Collaborative Mission Planning, Autonomy and Control Technology (CoMPACT) for Unmanned Surface Vehicles

Joshua Redding†, Jayesh Amin‡, Jovan D. Bošković§ and Joseph Jackson¶
Scientific Systems Company, Inc.

In this paper, a Collaborative Mission Planning, Autonomy and Control Technology (CoMPACT) is presented for enhancing Unmanned Surface Vehicle (USV) capabilities in a maritime setting. CoMPACT uses a Finite State Machine (FSM) structure, can entail an array of communication architectures, and is demonstrated here for USVs under anti-submarine warfare (ASW), mine countermeasures (MCM), and other search and inspection tasks. FSM decision rules enable event-based or information-based transitions to govern the behavior of the autonomous agents. As part of this application and testing of CoMPACT, a study of innovative multi-agent cooperative search patterns conducive to reconfiguration (e.g. addition or deletion of agents from the search team due to reassignment or failure) is included with the simulation results. CoMPACT features several levels of autonomy, from completely tele-operated to fully autonomous mission completion, designed to reduce operator fatigue when several USVs are issued routine tasks.

I. Introduction

A growing percentage of future conflicts are expected to take place in the littoral regions around the globe against adversaries who possess increasingly effective weapon systems. Unmanned air, ground, underwater, and surface systems present an effective paradigm for warfighting in these theaters of operation during missions too hazardous for manned vehicles. Hence unmanned systems are critical components of the future naval forces. While significant research and development has been performed on multiple unmanned underwater vehicles (UUVs) and unmanned aerial vehicles (UAVs), relatively little has examined coordination and cooperative control of unmanned surface vehicles (USVs).

A USV offers many benefits to the Fleet. The first, and most obvious, is ensuring the sailors’ safety. A USV can be deployed in waters where it is unacceptable to send a manned vessel, including high threat environments or areas contaminated by nuclear, biological, or chemical agents. A USV could also remain on station for extremely long periods of time (up to several weeks) without resupply or human intervention. Such a capability could allow for long-term anti-submarine warfare (ASW), mine countermeasures (MCM), and Sea Base protection operations in areas of the world where future conflicts are likely. Additionally, a USV squadron could be deployed in advance of a carrier battle group or amphibious ready group to sanitize the area of potential threats and assure access for troops. Furthermore, USVs have large payload capacity, allowing for a multi-function mission package to be deployed with each USV. A single USV could simultaneously conduct many operations.

Operating on the water’s surface, a USV can use conventional power sources, such as diesel or gas turbines rather than relying on more limited power supplies, such as batteries or fuel cells. USVs are also unique among unmanned systems since they may communicate in all three mediums of interest—undersea, air, and space—relaying information from submerged assets (submarines, UUVs) to and from any combination of surface vessels, aircraft, or satellites.

A. Motivation for Collaborative Autonomy for USVs

The current concept of operation that requires one or more operators per USV must be extended to broaden the capabilities of USVs through operational concepts which take advantage of concurrent multiple vehicle deployment.
Proceeding with multiple USV operations under the current vehicle command and control approach would require unacceptable numbers of operators and potentially result in diminishing returns if coordination techniques and tools are not developed.

When USVs operate cooperatively, these operations inherently become more complex than the one vehicle, single sensor, single search area operations. As multiple mission functionality is implemented, the system tools to optimize multi-vehicle/multi-task performance are necessary. Thus cooperative behavior amongst USVs, supporting USVs as a “Force Multiplier”, will focus on: (i) Reducing individual operator workload; (ii) Reducing the number of operators required; (iii) Improving the responsiveness (and decreasing reaction times) of deployed USVs; and (iv) Ensuring maintenance of overall USV search coverage, while optimizing interrogation/inspection capability. Since USV utility is directly related to number of operators required, the force multiplier potential of USVs can be attained only if the collaborative autonomy design achieves multiple vehicles per operator rather than multiple operators per vessel.

B. Current results in the area of cooperating unmanned vehicles

Many studies of cooperative control of unmanned vehicles have been published. While most of the results apply to unmanned aerial vehicles (UAV) and, to a lesser extent, to unmanned underwater vehicles (UUV), many proposed techniques are sufficiently general and hence applicable to surface vehicles as well.

A key problem in the area of coordinated control of multi-agent systems is the assignment problem that represents a well-known optimization problem. In its simplest form, it is to assign items (jobs, missiles/strike vehicles) to other items (machines, targets). The solution to the multi-agent assignment problem has been attempted using numerous techniques including task-decomposition and auction-based tasking. Although algorithms for solving the assignment problem subject to a performance criterion exist, the problem is NP-hard (i.e. it cannot be solved in polynomial time); thus, heuristic techniques are often used. In the context of coordinated control of multi-agent systems, the assignment problem is commonly coupled with that of trajectory optimization, making the overall problem highly complex. Numerous approaches have been proposed for UAV coordination, from those based on rigidly defined desired relative position vectors and trajectory optimization techniques, to approaches based on loosely defined inter-vehicle interactions such as through artificial potential field methods.

The coordinated control problem can be loosely defined as that where multiple agents interact and coordinate their dynamic behaviors in order to accomplish a set of tasks that are necessary to achieve mission goals. Several issues arise in the context of multi-UAV applications such as aerial surveillance and tracking, collision and obstacle avoidance, formation reconfiguration, high-level control, and hardware and communications. Most of these problems also arise in the context of USVs, where a typical mission consists of several highly coupled complex tasks such as area search, vehicle-to-target assignment, removal/addition of vehicles from/to the platoon, platoon reconfiguration, and target pursuit and inspection. All these tasks need to be accomplished under spatial, timing and vehicle constraints, often in a highly dynamic setting. Hence typical behavior-based approaches may not be best suited for coordination of the above tasks under dynamic constraints. On the other hand, optimization-based techniques, based on Mixed-Integer Linear Programming (MILP) may result in unbounded computing times and may not be implementable in real time. Heuristic approaches may alleviate this issue, but their robustness to uncertainty needs to be demonstrated.

