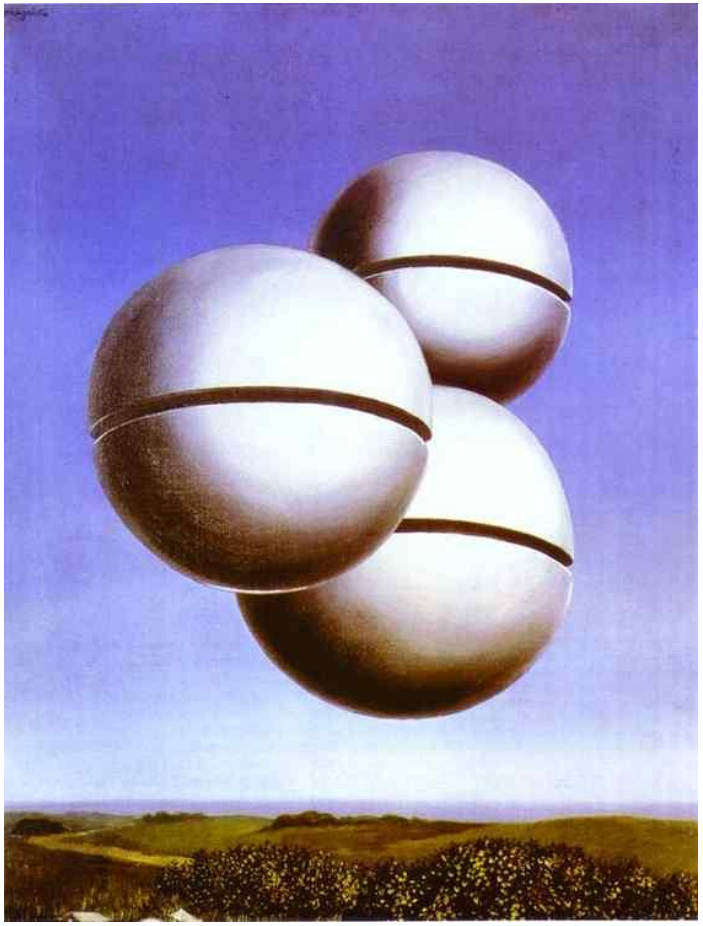
Intro to Discrete Structures



'The Voice of the Winds' - René Magritte

Discrete Structures (CS 173)

Derek Hoiem, University of Illinois

Today’s class

- A little about me
- Introduction to discrete structures
- Course logistics

About me

Raised in “upstate” NY

About me

1998-2002
Undergrad at SUNY Buffalo
B.S., EE and CSE

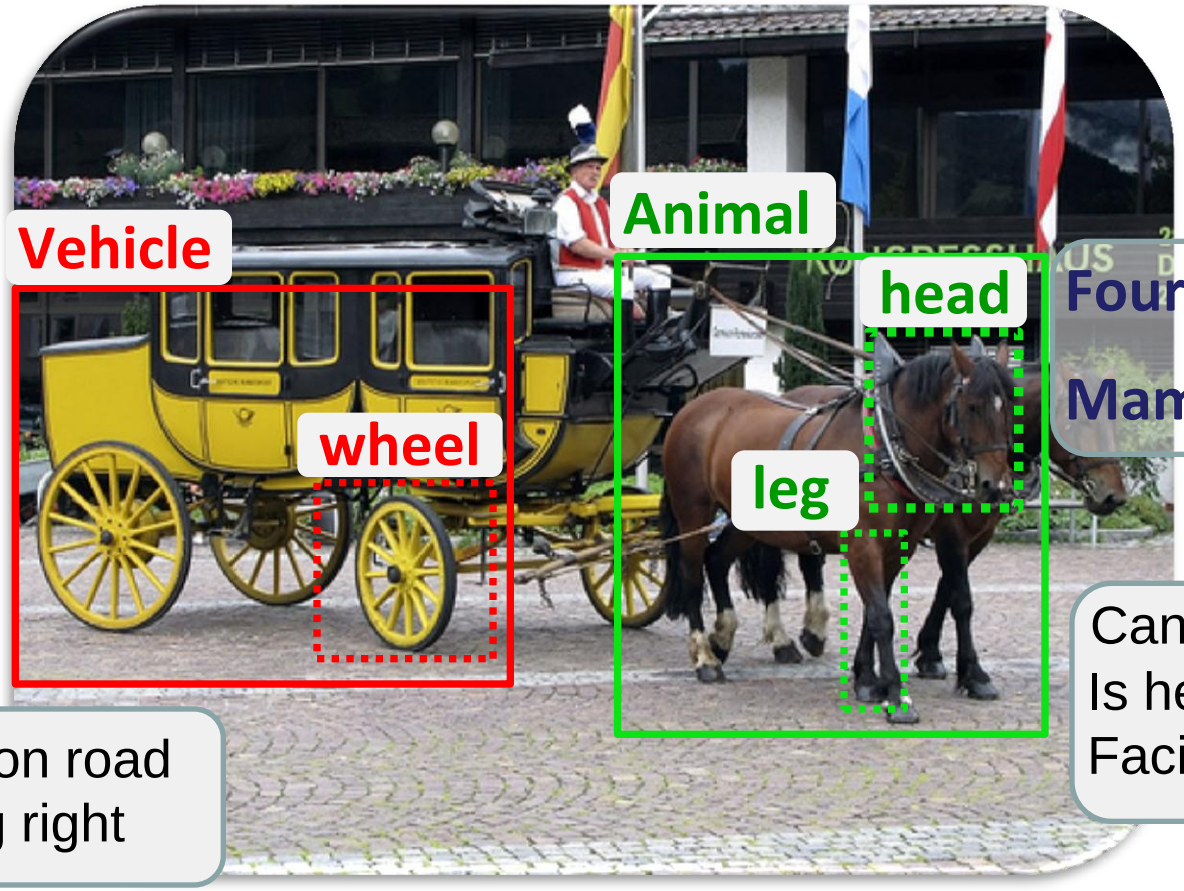
2002-2007
Grad at Carnegie Mellon
Ph.D. in Robotics

