Coordinated sea rescue system based on unmanned air vehicles and surface vessels

Francisco Fernández Ramírez, David Sánchez Benitez, Eva Besada Portas, José A. López Orozco
Dpto. Arquitectura de Computadores y Automática
Universidad Complutense de Madrid
Madrid (Spain)

***Abstract*-. This paper presents a sea rescue system based on a coordinated team of a sensing/monitoring Unmanned Aerial Vehicle (UAV) and a rescuing Unmanned Surface Vessel (USV) that exploits the measurements provided by the UAV to estimate the castaways position. The system models the castaway location evolution using an Artificial Neural Network (ANN) that is trained before the rescue starts using a map of the sea wind and currents. The UAV predicts the position of the castaways with the prediction ANN and searches the castaways using a controller implemented with another ANN trained with searching behaviors. The USV, with a bigger computational power, incorporates a Particle Filter (PF) to estimate the castaway location. This PF uses the prediction ANN to predict the position of the particles and the measurements of the UAV and USV to update their weights. Finally, the paper presents some simulated experiments that show the whole system performance under different situations.**

Keywords: Shipwreck, Artificial Neural Network, Particle Filters Unmanned Vehicles, Search and rescue.

I. INTRODUCTION

Searching, tracking and rescuing shipwrecked people in the sea is a complex task because the sea wind and currents stochastically spread the castaways in a region. Moreover, as the time elapsed between the shipwreck and the rescue grows, the position of the spread region is moved, its size incremented, its shape modified, … hindering the castaways search and tracking. In other words, the dynamic and uncertain behavior associated to the castaways makes their searching and tracking difficult, and the time needed to accomplish the rescue task make them harder.

The sea rescue problem, unsuccessfully tackled some decades ago due to lack of resources, can nowadays benefit from the combined use of the technological achievements in unmanned vehicles and the theoretical advances on the research areas of control, path planning, cooperation and artificial intelligence. The technological achievements in unmanned vehicles makes them useful and essential in a growing number of military and civil tasks such as reconnaissance, surveillance, aerial photography, etc [1-4]. The theoretical advances in the previously mentioned research areas can make them autonomous, and therefore they can expand the capability of decision of the unmanned vehicles, whose most basic behavior consists on being remotely controlled or following a pre-programmed plan. In this paper, we follow the same approach, and present an autonomous system that exploits the capabilities of unmanned vehicles and of some artificial intelligence techniques in order to search, track and rescue castaways, taking into account the uncertainty and dynamic behavior of the castaways and the sea, and the convenience of minimizing the rescue time.

The complementary capabilities of Unmanned Air Vehicles (UAVs, aircrafts without onboard pilots) and Unmanned Surface Vessels (USVs, ships without onboard captains) make them a good choice to tackle the sea rescue problem. On one hand, the exploration and searching capabilities of the UAVs, based on the fact that they can quickly traverse a wide sea region, make them ideal to collect visual information about the castaways in the predicted rescue region. On the other, the bigger payload of the USVs lets them collect the already located castaways and carry a computational powerful CPU. Hence, this paper proposes a coordinated system for rescuing people at the sea that takes advantage of the high speed and sensing capability of a searching UAV and of the computer power and load capability of a tracking and rescuing USV.

The control commands that guide the UAV and USV towards and inside the castaways spread regions are obtained by an expert system that coordinates the sensing and rescuing capabilities of each vehicle. This coordinated guiding system is distributed between the CPUs placed in each vehicle taking into account their different computational powers and the communication needed to coordinate their behaviors. Finally, as both vehicles already incorporate a stabilization and low level control system that drives the UAV and USV accordingly to high level commands, the designed system only generates the points towards the vehicles are directed.

The autonomous cooperative behavior of the expert system is supported by two artificial intelligence techniques: Artificial Neural Networks (ANNs) and Particle Filters (PFs). ANNs, which are parallel computing structures capable of learning nonlinear complex behaviors, efficiently model the functionality of some parts of the system, taking advantage of their high adaptability, low memory requirements, and real time response capability [5-6]. PFs, which are inference techniques to approximately obtain by sequential Monte Carlo Simulation the probability distribution of the variables of stochastic dynamic systems [7], are used to estimate the location of the castaways that are being spread by the sea wind and currents.

978-1-61284-4577-0088-0/11/$26.00 ©2011 IEEE

In order to design the system we also consider the different phases that appear in the sea rescue task. Initially, when the ship wrecks, the UAV and USV have to be sent towards the expected area where the castaways will be spread due to the sea wind and currents. The castaways expected locations are estimated online, while the vehicles are moving, using a *prediction* ANN, trained with a sea current and wind map, to predict the displacement of the castaways. The higher speed of the UAV makes it arrive to this estimated location quicker than the USV, and once its camera and visual system detect the first castaway, the UAV starts searching other castaways using a *searching* ANN based controller, trained with a set of searching rules that make it explore the region around the already detected castaways to locate undetected ones. When all the castaways are located by the UAV, its behavior changes again to make it transverse the castaways location area, gathering visual information about the position of all the castaways that remain in the water. In the last two phases, called searching and sensing hereafter, the UAV sends its position and the position of the visually observed castaways to the USV to let the USV improve its castaway estimates by means of a PF that uses the prediction ANN and the visual measurements of the UAV and USV. During all the phases, the USV is guided towards its own castaways estimates, improved with the UAV measurements when they are available, until the USV visual system locates one. In that case, the USV stops to collect it, informs the UAV and continues moving towards the estimates of the castaways that remain in the water, until all the castaways are rescued.

This paper details the interaction among the different parts of the expert system, the behaviors in each of the three working modes, and the artificial intelligence techniques, ANNs and PFS, that support these behaviors. It also presents a set of simulations that show their performance under different situations. No real experiments are included, but the promising simulated results make us believe that the approach followed by our coordinated searching and sensing UAV and tracking and rescuing USV system could be extended and applied to real experiments.

II. PROBLEM DESCRIPTION

Searching, tracking and rescuing shipwrecked people with UAVs and USVs is a difficult task due to the dynamics and uncertain behavior associated to the different elements of the system. On one hand, after the ship wrecks and before the UAV and/or the USV arrive at the shipwreck area, the castaways are stochastically moved and spread by the sea and wind currents. On the other, if the expert system succeeds on taking the unmanned vehicles to where the castaways have been translated, the visual systems of the UAV and USV can fail to detect the castaways and provide noisy measurements of their location. Hence, in order to successfully minimize the rescue time, that not only does it harden the searching problem but also decrements the chances of finding the castaways alive, the location of the castaways need to be predicted, at least while they are not located. In order to estimate this location, our system incorporates probabilistic models that define the uncertain behavior of the stochastic elements (castaways, sea and wind currents, and sensors) as well as deterministic models of the unmanned vehicles.

In this section, we define the main variables associated to the different elements of the problem, specify their relations, and describe the elements behavior and models. Additionally, we also introduce the notation used throughout the paper.

