Aqua-Quad - Solar Powered, Long Endurance, Hybrid Mobile Vehicle for Persistent Surface and Underwater Reconnaissance, Part II - Onboard Intelligence

Vladimir Dobrokhodov, Kevin Jones, Chase Dillard, Isaac Kaminer
Mechanical and Aerospace Engineering dept.
Naval Postgraduate School
Monterey, California 93943
Email: vldobr@nps.edu, jones@nps.edu, chasedillard@gmail.com, kaminer@nps.edu

***Abstract*—The paper describes the control system development of a novel hybrid autonomous vehicle - Aqua-Quad, a Multi-Rotor Vertical Take Off and Landing aircraft with environmentally hardened electronics, exchangeable sensor suite, communication links, and a solar recharge system. The key objective of this multi-modal autonomous system is to enable energy-aware ultra-long endurance autonomy to facilitate near real-time capture and transition of information from the undersea domain to the air and further to the ground. The key benefit of this rapid in-situ data delivery is to enable timely and efficient decision making that, in turn, improves the collective efficiency of the Aqua-Quad as a system. Each individual vehicle is designed to have mission-relevant sensors and sufficient computational power to reduce the false alarm rate, and navigate in open seas in an energy-optimal manner by optimizing its route while either in search of targets or when tracking them. Higher operational efficiency is envisioned when a flock of Aqua-Quads operates cooperatively. The paper focuses on the design of energy-optimal path planning for a single vehicle in the presence of ocean currents. A novel modification of rapidly exploring random tree algorithm is developed to fully utilize the energy savings provided by the transportation mechanism of ocean flows.**

I. Introduction

The paper builds on the work reported in [1]–[3] that describe development of a novel hybrid autonomous vehicle - Aqua-Quad. The vehicle combines a Multi-Rotor Vertical Take Off and Landing (MR-VTOL) aircraft with environmentally hardened electronics, exchangeable sensor suite and a solar recharge system in order to provide ultra-long endurance sensing in aquatic environments in support of a variety of civilian and military applications, see the vehicle's concept in Fig.1. To achieve this capability, the flock of cooperative vehicles must not only absorb all the energy necessary to function from the environment but will also have to radically change control strategies. The control includes both a strategy to minimize energy losses onboard a single platform as well as the cooperative control to minimize losses in an "expensive battle" with ocean currents.

The vehicle is envisioned to perform sensing below the thermocline layer while silently riding on ocean currents, and flying in the air when required for rapid relocation or local communication ($12-20$ *miles*). While riding the ocean currents the platform replenishes its energy storage by utilizing solar photovoltaic (PV) energy. The vehicle instrumentation combines (i) INS/GPS and aircraft control, data processing, and command and control capabilities of a VTOL aircraft, (ii) sensing and signal processing capabilities of an underwater robot with a vector sensor suspended below the thermocline, and (iii) solar energy harvesting for energy-independent operation. The onboard software suite includes, in addition to the typical UAV flight control software, the algorithms for energy management, local and global path planning, estimation of ocean currents, signal processing of the underwater sensor data delivered over the tether, and the inter-vehicle cooperation. The novel components of the Aqua-Quad that enable long endurance operations are the energy management and the path planning algorithms. The efficiency of these algorithms significantly depends on the ability to share information with the neighboring nodes. These algorithms represent a significant shift in the control paradigm typical for aerial applications. *In particular, the system does not "reject" external disturbances of ocean currents and waves, but rather uses them to harvest free energy and transitions in the desired direction by following the ocean flows.*

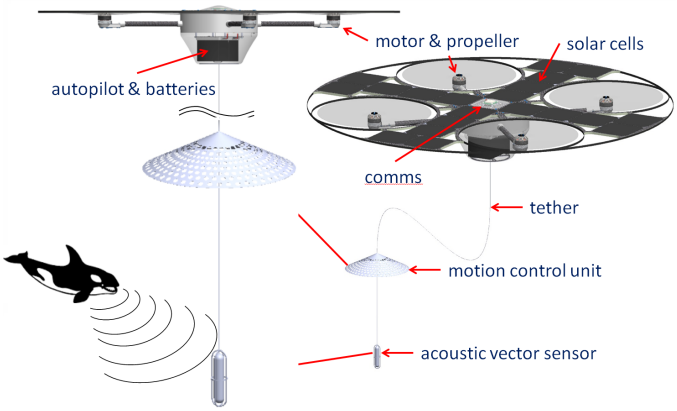
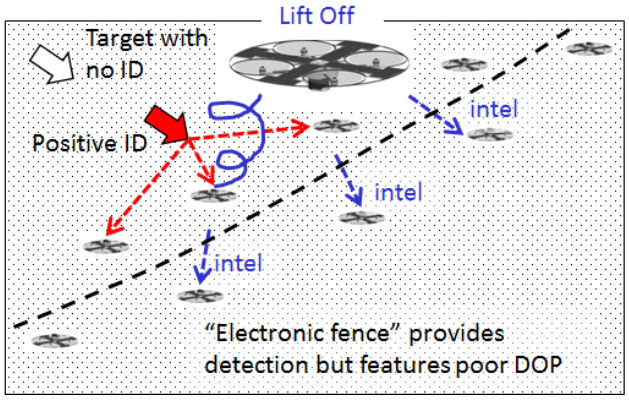


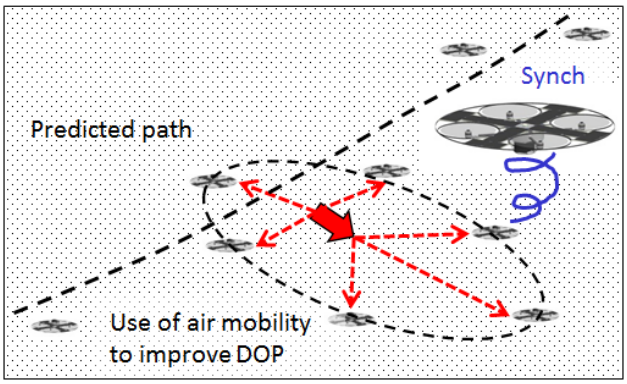
Fig. 1: Key components of Aqua-Quad system.

