

Development of an Autonomous Tactical Reconnaissance Platform

Dianne J. DeTurris¹

Cal Poly State University, San Luis Obispo, CA 93407

Jon C. Ervin²

Apogee Research Group, Los Osos, CA 93402

and

Sema E. Alptekin³

University of California at Berkeley, Berkeley, CA 94720-1774

&

Cal Poly State University, San Luis Obispo, CA 93407

In this paper, an application of Fuzzy Logic in the control of an autonomous parafoil surveillance platform is described. A unique methodology (Combs' method) was employed to reduce the common problem of 'exponential rule expansion' encountered in Fuzzy Control Systems. The use of this method also simplified the tuning process of the fuzzy system by requiring the optimization of only the membership functions and not the rule base. Modularity and simplicity of design were further enhanced by adopting a layered fuzzy logic architecture. The fuzzy system was optimized over a wide range of simulated flight conditions, providing a simple yet robust means of controlling the vehicle. Hardware limitations in terms of memory, computational speed and cost were critical factors driving the need for a simple yet robust control algorithm. This autonomous parafoil, developed under a grant from the Office of Naval Research, was designed to follow a predetermined flight pattern from altitudes of 350 meters or more down to the ground. Flight tests of a rocket-launched prototype device and simulation in the Matlab/Simulink software environment have verified that this fuzzy logic control method performs to the required specifications.

I. Introduction

THE reconnaissance vehicle that is the subject of this research study was given the somewhat unwieldy name Autonomous Tactical Reconnaissance Platform or simply ATRP. The main hardware elements of this device consist of a 3.5 square meter parafoil, an electronics payload with a protective aerodynamic housing and a launch system. The ATRP is designed to fill the need for an airborne reconnaissance device that is highly portable, low cost, easy to use, rugged and can be rapidly deployed. It is envisioned that this device would be used as a rapid response reconnaissance vehicle to perform short-range missions including locating enemy mortar positions and scouting in urban and/or rugged terrain. Training requirements for the ATRP are minimal since it is completely autonomous in flight, however a manual override exists for added user flexibility. Design requirements called for a re-usable flight vehicle and electronics that weighed less than 5 lbs. and cost less than \$1,000 in parts.

Conceptually, the ATRP can be launched in a number of ways including; small rocket launch, artillery shell or even tethered flight. Both rocket launch and compressed air guns have been used to launch the device in the current development effort. The vertical launch capability is a distinct advantage in environments where trees or buildings make landing and takeoff impossible for most micro-flyer concepts.

¹ Associate Professor, Aerospace Engineering Department, Bldg. 13, Room 257, Senior Member of AIAA.

² Senior Researcher.

³ Visiting Scholar, Berkeley Initiative in Soft Computing, Electrical Engineering and Computer Science Department, University of California at Berkeley, and Professor, Industrial and Manufacturing Engineering Department, Cal Poly State University, San Luis Obispo, Bldg. 26, Room 100.

Providing a close range, real-time view of the immediate surroundings can also be valuable in a number of civilian applications in the fields of land management and farming. Several researchers have found that aerial data can be of great value to farmers in evaluating the health and maturity of crops in the field [1,2]. Current collection methods such as satellites and manned aircraft are expensive, geographically imprecise, and frequently unavailable at critical times in the growing season. The ATRP, equipped with a Global Positioning System and the appropriate sensors overcomes these obstacles and would allow farmers to perform a more personalized crop evaluation. The ATRP could also aid in search and rescue missions such as in forest fires or avalanche.

In military applications, ATRP is designed to provide personnel in the field with a means of obtaining surveillance data of the near environment in a safe, timely and cost effective manner. The ATRP is launched to low altitude (typically <350 meters) and flies a pre-selected course while transmitting sensor data to the user on the ground. During flight, the device is fully autonomous allowing the user to concentrate on the sensor data (i.e. IR and/or visual imagery) being communicated to a hand held receiving device. The user can select one of several pre-programmed flight patterns and launch the device, all in a matter of seconds.

A parafoil design was chosen for several key advantages over rigid structure alternatives. It is lightweight, easily compacted and deployed, and very stable in flight as shown in the video. A parafoil is also much more rugged than rigid structures and will continue to perform well with minor rips and holes in the fabric canopy. Turns are made by pulling on either a left or right hand control string, making it much simpler to control than rotary or fixed wing craft. The one disadvantage to this design is the lack of a power plant to keep the craft aloft. This limits the user to a total of 5 minutes or less of surveillance time. However, the intended mission for this device is to respond to a known threat or reconnoiter an area soon to be entered and the limited dwell time over the target is not a serious drawback in this scenario. Additional time over the target area can be achieved through recovery and re-launch of the device or launching additional units.

An autonomous, fuzzy logic flight controller was chosen as the means to pilot the parafoil after launch and deploy. The controller executes a pre-selected program of flight maneuvers as it glides in the descent phase, which may include a return to the launch point if user safety permits. A fuzzy logic control method is employed because of advantages it has in fault tolerance and graceful response to missing and/or noisy sensor input. A unique methodology (Combs' method) was employed to reduce the common problem of 'exponential rule expansion' encountered in Fuzzy Control Systems. The use of this method also simplified the tuning process of the fuzzy system by requiring the optimization of only the membership functions and not the rule base. Modularity and simplicity of design were further enhanced by adopting a layered fuzzy logic architecture.

In the next section, we provide a brief background on Fuzzy Control and Combs method. Development of the Fuzzy Logic Architecture for the ATRP application is presented in the third section, followed by a discussion of the simulation and optimization of the control algorithm. In the fourth section, the design and fabrication of a prototype electronics package and current work on an upgraded system are described. The results of flight tests and conclusions are reported in the final two sections.

II. Fuzzy Control

A fuzzy logic control algorithm provides the decision making strategy for the ATRP. Attributes of fuzzy logic that make it appealing for this project are the ability to model nonlinear functions, robustness in the face of imprecise input and ease of code generation. Fuzzy logic algorithms are intuitively easy to understand and allow the user to encapsulate the experience of experts in an efficient manner. Fuzzy logic controllers are seeing much more widespread use in a wide range of industrial and consumer devices (i.e. washing machines, automobiles and microwave ovens). Many of the references cited at the end of this article can give the reader a much more comprehensive insight into fuzzy logic fundamentals [3-13].

Developed by Lotfi Zadeh in the early 1960s, Fuzzy Logic is a paradigm for modeling imprecision often found in engineering applications [14]. Zadeh recognized that, although computers operate in a world of ones and zeros, the real world is more analog in nature [15-16]. While traditional computing approaches require a rigid classification of objects into one category or another, fuzzy logic allows for partial membership of an object into multiple categories. This generalization to crisp logic allows for a more flexible, human like decision process and can be especially effective in systems where a closed form mathematical solution does not exist.

A simple example of this technique can be shown in the determination of a comfortable water temperature in drawing a bath. Typically bath water temperature would not be characterized in a binary fashion as hot or cold, but as a range between hot and cold. Indeed, the notion of what is hot, cold and just right will vary among individuals and environmental circumstances. Although our description of an environment may be imprecise, there is still potentially useful information contained in these descriptions. This information is ignored by traditional methods but can be captured in the fuzzy logic paradigm.

Figure 1 illustrates the three basic steps in the fuzzy logic decision-making process. Crisp inputs, such as a temperature reading of 40.3 degrees C in the bath water example, are presented to the fuzzification block. Inside this block crisp inputs are transformed into their respective fuzzy representation. These fuzzy variables are evaluated by the inference engine using the knowledge base, which typically consists of a number of fuzzy membership functions and rules governing the relationship between input and output variables. The fuzzy output is then put through a defuzzification process to obtain a crisp output.

