Smooth Nearness Diagram Navigation

IROS
Nice, France, September 22-26, 2008

Joseph W. Durham

Mechanical Engineering
Center for Control, Dynamical Systems and Computation

University of California at Santa Barbara
http://motion.mee.ucsb.edu

Ack: Francesco Bullo, NSF

UCSB

Motivation & Approach

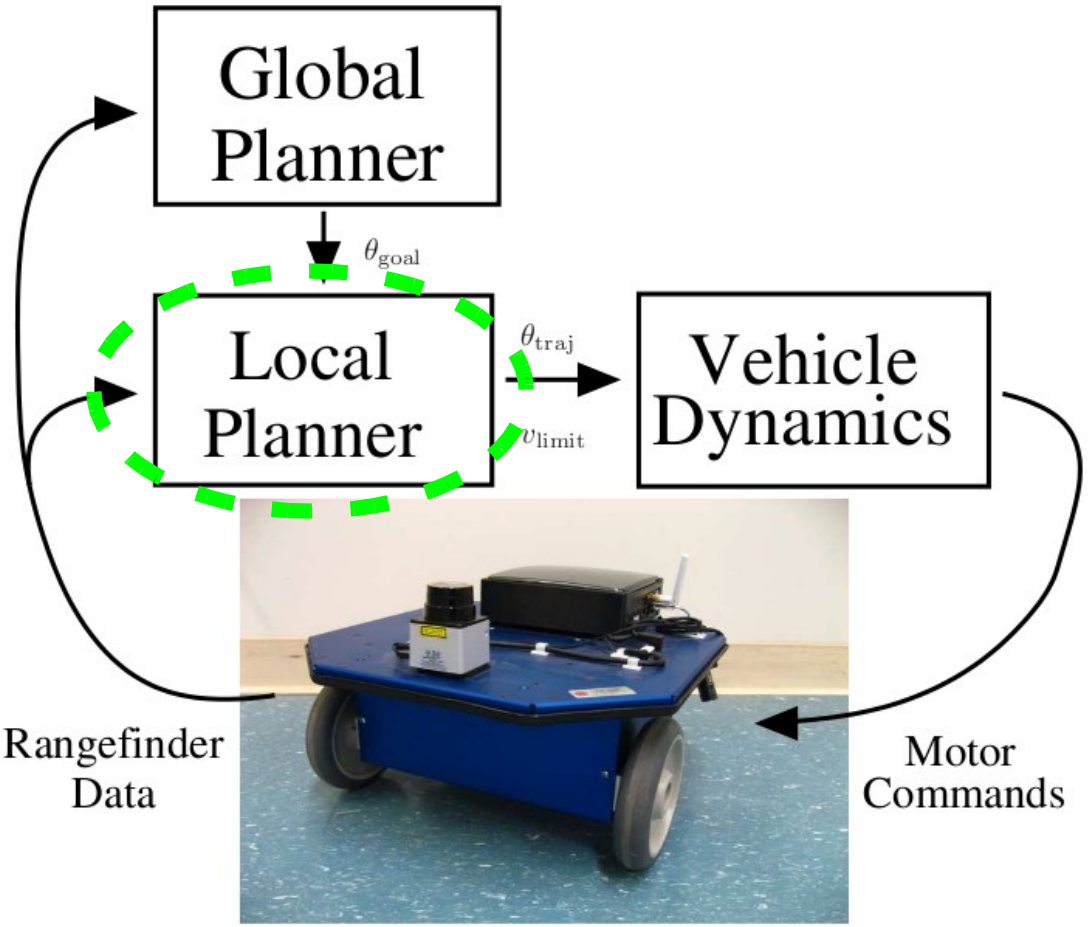
- Motivation
  - Smooth navigation through cluttered, potentially dynamic environments
- Approach
  - Built on Nearness Diagram+ method
  - “Gap”-based

• J. Minguez, J. Osuna, and L. Montano, “A 'divide and conquer' strategy based on situations to achieve reactive collision avoidance in troublesome scenarios,” ICRA, 2004

• J. Minguez and L. Montano, “Nearness diagram (ND) navigation”, *IEEE Transactions on Robotics and Automation*, vol. 20, no. 1, pp. 45–59, 2004.

