ABSTRACT
The problem addressed focuses on the development of navigational control and sensing techniques in the framework of fusion and distributed control pertinent to the problem of deploying autonomous unmanned surface platforms and clusters of platforms for mission applications in support of future naval capability. For example, the US Navy FORCEnet architecture consists of thousands of fully netted unmanned vehicles and unattended sensors. The baseline considerations in deploying autonomous platforms are: control of autonomous platforms and platform clusters; sensing capability for environment/mission; and the intelligent fusion of sensor, mission and control data. Key to the success of this concept is the ability to maintain functionality while restricting navigational and sensor information bandwidths to acceptable levels. A vector integration to endpoint model is employed as a distributed automata agent to satisfy the above operational and information rate concerns. In its basic form, the vector integration model is a real-time neural network which employs fuzzy logic to fuse all available information in autonomously generating appropriate navigation control with acceptable information levels. In addition, this approach to autonomy functions well in a scenario where both the operating environment and the mission objective vary with time. Simulation results illustrate the viability of the approach and inherent desirable characteristics for collision/obstacle avoidance, target interception and target following with standoff.

INTRODUCTION
The envisioned US Navy future naval capability incorporates the development of autonomous unmanned platforms and clusters of platforms. The autonomous platform architecture may be comprised of thousands of fully netted unmanned vehicles and unattended sensors as illustrated in Figure 1. The success of this concept depends on the capability to maintain a functional communication network without bandwidth overload. The intercommunication framework considered for the present autonomous vehicle network problem is depicted Figure 2. A mission control function communicates targets and actions to all platforms. In addition, the platforms selectively communicate amongst each other in accomplishing collision avoidance, survivability and coordinated action.

Crucial to the realization of such a system is the development of the capability of maintaining real-time sensor data turnaround in concert with autonomous formation control driven by mission requirements, Brock (1992), Egerstedt and Hu (2001), Schouwenaars (2001). Neural network based techniques have also been applied to the problem of path planning navigation by Bullock (1988), Grossberg (1987) and Konyk (1993). These methods have exhibited the capability to reduce computational load by directly generating a time evolution of trajectories without requiring precomputation. Fuzzy logic
applied to controller design has exhibited the capability to quantitatively fuse data from diverse sources, Nauck (1997) and Passino (1998). The problem addressed herein is one of joining the computationally efficient neural network approaches to navigation with the fusion capability of fuzzy logic to produce a distributed autonomous control agent capable of adapting in real-time to accomplish mission goals, survivability, avoidance and interception without excessive information transfer rates.

Fundamentally, an approach for data fusion is sought which may be incorporated for adaptive distributed autonomous control. Potential naval applications include the implementation of these techniques in unmanned surface vehicle control (USV), mine and undersea warfare (MUW), anti-submarine warfare (ASW), and anti-terrorism/force protection (ATFP).

**DISTRIBUTED AUTOMATA NAVIGATION AGENT**

A vector-integration-to-endpoint model of the neural dynamics of planned arm movements comprises the distributed automata navigation agent and consists of two coupled equations which may be directly applied to autonomous platform formation control. These equations form a basis for sequentially updating the present platform coordinate position in real-time to attain a desired target coordinate for the platform. These position updates are driven by any mismatch between the present coordinate position of the platform and that desired.

### Navigation to Target Path Generation

In its basic form the vector integration to endpoint model is a real-time neural network which, for the $i^{th}$ entry of the coordinate vector, may be expressed as, Grossberg (1987) and Bullock (1988):

$$\frac{d}{dt} V_i = \alpha(-V_i + T_i - P)$$

(1)

$$\frac{d}{dt} P_{i} = G[\max(V_i,0)]$$

(2)

where $T_i$ is the $i^{th}$ entry of the desired position coordinate vector $T$ of the platform; $P_i$ the $i^{th}$ coordinate of the present position command coordinate vector, $P$; $V_i$ the mismatch vector; and $G$ is a possibly time varying velocity command termed the "GO" signal.

The first equation generates a nonzero mismatch vector $V$ whenever the present position command coordinate vector $P$ differs from the desired position vector $T$. The mismatch vector is continually added to the present position command vector, $P$, until the present position command vector equals the desired position vector and a zero valued mismatch vector results.