Notation

In this paper, a capital italic letter (V) represents a unidimensional variable, a boldface capital italic letter ($\boldsymbol{V}$) a multidimensional one, and a lowercase roman letter (f) a function. Sub-indexes are used to distinguish variables: t associates the variable to the t-th timestep and i to any of its possible realizations. Super-indexes are used to distinguish the elements of multidimensional variables (x and y refer them to Cartesian coordinates, r and θ to polar coordinates) or to relate the variables to the UAV or USV. For example, $M_{t,i}$ represents the i-th variable labeled M at time step t and ${M^x}_{t,i}$ stands for its corresponding x coordinate. Finally, in the deterministic models we represent the relationships among the input and output variables as $O_1 = \mathrm{f}(I_1, I_2, \ldots I_R)$ and in the stochastic ones with the probability distribution $\mathrm{f}(O_1 | I_1, I_2, \ldots I_R)$.

Problem variables

To model the behavior of the problem, we assume that the number of elements needing rescue is fixed and equal to N. Besides, we consider the following variables: $\boldsymbol{U}_t^{UAV}$ and $\boldsymbol{U}_t^{USV}$ to represent the position of the UAV and USV; $\boldsymbol{A}_t^{UAV}$ and $\boldsymbol{A}_t^{USV}$ - the high level control command applied to the UAV or USV; $\boldsymbol{M}_{t,i}$ - the real (unobserved) position of the i-th castaway, $\boldsymbol{E}_{t,i}^{UAV}$ and $\boldsymbol{E}_{t,i}^{USV}$ - the estimated value of the position of the i-th castaway; $D_{t,i}^{UAV}$ and $D_{t,i}^{USV}$ - if the UAV or USV visual system has detected the i-th castway at time t; and $\boldsymbol{S}_{t,i}^{UAV}$ and $\boldsymbol{S}_{t,i}^{USV}$ the location measurement provided by the visual system of each vehicle for the detected i-th castway at time t.

Additionally, $\boldsymbol{U}_t^{UAV} = [U_t^{UAV,x}, U_t^{UAV,y}, U_t^{UAV,\theta}]$, $\boldsymbol{U}_t^{USV} = [U_t^{USV,x}, U_t^{USV,y}, U_t^{USV,\theta}]$, $\boldsymbol{M}_{t,i} = [M_{t,i}^x, M_{t,i}^y]$, $\boldsymbol{E}_t^{UAV} = [E_t^{UAV,x}, E_t^{UAV,y}]$, $\boldsymbol{E}_t^{USV} = [E_t^{USV,x}, E_t^{USV,y}]$, $\boldsymbol{S}_t^{UAV} = [S_t^{UAV,x}, S_t^{UAV,y}]$ and $\boldsymbol{S}_t^{USV} = [S_t^{USV,x}, S_t^{USV,y}]$. Finally, the control commands $\boldsymbol{A}_t^{UAV} = [A_t^{UAV,x}, A_t^{UAV,y}]$ and $\boldsymbol{A}_t^{USV} = [A_t^{USV,x}, A_t^{USV,y}]$ respectively indicate the next waypoint that the UAV and USV have to be driven to by the low level onboard UAV and USV controllers.

Castaways Dynamics

The castaways position $\boldsymbol{M}_{t,i}$ dynamically changes at every time step due to the influence of the sea wind and currents.

To model this behavior, we can use prediction models such as the Mercator Ocean [8] or the Spanish Project ESEOO [9], or dense wind and current displacement maps generated by numerical models such as HIRLAM (High Resolution Limited Area Model, [10]) or CEPPM (Medium Term Prediction European Center). We opt for the second approach within the simulator, but we model the behavior within the predictors of the UAV and USV CPUs with a nonlinear simpler function, learnt from the displacement map used in the simulator. Hence, on one hand, the simulator updates the castaway position $\boldsymbol{M}_{t,i}$ applying to its previous time step location $\boldsymbol{M}_{t-1,i}$ the displacement obtained from the dense wind and current map plus some Gaussian noise to model unknown disturbances. And on the other, the predictors/estimators implemented within the expert system apply a simpler and more efficient deterministic nonlinear model, that will be explained in the following section, to obtain the UAV and USV estimates of the castaways location ($\boldsymbol{E}_{t,i}^{UAV}$ and $\boldsymbol{E}_{t,i}^{USV}$). Finally, the probabilistic function that models the castaway location evolution within the simulator will be represented as r($\boldsymbol{M}_{t+1,i}| \boldsymbol{M}_{t,i}$), while the simpler non-linear model as $\boldsymbol{M}_{t+1,i}$ =g($\boldsymbol{M}_{t,i}$).

Unmanned Vehicle Dynamics

The positions of the UAV and USV, $\boldsymbol{U}_{t}^{UAV}$ and $\boldsymbol{U}_{t}^{USV}$, dynamically change at every time step due to the influence of the control waypoints $\boldsymbol{A}_{t}^{UAV}$ and $\boldsymbol{A}_{t}^{USV}$ obtained from the expert system at every time step. The behavior of both vehicles is modeled by two deterministic non-linear complex functions, $\boldsymbol{U}_{t+1}^{UAV}$=f^{UAV}($\boldsymbol{U}_{t}^{UAV}$, $\boldsymbol{A}_{t}^{UAV}$) and $\boldsymbol{U}_{t+1}^{USV}$=f^{USV} ($\boldsymbol{U}_{t}^{USV}$, $\boldsymbol{A}_{t}^{USV}$), which consider each vehicle characteristics. The two models are specified by complicated ordinary differential equations that represent the non-linear dynamic behavior of each vehicle and their steady-state conditions and definitions. To obtain the position of the vehicles during the simulations, these systems of differential equations have to be integrated, and for that purpose, we use the ode23 or ode34 functions of Matlab. Therefore, these models let us obtain the trajectory followed by the vehicle between *t* and *t+1* and not only its final position at *t+1*.

Our UAV model is based on [11], which presents the non-linear model for a F-16 implemented in Simulink and Matlab. This model is used to run simulations, and apply classic control theory in the linearized version obtained with Matlab. We have used this model [12] as example to create our set of dynamic equations. Our UAV model is slightly simpler than the selected F-16 one. Besides, the parameters that determine the UAV behavior have been changed too, to take into account the manoeuvrability properties of a simpler UAV. However, it is still rather complex: the state vector contains 13 different components (3D positions, angles, velocities, …) and not only the 3 components that the notation section of this paper identifies under $\boldsymbol{U}_{t}^{UAV}$.

Our USV model is based on [13], which presents a non linear dynamic model to describe the behavior of a vessel. Again, we have changed the parameters of the model, to take into account the maneuverability properties of a USV. The state vector of this model contains 12 different components (2D position, yaw, yaw rate, rudder orientation, velocities,…) and not only the 3 components that the notation section of this paper identifies under $\boldsymbol{U}_{t}^{USV}$.

