Deliberative Optimization of Reactive Agents for Navigation

Anthony D. Jones

***Abstract*—A simple reactive steering behavior for agents navigating through a field that influences their motion is presented along with a deliberative method to optimize the parameters of this behavior. The behavior is described in the context of its application to unmanned maritime vehicle navigation, and the results of three simulation-based experiments are discussed. These simulated tests include two within a slow, uniform, steady-state velocity field, and one within a fast, non-uniform, time-variant velocity field. The steering behavior is compared with simple homing and tracking behaviors, and outperforms both in the case of a high-speed, variable current field. In the other current fields, the optimized steering behavior performs better on average than the homing behavior but not as well as the tracking behavior.**

A navigation system based on the optimized steering behavior was implemented aboard a small-scale autonomous surface vehicle (ASV), which was deployed on a series of field trials in the Eau Gallie River and the Indian River Lagoon. These field tests consisted of simple point-to-point navigation tasks, and in all cases the ASV reached its objectives.

I. Introduction

Unmanned maritime vehicles (UMVs) must navigate through ocean currents, which vary significantly in space and time. Planning a path through a current field is an important task for mission optimization, and several methods have been applied. Solutions have been developed that take advantage of A* [1, 2], Genetic Algorithms (GA) [3], particle swarm-optimization [4], and iterative optimization [5]. [6] includes an overview of several path planning methods that are currently in use.

In general these methods operate at the deliberative level of intelligence. At the reactive level, [7] describes a number of steering behaviors for which agents respond instantly, without deliberation. Some examples of these reactive behaviors include turning to avoid obstacles, maintaining close formation with other agents, and homing to a target. Due to the versatility of these and other similar behaviors, [7] provides the foundation for the development of the steering behavior presented in Section II.

This project seeks to use a versatile, optimized reactive steering behavior to alleviate the need for storing a mission's trajectory onboard an UMV. Instead, an agent reactively chooses its heading based only on local information. Specifically, this information includes the agent's own speed, location, and heading, the heading to its goal, and the local current speed and direction. The reactive decisions that an agent makes are based on a deliberative optimization process that uses a simple kinematic model of the agent's transit through a forecast of its environment. Thus for brevity in the remainder of this paper, these agents will be called Deliberatively Optimized Reactive (DOR) agents, and the system will be referred to as DORA.

The following sections describe the steering behavior as well as an agent that implements it, an optimization process that tunes it, and the results of experimental simulations that test its performance using real-world ocean current data. Finally, the steering behavior is implemented aboard a small-scale autonomous surface vehicle (ASV), and a series of field trials are conducted.

II. Development

The deliberative optimization algorithm takes advantage of some of the principles of Swarm Intelligence by implementing a large number of simple agents, which act on local information. These agents behave according to the following preferences:

1) Travel with strong currents.

978-0-933957-40-4

2) Maintain heading.
3) Travel toward the objective.

The software models these agents' motion through a simulated environment, tracks their successes and failures, and attempts to optimize the preferences of following generations of agents to make them more effective in solving the problem. This optimization currently uses a stochastic search where values are normally distributed about the best solution found thus far. Other techniques, including genetic algorithms (GA) may be explored in later experiments. The preferences of the most successful agent are recorded as the solution to the problem.

Any of these preferences can take on negative values, and thus avoid rather than encourage its respective behavior. The preferences will often conflict, and the relative importance of each will depend upon the particular circumstances of the agent's environment. There are many potential means of resolving conflicting preferences, however in these experiments, a simple weighted vector addition is used. Equation 1 defines the preference vector. The DOR agent attempts to align its heading with the direction of the preference vector at each heading update.

$$\boldsymbol{p} = \boldsymbol{b}\left(1 - \frac{V_a}{V_a + V_\infty}\right) + \boldsymbol{c} + \boldsymbol{d} \tag{1}$$

$$\boldsymbol{b} = \langle B\sin\theta_\infty, B\cos\theta_\infty\rangle \tag{2}$$

$$\boldsymbol{c} = \langle C\sin\theta_a, C\cos\theta_a\rangle \tag{3}$$

$$\boldsymbol{d} = \langle D\sin\theta_T, D\cos\theta_T\rangle \tag{4}$$

θ_∞	Heading of Local Current
V_∞	Magnitude of Velocity of Local Current
V_a	Magnitude of Velocity of Agent
θ_a	Instantaneous Heading of Agent
θ_T	Heading to Agent's Target
B	Behavioral Tuning Coefficient for Current
C	Behavioral Tuning Coefficient for Agent's Course
D	Behavioral Tuning Coefficient for Heading to Destination

The term $\left(1 - \frac{V_a}{V_a+V_\infty}\right)$ diminishes the weight of the local current preference as the agent's speed becomes large compared to the current. B,C, and D are parameters to the algorithm, which may take on any value from -1 to 1 inclusive, but are normalized so that $|B| + |C| + |D| = 1$.