2007-2008
Postdoc at Beckman Institute

2009-
Assistant Prof in CS at UIUC

My research

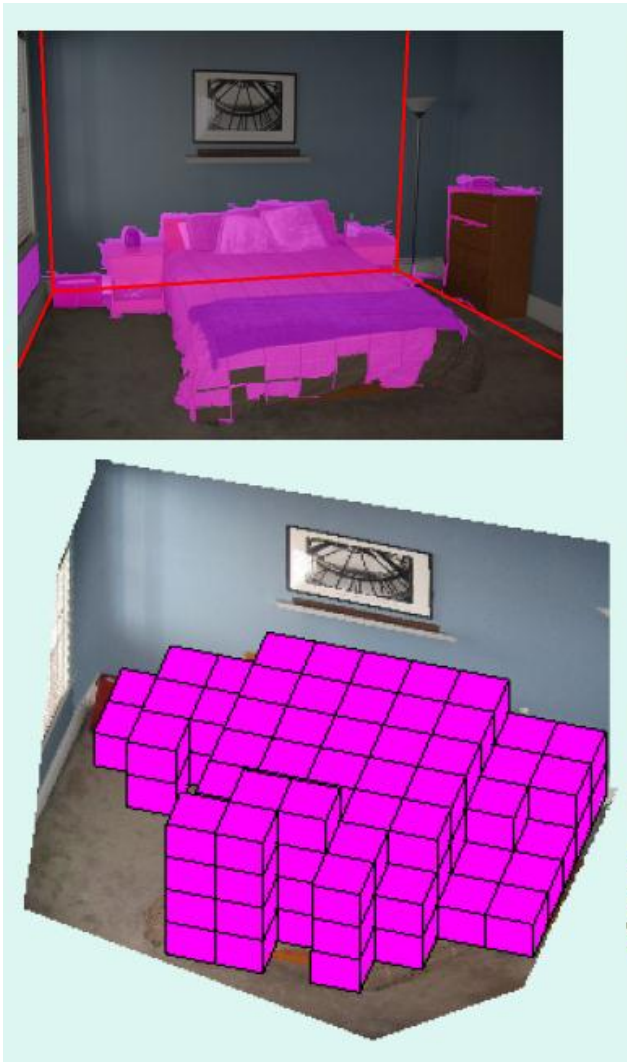
My Research

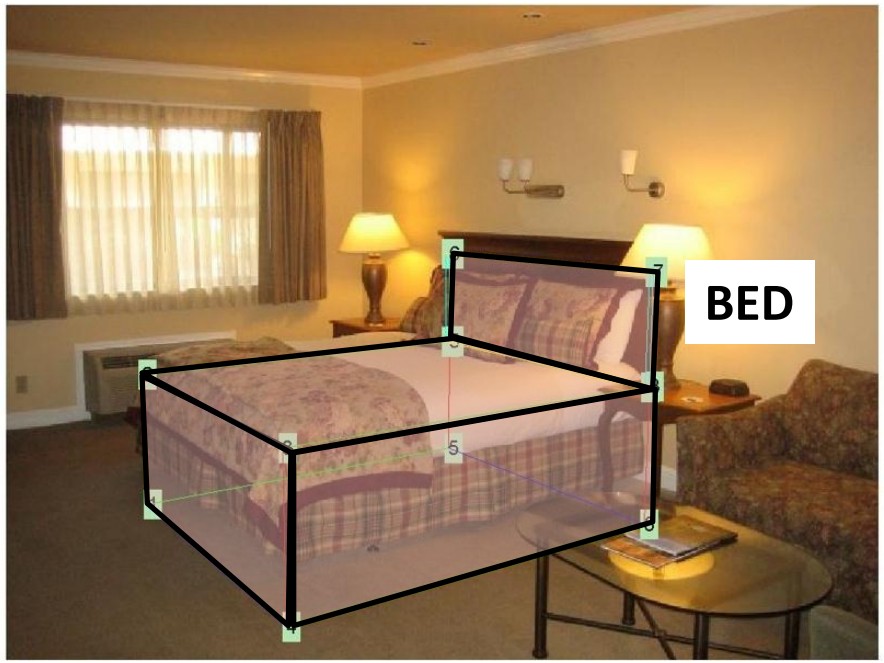


Farhadi et al. 2010

My Research

Recovering 3D layout and context

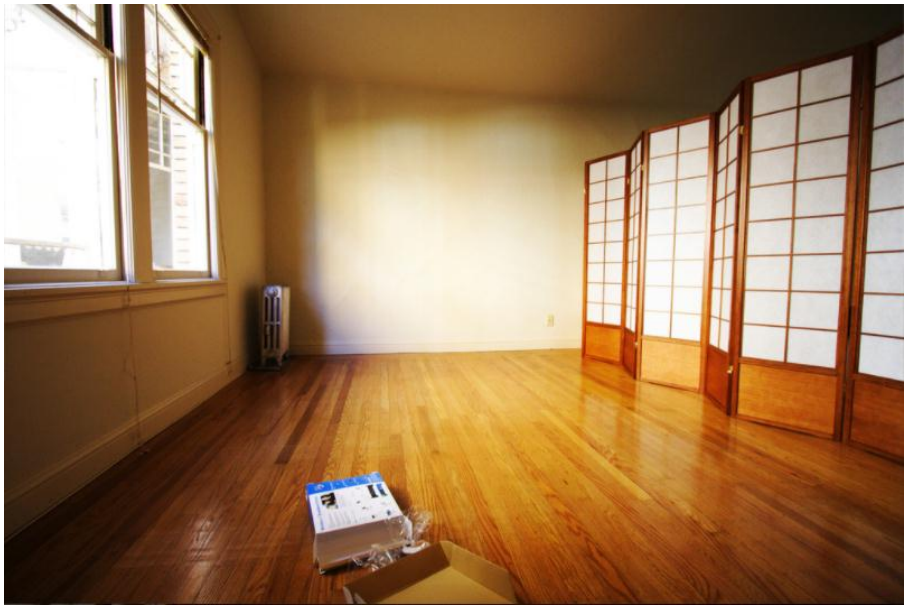