II. Autonomy Approach and Design Specifications

Autonomy is defined by the Merriam-Webster dictionary as “the quality or state of being self-governing” or alternatively as “the state of being free from the control or power of another.” In the context of military scenarios involving multiple unmanned vehicles, elimination of any and all human control and supervision is not yet feasible or even desirable at this stage. However, in accordance with the guidelines of “crawl-walk-run” as laid out in the USV and UUV Master Plans, designing systems that implement increasing levels of autonomy provides a transitional framework system that can take us from the existing multiple-operators-per-vehicle setup to multiple-vehicles-per-operator. Such improvements provide great benefits as a force multiplier while also reducing risks to the sailors’ lives amid dirty/dull/dangerous missions in a complex littoral environment. In order to achieve this vision, we have developed a Collaborative Mission Planning, Autonomy and Control Technology (CoMPACT) system for multiple USVs. The CoMPACT system is shown schematically in Figure 1.

Objectives of CoMPACT are to 1) provide a framework for autonomous collaboration, coordination & control, conflict resolution, re-planning, and assignment of heterogeneous USVs; 2) decrease operator workload using adjustable degrees of autonomy; 3) be modular, adaptable and capable of intelligent operation in a multitude of scenarios; 4) extend to a variety of different communication schemes and agent-level collaboration definitions.
Figure 1. Structure of the Collaborative Mission Planning, Autonomy and Control Technology (CoMPACT) system for multiple USVs

The implementation of CoMPACT presented here utilizes event-based task transitions, efficient obstacle representation, and fast dynamic path (re)planning. Ensuing sections address the autonomy architecture design, Common Operating Pictures (COPs), and how CoMPACT accommodates various levels of autonomy as required by the operator or mission scenario.

Mission specifications are generated by the command center that take into account vehicle availability, suitability for the mission or task, and dynamic constraints (maximum velocity, sensor range and field of view, maneuverability, etc.) The CoMPACT system receives high-level commands (“search the area”, “inspect a suspicious vessel”), and decomposes them into a set of coordinated tasks. For instance, for the area search, it assigns each vehicle to a part of the search area, and monitors vehicle assignment for target pursuit and inspection. To avoid a single point of failure, the assignment algorithm is decentralized so that the vehicles calculate their own corresponding performance index and communicate it to the neighboring vehicles. The vehicle with the best performance index is chosen for the task.

The proposed system is sufficiently general and covers the spectrum of control architectures, such as: (i) Fully centralized, where the Coordinated Control System (CCS) performs all the task assignments, and coordinates their execution; (ii) An architecture with a Local Leader, where the CCS assigns higher-level tasks to smaller platoons that each have their local leader responsible for task execution; and (iii) Highly decentralized, where USVs perform task assignment on their own by calculating the performance indices and communicating them to the neighboring vehicles; For instance, a USV is assigned a loosely-defined task (e.g. “go and inspect the vessel at coordinates (x, y)”, “when done return to the search area”), carries out pursuit and inspection, returns to the search area, carries out re-tasking for search while avoiding pop-up threats, such as mines, and collisions with friendly vessels.

The CoMPACT system is designed to efficiently handle the following diverse missions and tasks:

- Area search with a platoon of USVs;
- Vehicle-to-target assignment, and removal of vehicles from the platoon to pursue, inspect and neutralize mines and/or suspicious vessels;
- Trajectory generation with dynamic obstacle avoidance for intercept, inspection and data collection;
- Autonomous reconfiguration of remaining vessels to continue the search;
- Return of the pursuit & inspection vehicles to the platoon;
- Platoon re-tasking to continue the search with additional vehicles; and
- Replanning under failures and faults.

Specifically, this paper presents results of the implementation of the CoMPACT system under three scenarios. Documented here are missions such as reconfigurable coordinated search, where multiple agents are tasked to survey a region in a cooperative manner subject to configuration changes (Figure 2), and MCM and ASW missions, where teams are comprised of members with specific payloads and sequential roles in the mission (Figure 3).
Figure 2. Reconfigurable Search Scenario

Intelligence indicated a possible threat at this location so an agent was pulled from the search to detect and further service the threat as necessary.

Figure 3. MCM and ASW Scenario

Intelligence indicates possible submarine threat at this location.

Figure: MCM and ASW Scenario

Figure: Reconfigurable Search Scenario
III. Communication Architectures

An important driving factor for autonomy design are the communication constraints imposed by the mission scenario. A domestic urban unmanned surveillance operation would have available almost unlimited reliable communications along with sufficient human supervisors. On the other hand, a blockade mission in a remote hostile littoral environment would have extremely unreliable and constrained communications as well as requirements to minimize the risk to human lives. This range of communication constraints and available topologies drives the choice between centralized or decentralized autonomy designs. Mission scenarios we have addressed in this project mainly require a decentralized approach. However, it is also important to have a design that can easily be adapted to a centralized approach when necessary or appropriate.

CoMPACT implements an Intelligent Agent (IA) structure, as illustrated in Figure 1, that does not hinge on any one collaboration structure to operate. The autonomy can be adapted to a number of architecture designs based on the mission. Constraints that contribute to architecture selection include communications bandwidth, communications range, on-board processing limitations, robustness requirements, and typical mission profiles. The following section describes the various communication architectures and topologies for a collaborative distributed system.

A. Centralized or Global Leader Architecture

A common method for control of multi-agent systems is the centralized approach that may be implemented either through a central command center (LCS or remote ground station), or through a single USV designated as the global leader. Technically, these are similar in the sense that all the information is gathered at a single point for decision making and task allocation for the individual USVs by the decision maker. In the Global Leader collaboration architecture, Figure 4, the vehicles rely upon one USV to gather all of the information regarding USV states and capabilities, then make assignment decisions based upon the respondents status.

