ASTRO-K: Finding Top-k Sufficiently Distinct Indoor-Outdoor Paths

Vasilis Ethan Sarris,
Constantinos Costa, Panos K. Chrysanthis

Department of Computer Science, University of Pittsburgh, USA
vas82@pitt.edu, {costa.c, panos}@cs.pitt.edu

University of Pittsburgh

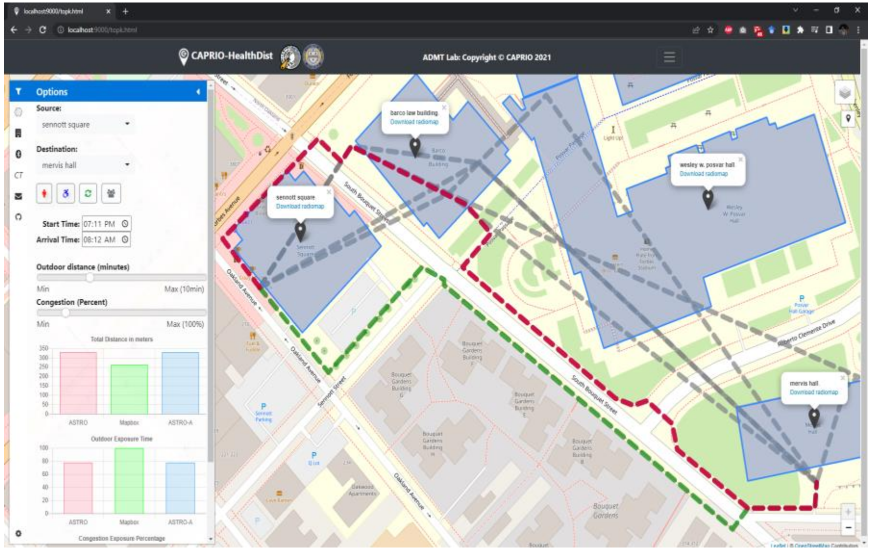
The 2nd IEEE Intl. Workshop on ALgorithms for Indoor Architectures and Systems
Paphos, Cyprus, June 6, 2022

Outline

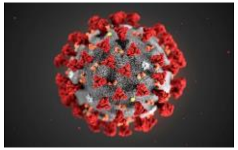
- Motivation
- Problem Formulation
- ASTRO-K
- Experimental Evaluation
- Conclusions & Future Work

Motivation

- CAPRIO: Indoor-outdoor pedestrian path recommendation system that suggests the shortest distance path while satisfying user-mobility constraints
- ASTRO: Constraint-based path-finding algorithm which factors in the predicted congestion of a space when constructing a path
- Observation: ASTRO may inadvertently contribute towards congestion problems by funneling people into a single area



Congested Spaces

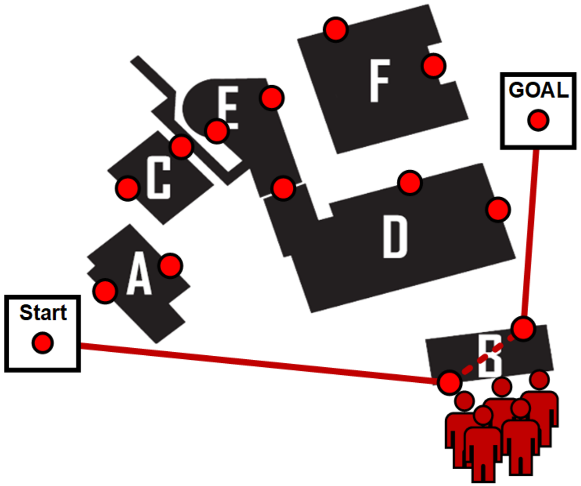


COVID-19 Distancing

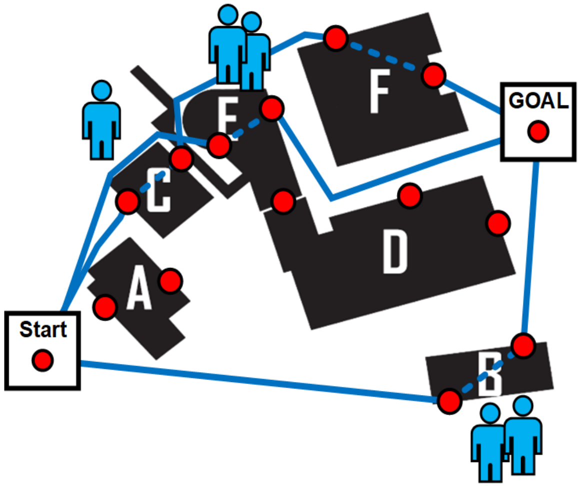
Mobility disability

Problem & Objective

ASTRO



ASTRO-K



- **Outdoor Vertices** – comprised of Indoor graphs and represent buildings
- **Indoor Graph** – the path between any two doors
- **Indoor Vertices** – doors allowing people to enter and exit buildings

ASTRO-K: Problem Formulation

- **ASTRO-K:** Given an Indoor-Outdoor graph, a source Outdoor
- vertex, a terminal Outdoor vertex, a set of constraints, and k,
- find the **top-k sufficiently distinct paths** from the source Outdoor vertex to the terminal Outdoor vertex which satisfies the given constraints.

