper Bound

- **Node v_1**

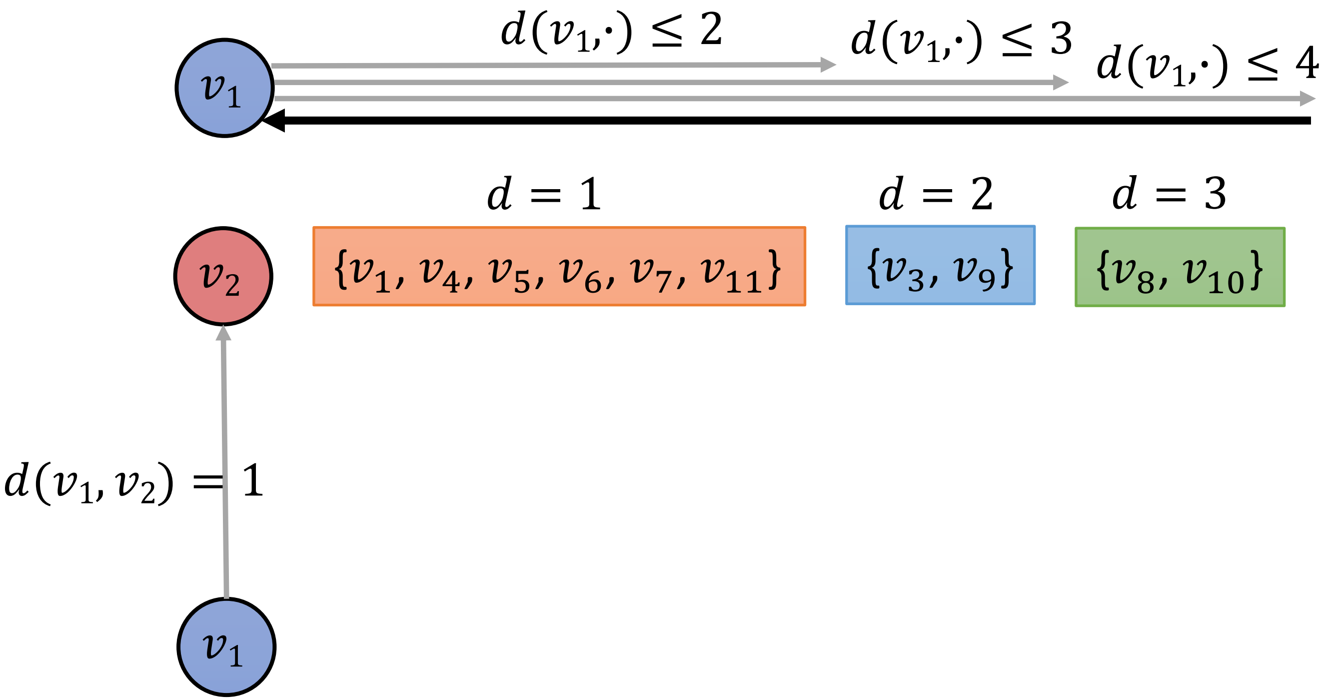




The distance of v_1 to **visited** nodes provides a **lower** bound of eccentricity.

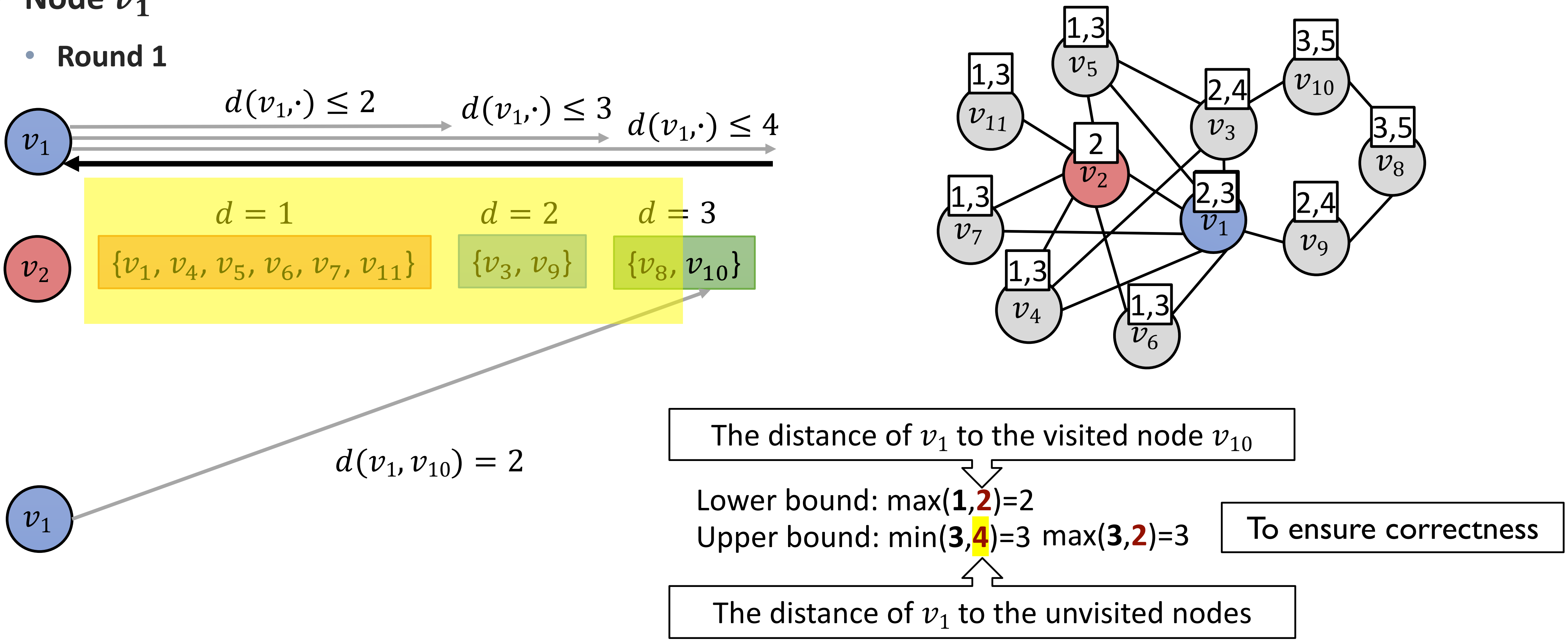
The distance of v_1 to **unvisited** nodes provides a **upper** bound of eccentricity on them.