The Aqua-Quad is most effective when multiple cooperative vehicles are deployed in an operational area with heterogeneous sensors. For example, with acoustic and magnetic

sensors the flock of Aqua-Quads might form an electronic fence in an operational area (see Fig.2). While silently sensing the undersurface environment, the platforms automatically distribute themselves to guarantee the required node-density of the protective "electronic fence". When a target is detected by one node, the individual onboard signal processing capability and the heterogeneity of sensors on other nodes allows for a lower false alarm rate, while the higher mobility afforded by the airborne mode provides rapid coordinated reaction of the team to enable superior target classification, higher precision of motion estimation, and tracking capabilities. It is envisioned that the extended endurance and quiet operation of the multi-modal Aqua-Quads, which are capable of on-demand communication, higher mobility, and cooperative reaction to potential "threats" could provide higher operational efficiency of the system in undersurface surveillance and reconnaissance operations than currently employed systems [4] (e.g., Sonobuoy, WaveGlider).



(a) Distributed fence



(b) Adopting shape

Fig. 2: The concept of "electronic fence."

The energy in a maritime domain is abundant in various forms, however a typical feedback control system would treat ambient energy as an adverse disturbance. The "disturbance rejection" may result in prohibitively expensive energy cost for the mission. We take a new perspective on this challenge, one that exploits environmental disturbances and incorporates them into the control and planning process while pursuing overall mission objectives. We focus on minimization of energy expenditure via computationally feasible optimal path planning that utilizes a priori given or in-situ measured ocean currents.

It is envisioned that the mission planning and execution for a flock of Aqua-Quads will be performed at two levels. At the *global* mission planning level the routes are designed to achieve the objective of a specific maritime mission and account for persistent seasonal ocean currents and obstacles at a given area of operation. This step will necessarily solve a number of global path planning tasks that also account for the operations research aspects of a particular application; they include the required number and type of sensors onboard the Aqua-Quads. This effort might require the capabilities of a super computing facilities available on-shore. The outcome of a global planner can be conveniently parameterized by a set of waypoints (WP), which can represent the "ends" of the "persistent" but time-varying segments of the geometrical mission boundary. At the *local* path planning and path following phase the onboard algorithms utilize the results of the global planner and approach the planning task from the perspective of a two point boundary value problem. At this phase the current and the desired position of each vehicle are defined, and the onboard solver finds the energy-optimal path that satisfies the energy constrains of the Aqua-Quad platform while accounting for local variability of ocean currents, unexpected obstacles, and target detection events. Overall, the two-layer approach to planning and execution of collaborative Aqua-Quad missions promises higher robustness to unforseen events in the presence of tight constraints of the onboard energy management system.

This paper focuses on presenting the software algorithms that enable ultra-long operations of this novel hybrid platform. This material follows Part I, which introduces the Aqua-Quad engineering design concept and discusses the feasibility of its hardware design. The primary focus here is on the algorithms of energy efficient mission planning in an aquatic environment. First we present the equations of total energy balance used to evaluate the cost function. The cost function also includes components that are responsible for optimal relative positioning of the vehicle with respect to underwater objects of interest. The total cost is used as a metric of energy-optimal path planning. Next we present a "dead-reckoning" (*DR*) modification of the rapidly exploring random tree (*RRT**) algorithm that accounts for the drifting mobility of the Aqua-Quad. Evaluation of the computational efficiency of this new *DR-RRT** modification is given special consideration. The paper ends with a discussion of the developed functional prototype and energy metrics evaluated in the initial open water experimentation.

II. Path Planning

Optimal motion planning is a fundamental problem in all areas of robotics. In application to the optimal path planning and control of marine vehicles (both surface and underwater), the key optimization criteria are time [5], energy [6]–[8], the spatial placement [9], [10], and the combinations these criteria [11], [12] that allow for better representation of the nature of a specific mission. Most realistic settings attempted to account for spatial and temporal variability of natural flows [6], [13]. Analysis of these works shows that a closed form optimal control solution exists only for an overly-simplified description of the vehicle dynamics and the flow fields; see for example a solution of Zermelo's navigation problem [14], [15]. Most methods of optimal path planning either fail when the environment becomes too complex and

dynamic, or computationally infeasible for real-time applications. As a result, new classes of evolutionary and random sample-based optimization algorithms have been developed that produce suboptimal solutions suitable for implementation by onboard modern microcontrollers.

The Aqua-Quad project adopts the (*RRT**) algorithm [16] and adds functionality to account for the unique hybrid mobility of the vehicle operating in a realistic environment described by a vector flow field with obstacles. The original *RRT* [17] is a sampling-based algorithm that quickly explores a given operational space and develops a feasible path between the initial and goal states that is free of obstructions. The algorithm randomly samples points in the state space and finds obstacle-free connections (edges - E) to them from existing points (vertices - V). The *RRT** algorithms modifies the *RRT* with an intermediate "re-wiring" step that, on one hand, preserves the probabilistic completeness of the search and, on the other hand, provides optimality [16] of the resulting solution while still keeping the algorithm computationally feasible for onboard microcontroller implementation.

A. Kinematics

Let $\mathbf{p}$ define a position vector in bounded and connected Cartesian space Ω, see Fig.3, where obstacles in Ω_{obs} are known, and thus obstacle-free space $\Omega_{free} = \Omega \backslash \Omega_{obs}$ is well-defined at all time. Consider the motion of a vehicle in a known position-dependent time-varying flow field given by $W(\mathbf{p}, t)$ with $u = u(x, y, t)$ and $v = v(x, y, t)$ being the x, y components of velocity W. Let $\mathbf{p}_0$ and $\mathbf{p}_f$ denote the position vector at the initial and desired final locations respectively. Let the speed of the vehicle, V, with respect to the velocity field be constant; it is zero $V = 0$ when the vehicle is in drifting mode, and it is constant $V \geq 0$ - when in flight mode. Transition between the modes is controlled by a switching function $\delta(t)$ that equals 0 in drifting and 1 in flight mode. Assume that the vector field $W(\mathbf{p}, t)$ is either known a priory or can be "rapidly" estimated by the multiple vehicles locally and shared across the communication network, thus quickly providing the "global vector" map. The body of the Aqua-Quad is assumed not to be dynamically interacting("fighting") with the currents; following this, the geometrical characteristics of the Aqua-Quad are not essential in path planning.