Motion Control Framework

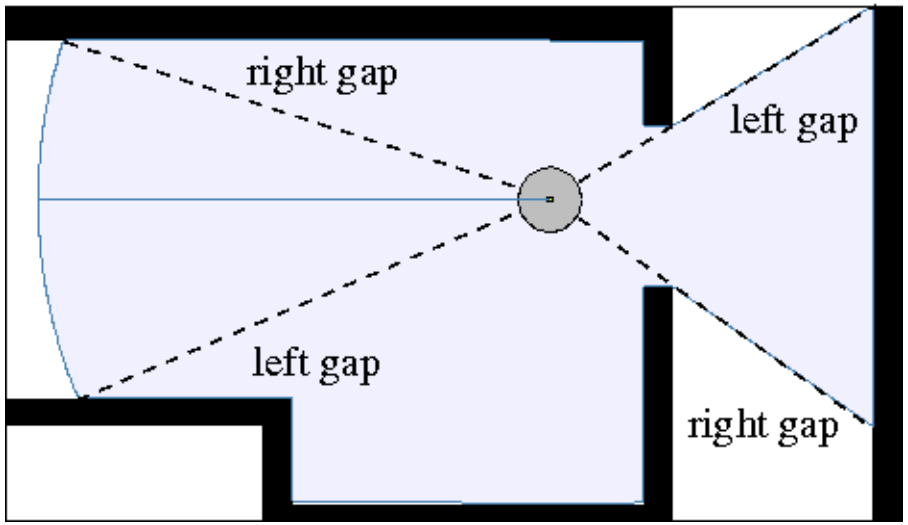
- SND Navigation
  - Local, reactive planner
  - Inputs:
    - Rangefinder data
    - Goal direction
  - Outputs
    - Safe direction
    - Linear speed



Brief Comparison

- ND+
  - Gap and valley based
  - 6 motion laws
  - Avoidance to closest left and right obstacle point
- SND
  - Gap and valley based
  - 1 motion law
  - Weighted avoidance of all obstacle points

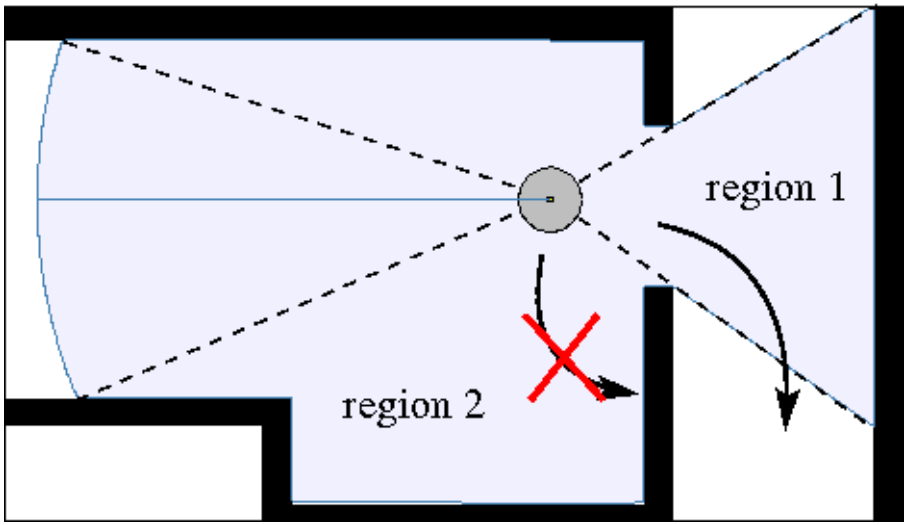
Gaps and Regions



- Gap: Discontinuity in measured distance or obstacle next to max range measurement
  - Also classified as right or left
- Region: Between two consecutive gaps