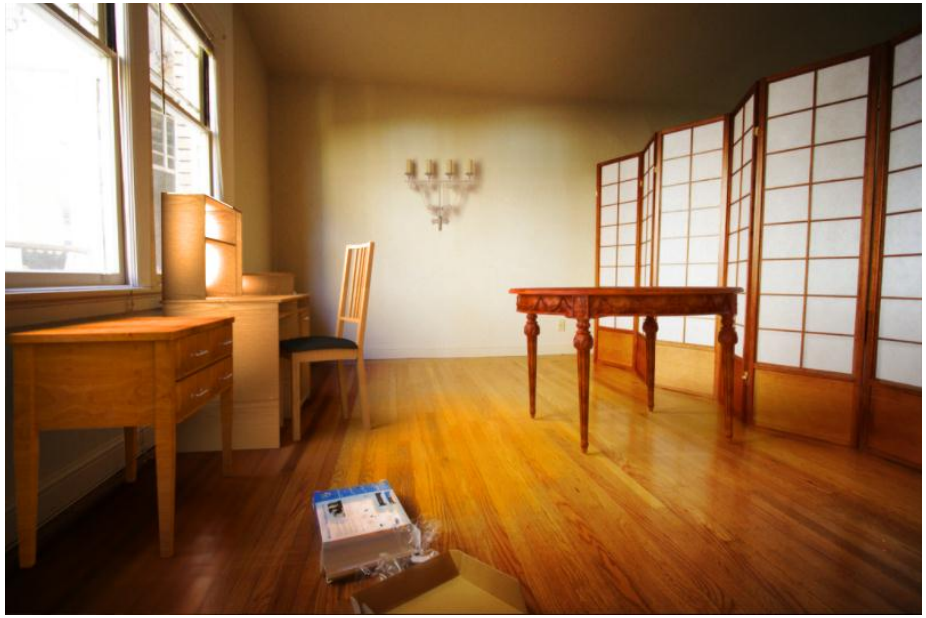


Hedau et al. 2009, 2010

My Research

Editing images as if they were 3D scenes





Karsch et al. 2011


GOLFER


GOLFER

Intro to Discrete Structures

Discrete Structures ≠ Discreet Structures

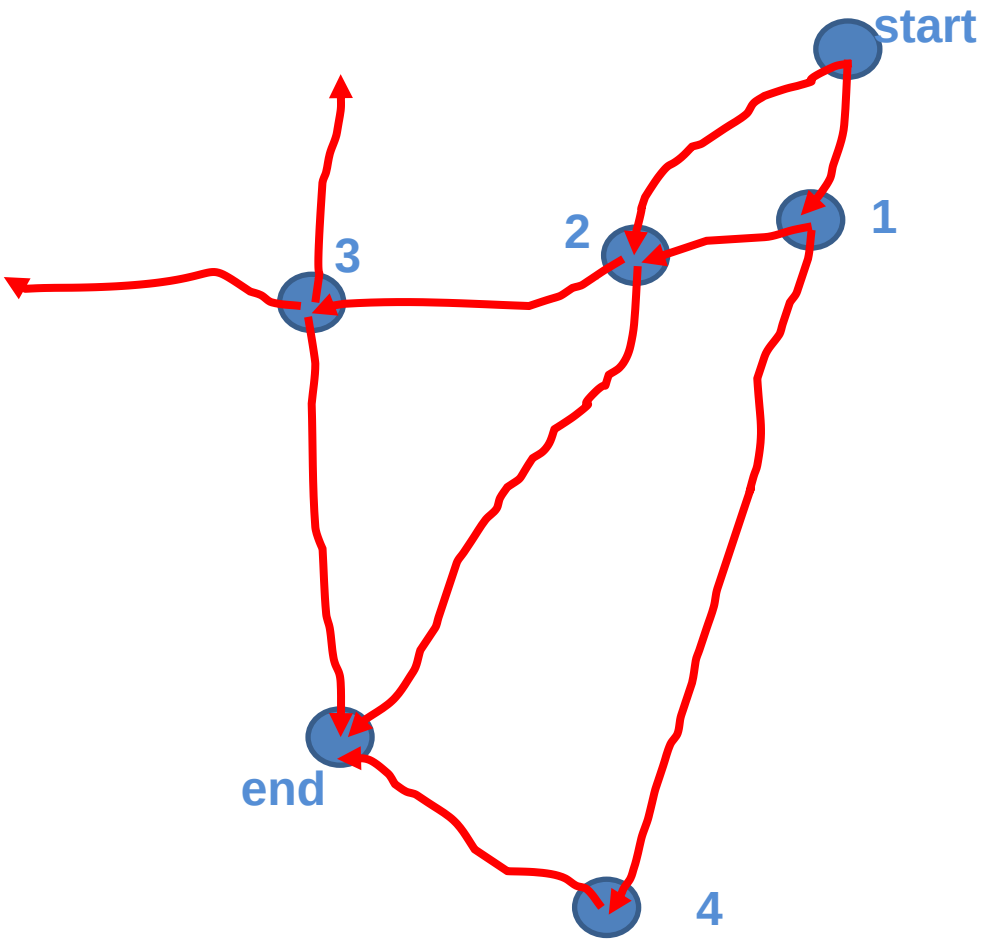
Discrete Structures

- We want to solve problems computationally
