Geometric Path-finding Algorithm in Cluttered 2D Environments

Nafiseh Masoudi

Advisor: Dr. Georges Fadel

Committee: Dr. Margaret Wiecek,
Dr. Joshua Summers, and
Dr. Gang Li

Department of Mechanical Engineering,
Clemson University

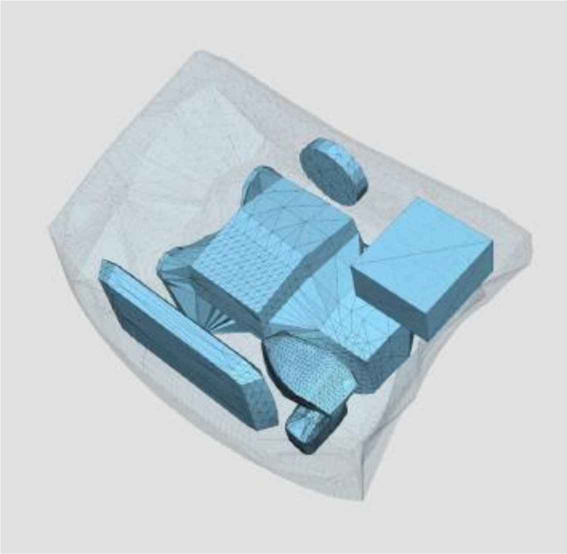
Outline

- Motivation
- Research Objectives
- Literature Review
- 2D Routing Problem
- Research Approach
- Conclusions
- References

nmasoud@clemson.edu
http://www.clemson.edu/ces/cedar

Background

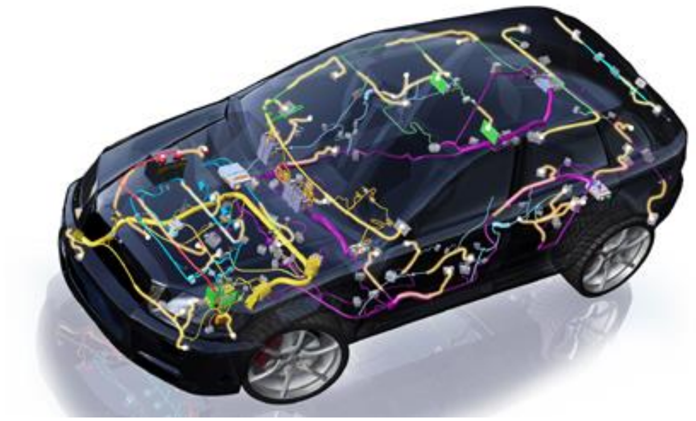
- Packaging Optimization: packaging of components in vehicle under-hood to achieve an optimum center of gravity, accessibility, survivability, dynamic behavior, etc.



- Connecting the components in an optimal way using cables, wires, and harnesses + placing breakouts

Motivations

- Cable harnesses, the third heaviest and costliest component in a car (Matheus,2015)
- Their layout is currently performed in CAD systems by human designers
- Current process lacks automation and the final solution often times is NOT optimal
- Routing is important in assembly planning, robot motion planning and geographic infor



Research Objectives

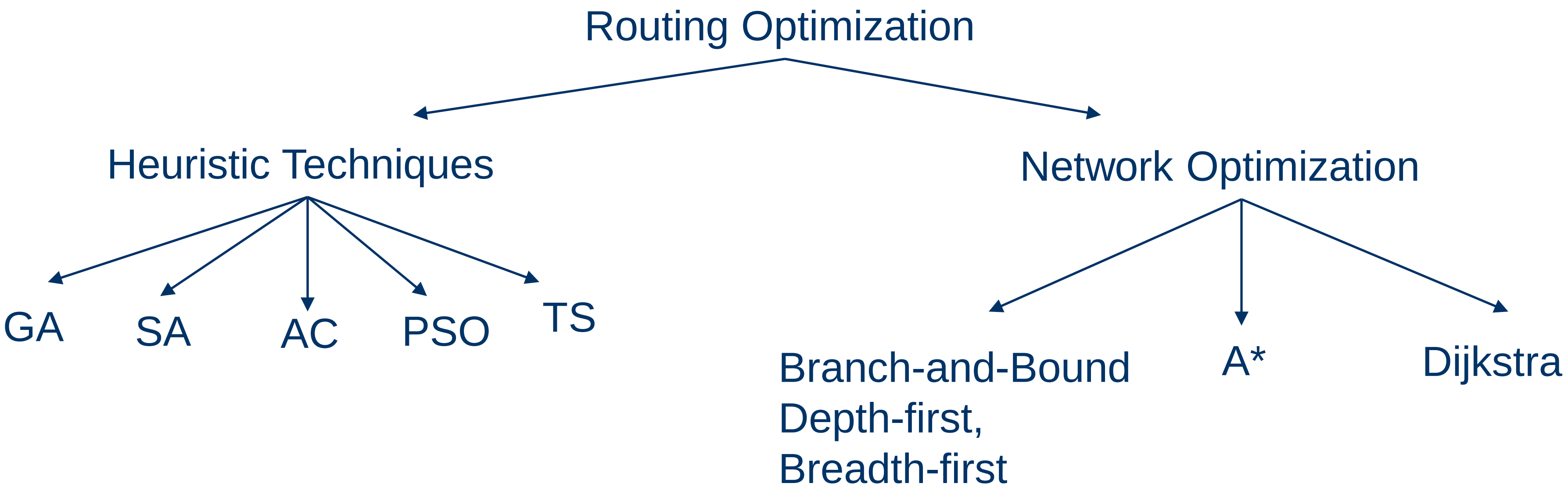
- Automate the routing process of wires, hoses, and cables (one dimensional components) in electromechanical systems → ultimate objective
- Avoid interference with other components of the environment
- Minimize the total weight of the harness (length) → goal
- Improve the efficiency of the optimizer through the appropriate choice of a graphical representation for the workspace and the free space