Finally, we also include in the models $\boldsymbol{U}_{t+1}^{UAV}$=f^{UAV}($\boldsymbol{U}_{t}^{UAV}$, $\boldsymbol{A}_{t}^{UAV}$) and $\boldsymbol{U}_{t+1}^{USV}$=f^{USV} ($\boldsymbol{U}_{t}^{USV}$, $\boldsymbol{A}_{t}^{USV}$) the onboard low-level controller and stabilization elements that make the vehicles smoothly follow the waypoints $\boldsymbol{A}_{t}^{UAV}$ and $\boldsymbol{A}_{t}^{USV}$.

Measurement models

Each vehicle is equipped with a camera and visual system that let it detect and locate the position of the castaways that fall inside their field of view. Additionally, we assume that the visual systems let the UAV and USV univocally recognize the measurements related with each castaway. In other words, our system doesn't consider so far the data association problem. We use the following noisy models to obtain each measurement:

- The detection probability of the UAV and USV visual system, represented as h^{UAV}($D_{t,i}^{UAV}$| $\boldsymbol{M}_{t,i}$, $\boldsymbol{U}_{t}^{UAV}$), depends on the castaway and UAV location and on the field of view of the camera. To model failures in the vision system, the probability is high, although not 1, within the centered area of its field of view, and decreases quickly in its corners. That is, the model assumes that the visual system usually detects any castaway inside the main area of its field of view and rarely does it in the corners.
- The detection probability of the USV visual system, represented as h^{USV}($D_{t,i}^{USV}$| dist($\boldsymbol{M}_{t,i}$, $\boldsymbol{U}_{t}^{USV}$)), depends on the distance between each castaway and the USV. To model failures in this vision system, the probability becomes zero outside a selected visual range and decreases with the distance inside it.
- The observed castaway location $\boldsymbol{S}_{t,i}^{UAV}$, provided by the visual system of the UAV when the castaways is detected, is radially displaced with some zero-mean Gaussian noise from the actual castaway location $\boldsymbol{M}_{t,i}$. The probability distribution that represents this behavior is q^{UAV}($\boldsymbol{S}_{t,i}^{UAV}$| $D_{t,i}^{UAV}$, $\boldsymbol{M}_{t,i}$). The USV obtains the castaways location $\boldsymbol{S}_{t,i}^{USV}$ in a similar way, using the probability model q^{USV}($\boldsymbol{S}_{t,i}^{USV}$| $D_{t,i}^{USV}$, $\boldsymbol{M}_{t,i}$).

Relationships among the variables

Finally, although the relationships among the different variables of the problem have been already established by the functions that represent the behavior of the castaways, vehicles and sensors, we want to underline their relationships from the point of view of the expert system, which is responsible of automating the behavior of the vehicles in order to minimize the time needed to locate and rescue the castaways. In order to achieve this objective, the expert system can only directly modify the behavior of the vehicles (i.e. change their positions $\boldsymbol{U}_{t}^{UAV}$ and $\boldsymbol{U}_{t}^{USV}$) by means of obtaining the control signals (i.e. waypoints $\boldsymbol{A}_{t}^{UAV}$ and $\boldsymbol{A}_{t}^{USV}$) that move the UAV and USV towards the castaways location $\boldsymbol{M}_{t,i}$. Once the vehicles are in the correct area, the castaways can be detected, by the visual system onboard each vehicle (i.e. $D_{t,i}^{UAV}$ and/or $D_{t,i}^{USV}$ can become true) and observations about the location of the castaways (i.e. $\boldsymbol{S}_{t,i}^{UAV}$ and $\boldsymbol{S}_{t,i}^{USV}$) can be collected. While the vehicles don't detect and gather any observation about the location of the castaways, the expert system can only drive the vehicles accordingly to the castaways estimates $\boldsymbol{E}_{t,i}^{UAV}$ and $\boldsymbol{E}_{t,i}^{USV}$, which depend on the initial location of the shipwreck $\boldsymbol{M}_{0,i}$ and on the onboard prediction algorithms that use the simpler non-linear model g(.). Once the observed locations become available, they can be used to obtain the control commands and/or to improve the estimates. In order to minimize the discrepancies between the estimations $\boldsymbol{E}_{t,i}^{UAV}$ and $\boldsymbol{E}_{t,i}^{USV}$ and actual castaway locations $\boldsymbol{M}_{t,i}$, the vehicles must be driven towards the real location of the castaways as soon as possible and the model g(.) must be accurate.

The objective of this research is to find efficient controllers that will let the UAV and USV coordinately respond to the evolution of the unknown castaway location ($\boldsymbol{M}_{t,i}$) and to the availability of measurements ($D_{t,i}^{UAV}$, $D_{t,i}^{USV}$, $\boldsymbol{S}_{t,i}^{UAV}$, $\boldsymbol{S}_{t,i}^{USV}$).

III. COORDINATED EXPERT SYSTEM

This section describes the behavior of the complete control system that is distributedly implemented in the UAV and USV onboard CPUs and the simpler non-linear model g(.) used to predict the location of the castaways within the subsystems related to each vehicle.

The whole system overview

At each time step, the high level control commands $\boldsymbol{A}_{t}^{UAV}$ and $\boldsymbol{A}_{t}^{USV}$ obtained by the coordinated expert system must increment the chances of finding never observed shipwrecked and of rescuing the already observed ones. In order to take advantage of the observations as they become available and increase the chances of rescuing all the castaways, the control commands $\boldsymbol{A}_{t}^{UAV}$ and $\boldsymbol{A}_{t}^{USV}$ are calculated in a closed loop. The UAV, which due to its higher speed and field of view can sense the castaways location more often, improves its estimates using the information provided by its own visual system ($D_{i,t}^{UAV}$, $\boldsymbol{S}_{t,i}^{UAV}$). The USV, which moves more slowly and has a smaller field of view, requires the measurements of the UAV in order to improve its estimates about the castaway location. Therefore, the USV uses the information provided by the UAV and USV visual systems ($D_{t,i}^{UAV}$, $\boldsymbol{S}_{t,i}^{UAV}$, $D_{t,i}^{USV}$ and $\boldsymbol{S}_{t,i}^{USV}$). Besides, the USV needs to know the UAV location $\boldsymbol{U}_{t}^{UAV}$ to be able to infer if the UAV should or shouldn't have detected $D_{t,i}^{UAV}$ the *i*-th castaway.

