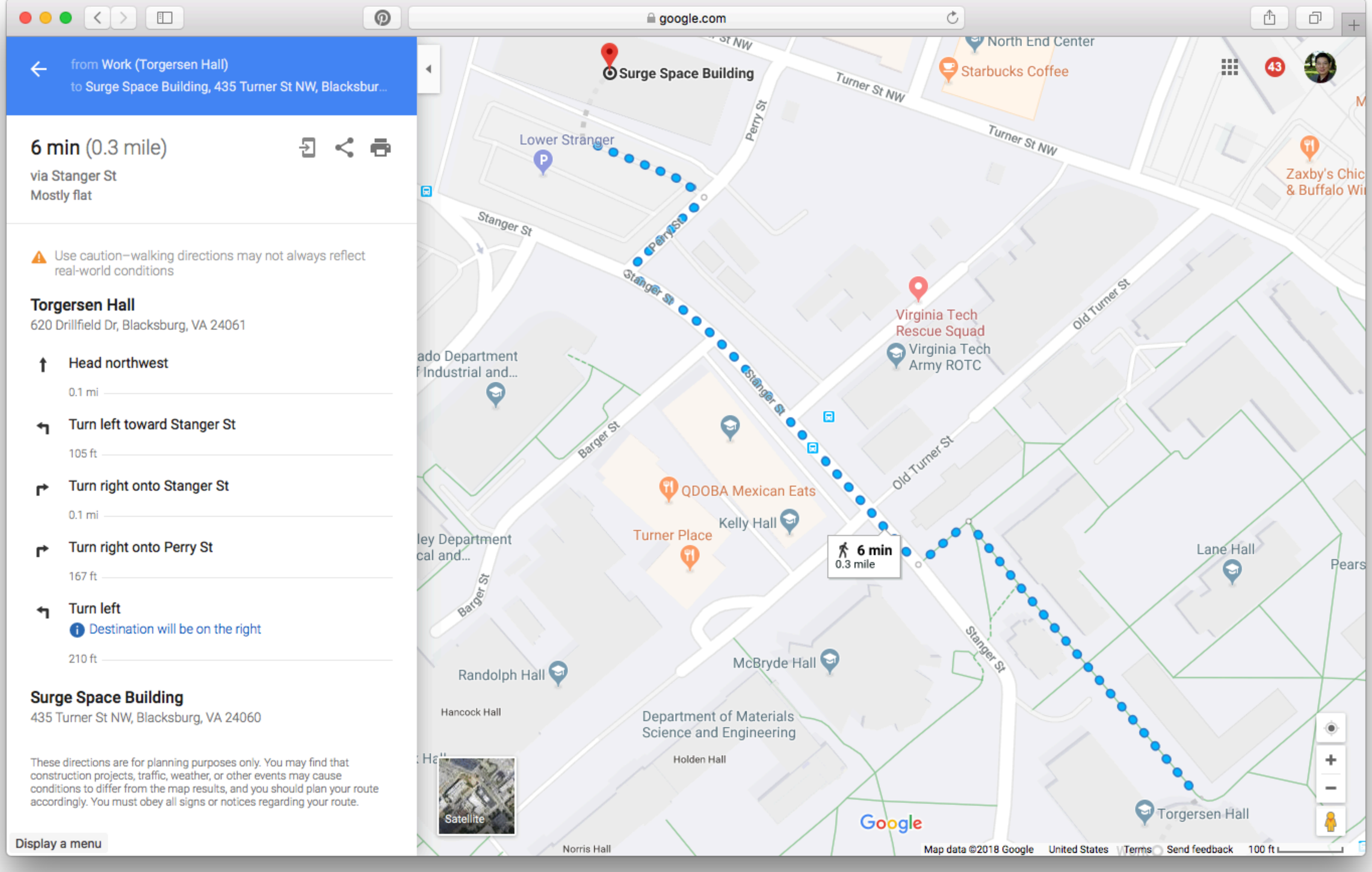

google.com
from Work (Torgersen Hall)
to Surge Space Building, 435 Turner St NW, Blacksbur...
6 min (0.3 mile)
via Stanger St
Mostly flat
Use caution–walking directions may not always reflect real-world conditions
Torgersen Hall
620 Drillfield Dr, Blacksburg, VA 24061
Head northwest
0.1 mi
Turn left toward Stanger St
105 ft
Turn right onto Stanger St
0.1 mi
Turn right onto Perry St
167 ft
Turn left
Destination will be on the right
210 ft
Surge Space Building
435 Turner St NW, Blacksburg, VA 24060
These directions are for planning purposes only. You may find that construction projects, traffic, weather, or other events may cause conditions to differ from the map results, and you should plan your route accordingly. You must obey all signs or notices regarding your route.
Display a menu
Surge Space Building
North End Center
Starbucks Coffee
Turner St NW
Lower Stranger
Perry St
Stanger St
Zaxby's Chic & Buffalo Wi
Virginia Tech Rescue Squad
Virginia Tech Army ROTC
Old Turner St
Barger St
QDOBA Mexican Eats
Kelly Hall
Turner Place
6 min
0.3 mile
Lane Hall
Randolph Hall
McBryde Hall
Hancock Hall
Department of Materials Science and Engineering
Holden Hall
Torgersen Hall
Satellite
Google
Map data ©2018 Google
United States
Terms
Send feedback
100 ft