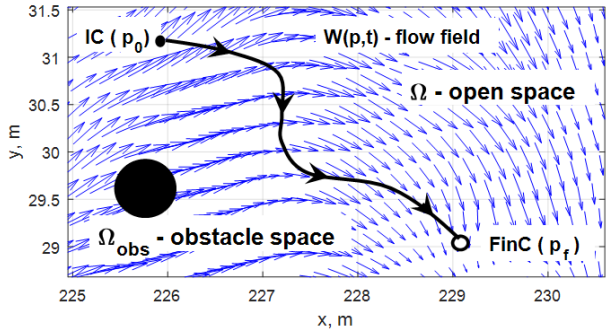


Fig. 3: Vehicle motion in flow field $W(\mathbf{p}, t)$

Using the notations above, the motion of the Aqua-Quad can be described by the following equations:

$$\begin{aligned} \dot{x} &= \delta(t) \cdot V cos(\psi(t)) + (1 - \delta(t)) \cdot u(x, y, t) \\ \dot{y} &= \delta(t) \cdot V sin(\psi(t)) + (1 - \delta(t)) \cdot v(x, y, t) \\ \delta &= \begin{cases} 0, & \text{if } drifting \\ 1, & \text{if } flying, \end{cases} \end{aligned} \tag{1}$$

where $\psi(t)$ and $\delta(t)$ are the control functions of the vehicle and $V \geq 0$ constant. Note, that $\psi(t)$ does not affect the motion when the vehicle is in drifting mode as the control authority is set to zero by $\delta(t)$. On the other hand, when in flight, we assume that there is no wind to affect the motion of the vehicle. The instance, t_{sw}, when to enable the flight mode, the commanded velocity, V, the direction, ψ, and the duration of flight, t_{fl}, are defined by the optimal route planner. It is also assumed that the vehicle is equipped with an autopilot capable of executing a *WP-following* command that autonomously transitions the vehicle to the desired WP over the shortest straight flight segment. The energy cost of this flight is directly proportional to the duration of the flight segment.

Thus, the *control task* is to optimally and safely steer each hybrid vehicle from initial state, $\mathbf{p}_0$, to a goal region, $\mathbf{p}_f$, by means of control consisting of pure drift and the heading angle, ψ, that is activated by $\delta(t)$, while minimizing some cost function, $\mathbf{J}$. The effect of flow uncertainty at the local path optimization level is beyond the scope of this work.

B. Definition of the cost function

In a typical scenario, a number of vehicles is assigned the task of maintaining a specifically shaped "protective fence" while constantly monitoring the undersurface environment. The shape of the "fence" results from solving a mission planning task that provides the entire flock of vehicles with a global path to follow. The global route can be parameterized by a finite set of WP that serve the purpose of boundary conditions (goal regions) for the path planning task of individual vehicles. In the current approach the temporal constraints of arriving at the boundary WP are not considered, as energy is the dominant constraint of the entire concept; solving the mission and path planning tasks for energy solves the task of feasibility of the entire concept. Therefore, the task of each individual vehicle is to steer along the WP-defined segments by implementing the optimal routing solutions in onboard controller. Position and intended route of each individual vehicle can be periodically shared with the neighboring vehicles to achieve the desired "density" of the "electronic fence."

The cost function of the path planning task includes the terminal and the running cost components. Since the solution adopts the random sampling approach, the optimization process includes a sequence of discrete steps. The cost function at each step includes three components. The terminal cost is defined by the *distance to a goal* region J_{tg}, while the running cost is a weighted sum of *energy expended* in a specific transition segment and the cost associated with the *quality of the target tracking* solution; we characterize this quality by a metric represented by either the dilution of precision (DOP) or the Posterior Cramer-Rao Lower Bound [9].

Until contact with an underwater object of interest is made, the vehicle is in a drifting mode ($\delta = 0$) while constantly

recharging its onboard battery using solar panels. The corresponding energy cost of this mode includes the *passive* energy consumption, J_p, of onboard electronics, communication links, and the deployed underwater sensor. The cumulative passive mode power loss is represented by the constant "burn" rate, e_p. Therefore, for the duration of a single step from t_i to t_{i+1} the total energy consumed in passive mode is $J_p = \int_{t_i}^{t_{i+1}}(e_p)dt$. While drifting in daylight the battery is *charged* by the PV system with the rate, e_{ch}, that is a function of the efficiencies of PV panels and the power conversion circuitry, as well as time of the day and the latitude at current location. Therefore, the drift time allows the energy proportional to $J_{ch} = \int_{t_i}^{t_{i+1}}(e_{ch})dt$ to accumulate. Similarly, when a path planning solution commands the vehicle to relocate via a *flight mode* to a specific position, the energy cost, J_f, of this segment depends only on the electrical "burn" rate, e_f, of the propulsion system and therefore $J_f = \int_{t_i}^{t_{i+1}}(\delta(t)\cdot e_f)dt$; it can reasonably be assumed that at a predefined altitude the magnitude of, e_f, is primarily defined by the constant commanded speed, V.

When contact with an underwater object is made and the object is identified, the target tracking might require better vehicle placement with respect to the estimated position of the target, while minimizing the energy expended for a constrained flight segment, thus ($\delta = 1$). With respect to the nonlinear estimation process, the concept of dilution of precision (DOP) provides us with a metric to explore an optimal configuration of sensors around the target; multiple cooperative vehicles are the sensing nodes and this is where optimal placement is needed. DOP defines the potential precision one can achieve in a particular measurement scenario. The project in [3] considers the models of linear target motion in $2D$ space and the nonlinear measurements of different types including the bearing only, range only, and the time-difference of arrival nonlinear models. Note that the latter is not typical in tracking scenarios and is, in fact, envisioned as a unique feature of the network-enabled Aqua-Quad, which would be capable of sharing the time-synchronized data over the network; the simplicity of measurements is the key feature of this estimator [3]. In either case the nonlinear estimation task is solved in $2D$ space providing estimates of the target position and velocity, and the DOP metric is calculated as $J_{dop} = \sqrt{\text{trace}((H^\top H)^-)}$, where H is the Jacobian of the nonlinear measurement matrix. The specific noise model and the corresponding covariance are defined by the specifications of mission sensors; the project uses the Acusonde [18] sensor. More details on the capabilities and quality of data provided by Acusonde, and comparison of different types of nonlinear estimation algorithms in application to the same classes of measurements can be found in [3].