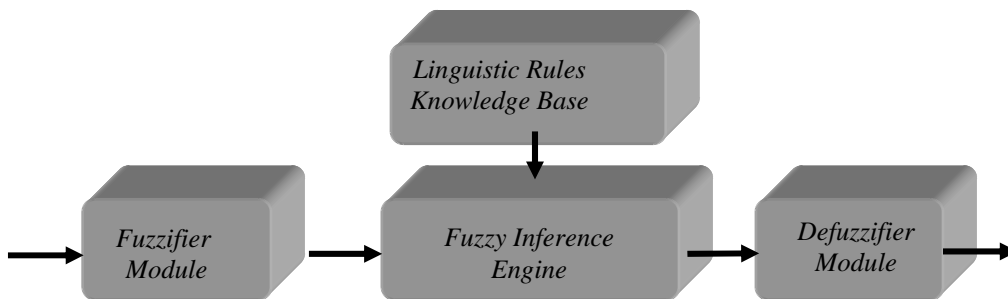


Fig. 1 The basic steps in the fuzzy logic decision-making process.

One of the disadvantages of a fuzzy logic algorithm is what is known as “exponential rule expansion”. In general, each input parameter has a number of associated fuzzy membership functions. In the typical fuzzy algorithm, the rule base is created through all possible combinations of the input fuzzy sets. In the bath water example, two inputs – “Water Temperature” and “Air Temperature”, are combined to form rules as shown in Fig. 2. If these two inputs have three input membership functions (MFs) each, the total number of rules generated are: 3 MFs for variable 1 x 3 MFs for variable 2 = 9 rules (three of which are shown in Fig. 2). This system of rules works well for a small two input problem, but as the number of inputs and/or corresponding membership functions increase linearly, the number of rules increases exponentially. This is a problem that has been encountered by many researchers as their fuzzy logic applications became more complex. Several solutions have been proposed in the recent literature [17-20]. One of the simpler and more promising solutions has been developed by W. Combs. An excellent introduction to Combs’ method is provided in Cox [21].

If bath water is “hot” and air temperature is “cold” then flow is “No Change”
If bath water is “warm” and air temperature is “cold” then flow is “hot”
If bath water is “cold” and air temperature is “cold” then flow is “hot”
 •
 •
 •

Fig. 2 Rules for Two Input Bath Water Temperature Control.

Combs calls the standard fuzzy logic formulation the Intersection Rule Configuration (IRC) because rule antecedents are formed through the intersection of the membership sets of the input variables. In the bath water example, the intersection of bath water is “hot” and air temperature is “cold” in the antecedent implies the consequent flow is “no change”. Combs’ alternative to the IRC formulation is the Union Rule Configuration (URC). Rules in the URC are formed by the union of implications that have only a single element in the antecedent. URC rules for the bath water example would be formed as shown in Fig. 3.

If bath water is “hot” then flow is “cold”
If bath water is “warm” then flow is “no change”
If air temperature is “cold” then flow is “hot”

•
•
•

Fig. 3 Rules for Two Input Bath Water Temperature Control.

Combs does emphasize that there may not be a simple transformation from a traditional rule base to a URC rule base. The construction of the Combs’ URC should be started from scratch with care taken in determining the input variables to use and in designing membership functions for those variables. He does assert that a URC, created with certain restrictions in mind, will model the same solution space to the same degree of accuracy as the traditional IRC. The restrictions on the construction of the rule base and membership functions are not particularly onerous and violating them does not necessarily invalidate a practical implementation of the method. A detailed discussion of these restrictions and a method for dealing with them if required can be found in a series of published commentary and clarification [22-26].

At first it may seem counter-intuitive that this simple change results in a dramatic reduction of rules. The effect of Combs’ formulation is that there is one rule for each membership function of each input variable. When a new variable is added, the existing rule base will likely remain the same and only rules for the new membership set need to be added. To illustrate the savings, suppose you have a fuzzy system of 5 inputs with 7 membership functions each, Combs method will have only 35 rules ($5 \times 7 = 35$) rather than the 16,807 (7^5) rules of the conventional technique.

In addition to the advantage of rule reduction, the URC also makes it possible to add inputs incrementally without creating the need for a total reformulation of the rule base. Several authors have performed research in creating a modular, multi-layered fuzzy architecture. One of the benefits of this technique is that it allows the designer to build functionality, as new modules are added, without redesigning the rule base for each of the existing modules. Typically, a higher level, governing module, needs to be completely reformulated to blend the output of the new lower level module with that of other modules at the same level. The URC can simplify this blending process since it makes it possible to add rules to the governing module without the need to reformulate the entire rule base.

The breakthrough that Combs has made has been very beneficial in this current project, where capability has increased incrementally throughout the design process. On the other hand, the theoretical restrictions that Combs has described for his method have not been a significant factor in the performance of the controller. Indeed, in previous research efforts at Cal Poly we have found that these restrictions can be relaxed and still provide good results in practical applications [27-31].

III. Development of the Fuzzy Logic Architecture

The process of developing the fuzzy logic modules was accomplished in stages throughout the life of the project. In the initial phase, a general plan was developed for a modular control system and work was focused on developing the electronic hardware and lower level software control capabilities. Subsequent phases of the project were devoted

to increasing both hardware capability and developing higher functions in the software. The modular structure of the fuzzy system was critical in allowing the project to proceed smoothly with increasing complexity in the overall design.

Due to the compressed schedule of the project, software development and hardware implementation were taking place concurrently. The first fuzzy logic software module developed was designed to provide the vehicle with the ability to follow a desired compass course. At the same time two very different compass models were purchased for testing with the hardware. One compass was very expensive and had the ability to compensate for tilt and nearby magnetic anomalies. The other compass was one-tenth the cost, but much less accurate and lacked any compensation capabilities. Through the analysis it was eventually determined that the fuzzy logic algorithm was robust enough to allow us to use the much cheaper compass with acceptable performance. The cheaper compass also opens up the possibility of using multiple compasses in an operational vehicle, thus increasing redundancy in the design and survivability of the system.

The first step in developing the course-following capability was to determine what inputs the fuzzy logic algorithm would need. Conceivably, it seemed possible that a single input, the angle difference between some predetermined desired heading and the true heading as measured by the compass, would be the only input needed to steer the ATRP to the desired heading. The desired course heading could be achieved simply by generating control commands that would drive this angle difference ($\Delta\theta$), to zero throughout the flight.

It was expected and soon confirmed that $\Delta\theta$ alone was insufficient to provide good course-following capability. Problems of overshoot and oscillation were too extreme to be considered acceptable for the ATRP application. A second input for the fuzzy logic module was produced by differentiating $\Delta\theta$ with respect to time. The addition of this angular velocity input was sufficient to provide the damping needed to bring both overshoot and oscillations within acceptable limits. Figures 4a and 4b provide a simulation comparison of the performance of the control algorithm with one and two inputs, respectively. These figures plot the ground track of a simulated parafoil flight with units given in meters along each axis. In these simulations the controller is directed to fly parallel to the X axis in the positive direction, perform a 180 degree turn and fly parallel to the X axis in the negative direction. The simulation begins with the vehicle flying a heading of 25 degrees above the X axis. Without the angular velocity input (Fig. 4a) the controller begins to oscillate out of control and cannot perform the desired 180 degree turn at all. In Fig. 4b stable flight pattern is shown with the simulated ATRP traveling 450 meters and executing a 180 degree turn to return to the launch area. A third input, angular acceleration, was also considered as a further refinement to the flight controller program. This third term has so far proven to add little value in improving accuracy, but it is maintained in the software as a hedge against possible future usefulness as the system continues to be developed.