The above trajectory generator permits multiple platforms to operate synchronously in time even though the coordinate distances from present position to final position may be different for each coordinate. Furthermore, the gain $G$, the "GO" signal, regulates the speed at which the path to the desired target position evolves. Even though $G$ varies with time synchronous platform operation is maintained. Thus, the vector integration to endpoint model is capable of
exhibiting many types of arm motion characteristics. These include the capacity to speed up, slow down and fast freeze anywhere along a trajectory path. This is accomplished by simply adjusting the $GO$ signal gains in the model since the value of these gains determines the rate at which the trajectory evolves in real-time. Increasing the gains results in faster motion, decreasing them slows motion and setting them to zero causes trajectory evolution to freeze. Such a feature is attractive in an environment where obstacles may suddenly appear along a trajectory.

The gain $G$ also provides a means of regulating the update rate of the present position vector, $P$, by employing feedback from position sensors in the platform. The gain $G$ is adjusted to maintain a desired tracking error. If the actual tracking error exceeds that desired, the gain $G$ may be reduced thereby sacrificing speed and improving accuracy. On the other hand, if the tracking error is less than that desired, the gain $G$ may be made larger to increase speed while sacrificing accuracy.

### Navigation to Target Path Tracking

In order to assure satisfactory tracking accuracy, the discrepancy between trajectory generator present position commands and the actual position of the autonomous platform must be established and employed by the trajectory generator. A structure for an automatic learning process employs error signals, derived from discrepancies between the autonomous platform actual position vector and that seen by the trajectory generator, to adaptively tune the present position vector until it coincides with the actual position. The gains $G$ and $GP$ enable external intelligent control of the update and learning functions.

#### NEURO-FUZZY OBSTACLE AND COLLISION AVOIDANCE

Employing distributed strategies to update the gains $G$ and $GP$ result in an efficient means of fusing information about present position, mission, targets and operational obstacles/threats. The strategy is one of updating the subject gains to assure motion toward a target location which avoids obstacles and collisions. Thus the gain magnitudes must be constrained in advancing motion in undesirable directions while remaining unconstrained in directions desirable for paths toward target locations. A linguistic fuzzy logic rule base provides the framework for efficiently achieving desired navigational characteristics. As an illustrative case, consider the scenario of collision avoidance of two autonomous platforms each steaming to a target destination. This situation may be characterized by the rule set below where $\Delta x$ and $\Delta y$ represent coordinate distance estimates to the autonomous platform to be avoided:
- If $\Delta x$ is large then make no correction.
- If $\Delta y$ is large then make no correction.
- If $\Delta x$ is small and $\Delta y$ is small then take corrective action.
- If $\Delta x$ is small and $\Delta y$ is smallest then take corrective action emphasizing $y$.
- If $\Delta x$ is smallest and $\Delta y$ is small then take corrective action emphasizing $x$.

The above rule base must be quantified and incorporated into the vector integration to endpoint navigation model as a gain adjustment mechanism to produce collision avoidance. The magnitudes of $\Delta x$ and $\Delta y$ along with a saturating membership function provide a mechanism to quantify the threat of collision. The process of quantifying an individual rule from the rule base is illustrated in Figure 3. The higher the membership function the higher the collision threat. The next step in implementing autonomous collision avoidance is the integration, quantification and fusion of the entire rule base. This may be accomplished by employing a weighted outcome combiner that produces a set of vector integration to endpoint gains yielding an avoidance response to a fusion of the entire collision environment. The weighted outcome rule base fusion is illustrated in Figure 4.
MIMO State Model Based Adaptation in Real-Time

The navigation path produced by the nonlinear equations (3), (4) and (5) consists of a continuous evolution of planned intended motion (PIM) points. In order to incorporate changes in: mission (i.e. targeted destination); moving obstacles; casualty mitigation; GPS updates and other variables in the deployment environment some means adjusting the autonomous navigation generator is required. A Multiple Input Multiple Output (MIMO) nonlinear state model was developed for the vector integration to endpoint navigational model. The resulting state vector provides a means for the autonomous navigation generator to be iteratively halted and updated with the latest sensor data and other pertinent information. Thus, real-time sequential processing results with sensor, GPS and mission data interleaved with the computation of the neuro-fuzzy PIM. The MIMO model also allows navigational computation to be multiplexed in real-time and provides the key to adaptation under changing mission and operating environment requirements.

Nearest Neighbors Strategy

As seen in Figure 2, the informational communications volume is a concern in the case of a truly interconnected distributed autonomous platform environment. In order to avoid collisions amongst the distributed autonomous platforms, each platform need only concern itself with the whereabouts of its nearest neighbors. This strategy is naturally utilized in biological systems and reduces communication volume, protocol complexity and computational complexity. Thus, autonomous platform communications are assumed to consist of: (1) the supervisory control function which provides mission target and obstacle information; and (2) a subset of nearest neighbor autonomous platforms for coordination, survivability and collision avoidance.