- This requires
  - modeling the world
  - devising an algorithm
  - determining the efficiency and correctness of that algorithm
- Discrete structures: how to model the world and think computationally and rigorously

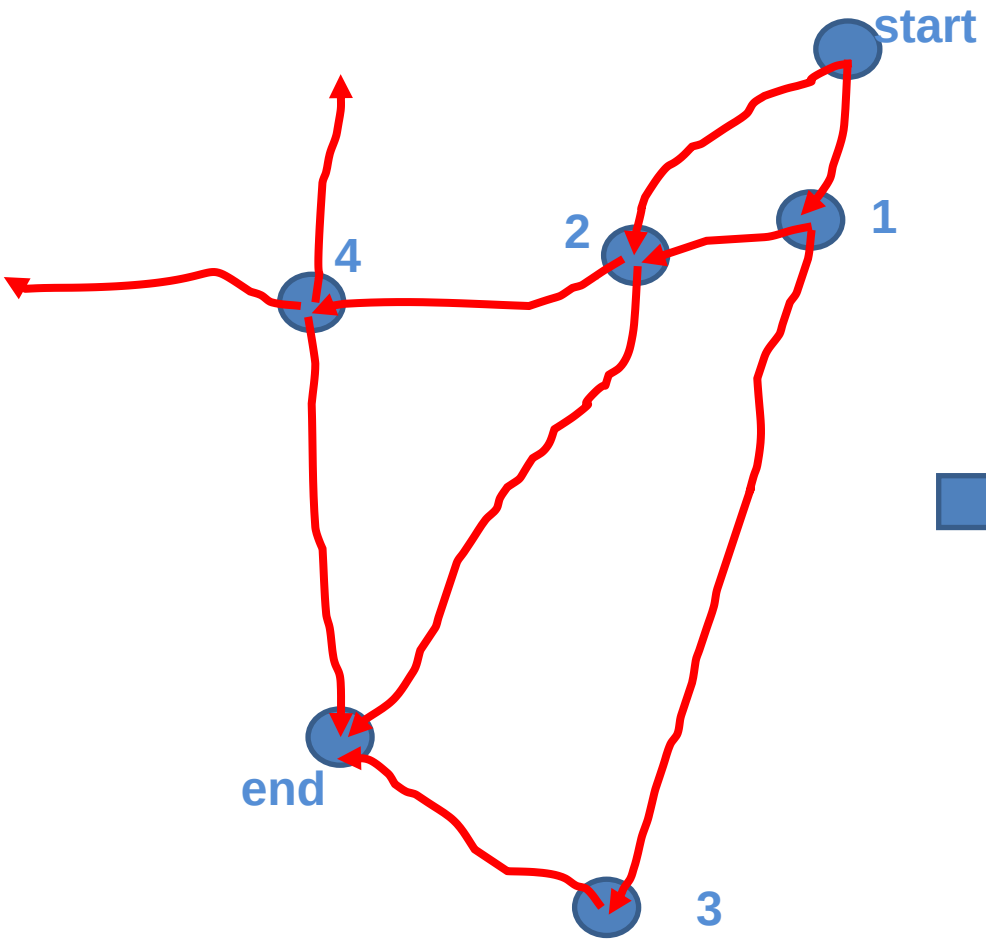
Fastest path from Chicago to Bloomington?