- **Node v_1**

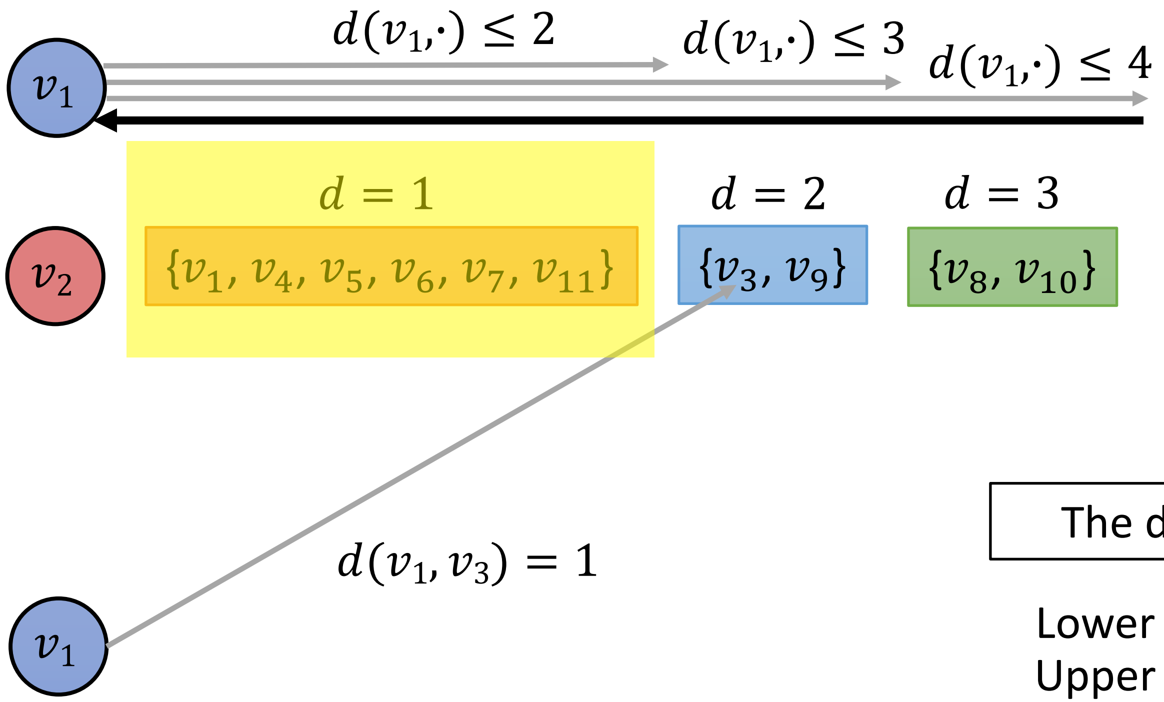

$d(v_1,\cdot) \leq 2$
$d(v_1,\cdot) \leq 3$
$d(v_1,\cdot) \leq 4$
v_1
$d = 1$
$d = 2$
$d = 3$
v_2
$\{v_1, v_4, v_5, v_6, v_7, v_{11}\}$
$\{v_3, v_9\}$
$\{v_8, v_{10}\}$
$d(v_1, v_2) = 1$
v_1