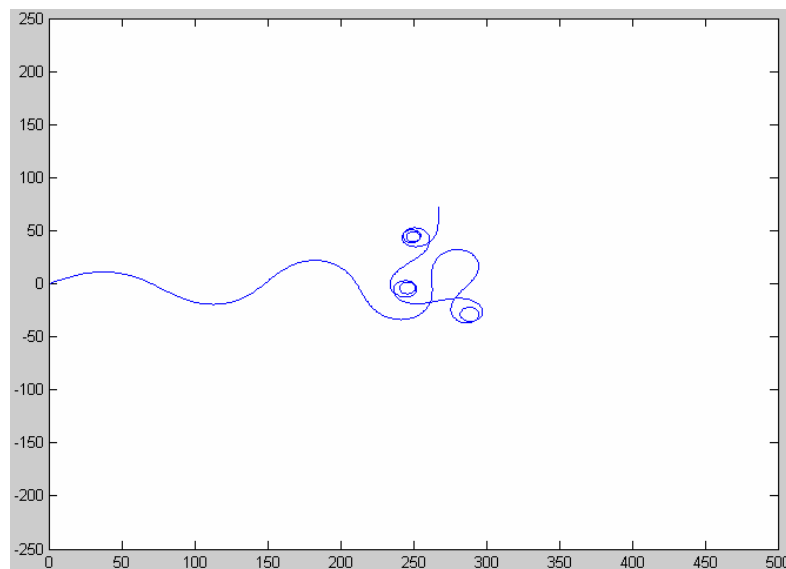


Fig. 4a Standard Fuzzy Control with one input.

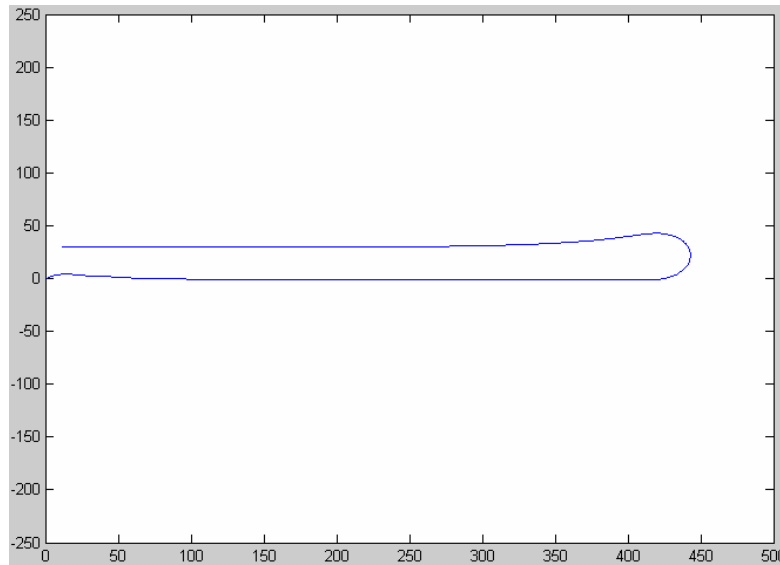


Fig. 4b Standard Fuzzy Control with two inputs.

The next step in the process was to develop membership functions for the three inputs ($\Delta\theta$, angular velocity and angular acceleration) and the one output (control line pull). The magnitude and sign of the output determines the magnitude and direction of the pull on the control lines, which in turn determined the magnitude and direction of turns of the ATRP.

Simple triangle functions were chosen for the membership sets of the three input variables and singletons were used for the output set. These choices were driven primarily by the desire to minimize memory storage and maximize computation speed. Despite the use of such simple shapes for the membership functions, the control surface for this system can still be quite complex. Singleton membership functions are a zero order form of a Takagi-Sugeno-Kang (TSK) fuzzy construct [29]. Defining the output membership functions in this manner simplifies the defuzzification process and further reduces computational requirements.

Combs method was used in the construction of the fuzzy algorithm, thus rules were automatically defined when the number of membership functions was determined. This helped to narrow the optimization task to the design of the input and the output membership functions. Here again the number of membership functions for each input was driven by a desire to keep the number low for computational reasons, while providing sufficient degrees of freedom to obtain acceptable performance. Once the shape and number of the membership functions had been established, the task that remained was to optimize their spacing and width for each fuzzy variable.

Five triangular membership functions were chosen for both $\Delta\theta$, and angular velocity, while only three were used for angular acceleration. In a standard fuzzy logic algorithm, a maximum of $5 \times 5 \times 3 = 75$ rules would have been required to fully define all possible combinations. However, following the principles of Combs method only 13 rules were needed to define the rule base. Figure 5 shows the same simulation run as was performed for Fig. 4b except that the standard fuzzy rules have been replaced with the much fewer Combs method rules. In this example and in an exhaustive series of simulation test runs, Combs method rules were shown to perform equally well as the standard fuzzy rule structure.

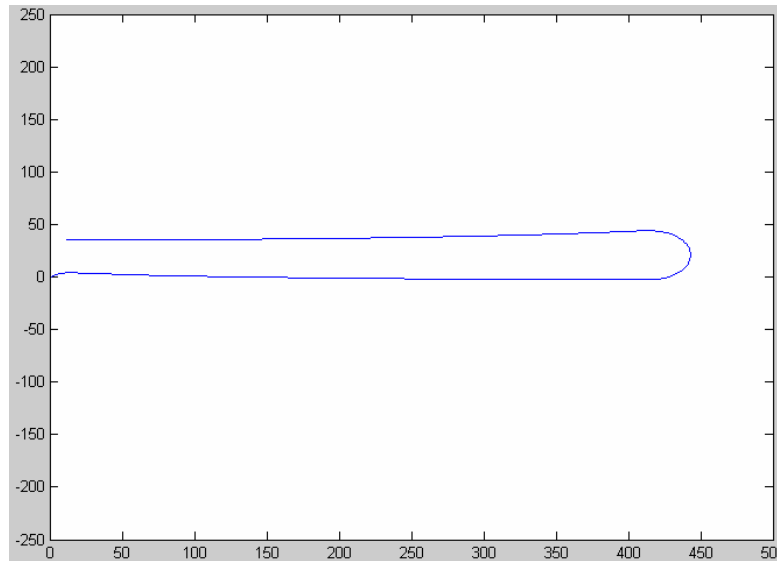


Fig. 5 Combs fuzzy control with two inputs.

The course-heading module described above cannot provide point-to-point navigation in a real world vehicle. Specifically, it will not maintain a desired ground track in the presence of cross winds and asymmetries in the vehicle aerodynamics. Figure 6 shows the effect of a cross wind on the ground track of a vehicle equipped only with the course-heading module. This displacement downwind is similar to what would happen to a boat crossing a stream if it is pointed directly across the current. Just as the boat must be pointed upstream in order to arrive at a point directly across the stream, so to must the parafoil be pointed slightly upwind to arrive at the desired destination. An additional fuzzy logic module had to be constructed as a front end for the course-heading module in order to provide proper navigation in ground based coordinates.