![Figure 4. The Global Leader architecture requires all assignments be issued from a central agent.](image)

Considerations for the centralized approach include requirements for the Leader to have enough bandwidth to receive updates from all USVs in the platoon, have sufficient broadcast range to communicate with the farthest USV, and have sufficient processing power to arrive at the best assignment configurations quickly. It is also important to note that, as the number of USVs in the operations increases, the communication and supervision requirements grow very rapidly making it hard to scale. However, having a Global Leader make decisions removes uncertainty about which agents are best suited to perform certain tasks, and likely can provide global optimal allocations compared to potential local optimal allocations in the decentralized implementations. Likewise, a Global Leader structure will not result in conflicting actions among the agents. The biggest disadvantage of a centralized approach is its vulnerability to a single point of failure. Any loss of the leader or its communication capabilities may render all the other assets unusable.

B. Local Leader Architecture

The next level of decentralization is achieved through clustering of vehicles. The idea of a Local Leader, Figure 5(a), is that decisions can be more efficient and robust when no one agent is responsible for issuing all the assignments. For example, if the platoons are organized by geographical location and a need arises for a USV to inspect an unknown
target at a certain location, the Local Leader of the platoon closest to the unknown threat may be assigned the task. The Local Leader may then request status from each USV within its platoon, and the agents will be given assignments after the responses are gathered.

This decision-making structure, when scaled properly, can be faster at arriving at efficient assignments, and tends to be more robust to single-point failures, since several agents are capable of distributing the decision-making process. Furthermore, the local decision making can be handed-off to another USV in the platoon if the Local Leader drops out of the communications link. Local Leaders enable a longer-range communications link since not all agents need to be within the communications range of the LCS.

C. Decentralized Architecture

A completely decentralized architecture moves the decision making ability to the individual vehicles (Figure 5(b)). However, to be able to make collaborative decision, the vehicles now need to have information about other vehicles and assets that are available. This is discussed in detail below.

1. Common Operating Picture, Common Relevant Operating Picture

Truly decentralized architectures necessitate the concept of a Common Operating Picture (COP). COP is a database of the locations, capabilities and status of all the other assets in the field. COP also includes other world information like missions parameters, target data, terrain or bathymetry data, and so on, which are necessary for performing the mission. When individual vehicles need to make a decision as to who needs to respond to an event, they all use the data from their own copy of the COP. If all the vehicles use the same algorithm, then barring inconsistencies in the COP data, they should all arrive at the same decision without the need for negotiations or centralized supervision. If COP is to be synchronized by sharing data every time a decision is required, most advantages of the decentralized approach are lost through these increased communications and delays.

One approach is to use the low activity periods in the field, when no significant events are occurring, to periodically share information between the vehicles. Additionally, not every vehicle needs to store all the information about all the other vehicles. This leads to a useful subset of COP that is relevant to decision making for a particular vehicle, e.g. a node probably needs more data about its complementary sensor vehicle than it does from a vehicle in another remote cluster. Common Relevant Operating Picture (CROP) is a subset of COP that is then stored on each vehicle. Each vehicle communicates its information periodically during the low activity periods which is used by all the other vehicles to synchronize their local copies of the CROP. Through this design, when a significant event occurs, decisions are made quickly by using the latest local CROP data.

Figure 5. (a) The Local Leader architecture allows the vehicles to be grouped according to capabilities, locale, or some other qualifier into platoons with one of the platoon members responsible for assimilating data and making locally viable decisions. (b) The Decentralized architecture requires all agents to represent themselves when deciding how to collaborate.
2. Decentralized implementation

The Decentralized structure, while robust to single-point-failures, does lead to potential decision conflicts if the CROP becomes stale or inconsistent across the nodes. This may lead to multiple nodes trying to respond or no vehicle may respond to significant events. This problem is easily solved by designing the decision rules and algorithms to be robust against the expected levels of uncertainties and using minimal negotiations between the potentially conflicting nodes.

As discussed above, various communications topologies and architectures are driven by the mission requirements. Though the USVs in hostile littoral warfare operations are likely to require a completely decentralized architecture, we have designed CoMPACT to be adaptable to other architectures as well. By using well structured approaches of Artificial Intelligence (AI), we have implemented a decision making process that can easily be migrated from individual vehicles to a centralized location or global leader if the mission warrants it. In the following section we present our process of arriving at the autonomy design.

IV. Artificial Intelligence and Autonomy

Since the goal of autonomy is the elimination of external human control of unmanned vehicles for prolonged periods, much of the decision-making performed by human operators is supplanted by some form and subset of Artificial Intelligence (AI). Artificial Intelligence (AI) is the study and design of intelligent agents, where an intelligent agent is a system that perceives its environment and takes actions which maximize its chances of success. This clearly differentiates itself from systems that sometimes masquerade as being autonomous, but are in fact time-scripted waypoint-following automata which are not truly reactive to the environment. AI research uses tools and insights from many fields, including computer science, psychology, philosophy, neuroscience, cognitive science, linguistics, operations research, economics, control theory, probability, optimization and logic.

A. Intelligent Agents

All control techniques that use various AI computing approaches like neural networks, networks based on Bayesian probability computations, fuzzy logic, machine learning, evolutionary computation and genetic algorithms, can be put into the class of intelligent identification and control strategies. Intelligent control involves design of Intelligent Agents (IA) which perceive and act upon the environment in pursuit of goal(s). IAs can vary in complexity from simple reflex agents to fully adaptive learning agents based on the application requirements and constraints. However, at the core of each IA lies the set of decision rules or logic that it follows to perform actions in response to the perceived environment. Figure 6 shows the basic IA concept.

Autonomy design is then the development of decision rules that the intelligent agents use while operating in an unstructured environment. This, in essence, defines the “behavior” of the intelligent agent. These behaviors can vary from being simple reactive reflex rules (bugs scattering when light is suddenly switched on, prey-predator interaction) to highly complex adaptive learning behaviors. There are two main methods of arriving at the decision rules that define an IA behavior:

1. Rules generated by automated learning (reinforcement or supervised, off-line or on-line); and
2. Rules generated by Domain Experts.

Choice of one of the above methods depends largely upon the specific application requirements and constraints. In some cases, a design which incorporates both of these methods (e.g. initial rules defined by domain experts, and
continued learning through environmental interaction), may be better suited for a specific application.