An example of the energy budget of the Aqua-Quad, based upon notional design, battery capacity, and the 24 *hour* solar power average estimated at the latitude of Paso Robles, CA in June is presented in Tab. I. The energy expended by the passive, sensing, and communication subsystems are combined together in the *Passive* row, see the corresponding details in Part I [1] and in [3]. It can be seen that with the limited battery capacity C_{batt} the energy management is a very challenging task that closely connects the optimal design of the vehicle and its operational use. Formally, the energy balance equation of the Aqua-Quad is represented by $C_{batt}+J_{ch}-J_p-J_{fl} \geq 0$ and the battery choice condition is $C_{batt} \gg J_p$. Note the positive

TABLE I: Energy budget of a prototype Aqua-Quad

	Power	Time	Energy
Passive	5 W	24 hrs	120 Wh
Flight	340 W	1.1 hrs	470 Wh
Solar	-	24 hrs	590 Wh
Battery	-	-	178 Wh

sign associated with the solar energy term J_{ch}. This is done to account for the energy gain associated with drifting during the daylight hours.

C. Statement of the path planning task

The formulation of the optimal path planning task adopts the terminology of the seminal work [16] that introduced the *RRT** algorithm. In what follows, we present the formal statement of the task and describe the modification of the *RRT** algorithm that accounts for the drifting mobility as the prevailing operational mode of the Aqua-Quad.

Optimal planning task: Given a vehicle with kinematics (1), a bounded connected open set, Ω, an obstacle space, Ω_{obs}, an initial state, $\mathbf{p}_0$, and a goal region, $\mathbf{p}_f$, find a path, $\sigma^* : [0, s] \rightarrow \Omega\backslash\Omega_{obs}$, such that (i) $\sigma^*(0) = \mathbf{p}_0$ and $\sigma^*(s) \in \mathbf{p}_f$ and (ii) $J(\sigma^*) = \min_{\sigma\in\Omega\backslash\Omega_{obs}} J(\sigma)$ with cost $J(\sigma)$ defined as follows:

$$J(\sigma) = \sum_i \lambda_{tg}\cdot J_{tg} + \sum_i \lambda_{dop}\cdot J_{dop} + \sum_i \frac{\lambda_p\cdot J_p + \lambda_{fl}\cdot J_{fl} - \lambda_{ch}\cdot J_{ch}}{C_{batt}} \quad (2)$$

s.t.

$$C_{batt} - J_p - J_{fl} + J_{ch} \gg 0, \forall t$$
$$\sum \lambda_{tg} + \lambda_{dop} + \lambda_p + \lambda_{fl} + \lambda_{ch} = 1, \forall i,$$

where i- is the step of the optimization task, λ's - are the weights of the corresponding elementary costs, and C_{batt} is the maximum energy capacity of the onboard battery.

D. Baseline RRT* algorithm and its DR modification

The *RRT** approach is very similar to most other incremental sampling-based planning algorithms. See for example the in-depth discussion of the probabilistic road map (PRM) [19], [20], RRT, RRG algorithms in [16], [17]. These algorithms typically include the following simple and computationally inexpensive procedures performed either on the *tree* or *graph* formulations of the optimization task: *Sampling*, *Steering*, *Nearest Neighbor*, *Near Vertices*, *Collision Test*. To better represent the modification introduced in this work we briefly explain the function of each primitive and refer an interested reader to the original description of the *RRT** algorithm in [16].

The function *Sample* returns independent identically distributed samples $\mathbf{p}_{rand}$ from Ω_{free}. The function *Near Vertices* returns all neighboring vertices of graph $G = (V; E)$ within the closed ball of some predefined radius centered at random state $\mathbf{p}_{rand}$. The function *Nearest Neighbor*, for a given graph $G = (V; E)$ consisting of vertexes V and edges E, returns a new vertex V' that is "the closest" to random state $\mathbf{p}_{rand}$ in terms of a given cost J. The *Collision Test* identifies if the

line segment between two states $\mathbf{p}$ and $\mathbf{p}'$ both from Ω_{free} lies entirely in Ω_{free}. Finally, the function *Steer* transitions to the new state within the time step Δt allowed; in kinodynamic planning the function also returns the corresponding control function. It is worth noting that most of the numerous modifications of the original *RRT** algorithm differ in the step that *extends* the underlying graph that implements the search of optimal path over *the graph* or *the tree*.

The logic of *RRT** starts at the initialization phase where the graph is initialized with a single vertex (V) at the current location and no edges; a simplified flow of the algorithm is shown in the right portion of Fig. 4. Then, a graph is incrementally grown on Ω_{free} by randomly sampling a state p_{rand} from Ω_{free} and *extending* the graph towards p_{rand}. Each *extension* step includes the procedures that verify whether the proposed extension is collision free and calculates the cost of that extension. Every such sampling step followed by extensions represents a single iteration of the incremental sampling-based algorithm. Every new vertex V represents a state and the edge E defines a transition to the vertex with the associated cost.

To better represent the *DR-RRT** modification introduced by this work we refer to Fig. 4 that depicts the difference with respect to the original *RRT** algorithm; the intent is not to duplicate the original algorithm but to illustrate the connection of the *DR* modification to the *RRT** logic. The logic behind the *DR* modification is to employ the power of *RRT** only if there is a need for "powered" repositioning either to jump to a lower energy path (since the map of currents is given), or if there is need to improve DOP, or when there is an obstacle. Depending on the implementation, the formal notation of the original $Extend_{RRT}$ algorithm in [16] can be changed to include the "switching nature" of the cost function; this can be done where the cost J' of the *proposed* obstacle-free extension is calculated:

$$J' \leftarrow J(\mathbf{p}_{near}) + J(\text{Line}(\mathbf{p}_{near}, \mathbf{p}_{new}))$$

where the Line$(\cdot)$ operator denotes a straight continuous path between two points $\mathbf{p}_{near}, \mathbf{p}_{new}$, see more details in [16].