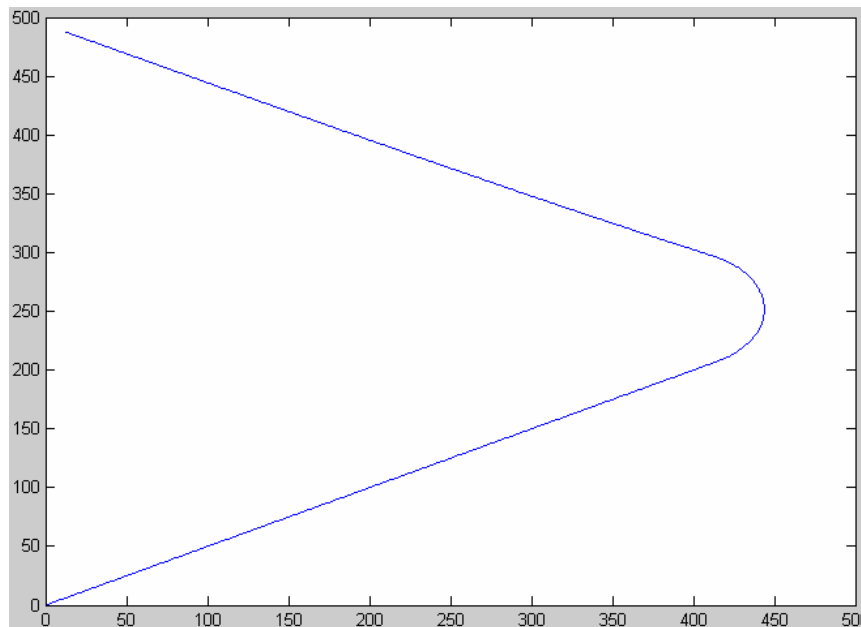


Fig. 6 Combs method with two inputs and 5m/s cross wind.

The purpose of the layered structure of these two modules is somewhat different from that described by other researchers. Raju et al. [30] use a hierarchical structure to reduce the number of rules required by the fuzzy algorithm. Outputs from the intermediate layers of this type of structure may have very little intuitive meaning to the human expert. Tunstel et al. use multiple layers to control so-called primitive behaviors with higher-level composite behaviors [31]. In contrast, here the top layer acts more like a pre-processor for the lower, course-heading layer.

The input to the course correction layer is provided by a Global Positioning Satellite (GPS) system locator unit. When four or more satellites are in view, the GPS unit provides absolute coordinates in longitude and latitude for the vehicle at periodic intervals. The controller uses the current position as provided by the GPS unit along with coordinates of the vehicle start position and destination to calculate a cross range component and angle offset from the ideal path as shown in Fig. 7. These two values along with the derivative with respect to time of the cross range component are used as inputs to the course correction layer.

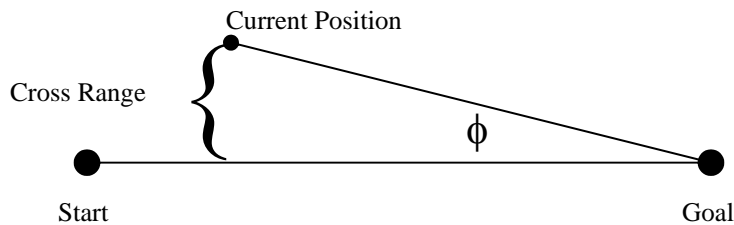


Fig. 7 Inputs for Course Correction module.

The course correction layer has seven membership functions for each of the three input values. A typical set of membership functions for the Cross Range input is shown in Fig. 8. In this figure the domain of the membership functions runs from a value of -100 meters designated “Negative Large” to $+100$ meters designated “Positive Large”. Any input that falls outside of the extreme values is truncated to the maximal value of the domain for purposes of the fuzzy logic computation. Values in between these two extreme are designated “Negative Medium”, “Negative Small”, “Zero”, “Positive Small”, and “Positive Medium”. The truth value for each membership function is read on the vertical axis as a number from 0 (false) to 1 (true) with the possibility of taking on any value in between these two extremes. A complete rule base in a standard fuzzy system would require 343 rules, however using Combs method only 21 rules are needed. The diagram in Fig. 9 shows the relationship of the two fuzzy systems and how they combine to provide an output command to the control actuators. By dividing the system into modular sub-units, logic problems can be easily traced and capability added with relative ease. The addition of the course correction layer allows the vehicle to fly a desired ground track despite the presence of a cross wind as shown in Fig. 10.

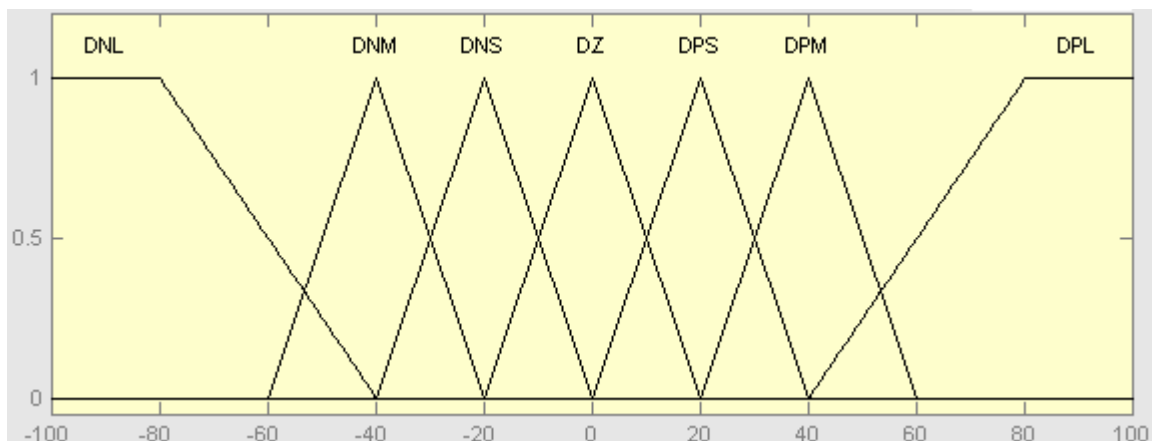


Fig. 8 Membership functions for Cross Range input.

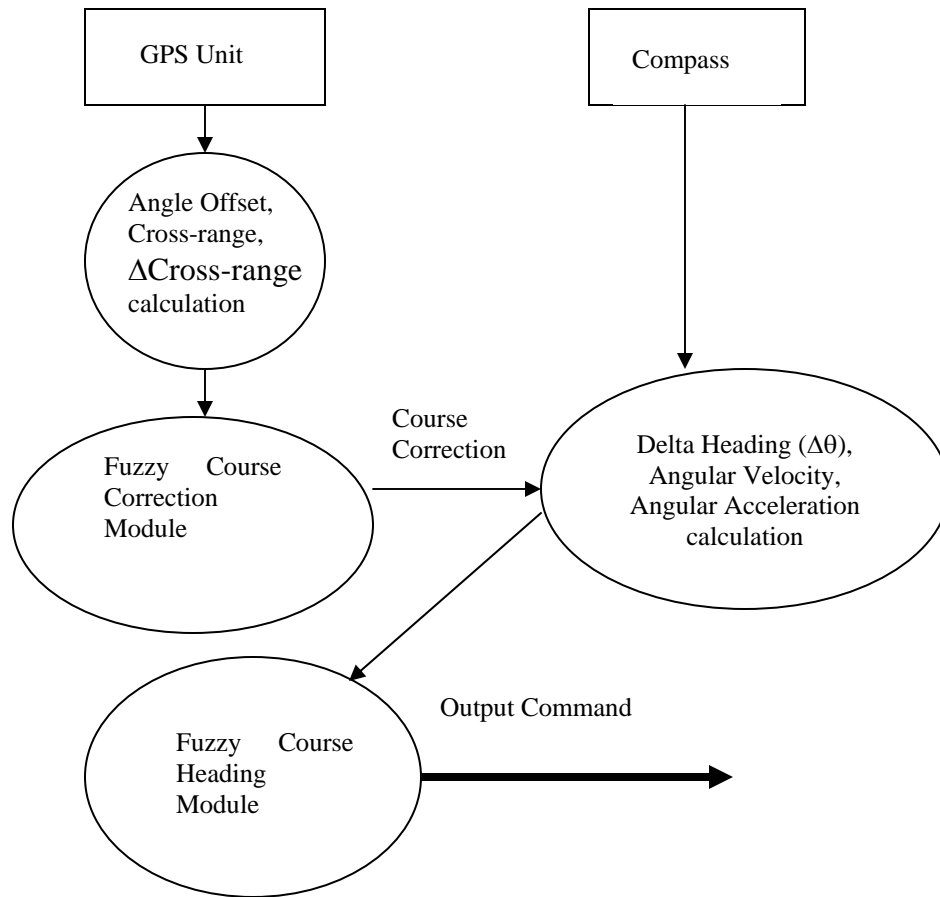


Fig. 9 Fuzzy Layered Structure.