nmasoud@clemson.edu
http://www.clemson.edu/ces/cedar

Path Planning Literature Review

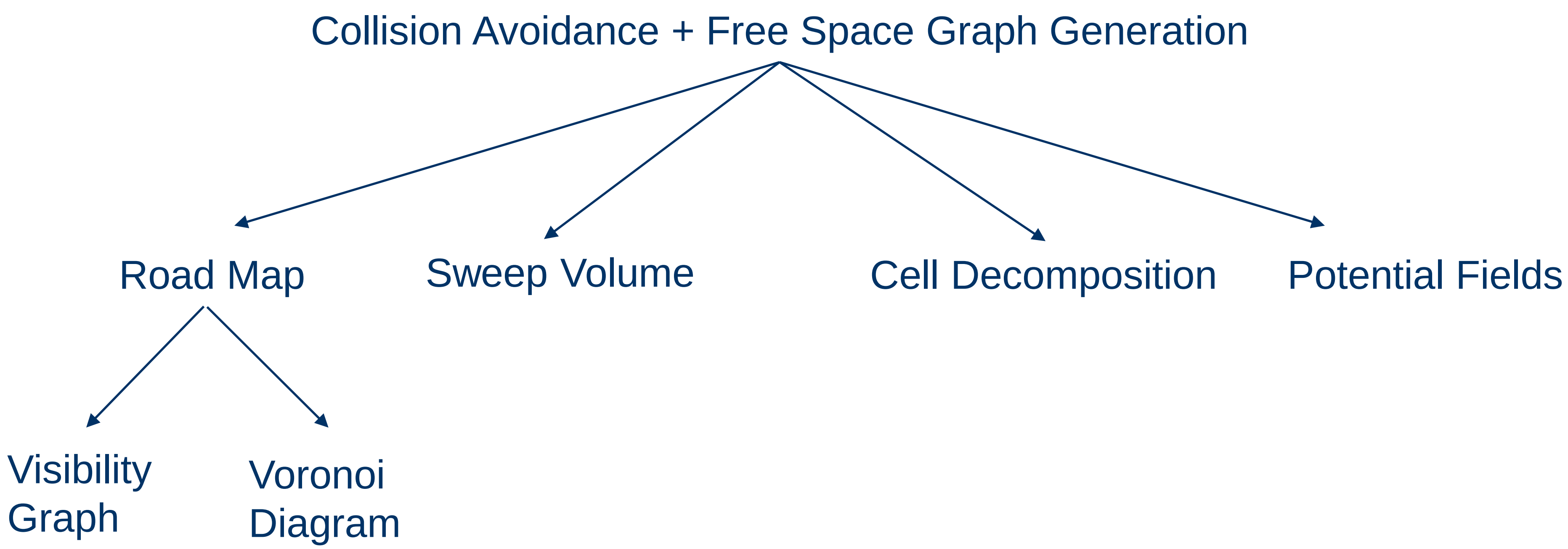
- An automatic pipe router using approximate cell decomposition and A* search algorithm is described in [1]
- 3D pipe routing problem is solved in [3], using convex hulls of barriers and visibility graphs to find candidate segments
- Chen and Sandurkar [4] solve 3D pipe routing using tessellations of obstacles and Genetic Algorithms
- Conru [6] uses GA to find near-optimal solutions to the 3D cable harness routing problem with collision avoidance constraints using cell decomposition
- Automotive wire routing and sizing for weight minimization is addressed in [7] using the Minimal Steiner Tree algorithm

Summary



nmasoud@clemson.edu
http://www.clemson.edu/ces/cedar

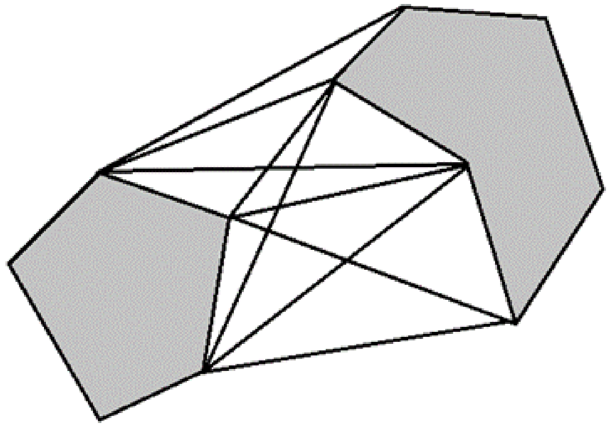
Summary

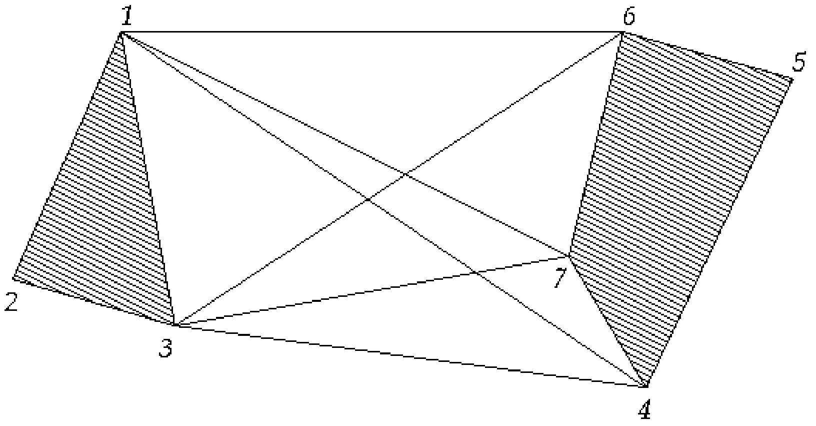


nmasoud@clemson.edu
http://www.clemson.edu/ces/cedar

Visibility Graph

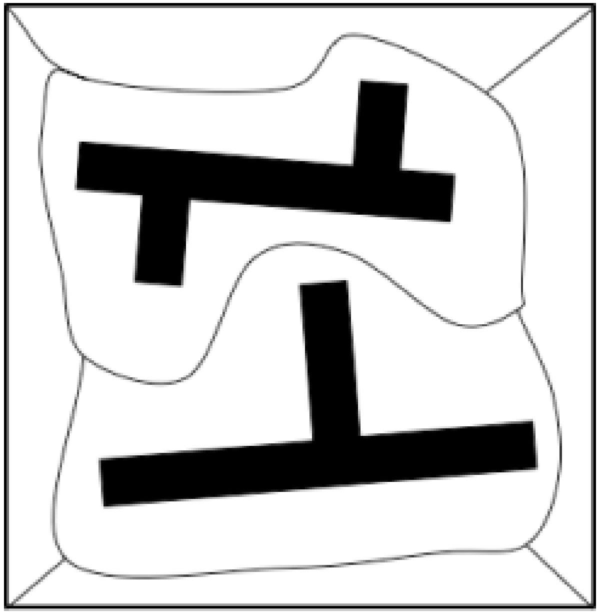
- A way to generate the collision free graph
- A Visibility graph: a finite set of nodes and edges. The nodes can “see” one another in the sense that the common edge does not meet the interior of any obstacles