The cost, J, that drives the *DR-RRT** algorithm, aims to use oceanic currents to the maximum extent possible. The resulting solution is expected to feature extended drifting periods during which we predict the future state of the vehicle given the known ocean current flow $W(\mathbf{p}, t)$ at the currently analyzed vertex of the tree. This is a basic "dead reckoning" (DR) step and, like most DR solutions, it is an estimate. In our case, its accuracy depends on the uncertainty of the ocean current map and the duration of the drifting time. As such, the DR step is integral to our algorithm, as the drift paths that are created can be linked via flight with hop steps. The power behind the *DR-RRT** algorithm is that it allows for periods where no control input occurs and hence minimal energy is expended. The bulk of the *RRT** process is dormant during these phases. Moreover, since the DR modification does not add any new "primitives", but modifies the calculation of analytical cost in the "extend step", the result holds the same asymptotic computational complexity.

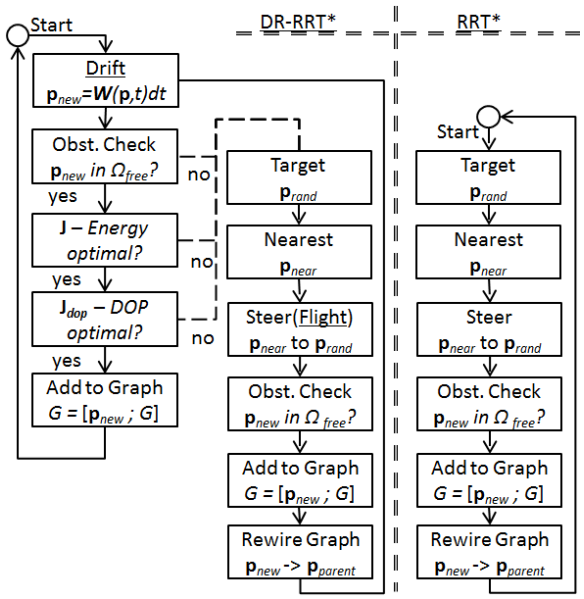


Fig. 4: Dead-reckoning modification (left) of the original *RRT** algorithm (right)

E. Calibration of the DR-RRT* *algorithm*

Construction of the cost function, J, in (2) utilizes a weighted sum approach that transforms the multi-criteria optimization problem into a single-objective one. One of the difficulties with the weighted sum method [21] is that varying the weights may not necessarily result in an accurate, complete representation of the true optimal set (Pareto) [22]. In the current work we adopt the *partial weighting approach* [23] in which the individual objective functions are grouped into sets with common characteristics: terminal conditions, relative positioning metric, and the multiple energy costs. Each set is used to form an independent functional with a unique weight, and in this way, the number of original objectives is reduced. To practically solve the task of defining the weights, the project in [3] designs a set of benchmark tasks such that the weights are explicitly expected to play one at a time. Then, for a simplified point-mass model of the Aqua-Quad and its energy subsystems, each of the developed benchmark scenarios is run through an optimization experiment ($fmincon()$ [24] or a statistical Monte-Carlo) with respect to *unknown weights* and a *well-known solution* that facilitates a more objective choice of the weights. In an attempt to find a more systematic definition of weights, the same benchmark test settings and simplified model were recently used to implement the approach in [25] that transforms the problem of optimal choice of the weights to a set of linear programming problems. The method provides practical criteria for the detection of the Pareto optimal weights of the linear programming task.

The results of defining the weight of the DOP metric (λ_{dop}) are illustrated in Fig. 5. The numerical result describes the influence of DOP on the positioning of eight Aqua-Quad sensing nodes with respect to a single stationary target. Fig. 5.a illustrates the evenly spaced nodes and the very low sensitivity of λ_{dop} to the nearly optimal DOP; the wide separation of nodes guarantees minimal values of DOP for many sensing scenarios [26]. Observe that the attempted "repositioning for better DOP" keeps the nodes close to their initial conditions.

TABLE II: Weight coefficients

	Value	Term	Objective
λ_{tg}	0.0225	Distance to goal	Bias toward the goal
λ_{dop}	0.0225	HDOP	Optimal repositioning w/r target
λ_{fl}	0.95	“Flight & Passive”	”Flight and Passive“ cost
λ_{ch}	0.005	PV charge	Increasing cost with distance

The result is very consistent for range and bearing-only sensing models. In turn, when range-only (see Fig. 5.b) and bearing-only (see Fig. 5.c) nonlinear sensing models are used, the response “for optimal repositioning”, that started from a clustered initial positioning of all nodes, is stronger and results in significant spread of the nodes relative to the target. Both figures also present the corresponding analytical expression of the DOP metric in the case of two nodes, for simplicity, where r_i, θ_i, and σ_i are the range, bearing, and the covariances of the sensors' measurement noise respectively. Next, the magnitude of the commanded “separation” is used to adjust the value of λ_{dop} in the *DR-RRT** algorithm, which is expected to produce a comparable repositioning. The final choice of λ_{dop} corresponds to its minimal value, which produces the “matching” separation of sensing nodes by *DR-RRT**.

Similar numerical experiments were designed to address the choice of the rest of the remaining weights. The cumulative result of all weights obtained is presented in Tab. II.

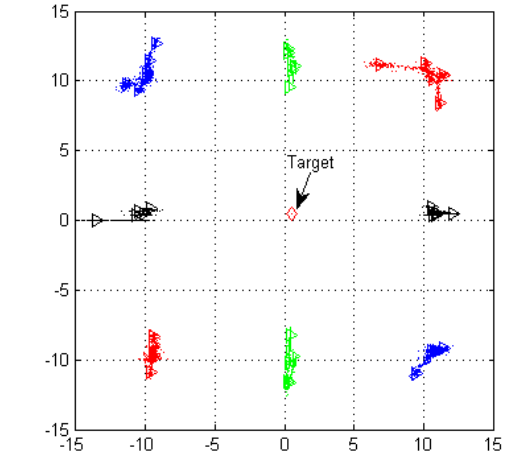
III. Numerical analysis of *DR-RRT** algorithm

This section is devoted to experimental numerical studies of the *DR-RRT** algorithm in application to tasks of navigating Aqua-Quads in various scenarios. All simulations were conducted utilizing MathWorks MATLAB and SIMULINK software [24] utilizing a computer with 4 GB RAM and an Intel Core i7-2600 processor operating at 3.40 GHz with a Windows 7 64-bit operating system.