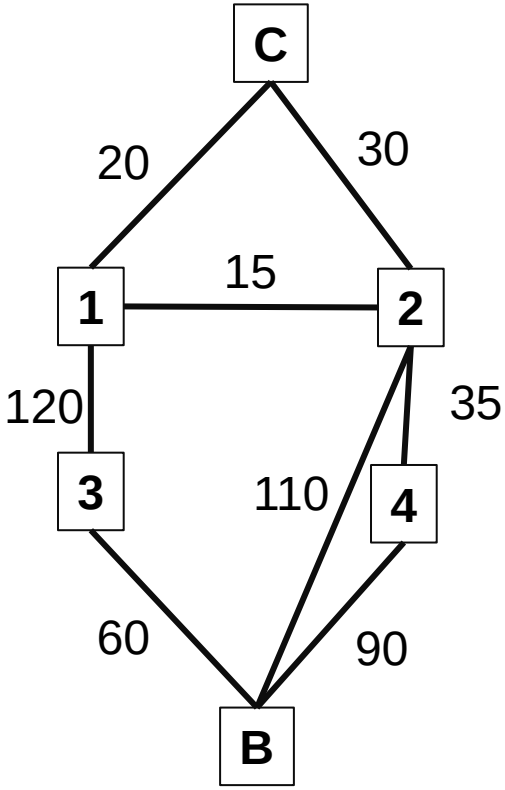
Fastest path from Chicago to Bloomington?

Fastest path from Chicago to Bloomington?



Fastest path from Chicago to Bloomington?

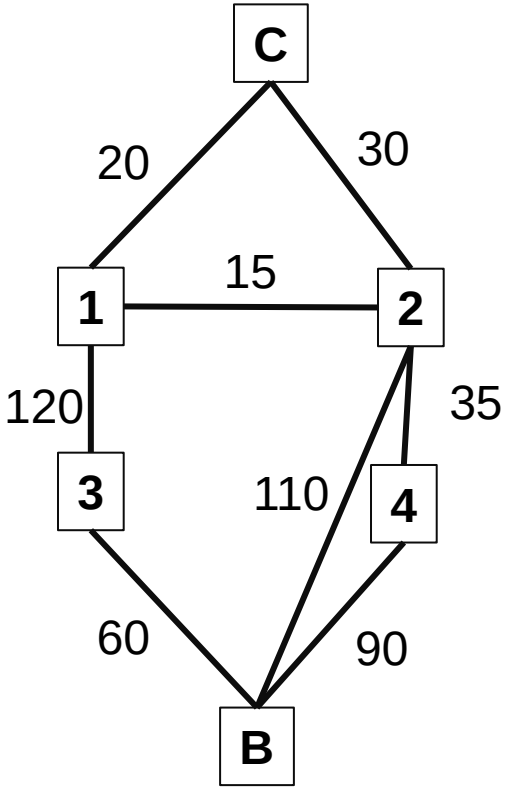




Fastest path from Chicago to Bloomington?