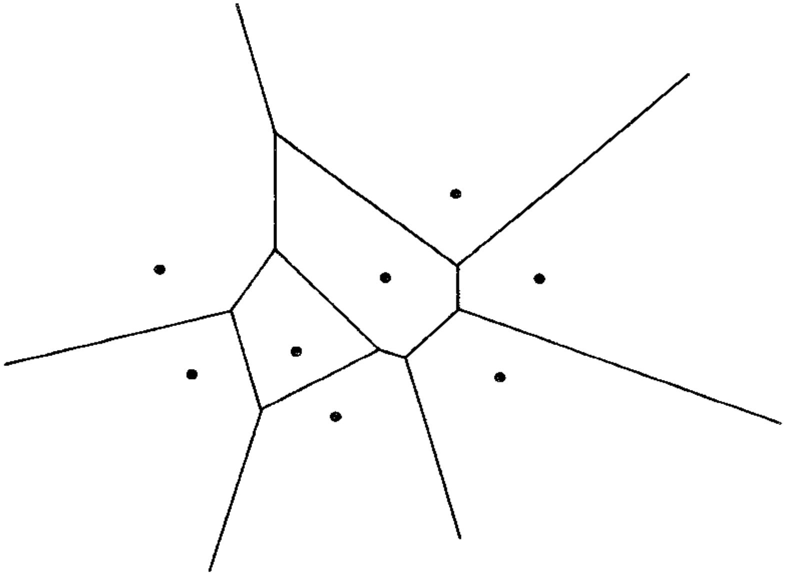




Voronoi Diagram

- Voronoi diagram of n sites partitions the workspace into n convex regions such that any point on an edge is equidistant from exactly two sites, hence generating max-clearance path



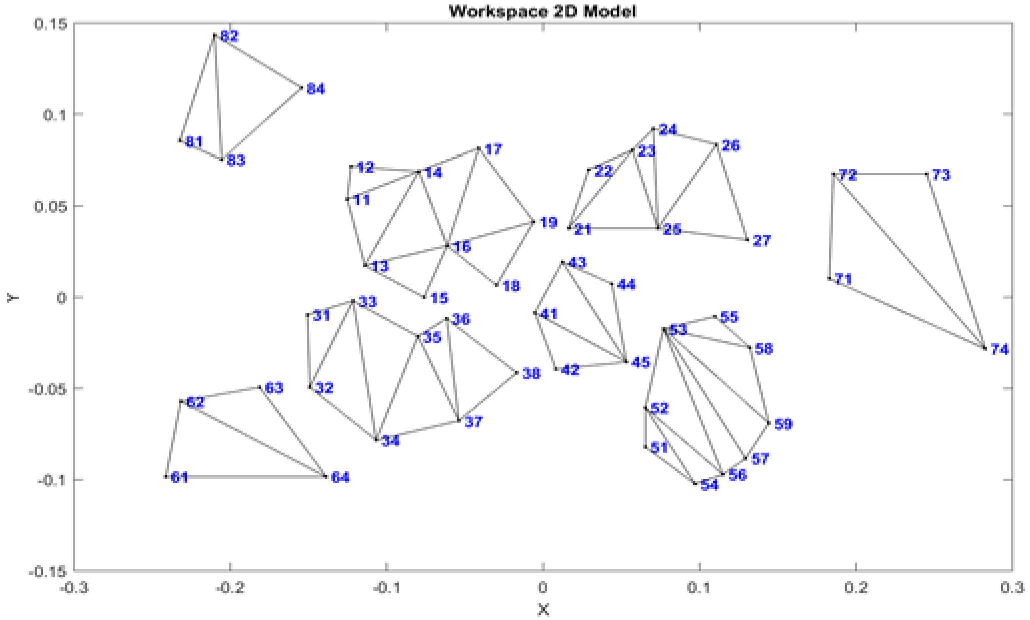


Issues with the previous work

- Visibility graph generates the graph of the visible nodes from a vertex through extensive search of the entire workspace
- Voronoi does not necessarily result in the shortest path
- Roadmaps that work well with 2D routing problems are not fast enough since they explore the entire workspace→ memory and time issues
- Non convex shapes are not well-addressed using the previous techniques.

Geometric representation

- Using tessellations, STL data of the workspace
- Efficient handling of convex as well as non convex shapes

