AutoVoyage: Autonomous path-planning, path-generation, and path-following for autonomous ships in transit

Aksel Knudsen Nordstoga, akselkn@stud.ntnu.no

NTNU
Norwegian University of Science and Technology

Supervisor: Roger Skjetne, co-advisors: Jon Bjørnø, Henrik Schmidt-Didlaukies and Einar

Introduction

In order for an autonomous ship to be able to operate safely and reliably, it has to sense and interpret its surroundings, and use that information to plan a path and speed and execute the plan accordingly. The system must also be able to perform obstacle avoidance and anti-collision when encountering unexpected obstacles (e.g. other ships), that is, re-plan the path whilst keeping the final destination in mind. This motivates the development of an intelligent guidance system that is able to execute a ship voyage from A to B. Such a system needs an efficient partitioning of the operation area, a path-planner that is able to plan a safe and efficient route, and it has to be able to generate and follow paths that are feasible for the vessel.

Problem definition

The path-planning problem is divided into two parts, a global and a local path-planner. The objective of the global path-planner is to use information about the environment that is known prior to departure, to plan a coarse route from A to B. The local path-planner aims to move the vessel along the path defined by the global path-planner, whilst handling reactive obstacle avoidance.

Since the system is online and the paths are generated on-the-go, a recursive path-generation algorithm is needed. The generated paths need to connect the waypoints given by the local path-planner in a smooth manner, to achieve feasible paths and avoid jumps in the signals in the control loops.

The resulting control problem is then to follow the path parametrized by the path-generation module.

Another important consideration is to keep the system computationally efficient, as is has to respond fast enough to a dynamic environment and respect the hardware limitations on-board vessels.

References

[1] Roger Skjetne: *Maneuvering control design of a low-speed fully-actuated vessel with stepwise path generation*, NTNU (2019)

[2] Roger Skjetne, Ulrik Jørgensen and Andrew R. Teel *Line-of-sight path-following along regularly parametrized curves solved as a generic maneuvering problem*, in Proc. IEEE Conf. Decision Control, IEEE, Orlando, USA, Dec. 2011.

[3] Simon X. Yang and Max Meng: *Neural network approaches to dynamic collision-free trajectory generation.* IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS-PART B:CYBERNETICS, VOL. 31. (2001)

Acknowledgements

I want to thank my supervisor Roger Skjetne, and my co-advisors Jon Bjørnø, Henrik Schmidt-Didlaukies and Einar Ueland. Special thanks to Jon Bjørnø for a lot of help at the MCLab.

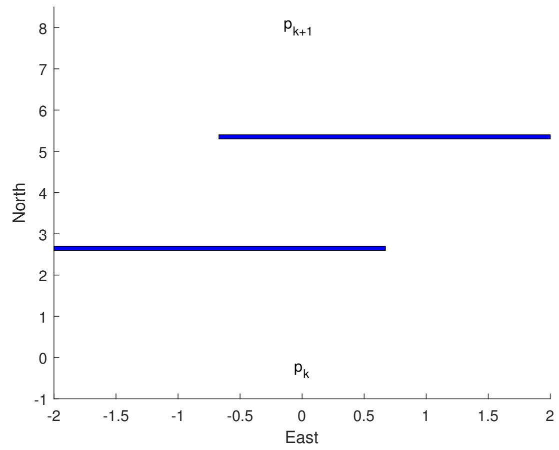
Method

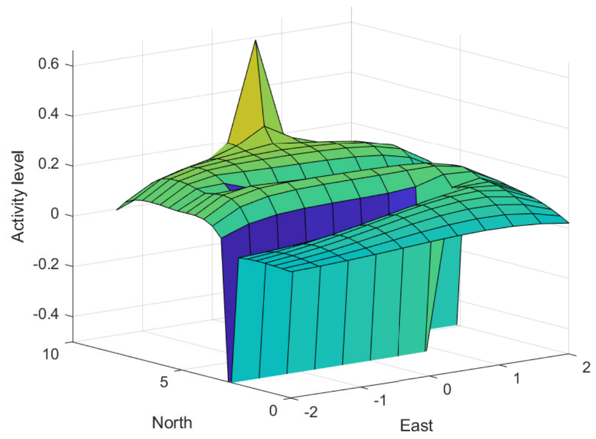
The operation area is first partitioned using Voronoi diagrams. This produces a raw obstacle-free roadmap of paths that are of maximum distance to the obstacles in the environment. A* search is then used to find in terms of waypoints the shortest route that also respects the specified clearance constraints. Excess waypoints are then removed from the path to reduce heading changes and path length.