Congestion (C) – max congestion (w.r.t the max amount of people) within a 3m^2 cell during a 5-minute window.
Outdoor Exposure (E) – max amount of time allowed to traverse any given edge.
Time Limit (T) – max time allowed to traverse a path
Path Similarity (θ) – max percentage of total edge distance shared between any two paths in the result set

Components of ASTRO-K

- **ASTRO**: Given an Indoor-Outdoor graph along with a source and a destination point, produce a path with the minimal total time from the source to the destination which satisfies the constraints.

- **ESX**: Given a source and terminal node, find k paths that are sufficiently dissimilar to each other with respect to a similarity threshold θ, and as short as possible.

- ASTRO is based on A* Algorithm that at each step it picks the node according to a score ‘f’ -> f = g+h
  - g = the movement cost
  - h = the estimated movement cost

- ESX is a performance-oriented heuristic algorithm that removes edges from the graph incrementally and then computing the shortest path on the updated graph

ASTRO-K Algorithm

Algorithm 2 *ASTRO-K*

Input: s: source, t: terminal, O: Outdoor Vertices, Π: Constraints;
Output: P: Best Paths

init List P, Priority Queue PQ, Set E_{DNR}, Set E_R
$P.append(ASTRO(s, t, O, E_R, \Pi))$
for $e \in P.tail()$ **do**
  $PQ.push(e)$
while $|P| < k$ **and** PQ not empty **do**
  $p \leftarrow P.tail()$
  while PQ not empty **and** $Sim(p, P) > \theta$ **do**
    $e \leftarrow PQ.pop()$
    if $e \in E_{DNR}$ **then**
      continue
    $E_R \leftarrow E_R \cup \{e\}$
    $p \leftarrow ASTRO(s, t, O, E_R, \Pi)$
    if $p = NULL$ **then**
      $E_R \leftarrow E_R - \{e\}$
      $E_{DNR} \leftarrow E_{DNR} \cup \{e\}$
  if $Sim(p, P) < \theta$ **then**
    $P.append(p)$
    $PQ.clear()$
    for $e \in p$ **do**
      $PQ.push(e)$
return P

Algorithm 1 Modified ASTRO used in Algorithm 2

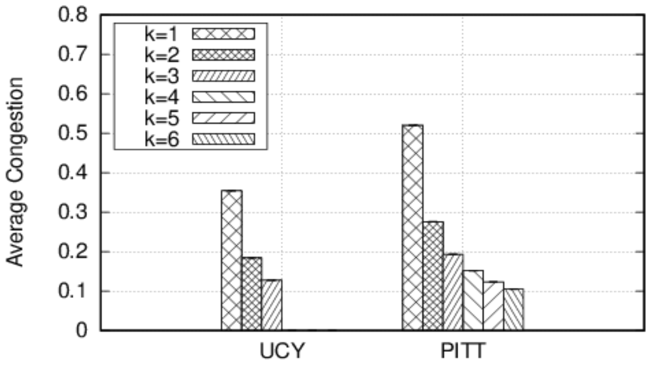
Input: s: source, t: terminal, O: Outdoor Vertices, E_R: Edges Removed, Π: Constraints;
Output: p: Best Path;

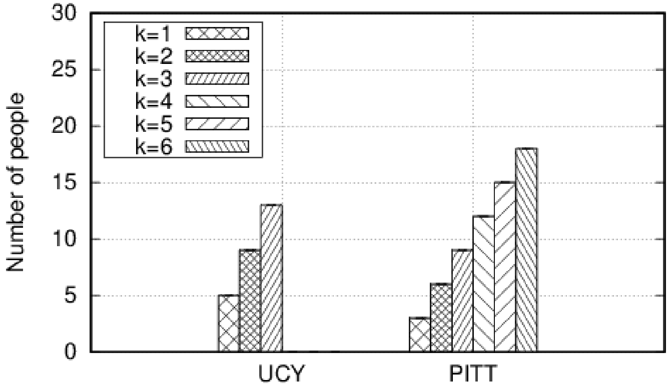
init PQ $OPEN$, Set $CLOSED$
init Outdoor Vertex $start$ with s
$OPEN.push(start)$
while $OPEN$ not empty **do**
  $curr \leftarrow OPEN.pop()$
  if $curr = NULL$ **then**
    return $ReconstructPath(curr)$
  if $curr \notin CLOSED$ **then**
    $CLOSED \leftarrow CLOSED \cup \{curr\}$
    for $o_i \in \{O - CLOSED\}$ **do**
      $IndoorGraph \leftarrow o_i[IndoorGraph] - E_R$
      for $in_i \in IndoorGraph$ **do**
        $\vec{s_1} \leftarrow Status(curr[out], in_i, curr[g])$
        for $out_i \in \{IndoorGraph - \{in\}\}$ **do**
          $now = curr[g] + \vec{s_1}[total]$
          $\vec{s_2} \leftarrow Status(in_i, out_i, now)$
          $\vec{s} \leftarrow \vec{s_1} + \vec{s_2}$
          $g \leftarrow curr[g] + \vec{s}[total]$
          if $g < o_i[g]$ **and** $Check(\Pi, \vec{s})$ **then**
            init Outdoor Vertex $toAdd$
            $OPEN.push(toAdd)$

