

# Autonomous and optimized ship routing

**Ioannis Z. Emiris<sup>1</sup>**

1. ATHENA Research Center, and National & Kapodistrian University of Athens, Greece

*We present state-of-the-art computational methods which are instrumental in autonomous maritime operations, and optimization of routing, scheduling as well as loading. Our aim is to survey mature algorithmic approaches developed within the Lab of Geometric & Algebraic Algorithms, towards exploiting intelligence and automation in modern shipping and, in particular, in various aspects of routing. We showcase our advances in two main axes: (a) geometric computing for collision avoidance in complex environments, thus allowing for semi-autonomous and fully autonomous navigation, and (b) optimization for routing under time constraints of the carrier ship, time windows of availability at the ports of call, and capacity constraints of various compartments within a vessel.*

**KEY WORDS:** Navigation; automation; scheduling; collision avoidance; optimization; Operations Research; AI.

## INTRODUCTION

We present state-of-the-art algorithmic and computational methods that are relevant in shipping, autonomous maritime operations, and optimization of routing, scheduling and loading. Our aim is to illustrate mature algorithmic techniques that should have a major impact in the aforementioned areas and would help shipping enter the era of intelligent agents and AI.

These tools are being developed within the Lab of Geometric and Algebraic Algorithms:

<http://erga.di.uoa.gr/>

at University of Athens (NKUA) and ATHENA Research Center, and would have a major impact towards exploiting intelligence, Operations Research, and Machine Learning towards semi-automation and full automation in all aspects of ship routing and path planning. Our methods are typically developed on top of existing standard software platforms.

More precisely, we articulate our discussion around two main axes:

(a) Geometric computing for collision avoidance in complex environments, thus allowing for semi-autonomous and fully autonomous navigation, and

(b) Optimization for routing under time constraints of the carrier ship and time windows of availability at the ports of call

as well as capacity constraints of various compartments within a vessel.

Our combined tools may support the onboard ship operations providing semi-autonomous decision support to the captain with regards to optimal routing, collision avoidance, optimizing fuel consumption and minimizing operational and voyage costs.

In the next two sections we examine the two major themes outlined above, and offer several technological and algorithmic methods to corroborate our claims, including relevant software platforms. Each section concludes with the expected impact to the maritime industry. Existing work is referenced in each respective section. One work of general interest, that concerns both themes developed below, is on major challenges and solutions for utilizing big data in the maritime industry (Koga, 2015).

## AUTONOMOUS NAVIGATION

The goal of this section is to offer computational methods for intelligent agents that would assist human agents in navigation, supporting semi-autonomous decision-making and eventually offering even fully automated solutions in several cases.

Collision detection and avoidance is of primary importance for addressing the main cause of marine accidents, namely human errors, by providing decision support ashore as well as on-board. Our work is expected to have an impact on a number of stakeholders. For ship owner, operator, and management companies we develop tools for e-navigation near ports and confined coastal spaces by exchange of data in real-time or on-

demand. These will provide more efficient monitoring of position and performance during operation, thus enabling a more efficient scheduling of activities with minimal requirements in terms of condition surveys or the risk of facing detention or even conducting emergency and unscheduled sailing, maneuvering or anchoring operations.

Inspired by breakthroughs in Artificial Intelligence, robotics motion and path planning and, more recently, self-driving cars and unmanned flight (drones), we showcase mature computational tools that could be integrated in existing software platforms in shipping. We develop software in Matlab, Maple, Mathematica, or directly on C and C++ platforms of geometric computing, the leading one being CGAL (CGAL), the state-of-the-art Computational Geometry Algorithms Library. The latter is an open library for academic users and requires a fee for commercial users, depending on the modules that one would like to license. In CGAL, we have already co-authored the module on Voronoi diagrams of disks (see below for definitions).

In short, (semi) autonomous navigation falls in the umbrella of the literature on autonomous vehicles, which is very active today. The targeted navigation systems would support dense

