Path Planning in Time Dependent Flow Fields using Level Set Methods

Conference Paper *in* Proceedings - IEEE International Conference on Robotics and Automation · May 2012

DOI: 10.1109/ICRA.2012.6225364

5 authors, including:

Tapovan Lolla
Massachusetts Institute of Technology
8 PUBLICATIONS **91** CITATIONS
SEE PROFILE

Mattheus P Ueckermann
Massachusetts Institute of Technology
9 PUBLICATIONS **118** CITATIONS
SEE PROFILE

Pierre F. J. Lermusiaux
Massachusetts Institute of Technology
123 PUBLICATIONS **2,673** CITATIONS
SEE PROFILE

2012 IEEE International Conference on Robotics and Automation
RiverCentre, Saint Paul, Minnesota, USA
May 14-18, 2012

Path Planning in Time Dependent Flow Fields using Level Set Methods

T. Lolla, M. P. Ueckermann, K. Yiğit, P. J. Haley Jr. and P. F. J. Lermusiaux

Abstract— **We develop and illustrate an efficient but rigorous methodology that predicts the time-optimal paths of ocean vehicles in continuous dynamic flows. The goal is to best utilize or avoid currents, without limitation on these currents or on the number of vehicles. The methodology employs a new modified level set equation to evolve a front from the starting point of a vehicle until it reaches the desired goal location, combining flow advection with nominal vehicle motion. The optimal path of the vehicle is then obtained by solving a particle tracking equation backward in time. The computational cost of this method increases linearly with the number of vehicles and geometrically with spatial dimensions. The methodology is applicable to any continuous flow and in scenarios with multiple vehicles. Present illustrations consist of the crossing of a canonical uniform jet and its validation using a classic optimization solution, as well as swarm formation in more complex time varying 2D flow fields, including jets, eddies and forbidden regions.**

I. INTRODUCTION

Planning a path in a complex environment is a problem as old as antiquity. Even then, prior knowledge, however limited, was employed in predictions for such planning. In contemporary science and engineering, modeling and computational approaches are utilized to plan paths that optimize an objective criterion. In many cases, the performance optimization is done for autonomous vehicles; the optimal plans are then provided to the vehicles which can further adapt their plans as they execute their mission. When the environment is the ocean, a highly dynamic and multiscale system with considerable variability in both time and three-dimensional space, the planning of optimal paths is challenging. In addition, currents can be strong and much larger than vehicle speeds, and the ocean geometry can be complex, especially in the coastal zone. Finally, the number and capabilities of ocean vehicles are increasing rapidly. As a result, our motivation is to develop and illustrate efficient but rigorous methodologies that predict the optimal paths of swarms of ocean vehicles in dynamic ocean currents, without any limitation on the currents or on the number of vehicles.

Path planning for autonomous underwater vehicles (AUVs) in general aims to optimize at least one of the following aspects of performance: i) travel time between two given points; ii) energy spent by the vehicle; iii) safety of the vehicle. Optimal navigation of autonomous vehicles in the coastal ocean has become crucial for many applications, from security and acoustic surveillance to the collection of ocean data at specific locations for ocean prediction and monitoring. In all of these applications, AUVs (gliders, propelled vehicles and other platforms) must be navigated so that they optimize one or more of the above mentioned performance criteria. In what follows, we will show that if AUVs travel at nominal speed in a time dependent flow field, our methodology will compute the exact fastest path between any two locations with a computational cost that grows linearly with the number of vehicles and geometrically with spatial dimensions (instead of exponentially).

Results on planning the path of autonomous vehicles for robotic applications have been obtained for several years now. However, the literature on path planning in complex realistic time-dependent flow fields is rather limited. Most methods for path planning either fail when the environment becomes complex, or are computationally expensive thus making them unsuitable for real time applications with large number of vehicles. Well established methods in robotic path planning applications have not been designed to handle situations with dynamic environments. A recent trend in research on path planning methods has been to develop algorithms which use the dynamic nature of the environment to reduce the energy expended by the vehicle. A closely related problem is to obtain paths which minimize the total travel time of the vehicle when propelled at nominal speed. In [1], the authors propose a genetic algorithm based on Darwinian theories of natural selection for path planning in strong ocean currents. A set of feasible paths is generated and these paths are iteratively transformed by using genetic operators like crossover and mutation. The path that minimizes a suitable cost function is chosen. In [2], a path planning scheme based on mixed integer linear programming (MILP) is presented. This work focuses on the problem of adaptive sampling in the ocean. Adaptive sampling refers to the task of predicting the types and locations of ocean measurements that would be most useful to collect [3]. The usefulness of measurements is governed by an objective function and the path planning algorithm finds a vehicle path along which the line integral of this objective function is optimized. This problem is NP-hard even though uncertainty fields are assumed to be stationary and ocean currents are ignored. An application of the A* search scheme for path planning of AUVs in the ocean is described in [4]. In their work, energy optimal paths are calculated in a simulated ocean environment with high spatial variability in the form of different types of eddies. The effect of different heuristic functions on the performance of the A* scheme is analyzed. The main drawback of this A* search is that the ocean currents are assumed to be steady. In [5], Rapidly-exploring Random Trees (RRTs) are used to solve the path planning problem. These trees are rooted at the start and the goal locations and are incrementally built to explore

All authors are with Department of Mechanical Engineering, Massachusetts Institute of Technology, 77 Mass. Ave., Cambridge - MA 02139, USA. Corresponding author: pierrel@mit.edu the space around them, using a greedy heuristic. RRTs have been widely used in robotic path planning, particularly in situations with dynamic obstacles. This randomized approach to path planning has also been used for underwater vehicles to obtain obstacle free paths [6].

An alternate path planning technique based on potential field algorithms is described in [7]. These algorithms employ artificially generated potential fields on obstacles and on the goal to compute a safe path for the vehicle. Although this method works particularly well in avoiding obstacles, it tends to produce paths which are not globally optimal and that do not account for all flow variabilities.

Path planning for underwater gliders using a variational calculus approach is discussed in [8]. The authors derive 'ray' equations for routing gliders through steady velocity fields. Under these restrictions, their method is similar to ray tracing for non-dispersive waves. In [9], a first order fast marching scheme is introduced. This scheme solves the discretized Hamilton-Jacobi equation for a trajectory optimization problem in steady fields. The fast marching algorithm for continuous trajectory optimization is similar to a continuous version of Dijkstra's algorithm.