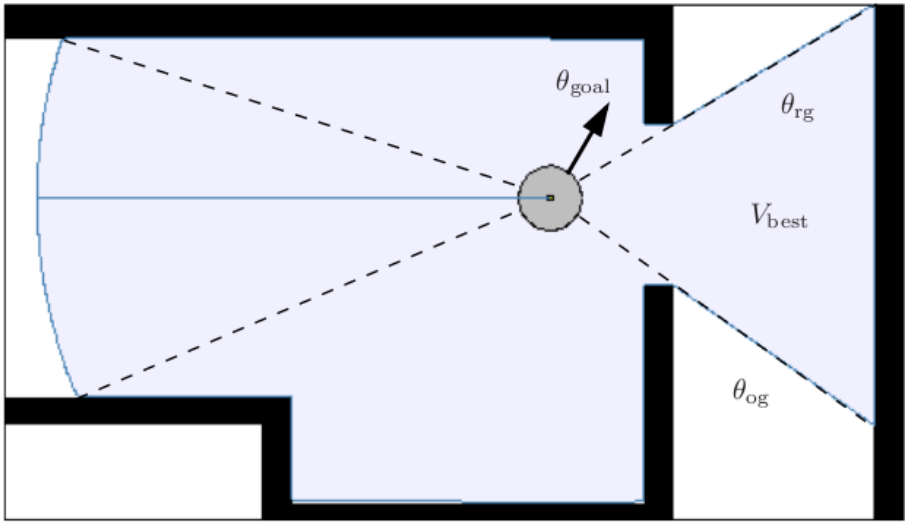
• B. Tovar, L. Guilamo, and S. M. LaValle, "Gap navigation trees", *Algorithmic Foundations of Robotics VI,* vol. 17 of *Springer Tracts in Advanced Robotics*, 2005

Rising Gaps



- Rising gap: Right gap on right side of region, left gap on left side of region
  - Region dependent
  - Indicate a potential path into area robot cannot currently see

Best Valley



- Valley: A region with at least one rising gap
  - If region has two rising gaps, closest to θ_{goal} is used
  - Only consider valleys wide enough for robot
- Best valley (V_{best}): Valley closest to θ_{goal}

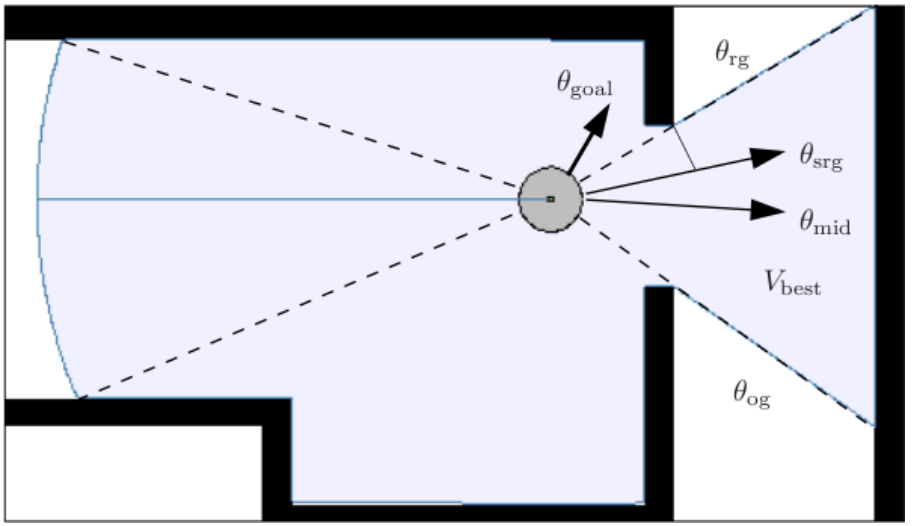
Headings from Best Valley

- Safe rising gap, θ_{srg}
  - Deflected around obstacle creating gap

$$\theta_{srg} = \theta_{rg} \pm \sin\left(\frac{R + D_s}{D_{rg}}\right)$$

- Valley bisector, θ_{mid}

$$\theta_{mid} = \theta_{rg} \pm \frac{\text{dist}(\theta_{rg}, \theta_{og})}{2}$$



