DISCRETE MATHEMATICS

IN COMPUTER SCIENCE

Hsien-Chih Chang

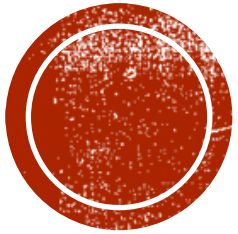
January 28, 2022

Administrivia

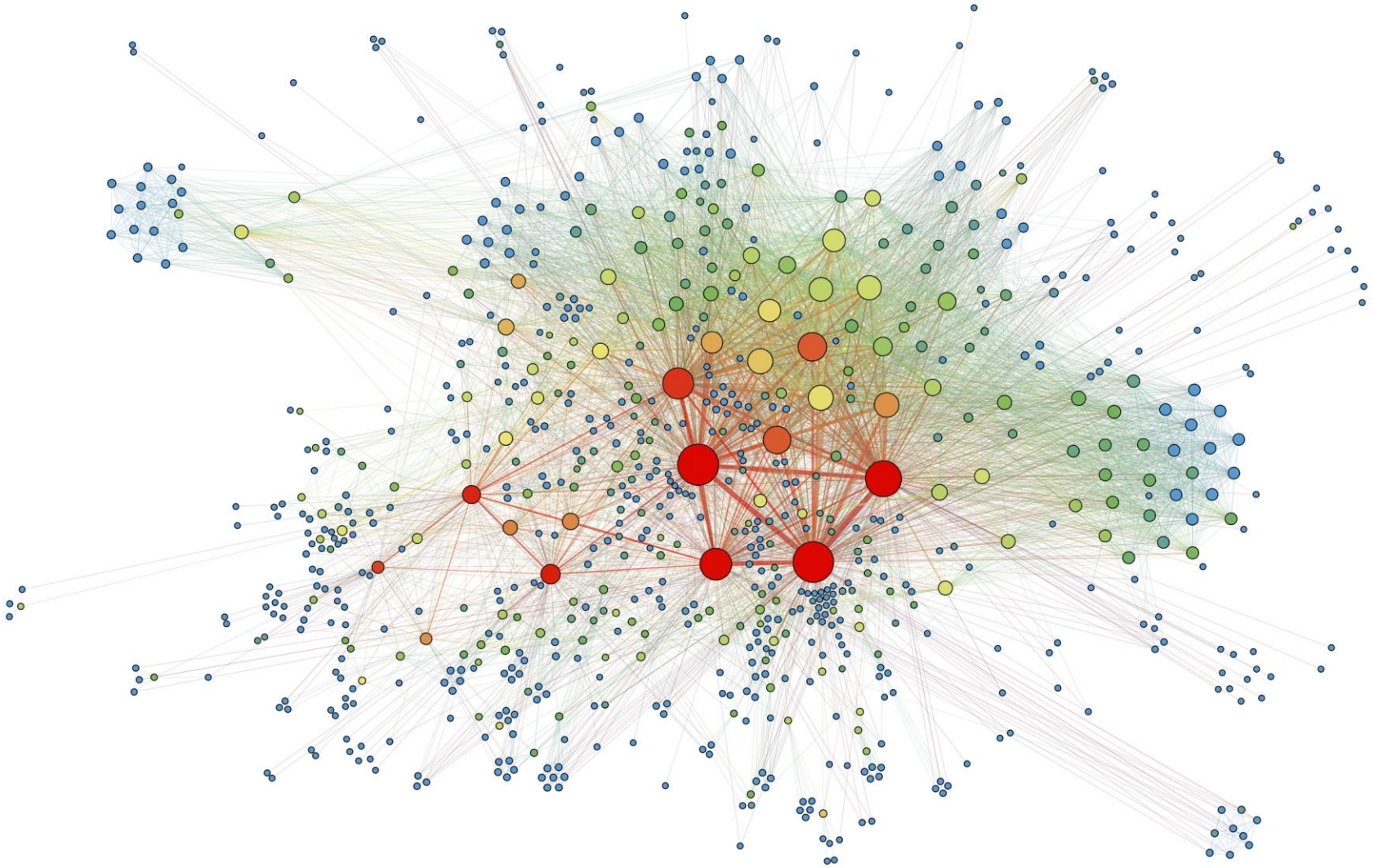
- Midterm 1
  - Jan 31 (Mon) 6–9PM
  - Moore Hall B13 Filene Auditorium
- Conflict Midterm 1
  - Feb 1 (Tue) 5–8PM
  - Moore Hall B13 Filene Auditorium
- SAS/Conflict Conflict/COVID
  - Come talk to me
- Closed-book written exam
- Scope: Module P on proofs
- One-page two-sided cheatsheet
  - Must be hand-written