**Learning methods**

Learning methods typically involve finding an input-output mapping by exploring and exploiting the environment. Various actions are attempted in response to environment inputs and the ones which receive the most reinforcement increase their chances of being kept (Figure 7). Learning can be performed off-line using simulations against modeled environment, or on-line while operating in real-time. Alternatively, off-line learning is followed by on-line adaptation. These methods are suitable when the environment and its inputs can be modeled with sufficient fidelity and accuracy. Simulation coverage of expected environmental inputs becomes important in determining the performance of learned decision rules in the presence of untested inputs. On-line learning also requires the acceptability of making mistakes while working in real-time, which is unacceptable in many cases.

![Figure 7. Reinforcement Learning.](image)

**Domain experts**

Domain experts or subject matter experts (SMEs) are individuals who are both knowledgeable and extremely experienced with the intended application. Domain experts are consulted when coding the decision rules for intelligent agents. The IA should make the same (or similar) decisions that the domain experts would make while operating in the same environment. Domain experts are also valuable in defining the concept of operation (CONOPS) for the system which can guide the decision rule design process.

Since the missions involving unmanned vehicles are trying to replace dangerous or labor intensive manned missions, vast amounts of human expertise available from operations like MCM and ASW can be utilized. Moreover, real-world scenarios become very hard to simulate with sufficient fidelity, accuracy and richness that can be used by the learning methods. Domain experts can also provide low probability but important scenarios that would be hard to replicate in automated simulation setups.

**Decision Rule Implementation**

Once the method of determining the decision rules has been selected, the next choice we confront is one of the various methods that can be used to represent and implement these decision rules. Various possible methods are: finite state machines (FSMs), Petri nets, decision tables, genetic algorithm (GA) strings, neural network (NN) mappings, or, in some simple cases, even ladder logic programming. GA and NN methods are suitable for applications where learning methods are used and the process inherently generates these mappings. However, as discussed above, some applications do not lend themselves easily to accurate modeling, and make the learning method results somewhat unreliable. Domain experts method currently seems better suited for designing the decision rules for the USV missions, especially given the data available. Among other implementation methods, FSMs have been extensively used and tested in the aerospace as well as in automotive industries for hybrid systems design. FSMs are also now starting to be a valuable tool in fault tolerant system designs for aerospace and space applications.
B. Finite State Machines

A finite state machine (FSM) or automaton is a paradigm for modeling behavior composed of a finite number of states, rules for event-triggered transitions between those states and actions. In addition to their use in modeling reactive systems presented here, finite state automata are significant in many different areas, including electrical engineering, linguistics, computer science, philosophy, biology, mathematics, and logic. In computer science, finite state machines are widely used in modeling of application behavior, design of hardware digital systems, software engineering, compilers, network protocols, and the study of computation and languages.

Finite State Machines (FSMs) can be typically designed as a hierarchical decomposition of states/modes, tasks, subtasks, actions and action primitives. Examples at various levels may include:

- States/Modes: Search, Inspection, ISR, Build World Model, ...
- Tasks: Plan search, Detect, Localize, Reacquire, Take & send video, Interrogate, Neutralize, ...
- Subtasks: GoTo, Start, Stop, Loiter/Stationkeep, Wait, ...
- Actions: Path & Trajectory Planning, ...
- Action Primitives: Constant Velocity cruise, Accelerate, Decelerate, Turn, ...

Events guarded by specified conditions trigger transitions between states. Events and transitions may lead to specific tasks and actions. At the lowest level, tasks and actions are implemented as suitable algorithms. For example, a target detection event may trigger a transition for the vehicle into a pursue state, which may require an intercept transit task, which in turn requires invoking a path planning algorithm to find a safe and feasible pursuit path. FSMs can be represented graphically using state diagrams, Figure 8, or using one of various state transition table formats.

Using state diagrams, we designed the autonomy for the heterogeneous USV search, MCM and ASW missions. Simulation results from this design gathered tested against a low fidelity simulation are presented. Further research requires higher fidelity world modeling as well as a more complex FSM design.

![Figure 8. Finite state machine (FSM) design for a USV search and respond mission.](image)

Due to the various threat types, it was important to include diversity in the vehicle capabilities. Vehicles equipped with sonar, or other underwater detection methods are essential, as are vehicles carrying radar to detect surface threats. The simulations include multiple USVs, UUVs and pairs of unmanned mine-hunting vehicles (AMVs and PMVs). The USVs have surface searching capabilities and are faster than the other vehicles. The UUVs are included as kill vehicles only and are deployed from the USVs. The mine-hunting vehicles are conceptually based on Lockheed Martin’s *Sea TALON* system and are deployed in pairs as one carries an active sonar array and the other carries a passive array.

Each vehicle carries specific sensors, and each sensor is simply modeled by a range and field-of-view. To advance a threat’s status, a specific task needs to be carried out, which requires the use of a specific sensor. Thus, only certain vehicles are capable of accomplishing certain tasks for a given threat type. Table 1 shows the breakdown of vehicle capabilities by task and threat type.
Table 1. Vehicle capabilities chart showing which vehicles can accomplish which tasks on the included threat types. Vehicles are as follows: 
UMV Unmanned Mine-hunting Vehicle pair; USV Unmanned Surface Vehicle, UUV Unmanned Underwater Vehicle. For example, a UMV can perform the 'Clear' task for threats of type 'Submarine', 'Mine', or 'Ship'.