The way of proceeding to obtain $\boldsymbol{A}_{t}^{UAV}$ and $\boldsymbol{A}_{t}^{USV}$ depends on the available information and vehicle type. Before the first shipwrecked is located by the vehicle visual systems, the UAV and USV can only use their estimates $\boldsymbol{E}_{t,i}^{UAV}$ and $\boldsymbol{E}_{t,i}^{USV}$ about the castaways location to decide where to go. Once the first castaways location is observed by the UAV[1] (i.e. $D_{t,i}^{UAV} = true$ and there is one $\boldsymbol{S}_{t,i}^{UAV}$) the UAV can 1) improve the estimates $\boldsymbol{E}_{t,i}^{UAV}$ of the detected castaways and 2) start searching the unobserved castaways using the estimates $\boldsymbol{E}_{t,i}^{UAV}$ of the observed ones. Besides, the UAV can send its visual information and location ($D_{t,i}^{UAV}$, $\boldsymbol{S}_{t,i}^{UAV}$, $\boldsymbol{U}_{t}^{UAV}$) to the USV, to let the USV improve its castaways estimates using the measurements of the UAV, and therefore, to let it benefit from the bigger sensing capability of the aerial vehicle. The UAV maintains its searching behavior until it has located all the castaways, and afterwards it starts a sensing/monitoring behavior that makes it transverse the castaway location area, gathering visual observation of the positions of all the castaways that stay in the water. Simultaneously, the UAV sends its visual information and location ($D_{t,i}^{UAV}$, $\boldsymbol{S}_{t,i}^{UAV}$, $\boldsymbol{U}_{t}^{UAV}$) to the USV to let the USV improve its castaways estimates $\boldsymbol{E}_{t,i}^{USV}$. The USV updates its castaway estimated positions $\boldsymbol{E}_{t,i}^{USV}$ during the three behavior phases (prediction, searching and sensing) of the UAV using the prediction model g(.) and all the available observations ($D_{t,i}^{UAV}$, $\boldsymbol{S}_{t,i}^{UAV}$, $D_{t,i}^{USV}$, $\boldsymbol{S}_{t,i}^{USV}$). Besides, while the USV doesn't detect a castaway, the USV moves towards the castaway estimated location $\boldsymbol{E}_{t,i}^{USV}$ and when it detects any, it stops moving until it rescues it. Once all the observed castaways are collected, the USV informs the UAV and continues moving towards the next estimated castaway location $\boldsymbol{E}_{t,i}^{USV}$ until all the castaways have been rescued.

Figure 1 schematizes this whole system overview, representing for each vehicle the main modules that implement the expert system behavior related to it, the flow of information among the different modules and some of the functional dependencies among the different variables. The UAV subsystem has several modules, one in charge of estimating the location of the castaways, three controllers related with each working mode, and another in charge of selecting which of the controllers is run. The USV subsystem estimates the location of the castaways with a module that incorporates the prediction model p(.), the sensing information and sensing models, and obtains its control commands accordingly to the castaways estimates $\boldsymbol{E}_{t,i}^{USV}$. The elements belonging to each subsystem are explained in more detail in sections IV and V, except the onboard prediction model g(.) that is used by both subsystems, and which is explained in the current section.

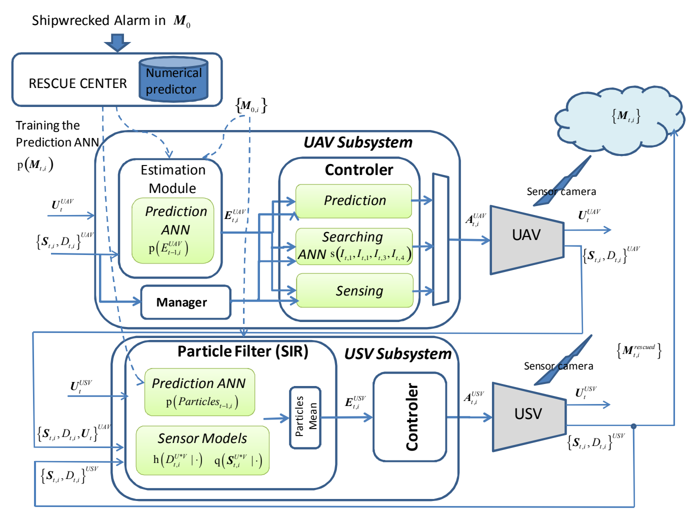

Figure 1. Complete coordinated system

The onboard prediction model $\boldsymbol{M}_{t+1,i}$ = g($\boldsymbol{M}_{t,i}$)

To be able to estimate the locations of the castaways onboard taking into account the measurements and the position

[1] The UAV usually arrives and observes first the castaways due to its higher speed.

where the ship wrecked $\boldsymbol{M}_{o,i}$, both subsystems require a prediction model – at least while no measurements are available- that requires low memory and computational resources. In order to achieve this requirement, we decided to implement it as the model represented in Fig. 2 that obtains $\boldsymbol{M}_{t+1,i}$ from $\boldsymbol{M}_{t,i}$ using the prediction ANN p(.) to generate the displacement $\Delta\boldsymbol{M}_{t+1,i}$.

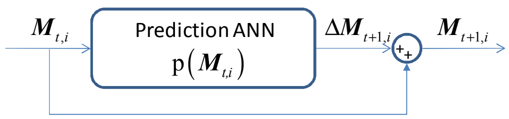

Figure 2. Prediction ANN

This prediction ANN, whose input is a location and whose output is a displacement, is trained with the displacements that the dense wind and current map model implemented in the castaways simulator generate for different points of the sea. So, the prediction ANN efficiently generalizes the behavior of the set of training pairs (position, displacement) to any other point within the training region. Additionally, to speed up the operations of this ANN within each subsystem, we implement it with a feedforward ANN with two inputs $[M^x_{t+1,i}, M^y_{t+1,i}]$, two outputs $[\Delta M^x_{t+1,i}, \Delta M^y_{t+1,i}]$, and three layers with only two neurons in the input layer, four in the hidden and two in the output. Its training time is small too: a convergent ANN is usually available in only 5 minutes using a Pentium Core Duo. So, we train it offline, while the UAV and USV get ready for its mission, and load it in the predictor implemented within each subsystem before the mission starts. In the top of Fig. 1, we also schematize this initial training and loading step.

Finally, it is worth highlighting that as we can't observe the actual castaways positions $\boldsymbol{M}_{t+1,i}$, the prediction ANN is applied to the variables that each subsystem estimator uses to update the estimations $\boldsymbol{E}^{UAV}_{t,i}$ and $\boldsymbol{E}^{USV}_{t,i}$. Besides, a more detailed description of this ANN can be found in [14].

IV. UAV Control Subsystem

In order to obtain the control commands $\boldsymbol{A}^{UAV}_{t}$ that displace the UAV towards and inside the castaway location area, the UAV expert subsystem estimates the castaways positions $\boldsymbol{E}^{UAV}_{t,i}$ directly with the onboard prediccion ANN and implements three different controllers that drive the UAV accordingly with the estimates $\boldsymbol{E}^{UAV}_{t,i}$ and the behavior required in the prediction, searching and sensing phases. The subsystem, which is schematized in Fig. 3, also includes a controller manager that activates the controller in each of the phases. This section describes all these elements. For a more detailed description of the two first controllers the reader is referred to [14], although he/she must notice that the searching controller of our previous work (called sensing in [14]) uses the measurements provided by a beacon, while the searching controller presented in this paper uses the UAV castaways estimates $\boldsymbol{E}^{UAV}_{t,i}$.