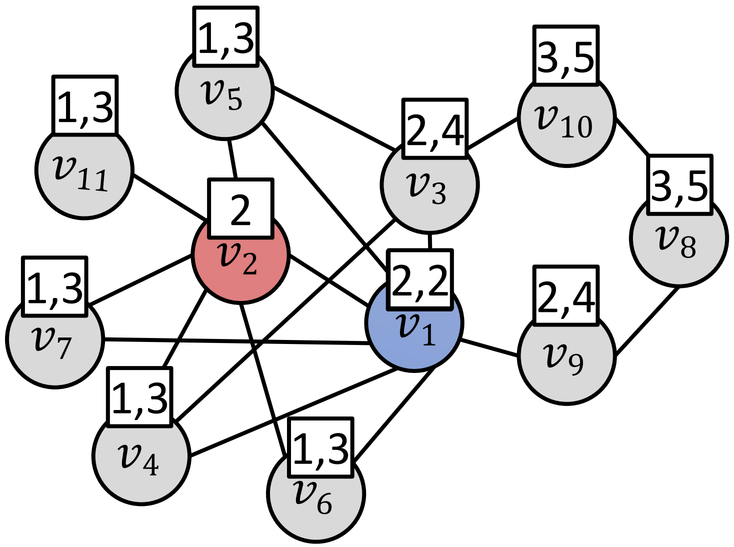
Try Again

- **Node v_1**
  - **Round 1**



- **Node v_1**
  - **Round 4**



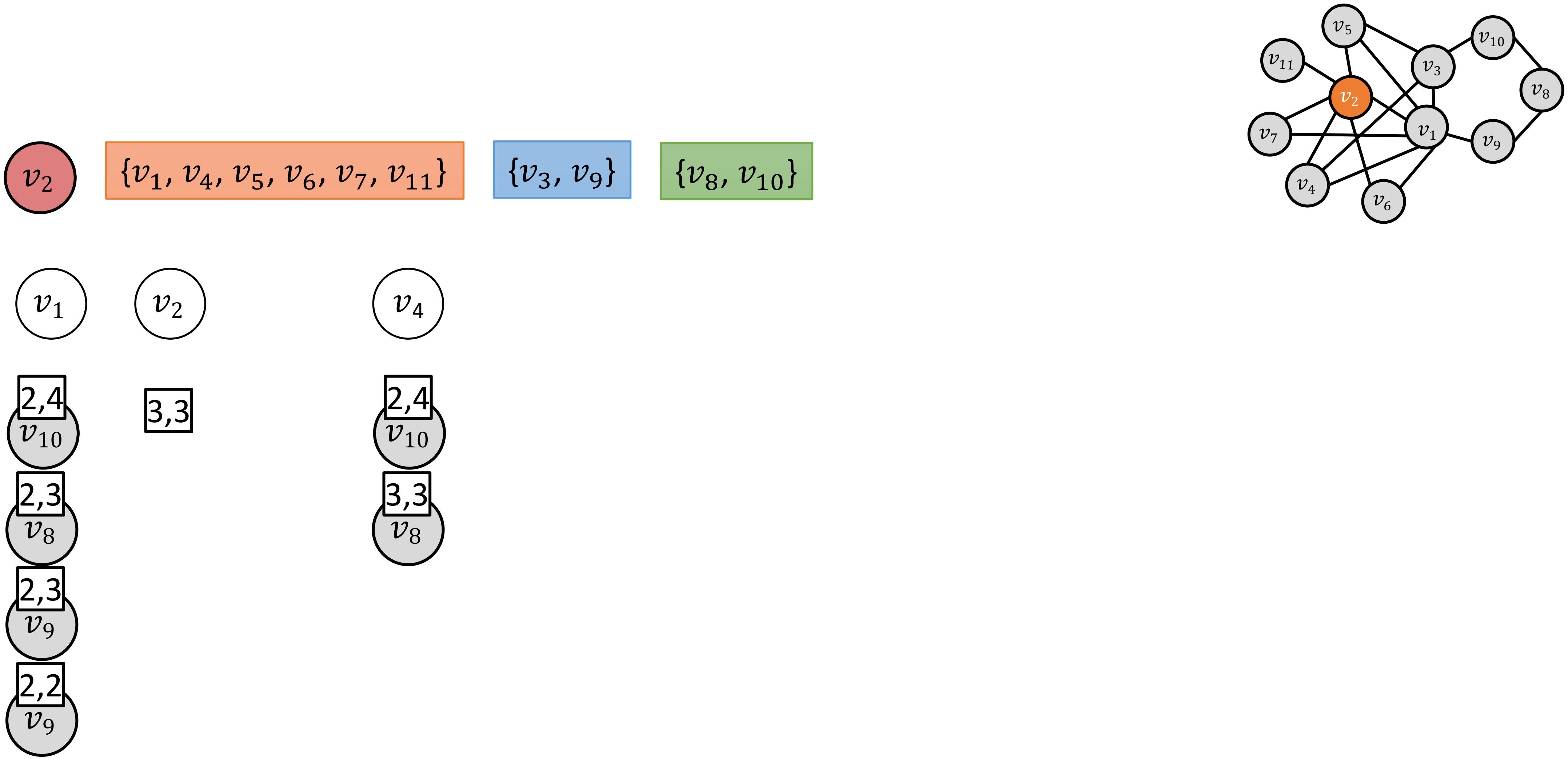


The distance of v_1 to the visited node v_3

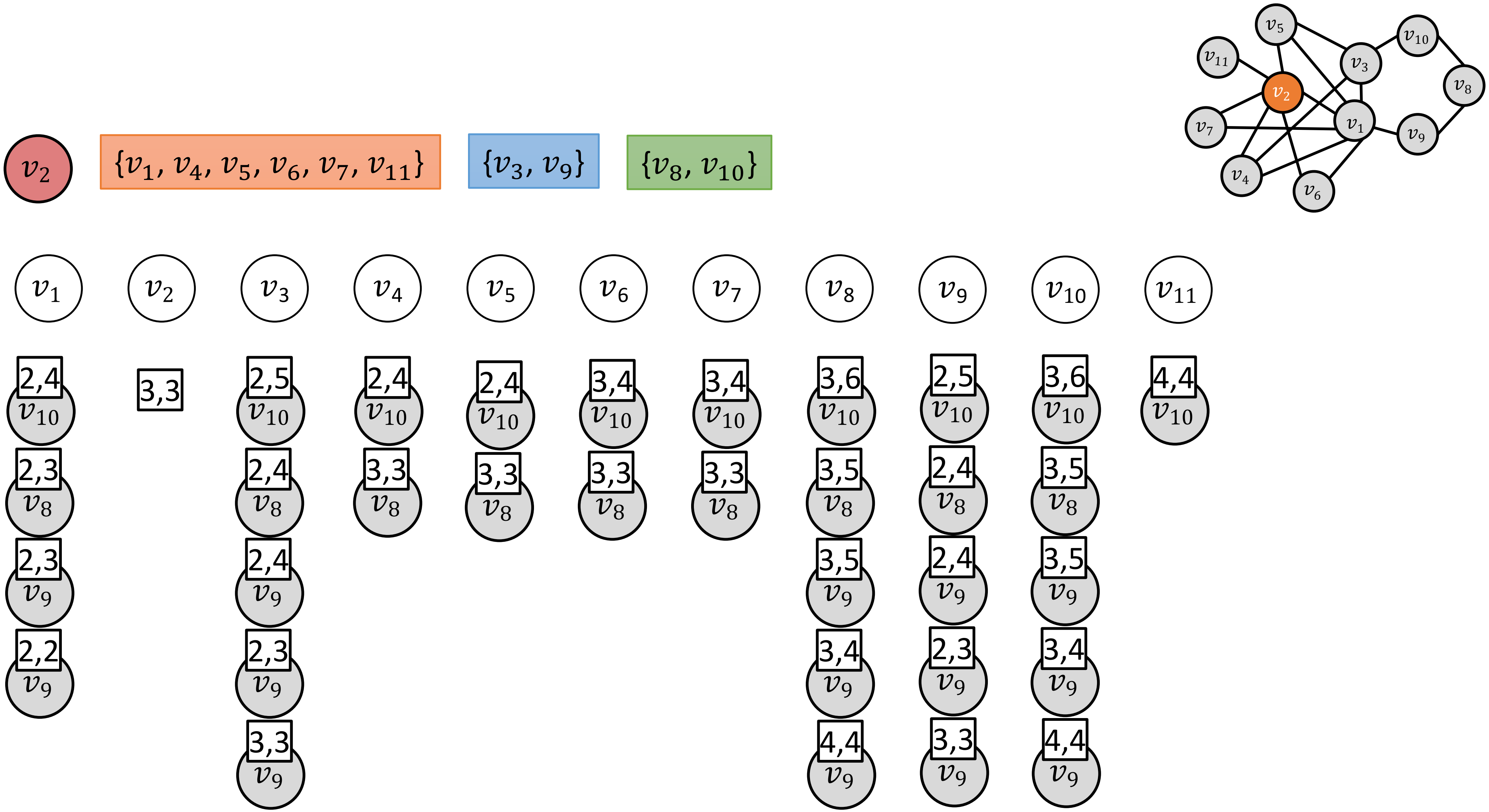
Lower bound: max(**2**,**1**)=2

Upper bound: min(**3**,**2**)=2 max(2,**1**)=2

The distance of v_1 to the unvisited nodes



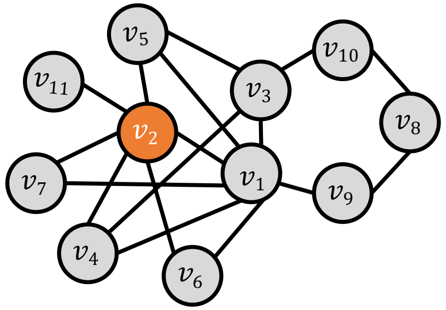
v2
{v1, v4, v5, v6, v7, v11}
{v3, v9}
{v8, v10}
v1
v2
v4
2,4
v10
2,3
v8
2,3
v9
2,2
v9
3,3
2,4
v10
3,3
v8
v5
v10
v11
v3
v2
v8
v7
v1
v9
v4
v6