<table>
<thead>
<tr>
<th>Threat</th>
<th>Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Travel</td>
<td>Observe</td>
</tr>
<tr>
<td>Submarine</td>
<td>UMV</td>
</tr>
<tr>
<td>Mine</td>
<td>UMV</td>
</tr>
<tr>
<td>Ship</td>
<td>USV</td>
</tr>
</tbody>
</table>

Table 2. Threat status descriptions and associated vehicle tasks with descriptions

<table>
<thead>
<tr>
<th>Status</th>
<th>Description</th>
<th>Task</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>alert</td>
<td>Presence of a possible threat is likely</td>
<td>travel</td>
<td>Vehicle must travel to within sensor range of the threat</td>
</tr>
<tr>
<td>detected</td>
<td>Threat has been detected by an on-board sensor</td>
<td>observe</td>
<td>Vehicle plans an observation path around the threat for positive classification</td>
</tr>
<tr>
<td>posID</td>
<td>Threat has been identified as either friend or foe</td>
<td>clear</td>
<td>If threat is foe, path is planned to check and clear the immediate area of other possible threats</td>
</tr>
<tr>
<td>cleared</td>
<td>Immediate surroundings are clear of other threats</td>
<td>deliver</td>
<td>Path is planned to deliver the kill vehicles to the threat area</td>
</tr>
<tr>
<td>marked</td>
<td>Kill vehicles are in place</td>
<td>strike</td>
<td>Kill vehicles are given paths to intercept threat</td>
</tr>
<tr>
<td>hit</td>
<td>Threat has been hit by kill vehicles</td>
<td>confirm</td>
<td>Plan a path to observe the threat and confirm that it has been killed</td>
</tr>
<tr>
<td>killed</td>
<td>Threat has been destroyed and/or disabled</td>
<td>none</td>
<td></td>
</tr>
</tbody>
</table>

C. Dynamic Mission Planning and Task Assignment in the Context of Multiple USVs

We used the vehicle assignment algorithm based on a performance criterion and constraints arising from the capability matrix so that the related cost and constraints depend on the following for each individual USV: (i) sensor suite, (ii) sensor characteristics, (iii) maximum acceleration, turn rate and velocity, (iv) available fuel, and (v) current distance from the target.

The key element in this cost are the paths generated by the Rapidly-exploring Random Trees (RRT) algorithm that enable rapid calculation of the distances between the vehicle and the target, and allow quick recalculations in the case of path re-planning for pop-up threat avoidance. The cost also takes timing constraints into account and, essentially, represents a time-parameterized greedy algorithm.

However, it is also clear that other assignment algorithms arising in the context of the multi-agent task allocation (MATA) problem are fully applicable to coordinated mission management and autonomy for multiple USVs. In all cases these algorithms can be implemented as Dynamic Task Allocation (DTA) strategies. This is because, regardless of the technique chosen to address the allocation problem, the assignment is recalculated whenever there is a change in the system state or environment. Hence this is an event-driven DTA where, for instance, events include "task accomplished", "presence of new targets", "presence of pop-up threats", etc. It should be noted that the DTA is closely affected by the sensor capabilities and communications architecture used.
V. Levels of Autonomy in Autonomous Control of Multiple USVs

A primary objective of the project was to reduce the operator workload during multi-USV missions in complex littoral environments. One of the key issues in this context is a suitable framework within which the autonomy levels can be defined in a meaningful way. The goal of studying autonomy levels is to arrive at a thorough understanding of the operator role at each level, and to correlate the “amount” of autonomy with the operator workload. This has been set within the framework of CoMPACT system allowing, at the highest level, substantially reduced workload for individual operators, along with the reduction in number of operators needed to handle a large fleet of autonomous USVs.

As an example mission to define autonomy levels, we have chosen the ship inspection mission carried out by a single USV. The objective is to use the USV, starting from some initial position at initial time, to inspect a stationary ship by taking a video, and return to the starting point. The video stream is processed to carry out Automatic Target Recognition (ATR). Several levels of autonomy have been defined in this context, from fully manual to fully autonomous.

Our first step was to define autonomy levels in the context of this specific mission. The resulting levels are based on the basic tasks during a ship inspection mission, but can easily be abstracted to other autonomous vehicle missions. These include: (i) USV trajectory tracking control; (ii) Inspection data processing; and (iii) Dynamic reactive re-planning.

**Level 1: Tele-Operation.** In this case all the tasks are carried out manually (trajectory following, data processing, re-planning). This level is characterized by the highest operator load, and no autonomous control.

**Level 2: Assisted Tele-Operation.** In this case the CoMPACT system generates waypoints corresponding to the mission path, and also suggests optimal or sub-optimal trajectories displayed on the mission GUI. The operator manually steers the USV through the waypoints trying to also track the trajectory. All other tasks are also carried out by the operator.

**Level 3: Semi-Autonomous Operation.** CoMPACT generates waypoints based on the ship location, as well as the trajectory and outer-loop control & guidance. Reactive task or mission re-planning is carried out by the operator, while ATR is carried out at the command station.

**Level 4: Autonomous Operation.** Besides the tasks from Level 3, the CoMPACT also carries out reactive re-planning of tasks or mission segments. ATR is still carried out at the command station.

**Level 5: Fully Autonomous Mission.** Besides the tasks from Level 4, CoMPACT also carries out ATR and declares “Friend” or “Foe”.

Relationships between different autonomy levels are illustrated in Figure 9.

In more general cases, different autonomy levels are applicable to a variety of missions carried out with a single UAV. These levels are consistent with ASTM Standard F2541.29298-1: Guide for Autonomy and Control of Unmanned Maritime Vehicles.

VI. Autonomous Search using Multiple USVs

The idea of a cooperative reconfigurable search is to dynamically allocate resources to guarantee coverage of a specified region in the presence of a changing environment and varying available resources. Under this project, the available resources are unmanned agents, and the littoral environment changes due to pop-up obstacles and threats.

Any vehicle participating in the search can become unavailable at any time due to a higher priority task or a system malfunction. It is assumed that a vehicle becoming unavailable will make its status known to the team either through communications, in the case of a higher priority task, or through communication silence in the case of a system malfunction. The remaining vehicles then cooperate to reconfigure their search patterns to provide coverage of the search region while minimizing the number of redundant search cells. Cooperative reconfigurable search-related simulation results are included below.