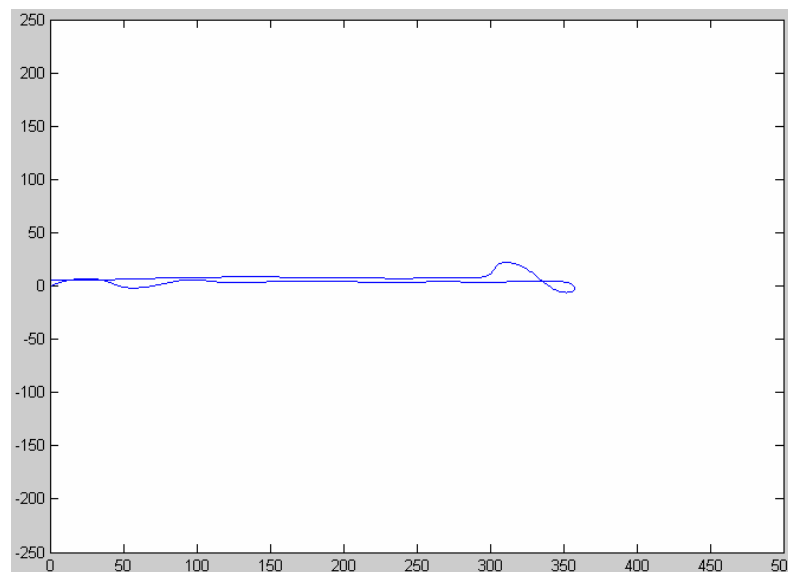


Fig. 10 Combs Method with Two Inputs and 5m/s Cross Wind and GPS Correction.

An alternative guidance scheme to the multi-layered structure employed here would have been to use input from a GPS unit alone to guide the vehicle to its waypoint destinations. Theoretically, this solution seems quite feasible and even less complicated since it eliminates the need for a compass altogether. This GPS option was not seriously pursued for reasons of overall system reliability. The ATRP is designed to be a quick reaction, work anywhere device, including conditions where the skies may be partially obscured by buildings, trees, atmospheric disturbances and signal jamming. In such cases, satellites may be obscured and the GPS device may function only intermittently at best. As it is designed, the vehicle can be guided by the compass alone and will make use of even intermittent input from the GPS unit. The compass also is a much simpler device and is expected to have a much greater mean time between failure (MTBF) rate. The compass is also capable of providing a continuous output that has proven to be essential during high turn rate maneuvers. The high update rate helps the fuzzy controller to minimize turn overshoot and oscillation effects.

The controller response has proven to be quite robust even when electronic noise, instrument inaccuracies and gusting winds are introduced into the simulation. These simulation runs were performed in the Matlab/Simulink software environment while the fuzzy logic modules were developed in the Matlab Fuzzy Logic Toolbox. It is this simulation capability that made it possible to refine the control software and thus drastically reduce the number of test flights necessary for control optimization.

IV. Simulation and Optimization

The parafoil is inherently a very simple and stable flight vehicle. Directional control is provided by pulling on one of two control lines with connections to points on either side of the parafoil wing. Simulation of the flight of the parafoil from first principles is still a daunting task even for such a simple control mechanism. The large number of unknown, and likely unknowable, factors makes an exact simulation impossible. Unknown factors include variations in wind speed and direction, sensor inaccuracies, controller inaccuracies and physical imperfections affecting the aerodynamics of the vehicle. The challenge then becomes one of how to use simulation to design an effective control algorithm with so much uncertainty and inaccuracy in the simulation model.

One factor that helps mitigate the uncertainty in the simulation is the robust nature of the fuzzy logic control mechanism. Parameters within the simulation including wind speed, wind direction, wind gusts, electronic noise, instrument inaccuracy, initial heading, parafoil speed and parafoil turning rate have all been varied over their respective, expected ranges. The fuzzy control algorithm has proven to be capable of performing well over a wide range of values for each of these factors. Thus, the expectation is that although the simulation may be lacking in absolute accuracy, it does cover the range of values that would be encountered by actual flight hardware.

Despite its limitations, the fidelity of the simulation has improved as results of flight tests have been incorporated into the simulation model. Parafoil flight speed, turn rate versus control line pull and lift over drag (L/D) ratio can all be derived from onboard instrumentation and ground based measurements. These refined parameters have then been incorporated into the simulation model, which in turn is used to improve the flight vehicle control algorithm. This cycle of continuous improvement has allowed the development of the prototype vehicle to meet an aggressive series of scheduled milestones.

One simplifying assumption made in the simulations was that the parafoil flies at a constant speed with no need for pitch axis control. This assumption is at least valid to a first degree of approximation and was not a limiting factor in optimizing the control algorithm. The objective of the simulation was not to optimize the aerodynamic qualities of the parafoil platform and therefore only a rough degree verisimilitude was necessary to achieve our goals for the simulation. The goals for the simulation have been to:

- determine the minimum number of sensor inputs to achieve satisfactory, stable control
- determine the most effective sensor inputs for achieving control
- optimize various fuzzy logic components (i.e. membership functions, rule base, etc.) under ideal conditions
- test/re-optimize the algorithm under simulated sensor and environmental sources of imprecision (i.e. sensor electronic noise, wind gust disturbance, wind drift)

Figure 11 shows a block diagram of the various software components used in the simulation and optimization process.

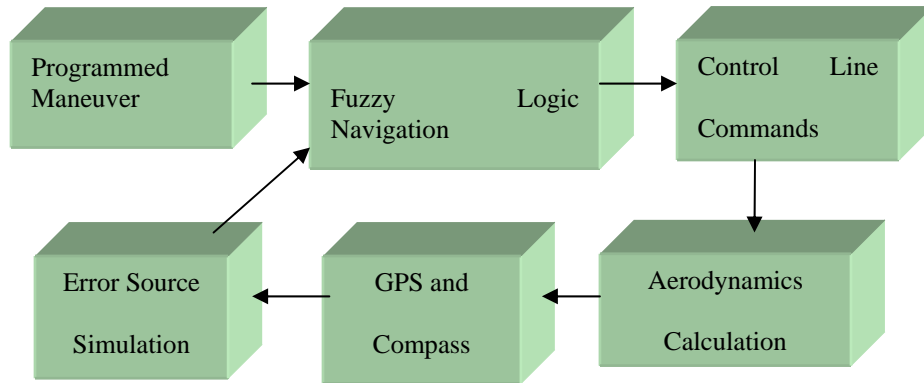


Fig. 11 Simulation Block Diagram.

Following the block diagram in Fig. 11, the simulation begins with the selection of a desired flight path. The difference between the desired and actual heading ($\Delta\theta$) along with the first and second derivatives of this difference are supplied to the course-following fuzzy logic module. Any adjustments from the course correction module are added to $\Delta\theta$ and the controller uses these inputs to produce a control line pull command. The effect of the control line pull command on the aerodynamic performance of the parafoil is computed and an incremental parafoil heading vector is determined. This heading vector is used to update the ground track map and to produce a simulated GPS measurement. Any simulated electronic errors are combined with the heading vector. The current GPS measurement along with the start and goal coordinates are used to compute the inputs for course correction module. The output from the course correction module completes the loop and the simulation runs for the length of an entire simulated mission profile.