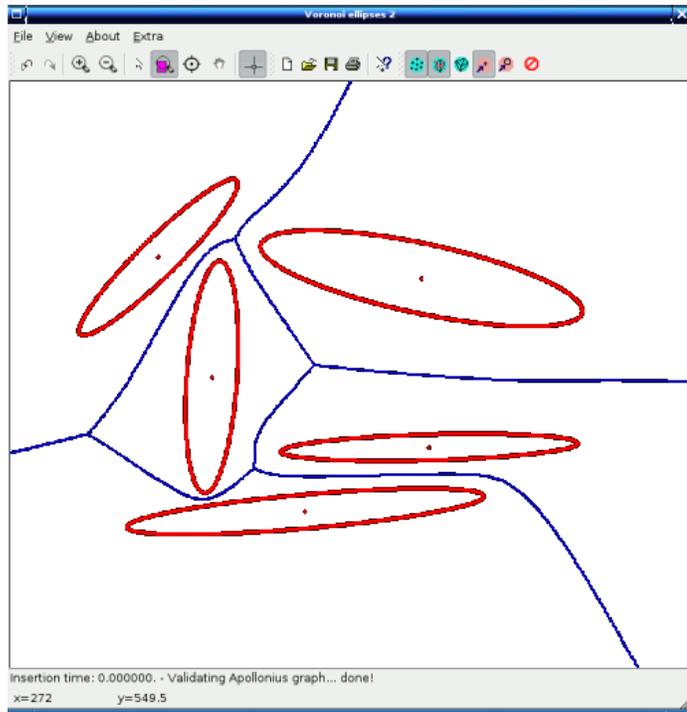


Fig. 1: Voronoi diagram of (red) ellipses

traffic areas, berthing operations, open seas navigation, tug operations, and manoeuvring using localization among complex environments, real-time collision detection and obstacle avoidance, see Geng et al. (2019).

Our Lab focuses on collision avoidance given two objects in the 2D plane; the objects are convex or non-convex (red ellipses in Figure 1). Our first specific contribution is the use of advanced proximity (Voronoi) diagrams. Voronoi (or proximity) diagrams partition the plane into areas, known as Voronoi regions, which are the geometric locus of all points closer to one specific obstacle than to any other obstacle. Hence, when a ship is located in such an area, it is immediately known which is its closest obstacle.

The boundary of Voronoi regions is composed of Voronoi edges, which are straight or curved line segments (blue in Fig. 1). The intersection of three Voronoi edges defines a Voronoi vertex; notice that there is no intersection of two Voronoi edges since this intersection would immediately belong to a third edge. As a result, if one wishes to define a course that maximized distance from the obstacles, one would choose to navigate as close as possible to the Voronoi diagram.

Our contribution concerns Voronoi diagrams that go beyond the standard diagrams of points and segments, in order to tackle disks and ellipses. Ellipses are very general shapes since they can model elongated objects or nearly circular ones. Our

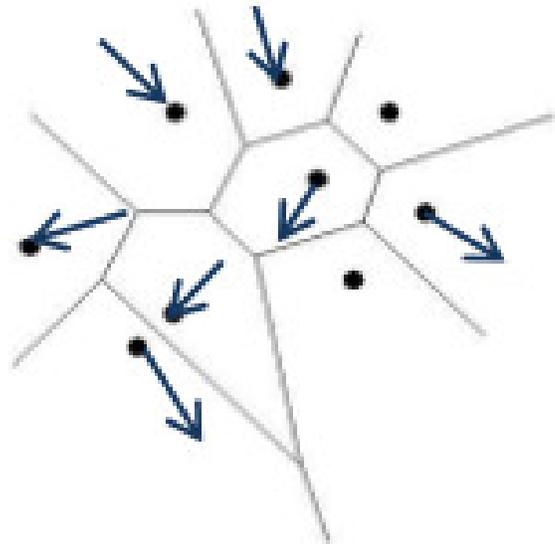


Fig. 2: Kinetic Voronoi diagram of 7 moving ships (motion given by arrows) and 2 stationary obstacles or ships

algorithms allow for overlapping ellipses. Any non-convex obstacle is modeled by a set of possibly overlapping ellipses.

In Figure 1, red ellipses model the obstacles or prohibited areas, each corresponding to a Voronoi area, while blue are the Voronoi edges that separate the Voronoi areas. For navigation among obstacles, it suffices to follow the Voronoi edges which separate the areas, so as to maximize distance from any obstacle. The Voronoi diagram gives a definite answer on whereas a path is feasible among obstacles, and what shall be the minimum distance to an obstacle during a given route.

Our work on the Voronoi diagram of ellipses was called a “success story” since it achieves a run-time of 1 sec for every new ellipse added to the diagram, as designed and developed by Emiris et al.. It has been implemented in C++ within the standard geometric computing platform CGAL (CGAL). The figure shows a screen-shot of our software on a small example with 5 obstacles.