Figure 10(a) shows the initial layout of the simulation. All threats are stationary and the search area is pre-defined. All vehicles will initially participate in the search. Since all agents initially participate in the search, the area is divided into 5 equal cones and each agent is assigned to search a cone and is given a path that accommodates this assignment. These paths are shown in Figure 10(b). In Figure 11(a) we see that the vehicles’ paths initially cover the search area such that if no re-plans are necessary, coverage of the search region is guaranteed. In Figure 11(b) we see that one of the vehicles is no longer available to participate in the search. As seen in the Simulation Messages window on the right side of the figure, Vehicle 3 is not available. This simulates a variety of situations, such as when a vehicle becomes disabled and can no longer function as a member of the team. Becoming unavailable in this manner can also simulate a vehicle being pulled from the search to service a threat outside of the search area.
Levels of Autonomy for Ship Inspection

(a) Levels of Autonomy

(b) Levels of Autonomy Details

Figure 9. The levels of autonomy for ship inspection.
Continuing, we see in Figure 12(a) paths are successfully re-planned for the remaining agents that cover the unsearched portion of the search area. Notice that the region searched by Vehicle 3 prior to it becoming unavailable was recorded and will not be searched again. Figure 12(b) shows that Vehicle 3 is now available to join the search. This triggers another cooperative reconfiguration of the search pattern. Figure 13(a) shows the team of agents cooperating to reconfigure the search patterns and include a new vehicle in the search. Figure 13(b) shows the search is complete, leaving all participating agents along an arc awaiting a new assignment or ready for easy recovery by an LCS.

Figure 10. (a) The initial layout of the search area, USV team, and threats. (b) The search region is divided among available agents and initial assignments are made.

Figure 11. (a) The agents' paths are designed to cover the region of interest. (b) One of the vehicles is no longer available to participate in the search.
Figure 12. (a) The remaining available agents cooperatively reconfigure their search paths to cover the region left open by Vehicle 3 becoming unavailable. (b) Vehicle 3 has become available to rejoin the search.

Figure 13. (a) The remaining search area is cooperatively reconfigured among the available vehicles. (b) The search is complete.
VII. Fully Autonomous MCM & ASW

A mission scenario used for this work consists of the following tasks: (i) Predefined search, (ii) Detection of a suspicious vessel, (iii) Assignment of one or more USVs from the team to inspect the vessel, (iv) Communication of the results of inspection to the control center, (v) Re-tasking of the remaining USVs to continue the search, (vi) Return of the USV that has carried out the inspection to the search team, and (vii) USV team re-tasking to continue the search with an additional vehicle.

Figure 14. A littoral region was chosen as the theater of operations for our mission scenario. One area of the region was chosen for a cooperative search mission and multiple threats are scattered throughout the region. The mine threats are initially at unknown locations.

Figure 14 shows the mission scenario’s layout and theater of operation. As seen, the littoral environment can harbor many different threat types, such as submarines, mines and ships. A heterogeneous team of unmanned vehicles is deployed from a Littoral Combat Ship (LCS). The team consists of multiple unmanned surface vehicles (USVs), active and passive pairs of unmanned mine-hunting vehicles (AMVs, PMVs) and unmanned underwater kill vehicles (UUVs), the latter being deployed from the USVs. The unmanned vehicle team immediately begins a cooperative area search, scanning for both surface and subsurface threats. When a threat is detected, or when intelligence alerts the team of a possible threat, a “re-plan” is triggered amongst all team members. Associated with this re-plan is key information about the possible threat, such as its probable type and location. A specific vehicle task is then derived and each vehicle capable of performing this task on the threat submits an associated “score”. The vehicle, or vehicle pair, depending on the task, with the highest “score” is chosen to carry out the required task. Those vehicles not assigned to service the threat remain assigned to the cooperative area search, which must be reconfigured due to the now fewer available team members.

The vehicles servicing the threat rejoin the cooperative area search upon assignment completion and the search is reconfigured to accommodate the additional resources. This process of dynamically reassigning vehicles to and from the area search to observe, identify, strike, confirm and otherwise service various potential threats continues until the cooperative area search is complete. Note that the autonomy involved in the MCM/ASW activities is designed to be resilient amid sparse obstacle fields. Next we present results on maneuvering amid dense obstacle fields using various levels of autonomy, and describe the results of our mission scenario simulation.
The simulation has the following capabilities: (i) Team-to-target assignment algorithms, where teams of two or more vehicles are assigned to the target based on a capability matrix and cost minimization; (ii) Simultaneous arrival of two or more vehicles to the target area with prespecified headings; (iii) Pop-up threat avoidance with velocity adjustment to guarantee simultaneous arrival; and (iv) Vehicle loitering and waiting for the next assignment.

The simulation also includes: typical vehicle velocities; an arbitrary theater of operation size; 1 LCS to deploy the team of unmanned vehicles; 1 UAV to provide communication between the team and the LCS; 2 UMV pairs (1 UMV pair = 1 AMV + 1 PMV) to service the subsurface-related tasks and threats; 2 USVs to service the surface-related tasks and threats; 3 UUVs on board each USV (6 total) to strike unfriendly threats; 1 submarine threat; 1 mine threat; and 1 ship threat. As mentioned previously, each USV has a simplified sensor model describing its range, performance and field of view. The following results present the results of the USVs that were pulled from the simulation of cooperative area search. This simulation is focused on task generation and the assignment of vehicles to pop-up threats to advance the status of each detected threat to either friend or killed. The assignment algorithm is built around a “re-plan” flag which indicates one of the following three states:

1. Keep all current assignments, no re-plan needed
2. Obstacle detected in vehicle i’s path, re-plan around this assignment. In other words, pick the best vehicle (not necessarily vehicle i) to accomplish the task vehicle i was assigned to before it detected an obstacle in its path.
3. The status of one or more threats has changed, issue a global re-plan. A global re-plan assigns the best vehicle to each threat in need of a task. Often, this results in vehicles being reassigned to the same task they previously had. However, this method is kept for generality.

Multiple snapshots from the simulation described above are included.

Figure 15(a) shows the initial layout of the simulation. All threats are stationary, as is the LCS. The UAV launches from the LCS and loiters around the midpoint between the farthest unmanned vehicle and the LCS. All USVs (orange) and UMV pairs (red = AMV, green = PMV) not assigned to MCM or ASW tasks are assigned to cooperative area search.

Notice in Figure 15(b) in the Simulation Messages window on the right side of the Figure, a message indicates that “Intel alerts of Threat 3” and “Intel alerts of Threat 1”. This message window is updated at every time step with the most recent messages at the bottom of the stack. Also seen in Figure 15(b), assignments are made to travel to Threat 1 and Threat 3 so they can be detected with vehicle sensors.