Administrivia

- Homework 3 due today
- Homework 4 out today



Graph Traversal

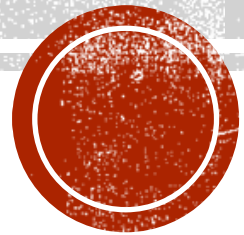


Jargon

walk
path
closed walk
cycle
length
distance

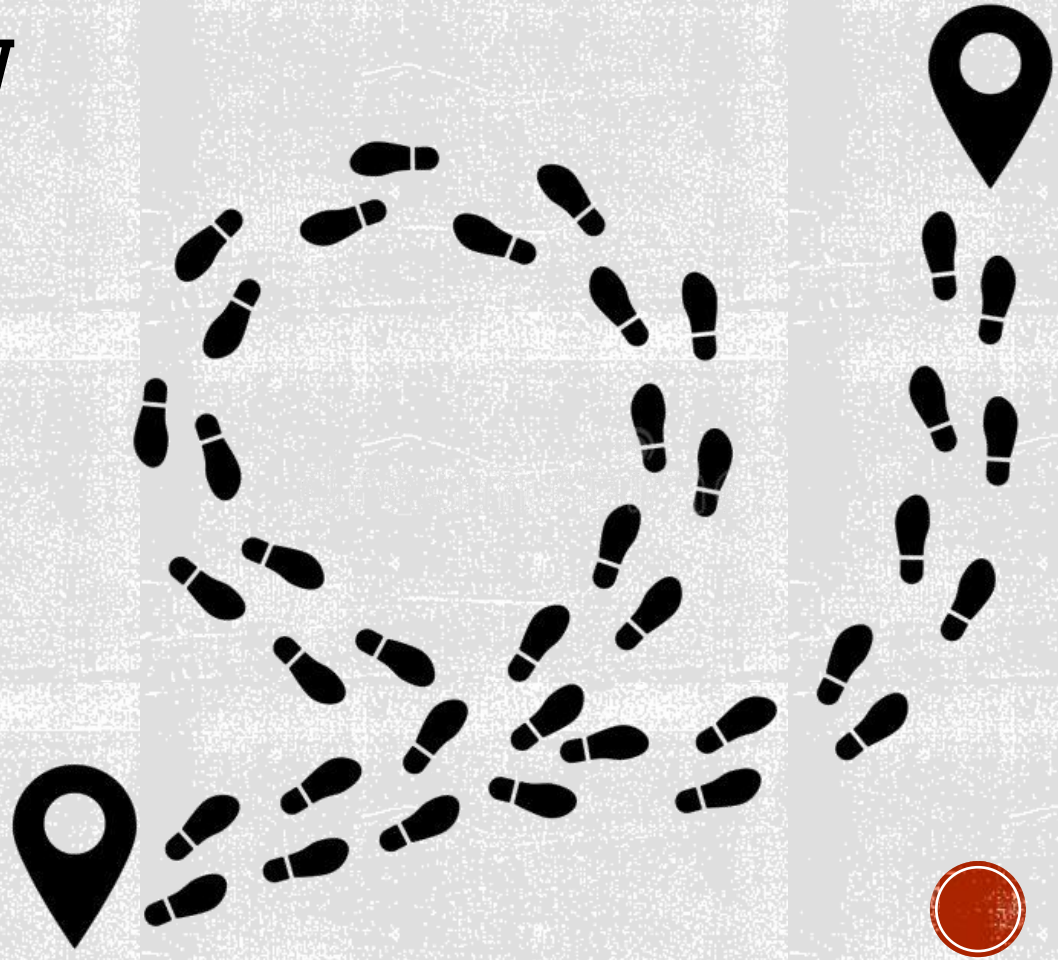
reachable
connected
components

Is there a path from u to v?



Wandering Lemma

- For any graph G and two vertices u and v
  - if there is a walk from u to v,
    then there is a path from u to v.



Corollary

Any shortest walk between two vertices must be a path.

Any shortest path in an n-vertex graph must have length at most n-1.

```
WhateverFirstSearch(s, G):
    put s in an empty bag
    while the bag is not empty:
        take v from the bag
        if v is unmarked:
            mark v
            for each edge v→w in G:
                put w in the bag
```

Theorem. WhateverFirstSearch(s, G) marks all vertices reachable from s in G.

Whatever-First Search

Implement the bag

- stack: DFS
- queue: BFS
- priority queue:
  - MST (weight)
  - Dijkstra (distance)

Whatever-First Search

```
WhateverFirstSearch(s, G):
    put ε→s in an empty bag
    while the bag is not empty:
        take p→v from the bag
        if v is unmarked:
            mark v
            parent(v) ← p
            for each edge v→w in G:
                put v→w in the bag
```

Theorem. WhateverFirstSearch(s, G) marks exactly the vertices reachable from s in G.

Breadth-First Search

```
BreadthFirstSearch(s, G):
    put ε→s in an empty queue
    while the queue is not empty:
        take p→v from the queue
        if v is unmarked:
            mark v
            parent(v) ← p
            for each edge v→w in G:
                put v→w in the queue
```

Theorem. BreadthFirstSearch(s, G) computes the shortest path from s to any vertex v in G.

Can the set of parent edges form a cycle?

Next time.
Forests and Trees