Initial optimization runs were performed assuming ideal conditions and in later analysis error sources such as electronic noise, wind gusts, instrument bias errors, and random heading after initial deploy were added. One of the more critical error modes that required evaluation was the effect of compass bias errors on system performance. The effect of this error contributor was the determining factor in the selection between the more accurate TCM2-50 or less expensive Dinsmore compass. Comparison of simulation runs such as those shown in Figures 12a and 12b were used to determine that the Dinsmore compass would provide adequate accuracy for the ATRP project. The figures show the effect of a constant 3 degree bias in the compass reading on the vehicle ground track. A variable error that was dependent on compass heading was also tested in the simulation and proved to be less of a factor than even the constant bias condition. The effect of compass bias error was ameliorated by the GPS driven Fuzzy Course Correction module. Without this module, the bias error would have caused a constant drift in the ground track similar to that shown in Fig. 6 for a cross-wind condition.

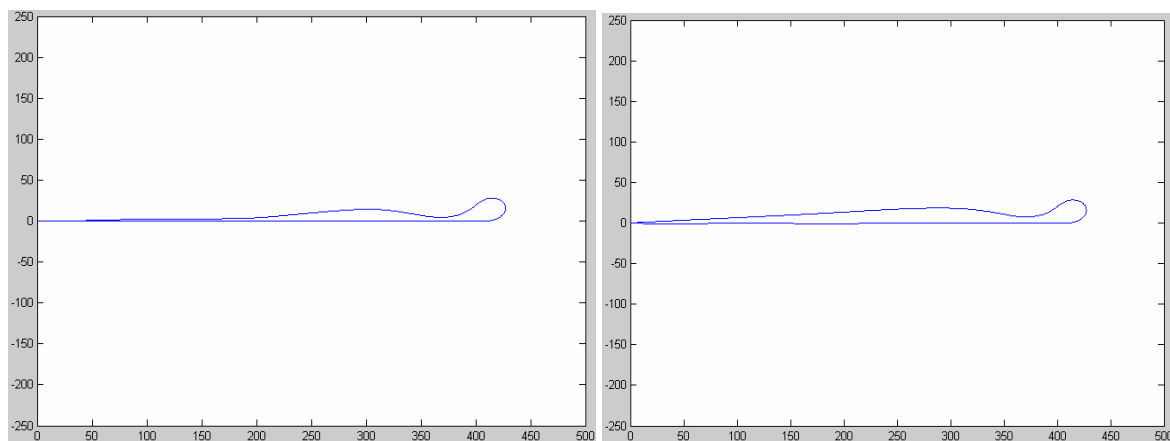


Fig. s 12a & 12b Without and With 3 Degree Compass Error, Respectively.

The shape of the vehicle trajectory in Fig. 12a suggests that additional tuning of the fuzzy logic modules may yield even better results. The simulated vehicle appears to over correct slightly when coming out of the turn, where instead a smoother transition to straight flight would be preferred. Regardless, it is apparent that a bias in the compass reading has a minimal effect on the performance of the system.

Electronic noise was modeled as a simple normal distribution of white noise. Despite the havoc this error mode played with the calculated derivatives of the instrument inputs, the fuzzy modules responded well even when no data smoothing was employed. Simulation runs performed with much higher levels of noise than were observed from bench tests of the instruments still showed excellent vehicle control.

Deployment angle is an error mode that occurs as the result of a random vehicle orientation that occurs during initial deployment from its altitude delivery mechanism. However the vehicle arrives at altitude; either by rocket, compressed air canon, artillery shell or air drop; it is likely to begin its mission at some arbitrary angle to the desired direction of travel. The guidance system must get the vehicle traveling in the desired direction as quickly as possible due to the limited amount of flight time available to the unpowered parafoil. Fortunately the fuzzy controller can be tuned to minimize overshoot and oscillations, thus extending as much as possible the ground surveillance capability of the vehicle.

Unlike cross-wind conditions, wind gusts can act asymmetrically on the vehicle to cause it to make unpredictable turn maneuvers. As an error source, the frequency of gusts is expected to lie somewhere between that of electronic noise and the near constant bias of cross-winds. Gusts were modeled as a sinusoidal function of varying amplitude over the course of a simulation run. Gust magnitudes of up to 50% of vehicle velocity and varying frequencies were simulated and these effects still allowed the vehicle to perform within acceptable limits. Fig. 13 shows a worst case scenario of all of the error sources described above.

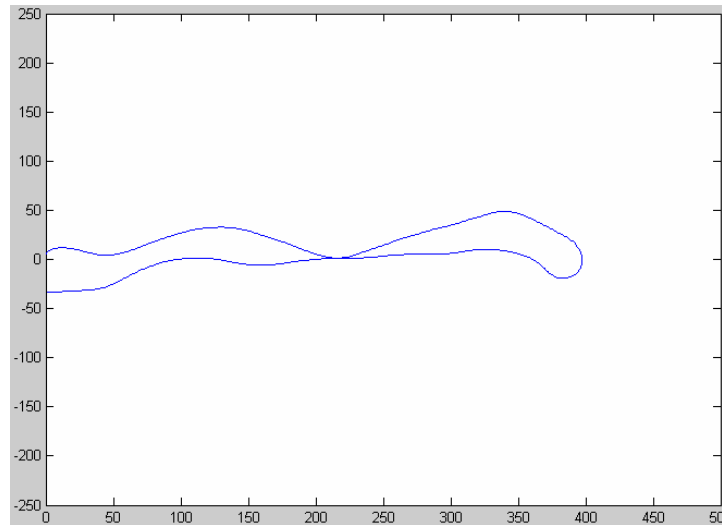


Fig. 13 Worst Case Scenario of Various Error Sources.

Compared to the ideal case shown in Fig. 12a, this trajectory more closely resembles that of a drunken sailor. The main contributing error source is the variability in the strength of wind gusts. However, despite the degradation in performance seen in the figure above, this trajectory would likely be considered “good enough” to accomplish the reconnaissance mission.

The majority of the analysis effort was devoted to evaluating various fuzzy architectures under the non-ideal conditions described above. The membership functions for both fuzzy modules were adjusted to account for these perturbations and the resulting system produced was found to be capable of handling the most extreme conditions applied. All of these adjustments were performed manually and it is expected that this tedious process will be replaced in the future with an automated procedure.

V. Electronics Hardware

The prototype electronics package that was flown on all the early test flights is shown in Fig. 14. The board at the top of this figure holds the TCM2-50 compass manufactured by Precision Navigation Inc. This compass includes a tilt sensor that allows the unit to provide $\pm 1.5^\circ$ accuracy when tilted up to 50° from the horizontal plane. The compass can also be calibrated to compensate for magnetic distortions induced by close proximity to ferrous materials and magnetic fields. In calibration mode, the compass is tilted and rotated through one or more full circles thus allowing the compass to collect magnetic field data in three dimensions. The three dimensional map of the distorting fields is computed and stored in the compass electronics.