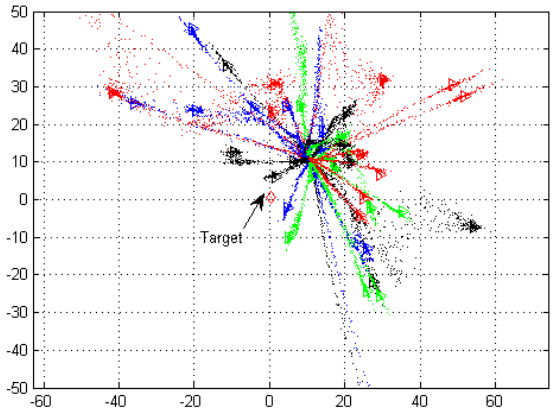
A. Evaluation of path planning in a vortex field

For this experiment we chose a nonlinear velocity field represented by a vortex (see Fig. 6). Initial and final conditions of the vehicle are chosen such that there is no direct drifting path connecting them and the direct flight segment is prohibitively expensive. The obstacles (black blocks) are chosen to force the path planner to ”hit“ at least one of them (at the center of the vortex) to examine the ability of the algorithm to explore the entire state space in drifting and in flight modes.

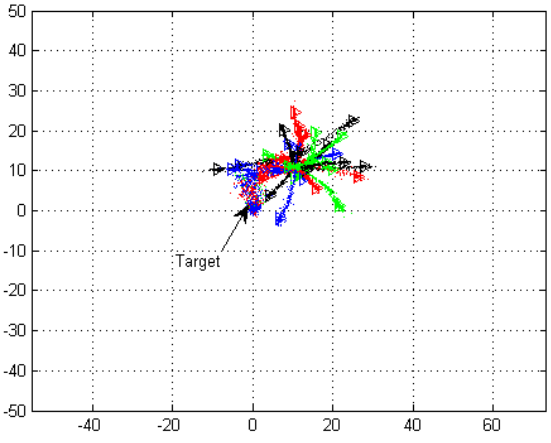
Analysis shows that the resulting path (Fig. 6, in magenta) consists of three long drifting and four short hopping flight segments; combined duration of flight segments is less than that required for direct flight. Graphics also include exploratory steps that produce a set of *Near* vertices (Fig. 6, reg circles) and the exploratory edges of two types: the green arcs correspond to flight hops and the black arcs define drifting paths. If it were possible to drift directly to the goal state, this would often be the best possible path. Because of an obstacle or the distance to target J_{tg} cost that verifies a negative gradient in range to the goal, the algorithm activates the primary functions of the *RRT** thus leading to the hopping phase. When conducting the “hop” phase, the Aqua-Quad is allowed limited flight time. The direction of that flight is randomly selected in an *RRT** manner; it originates from the most cost-efficient nearby parent node, and the final position of the Aqua-Quad is then tested for *Obstacle presence*. The random nature and the *Steering* toward *New* vertices of the flight steps is important, because it explores the entire configuration space and evaluates the cost



(a) Evenly spaced initial conditions require negligible adjustment



(b) Range-only sensing, $DOP_{range} = \sqrt{\frac{\sigma_1^2+\sigma_2^2}{\sin^2(\theta_1-\theta_2)}}$



(c) Bearing-only sensing, $DOP_{bearing} = \sqrt{\frac{r_1^2\sigma_1^2+r_2^2\sigma_2^2}{\sin^2(\theta_1-\theta_2)}}$

Fig. 5: Calibration of the λ_{dop} in a task of optimal placement of 8 sensors in a range-only sensing scenario

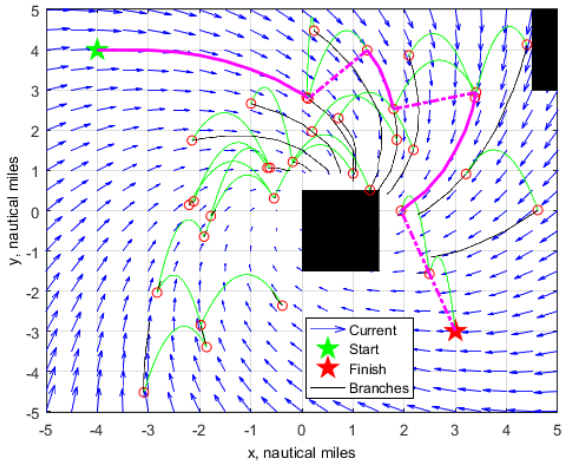


Fig. 6: *DR-RRT** solution in a vortex field with an obstacle

of different paths. *Rewiring* also occurs in the hopping phase, whereby drifting paths in adjacent branches can be joined by flight to the newest node if this joining results in a lower cumulative cost. In order to preserve the continuity of final paths in the *DR-RRT** version of the algorithm, rewiring is only allowed on nodes that originate from or terminate in a "hop" step. This ensures that control authority exists for the Aqua-Quad on the new path.

B. Optimality vs numerical feasibility

The optimality of the *RRT** solution is guaranteed asymptotically, as $t \to \infty$. In a practical implementation the *RRT** algorithm can terminate once an *initial solution* is found within a given number of steps, or it can continue to find paths that further minimize the objective function. This comes at the cost of computation time, whereas it may be preferable to find a rapid good enough solution instead. A simple question is then: What is "good enough?"

