

# Geometrically-Constrained Pathfinding: An Arc-Based Algorithm for Navigation in Restricted Domains

Michal Krupa  
Independent Researcher  
USA

## ABSTRACT

This paper introduces an arc-based pathfinding algorithm designed for navigating structured networks formed by intersecting circles, such as biological systems, fiber routing, or mechanical movement constrained to rails. The algorithm leverages geometric relationships and localized search to efficiently compute approximate shortest paths that respect curvature and structural boundaries. Applications include neuron tracing, microfluidic path optimization, vascular modeling, and layout routing for tightly curved circuit or fiber systems.

## KEYWORDS

pathfinding, geometric constraints, KD-tree, circular intersections, robotic navigation, fiber routing

## ACM Reference Format:

Michal Krupa. 2025. Geometrically-Constrained Pathfinding: An Arc-Based Algorithm for Navigation in Restricted Domains. In . ACM, New York, NY, USA, 1 page. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

## 1 INTRODUCTION

Pathfinding in constrained environments arises in numerous domains, including robotics, biological modeling, and visual storytelling. Classical algorithms perform poorly when paths must conform to geometric constraints such as arcs or structural layouts. This paper introduces an arc-based method that follows the inherent geometric limitations of the domain.

## 2 ALGORITHM OVERVIEW

Given a set of spatial endpoints  $W = \{w_1, \dots, w_n\}$ , circles are formed between each point and its  $k$ -nearest neighbors. These define intersections  $P$  used to form an arc-based graph  $G$ . Each arc's length is computed as:

$$d_{\text{arc}}(p_a, p_b; c) = r \cdot \min(|\theta_a - \theta_b|, 2\pi - |\theta_a - \theta_b|)$$

with  $r = \|p_a - c\|$  and angles  $\theta$  measured from the circle center  $c$ .

## 3 EXPERIMENTAL RESULTS

We compare the arc-based algorithm with a KD-tree-optimized version of Dijkstra's algorithm on a 10,000-point dataset.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

Conference'17, July 2017, Washington, DC, USA

© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM

<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

Table 1: Runtime comparison and qualitative performance

Algorithm	Runtime (s)	Curvature-Aware	Visual Quality
KDTree Dijkstra	0.48	No	Low
Arc Pathfinder	2.51	Yes	High

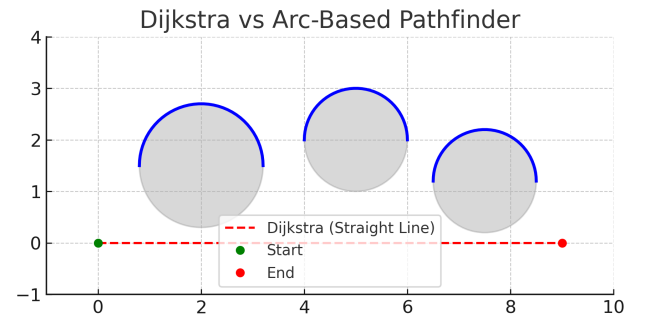


Figure 1: Comparison of direct vs. arc-based pathfinding in the presence of obstacles.

## 4 CODE AVAILABILITY

The full implementation of the algorithm is available as open-source code at: <https://github.com/michalkrupa/pathfinding>

## 5 CONCLUSION

The arc-based pathfinder offers interpretability and structural realism for geometric domains. Future work includes hybridization with linear navigation and deployment in hardware-constrained path systems.

## ACKNOWLEDGMENTS

We thank the contributors to open-source geometry libraries and acknowledge the support of interdisciplinary visualization research.

## REFERENCES

- [1] J. A. Reeds and L. A. Shepp. 1990. Optimal paths for a car that goes both forwards and backwards. *Pacific J. Math.* 145, 2 (1990), 367–393.