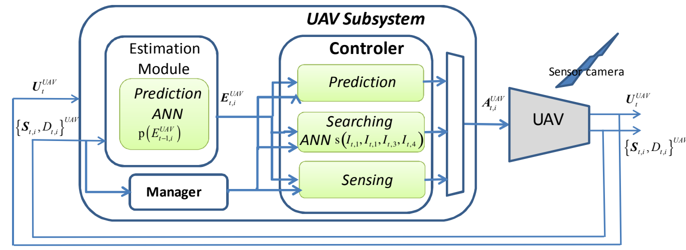

Figure 3. UAV Control Subsystem

The UAV estimation module

Due to the low CPU capabilities of the UAV, we obtain the castaway estimates $\boldsymbol{E}^{UAV}_{t,i}$ using the following two steps:

1) We apply directly to the previous time step estimates $\boldsymbol{E}^{UAV}_{t-1,i}$ the onboard predictor ANN to obtain the displacement $\Delta\boldsymbol{E}^{UAV}_{t,i}$ that the estimate location will undertake, and we make $\boldsymbol{E}^{UAV}_{t,i} = \boldsymbol{E}^{UAV}_{t-1,i} + \Delta\boldsymbol{E}^{UAV}_{t-1,i}$
2) When a new castaway location has been measured by the UAV visual system (i.e. when $D^{UAV}_{t,i} = true$ and $\boldsymbol{S}^{UAV}_{t,i}$ exists) we modify the just updated value $\boldsymbol{E}^{UAV}_{t,i}$ with the current measurement (i.e. we make $\boldsymbol{E}^{UAV}_{t,i} = \boldsymbol{S}^{UAV}_{t,i}$).

This way of proceeding resets the predicted estimated values $\boldsymbol{E}^{UAV}_{t,i}$ to the current noisy measurement $\boldsymbol{S}^{UAV}_{t,i}$. Therefore, the measurement update (second step) is not accurate, but if the gaussian noise associated to the measurement in the model $q^{UAV}(\boldsymbol{S}^{UAV}_{t,i} \mid D^{UAV}_{t,i}, \boldsymbol{M}_{t,i})$ is small, the system prefers to assume its associated error to the one accumulated by the ANN prediction during the whole estimation. Besides, when the measurement is not available because the detection probability model $h^{UAV}(D^{UAV}_{t,i} \mid \boldsymbol{M}_{t,i}, \boldsymbol{U}^{UAV}_{t})$ has produced a false negative, we don't change the predicted (first step) $\boldsymbol{E}^{UAV}_{t,i}$ value. Moreover, it is a really efficient way to update the estimates and as far as it lets the UAV field of view, significantly bigger than the measurement variance, observe the castaways position again, the value can be reset often. Finally, the searching and sensing behaviors that we will explain in the remaining of this section let the UAV cope with the measurement noise in the majority of the cases.

The UAV prediction controller

Before the UAV arrives to the position where the castaways are located, it can only update its estimate based on the predicted ANN and it cannot provide any location measurement $\boldsymbol{S}^{UAV}_{t,i}$ to the USV. Therefore, the UAV needs to be conducted to its best guess about the castaway location, i.e. the castaways estimates $\boldsymbol{E}^{UAV}_{t,i}$, as soon as possible. The easiest way to proceed is to conduct the UAV towards the mean value of the estimates $\boldsymbol{E}^{UAV}_{t,i}$, and therefore the prediction controller makes $\boldsymbol{A}^{UAV}_{t} = \sum_{i=1:N} \boldsymbol{E}^{UAV}_{t,i}/N$. This way of proceeding adapts the trajectory of the UAV to the estimates updates, and therefore it is crucial to have trained the prediction ANN properly. If no castaways are found in the expected location area, the UAV stays in this mode, following the mean estimated value for a while, until it contacts the rescue center to require some input to reset the estimates $\boldsymbol{E}^{UAV}_{t,i}$ accordingly.

Once the UAV visual system detects the first castaway (i.e. obtains the first $D_{t,i}^{UAV} = true$ and $\boldsymbol{S}_{t,i}^{UAV}$), it starts working in searching mode and sends all its visual information about the castaways to the USV to let it improve the values of the USV estimates $\boldsymbol{E}_{t,i}^{USV}$ with the UAV visual information. Therefore, the UAV starts cooperating with the USV by becoming a quickly moving remote USV sensor. This cooperative behavior remains when the UAV starts working in sensing mode.

The UAV searching controller

Once the UAV visual system detects the first castaway (i.e. obtains the first $D_{t,i}^{UAV} = true$ and $\boldsymbol{S}_{t,i}^{UAV}$), its prediction module can use it to update its corresponding castaway estimate $\boldsymbol{E}_{t,i}^{USV}$ and the UAV needs to start searching for unobserved castaways. As the searching behavior is more complex than the prediction one, we implement it with a searching ANN, whose inputs are related with the estimates values, and whose output is related to the velocity $O_t^{v}$and orientation O_t^{θ} that the UAV has to follow. To evict the errors accumulated by the prediction of the never observed castaways, the ANN inputs only consider the estimated values $\boldsymbol{E}_{t,i}^{UAV}$ of the castaways that have been already observed in the past and that remain in the water (i.e. *{* $\boldsymbol{E}_{k,i}$*|any(*$D_{k\leq t,i}$*=true^i is not rescued)})*. Besides, we compact the information related to the estimation of the already observed castaways in the four following input variables:

- $I_{t,1}$, the orientation of the mean direction of the estimates of the castaways that have been already observed in the previous time step.
- $I_{t,2}$, the distance of the UAV to the mean estimated location of the castaways that have been already observed in the previous time step.
- $I_{t,3}$, the orientation of the mean direction of the estimates of all the already observed castaways.
- $I_{t,4}$, the percentage of already located castaways.

The velocity and orientation outputs are translated into high level commands $\boldsymbol{A}_t^{UAV}$ before being applied to the UAV. In order to train this ANN we obtain inputs and output pairs from an expert rule-based system that implements different searching behaviors. A more detailed description of this learning process is presented in [14]. Besides, in order to speed up the behavior of the searching ANN, we implement it with a feedforward neural network with four inputs $[I_{t,1}, I_{t,2}, I_{t,3}, I_{t,4}]$, two outputs $[O_t^{v}, O_t^{\theta}]$, and three layers with only four neurons in the input layer, eight in the hidden and two in the output.

When all the castaways have been observed once, the searching behavior is replaced by the sensing/monitoring one implemented by the following controller.

The UAV sensing/monitoring controller

Once the UAV has observed all the castaways at least once, its behavior changes again in order to monitor and sense the location of all the castaways that remain in the water as often as possible. For this purpose, it implements a really simple behavior that makes it move towards the estimated position $\boldsymbol{E}_{t,i}^{UAV}$ of the closest castaway that is not currently being observed. This way of proceeding makes it collect measurements of many castaways often and send multiple sensed information ($D_{t,i}^{UAV}$ and available $\boldsymbol{S}_{t,i}^{UAV}$) to the USV to let it improve its castways estimates $\boldsymbol{E}_{t,i}^{UAV}$ with sensorial information as often as possible.

The manager module

This module is in charge of 1) deciding when the UAV has to be driven by each of the UAV controllers and of 2) letting the UAV basic prediction module know when a castaways has been collected by the USV and doesn't require being tracked any longer.