Jarvis [10] introduced a wavefront expansion algorithm for obstacle avoidance using ideas of the 'distance transform' methodology [11]. Distance transforms have been widely used for path planning in stationary fields. This approach propagates a distance wave through the domain, from start to goal. The shortest path to the goal is then traced by following the steepest descent. A comparison of A* search, RRTs and distance transforms is presented in [12].

A continuous approach to path planning in a field of currents is presented in [13]. The paper describes a fast marching algorithm for path planning using anisotropic cost functions. Directional constraints such as those enforced by ocean currents are taken into account using these cost functions. Two main drawbacks of the scheme are that it only accommodates linear energy cost functions and it can lead to infeasible paths in the presence of strong ocean currents. In [14], the scheme is improved to yield better results for vehicle motion in such strong currents. The technique uses a 'symbolic wavefront expansion' to calculate shortest time paths and also determines the departure time of the vehicle from the starting point. More recently, [15] introduces a 'sliding wavefront expansion' technique for path planning in strong currents. The algorithm combines appropriate cost functions with continuous optimization techniques to guarantee the existence of a feasible path.

In [16], the authors study the same problem which we are interested in. The problem is to steer a vehicle from its initial position to a desired target position in minimum time. Optimal vehicle trajectories are computed using a modified iterative extremal field approach that indirectly solves a Hamilton Jacobi Bellman equation for the feedback control law using Euler-Lagrange equations and a two point boundary value problem. Our goal here is to derive a methodology that solves differential equations, providing the rigorous exhaustive solution while still being computationally efficient.

In what follows, in Section II, we formally define the problem we wish to solve and introduce relevant notation. In Section III, we describe level set methods and provide a theorem that we employ for time optimal path planning. In Section IV, we describe our algorithm and its numerical implementation. In Section V, we first validate the method using simple test cases and then present its results for swarm formation in complex time varying 2D flow fields, including jets, eddies and forbidden regions. Conclusions and directions for future work are in Section VI.

II. Problem Statement

Let $\mathbf{x}$ denote a position vector in space (Fig. 1). Consider the motion of a vehicle in a time dependent external velocity field given by $\mathbf{V}(\mathbf{x}, t)$. Let $\mathbf{x_s}$ and $\mathbf{x_f}$ denote the position vectors of the starting location and desired goal of the vehicle, respectively. Let the nominal speed of the vehicle with respect to the velocity field be a constant F. We wish to steer the vehicle from $\mathbf{x_s}$ to $\mathbf{x_f}$ in minimum possible time. In other words, we wish to predict the headings of the vehicle that will minimize its travel time. For convenience, we assume $\mathbf{x_s}$ to be the origin of the coordinate system. In this study, we assume that the distance traveled by the vehicle is much larger than its dimensions and neglect the hydrodynamic interactions of the vehicle and the flow. We also assume that the predicted flow is exact. The effects of accounting for flow uncertainties using dynamically orthogonal equations [17], [18] will be reported elsewhere.

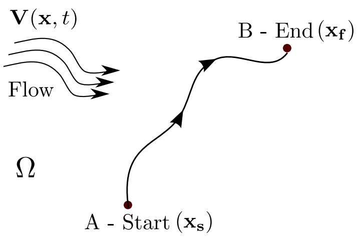


Fig. 1. Vehicle motion in time-varying flow field

III. Path Planning using Level Set Methods

In this section, we describe the level set method and its application to path planning, and provide a theorem that sets the foundation of our approach. Consider a front $\partial\Omega$, in two or three dimensions. A simple example of such a front is the interface between two fluids. The simplest way to represent the front is by using a discrete set of points or 'markers'. This *explicit* representation involves approximating an infinite set of points on the front by a finite number of points assuming that they characterize the behavior of the front. This method of front representation is also called *marker* or *string* method [19]. An alternate representation is to assume that the front is an isocontour of a suitable function. A *level set* of a function $f(\mathbf{x})$, $\mathbf{x} \in \mathbb{R}^n$ is then defined as the set of points at which the function takes a given constant value. Mathematically, the level set of $f(\mathbf{x})$ is the set, $\{\mathbf{x} | f(\mathbf{x}) = C\}$, where C is a given constant. This is an *implicit* representation of the front.

Level set methods are used to add dynamics to the implicit front and to capture the interaction between surface motion and the fluid forcing [21]. They were originally designed to solve problems related to fluid-interface motion [20]. They enable interface tracking in systems where front evolution is intricately connected to various physical properties of the system. These methods use an implicit representation of the interface and are formulated by an initial value partial differential equation. A Hamilton-Jacobi approach can be used to derive this equation that governs the propagation of the front or interface [19]. Therefore, the level set equation is a Hamilton-Jacobi equation. Level set methods can also be used to compute and analyze the motion of an interface in a velocity flow field. An example of the use of level set approaches in the ocean is the mapping of ocean data in complex regions [22]. We will employ an implicit representation for the front because of the numerous advantages it offers over an explicit one [21].

In an explicit representation of the front, the trajectory of each marker particle $\mathbf{x}_i$ evolves according to (1).

$$\frac{d\mathbf{x}_i}{dt} = \mathbf{U}(\mathbf{x}_i) \tag{1}$$

where $\mathbf{U}(\mathbf{x}_i)$ is the total marker velocity at $\mathbf{x}_i$. We remarked earlier that the implicit front representation uses the concept of level sets of a suitable function. The choice of this function is somewhat arbitrary. The most common type of function used for this purpose is the *signed distance function*, denoted by $\phi(\mathbf{x})$. As the name suggests, a distance function, $d(\mathbf{x})$, is the shortest distance from point $\mathbf{x}$ in space to the front $\partial\Omega$. Mathematically,

$$d(\mathbf{x}) = \min_{\mathbf{x_i}} |\mathbf{x} - \mathbf{x_i}|, \text{ for all } \mathbf{x_i} \in \partial\Omega \tag{2}$$

A signed distance function, is defined as:

$$\phi(\mathbf{x}) = \begin{cases} d(\mathbf{x}), & \text{if } \mathbf{x} \text{ is outside the front} \\ -d(\mathbf{x}), & \text{if } \mathbf{x} \text{ is inside the front} \end{cases} \tag{3}$$

For every point $\mathbf{x_i}$ on the front, $\phi(\mathbf{x_i}) = 0$. Consequently, the front is *implicitly* represented as the zero level set of $\phi(\mathbf{x})$. Signed distance is a preferred choice for the implicit function because it is smooth and avoids steep gradients in the isocontours. The idea of level set methods is to evolve ϕ, whose zero level set always corresponds to the front. Since $\phi(\mathbf{x})$ changes with time as the front evolves, it is also a function of time t. We shall occasionally write $\phi(\mathbf{x}, t)$ simply as ϕ with the understanding that it denotes a time-varying scalar field in space.