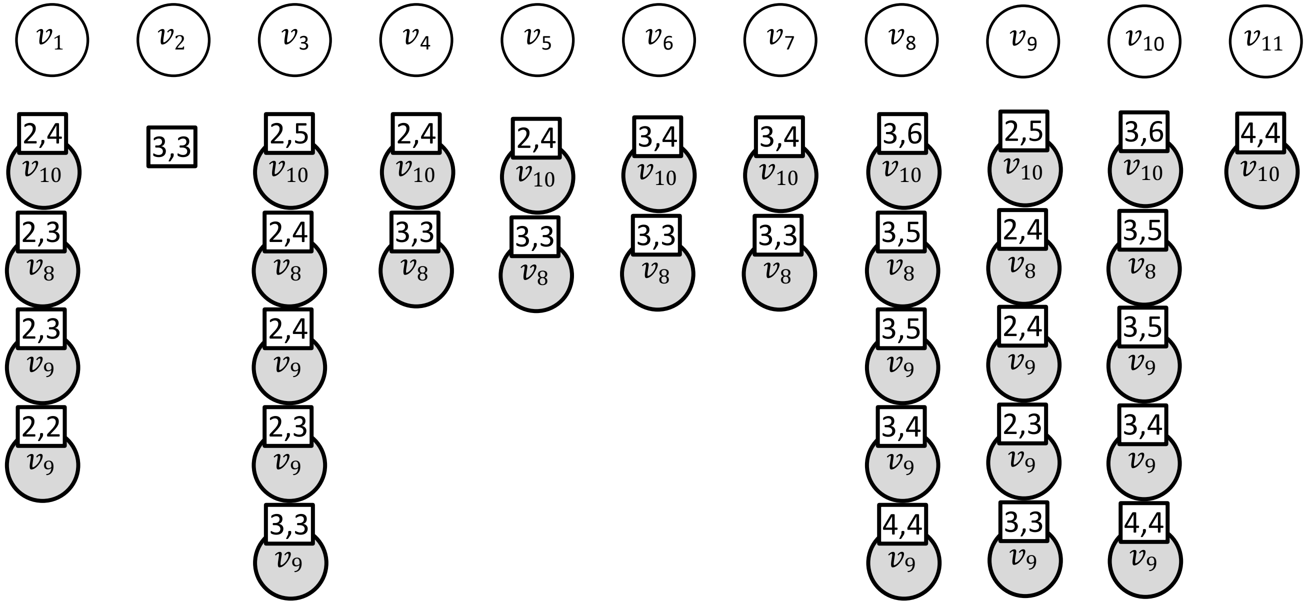


v_2
$\{v_1, v_4, v_5, v_6, v_7, v_{11}\}$
$\{v_3, v_9\}$
$\{v_8, v_{10}\}$
v_1 v_2 v_3 v_4 v_5 v_6 v_7 v_8 v_9 v_{10} v_{11}
v_1: 2,4 v_{10}; 2,3 v_8; 2,3 v_9; 2,2 v_9
v_2: 3,3
v_3: 2,5 v_{10}; 2,4 v_8; 2,4 v_9; 2,3 v_9; 3,3 v_9
v_4: 2,4 v_{10}; 3,3 v_8
v_5: 2,4 v_{10}; 3,3 v_8
v_6: 3,4 v_{10}; 3,3 v_8
v_7: 3,4 v_{10}; 3,3 v_8
v_8: 3,6 v_{10}; 3,5 v_8; 3,5 v_9; 3,4 v_9; 4,4 v_9
v_9: 2,5 v_{10}; 2,4 v_8; 2,4 v_9; 2,3 v_9; 3,3 v_9
v_{10}: 3,6 v_{10}; 3,5 v_8; 3,5 v_9; 3,4 v_9; 4,4 v_9
v_{11}: 4,4 v_{10}