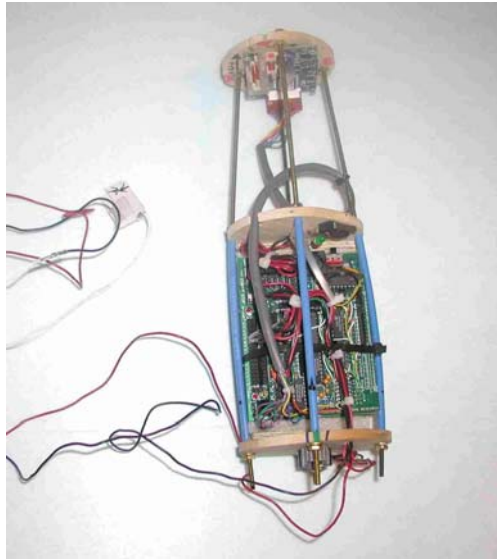


Fig. 14 Prototype Electronics Package.

The TCM2-50 performed very well in bench tests and in flight testing, however the relatively high cost of the unit prompted the search for a replacement. The Dinsmore Analog Sensor 1655 proved to be an adequate replacement for a fraction of the cost. Once a circuit had been developed to broaden the dynamic range, response curves were painstakingly mapped and the accuracy of this unit was brought to within $\pm 3^\circ$. Simulations have shown that this degree of accuracy is sufficient for the fuzzy logic controller to steer the vehicle on the desired flight path.

The electronics shown in the bottom half of the picture consists mainly of a Handy Board microcontroller developed at MIT [32]. The Handy Board has a clock speed of 2MHz and has 32 kilobytes of battery backed RAM for user programming. When combined with the optional Expansion Board as shown, it has a robust array of digital and analog data ports. In addition, the Handy Board has provision for 6 servo motor output ports and an RS232 serial port. This device provided excellent service during the bench testing and prototype flight testing.

This prototype electronics package also included an ADXL 190 accelerometer produced by Analog Devices Inc. This device, capable of measuring acceleration up to 100 g's, provided experimental verification of the g profile of the ATRP in flight. An accelerometer may play an essential role in the final design as a means to determine optimum deployment altitude and as a means of detecting landing touchdown. In the prototype flight tests, deployment was detected and communicated to the Handy Board microcontroller by means of a simple magnetic switch. The two halves of this switch were located on either side of the separation seam.

A vastly improved electronics package is currently under development. This new board has been developed in-house and is faster, smaller, less expensive and suits the specific requirements of the ATRP project. Most importantly, the new electronics suite will be much more energy efficient and require will much smaller batteries. The smaller battery requirement is expected to shave as much as 50% of the weight off of the electronics package as a whole.

The CPU of the new unit is provided by a Rabbit 3000 microprocessor embedded in a Rabbit Model RCM3400 core module. The microprocessor operates at 29.4 MHz which is a 15-fold increase over the Handy Board used in the prototype. The RCM3400 also has 16 times the memory storage than was available in the prototype device. This unit also has more than adequate numbers of input and output ports for interfacing with the variety of sensors that will be incorporated into the advanced electronics suite. Sensors incorporated into the expanded electronics package include;

- a GPS module capable of updates every .25 second
- a pressure sensor for altitude measurements with 3 meter resolution
- a separate pressure sensor for air speed measurement
- a yaw rate gyro capable of measuring slew rates up to 150 deg/sec
- a low cost magneto-inductive sensor for compass measurements
- an accelerometer capable of measuring up to 100 g acceleration

These sensors will provide inputs to an expanded version of the hierarchical fuzzy logic structure described in the preceding sections. An evaluation of each sensors contribution to overall system success will be made through simulation and flight test. Instruments that do not contribute significantly toward mission success and/or survivability at a low cost will not be included in the final vehicle design.

VI. Flight Tests

A series of flight tests were performed to test the rocket design, the deploy mechanism, and to gather information about the aerodynamic characteristics of the parafoil that could be used to optimize the flight controller. The four ft tall, 3 inch diameter rocket was made out of phenolic tubing and used a plastic nosecone and fiberglass fins. The rocket was launched on commercially available solid rocket motors of varying sizes depending on vehicle weight, to allow the rocket to reach apogee at an acceptable test altitude between 500 and 1000 ft. A standard motor burnout delayed ejection charge was used in conjunction with a piston to separate the nose from the rocket body and release the parafoil.

The prototype vehicle shown in Fig. 15 below represents the culmination of a development program that experienced a gradual evolution in design. The parafoil control line separator bar alone, the short aluminum bar running perpendicular to the rocket body, has undergone significant reduction and modification since the construction of the first flight vehicle. In the initial test flights, the parafoil was controlled from the ground by means of a radio control mechanism designed and built for this project. A series of flights under autonomous control were then performed at Camp Roberts Army Training base in California. Early autonomous flights were generally marred by flaws in the parafoil deployment mechanism. However, in the second autonomous flight, despite a late deploy event and tangled support lines, the fuzzy logic controller was able to follow the desired path albeit with significant oscillations.



Fig. 25 Improved Parafoil Control Design (note the small control line separator mounted sideways through the rocket).

In the design configuration shown, the separation joint is located 10 centimeters below the parafoil control line separator bar. When separation occurs at apogee the two halves split apart and the aft section releases a simple parachute to drift back to the ground. The forward section contains the parafoil and electronic guidance payload. The parafoil is released from the aft part of the forward section and flies the preprogrammed course with the nose of the rocket pointing down toward the ground. In the next development phase, the vehicle design requirement will be that all parts return as a single unit under autonomous control.

A new launch mechanism, using a compressed-air canon pictured in Fig. 16, has since been developed to allow for cheaper and more frequent flight tests. The electronics package and projectile have been redesigned to withstand the much greater g forces imparted by this device. Initial tests on prototype projectiles, such as the one shown in Fig. 17, have shown that this device is capable of delivering a 0.5 lb payload to altitudes well in excess of 500 feet. Flight testing of functional hardware is scheduled to recommence using this device in the next phase of the project.



Fig. 36 & 17 Compressed Air Launch Mechanism and Prototype Test Projectile.

VII. Conclusions and Future Work

This project has shown that a fuzzy logic algorithm can provide the means for autonomous control of a parafoil in flight. The fuzzy logic control software was relatively simple to construct and proved to be both robust and forgiving of noisy inputs in simulations and test flights. These attributes made it ideal in an application where a major goal was to keep total product costs and system complexity to a minimum.

Combs method of formulating fuzzy rules proved to be an effective means of minimizing the number of rules, resulting in tremendous savings in memory and execution speed. Although we cannot generalize our results to all fuzzy logic applications, the method seems to hold promise for a number areas where the “curse of rule expansion” has been shown to be a limiting factor. The layered organization of the fuzzy modules made for an easily understood and modifiable architecture. Future development may include adding additional modules to further enhance capability.

The Matlab/Simulink software suite proved to be a powerful tool for developing and simulating the flight system. The Matlab Fuzzy Logic Toolbox was also very helpful in the development of fuzzy membership functions and production rules. The optimization of the membership functions was performed through a tedious, trial and error exercise which will be automated in future work.

Preliminary flight tests served to produce data to improve the fidelity of the simulation, however autonomous tests were often marred by flaws in the deployment mechanism. The flight testing also proved to be relatively expensive and was cumbersome to coordinate and execute. A compressed-air launch mechanism has been designed and tested to alleviate all of these difficulties.

The ATRP has the potential to be superior in terms of cost, ruggedness/durability, ease of use, portability, time to activate and reusability when compared with competing technologies in a variety of mission scenarios. Future improvements will include incorporating the new technologies necessary to field a system ready for operational activity. The size and weight of the prototype system electronics and motor control package can be reduced by using more basic electronic elements in the design, a lighter and more efficient motor and battery and lighter instrument packaging. Other enhancements that will be incorporated are collision avoidance sensors, and a remote sensing device (e.g. IR camera) with telemetry transmitted to a ground station. The remote sensing device will communicate to a handheld wireless device with the user in real-time.