R = robot radius
D_s = safety buffer around robot
D_{rg} = distance to obstacle at rising gap

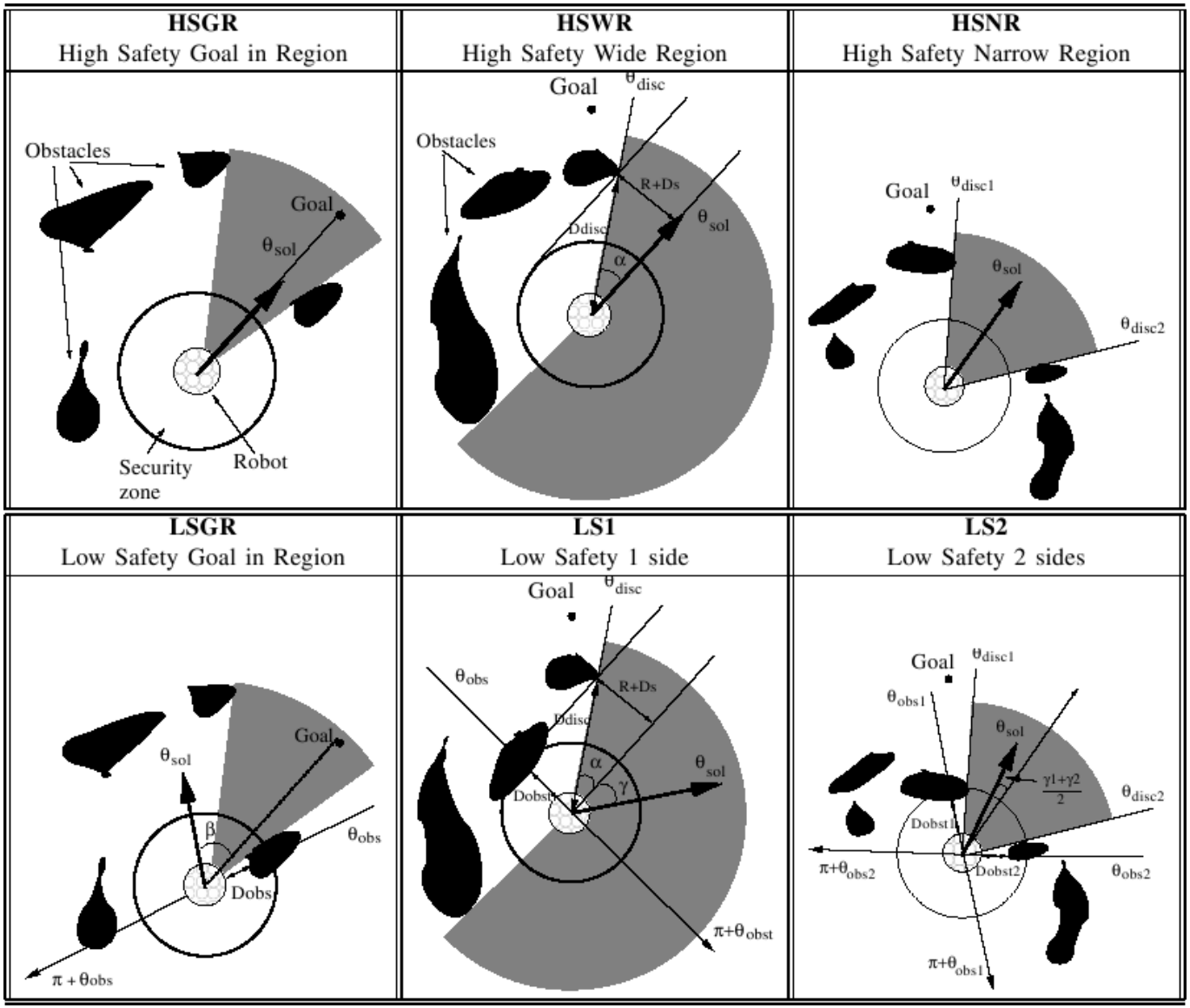
ND+ Method

- "Divide and conquer" strategy
  - 4 binary conditions defining 6 situations
- Robot behavior defined for each situation
- Smooth transitions between some pairs of behaviors

• J. Minguez, J. Osuna, and L. Montano, "A 'divide and conquer' strategy based on situations to achieve reactive collision avoidance in troublesome scenarios," ICRA, 2004

• J. Minguez and L. Montano, "Nearness diagram (ND) navigation", *IEEE Transactions on Robotics and Automation*, vol. 20, no. 1, pp. 45–59, 2004.

ND+ Figure

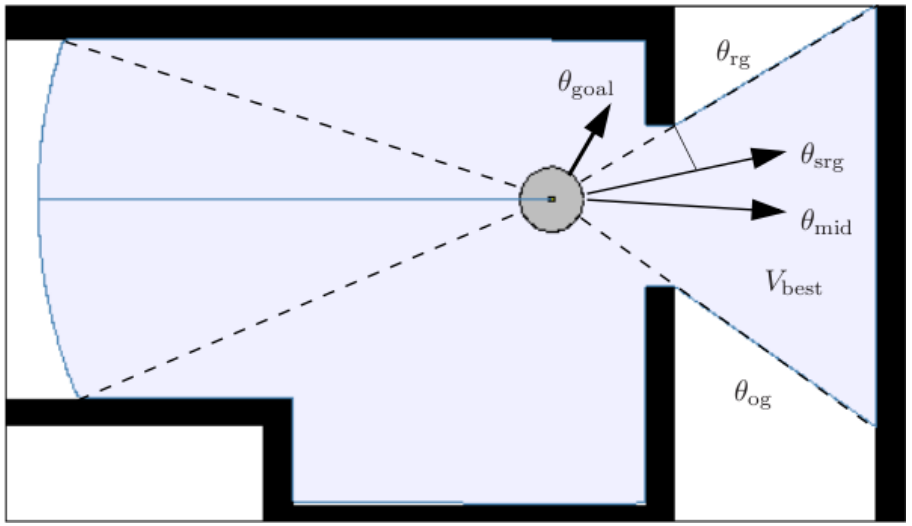


From J. Minguez and L. Montano, 2004.

SND Desired Heading

- Three choices for desired heading, θ_{des}:

$$\theta_{des} = \begin{cases} \theta_{goal} & if\ \theta_{goal} \in V_{best} \\ \theta_{srg} & elif\ \text{dist}(\theta_{srg}, \theta_{rg}) < \text{dist}(\theta_{mid}, \theta_{rg}) \\ \theta_{mid} & else \end{cases}$$



SND Obstacle Avoidance

- For each obstacle point
  - Threat measure:

$$t_i = \mathrm{sat}_{[0,1]}\left(\frac{D_s + R - D_i}{D_s}\right)$$

  - Deflection angle:

$$\delta_i = t_i \cdot \mathrm{dist}(\theta_i + \pi, \theta_{des}) \quad \in [-\pi, \pi[$$

  - If robot is touching ith obstacle, $\theta_{des} + \delta_i$ points directly away from obstacle