Multiple Reference Nodes

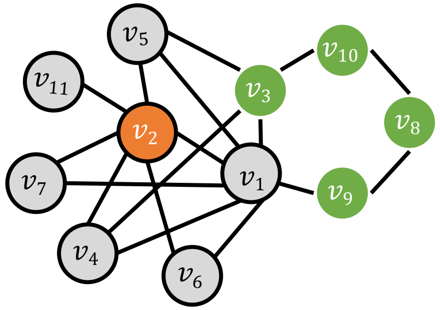
- One reference node

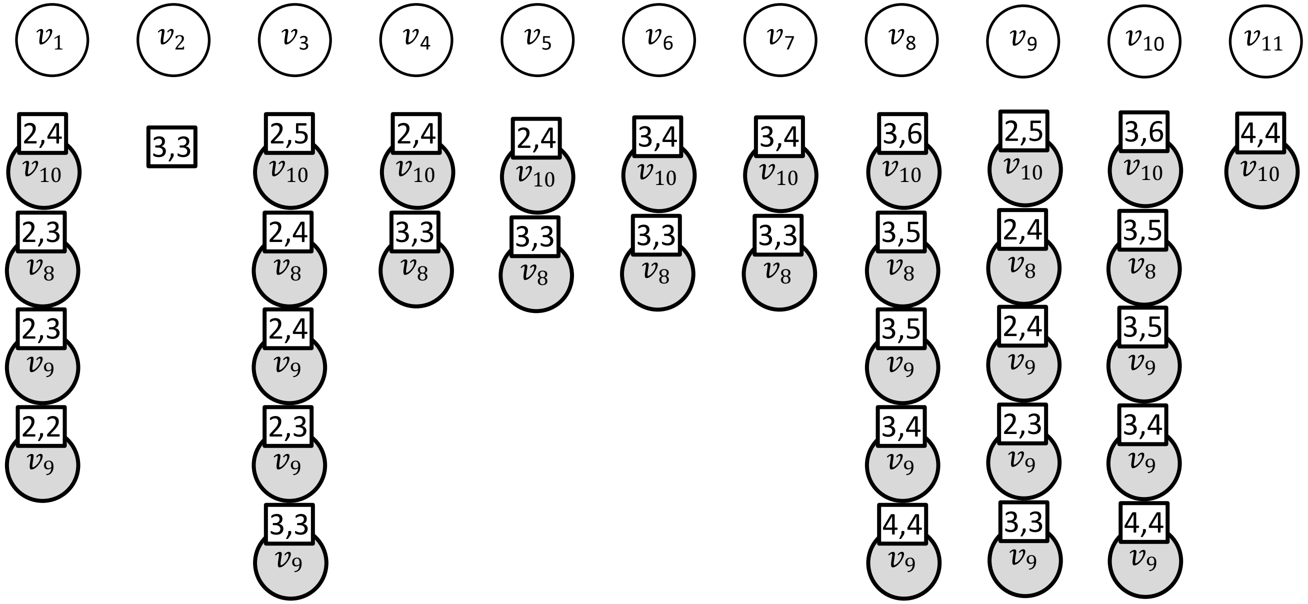




Multiple Reference Nodes

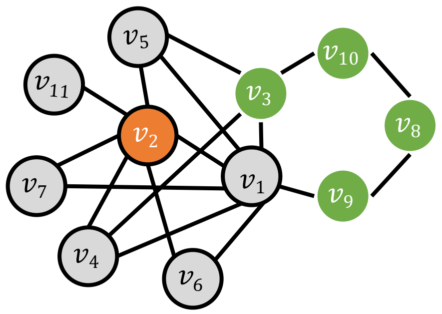
- One reference node

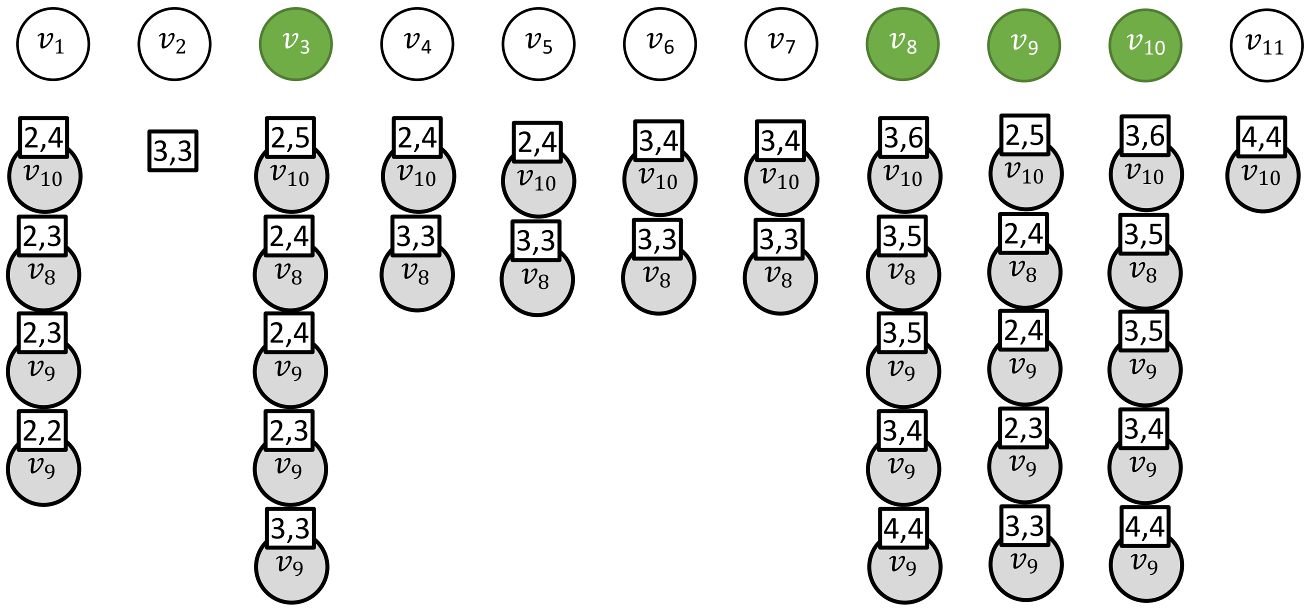




Multiple Reference Nodes

- One reference node

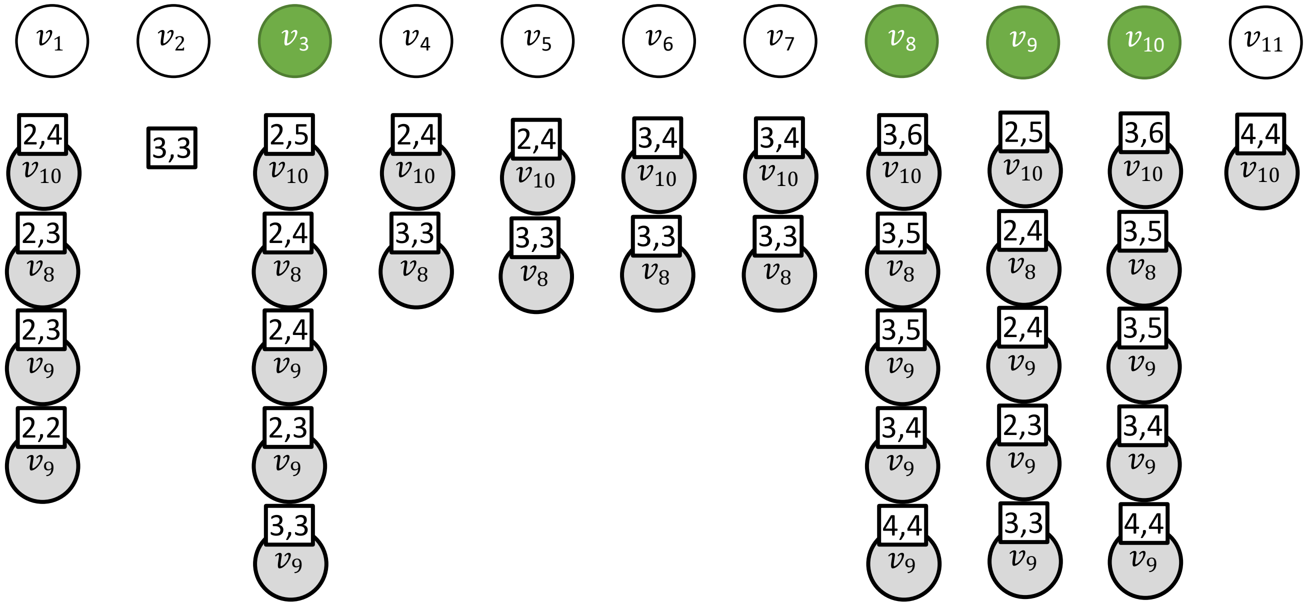




Multiple Reference Nodes

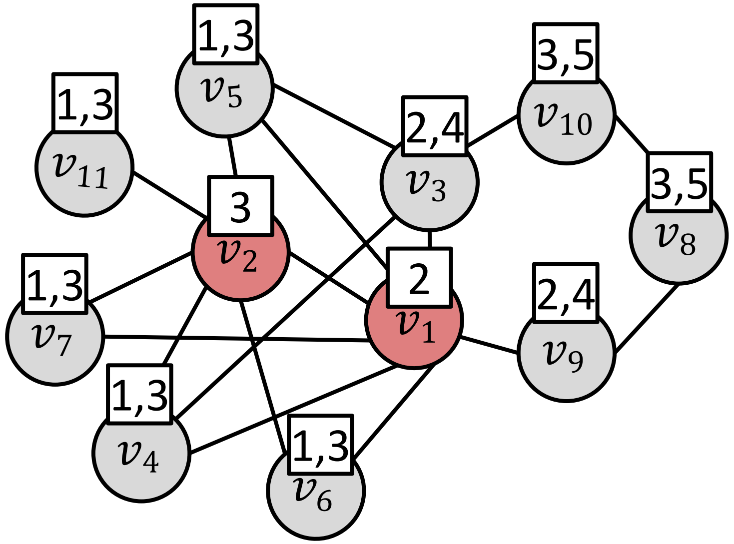
Each node are more likely to find the nearby reference node to use

- Multiple reference nodes
  - How to select the reference nodes?
  - Which reference node to refer for a certain node?



How to select reference nodes?

- High degree nodes
  - They are the hub nodes with more connections than other nodes in the small world graphs.
  - Easy to obtain the high degree nodes.