V. USV CONTROL SUBSYSTEM

In order to obtain the control commands $\boldsymbol{A}_t^{USV}$ that displace the USV towards and inside the castaway location area, the USV expert subsystem, which is schematized in Fig. 4, estimates the castaways positions $\boldsymbol{E}_{t,i}^{USV}$ with all the available sensing information ($D_{t,i}^{UAV}$, $\boldsymbol{S}_{t,i}^{UAV}$, $D_{t,i}^{USV}$ and $\boldsymbol{S}_{t,i}^{USV}$) using a PF that incorporates the onboard prediction ANN and the sensing probability models of the UAV and USV. Additionally, the control subsystem incorporates a controller that, while the USV is not detecting a castaway, moves the USV towards the closest castaway estimate $\boldsymbol{E}_{t,i}^{USV}$ and, when the USV has observed some castaways, stops the USV to be able to collect them safely. This section describes the USV PF estimator and controller in detail.

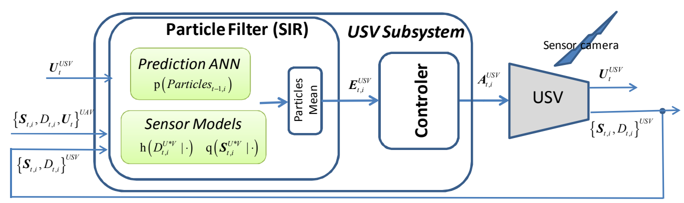

Figure 4: USV Control Subsystem

The USV PF estimation module

Taking advantage of the more powerful CPU of the USV, the USV estimator module updates the values of the castaway location estimates $\boldsymbol{E}_{t,i}^{USV}$ by means of a set of N PFs that use the onboard prediction ANN as the castaway prediction model, the sensing probabilities models of the UAV and USV to calculate the likelihood of the measurements, and the measurements provided by the UAV and USV ($D_{t,i}^{UAV}$and $D_{t,i}^{USV}$, and available $\boldsymbol{S}_{t,i}^{UAV}$ and $\boldsymbol{S}_{t,i}^{USV}$).

In this paper, each PF updates the estimate $\boldsymbol{E}_{t,i}^{USV}$ related to each castaway independently, because the selected prediction ANN based model obtains the displacement undertaken by the castaway taking into account only the previous position related with that castaway. Other models where the same displacement is applied to all the castaways, such as the ones explained in [15] require a unique PF for all the castaways. If that is the case, a PF such as the Parallel PFs in [16] can provide good estimation with a low number of particles.

Each of the PFs implemented in the USV estimation module is a Sampling Importance Resampling (SIR) algorithm [7]. SIR predicts the new values of the M particles associated with each castaway location using the prediction ANN p(.) to obtain the displacement undertaken by each particle and some additive zero mean Gausian noise to represent unmodeled disturbances. After the prediction step is carried out, SIR updates the weights of each particle with the product of the likelihood of all the available measurements given the location of the vehicles and of the particles associated to each castaway. Finally, the particles values are resampled using the weights of the particles and their weights set to 1/M. These three steps (prediction, weight update, and resampling) let SIR implement the Importance Sampling and Weighted Resampling steps serially [7], and place the particles in the sea taking into account first the past history of the particles and secondly the measurement likely regions.

It is worth noting that we have implemented the USV estimator by means of PFs because the prediction ANN is a non linear model and the probability distributions associated to the detection models of each visual system are not Gaussians. Hence, the selected estimator has to be able to use different types of probability models, a property associated to PFs. Taking into account the measurement probability models of the problems, which are used within the PFs, those particles which are not likely according to the detection model $h^{U*V}(D_{t,i}^{U*V}|.)$, and/or the observed location $q^{U*V}(\boldsymbol{S}_{t,i}^{U*V} \mid D_{t,i}^{U*V}, \boldsymbol{M}_{t,i})$ have a low chance to survive in the resampling step, making the PF adapt the location of the particles of each castaway to all the sensed information ($D_{t,i}^{UAV}$ and $D_{t,i}^{USV}$, and available $\boldsymbol{S}_{t,i}^{UAV}$ and $\boldsymbol{S}_{t,i}^{USV}$).

Finally, in order to estimate the castaway location $\boldsymbol{E}_{t,i}^{USV}$ from the values of the particles, we calculate the mean value of all the particles associated to each of the castaways.

The USV controller

The USV controller implements a behavior that chases the castaways estimated locations $\boldsymbol{E}_{t,i}^{USV}$ obtained by each PF with the sensorial information provided by the UAV and USV. Therefore, it benefits for the improvement that the measurements provided by the bigger field of view and higher speed of the UAV.

So far, we implement a really simple behavior to chase the estimates: the USV moves towards the closest estimated value, with the purpose of rescuing as soon as possible those people we expect to arrive at sooner. The USV maintains that behavior until its visual system lets it observe a castaway. When that happens, the USV is stopped to be able to rescue the observed castaways safely. During the simulations, we calculate the time that the USV has to be stopped based on the number of castaways that fall under the USV field of view. Once the USV has rescued all the castaways that it is observing, the USV controller makes it chase the closest estimated value of the castaways that remain in the water.

Whenever a castaway is rescued, the USV subsystem disables its corresponding PF and informs the UAV. The rescuing information affects the UAV behavior in any of its working modes: in the prediction mode, there is an element not longer used to calculate the mean, in the searching mode an estimate that can't be longer considered in the four inputs of the searching ANN, and in the sensing mode a castaway that won't fall outside the UAV field of view any longer. Therefore, both vehicles change the behavior of the other: the UAV sends measurements that let the USV PF improve its castaways estimates, and the USV informs of rescued castaways that modify the UAV behavior.

VI. SIMULATIONS

In this section we illustrate the advantages of the coordinated behavior achieved by our two expert subsystems in two simulated experiments. In the first, the USV has to follow the castaway estimates using only its own measurements, because the UAV never takes off. In the second, both vehicles are sent to cooperate in the rescue, and therefore the USV benefits from the continuous flow of sensorial information provided by the UAV. In both experiments the number of castaways N=10 and in the second, the number of particles within each PF M=100.