The local high-resolution path-planner is implemented based on the neural network (BINN) approach first proposed in [3] for use in robot trajectory planning. A local reference frame is defined between each two consecutive waypoints p_k and p_{k+1} given by the global path-planner, with the x-axis pointing from p_k to p_{k+1}. A rectangular area aligned with this reference frame is then placed between p_k and p_{k+1} and partitioned into a set of rectangular nodes, using a fixed grid size. A topologically organized neural network architecture is then placed on the grid, where each node is associated with an activity level. The dynamics of the ith neuron in the grid, where the excitatory and inhibitory inputs $[I]^+$ and $[I]^-$ arise from target and obstacle lateral connections, is expressed as a shunting differential equation

$$\frac{dx_i}{dt} = -Ax_i + (B - x_i)\left([I_i]^+ + \sum_{j=1}^{k} w_{ij}[x_j]^+\right) - (D + x_i)[I_i]^- \tag{1}$$

This translates the environment information into a dynamic activity landscape, as shown below, where the obstacle course on the left produces the neural activity landscape on the right.



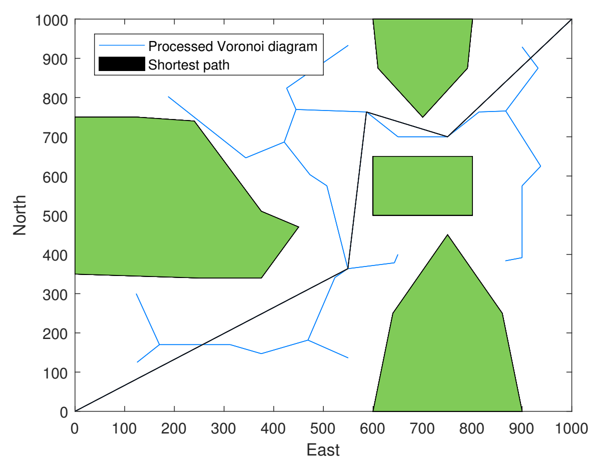


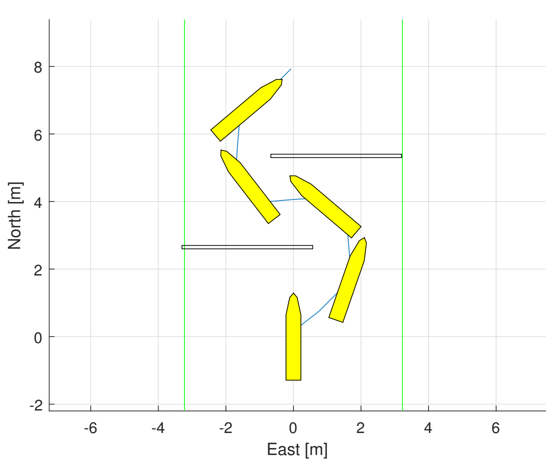
The optimal path from p_k to p_{k+1} is then found by following a steepest gradient ascent rule, until the peak of the activity landscape is reached.

Hybrid path parametrization as shown in [1] is used to ensure continuous path derivatives in connection points between the generated paths. This parametrization also allows for tuning of the path curvature, so as to achieve feasible paths. For the fully actuated case, a backstepping controller design adopted from [1] is implemented.

Simulation results

The figure below on the left shows the global path-planner using the Voronoi partitioning to find the shortest route from $(0, 0)$ to $(1000, 1000)$, that does not violate the clearance constraints, with unnecessary waypoints removed.





The guidance system is tested on a 1:90 scale model of C/S Inocean Cat I Drillship (CSAD). The figure on the right is a north-east plot from a simulation of CSAD navigating through the obstacle course shown in the previous section, which is of similar size to the basin at the marine cybernetics laboratory (MCLab). It is seen that the BINN-based local planner produces a path with sufficient clearance to the obstacles, and that the vessel is able to follow the desired parametrized path with the desired heading, which is tangent to the path.

Experimental testing of the system will be carried at MCLab.