Problem: find shortest path from C to B

- Many solutions: try random paths, enumerate all paths, depth first search



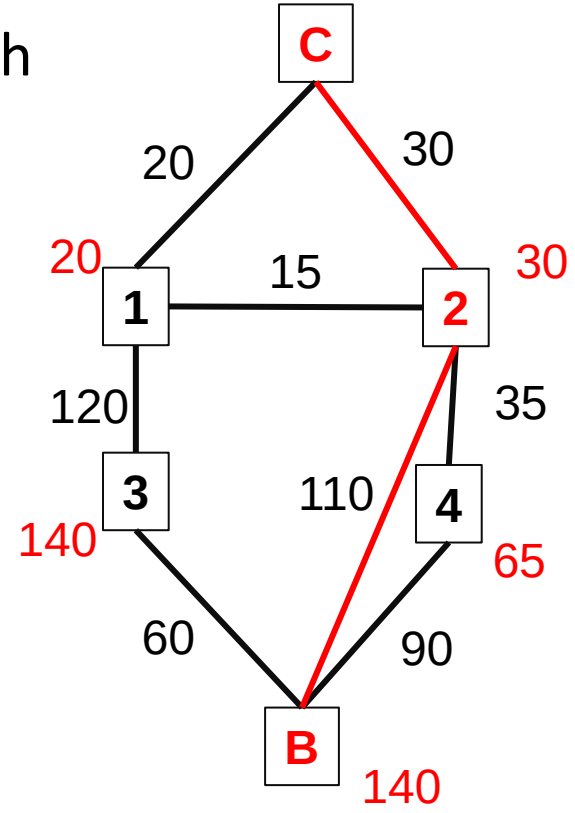
Fastest path from Chicago to Bloomington?

Problem: find shortest path from C to B

- Many solutions: try random paths, enumerate all paths, depth first search

Dijkstra's algorithm

- Explore neighboring nodes and keep track of shortest path to each



Fastest path from Chicago to Bloomington?

Problem: find shortest path from C to B

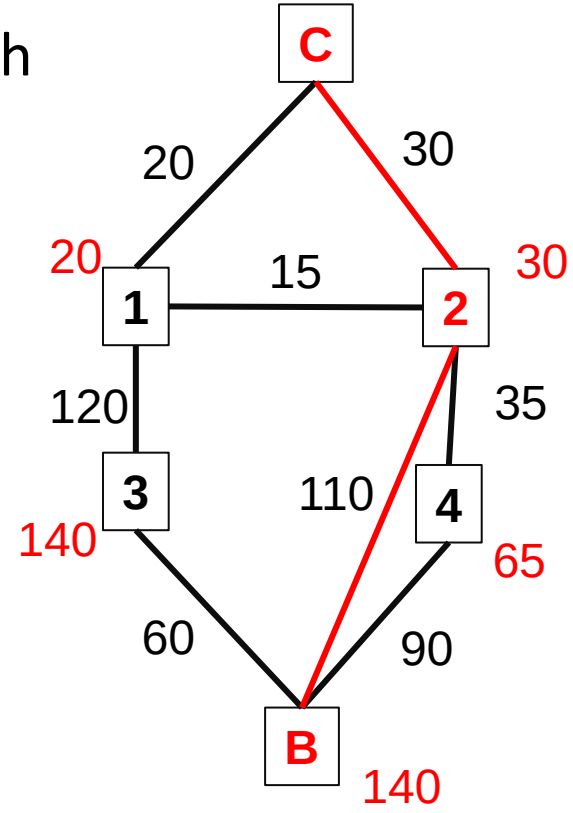
- Many solutions: try random paths, enumerate all paths, depth first search

Dijkstra's algorithm

- Explore neighboring nodes and keep track of shortest path to each

How good is this algorithm?

- Does it find the best solution?
- How long does it take to compute?



Fastest path from Chicago to Bloomington?

Problem: find shortest path from C to B

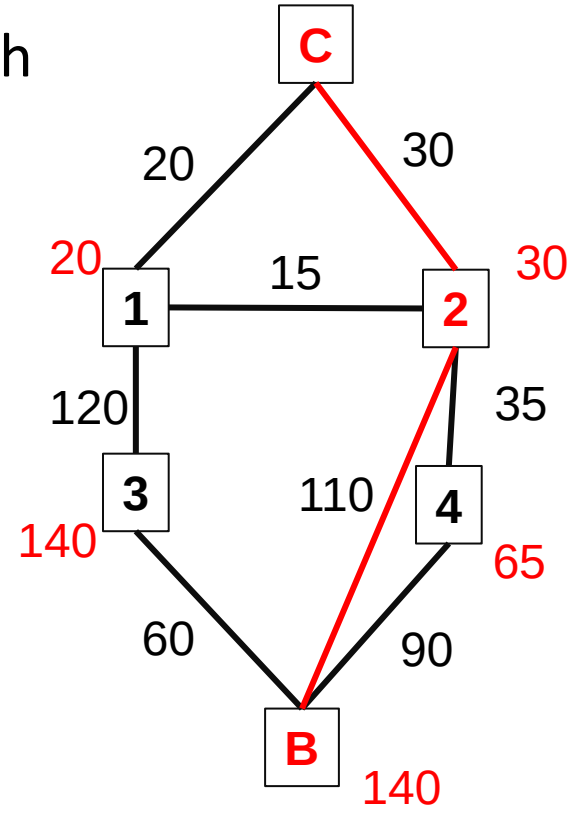
- Many solutions: try random paths, enumerate all paths, depth first search

Dijkstra's algorithm

- Explore neighboring nodes and keep track of shortest path to each

How good is this algorithm?