To address this question we performed a numerical experiment in which the energy-optimality of path planning and its computational time are explicitly analyzed. The project used a Monte Carlo approach in which the data from 1,000 simulations was collected for analysis. An initial obstacle set was randomly generated and then fixed in place for future simulations, to ensure consistency. The initial state $\mathbf{p}_0$ and goal region $\mathbf{p}_f$ were selected so that a fixed obstacle Ω_{obs} would lie between them and so that they would exist in opposing ocean current regimes $W(\mathbf{p}, t)$. Three other fixed sensing nodes were simulated, but not shown, to allow evaluation of TDOA-based (time difference of arrival-based) DOP within the cost term J_{dop}. The primary data of interest from the Monte Carlo simulation are the energy expended in the successful path and the computational time it took to run the algorithm to its conclusion. These values were stored at the completion of every tree that reached the goal and are displayed in Fig. 7. The red line at the top of Fig. 7 represents the $49.5Wh$ of energy that would be expended in direct flight between the Start and Goal positions. It can be seen that occasionally the *DR-RRT** found paths that utilized more energy than that consumed during direct flight. With a mean computation time of $190msec$, the algorithm could easily be re-executed when this occurs.

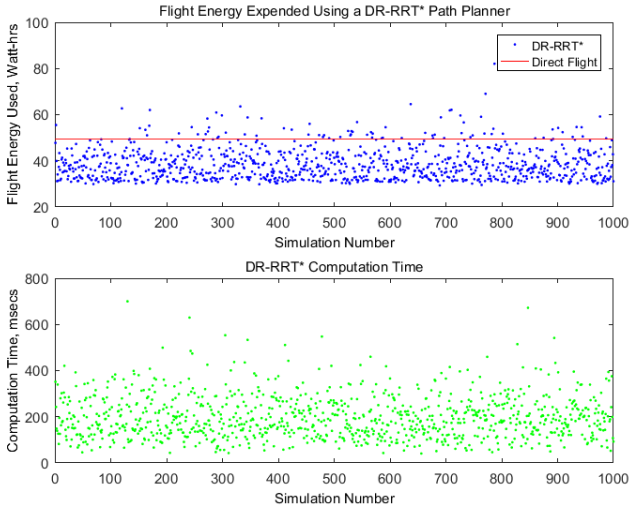


Fig. 7: Energy expenditure and computation time of 1,000 runs of the *DR-RRT** algorithm.

Next, with the statistics from the *initial solution DR-RRT** determined, we allow the algorithm to run far past its initial near-optimal solution thus we call it "optimality-seeking". It will continue to find new paths to the goal while rewiring existing paths in order to make the final path closer to true optimal. Therefore, another set of statistics was collected from 1000 Monte Carlo simulations of the optimality-seeking *DR-RRT**; once again the $49.5Wh$ of energy would be expended in direct flight. Each simulation was allowed to run for $25,000$ iterations and the successful paths to the goal were aggregated. The path with the minimum energy expenditure that reached the goal was selected, the flight energy expended on that path and computation time recorded. Fig. 8 displays the results of the optimality-seeking *DR-RRT**. It is worth noting that

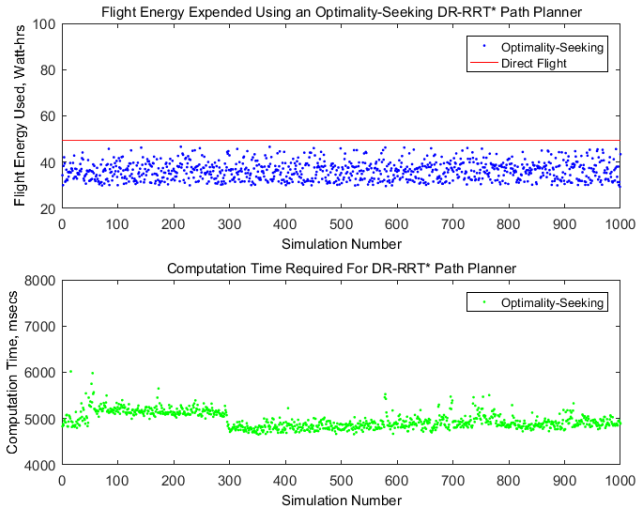


Fig. 8: Energy expenditure and computation time of 1,000 runs of the optimality-seeking *DR-RRT** algorithm.

none of the 1,000 simulations produced a path that used more energy than direct flight. This verifies the *RRT**s pursuit of

TABLE III: Statistics generated from two Monte Carlo simulations of *DR-RRT**

	Single solution			Optimality seeking		
	Mean	St. Dev	improve	Mean	St.Dev	improve
Energy, *Wt*	38.69	6.81	22%	36.26	3.93	27%
Time_{cpu}, *sec*	190	90	–	4964	18	–

infinite-time optimality in the case of energy expenditure. This efficiency came at some price, as the mean computation time for the sample set was almost five seconds. Considering the Aqua-Quads prevailing hours of drifting, this computational burden is perfectly feasible.

Table III provides a summary of the pertinent statistics collected in the Monte Carlo analysis of the *DR-RRT**. The benefit from running the algorithm beyond its initial solution point is irrefutable, but these gains (from a 22% improvement over direct flight to a 27% improvement) must be weighed against the computational cost. The value of computation time is mission-dependent and linked to the performance of the microprocessor that runs the code.

IV. Conclusion and future work

This paper provided an overview of the prototype Aqua-Quad platform and the capabilities that make it a major force multiplier. This envisioned mission motivated the efforts of this work to design and evaluate the Aqua-Quad's onboard path planning algorithms that support the conduct of sustained, energy-efficient surveillance and environmental monitoring.

The contribution of this work builds upon sampling-based methods by constructing a new tool for path-planning: the Dead-Reckoning Rapidly-Exploring Random Tree Star (*DR-RRT**) algorithm. The modification of the original *RRT** algorithm integrates the unique mobility characteristics of Aqua-Quad by implementing the "switching" control functionality. Numerical results confirm the expected energy optimal functionality of the resulting path planning solutions. Statistical results of a Monte Carlo experiment confirm a mean energy savings of up to 27% when the *DR-RRT** was allowed to run beyond its initial solution. The resulting computational burden is feasible for onboard implementation in modern microprocessors.

In the current work, the concept of harvesting solar energy is presented based on a 24-hour average of solar radiation as a baseline. To ensure persistence in the operational space, a more complex representation of available solar energy must be used based upon location, angle of incidence, and most important time of day. These variables can be incorporated into the *DR-RRT** planning framework to better estimate the expected energy balance. Furthermore, employment of the sensor beneath the Aqua-Quad will produce varying uncertain forces and moments while drifting or repositioning. Modified control laws for slung loads can be investigated to account for those uncertainties. A global cooperative mission planner can then be designed that takes into account the projected battery life of each Aqua-Quad in the flock and selects elements for repositioning based upon it.