The first preference, weighted by B, reflects an agent's tendency to either turn into or away from the local current. If B takes on a positive value, this preference will tend to align the agent with the local current, which prevents the agent's expending energy by trying to overcome a strong current. In some cases it may be advantageous for the agent to turn against the local current, and for these a negative value of B would be appropriate.

The second term influences stability of the agent's course. In the absence of any other influencing factors, a straight line course is a better solution than a curved one due to the shorter distance required. A positive value of C acts to reduce deviation from the agent's course, and a negative value of C promotes changing course. C can be viewed as adding inertia to the agent's heading.

Finally the third preference, with the weight D, tends to face the agent toward the objective. If all other terms are ignored, a positive value of D produces an agent that homes toward its goal. All other preferences act to compensate for mission-specific conditions, whereas this one independently comprises a valid solution to the problem.

III. Experiment

To evaluate the algorithm's potential for use in autonomous navigation, a series of experiments was conducted. Each experiment consisted of either 200 or 1200 individual trials. In all cases, starting and ending points were randomly generated, but were constrained to be approximately 150 kilometers apart.

A. *Program Environment Specifications*

All trials were conducted using the Python 2.7 programming language employing the included modules `time`, `random`, and `sys`. The third-party modules `numpy` and `scipy` were used for most mathematical and data access operations. The experiments were executed on a laptop computer with an Intel i7-2630QM 2.0GHz quad-core processor and 8 GB of RAM running the 64-bit Ubuntu 12.04 operating system.

B. Agents

All experiments began with 50 randomly initialized DOR agents, as well as one agent that used a homing behavior and one agent that used a tracking behavior.

The homing agent aligned its heading directly toward its objective. Since all of its available speed was in the direction of its objective, it made no attempt to compensate for local currents, which resulted in its traveling faster along its path, but over greater distance than the tracking agent.

The tracking agent resolved the local current into vector components with one oriented parallel to the agent's path, and the other perpendicular. The parallel component only acted to speed or slow the agent's approach to its objective, but did not alter the agent's path. The perpendicular or cross-track component pushed the agent off its course.

The tracking agent oriented itself to compensate for the cross-track component when this component was slower than the agent's speed. This compensation minimized the distance that the agent was required to travel, and represented an ideal solution in the case of a constant, uniform velocity field that was slower than the agent's speed. When the cross-track component was greater than or equal to the agent's speed, the agent used the homing behavior.

C. Physical Model

The agents in this algorithm are intended to model the behavior of physical vehicles deployed on a real-world mission, so the agents operate in a simplistic kinematic model of an ocean current field. Each agent has a heading, a speed, and a position stored as internal, local variables. At each time step the agent's speed is resolved into East and North components of velocity. These are then added to the East and North components of local current, and the agent's position is updated by multiplying the resulting total velocity vector by the model's timestep.

In experiments to evaluate the agents' ability to solve known problems, the model is constrained to moving and navigating on a planar surface in a constant, uniform velocity field. However, in the more practical experiments, motion is modeled along a

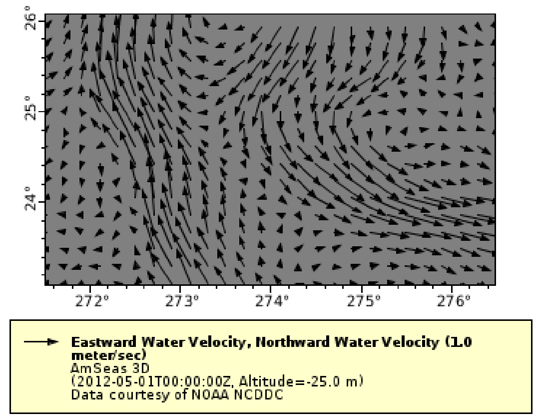

Fig. 1. Velocity Field at Initial Timestep. Graphics generated by ERDAP Service at http://wwww.northerngulfinstitute.org

spherical approximation of Earth that assumes a radius of 6371.0 kilometers [8]. Agent position is tracked as latitude and longitude. Since the kinematic model uses normal SI units, the agent's change in position is calculated in meters, then converted to a change in latitude and longitude, where the conversion factor is based on the agent's position before the update. This introduces an error since the conversion from meters to degrees of longitude changes continuously as latitude changes, but since the change in position during a typical timestep of the model is small compared to the radius of Earth, the error is also small.