SND Obstacle Avoidance II

- Net threat measure:

$$T_{total} = \sum_{i=1}^{N} t_i^2$$

- Net deflection angle:

$$\Delta_{avoid} = \sum_{i=1}^{N} \frac{t_i^2}{T_{total}} \delta_i$$

- Final trajectory:

$$\theta_{traj} = \theta_{des} + \Delta_{avoid}$$

$$v_{limit} = \left(1 - max\left(t_i \ldots t_N\right)\right) \cdot v_{max}$$

Smoothness Conjecture

- For a rangefinder with infinitesimal resolution:

$$T_{total}(x, y) = \oint t(\alpha, x, y)^2 \, d\alpha$$

- Reminiscent of formula for area of visibility space:

$$A_{visible}(x, y) = \oint r(\alpha, x, y)^2 \, d\alpha$$

- Visibility area is Locally Lipschitz continuous in non-convex polygonal environment with holes

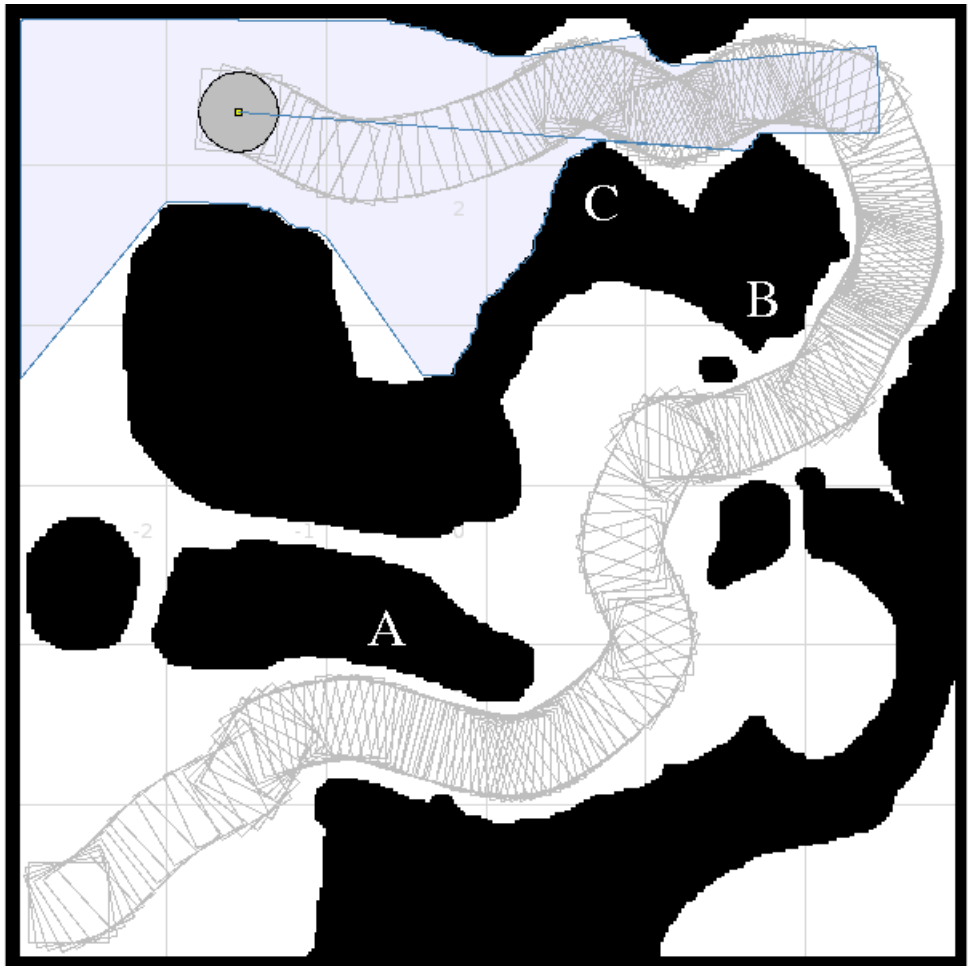
- A.Ganguli, J.Cortes, and F.Bullo, "Maximizing visibility in non-convex polygons," SIAM Journalon Control and Optimization, 2006