V. Acknowledgment

The authors would like to acknowledge the project support provided by the Consortium for Robotics and Unmanned Systems Education and Research (CRUSER). Also, we are grateful to Prof. Kevin Smith at the Physics Department of the Naval Postgraduate School for insightful and operationally relevant technical discussions that significantly reshaped the original concept of the work.

References

[1] K. Jones, V. Dobrokhodov, and C. Dillard, "Aquaquad - solar powered, long endurance, hybrid mobile vehicle for persistent surface and underwater reconnaissance, part I- platform design," in *OCEANS 2016. MST/IEEE Conference*. IEEE, 2016.

[2] K. Jones and V. Dobrokhodov, "Hybrid mobile buoy for persistent surface and underwater exploration," April 2013, US Patent 9,321,529.

[3] C. H. Dillard, "Energy-efficient underwater surveillance by means of hybrid aquacopters," Master's thesis, Monterey, California: Naval Postgraduate School, 2014.

[4] C. C. Eriksen, T. J. Osse, R. D. Light, T. Wen, T. W. Lehman, P. L. Sabin, J. W. Ballard, and A. M. Chiodi, "Seaglider: A long-range autonomous underwater vehicle for oceanographic research," *IEEE Journal of Oceanic Engineering*, vol. 26, no. 4, pp. 424–436, 2001.

[5] G. N. Roberts and R. Sutton, Eds., *Advances in Unmanned Marine Vehicles*, ser. Control, Robotics & amp; Sensors. Institution of Engineering and Technology, 2006, vol. 69. [Online]. Available: http://digital-library.theiet.org/content/books/ce/pbce069e

[6] B. Garau, A. Alvarez, and G. Oliver, "Path planning of autonomous underwater vehicles in current fields with complex spatial variability: an $a*$ approach," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, April 2005, pp. 194–198.

[7] I. Spangelo and O. Egeland, "Generation of energy-optimal trajectories for an autonomous underwater vehicle," in *Robotics and Automation, 1992. Proceedings., 1992 IEEE International Conference on*, May 1992, pp. 2107–2112 vol.3.

[8] M. Chyba, T. Haberkorn, S. Singh, R. Smith, and S. Choi, "Increasing underwater vehicle autonomy by reducing energy consumption," *Ocean Engineering*, vol. 36, no. 1, pp. 62–73, 2009.

[9] M. L. Hernandez, "Optimal sensor trajectories in bearings-only tracking," in *Proceedings of the Seventh International Conference on Information Fusion*, vol. 2, 2004, pp. 893–900.

[10] T. A. Wettergren and R. Costa, "Optimal multiobjective placement of distributed sensors against moving targets," *ACM Trans. Sen. Netw.*, vol. 8, no. 3, pp. 21:1–21:23, Aug. 2012. [Online]. Available: http://doi.acm.org/10.1145/2240092.2240095

[11] I. Spangelo and O. Egeland, "Trajectory planning and collision avoidance for underwater vehicles using optimal control," *IEEE Journal of Oceanic Engineering*, vol. 19, no. 4, pp. 502–511, Oct 1994.

[12] C. Petres, Y. Pailhas, P. Patron, Y. Petillot, J. Evans, and D. Lane, "Path planning for autonomous underwater vehicles," *IEEE Transactions on Robotics*, vol. 23, no. 2, pp. 331–341, April 2007.

[13] T. Lolla, M. P. Ueckermann, K. Yiit, P. J. Haley, and P. F. J. Lermusiaux, "Path planning in time dependent flow fields using level set methods," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, May 2012, pp. 166–173.

[14] E. Zermelo, "Uber das navigationsproblem bei ruhender oder vernderlicher windverteilung," *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift fr Angewandte Mathematik und Mechanik*, vol. 11, no. 2, pp. 114–124, 1931. [Online]. Available: http://dx.doi.org/10.1002/zamm.19310110205

[15] A. E. Bryson and Y.-C. Ho, "Applied optimal control: optimization, estimation, and control," *Washington, DC: Hemisphere*, 1975.

[16] S. Karaman and E. Frazzoli, "Optimal kinodynamic motion planning using incremental sampling-based methods," in *Decision and Control (CDC), 2010 49th IEEE Conference on*. IEEE, 2010, pp. 7681–7687.

[17] S. M. LaValle, "Rapidly-exploring random trees a new tool for path planning," 1998.

[18] Acusonde, "Acousonde-3a underwater acoustic recorder," http://www.acousonde.com/downloads/Acousonde3A_Brochure.pdf, accessed:2016-06-29.

[19] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566–580, 1996.

[20] L. E. Kavraki, M. N. Kolountzakis, and J.-C. Latombe, "Analysis of probabilistic roadmaps for path planning," in *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on*, vol. 4. IEEE, 1996, pp. 3020–3025.

[21] R. T. Marler and J. S. Arora, "Survey of multi-objective optimization methods for engineering," *Structural and multidisciplinary optimization*, vol. 26, no. 6, pp. 369–395, 2004.

[22] I. Das and J. E. Dennis, "A closer look at drawbacks of minimizing weighted sums of objectives for pareto set generation in multicriteria optimization problems," *Structural optimization*, vol. 14, no. 1, pp. 63–69, 1997.

[23] J. Koski and R. Silvennoinen, "Norm methods and partial weighting in multicriterion optimization of structures," *International Journal for Numerical Methods in Engineering*, vol. 24, no. 6, pp. 1101–1121, 1987.

[24] MATLAB, *version 8.4.0.150421 (R2014b)*, The Mathworks, Inc., Natick, Massachusetts, 2014.

[25] I. P. Stanimirovic, M. L. Zlatanovic, and M. D. Petkovic, "On the linear weighted sum method for multi-objective optimization," *Facta Acta Universitatis*, vol. 26, no. 4, 2011.

[26] C. Yang, E. Blasch, and I. Kadar, "Geometric factors in target positioning and tracking," in *Information Fusion, 2009. FUSION '09. 12th International Conference on*, July 2009, pp. 85–92.