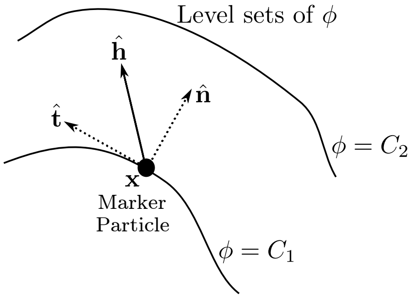


Fig. 2. Level Sets of ϕ: Tangential and Normal directions

Theorem: Let $T(\mathbf{y})$ denote the minimum time in which a vehicle can reach $\mathbf{y}$, if it starts from $\mathbf{x} = 0$ at $t = 0$, at nominal speed of constant magnitude F in a velocity field given by $\mathbf{V}(\mathbf{x}, t)$. Consider the evolution of $\phi(\mathbf{x}, t)$ according to the following initial value partial differential equation:

$$\frac{\partial \phi(\mathbf{x}, t)}{\partial t} + F|\nabla\phi(\mathbf{x}, t)| + \mathbf{V}(\mathbf{x}, t) \cdot \nabla\phi(\mathbf{x}, t) = 0 \tag{4}$$

with

$$\phi(\mathbf{x}, t = 0) = ||\mathbf{x}||_2 \tag{5}$$

where $|| \bullet ||_2$ stands for the L-2 norm. Then,

(a) $\phi(\mathbf{y}, T(\mathbf{y})) = 0$ and $\nexists\ t < T(\mathbf{y})$ such that $\phi(\mathbf{y}, t) = 0$.

(b) The optimal path is governed by:

$$\frac{d\mathbf{x}}{dt} = -\mathbf{V}(\mathbf{x}, t) - F\frac{\nabla\phi(\mathbf{x}, t)}{|\nabla\phi(\mathbf{x}, t)|} \tag{6}$$

integrating backward in time starting from $\mathbf{y}$.

Sketch of Proof: (a) The equation describing the evolution of a front in an externally generated velocity field $\mathbf{V}(\mathbf{x}, t)$, is given by the level set equation, (7), [21]:

$$\frac{\partial\phi}{\partial t} + \mathbf{V}(\mathbf{x}, t) \cdot \nabla\phi = 0 \tag{7}$$

Now, let us assume that in addition to the external velocity field, the front also moves in a direction normal to itself at a constant speed, $F(> 0)$. In other words, this component of the front's motion can be thought of as an internally generated velocity field, $F\hat{\mathbf{n}}$. Since $\hat{\mathbf{n}} = \frac{\nabla\phi}{|\nabla\phi|}$, we get, $\hat{\mathbf{n}} \cdot \nabla\phi = \frac{\nabla\phi}{|\nabla\phi|} \cdot \nabla\phi = |\nabla\phi|$. Therefore, the governing level set equation for a front which moves normal to itself at constant speed F, in an external velocity field $\mathbf{V}(\mathbf{x}, t)$ is given by (4).

This front is implicitly represented as the zero level set of the function $\phi(\mathbf{x}, t)$. The initial condition (5) in the theorem ensures that the zero level set is initially, a point located at the origin. The level set is a closed hyper-surface which, at the initial time, has a singularity at the origin. As time progresses, this level set evolves according to (4).

Let us introduce an imaginary vehicle P, that stays on the front at all times. In other words, this point vehicle P experiences the same flow field $\mathbf{V}(\mathbf{x}, t)$ as the front and has a speed F in a direction normal to the level set. This vehicle is analogous to a fixed 'marker' particle on the front. Let the path of this vehicle be denoted by $\mathbf{x}(t)$. Since the vehicle is advected along with the zero level set, it is always located on the zero level set, and we have,

$$\phi(\mathbf{x}(t), t) = 0 \tag{8}$$

$T(\mathbf{y})$ is the minimum possible time in which a vehicle can reach point $\mathbf{y}$. For any path taken by the vehicle to reach $\mathbf{y}$, we can construct a front which moves in the same way as the vehicle so that the vehicle always stays on the front, using (4). Following the discussion in the previous paragraph, whenever $\mathbf{x}(t) = \mathbf{y}$, i.e. whenever the vehicle reaches $\mathbf{y}$, we get, $\phi(\mathbf{y}, t) = 0$. $T(\mathbf{y})$ is simply the minimum of all such times, t, and therefore, $\phi(\mathbf{y}, T(\mathbf{y})) = 0$. This completes the proof of the first part of part (a) of the theorem.

For the second part of (a), it suffices to show that $T(\mathbf{y})$ is minimum when the level set evolves normally to itself. In other words, we need to show that the heading of the vehicle at every time is the normal to the zero level set passing through the vehicle location at that time. To do this, we assume that the vehicle heading direction is a linear combination of the unit vectors in the tangential ($\hat{\mathbf{t}}$) and normal ($\hat{\mathbf{n}}$) directions (Fig. 2). We re-write the level set equation under such a heading of the vehicle and write down expressions for $T(\mathbf{y})$. We argue then, that to minimize $T(\mathbf{y})$, the component of $\hat{\mathbf{h}}$ along $\hat{\mathbf{t}}$ must be zero. This terminates the proof of (a).