In order to capture movement, it is possible to assign points in the diagram a velocity vector and consider a different kind of Voronoi diagram that is instrumental in collision avoidance. Now, “obstacles” are points that move in the plane so as to model the presence of several ships in a given region, and the diagram is subject to change at every time step. These are called kinetic Voronoi diagrams (see Fig. 2). A kVD extends classic Voronoi diagrams and partitions the area into regions consisting of all points closer to one ship than to any other, while adapting to the ships' movement (Russell et al.).

The work by Goralski et al. (2007) shows that the concept has been introduced to marine applications, but no system on the market actually uses it. In particular, the authors describe a type of GIS system for maritime navigation using the kinetic Voronoi diagram and a standard onboard “AIS transponder”.

We improve upon existing prototype software on “kinetic Voronoi diagrams” with information for collision avoidance, and more realistic velocity functions. functions for ship design, simulation and ship-building as well as powerful methods for collision detection and navigation.

The ensuing methods aim at monitoring and forecasting in time the spatial relationship of a ship with navigating objects in its operation area, based on the kinetic Voronoi diagram (kVD) for a set of ships in a planar area. Monitoring each ship's region locally provides all necessary information for deciding the ship's route in the overall configuration.

## Results and Industrial Impact

We adapt existing Voronoi methods to the ship navigation and examine problem arising in tight / crowded areas (ports, capes, straits) or high seas. We are able to deliver e-Navigation solutions and decision support for collision avoidance, berthing, tug operations, and manoeuvring.

The C++ tool (developed within the CGAL software environment) and based on kinetic Voronoi diagram, should be tested on real data. The tool may eventually provide e-Navigation assistance and decision support to ships online. To give one example, it may be combined with the routing and loading methods described in the next section.

Results expected from our system target safe navigation enhanced with collision detection and collision avoidance functions that can handle the aspect of real-time response. In short, we are interested in the risk assessment of autonomous or semi-autonomous navigation decisions, and avoiding or minimizing any such risk.

## OPTIMAL ROUTING

Today, the cost of fuel increases and environmental regulations get tougher, hence it becomes more important than ever to transport cargo in an efficient way, especially with regards to fuel consumption, but also with respect to time usage for obvious economic reasons. Fig. 3 shows a heatmap of global cargo travel in 2015.

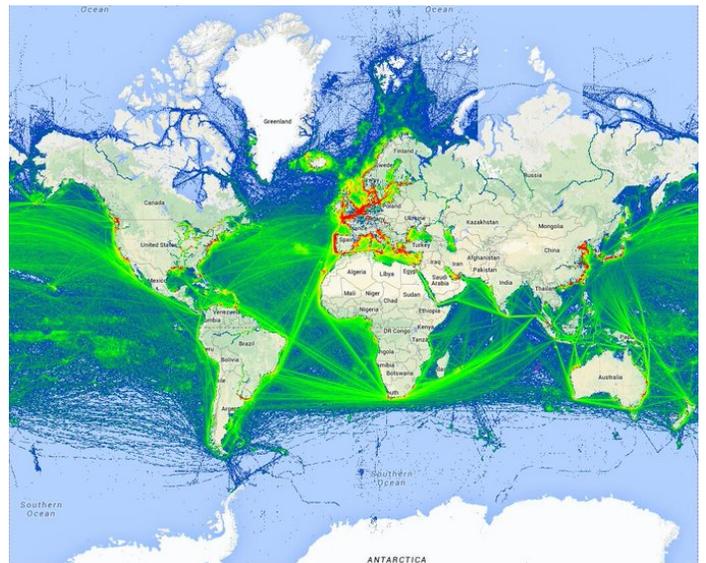


Fig. 3: Cargo travel heatmap (2015)

Besides the cost of cargo, the cost of shipping can be divided into three approximately equal groups: the cost of fuel, the cost of the ship maintenance and operation, and the cost of the crew employment. Most e-Navigation tools developed in the previous section are mainly targeted to the micro-scale of routing. Here, we focus on Business Intelligence solutions for improved operational performance of the ship and equipment at the macro-scale (see Fig. 4), aiming at significant gains in energy efficiency by means of lower fuel consumption and emissions.

The ultimate goal is to make sea transport safer, cheaper and more environmentally friendly, based on route optimization under a variety of optimization constraints, see (Avgouleas, 2008).

This section focuses on route optimization for loading scheduling, time constraints, and fuel consumption minimization. More specifically, this calls for optimization of routing under capacity constraints on the various compartments within a vessel as well as time constraints of the carrier ship and time windows of availability at the ports of call. This falls in the area of optimization and Operations Research, see (Le Carrer et al, 2020). In some sense, it generalized the famous Traveling Salesman Problem into many salesmen, each endowed with a capacity constraint.