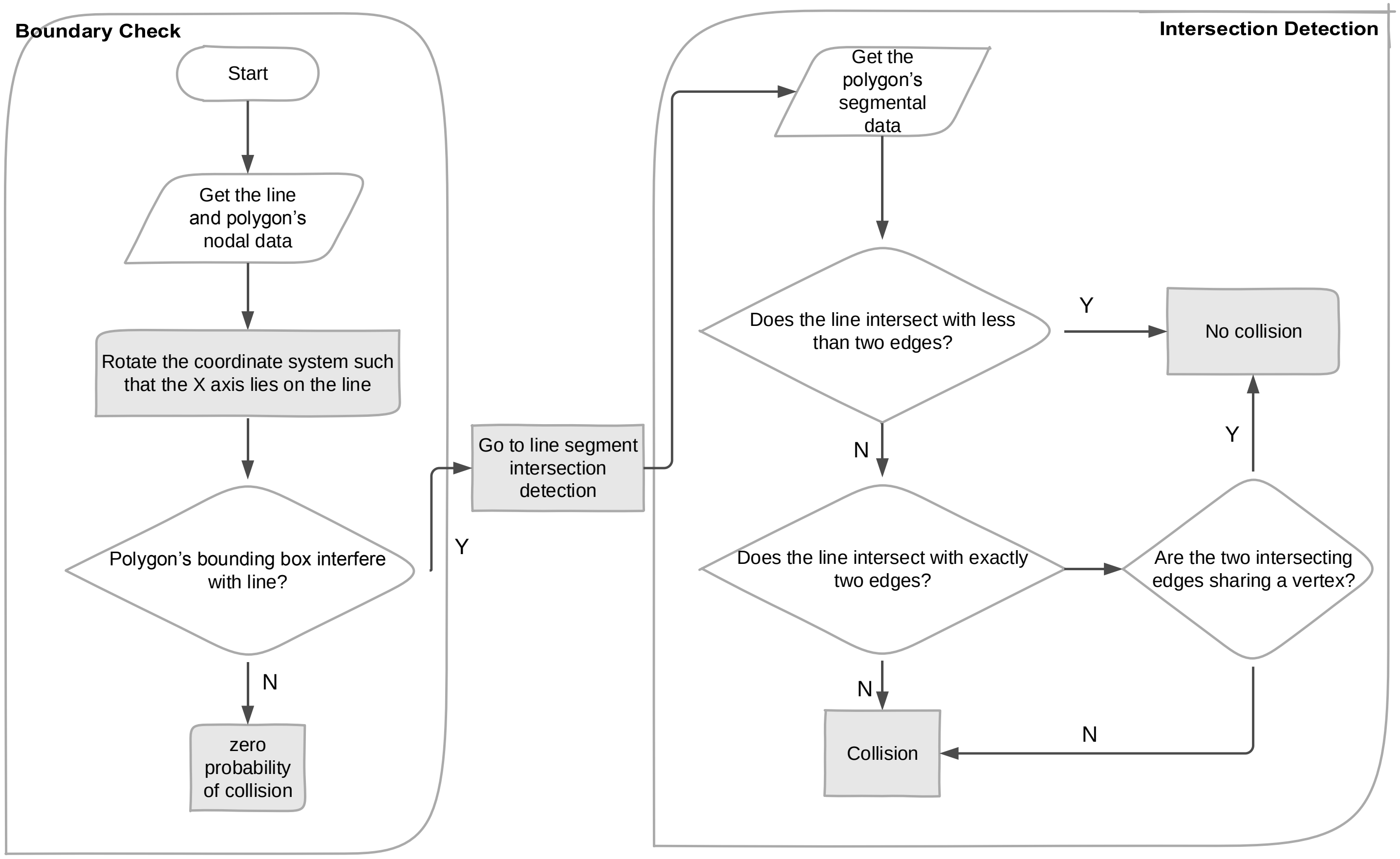


Node numbering:

- first digit=object number
- Other digits=vertex number

Intersection Detection

- Bi-level Intersection detector



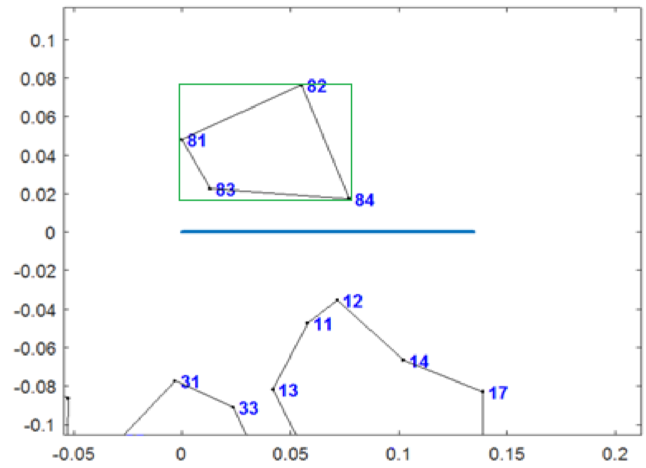
Intersection Detection

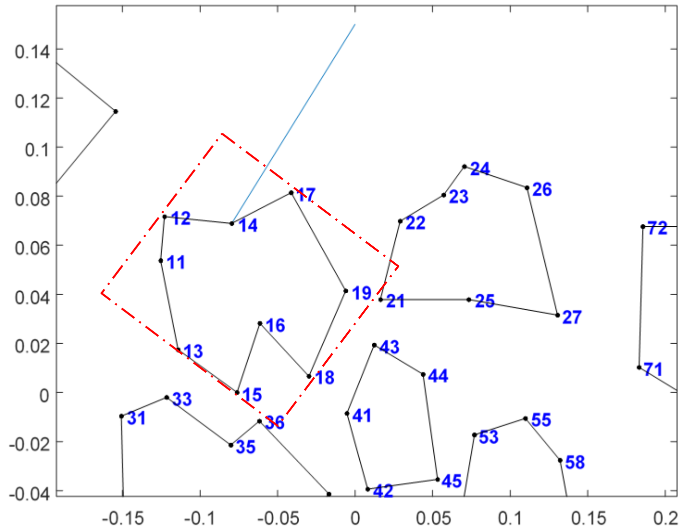
- Bi-level intersection detector:

1. Filtering out the out-of-bound obstacles
2. Checking the intersection between line segments for in-bound obstacles

nmasoud@clemson.edu
http://www.clemson.edu/ces/cedar

Out-of-bound Example





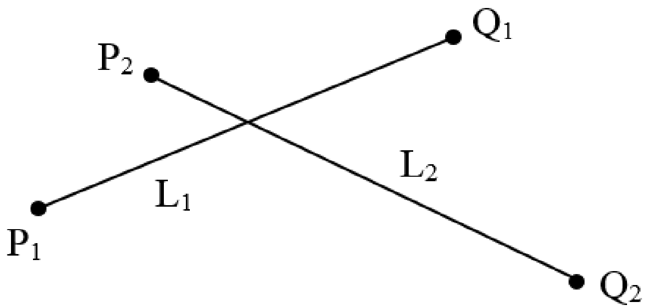
nmasoud@clemson.edu
http://www.clemson.edu/ces/cedar