Other potential applications include airdrop guidance, battle damage assessment, and communications enhancement in rugged terrain. Civilian applications include; an aid in search and rescue efforts, evaluating plant health by farmers and land management workers, and as a communications and observation device for forestry fire fighting crews. The ATRP appears to have the potential for improving efficiency in both military and civilian applications with a very favorable cost/benefit ratio.

Acknowledgments

This project is sponsored by the Department of the Navy, Office of Naval Research (N00014-02-1-0754). The authors gratefully acknowledge the Cal Poly Research Dean's office for their support of this effort. We are also grateful for the enthusiasm and effort displayed by the interdisciplinary team of Cal Poly undergraduate and graduate student researchers that have contributed to this project. Special thanks go to Dan Macy, Trevor Foster, André von Muhlen, Tim Reed, and Chris LaFlash.

References

- ¹GopalaPillai, S., Tian, L., Beal, J., "Aerial imaging to detect field nitrogen deficiency in corn", ASAE paper no. 983030, ASAE St Joseph, MI, 1998.
- ²Johannsen, C.J., Baumgardner, M.F., Willis, P.R., and Carter, P.G., "Advances in Remote Sensing Technologies and their Potential Impact on Agriculture." *1st International Conference on Geospatial Information for Agriculture and Forestry*, Paper 2-1, Vol. I: 6-11, Lake Buena Vista, FL, 1998.
- ³Yager, R.R., and Zadeh, L.A., (Eds.), *An Introduction to Fuzzy Logic Applications in Intelligent Systems*, Kluwer Academic Publishers, Boston, 1992.
- ⁴Zadeh, L.A. and Kacprzyk, J., (Eds.), *Fuzzy Logic for the Management of Uncertainty*, John Wiley & Sons, New York, 1992.
- ⁵Zadeh L.A., "Fuzzy Logic, Neural Networks and Soft Computing", *Communications of the ACM*, 37(3), 1994, pp. 77-84
- ⁶Zadeh L.A., "Soft computing and Fuzzy Logic", *IEEE Software*, 11(1-6), 1994, p. 48-56.
- ⁷Zadeh L.A., "Fuzzy Logic and the Calculi of Fuzzy Rules and Fuzzy Graphs", *International Journal of Multiple-Valued Logic*, 1, 1996, p. 1-39.
- ⁸Zadeh L.A., "From Computing with Numbers to Computing with Words -- From Manipulation of Measurements to Manipulation of Perceptions", *IEEE Transactions on Circuits and Systems*, 45, 1999, pp. 105-119.
- ⁹Sinha, N.K., Gupta, M.M., and Zadeh, L.A., (Eds.), *Outline of Computational Theory of Perceptions Based on Computing with Words, Soft Computing & Intelligent Systems*, 3-22, New York; Academic Press, 2000.
- ¹⁰Jamshidi, M., Titli, A., Zadeh, L. A., and Boverie, S., (Eds.), *Applications of Fuzzy Logic*, Prentice Hall, 1997.
- ¹¹Mendel, J. M., *Uncertain Rule-Based Fuzzy Logic Systems Introduction and New Directions*, Prentice-Hall, Upper Saddle River, NJ, 2001.
- ¹²Passino, K., and Yurkovich, S., *Fuzzy Control*, Addison Wesley Longman, Menlo Park, CA, 1998.
- ¹³Yen, J., Langari, R., and Zadeh, L. A., (Eds.), *Industrial Applications of Fuzzy Logic and Intelligent Systems*, IEEE, 1995.
- ¹⁴Zadeh, L. A., "Fuzzy Sets", *Information and Control*, Vol.8, 1965, p. 338-353.
- ¹⁵Zadeh L.A., "Outline of a new approach to the analysis of complex systems and decision Processes", *IEEE Trans. Man. Cybernetics*, No.3, 1973, pp. 28-44.
- ¹⁶Zadeh L.A., "Fuzzy sets as a basis for a theory of possibility", *Fuzzy Sets and Systems*, Vol.1, 1978, pp. 3-28.
- ¹⁷ Cox, E., *The Fuzzy Systems Handbook: A Practitioner's Guide to Building, Using, and Maintaining Fuzzy Systems*, Academic, Boston, MA, 1994.
- ¹⁸Guven, M. K. and Passino, K. M., "Avoiding Exponential Parameter Growth in Fuzzy Systems", *IEEE Transactions on Fuzzy Systems*, Vol. 9, No. 1, 2001, pp: 194-199.

- ¹⁹Wang, L.-X., "Analysis and Design of Hierarchical Fuzzy Systems", *IEEE Transactions on Fuzzy Systems*, Vol. 7, No. 5, 1999, pp: 617–624.
- ²⁰Raju, G.V.S., and Zhou, J., "Adaptive Hierarchical Fuzzy Controller", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 23, No. 4, 1993, pp. 973-980.
- ²¹Linkens, D. A. and Nyongesa, H. O., "A Hierarchical Multivariable Fuzzy Controller for Learning with Genetic Algorithms", *International Journal of Control*, Vol. 63, No. 5, 1996, pp. 865–883.
- ²²Combs, W.E., and Andrews, J.E., "Combinatorial Rule Explosion Eliminated by a Fuzzy Rule Configuration", *IEEE Transactions on Fuzzy Systems*, Vol. 6, No. 1, 1998, pp. 1–11.
- ²³Weinschenk, J. J., Marks II, R. J. Combs, W. E., "Layered URC fuzzy systems: a novel link between fuzzy systems and neural networks", *International Joint Conference on Neural Networks*, Portland, Oregon, 2003, pp. 2995–3000.
- ²⁴Weinschenk, J. J., Combs, W. E., Robert J. Marks II, "Avoidance of rule explosion by mapping fuzzy systems to a disjunctive rule configuration", *International Conference on Fuzzy Systems (FUZZ-IEEE)*, St. Louis, MO, 2003.
- ²⁵Dick, S. and Abraham Kandel, "A comment on 'Combinatorial Rule Explosion Eliminated by a Fuzzy Rule Configuration' by William E. Combs and James Andrews", *IEEE Transactions on Fuzzy Systems*, Vol. 7, No. 4, 1999, pp. 475–477.
- ²⁶Mendel, J. M. and Liang, Q., "Comments on Combinatorial Rule Explosion Eliminated by a Fuzzy Rule Configuration", *IEEE Transactions on Fuzzy Systems*, Vol. 7, No. 3, 1999, pp: 369–373.
- ²⁷DeTurris D., Ervin, J., and Alptekin, S. E., "Development of an Autonomous Tactical Reconnaissance Platform", 17th AIAA Aerodynamic Decelerator Systems Technology Conference, Monterey, California, 2003, AIAA-2003-2117.
- ²⁸Gasmi, C. and Alptekin, S. E., "Supplier Selection Using Fuzzy Logic", *WSEAS Transactions*, 2003, pp. 210–215.
- ²⁹Sugeno, M., *Industrial Applications of Fuzzy Control*, Elsevier Science Inc., New York, 1985.
- ³⁰Raju, G.V.S., et al., "Hierarchical Fuzzy Control", *International Journal of Control*, Vol. 54, No. 5, 1991, pp. 1201–1216.
- ³¹Tunstel E., Lippincott, T. and Jamshidi, M., "Behaviour Hierarchy for Autonomous Mobile Robots: Fuzzy-behaviour Modulation and Evolution", *International Journal of Intelligent Automation and Soft Computing*, Vol. 3, No. 1, Special Issue on Autonomous Control Engineering, 1997, pp. 37–50.
- ³²Martin, F. G., "The Handy Board Technical Reference", Gleason Research, 2000.