Our Operations Research tools allow us to specify the desired function to be minimized, typically a conjunction of the concerns stated above with appropriate significance weights. Our goal is an autonomous navigation solution by means of our optimization algorithms. Such solvers integrate cost minimization but also safety constraints and regulatory requirements, which are very important in real life and can certainly affect the feasible solutions available.

Moreover, we can distinguish between hard and soft constraints, the former being required for every feasible solution considered. On the other hand, soft constraints are allowed to be violated with a certain penalty. Minimizing the aggregate penalty is also part of the objective function. The weights associated with different penalties is the object of an experimental determination.

Our approach is based on a powerful C++ platform today openly available by Google, namely OR-Tools, available at:

<https://developers.google.com/optimization/>

OR-Tools is open source software for *optimization*, which seeks to find the optimal or approximately optimal solution out of a very large set of possible solutions. It offers programming interfaces in C++, C#, and Python, which makes it very handy for connecting it with existing software libraries. This is the current industry standard for routing under capacity and time constraints, although they do not claim to hold the record in solving all problems listed below; yet, they offer the best compromise for industries in quest of a variety of tools and of addressing different optimization problems. Here are some examples of problems that OR-Tools solves:

- Routing for fleets that pick up and deliver cargo given volume and weight constraints as well as time windows.
- Scheduling for a complex set of tasks, some of which need to be performed before others, on a fixed set of resources.
- Packing as many goods of various sizes as possible into a fixed number of holds with maximum capacities.

In most cases, problems like these have a vast number of possible solutions: too many for a computer to search them all. We employ state-of-the-art algorithms and heuristics to narrow down the search set, in order to find an optimal (or close to optimal) solution.

A first approach is based on geometric clustering of the destinations, where a state-of-the-art algorithm has been developed by Avrithis et al., (2015). The latter is an acceleration

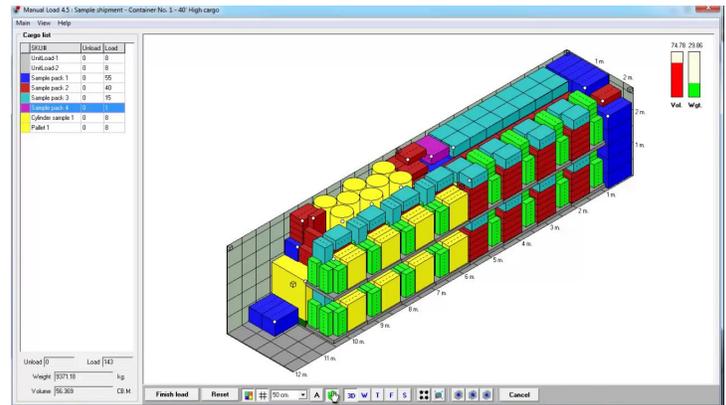


Fig. 4: Illustration of ship loading with respect to volume and weight optimization

of the celebrated k-means method, pioneered by Lloyd's. Our algorithm and open-source implementation (github) improve the runtime by introducing a small overhead on accuracy. Our method had been the fastest one since its publication and for at least a couple of years.

However, combinatorial optimization is key in this problem, since clustering only captures certain aspects only. To be precise, we leverage the solvers of OR-Tools in ways that adapt to the particular shipping problem. OR-Tools is a free package and offers no guarantee on execution or the functionalities supported. Our Lab has been contributing to this software library by adding the necessary enhancements for ships with several compartments dedicated to different types of cargo, safety constraints due to national or international regulations, time delays for loading/unloading and cleaning the holds, and possible delays at the port, hence adapting this generic platform to specific needs.

More formally, OR-tools allows us (a) design geographically optimal or nearly optimal routes, and (b) to capture the many-to-one correspondence between compartments and items, once the routes have been designed. Phases (a) and (b) are run consecutively when the instance is large, thus accelerating execution. Phase (b) reduces to a matching problem between loads and compartments of the vessel(s) used in the given route. Several further optimizations have been applied.

- In large scale problems, we partition the geographic domain of interest to smaller domains which are independently solved in parallel to the main domain, then routes are combined so as to exclude the possibility of some destination being non-served.
- Supervized machine learning, in particular neural networks may be used for the software to learn the profile of different vessels so as to better assign cargoes to vessels while maximizing the loading factors.