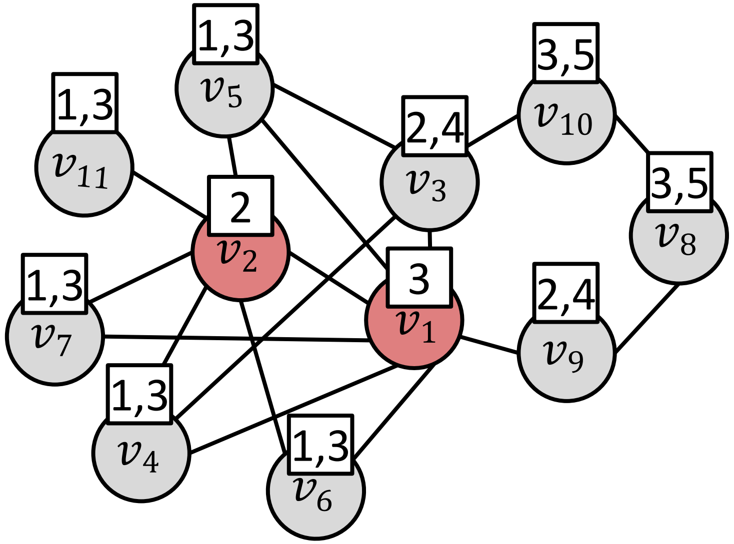


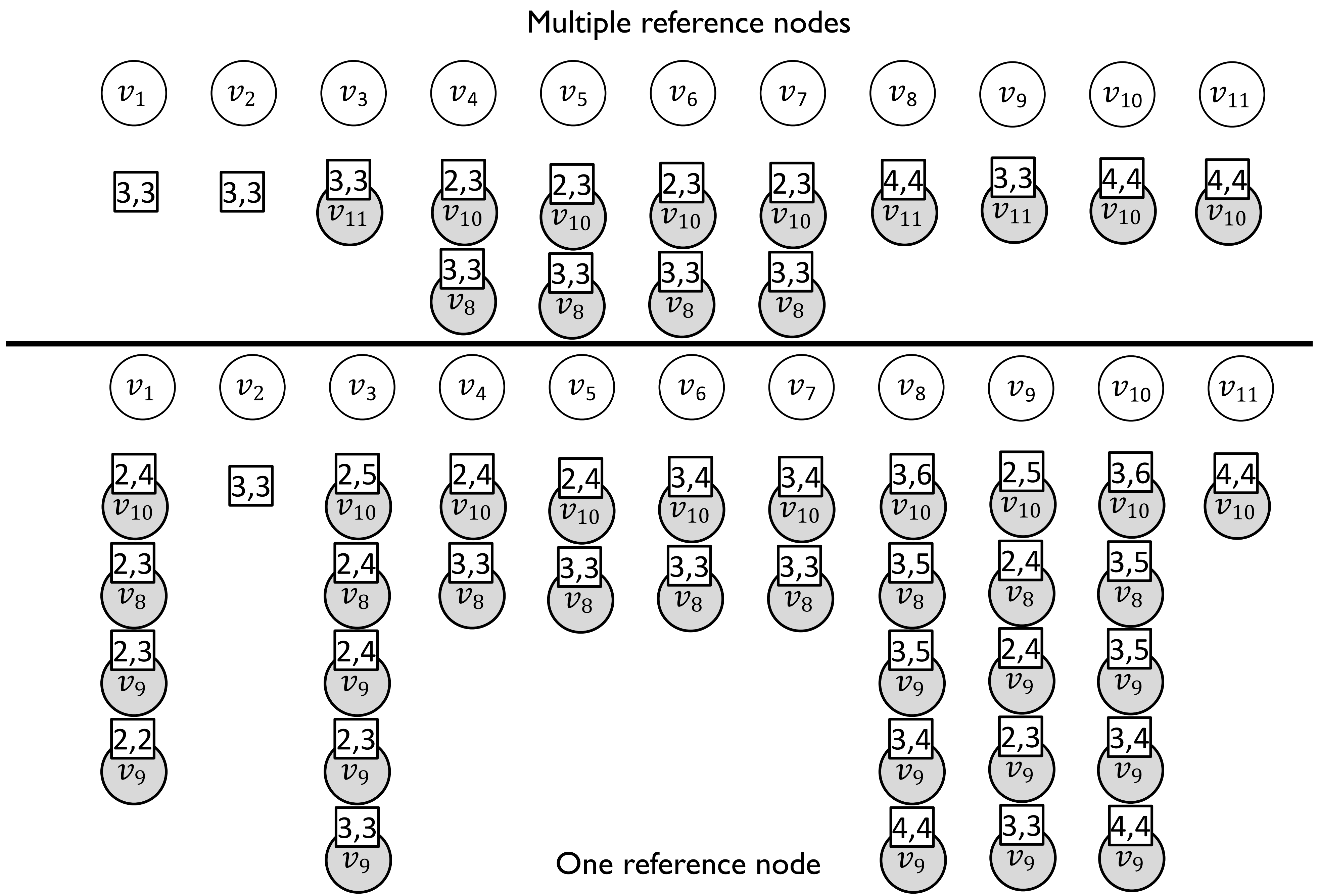
v_2: $d = 1$: $\{v_1, v_4, v_5, v_6, v_7, v_{11}\}$; $d = 2$: $\{v_3, v_9\}$; $d = 3$: $\{v_8, v_{10}\}$

v_1: $d = 1$: $\{v_2, v_3, v_4, v_5, v_6, v_7, v_9\}$; $d = 2$: $\{v_8, v_{10}, v_{11}\}$

Which reference node to refer?

- The nearest reference node to use.
  - v_{11} chooses v_2 (v_1: 2 - v_2: 1)
  - v_{10} chooses v_1 (v_1: 2 - v_2: 3)
  - If equal
    - we can choose the larger degree one.
    - v_4 chooses v_1 (v_1: 1 - v_2: 1)



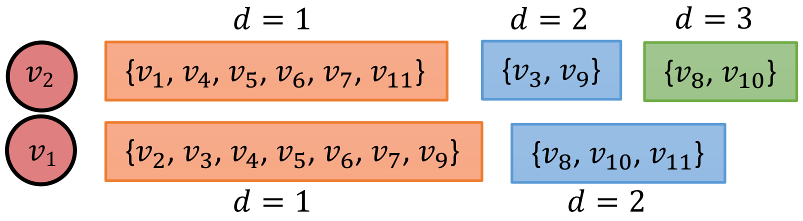

Multiple reference nodes
v_1 v_2 v_3 v_4 v_5 v_6 v_7 v_8 v_9 v_{10} v_{11}
3,3
3,3
3,3 v_{11}
2,3 v_{10} 3,3 v_8
2,3 v_{10} 3,3 v_8
2,3 v_{10} 3,3 v_8
2,3 v_{10} 3,3 v_8
4,4 v_{11}
3,3 v_{11}
4,4 v_{10}
4,4 v_{10}
v_1 v_2 v_3 v_4 v_5 v_6 v_7 v_8 v_9 v_{10} v_{11}
2,4 v_{10} 2,3 v_8 2,3 v_9 2,2 v_9
3,3
2,5 v_{10} 2,4 v_8 2,4 v_9 2,3 v_9 3,3 v_9
2,4 v_{10} 3,3 v_8
2,4 v_{10} 3,3 v_8
3,4 v_{10} 3,3 v_8
3,4 v_{10} 3,3 v_8
3,6 v_{10} 3,5 v_8 3,5 v_9 3,4 v_9 4,4 v_9
2,5 v_{10} 2,4 v_8 2,4 v_9 2,3 v_9 3,3 v_9
3,6 v_{10} 3,5 v_8 3,5 v_9 3,4 v_9 4,4 v_9
4,4 v_{10}
One reference node