(b) The total velocity of the vehicle P has two components: the advection due to the flow field, $\mathbf{V}(\mathbf{x}, t)$, and its nominal motion in a direction normal to the level set, $F\hat{\mathbf{n}}$. This gives:

$$\frac{d\mathbf{x}}{dt} = \mathbf{V}(\mathbf{x}, t) + F\hat{\mathbf{n}} = \mathbf{V}(\mathbf{x}, t) + F\frac{\nabla\phi(\mathbf{x}, t)}{|\nabla\phi(\mathbf{x}, t)|} \tag{9}$$

with $\mathbf{x}(0) = \mathbf{0}$ and $\mathbf{x}(T(\mathbf{y})) = \mathbf{y}$. Thus, if we solve (9) backward in time, i.e. with $\mathbf{x}(0) = \mathbf{y}$ and $\mathbf{x}(T(\mathbf{y})) = \mathbf{0}$, both components of velocity are reversed and this leads to (6). Hence, the optimal path can be computed by solving (6) backward in time, starting from $\mathbf{y}$. This completes the proof.

In conclusion, the solution to our fastest path planning problem is defined by the solution of our modified level set equation (4) up to time $T(\mathbf{y})$ and by the corresponding headings along a path reaching $\mathbf{y}$ at that time, starting at $\mathbf{x} = \mathbf{0}$ and always remaining normal to the evolving modified level set from $t = 0$ to $t = T(\mathbf{y})$. We note that there are several corollaries and remarks to this theorem, considering variable vehicle speeds, specific flow field properties, multiple arrival times and feasibility of the paths. For an example of the latter, in certain flow fields, some end points can not be reached in finite time and our method would provide this conclusion. All of these corollaries and remarks, as well as relations with classic control theory, have been studied [23] and will be reported elsewhere. Specific relationships with Hamilton-Jacobi-Bellman equations and applications to planning in 3D-space time-dependent currents are developed in [23], [24]. Next, we present a computational algorithm for path planning using our modified level set equation.

IV. Algorithm and Numerical Implementation

A. Algorithm

We now outline an algorithm for path planning in time dependent flow fields using the level set method. The algorithm has the following two steps:

1. *Forward Level Set Evolution*: We evolve a wavefront normal to itself in the dynamic flow field, from the starting point of the vehicle ($\mathbf{x_s} = \mathbf{0}$) and track the evolution of this wavefront until the first time it reaches the goal ($\mathbf{x_f}$). For a given external flow field, this wavefront is a set of points which are 'furthest away' a vehicle can reach at the current time, from $\mathbf{x_s}$. The evolution of the wavefront is governed by the level set equation (4) with the initial conditions (5). Initially, the zero level set is a point at the origin. To resolve this numerically, we can assume the initial level set to be a circle centered at the origin, with radius approximately equal to the size of the grid and account for this initial motion analytically.

2. *Backward Particle Tracking*: Once the wavefront reaches the goal, the rest of the algorithm tries to identify which points on the intermediate wavefronts correspond to the path which terminates at the destination, in minimum time. In other words, we 'track' the path of the vehicle by solving particle tracking equation (6) backward in time starting from $\mathbf{x} = \mathbf{x_f}$ at $t = 0$ to $\mathbf{x} = \mathbf{0}$ at $t = T(\mathbf{x_f})$. The trajectory of the particle obtained by solving (6) is the time optimal path of the vehicle. As concluded earlier, the normals to the zero level set at the location of the particle, $\hat{\mathbf{n}}(\mathbf{x}, t)$ are the optimal heading directions of the vehicle at every time t. Next, we discuss the details of the numerical implementation of the algorithm.

B. Numerical Schemes

Forward Level Set Evolution: We discretize (4) in time using a fractional-step method as follows:

$$\frac{\phi^\star - \phi(\mathbf{x}, t)}{\Delta t/2} = -F|\nabla\phi(\mathbf{x}, t)| \tag{10}$$

$$\frac{\phi^{\star\star} - \phi^\star}{\Delta t} = -\mathbf{V}\left(\mathbf{x}, t + \frac{\Delta t}{2}\right) \cdot \nabla\phi^\star \tag{11}$$

$$\frac{\phi(\mathbf{x}, t + \Delta t) - \phi^{\star\star}}{\Delta t/2} = -F|\nabla\phi^{\star\star}|, \tag{12}$$

which can also be written as

$$\begin{aligned} \frac{\phi(\mathbf{x}, t + \Delta t) - \phi(\mathbf{x}, t)}{\Delta t} = &- F\frac{|\nabla\phi(\mathbf{x}, t)| + |\nabla\phi^{\star\star}|}{2} \\ &- \mathbf{V}\left(\mathbf{x}, t + \frac{\Delta t}{2}\right) \cdot \nabla\phi^\star \end{aligned} \tag{13}$$

(10) and (12) are spatially discretized using a first-order upwind scheme. (11) is solved using a second-order Total Variation Diminishing (TVD) advection scheme, on a staggered C-grid [25]. At every time level, the zero-level set is extracted as a set of discrete points, $\mathbf{p}_\phi(t)$ using a contour algorithm. At each of these points, the velocity is also saved, to be used in the backward calculation for the optimal path.

Backward Particle Tracking: The discrete zero-level set points, $\mathbf{p}_\phi(t)$, extracted from the forward calculation at each time step, form a piece-wise linear contour of the maximum reachable set at time t. Since we only have information on these contours, the first step is to ensure that the starting point is on the present contour. Thus, first we calculate $\hat{\mathbf{x}}_\mathbf{f}$, the minimum projection of $\mathbf{x_f}$ onto the piece-wise linear level set contour at time $T(\mathbf{x}_f)$ (i.e. find the closest point on the piece-wise linear contour to $\mathbf{x_f}$). Starting at this point, the backward path from these contours are calculated by discretizing (6) as:

$$\frac{\mathbf{x}(t-\Delta t)-\mathbf{x}(t)}{\Delta t} = -\mathbf{V}(\mathbf{x},t) - F\frac{\nabla\phi(\mathbf{x},t)}{|\nabla\phi(\mathbf{x},t)|}. \tag{14}$$

However, due to the discrete nature of the extracted curves, each new vehicle location will not fall exactly on our piece-wise linear level-set representation, and we will have to perform the minimum projection again. Our solution method is then as follows:

1) Find $\hat{\mathbf{x}}(t)$, the minimum projection of $\mathbf{x}(t)$ on to the piece-wise linear level set contour at time t.
2) Calculate the outward pointing, weighted normal, $\hat{\mathbf{n}}_w$, at $\hat{\mathbf{x}}(t)$ on the level set contour formed by $\mathbf{p}_\phi(t)$. The normal is weighted as follows: if $\hat{\mathbf{x}}(t)$ is at the midpoint of the line joining two points, then $\hat{\mathbf{n}}_w$ is exactly perpendicular to this line; whereas if $\hat{\mathbf{x}}(t)$ is exactly on a point in $\mathbf{p}_\phi(t)$, then $\hat{\mathbf{n}}_w$ is the average of the normals of the two lines originating from that point.
3) Calculate $\mathbf{x}(t-\Delta t) = \hat{\mathbf{x}}(t) - \Delta t\,[F\hat{\mathbf{n}}_w + \mathbf{V}(\hat{\mathbf{x}}(t),t)]$.
4) Repeat steps 1-3 until the final saved level set is reached.