D. Data and Region of Interest

Oceanographic data for these trials is a four-dimensional array of North and East components of velocity. The region of interest lies between $23.1°$ and $26.1°$ North latitude, and between $83.5°$ and $88.5°$ West longitude. The time chosen for these trials is the week beginning at midnight, May 5, 2012. The data grid has a temporal resolution of three hours, and a horizontal resolution of approximately $1/36$-degree at standard depths (www.northerngulfinstitute.org). The source of the data was described in e-mail correspondence with Dr. John Harding of the Northern Gulf Institute as follows:

The U.S. Navy AMSEAS

nowcast/forecast retrospective data was provided through a NOAA/Navy cooperative effort including the Naval Oceanographic Office, NOAA's National Coastal Data Development Center and the Northern Gulf Institute, A NOAA Cooperative Institute.

For these experiments, all trials were conducted using a fixed depth of 25.0 meters. Since the physical model and planner can run at an arbitrarily specified resolution, four-dimensional linear interpolation is used to estimate the velocity at any requested depth, time, and position within the data set. For any request outside the valid range of the data, the interpolation function returns a non-numeric error value. The first timestep is shown in Fig. 1.

E. Test Conditions

The objective of each trial was to minimize the time required to navigate over a 150-kilometer course. To ensure a wide variety of problems, the start and end points of the test mission were selected using the programming environment's native pseudo-random number generator.

In each case, 50 agents were initialized at the trial's starting point, and their positions and headings were updated at each timestep of the physical model according to their respective behaviors. If an agent moved outside the boundaries of the data or arrived at the destination, the agent was removed from the trial. If an agent reached the destination, its time was recorded, and the overall minimum time for each type of agent was tracked for each trial. Additionally, the behavioral tuning parameters for the DOR agent with the minimum time was recorded. For this project, the starting point and the objective were only varied between trials, so all agents for a given trial attempted to solve the same problem.

The criterion for successfully reaching the intended destination was that the agent must be no farther from its destination than the spatial resolution of the physical model, which was calculated by adding the maximum agent speed to the maximum local current speed, and multiplying by the model's time step. While there is some error introduced by using this simple range threshold criterion, the error can be kept to an acceptable minimum by choosing a sufficiently high temporal resolution.

To reduce the error produced by the range threshold approximation, the agent's rate of change of distance to its objective was calculated for the timestep preceding the agent's crossing the range threshold. The remaining distance to the objective was then divided by the rate of change to give an estimate of the remaining time required to reach the destination assuming that rate of change remained constant.

A single epoch of a trial consisted of the span of time during which any agent was still being modeled. Once all agents were removed due to success or failure, the next epoch began with the introduction of another 50 DOR agents, one tracking agent, and one homing agent. Each trial continued until it had executed seven epochs.

At the end of the trial, relevant data were recorded to a file including the trial number, the starting and ending points, the minimum time achieved by each class of agent, and the tuning parameters of the best DOR agent.

IV. Results

A. Planar Earth with High Velocity

To test the performance of the algorithm and of the physical model, an experiment was run in which all navigation and physical modeling was carried out using a planar model of the mission area. The local velocity field was set to a constant 2.0 m/s directly from West to East, and the agents' speed was set at 3.0 m/s. This choice of a simplistic field and a relatively high agent speed ensured that all trials would have valid, analytically calculable solutions. The results of these trials are shown in Table I.

As expected, all agents found solutions in this simple case. Analytically, the tracking agents should arrive at a point objective in a predictable time, which is calculated by dividing the distance from the starting point to the objective by the net speed remaining after correcting for a local current. Over the 200 trials in this experiment, the average absolute difference between the predicted transit time

TABLE I
RESULTS OF PLANAR EARTH TRIAL

Tracking Solved	100.00 %
Homing Solved	100.00 %
DORA Solved	100.00 %
Tracking Best	97.5 %
Homing Best	0.00 %
DORA Best	2.50 %
Average Tracking Error	2.23×10^{-6} s
Average DORA v. Tracking	2.37×10^{-2} %
Average DORA v. Homing	-13.39 %
Average Epoch	4.41
Average Time	163.10 s

and the modeled transit time was 2.23 microseconds for tracking agents, while their average total mission duration was $18\,780.7$ seconds.

The tracking agents implemented the best solution found in 97.5% of trials, while the DOR agents found a better solution in 2.5% of trials. The reason that the tracking solution was not the best possible solution found in all cases is that the range threshold criterion for arrival permitted a slight improvement in mission time by using more of the agents' available speed to move toward the objective, and accepting a small amount of leeward drift. Finding this slightly improved solution was a rare occurrence, and on average the DOR agents performed about $2.37 \times 10^{-}2\%$ worse than the tracking agents.