Experimental Methodology

- **Datasets**:
  - **PITT**: University of Pittsburgh campus and consists of 9 buildings, 2-6 doors per building
  - **UCY**: University of Cyprus campus and consists of 9 buildings, 2-7 doors per building
- **Congestion Data**:
  - Camera analysis on a 2-hour session and generate congestion data using the procedure EPICGen generator using $\mu\_a = 4$, $\sigma\_a = 1$, $\mu\_d = 1$, $\sigma\_d = 0.2$ and a walking speed of 1.4 m/s
- **Metrics:**
  - **Average Congestion (C):** average percentage of congestion (with respect to the maximum amount of people) within a $3m^2$ cell of a building during a 5 minute window.
  - **Number of People (P):** amount of people introduced within a 5 minute window.

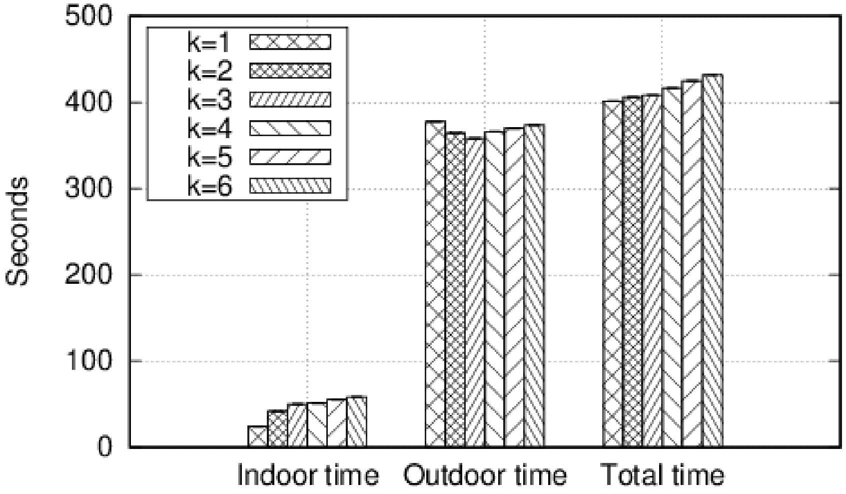
Experiments





- **Experiment 1:** average congestion of the recommended paths is reduced by 4.5X for the PITT dataset with K = 6 and 2.6X for the UCY dataset with K = 3.
- **Experiment 2:** the number of people which can follow the recommended paths is increased by 5.9X for the PITT dataset with K = 6 and 2.8X for the UCY dataset with K = 3.

Experiments (cont.)



Time	k=1	k=2	k=3	k=4	k=5	k=6
Indoor	23.94	41.62	49.88	50.54	55.09	58.12
Outdoor	377.215	364.52	358.12	365.88	369.60	373.25
Total	401.16	406.14	408.01	416.43	424.70	431.38

- **Experiment 3:** average total time of the recommended paths is increased by 7% for K = 6. Furthermore, the average indoor time increases by 2.4X and the average outdoor time decrease by 2% for K = 6.

Conclusions & Future Work

- Major Contribution:
  - A novel top-k sufficiently distinct path finding algorithm ASTRO-K, which is able to recommend paths to more people without inadvertently congesting an area
- Future Work:
  - Include balancing or scheduling of route recommendations
  - Make Congestion a first-class constraint

ASTRO-K: Finding Top-k Sufficiently Distinct Indoor-Outdoor Paths

Thank You! Any Questions?

Vasilis Ethan Sarris,
Constantinos Costa, Panos K. Chrysanthis

Department of Computer Science, University of Pittsburgh, USA
vas82@pitt.edu, {costa.c, panos}@cs.pitt.edu

University of Pittsburgh

The 2nd IEEE Intl. Workshop on ALgorithms for Indoor Architectures and Systems
University of Cyprus (UCY), Paphos, Cyprus, June 6, 2022