Line segment intersection detection

- Determining intersection point of the two line segments

$$L_2 = (1-\mu)P_2 + \mu Q_2$$

$$L_1 = (1-\lambda)P_1 + \lambda Q_1$$



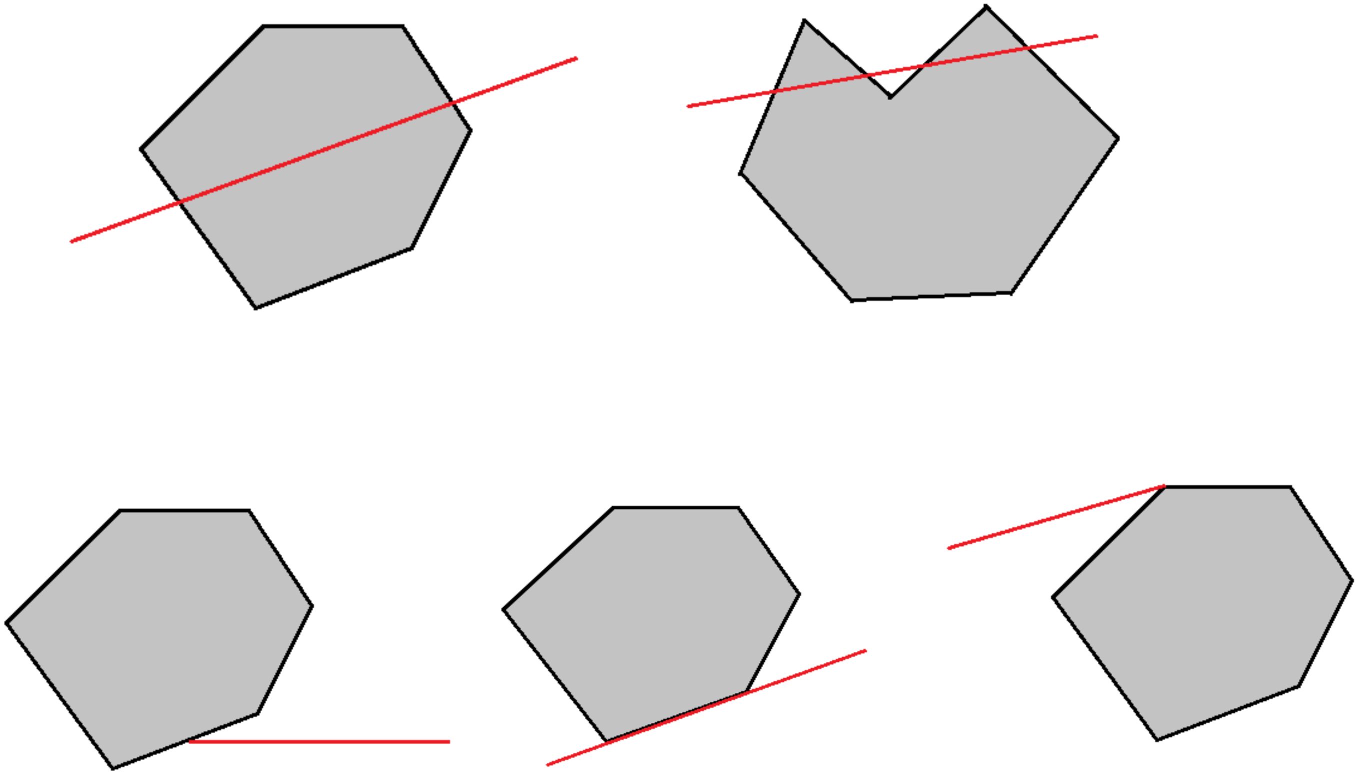
$$(1-\lambda)X_{P_1} + \lambda X_{Q_1} = (1-\mu)X_{P_2} + \mu X_{Q_2}$$

$$(1-\lambda)Y_{P_1} + \lambda Y_{Q_1} = (1-\mu)Y_{P_2} + \mu Y_{Q_2}$$

- Intersected if $0 \le \lambda, \mu \le 1$

nmasoud@clemson.edu
http://www.clemson.edu/ces/cedar

Intersection Examples



Finding the free space graph

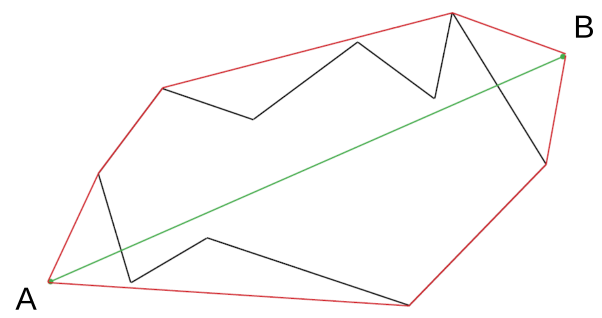
W is the workspace; W $\subseteq \mathbb{R}^2$

there are n polygonal obstacles in the workspace: P1,..,Pn

Known: geometry and location of all obstacles+ start and end points

Assume: obstacles are stationary and disjoint

$$C_{free} = W \setminus \bigcup_{i=1}^{n} P_i$$



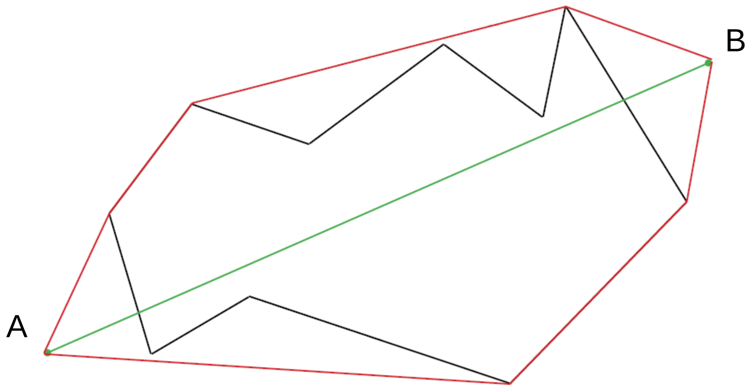
Find the graph G such that:

$$G = \{V,E\} \subseteq C_{free}$$

Graph Construction

- Assume there is one polygonal obstacle P in the workspace W: $P \subseteq W \subseteq \mathbb{R}^2$ and the start and end points are denoted by $A,B \in \mathbb{R}^2$, and if:

$\partial Conv(A,B) \cap P \subseteq \mathrm{Int(P)} \cup \partial(P)$



Where:

$Conv(r_1,..,r_n)$: convex hull

$\mathrm{Int(P)}$: interior of set P

$\partial(P)$: boundary of set P

- There is an intersection between line segment AB and the polygon P. otherwise:

$$Conv(A,B) \cap P \subseteq \partial(P)$$

Define $Conv(A,B,P)$ such that:

$$\partial Conv(A,B,P) \cap P \subseteq \partial(P)$$

nmasoud@clemson.edu
http://www.clemson.edu/ces/cedar

Graph Definition

- G = {V,E} Graph of the free space
- V: set of all nodes of the free space graph

$$v_i \epsilon V \Leftrightarrow v_i \epsilon\ \partial\ Conv(A, B, P), \qquad \exists\ v_{i,i+1}\ or\ v_{i-1,i} \epsilon\ E$$

- E : set of all segments/edges of the free space graph

$$e_{ij} \epsilon E \Leftrightarrow e_{ij} \subseteq \partial\ Conv(A, B, P), \qquad e_{ij} \cap P \subseteq \partial(P)$$

nmasoud@clemson.edu
http://www.clemson.edu/ces/cedar

Optimization Problem

- Given graph G={V,E}, find the shortest path between nodes i,j, i≠j;
- Problem formulation:

$$\min \sum_{(i,j)\in G} C_{ij} X_{ij}$$

Where:

C_{ij}: cost, the L2 norm (Euclidean) of arc e_{ij}

$$X_{ij} = \begin{cases} 1 & if\, e_{ij}\ is\ in\ the\ path \\ 0 & otherwise \end{cases}$$

s.t.

$$\sum_{\{j:(i,j)\in G\}} X_{ij} - \sum_{\{i:(i,j)\in G\}} X_{ji} = \begin{cases} 1 & i = 1 \\ 0 & i \neq 1, m \\ -1 & i = m \end{cases}$$

nmasoud@clemson.edu
http://www.clemson.edu/ces/cedar

Dijkstra's Algorithm

- Originated from Dynamic Programming (DP); solving an optimization problem by breaking it into multiple sub-problems
- Starting from s, in a given graph, at each step i, the node with the minimum distance from node among all adjacent nodes is added to the path until it reaches t (explain step by step)

nmasoud@clemson.edu
http://www.clemson.edu/ces/cedar

Dijkstra

Dijkstra's Algorithm (Sneidovich,2006)

- Initialization:

$j = 1; F(1) = 0; F(i) = \infty, i \in \{2 . . . , n\}; U = C= \{1, . . . , n\};$

j :current node, F(j): the objective value assigned to node j.

U: set of unvisited nodes

- Iteration:

While $(j \neq n$ and $F(j) < \infty)$ Do:

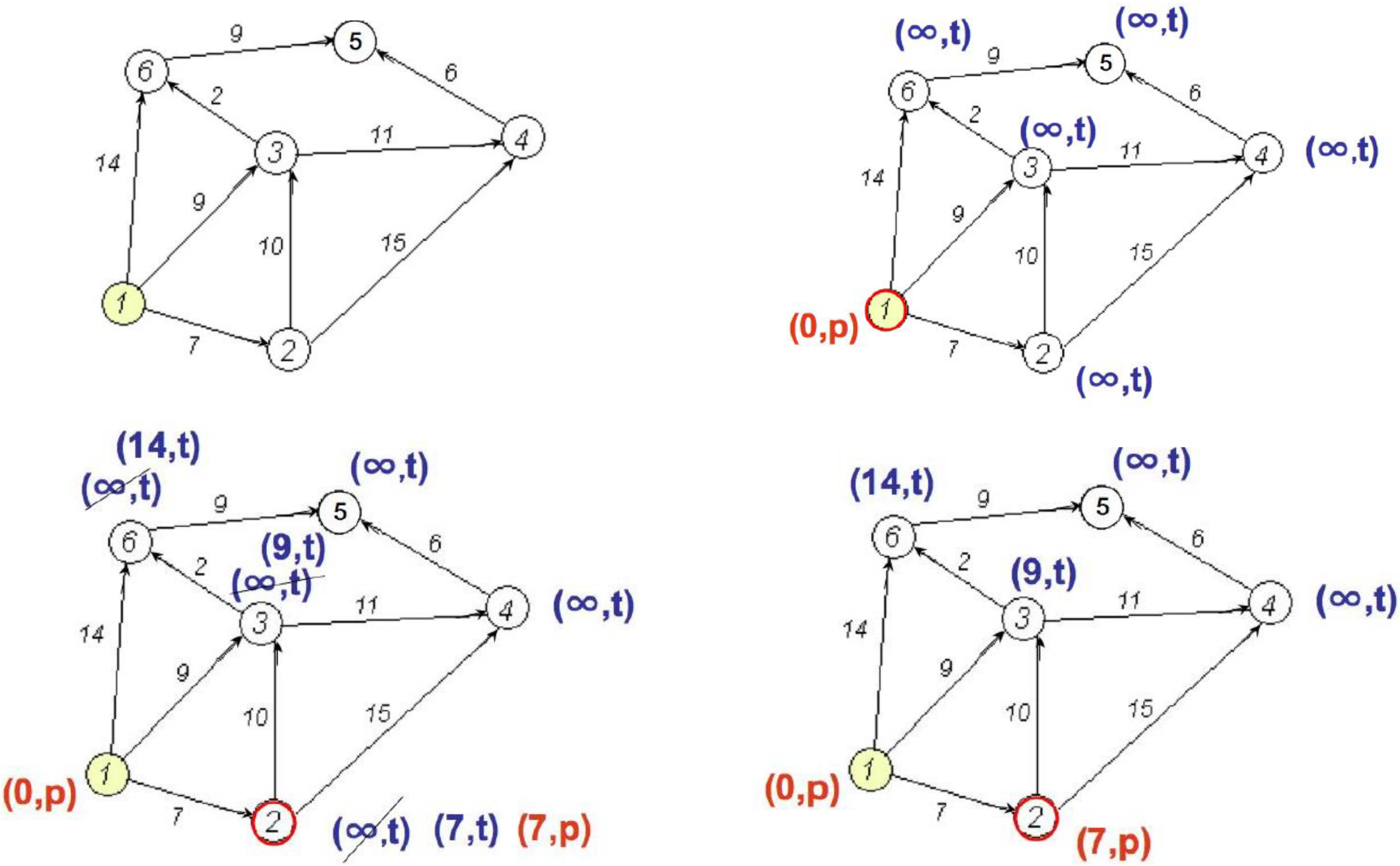
Update U : $U = U \backslash \{j\}$

Update F : $F(i) = \min\{F(i), F(j) + D(j, i)\}, i \in A(j) \cap U$

Update j : $j = \arg\min\{F(i) : i \in U\}$

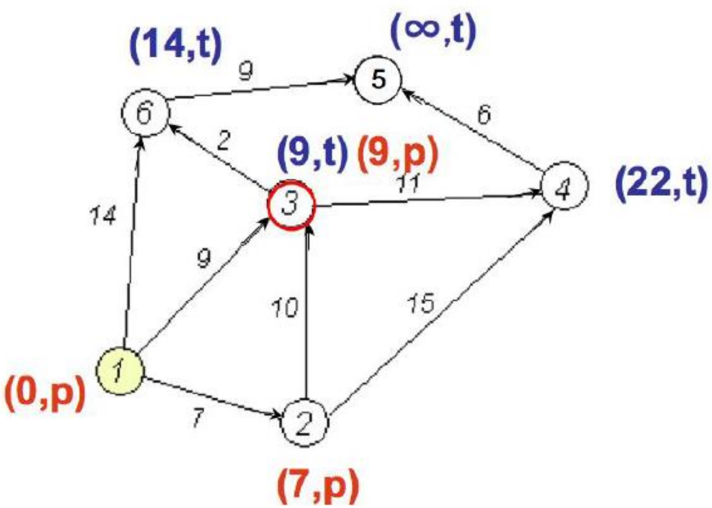
Where; A(j) denotes the set of node j's immediate successors