This result emphasizes some of the advantages and disadvantages of the algorithm. Specifically, it consistently approached the optimal solution, and produced results that were comparable to the tracking agents', but without any assurance of optimality.

The average amount of time required for each trial was 163.10 seconds for all seven epochs, however the average number of epochs required to reach the DOR agents' best solution was 4.41. This suggests that some time may be saved in future iterations by concluding the trials after fewer epochs. However, some trials required the full seven epochs to find their solutions, so using fewer epochs will necessarily sacrifice some optimality.

B. Spherical Earth with High Velocity

To verify that the agents' performance and the physical model transitioned well from the simplistic planar approximation to the spherical model of Earth, the same initial configuration from the previous trial was retained, but all navigation and physical model functions were replaced with the spherical model. Table II summarizes the results of this experiment.

TABLE II
RESULTS OF SPHERICAL EARTH WITH HIGH VELOCITY

Tracking Solved	100.00 %
Homing Solved	100.00 %
DORA Solved	100.00 %
Tracking Best	83 %
Homing Best	0 %
DORA Best	17 %
Average DORA - Tracking	3.70×10^{-2} %
Average DORA - Homing	-15.07 %
Average Epoch	4.47
Average Time (s)	120.90

Again all three classes of agent found solutions in every trial. In this experiment, the tracking agents found the best solution in 90.5% of cases, while the DOR agents found the best solution in the remaining 9.50%.

The DOR agents performed comparably to the tracking agents with an average performance $3.70 \times 10^{-2}\%$ worse than the tracking agents', but 15.07% better than the homing agents'.

The average time required for execution of all seven epochs was 120.90 seconds, while the DOR agents' required an average 4.47 epochs to reach their best solutions.

These results are very similar to those from the planar approximation, as would be expected. The tracking behavior still represents the best solution for navigation to a point, but the DORA optimization is again able to find a better solution for navigation to a small circle around the objective in a few cases. The overall result is that the DOR agents are nearly equivalent to the tracking agents in this experiment, as they were in the preceding planar experiment.

C. Spherical Earth with Low Velocity

The preceding experiments were conducted to validate the performance of the DORA algorithm in fields with known solutions, and to compare the model's results with the analytical solution to the tracking problem. Direct comparisons with the tracking agents showed that on average, the DOR agents found comparable solutions to the problem, however since a constant, uniform current is a trivial case, there is hardly justification for using any optimization at all.

A more practical problem includes the variations observed in currents in a real ocean. Here, currents vary significantly in space and time, and are not guaranteed to be less than the vehicle's speed. This presents a much more challenging environment for navigation and a much more reasonable approximation of the environment that an UMV employing this steering behavior might face. So for the next trial, the data described in Section III-D were used as the source for ocean currents, and the vehicle speed was limited to 0.50m/s.

Since the maximum local velocity in the data set was slightly larger than 2.0m/s, there was no guarantee that the slow-moving agents would be able to navigate to their objectives. To ensure a sufficient number of successful results for evaluation, this experiment was run for 1200 trials rather than the 200 in the preceding experiments. The fact that no solution was guaranteed to exist allowed for evaluation of the DOR agents' robustness and ability to be adapted for use in situations when simple tracking or homing behaviors failed. The success rates of the three classes of agents are shown in Table III.

TABLE III
SUCCESS RATES FOR SPHERICAL EARTH WITH LOW SPEED

Tracking Solved	24.92 %
Homing Solved	21.25 %
DORA Solved	43.50 %
Average Epoch	4.38
Average Time (s)	124.55

Of the 1200 trials conducted, only 524 were solved by any of the agents. DOR agents found solutions to 43.5% of cases, which was almost twice the number of solutions found by either of the other classes of agents. This is instructive when considering the versatility of the agents, but for comparison between them, only the cases when at least one class of agent found a solution are considered. This is the basis of the percentages shown in Table IV.

TABLE IV
RESULTS OF SOLVED TRIALS ON SPHERICAL EARTH WITH LOW VELOCITY

	Tracking	Homing	DOR
Successful Trials	299	255	522
Success Rate (%)	57.06	48.66	99.62
Best Solution	31	0	493
Best Solution (%)	5.92	0.00	94.08

For the 524 trials with successful solutions, the DOR agents had the best performance with both the highest number of solutions found and the highest number of best solutions found.