Summary

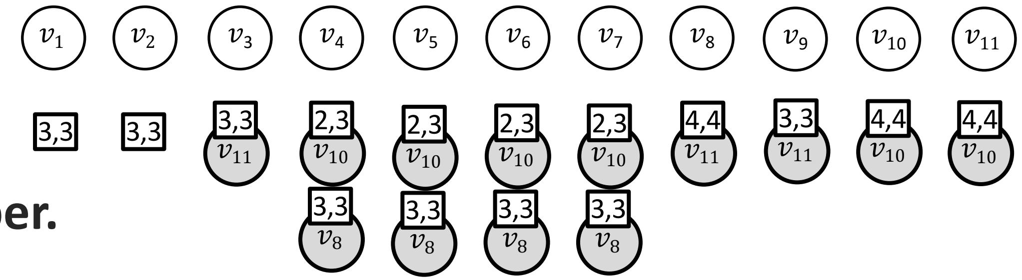
- Step 1: Labeling
  - to construct the distance labels (PLL).
- Step 2: Reference
  - to select reference nodes and obtain the distance information.
- Step 3: Eccentricity
  - to gain the eccentricity of each node by reversing visiting.

- **More optimization techniques are introduced in the paper.**

Step 1: Labeling



Step 2: Reference



Step 3: Eccentricity

Performance Studies

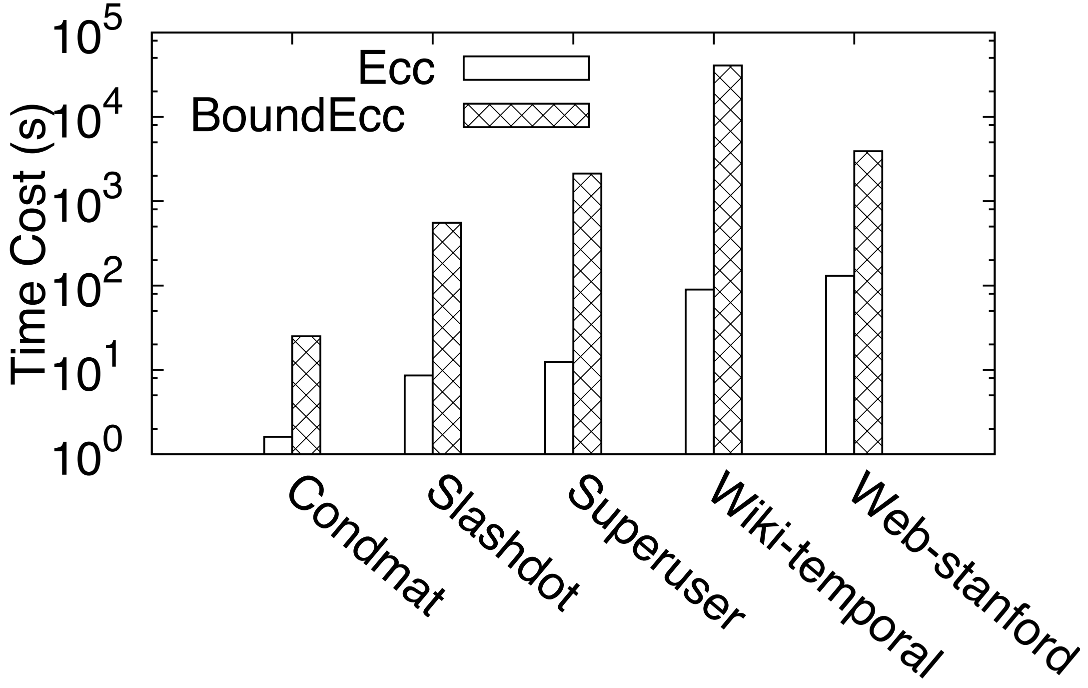
- Intel Xeon 3.1GHz CPU and 128 GB main memory running Linux (Red Hat Linux 4.4.7, 64bit).

Algorithms

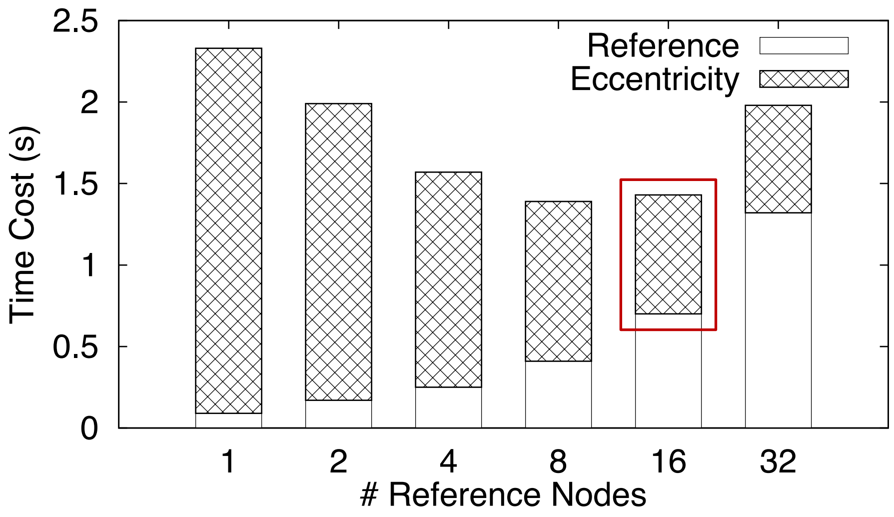
- BoundEcc
  - The node with the maximum (and minimum, resp.) eccentricity upper bound (and lower bound, resp.) are visited alternatively
- Ecc

Comparison with BoundEcc

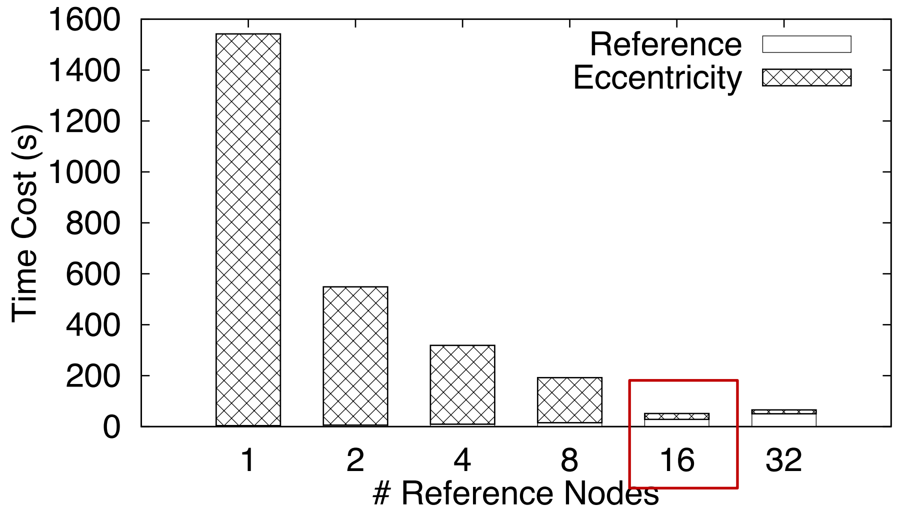
- Speedup
  - Condmat (Collaboration): 15.5
  - Slashdot (Social): 64.8
  - Superuser (Interaction): 171.4
  - Wiki-temporal (Communication): 452.6
  - Web-standford (Web): 29.9