To run the experiments, we include our expert system in a MATLAB simulator that is also responsible of calculating at every time step the 'real' castaways and vehicles positions. The castaways positions, obtained with the simulator prediction model $r(\boldsymbol{M}_{t+1,i} \mid \boldsymbol{M}_{t,i})$ instead of the onboard ANN based predictor $p(\boldsymbol{M}_{t,i})$, are used to obtain the shipwrecked measurements ($D_{t,i}^{UAV}$, $\boldsymbol{S}_{t,i}^{UAV}$, $D_{t,i}^{USV}$ and $\boldsymbol{S}_{t,i}^{USV}$). The vehicle positions ($\boldsymbol{U}_t^{UAV}$ and $\boldsymbol{U}_t^{USV}$) are obtained with the vehicle dynamics and onboard controller models ($\boldsymbol{U}_{t+1}^{UAV} = f^{UAV}(\boldsymbol{U}_t^{UAV}, \boldsymbol{A}_t^{UAV})$ and $\boldsymbol{U}_{t+1}^{USV} = f^{USV}(\boldsymbol{U}_t^{USV}, \boldsymbol{A}_t^{USV})$). Besides, the simulator also randomly generates the initial positions of the shipwrecked in a small area around the initial wrecked position $\boldsymbol{M}_{0,i}$. Therefore, the simulator closes the control loop: from the point of view of the expert systems it applies its outputs ($\boldsymbol{A}_t^{UAV}$ and $\boldsymbol{A}_t^{USV}$) to the UAV and USV to generates its inputs ($\boldsymbol{U}_t^{UAV}$, $\boldsymbol{U}_t^{USV}$, $D_{t,i}^{UAV}$, $\boldsymbol{S}_{t,i}^{UAV}$, $D_{t,i}^{USV}$ and $\boldsymbol{S}_{t,i}^{USV}$) considering the vehicles and castaway simulated positions.

The results of the two experiments are presented in the following figures, where the trajectories of the USV and UAV (only for the second case) are represented by the magenta and red lines. The figures also represent with different lines, which follow the color chart at the left of the figures, the real trajectories of the castaways and with colored dots[2] the values of the particles associated to each of the castaways at the final time step or when the castaway was rescued. Finally, the black dots represent the real position of the castaway in the last time step or when they were rescued, and the dark blue ones the castaway location estimated by the USV as the mean value of its particles, and the red ones (only for the second one) the castaways location estimated by the UAV. Finally, the colored circles that appear around the black dots represent the place where the castaways are rescued.

[2] As multiple particles fall in the same region of the space, the shape and size of each colored area is related with the particle dispersion.

Rescue simulation without UAV

In this simulation, the ship wrecks at the upper left corner of Fig. 5 and the rescue centre is placed 8000 m apart in the upper right corner. Therefore, the castaways are adrift by the simulator while the USV follows the castaway location estimates, obtained with the PFs placed in the USV expert subsystem, which can only predict the new values of the particles, using the prediction ANN model while there are no location measurements available. Following these estimates, the USV reaches the area where the castaways are placed, and is able to rescue 6 of the 10 (60%) with only its visual information. While the USV is rescuing the located castaways, the others keep on moving and dispersing, and therefore, the USV has to follow their estimates. With this chasing behavior, a bit later is able to locate and rescue two other castaways (reaching a 80% succeed rate), while the estimates of the remaining accumulate uncertainty and errors due to the lack of measurements about them.

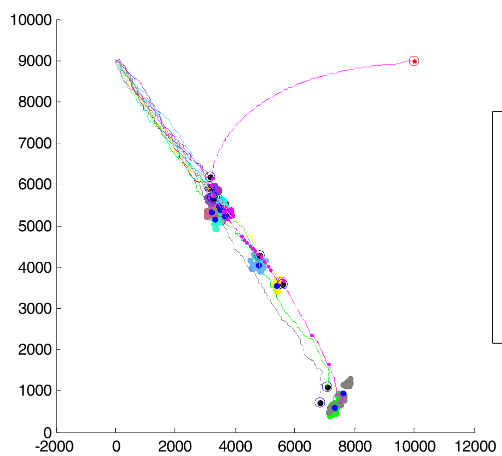

Figure 5: Rescue simulation without UAV: global view

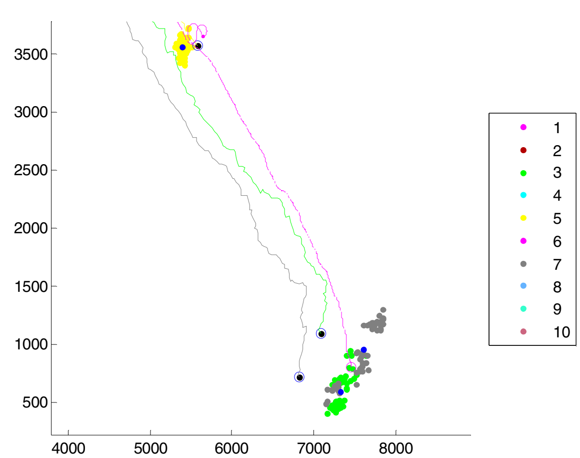

Figure 6. Rescue simulation without UAV: final view

The remaining of the simulation is zoomed in Fig. 6, where the 8th rescued castaway appears in its right top. Due to the time elapsed between the shipwreck and the rescue of the 8th castaway, the estimates of the castaways that remain in the water have accumulated an error that makes the USV follow them (the two blue points) leaving the real castaways (black points) behind because the USV range of view doesn't let it observe the castaways real locations using the chasing behavior implemented by the USV controller. The error keeps on accumulating, and the USV can't finish the rescue, so the simulation is stopped.

Rescue simulation with UAV

In order to avoid the behavior observed in the previous simulation, although we could change the chasing behavior of the USV by a more complex one, we opt to improve the USV estimates with the sensorial information provided by a searching/ sensing UAV, which can take advantage of its higher speed and bigger field of view. So, we repeat the simulation including the UAV, and observe how the USV is able to rescue all the castaways quicker. The results are presented in Fig.7, where the 3 phases of the UAV are also framed and labeled.

Originally, while the UAV trajectory doesn't arrive at the area where its basic predictor locates the castaway, the UAV follows its predicting behavior, that is, it moves towards the mean of its predicted values. As soon as the UAV locates the first castaway, its behavior changes to be driven by the searching ANN that makes the UAV move erratically and/or circularly around the positions of the observed castaways. The duration of this behavior, that is presented in bigger detail in Fig. 8, depends mainly on the dispersion of the castaways because the error accumulated by the observed castaways, whose estimated values are the only ones used by the searching ANN, is small. Once all the castaways have been observed once by the UAV visual system, the UAV behavior changes again into its sensing mode, which moves the UAV towards the closest estimate of the currently unobserved castaways, causing the zigzag movement presented in Fig. 9. The zigzag size depends on the dispersion of the UAV castaway estimates, as the 3 framed regions of Fig. 9 show.

The behavior of the USV is improved by the UAV measurements, because the USV PFs are able to improve the castaway estimates long before the USV is able to intercept the estimated region and find itself any castaway. So, when the UAV is present, the USV chasing behavior makes the USV follow estimates that have a smaller error that when the UAV is not present. Hence, the USV basic chasing behavior lets it locate the castaways more easily when the USV uses the UAV as a quick remote searching and sensing complement. Finally, it is worth remarking that the USV cannot rescue the castaways significantly quicker due to its dynamics and limited speed, although some improvement could be obtained by substituting the behavior implemented in the USV controller by a more intelligent one, following the approach used in the UAV ANN-based searching controller.