- Does it find the best solution?
- ~~How long does it take to compute?~~
- How does the computation grow as the number of vertices or edges increases?



Another example: recommender systems

- Shopper buys a “badonkadonk tank”

JL421 Badonkadonk Land Cruiser/Tank
by NAO
(263 customer reviews) | Like (209)

Currently unavailable.
We don't know when or if this item will be back in stock.

Sell on amazon
Sell this Product on Amazon
Just like you, there are thousands of other customers looking for the same product everyday. Click here to start selling on Amazon.

- What will she buy next?

Customers Who Viewed This Item Also Viewed

Page 1 of 7


5.17 CT Fancy Diamond and Platinum Ring (FY 4.01ct, RD 0.62ct, FY 0.54ct)
(1)
$125,000.00


SHOP FOX W1756 25 HP 43-Inch Three Phase Wide-belt Sander
$16,899.00


Darth Vader Supreme Cost Adult SKU-PAS772688
$1,129.19


Sharp, Provolone Piccante Cheese (Whole Wheel) Approximately 60 Lbs
$849.53
Add to Cart
In stock. Ships from and


LOTR Narsil Sword
(1)
$175.99


Zanies 7-Inch Plush Squeaktaculars Dog Toy, Pig
$7.57

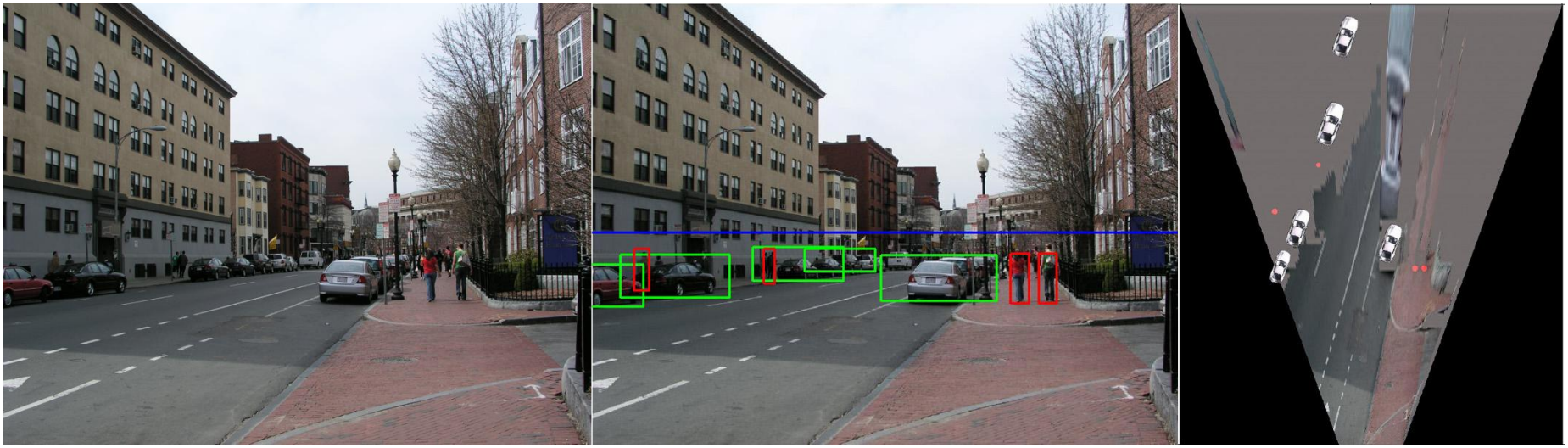
Another example: recommender systems

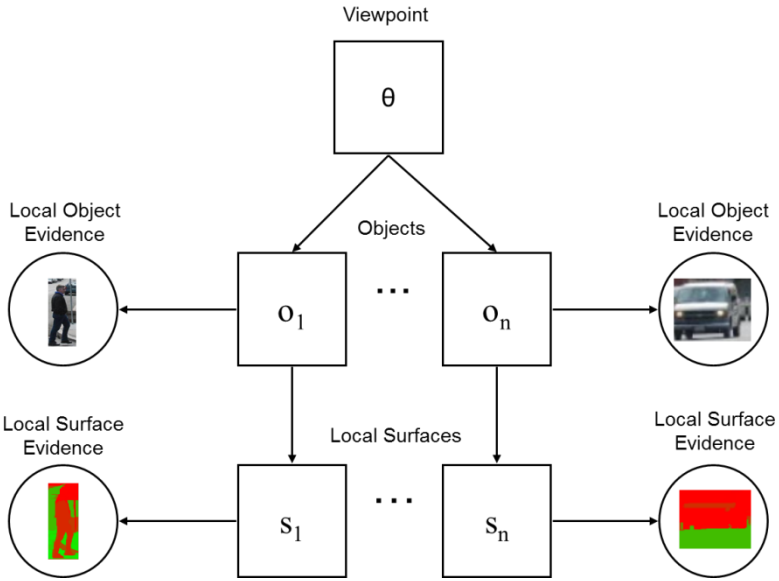
- Shopper buys a “badonkadonk tank”

- What will she buy next?