V. FIELD TESTING

To evaluate the potential of using the DORA system for practical vehicle navigation, a small ASV was programmed to use the steering behavior defined by Equation 1. The ASV used an Arduino Mega 2560 rev. 3 compatible microcontroller to implement the algorithm, and sensed its location and heading using a GPS module and tilt-compensated compass respectively. To estimate ambient current, the ASV's software calculated its expected position based on dead reckoning with an assumption of zero current. This expected position was compared with a new GPS fix, and the difference between the two positions was assumed to be a result of the local current. All mission-relevant data was logged to an onboard micro SD memory card for post mission analysis.

The ASV performed five test missions in the Indian River Lagoon, and one in the Eau Gallie River. For each mission, when the ASV was launched it would record its starting position and heading, and would project a waypoint 250 meters ahead. It would then use the DOR steering behavior to navigate to its destination, and then to return. Since a solution is not symmetric, the outbound and inbound legs function as separate trials, for a total of twelve tests.

In all cases, the ASV successfully navigated to its objective and returned to its initial point. Fig. 2

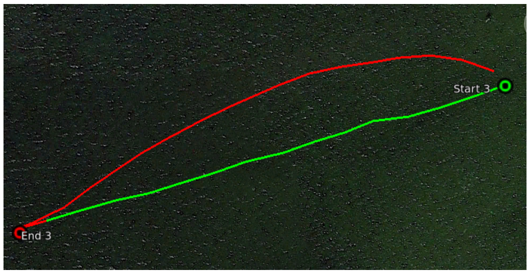

Fig. 2. Path recorded from the ASV's mission log. The Southern segment represents the outbound leg and the Northern segment the return. Image recorded from Google Earth.

shows a plot of the ASV's path during its third field trial in the Eau Gallie River.

VI. Conclusions

In all three experiments conducted, the DOR agents' steering behavior performed well with deliberatively optimized preference parameters. It was consistently better than a simple homing behavior, performed comparably with the tracking behavior in low speed, constant-current fields, and outperformed both other behaviors in the high speed, variable-current field. The behavior was then implemented as a navigation system for an autonomous surface vehicle, and though it was not optimized for its specific operational environment, the ASV successfully navigated to each of its objectives.

The results from Section IV-B show that if the local current is slow compared to the vehicle speed, there is no advantage to the optimization process, and a simple local current correction works well. However, Section IV-C demonstrates that if the vehicle speed is slow compared to the local current, the flexibility of the DORA algorithm allows for significantly improved performance.

VII. Acknowledgements

The author would like to acknowledge the contributions of Ph.D. committee members Dr. S. Wood, Dr. R. Reichard, Dr. M. Silaghi, and Dr. H. Gutierrez as well as the assistance of colleagues at Florida Institute of Technology's Underwater Technology Laboratory.

References

[1] B Garau, M Bonet, A Alvarez, and S Ruiz. Path Planning for Autonomous Underwater Vehicles in Realistic Oceanic Current Fields: Application to Gliders in the Western Mediterranean Sea. *Journal of Maritime Research*, VI(Ii):5–22, 2009.

[2] E Fernandez-Perdomo, J Cabrera-Gandmez, D Hernandndez-Sosa, J Isern-Gonzandlez, A C Domiandnguez-Brito, A Redondo, J Coca, A G Ramos, E A Fanjul, and M Garcianda. Path Planning for gliders using Regional Ocean Models: Application of Pinzon path planner with the ESEOAT model and the RU27 trans-Atlantic flight data. In *OCEANS 2010 IEEE - Sydney*, pages 1–10, May 2010.

[3] Alberto Alvarez, Andrea Caiti, and Reiner Onken. Evolutionary path planning for autonomous underwater vehicles in a variable ocean. *Oceanic Engineering, IEEE Journal of*, 29(2):418–429, 2004.

[4] Jonas Witt and Matthew Dunbabin. Go with the Flow : Optimal AUV Path Planning in Coastal Environments Autonomous Systems. Technical report, 2008.

[5] E Fernandez-Perdomo, D Hernandez-Sosa, J Isern-Gonzalez, J Cabrera-Gamez, A C Dominguez-Brito, and V Prieto-Maranon. Single and multiple glider path planning using an optimization-based approach. In *OCEANS, 2011 IEEE - Spain*, pages 1–10, June 2011.

[6] Mae L. Seto, editor. *Marine Robot Autonomy*. Springer, New York, NY, 2013.

[7] Craig W Reynolds. Steering Behaviors For Autonomous Characters. In *Game Developers Conference*, pages 763–782, San Jose, California, 1999. Miller Freeman Game Group.

[8] Nathaniel Bowditch. *THE AMERICAN PRACTICAL NAVIGATOR*. Number 9. National Imagery and Mapping Agency, Bethesda, MD, 2002.