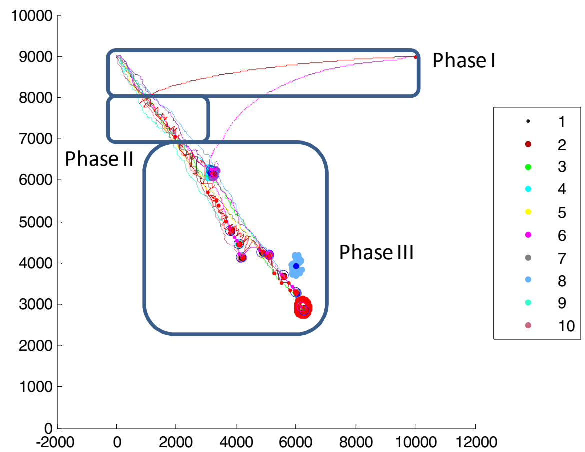

Figure 7: Rescue simulation with UAV: global view

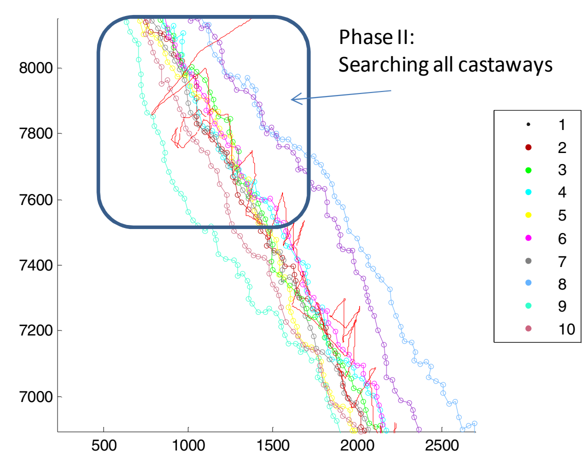

Figure 8. Rescue simulation with UAV: searching phase

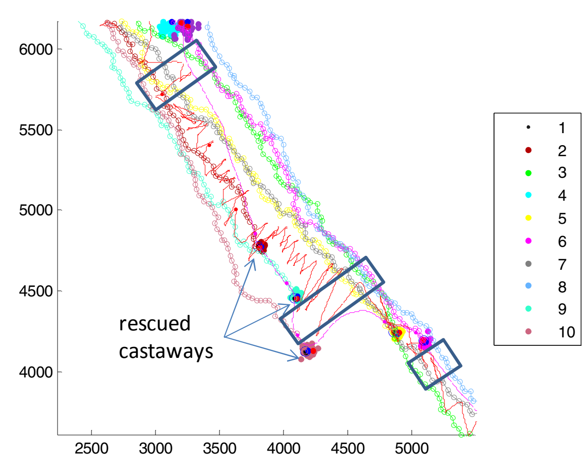

Figure 9. Rescue simulation with UAV: sensing phase

VII. Conclusion

In this paper, we present a sea rescue coordinated system, where the USV in charge of rescuing the castaways benefits from the information provided by a searching and sensing UAV, which is capable of locating the castaways quicker than the USV. The coordinated use of both vehicles reduces the search time, because the USV can predict the castaway positions better, using the information provided by its remote sensing UAV and its more powerful CPU.

Each of the vehicles is controlled by a subsystem, whose algorithms incorporate some artificial intelligent techniques. Both subsystems use an ANN trained with sea wind and currents information, to predict the location of the castaways. Additionally, the UAV subsystem uses another ANN, which has been trained to efficiently implement a searching behavior that lets the UAV locate unobserved castaways around the observed ones. And the more powerful USV CPU estimates the location of the castaways using a group of PFs that take advantage of the sensorial information provided by both the UAV and USV.

The two subsystems can work in real-time, using the information that the vehicles will have available in real world experiments. So far they are working successfully with really simple defined behaviors. In the future, we are planning to expand the behaviors of both vehicles, following the philosophy of the searching UAV ANN-based controller, in order to minimize further the rescue time. For instance, the USV behavior could benefit from a controller that takes into account the dispersion of the particles to optimize the order followed to rescue the castaways.

Acknowledgment

This work is supported by the Spanish Grants DPI-2009-14552-C02-01.

References

[1] Besada-Portas, E. de la Torre, L. de la Cruz, J. M. de Andrés-Toro, B. (2010) Evolutionary Trajectory Planner for Multiple UAVs in Realistic Scenarios. IEEE Transactions on Robotics 26 (4), 619 – 634.

[2] Farzad Kamrani and Rassul Ayani (2009). UAV Path Planning in Search Operations, Aerial Vehicles, Thanh Mung Lam (Ed.), Available from: http://www.intechopen.com/articles/show/title/uav_path_planning_in_search_operations [Accessed 10 December 2010]

[3] Howden D. and Hendtlass T. (2008). Collective intelligence and bush fire spotting, in Proc. Genet. Evol. Comput. Conf., pp. 41–48.

[4] Rubio J.C., Vagners,J, and Rysdykz R. (2004) Adaptive Path Planning for Autonomous UAV Oceanic Search Missions, AIAA 1st Intelligent Systems Technical Conference, 20 - 22 September, Chicago, Illinois, pp 1-10.

[5] Haykin, S. (1999). Neural Networks: A comprehensible Fundation. Second edition, Prentice-Hall New Jersey, USA.

[6] Patterson D.W. (1996) Artificial Neural Networks. Theory and Applications, Prentice Hall.

[7] Doucet, A., Freitas, N., Gordon, N. (eds.) (2001). Sequential Monte Carlo methods in practice. Springer, Heidelberg

[8] MERCATOR, http://www.mercator.eu.org/html/mercator/index_en.html [Accessed 18 December 2010]

[9] Alvarez Fanjul, E. (2005). Establecimiento de un Sistema Español de Oceanografía Operacional. Revista de Puertos del Estado. March 2, 57-64.

[10] Gómez Lahoz, M. and Carretero Albiach, J. C. (2005). Wave forecasting at the Spanish coasts. Journal of Atmospheric & Ocean Science, 1741-7546, Volume 10 (4), 389-405.

[11] Software Enable Control home page, http://www.aem.umn.edu/people/faculty/balas/darpa_sec/SEC.Software.html#F16Manual [Accessed 30 March 2011].

[12] Richard S. Russell. "Non-linear F-16 simulation using Simulink and Matlab". User Manual. Available from: http://www.aem.umn.edu/people/faculty/balas/darpa_sec/software/F16Manual.pdf [Accessed 30 March 2011]

[13] Fossen, T.I, "Guidance and control of ocean vehicles". Pp. 173-175, John-Wiley &Sons, 1994.

[14] Francisco Fernández Ramírez, Eva Besada-Portas, David Sánchez Benito, José A. López-Orozco "Expert guidance system for unmanned aerial vehicles based on artificial neural networks", Accepted for publication in an special issue of the Journal of Maritime Research.

[15] Besada-Portas, E., Pliz, S.M., de la Cruz, J.M., Lane, T.(2009). "Parallel subspace sampling for particle filtering in dynamic bayesian networks". In: Proc. of ECML 2009.

[16] Besada-Portas, E., Pliz, S.M., de la Cruz, J.M., Lane, T. (2010). "Adaptive parallel/serial sampling mechanisms for particle filtering in dynamic bayesian networks". In: Proc. of ECML 2010