Reference Nodes



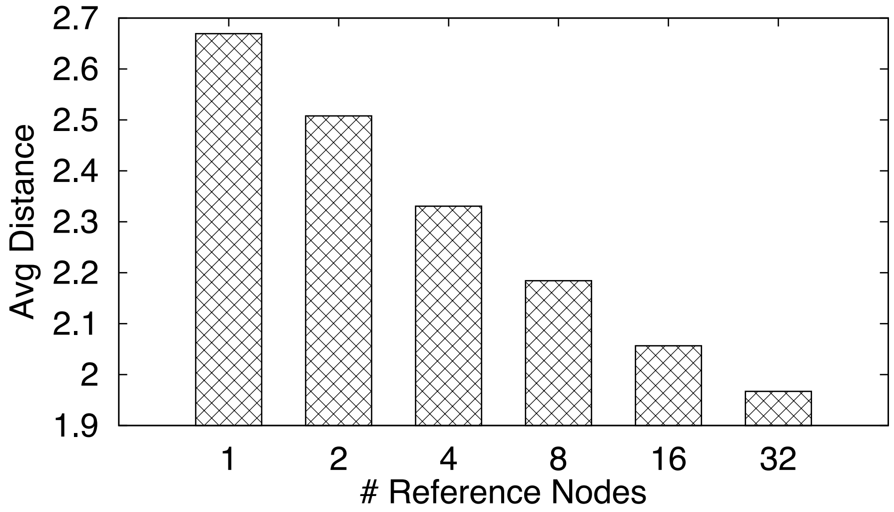
Slashdot



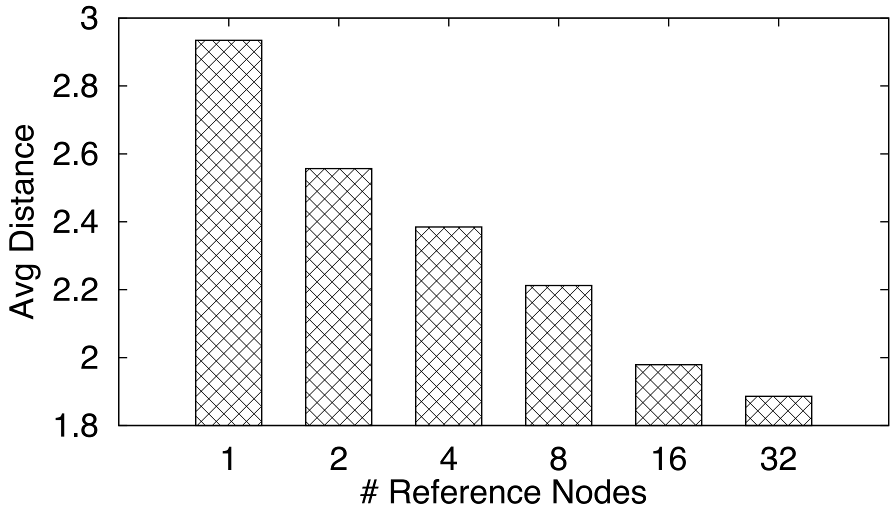
Wiki-talk

Distance To Reference Nodes

- Observation 1: There exists some ***reference nodes*** that are near to all the nodes in a small-world network.

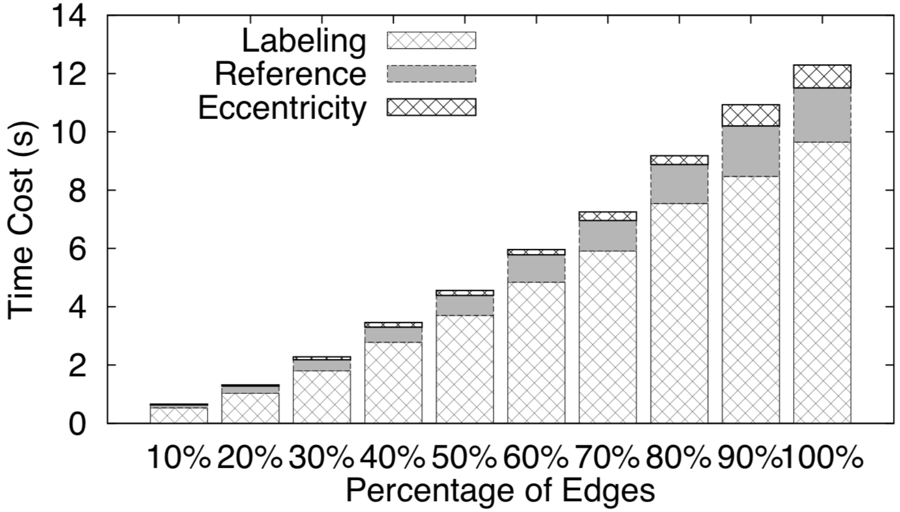


Slashdot

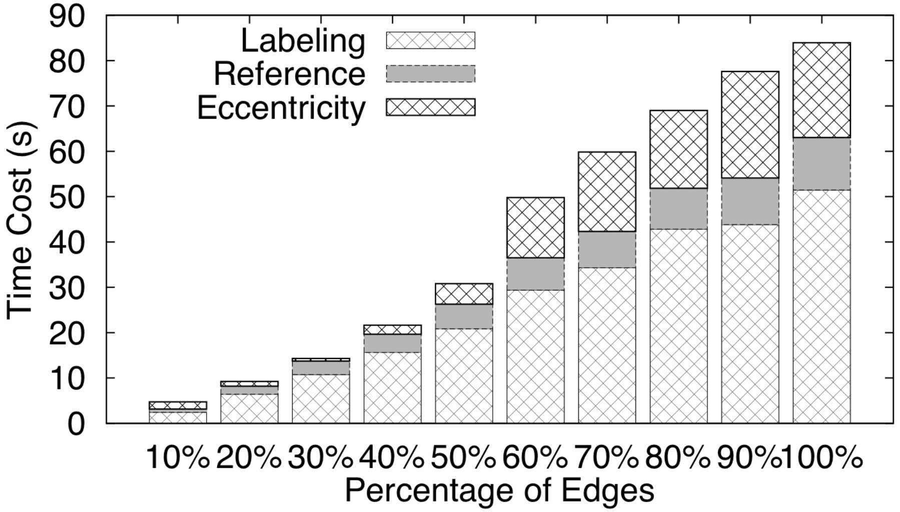


Wiki-talk

Scalability



Superuser



Wiki-temporal

Conclusion

Conclusion

- Exacting eccentricity for small-world networks.
- Determine the eccentricity of a node in an early stop way rather than travel the entire graph.
- Extensive experimental studies on real world networks with millions of nodes and edges.

Thank you

- Q&A