Since all agents initially participate in the search, the area is divided into 5 equal cones and each agent is assigned to search a cone and is given a path that accommodates this assignment. These paths are shown in Figure 15(b).

In Figure 16(a) we see that the USV (orange) has detected an obstacle in its path. In this case, the re-plan flag was set to 2 to initiate a re-plan for this assignment only. As seen by the adjusted path (orange), the same vehicle was chosen again to travel to Threat 3.

In Figure 16(b) we see that a UMV pair (red + green) have detected a possible threat. This is indicated in the Simulation Messages window as well as by the outline of the gray threat turning yellow. Continuing, we see in Figure 17(a) that the same UMV pair that detected the possible threat (Threat 2) has been assigned to observe the threat.
observe halted due to lack of resources. They can only 2 UMV pairs, so the assignment algorithm assigns the two pairs to the tasks with the highest score (i.e. lowest cost). The threat with the lowest score (Threat 1) is left to wait for available resources, while Threat 2 and Threat 3 require the use of a UMV pair. Threat 1 needs to be detected, but since it is a submarine, it can only be detected by a vehicle. (a) The USV (orange) detects an obstacle en-route and reacts, planning a new, safe path. (b) A UMV pair detect a mine en-route.

Threat Status:

- Threat 1: Detected
- Threat 2: Observed
- Threat 3: Identified

Vehicle Status:

- USV: Travel on tgt 3, ETA = 720.3
- AMV: Travel on tgt 1, ETA = 1972.4
- PMV: Travel on tgt 2, ETA = 1096.5
- UAV: Loiter
- LCS: Loiter

Simulation Messages:

- Obstacle alert: ETA = 672.5
- Intel alerts of Threat 1: ETA = 110.0
- Intel alerts of Threat 3: ETA = 72.5

Figure 17. (a) One UMV pair observes the newly detected mine, while the other is assigned to travel to threat #1. The USV has detected threat #3, and planned an observe path around it. (b) The USV has identified threat #3 as unfriendly and called a UMV pair to clear the area of further threats. Also, a UMV pair has identified threat #2 as a mine and are beginning to clear the area. Progress on threat #1 is halted due to lack of resources.

and attempt to classify it. Also shown in Figure 17(a) the USV has detected Threat 3 and has now been assigned to observe it. These assignments match those organized in Table 1 and Table 2.

Figure 17(b) shows some interesting features of the assignment algorithm. All threats are in need of some task that requires the use of a UMV pair. Threat 1 needs to be detected, but since it is a submarine, it can only be detected by a UMV pair. Threat 2 has been positively identified and needs the area surrounding it cleared in preparation for striking it. To clear an area requires checking for mines and other threats, this is always carried out with a UMV pair, regardless of threat type. Threat 3 has also been positively identified and needs its surrounding area cleared. However, there are only 2 UMV pairs, so the assignment algorithm assigns the two pairs to the tasks with the highest score (i.e. lowest cost). The threat with the lowest score (Threat 1) is left to wait for available resources, while Threat 2 and Threat 3 have their surroundings cleared by the available UMV pairs.

Figure 18(a) shows that the area surrounding Threat 2 has been successfully cleared and this UMV pair has been assigned to clear Threat 1, who has been waiting. At the same time the UMV pair is reassigned to Threat 1, a USV is assigned to Threat 2 to carry out the deliver task required to advance the status of Threat 2 from cleared to marked.

After the UUVs have been delivered to the area, as seen in Figure 18(b), they are assigned a strike task to attempt to kill the threat. It is seen that the two UUVs hit their target, but it will need to be confirmed that the target is killed. The UMV pairs continue their previous assignments. When one pair is finished, they will become available to confirm the hit on Threat 2.

In Figure 19(a) we see that the UMV pair traveling to Threat 1 have detected it and have been reassigned to now observe it. Note that all vehicles in this simulation are NOT to drawn to scale against the littoral environment.
In Figure 19(b), we see that Threat 1 has been positively identified and the UMV pair previously assigned to observe it has now been assigned to clear the area surrounding it, in preparation for a strike. The other UMV pair has finished clearing the area around Threat 3 and have been reassigned to confirm the hit on Threat 2 made earlier. Meanwhile a USV has been assigned to deliver UUVs to the area near Threat 3, which has just been cleared.

In Figure 20(b) it is shown that Threat 3 has been hit and the USV that delivered the UUVs has been assigned to confirm the hit and pronounce the threat as killed. The UMV pairs are continuing with their previous assignments, namely to clear the area around Threat 1 and to confirm the hit on Threat 2.

Both Threat 2 and Threat 3 were pronounced killed after the hits were confirmed. The area surrounding Threat 1 has been cleared and a USV has been assigned to deliver kill UUVs nearby, as noted in the Messages of Figure 21(a). Also in that Figure, the USV delivers UUVs to the front door of Threat 1. The UMV pairs ready for reassignment since there are no threats in need of tasks that require the use of a UMV pair. These vehicles are sent back to assist in the cooperative area search.

In Figure 21(b) we see that the UUVs have hit Threat 1, and the UMV pair nearest have been assigned to confirm the hit and declare the threat killed. The USV has successfully confirmed that Threat 1 is killed. All vehicles are now available for participating in the cooperative area search as a task with low priority. Other tasks involving potential threats will take precedence over the search, requiring the search to reconfigure as often as the shifting resources requires.
Figure 20. (a) Figure 20(a) shows a nice closeup of the UUV kill vehicles being deployed from the USV and setting up to strike Threat 3. (b) The UUVs have hit threat #3 and a USV has been assigned to confirm the hit. Meanwhile, UMV pairs continue to clear threat #1 and confirm the hit on threat #2.

Figure 21. (a) A USV has delivered the UUVs to threat #1 and they are ready to strike. (b) Threat #1 has been hit, and a UMV pair is assigned to confirm the hit. Once the Threat is confirmed killed, the simulation ends.
VIII. Autonomy Levels for Target Intercept and Inspection

One of the main features of USVs is their ability to autonomously intercept and inspect potential threats and suspicious vessels. Achieving accurate intercept is essential to the success of the USV missions. Often, allowing the USV greater levels of autonomy over path planning, trajectory generation and control result in more efficient navigation through dense obstacle fields to the potential threat.