C. Discussion

(i) In our present scheme, the time discretization of the forward and backward schemes is not consistent. While the error will be small $\mathcal{O}(\Delta t)$, accuracy would be improved if, for example, a fully implicit scheme is used in the forward equations, and the present scheme (14) is kept for the backward calculation.

(ii) An alternative approach to extracting the zero-level set contour at each time is to, instead, save the crossing times at the grid locations. This approach has a clear memory/storage advantage in three dimensions and we have implemented it for real ocean fields [23], [24]. However it complicates the data-structure if $|\mathbf{V}(\mathbf{x},t)| > F$ since multiple crossing times are stored at some grid points.

(iii) (4) can be solved efficiently by using narrow band level sets [19]. This scheme has been implemented [23] and its details and results will be reported elsewhere.

(iv) The worst case computational cost of this algorithm is of the order of the number of spatial grid points used. This is a significant improvement over using a network based approach for path planning in time dependent flows where the worst case computational cost is exponential with the number of grid points.

V. Results

In this section, we illustrate the performance of our path planning algorithm by means of a few simulations. In the first example, we consider a relatively simple flow field, a uniform jet. To illustrate the optimality of the algorithm, we compare the results obtained from the algorithm to those obtained from formulating the problem as a nonlinear optimization problem. In the second example, we apply the algorithm to swarm formation in more complex time varying 2D flow fields, which include jets and eddies. In the third example, we add constraints to the planning, specifically, vehicles are prevented from entering certain regions of the flow. For other examples with varied flow conditions and vehicle scenarios, we refer to [23], [24].

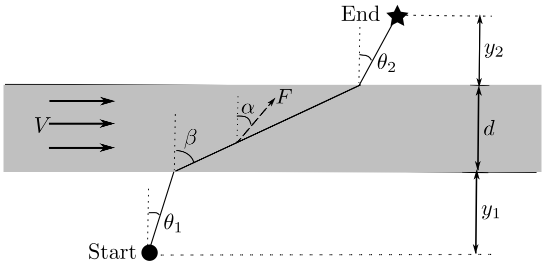


Fig. 3. Parameters involved in optimal crossing of a jet flow: jet speed V and width d; Start (Circle), End (Star), distances from jet y_1, y_2; vehicle speed F and headings $\theta_1, \theta_2, \alpha$; resultant trajectory β.

A. Optimal Crossing of a Jet Flow

Consider a flow field (see Fig. 3) in the form of a uniform jet, from left to right, of constant velocity V. This region of the flow field is shaded in Fig. 3. There is no flow in the rest of the domain. We wish to determine the minimum time path of the vehicle from the starting location (marked by a filled circle) to the goal (marked by a star). Consistent with our earlier notation, we denote the nominal speed of the vehicle with respect to the flow as F. Let us denote the vehicle heading angles before reaching the flow field, in the flow field, and after exiting the flow region respectively as θ_1, α and θ_2. While the vehicle advances in the jet, it is also advected due to the flow. Therefore, the actual direction of vehicle motion is different from the heading. Let us denote this angle by β. Notations for various distance parameters in the problem can be read off from Fig. 3.

As per our algorithm, we propagate a wavefront from the starting position of the vehicle according to the level set equation (4) until the zero level set reaches the goal. Then, we solve (6) backward in time to calculate the optimal trajectory of the vehicle. Fig. 4 shows the shapes of the level sets for this example. The level sets are primarily radial expansions outside of the jet and advected to the right in the jet. By continuity, this advection elongates the level sets outside of the jet on the downstream side. As the desired goal is downstream to the jet, the vehicle must make use of this favorable current in order to reach its destination. The vehicle path computed by our algorithm is shown by discrete points on intermediate level sets in Fig. 4.

Validation: In order to verify the results of the algorithm, we formulate this problem as a nonlinear optimization problem. The constraints of this optimization problem are obtained as follows. Let U denote the speed of the vehicle in the flow, as seen by a ground observer. We have,

$$U_x = F\sin\alpha + V \tag{15}$$

and

$$U_y = F\cos\alpha \tag{16}$$

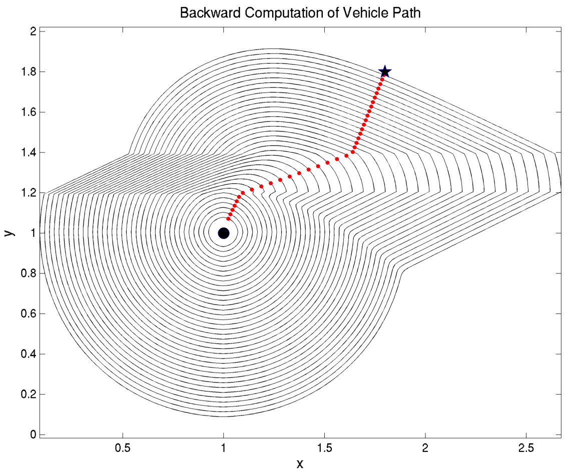


Fig. 4. Level sets and time-optimal path for jet flow in Fig. (3)