Testing Setup

- Player/Stage Robotics Software
  - Open-source tools for robotics
  - Easy portability from simulation to hardware
  - Implemented both ND+ and SND
  - Version 2.0.3
- Videre Designs “Erratic” mobile robot platform
- Hokuyo URG laser rangefinder

Simulations

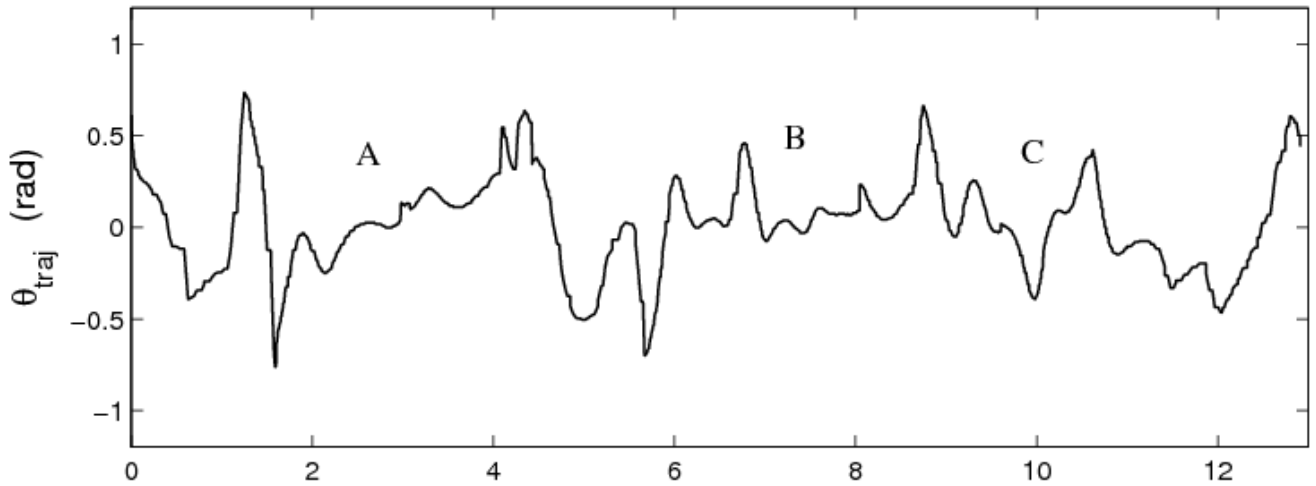
SND



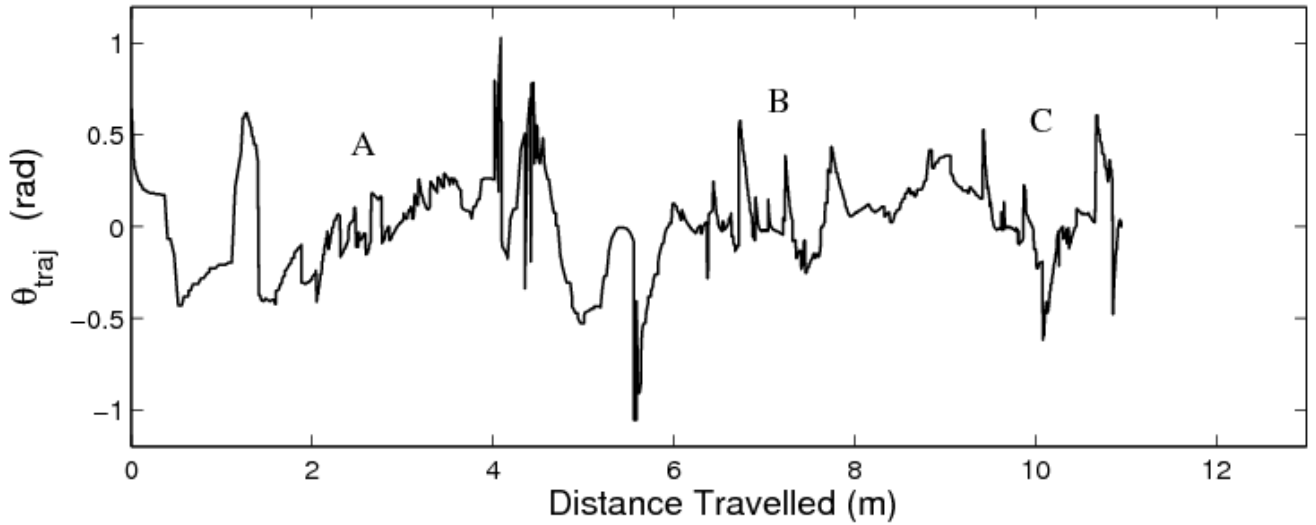
ND+

Simulation Trajectories

SND

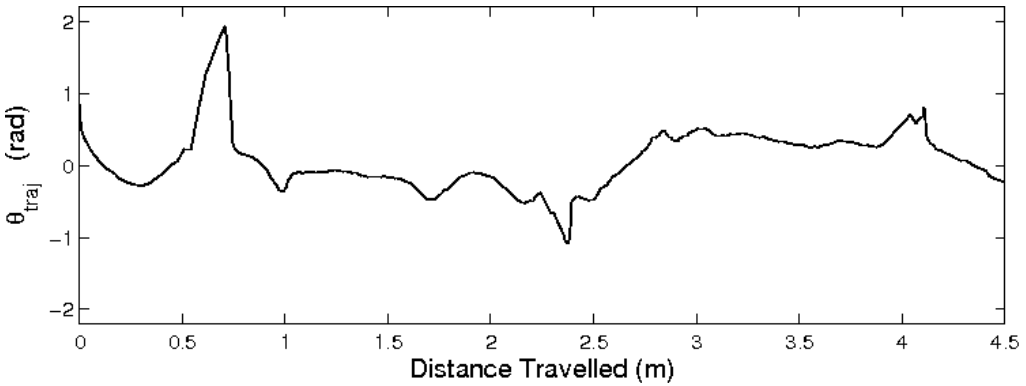