- Solutions
  - Frequent itemset mining: find sets of items that are often purchased or viewed within the same browser episode
  - Associative graph: learn probability of transition between product pages and compute overall strength of connection

Example from my research: Object Recognition





Graphical Model for Reasoning

What you'll learn in this course

- How to model the world
  - Logic, sets, trees, graphs, functions, etc.
- Strategies to prove and disprove statements
  - Direct, existential, contradiction, contrapositive, induction, etc.
- How to model the computational behavior of algorithms
- How to think computationally

Course Logistics: Weekly Schedule

Time	Monday	Tuesday	Wednesday	Thursday	Friday
Class		**Lecture** 11am-12:15pm		**Lecture** 11am-12:15pm	
Office Hours	**Derek** 11am-12pm		**Derek** 10am-11am		
Discussion Sections				**Discussion Sect.** 2pm-5pm	**Discussion Sect.** 9am-11am
Homework		**Reading Quiz** due 11:59pm (material from that day)	**Mini-HW** due 9pm **Long-form HW** due 9pm (material from previous week)	**Reading Quiz** due 11:59pm (material from that day)	

- Tues: come to class, read textbook, complete reading quiz
- Wed: submit HW if not done already
- Thurs: come to class, read textbook, complete reading quiz
- Thurs/Fri: attend one discussion section
- Weekend: complete as much of HW as possible
- Throughout week: attend office hours, as needed

Course Logistics: Grading

- Homework/quizzes (35%)
  - Reading quiz: 5% (lowest 3 grades dropped)
    - Auto-graded: can see score, fix, and resubmit
  - Mini-HW: 10% (lowest dropped)
    - Auto-graded for fast feedback
  - Long-form HW: 20% (lowest dropped)
- Exams (65%)
  - Midterm 1: 20%
  - Midterm 2: 20%
  - Final Exam: 25%
- Late policy
  - Generally, late assignments not accepted
  - See website for details
- Cheating
  - http://courses.engr.illinois.edu/cs173/sp2013/B-lecture/Info/cheating.html

Course Logistics: Grading

- Thresholds for guaranteed grades

  A 94%

  A- 90%

  B- 80%

  C- 70%

  D- 60%

- We reserve right to *curve up*

- In previous terms, this course has given about 20% A's, 30% B's, 30% C's, 15% D's, and 5% F's

Reading and Textbook

- Margaret Fleck's "Building Blocks for Theoretical Computer Science" http://www.cs.uiuc.edu/~mfleck/building-blocks/

- Optional book: Rosen "Discrete Mathematics and its Applications", 5th to 7th edition
  - More detail, practice problems

Getting help

- Discussion sections and office hours

  http://courses.engr.illinois.edu/cs173/sp2013/B-lecture/Info/staff.html

  - Starts next week!

- Newsgroup: piazza.com

  https://piazza.com/class#spring2013/cs173b

  Password: athena

- Major problems

  http://courses.engr.illinois.edu/cs173/sp2013/B-lecture/Info/help.html

Tips

- This class is very difficult for most students
- New ways of thinking require lots of practice
- Basic approach to taking this class
  - Come to lectures, take notes
    - If you have trouble following lecture, read the relevant sections of the book first
  - Try the homework
  - Read the online textbook
  - Complete the homework (use TAs, Piazza, Rosen book, online resources as needed)
- Preparing for exams
  - Find extra problems from past homeworks and exams (see past courses linked via website)
  - Do a few extra problems every week in a topic that is difficult for you
  - 2 weeks before exam, start doing extra problems every day until all the past homework and exam problems are easy

Prerequisites, testing out

- Prerequisites
  - Calculus I (Math 220 or 221 or 234) and CS 125 or ECE 190 or CS 101 or a high grade in INFO 103
  - If you aren't sure whether you have the right background, speak to me (or Margaret Fleck)

- Proficiency exam
  - Saturday, Jan 19th 1:30-4:30pm in 1404 Siebel Center
  - Conflict exam: 7-10pm Thursday January 17th in 2405 Siebel
  - For the main exam, you must sign up by 11pm on Thursday the 17th. For the conflict exam, you must sign up by 8am on Wednesday the 16th.
  - Details here: https://wiki.engr.illinois.edu/display/cs173/Computer+Science+Proficiency+Exams

To do now

- Read syllabus and course information online
  - Syllabus available at front
- Enroll in Piazza to get announcements
  - Access code: athena
- Make sure you can access Moodle
  - First “reading quiz” on Thursday (due at midnight)
- If you’re not yet registered, come up front after lecture and put your name on a list
- Note: no discussion section this week!

Thank you

- Next class: Propositional Logic