## CONCLUSIONS

We have presented state-of-the-art methods relevant in shipping, autonomous maritime operations, and optimization of routing. These are mature algorithmic approaches developed within our Lab. We aspire at a technology transfer towards shipping, expecting a major impact in the aforementioned areas towards exploiting intelligence and automation. More precisely, we showcased our advances around (a) collision avoidance in complex environments, thus leading to semi-autonomous and fully autonomous navigation, and (b) optimization for routing under capacity constraints of various compartments within a vessel as well as time constraints of the carrier ship and time windows for loading and unloading at ports. The overall scheme of the paper is illustrated in the puzzle (see Fig. 5).

Further aspects of technology transfer regarding Artificial Intelligence and automation are currently possible, in the same vein as those developed in this article. To name only a few examples studied in our Lab, we mention Machine Learning for predicting fuel consumption by making use of the massive data available from the various sensors on a modern ship, smart and predictive maintenance based on same information, as well as intelligent machines and autonomous robots for cleaning, loading, and other such heavy-duty operations which are today costly in terms of personnel and time, as well as accident-prone.

## ACKNOWLEDGMENTS

We acknowledge partial support by LAMBDA: This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 734242.

## REFERENCES

Avgouleas, K. *Optimal ship routing*, MSc Thesis, Massachusetts Institute of Technology. Dept. of Mechanical Engineering. 2008.

Y. Avrithis, Y. Kalantidis, E. Anagnostopoulos, and I.Z. Emiris. "Web-Scale Image Clustering Revisited". In Proc.

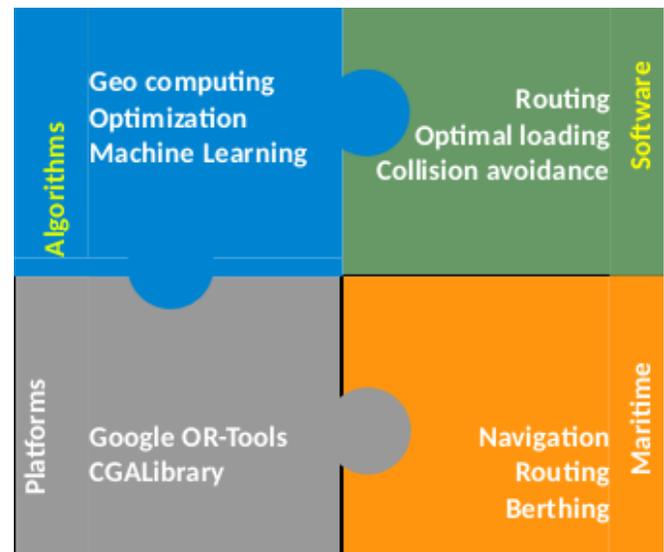


Fig. 5: Main components of our approach

- Intern. Conf. Computer Vision, pp. 1502-1510 (2015)
- CGAL, Computational Geometry Algorithms Library, <http://www.cgal.org>
- Emiris, I.Z., Tsigaridas E.P., and Tzoumas G.M., "Exact Voronoi diagram of smooth convex pseudo-circles: General predicates, and implementation for ellipses". *Computer Aided Geometric Design* 30:8 (2013): 760-777.
- Emiris, I.Z., and Tzoumas G.M., "A real-time and exact implementation of the predicates for the Voronoi diagram of parametric ellipses". In Proc. Symposium on Solid and Physical Modeling, pp. 133-142 (2007).
- R. Goralski, C. Gold, and M. Dakowicz. Application of the Kinetic Voronoi Diagram to the Real-Time Navigation of Marine Vessels, in Proceedings of "IEEE International Conference on Computer Information Systems and Industrial Management Applications (CISIM)", Minneapolis, USA, pp. 129-134 (2007)
- S. Koga. Major challenges and solutions for utilizing big data in the maritime industry. PhD Thesis, World Maritime University (2015)
- Le Carrer, N., Ferson, S., and Green, P.L. "Optimising cargo loading and ship scheduling in tidal areas", *European Journal of Operational Research*, 280:3 (2020).

Russel D., Karavelas MI, and Guibas L.J. "A package for Exact Kinetic Data Structures and Sweepline Algorithms". Computational Geometry: Theory and Applications, 38:1-2 (2007):111-127. Special Issue on CGAL.

Geng, X., Yongcai, W. Wang, P., and Zhang, B. "Motion Plan of Maritime Autonomous Surface Ships by Dynamic Programming for Collision Avoidance and Speed Optimization". Sensors (Basel). 19:2 (2019): 434.