TABLE I

Comparison of Results of Level Set Algorithm and Nonlinear Optimization Method

	Level Set Method	Optimization Method
θ_1	22.68°	22.66°
θ_2	22.68°	22.66°
β	70.06°	69.99°
α	45.90°	45.77°
T	0.936	0.937

where U_x and U_y are the x and y components of the total vehicle velocity, $\mathbf{U}$. This gives,

$$\tan\beta = \frac{U_x}{U_y} = \tan\alpha + \frac{V}{F}\sec\alpha \tag{17}$$

Let X be the total downstream displacement of the vehicle, i.e. in the x direction. We have, from trigonometry:

$$X = y_1\tan\theta_1 + d\tan\beta + y_2\tan\theta_2 \tag{18}$$

Finally, the total travel time T can be written as the sum of travel times in each individual regions. Hence, the optimization problem we wish to solve is:

$$\min T = \frac{y_1}{F\cos\theta_1} + \frac{d}{F\cos\alpha} + \frac{y_2}{F\cos\theta_2} \tag{19}$$

$$\text{s.t. } X = y_1\tan\theta_1 + d\left(\tan\alpha + \frac{V}{F}\sec\alpha\right) + y_2\tan\theta_2 \tag{20}$$

and

$$\theta_1, \theta_2, \alpha \geq 0$$

This optimization problem is solved numerically in MATLAB®. We present the results for $F = 1$, $V = 1.2$, $y_1 = 0.2$, $y_2 = 0.4$, $d = 0.2$ and $X = 0.8$ in Table I. From this, we find that the two methods yield the same solution, up to very small differences. These differences in the angles are due to small numerical and truncation errors in the level set computation (e.g. limited grid resolution and so limited angles precision).

B. Sudden Expansion in Coastal Ocean and Fluid Flows

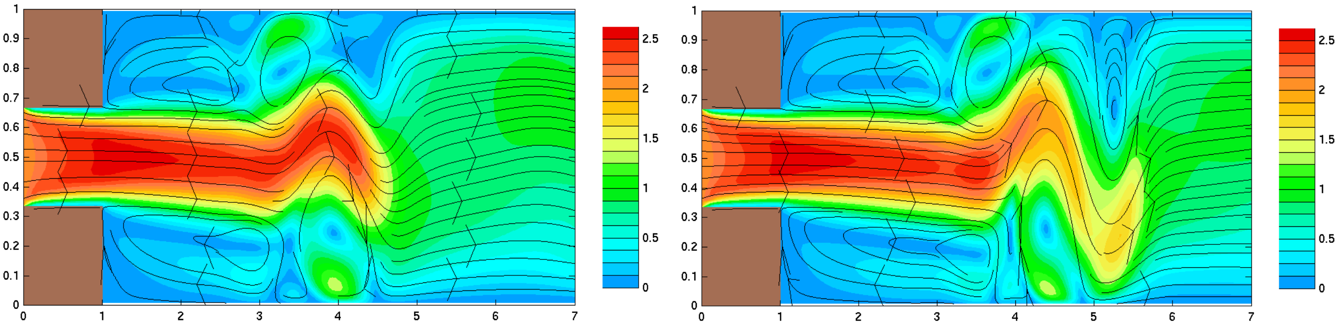
We now apply our path planning algorithm to a more realistic ocean flow field with dynamic jets and eddies, and discuss the results. We consider a uniform barotropic jet (2D flow in the horizontal plane) exiting a strait or estuary. Such flows commonly occur in the coastal ocean and generally lead to meanders and vortices as the jet exits the constriction. This situation corresponds to a highly unsteady flow field. If the width of the constriction is small enough, effects of the earth's rotation (Coriolis acceleration) can be neglected. We refer the reader to Fig. 5 for two snapshots of the flow field at two successive non-dimensional times. Shown are flow streamlines overlaid on the magnitude (in color) of the velocity field. In what follows, we will consider the scenario of a swarm of underwater vehicles that are released from a fixed point near the exit of the strait or estuary (this start point could correspond to a harbor or larger platform such as a ship or oil rig). The goal for the swarm is to reach a predetermined formation in the open ocean in fastest time, optimally using (or avoiding) the multiscale flow structures as they occur along the way. The formation can for example be selected based on security, surveillance, pollution monitoring or ocean sampling considerations. In all cases, our methodology will compute the optimal heading time-series for each vehicle based on our predicted time-dependent flow field. The computational cost is overall proportional to the geometric dimensions of the formation pattern.

Another setting where this example can be useful is for the monitoring of the flow in a pipe or channel which encounters a sudden increase in cross sectional area. In that situation, our example would illustrate how mobile sensors released at the junction would have to be navigated to reach a specific formation in fastest time. Such a formation could then be designed to monitor possible pressure drops, release of toxic material, status of pipe wall conditions or other properties.

In this example, we set the speed of the vehicles in still water to $F = 0.5$. The maximum speed $U_{\max}$ of the flow is 2.5 (see Fig. 5). The width of the inlet is one third of the total width of the channel. The Reynolds number is Re$=(\frac{h}{2})\frac{U_{\max}}{\nu} = 417$ with $h = \frac{1}{3}$ and $\nu = 10^{-3}$. In our scenario, we enforce that the swarm of vehicles take a triangle shape at final time, as shown in Fig. 6(a). The vehicles are released at the lower edge of the inlet. Fig. 6(a) shows the optimal paths of the vehicles computed using our path planning algorithm. From Fig. 5(b), we can intuitively see that to reach the tip of the triangle in shortest time, the vehicle must ride along a favorable current. For the four end-points that are closest to the inlet, the vehicles clearly utilize the upper and lower re-circulation eddies. Overall, we find from Fig. 6(a) that the algorithm correctly predicts the shapes of the optimal paths.

C. Ocean flows with forbidden regions

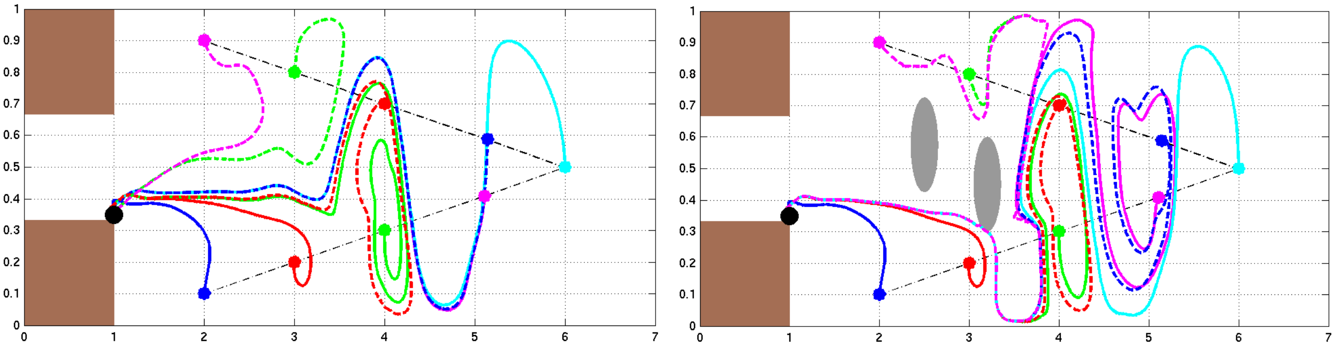
We now consider the situation where the swarm of vehicles cannot enter specific regions, either because of safety, hazardous conditions, security or naval considerations. We