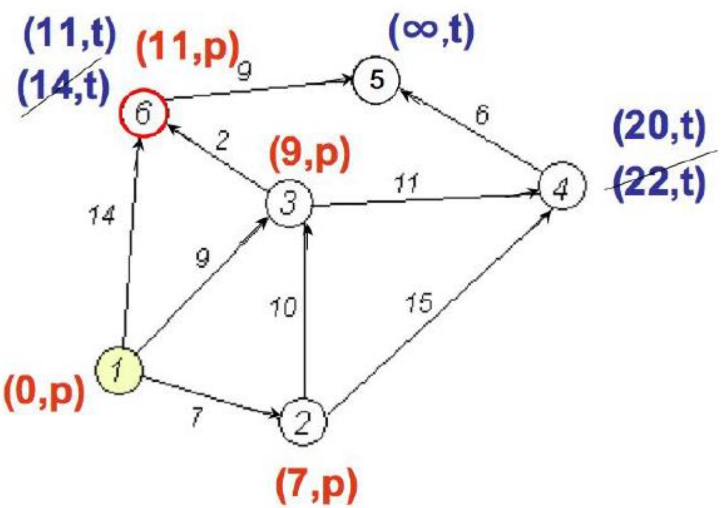
Dijkstra Example

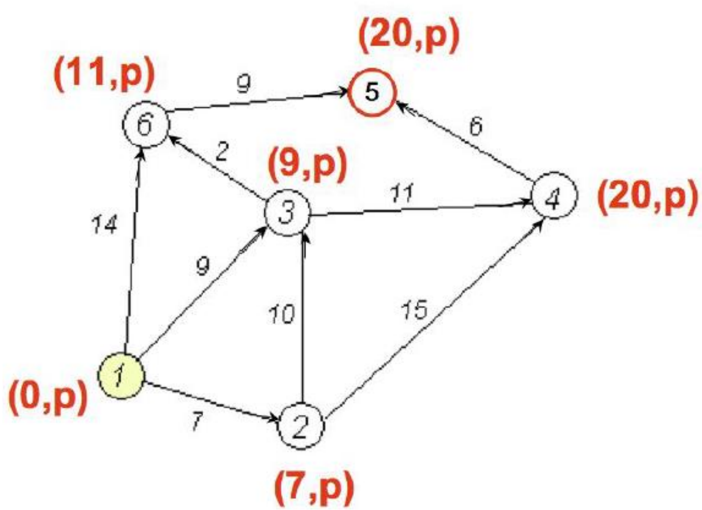


nmasoud@clemson.edu
http://www.clemson.edu/ces/cedar

Example Cont'd







Shortest Path ={1,3,6,5}
Path Length = 20

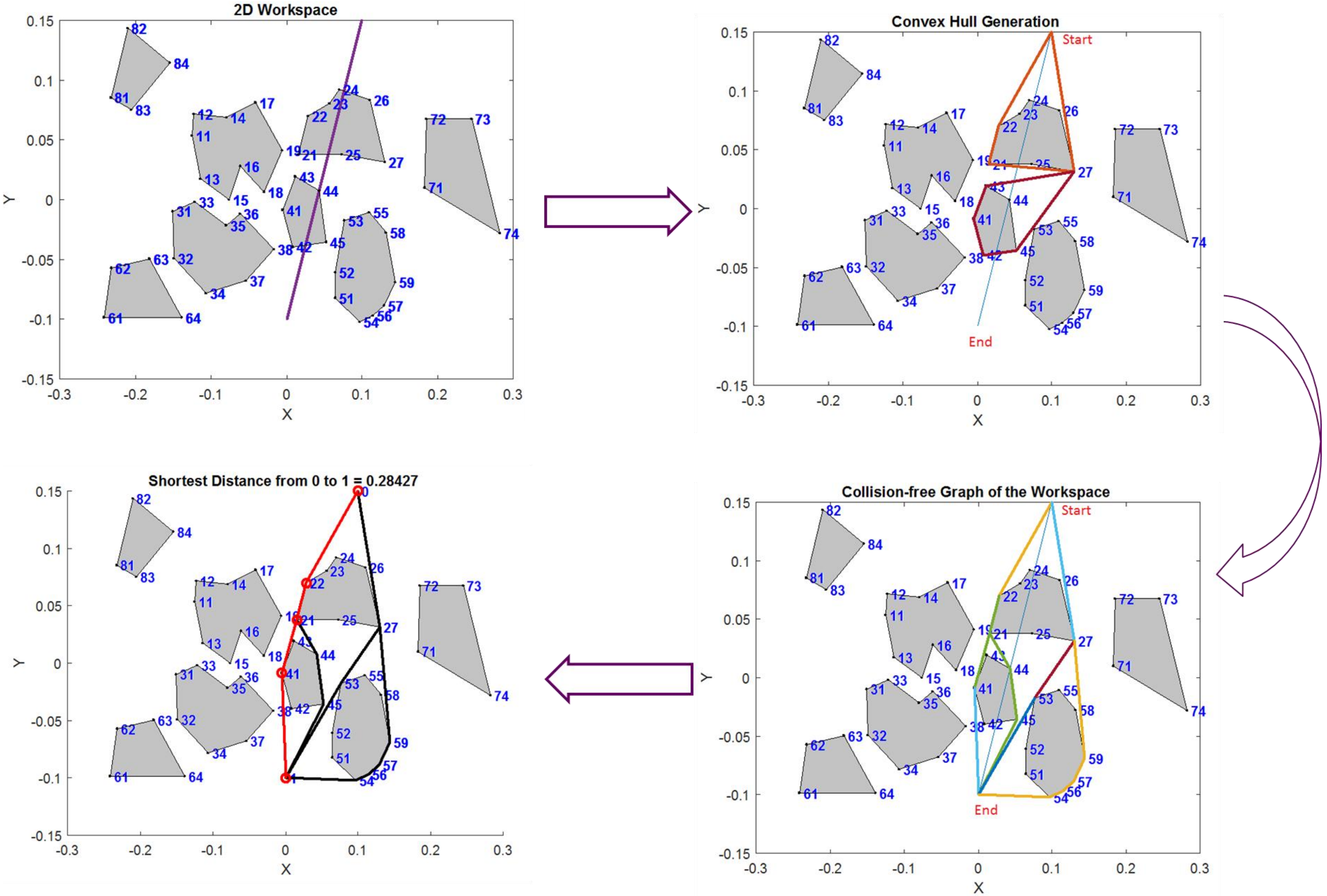
A* Search Algorithm

- Similar to Dijkstra in finding the shortest path based on DP
- But, it also has a heuristic term which helps favoring vertices that are close to the goal
- Cost: $f(n) = g(n)+h(n)$

- Definition of the heuristic term is challenging.

nmasoud@clemson.edu
http://www.clemson.edu/ces/cedar

Results



CEDAR
Clemson Engineering Design Applications and Research

nmasoud@clemson.edu
http://www.clemson.edu/ces/cedar

CLEMSON UNIVERSITY

Results

- The algorithm is tested on different cases with varying number of objects, start and end points and various shaped objects(convex and non convex)
- The algorithm depends on the number of colliding obstacles while being independent of the total number of obstacles
- The algorithm is independent of the shape of the objects

nmasoud@clemson.edu
http://www.clemson.edu/ces/cedar

Conclusions

- A two-level collision detection algorithm is developed that checks for intersections
- Instead of generating the entire visibility graph of the workspace, we find a portion of it using convex hulls of the intersecting objects
- A network optimization algorithm, Dijkstra, is implemented to find the global optimal solution of the SP problem (if one exists)

Future Work

- A sensitivity analysis will be deployed to analyze the changes of the path with respect to small changes in the workspace configuration
- The algorithm will be modified as needed and implemented on a 3D environment
- An optimizer will be developed to optimize the path of harnesses for automation of harness assembly process in manufacturing lines

References

[1] Zhu, D., Latombe, J., and Science, C., 1991, "Pipe Routing = P a t h Planning (with Many Constraints)," (April), pp. 1940–1947.

[2] Conru, A. B., and Cutkosky, M. R., 1993, "Computational Support for Interactive Cable Harness Routing and Design," Adv. Des. Autom., 1, pp. 551–558.

[3] Yin, Y. H., Zhou, C., and Zhu, J. Y., 2010, "A pipe route design methodology by imitating human imaginal thinking," CIRP Ann. - Manuf. Technol., 59(1), pp. 167–170.

[4] Sandurkar, S., and Chen, W., 1999, "GAPRUS - genetic algorithms based pipe routing using tessellated objects," Comput. Ind., 38(3), pp. 209–223.

[5] Szykman, S., and Cagan, J., 1996, "Synthesis of optimal nonorthogonal routes," J. Mech. Des., 118(3), pp. 419–424.

[6] Conru, A. B., 1994, "A genetic approach to the cable harness routing problem," Ieee, pp. 200–205.

[7] Lin, C. W., Rao, L., Giusto, P., D'Ambrosio, J., and Sangiovanni-Vincentelli, A. L., 2015, "Efficient Wire Routing and Wire Sizing for Weight Minimization of Automotive Systems," IEEE Trans. Comput. Des. Integr. Circuits Syst., 34(11), pp. 1730–1741.

[8] Lin, C., Rao, L., Ambrosio, J. D., and Sangiovanni-vincentelli, A., 2014, "Electrical Architecture Optimization and Selection - Cost Minimization via Wire Routing and Wire Sizing," pp. 502–509.

[9] Automatic, E., Electrical, A., and Interconnection, W., 1998, "S. A. F A W A Z Faculty of Aerospace Engineering, Delft University of Technology, Kluyverweg 1, 2629 HS Delft, The Netherlands," 59(3), pp. 327–342.

[10] Nath, Pooja "Motion Planning in Robotics"

[11] http://www.24auto.ro/newsroom/O_noua_fabrica_Delphi_in_Romania.html

Thank You

Questions ?

nmasoud@clemson.edu
http://www.clemson.edu/ces/cedar