ND+

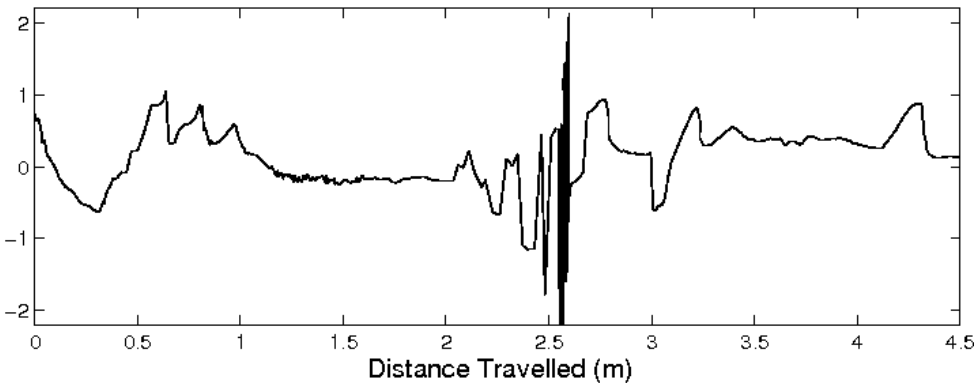


Experiments

SND



ND+



Summary

- Smooth Nearness Diagram Navigation
  - Adapted from ND+ method
  - Based on gaps
  - Single motion law for all situations
  - Improved smoothness in angular heading
  - Single parameter: size of safety buffer
- Future directions
  - Explore proofs of smoothness
  - When is SND guaranteed to find a safe path?

Thank you

Questions?