PERFORMANCE

The performance of the neuro-fuzzy augmented vector integration to endpoint navigational model may be considered in the framework of natural response characteristics. These encompass two modes of inherent navigational capability, viz., real-time targeting and real-time avoidance.

In the real-time targeting mode, the neuro-fuzzy augmented vector integration to endpoint model is inherently capable of navigation control to generate: (1) PIM to a set of desired coordinates; (2) PIM to intercept a moving platform target; (3) PIM to accomplish a standoff intercept of a moving platform target. The standoff intercept performance is attained by specifying the target platform as both a target and an obstacle.

In the real-time avoidance mode, the neuro-fuzzy augmented vector integration to endpoint navigation model implicitly includes the capability to: (1) characterize obstacles by centroid location and effective radius parameters which allows micro-perimeter outlining of obstacles or macro-centroid/radius avoidance; (2) stationary obstacle avoidance; (3) moving obstacle avoidance.

SIMULATION

The various modes of performance of the neuro-fuzzy augmented vector integration to endpoint navigation PIM path generation are depicted in Figures 5 through 9. Figure 5 illustrates a path to a designated target with no obstacles present. Figure 6 shows obstacle avoidance in proceeding to a targeted location. Figure 7 gives a representative path response to a moving platform obstacle and a fixed obstacle both in the direct path to a targeted location. Figure 8 shows the intercept capability for a moving target while Figure 9 shows a moving intercept with the capability to maintain a standoff distance.
CONCLUSION

The research effort focuses on the exploration of sensing and control techniques in the framework of fusion and distributed control pertinent to the problem of deploying autonomous unmanned surface platforms and clusters of platforms in mission applications in support of future naval capability. The present effort deals with the development of navigational modeling approaches which may be augmented with data fusion mechanisms to achieve appropriate distributed control action. Specifically, a neural vector integration to endpoint navigation model was augmented with a fuzzy fusion mechanism to produce a viable distributed autonomous navigational agent. From an application perspective, the neuro-fuzzy distributed navigational agent addresses real-time...
turnaround, information rate and adaptability concerns in autonomous platform networks. Potential mission applications may include the implementation of these techniques in unmanned vehicle control, mine warfare, anti-submarine warfare, and anti-terrorism/force protection.

REFERENCES

ACKNOWLEDGMENTS
bThe support of the work by the Office of Naval Research under Grant Number N00014-04-1-0620 and is most gratefully acknowledged.

Stephen Konyk Jr. received a Ph.D. in Electrical and Computer Engineering and an MSE from Drexel University; an MA in Mathematics from Villanova University; a BSEE from the Pennsylvania State University. He was employed by the US Navy in the area of integration, testing and at-sea certification of surface ship and submarine combat systems. His experience also includes design and modification of various shipboard electronic systems and associated hull, mechanical and electrical systems. He has developed and presented training to ship's crew and at the graduate level to government and industry engineers in the area of control systems and advanced radar systems. Dr. Konyk is currently with the Department of Electrical and Computer Engineering at Villanova University. His research interests include control, signal processing, array antennas, sensor fusion and intelligent systems. He has served as a reviewer for the IEEE Transactions on Signal Processing and is a member of Sigma Xi and the IEEE.

John Metzer received his B.S. Degree in Mechanical Engineering from Drexel University. He is a Senior Systems Engineer in the Major Programs Branch of the Machinery Research and Engineering Directorate and is also a contributing member of the Division Unmanned Vehicles Office (UVO). Mr. Metzer has nearly twenty years experience in the research, development, test and evaluation of unmanned vehicles. Prior to his employment at NSWCCD, Mr. Metzer worked for NAVAIR where he supported the Joint Cruise Missile Project Office and the Unmanned Air Vehicle Joint Program Office in the technical management of aerial missile target and unmanned tactical reconnaissance systems programs. His responsibilities included air vehicle development and design, aircraft and weapon systems integration including the design and integration of mission payloads and core
avionics, and platform integration for tactical air launch. Mr. Metzer also has extensive test and evaluation experience including on-site flight test direction and support, and ground development tests. Since 1999 he has served in the Machinery Automation and Controls Branch of the Machinery Research and Development Directorate (formerly Annapolis) where he has been responsible for architecture analyses in support of the all-electric ship.

Ashish Swaminathan received his BE in Electronics & Communication Engineering from Visveswaraiah Technological University, India. His area of expertise being control systems and signal processing. He is currently working towards his MS degree in Electrical Engineering at Villanova University. He is a student member of IEEE and is a research assistant in Electrical & Computer Engineering Department at Villanova University. His current research interest is motion control, particularly distributed control of autonomous vehicles.