(a)

(b)

Fig. 5. Snapshots of the flow field for a jet exiting a strait or estuary (sudden expansion/2D coastal flow) showing color maps of the total magnitude of the flow velocity overlaid with streamlines (a) at the time of initial vehicle deployment and (b) near the final time of vehicle maneuvers in Fig. 6(a).



(a)

(b)

Fig. 6. Optimal vehicle paths for 9 vehicles deployed from a single point (black dot) in the flow illustrated by Fig. 5. Results for two situations are shown: (a) No constraints or forbidden regions: Vehicle paths then take full advantage of evolving jets and eddies to reach their final positions (colored dots) in shortest time. (b) Two forbidden regions: Vehicles are denied access to the gray shaded regions. Our algorithm provides seven new time optimal paths for the paths computed in (a) that are blocked while it correctly leaves unchanged the two paths that are not blocked.

refer to these regions as forbidden regions because they cannot be entered by vehicles but they have no effect on flow fields, i.e. currents are not affected by them. To implement the forbidden regions in the forward calculation we replace the right-hand sides of (10-12) with zero in the forbidden regions. In the backward calculation, we only need to mask $\mathbf{V}(\mathbf{x}, t)$ in (14) with zeros in the forbidden regions since the level sets evolved from the modified forward algorithm have normals that correctly go around the forbidden regions. Handling such obstacles is thus straightforward, which is a major advantage of our approach.

Path planning with forbidden regions is illustrated on Fig. 6(b). The physical setup is as in Fig. 6(a) but we prevent the paths from entering the two regions shown in gray. Collectively, these two regions block seven of the nine optimal paths of Fig. 6(a). The new optimal paths for these seven vehicles all ride the lower edge of the main jet, just skirting the bottom of the second forbidden region. They then ride down one eddy and up an adjoining eddy, Fig. 5(a), to rejoin the main jet behind the forbidden regions. The two paths from Fig. 6(a) that did not pass through the forbidden areas remain unaffected by the forbidden areas.

VI. Conclusions and Future Work

In this paper, we derived and described a new methodology for time optimal path planning of autonomous vehicles navigating through a time varying flow field. The methodology uses rigorous partial differential equations and obviates the need for heuristic or ad-hoc assumptions. It employs the level set method to evolve a wavefront from the starting point of the vehicle until it reaches the desired goal location, combining advection by the variable flow with the vehicle nominal speed. The optimal path of the vehicle is then obtained by solving a particle tracking equation backward in time. In some sense, our approach combines fluid dynamic equation ideas with control theory and path planning schemes. It also rigorously and easily accommodates for forbidden regions that only affect vehicles and for real obstacles that affect both the flow and the vehicles, but can also handle other fluid-like constraints. Importantly, the computational cost varies linearly with the number of vehicles and geometrically with the spatial geometric dimensions of the ocean domain (and not exponentially). Of course, we focused mostly on the planning of underwater vehicles but results apply to any other flows and vehicle sizes, from airplanes or other air-vehicles in large-scale atmospheric flows to small robotic devices in quasi-microscopic flows.

Although we have illustrated only a few examples in this paper, the present methodology is quite versatile and we have applied it to many idealized and realistic situations, including cases with coordinated time-dependent formation plans and complex coastlines [23], [24]. This latter application of coordinated motion of autonomous vehicles has been extensively developed recently. A first possible future direction would indeed be to combine our methodologies with the results of [26], [27], [28], [31], [32] on efficient coordinated motions. Specifically, using our path planning algorithm to coordinate the motions of multiple vehicles only leads to a linear increase in our computational cost. Secondly, the problem of avoiding physical obstacles that are time-dependent is also important for the safety of vehicles. This can be easily incorporated into our algorithm by imposing certain boundary conditions around the time-dependent obstacles while evolving the level sets forward in time. Of course, time-dependent obstacles that alter the flow field itself (such as ships) are also easily accounted for. Thirdly, we have also started to investigate the effects of uncertain flow field predictions [17], [18] on our path planning problem. This is important since forecast flow fields are uncertain, especially in oceanic applications. Finally, another extension is to update the plan along the path using onboard routing [29], [30].

VII. ACKNOWLEDGMENTS

We are very thankful to the members of the MSEAS group at MIT, in particular to Mr. T. Sondergaard for his inputs and many intellectually stimulating discussions related to this work. We are grateful to the Office of Naval Research for support under grant N00014-09-1-0676 (Science of Autonomy – A-MISSION) and especially thank Dr. M. Steinberg as well as Drs. T. Paluszkiewicz and S. Harper. PFJL is also thankful to SeaGrant at MIT for the Associate Doherty Professorship award.

REFERENCES

[1] A. Alvarez, A. Caiti and R. Onken,"Evolutionary path planning for autonomous underwater vehicles in a variable ocean", *IEEE Journal Of Oceanic Engineering*, Vol. 29, No. 2, 2004, pp 419-429.

[2] N. K. Yilmaz, C. Evangelinos, P. F. J. Lermusiaux and N. A. Patrikalakis, "Path Planning of Autonomous Underwater Vehicles for Adaptive Sampling Using Mixed Integer Linear Programming", *IEEE International Conference on Robotics and Automation (ICRA)*, Vol. 33, No. 4, 2008, pp 522-537.

[3] P.F.J. Lermusiaux, "Adaptive Modeling, Adaptive Data Assimilation and Adaptive Sampling", *Special issue on Mathematical Issues and Challenges in Data Assimilation for Geophysical Systems: Interdisciplinary Perspectives, Physica D*, Vol. 230, 2007, pp 172-196.