Search Algorithms: Plan

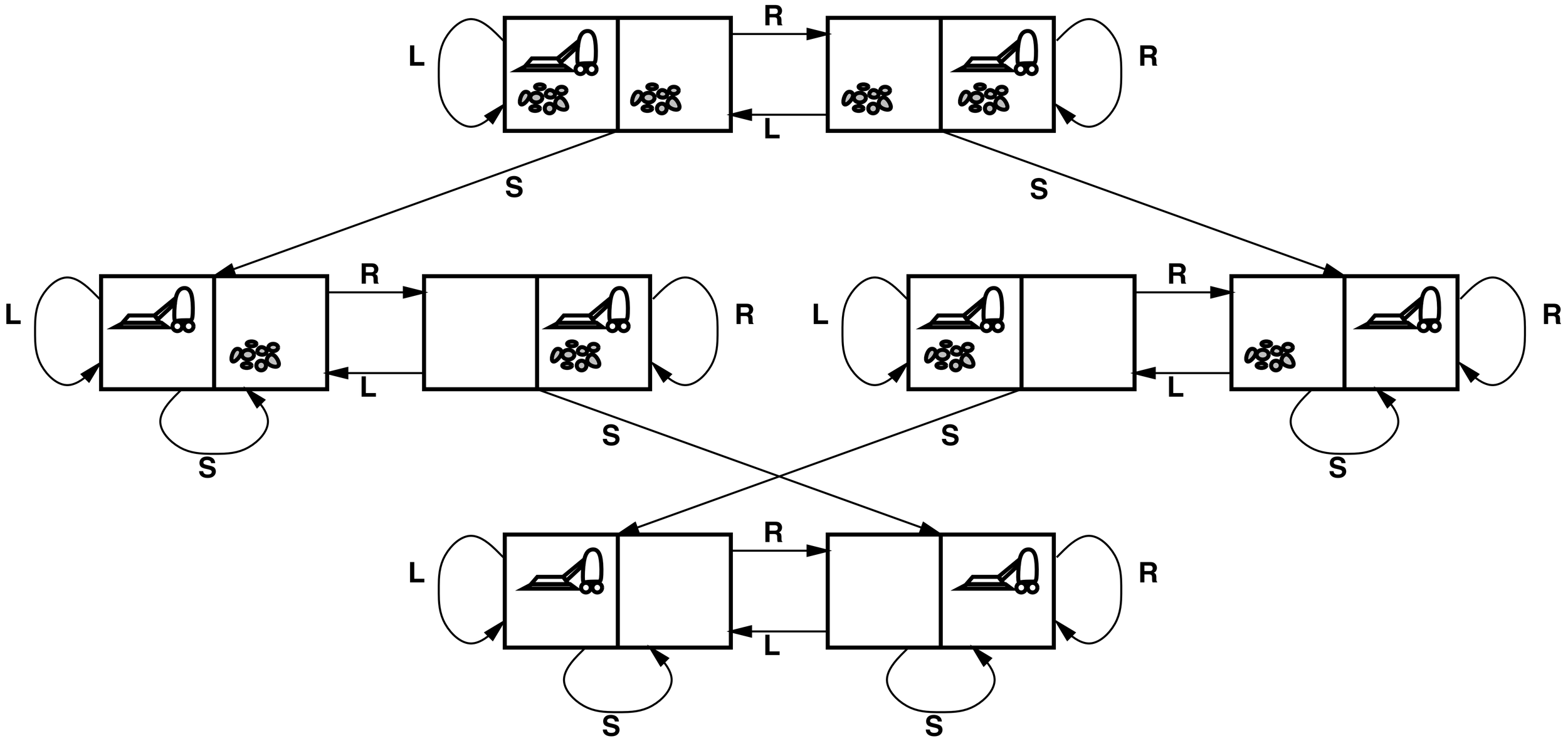
- Environments as search spaces
- Properties of search algorithms
- Uninformed search algorithms
  - Breadth-first search
  - Depth-first search

Search Spaces

- Initial state
- Actions, transitions
- Goal state(s)
- Transition costs

Problem Description

- Search tree/graph
- Expanding a node
- Frontier


L
R
L
R
S
S
L
R
L
R
S
S
L
R
L
R
S
S
L
R
L
R
S
S

Properties of Search Algorithms

- Completeness: algo will find a goal state if one exists
- Optimality: algo will find the least-cost path to goal state
- Time complexity: number of operations needed to find goal state
- Space complexity: amount of memory needed to find goal state
  - Complexity usually expressed in terms of *depth* of goal and *branching factor*

function TREE-SEARCH(*problem*) **returns** a solution, or failure

initialize the frontier using the initial state of *problem*

repeat:

if the frontier is empty **then return** failure

extract a node from the frontier

if the node contains a goal state **then return** the solution

expand the chosen node, adding the resulting nodes to frontier

(FIFO: Breadth-first, LIFO: Depth-first) Graph?

Additional Variants

- Depth-limited search:
  - Run DFS, but don't search deeper than depth limit **L**
- Iterative-deepening:
  - Run depth-limited search with depth **L** = 1, then **L** = 2, …
  - Until goal state found
- Uniform-cost search:
  - Variation of breadth-first. Expand nodes in order of cost from initial state.

Properties

	Complete	Optimal	Time	Space
Breadth-First	Yes	Yes if uniform cost, no otherwise	$O(b^d)$	$O(b^d)$
Depth-First	Yes if and only if (iff) finite	No	$O(b^{max\_depth})$	O(b(max_depth))
Depth-Limited	No	No	$O(b^L)$	O(bL)
Iterative Deepening	Yes	Yes if uniform cost, no otherwise	$O(b^d)$	O(bd)
Uniform Cost	Yes iff costs are positive	Yes iff costs are positive	$O(b^{max\_depth})$	???

Graph Search

- Simple modification: only add nodes to frontier if
  - they are not already in the frontier
  - and they have not already been expanded

Reading and Next Class

- Today's material: AIMA 3–3.4
- Next class: AIMA 3.5–3.6