With the terminal trajectory concepts developed, and the levels of autonomy designed for CoMPACT in Chapter 2, the task of integrating the two results in an efficient ship inspection strategy that has a great potential for reducing operator workload. The technology is demonstrated by developing a Graphical User Interface (GUI) so that an operator may assume the role of path planner, controller, trajectory generator, and motion re-planner, according to the defined levels of autonomy. The GUI design allows the operator to control heading and velocity through keyboard inputs, subject to the USV's dynamic constraints. The GUI accommodates multiple USVs which can be commanded to travel in formation. An illustration of the GUI is shown in Figure 22.

![GUI Illustration](image)

**Figure 22.** The levels of Autonomy were implemented in a Graphical User Interface to attribute varying amounts of autonomy to the USVs

A. Levels of Autonomy Simulations for Single USV

As an exercise to demonstrate the levels of autonomy integrated in CoMPACT, the objective of the tests were to maneuver through an obstacle field to reach a target moving in open water, then inspect the moving target.

For a single USV, the autonomy levels varied from the operator being responsible for all decision, control, and planning, to the operator chasing a virtual “carrot,” to turning all control and planning to CoMPACT. Through these simulations, it was observed that unless the obstacle field was very sparse, avoiding the obstacles was very tedious. Furthermore, performing ship inspection is very difficult without autonomous control. Results from the autonomous ship inspection using a single USV are presented in Figures 23(a) through 24(b).

B. Levels of Autonomy Simulations for Two USVs

When the operator is tasked with controlling two USVs, monitoring both vehicle’s progress individually proves a monumental problem. However, when the operator is guiding the USVs manually, if they are virtually linked together through a task to travel in formation, a single operator is still able to manage both vessels. Enabling the vessels to maintain autonomous formations helps ensure successful traversal through the obstacle field, but turning the CoMPACT system to fully autonomous further increases the likelihood of efficient inspection of suspicious moving targets.
Figure 23. (a) Level 1 Autonomy: The USV is driven solely by operator control with no path-planning assistance. Amid obstacle fields, avoiding collisions requires rapt attention. (b) Level 2 Autonomy: CoMPACT evaluates the obstacle field and develops a recommended path for the USV to follow. The moving red star represents the desired position for following the trajectory through a successful inspection.

Figure 24. (a) Level 2 Autonomy with Replanning: Even when a recommendation is given, if the operator decides to take a different course than the recommended trajectory, CoMPACT can re-plan to offer another suitable trajectory from the current position. (b) Level 3 Autonomy: The USV is controlled by CoMPACT along a planned trajectory to complete the inspection mission without colliding into known obstacles. Pop-up threats are still possible hindrances, and the operator is responsible for re-planning given such interruptions.

Since CoMPACT instructs the vessels to travel in tandem when under autonomous control (Level 3, 4, and 5), the results for these cases look the same for a single vessel or multiple vessels. The manual control cases, on the other hand, allow the operator to control the USVs individually or in tandem. Figure 25(a), illustrates how the operator may chose to control the USVs individually during a portion of the task. After each USV has cleared the island, they are again synchronized so the blue USV assumes a position in formation with the green vessel.

For Level 2 autonomy, CoMPACT generates only one trajectory, as it is assumed valuable for both USVs to follow. Furthermore, with the capability of having the operator assume direct control of only one of the USVs and relying upon autonomous formation control for the other USV lightens the burden of the operator through the course of the trajectory.

Figures 25(b) and 26 show details of two USVs tasked to inspect the suspicious vessel, and the representative terminal trajectory they are tasked with. The vessels are initially being controlled autonomously in a formation configuration, and then they break into individual courses to inspect the vessel. In this way, the USVs cooperate autonomously to accomplish the ship inspection task.

IX. Conclusions

This paper presents research and development of an effective Collaborative Mission Planning, Autonomy and Control Technology (CoMPACT) for multiple Unmanned Surface Vehicles. CoMPACT enables autonomous collaboration,
Figure 25. (a) Level 0 Autonomy– Two USVs: In this simulation, the USVs are controlled individually around an island, being navigated through significantly different paths. (b) Level 3 Autonomy– Two USV Inspection: In this simulation, the USVs are controlled autonomously through the obstacle field, then tasked to inspect the suspicious vessel.

Figure 26. Level 3 Autonomy– Two USV Inspection: The two USVs work in concert to perform the inspection. One USV crosses ahead of the suspected target to survey the port side, while the other remains on the starboard side.

coordination & control, and is characterized by a versatile system architecture for different methods of collaborative decision making (e.g. centralized, semi- and fully-decentralized). CoMPACT achieves effective conflict resolution, dynamic mission/mode/task re-planning, and efficient vehicle-to-target assignment using Finite State Machine (FSM) based task execution (state or event-based transitions).

A major benefit of CoMPACT is its ability to decrease operator workload while increasing mission efficiency. Furthermore, the technology is modular in design which allows straightforward expansion, and is well suited for multiple heterogeneous USVs or other types of unmanned vehicles (UAV, UUV, UGV) within the overall architecture.

CoMPACT integrates a conceptual solution of a user-friendly operator interface plug-in that enables quick transitions between varying levels of autonomy. The approach is strengthened by its highly efficient path planning using modified Rapidly-exploring Random Trees (RRT) algorithm, effective theater of operation representation using Quadtrees and grid techniques, vehicle assignment algorithms based on a suitably chosen capability matrix and RRT, highly efficient pursuit algorithms for moving targets through obstacle fields. Other strengths of the research presented here are that higher-fidelity models, communication and sensor models can readily be incorporated into the overall architecture.

The CoMPACT system has been successfully implemented within several complex maritime mission scenarios involving multiple coordinated USV platoons and including area search, mine countermeasures, and anti-submarine warfare.

Acknowledgments

The authors would like to thank Louis Larkin and Eric Holmes from Lockheed Martin Undersea and Security Systems for their technical guidance and support for this project, including scenario definitions and concept of operations.
References