[4] B. Garau, A. Alvarez, G. Oliver, "Path Planning of Autonomous Underwater Vehicles in Current Fields with Complex Spatial Variability: an A* Approach", *IEEE International Conference on Robotics and Automation (ICRA)*, 18-22 April 2005, pp 194-198.

[5] J. J. Kuffner, and S. M. LaValle, "RRT-connect: An efficient approach to single-query path planning", *Proceedings of the IEEE International Conference on Robotics and Automation*, 2000.

[6] C. S. Tan, R. Sutton and J. Chudley, "An Incremental Stochastic Motion Planning Technique for Autonomous Underwater Vehicles", *Proceedings of IFAC Control Applications in Marine Systems Conference*, July 2004, pp 483-488.

[7] C. W. Warren, "A technique for autonomous underwater vehicle route planning", *Proceedings of the Symposium on Autonomous Underwater Vehicle Technology*, 5-6 June 1990, pp 201-205.

[8] R. E. Davis, N. E. Leonard, D. M. Fratantoni, "Routing strategies for underwater gliders", *Deep Sea Research Part II: Topical Studies in Oceanography*, Vol. 56, Issues 3-5, February 2009, pp 173-187.

[9] J. N. Tsitsiklis, "Efficient algorithms for globally optimal trajectories", *IEEE Trans. Autom. Control*, Vol. 40, No. 9, 1995, pp 1528 -1538.

[10] R. A. Jarvis and J. C. Byrne, "Robot Navigation: Touching, Seeing and Knowing", *Proceedings of the 1st Australian Conference on Artificial Intelligence*, November 1986.

[11] J. Barraquand and J.C. Latombe, "Robot Motion Planning: A Distributed Representation Approach", *International Journal of Robotics Research*, Vol. 10(6), 1991, pp. 628-649.

[12] R. A. Jarvis, "Robot path planning: complexity, flexibility and application scope", *International Symposium on Practical Cognitive Agents and Robots (PCAR '06)*. NY, USA, 2006, pp. 3-14.

[13] C. Petres, Y. Pailhas, P. Patron, Y. Petillot, J. Evans, D. Lane, "Path Planning for Autonomous Underwater Vehicles", *IEEE Trans. Robot.*, Vol.23, No.2, April 2007, pp. 331-341.

[14] M. Soulignac, P. Taillibert, M. Rueher, "Time-minimal path planning in dynamic current fields", *IEEE International Conference on Robotics and Automation*, 12-17 May 2009, pp. 2473-2479.

[15] M. Soulignac, "Feasible and Optimal Path Planning in Strong Current Fields", *IEEE Trans. Robot.*, Vol.27, No.1, February 2011, pp. 89-98.

[16] B. Rhoads, I. Mezic, A. Poje, "Minimum time feedback control of autonomous underwater vehicles", *Proceedings of the 49th IEEE Conference on Decision and Control, CDC 2010*, Dec. 15-17, 2010.

[17] T. P. Sapsis and P. F. J. Lermusiaux, "Dynamically orthogonal field equations for continuous stochastic dynamical systems", *Physica D*, Vol. 238, 2009, pp. 2347-2360.

[18] M. P. Ueckermann, T. P. Sapsis, and P. F. J. Lermusiaux, "Numerical Schemes for Dynamically Orthogonal Equations for Stochastic Fluid and Ocean Flows", *Journal of Computational Physics*, 2011. In prep.

[19] J. A. Sethian, "Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science" Cambridge University Press: Cambridge, U.K., 1999.

[20] S. Osher and J. A. Sethian, "Fronts Propagating with Curvature-Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations", *J. Compt. Phys.*, Vol. 79 No. 1, 1988, pp 12-49.

[21] S. Osher, R. Fedkiw, "Level Set Methods and Dynamic Implicit Surfaces", Springer Verlag, 2003.

[22] A. Agarwal and P. F. J. Lermusiaux, "Statistical Field Estimation for Complex Coastal Regions and Archipelagos", *Ocean Modeling*, Vol. 40 No. 2, 2011, pp 164-189.

[23] MSEAS Group (2011). Autonomous Marine Intelligent Swarming Systems for Interdisciplinary Observing Networks (A-MISSION). Reports in Ocean Science and Engineering 10, Department of Mechanical Engineering, MIT, Cambridge, 101 pp.

[24] K. Yigit, "Path Planning Methods for Autonomous Underwater Vehicles", SM Thesis, MIT, Dept. of Mechanical Engineering, June 2011.

[25] R. J. Leveque, "Finite Volume Methods for Hyperbolic Problems", Cambridge University Press, 2002.

[26] D. Paley, F. Zhang and N. E. Leonard, "Cooperative Control for Ocean Sampling: The Glider Coordinated Control System", *IEEE Trans. Cntr. Syst. Technol.*, Vol. 16, No. 4, 2008, pp. 735-744.

[27] F. Zhang, D. M. Fratantoni, D. Paley, J. Lund, N. E. Leonard, "Control of Coordinated Patterns for Ocean Sampling", *International Journal of Control*, Special issue on the Navigation, Guidance and Control of Uninhabited Underwater Vehicles, Vol. 80, No. 7, 2007, pp 1186-1199.

[28] N. E. Leonard et. al., "Coordinated Control of an Underwater Glider Fleet in an Adaptive Ocean Sampling Field Experiment in Monterey Bay", *Journal of Field Robotics*, Vol. 27, No. 6, 2010, pp 718-740.

[29] D. Wang, P. F. J. Lermusiaux, P. J. Haley, D. Eickstedt, W. G. Leslie and H. Schmidt, "Acoustically Focused Adaptive Sampling and On-board Routing for Marine Rapid Environmental Assessment" Special issue of the *Journal of Marine Systems* on "Coastal Processes: Challenges for Monitoring and Prediction", 2009.

[30] D. Moore et. al., "Simultaneous Local and Global State Estimation for Robotic Navigation", *International Conference on Robotics and Automation* 2009.

[31] M. Ani Hsieh, E. Forgoston, T. W. Mather and I. B. Schwartz, "Robotic Manifold Tracking of Coherent Structures in Flows", *Personal Communication*.

[32] R. W. Brockett, "On the control of a flock by a leader", *In Proceedings of the Steklov Institute of Mathematics*, Vol. 268, No. 1, 2010, pp. 49-57.