

# Software Architecture for Validation and Verification of Cooperative Autonomy of Multiple Thermaling Gliders\*

Dmitrij Koniajev<sup>1</sup>, Vladimir Fedorov<sup>2</sup>  
Nahum Camacho<sup>3</sup>, Vladimir Dobrokhodov<sup>4</sup>, and Kevin Jones<sup>5</sup>

**Abstract**—This paper describes the evolutionary steps in the design of cooperative control capability of multiple autonomous soaring gliders. The paper presents the envisioned concept of operation and reviews the principal components necessary to enable the "eternal" flight of a fleet of gliders. The paper then focuses on the design of the software architecture that allows to verify the designed control algorithms in a high-fidelity simulation prior to the actual flight. The software architecture is based on the tight integration of Condor soaring flight simulator with the advanced capabilities of MatLab/Simulink design environment. The key benefits of the software for the flight control system design include the ability to realistically represent the atmospheric convective airflow and its interaction with 3D terrain, the high-fidelity flight dynamics of a variety of soaring gliders, and the advance MathWorks' tools of the control development and flight data analysis. The cooperative capability is implemented by communicating the states of multiple gliders over the network. Ultimately, the developed system allows verification and validation of advanced cooperative control strategies and their comparison against best practices of human piloted soaring flight.

## I. INTRODUCTION

Imagine a large team of gracefully soaring autonomous gliders, instrumented with sensors capable of detecting convective air currents in the environment. The gliders are launched from a remote location and assigned to provide, for example, wide area network coverage or to serve as pseudo Low-Earth-Orbit satellites to aid in fighting forest fires or to support border protection. The gliders reach the area of operation and remain there unattended for an extended period of time, perhaps up to a year. When a need for maintenance arises, the distributed intelligent algorithm reconfigures the team of gliders and calls back the aircraft in need of service. In turn, when a substitute or serviced aircraft returns, the same algorithm reconfigures the team to accept the new player. Members of the flock can either operate in a distributed fashion or fly together in a suitable formation to provide a more focused capability. The latter may include cooperative distributed sensing to achieve a desired sensor

resolution, tracking of weather formations, border patrol, and many other tasks that are currently provided by much larger, heavier, and more expensive systems.

There have been several projects that have sought to capitalize on convective lift and solar photovoltaic energy in the environment to offset or remove the need for propulsion. First demonstrated by human pilots in 1900s [1], the idea of soaring in convective air became feasible for onboard autonomous implementation only in the 1990s, see [2]. A challenge for these vehicles revolves around locating the regions of advantageous lift. While enabling the desired functionality by primarily mimicking the birds flight and indeed achieving significant extended flight capabilities, see [3], [4], [5], most of the algorithms used heuristics in the identification of the updraft strength, its potential utility in energy gain, and the decision of when and how to enter the updraft. The reason for employing heuristic approaches is obvious, since the strength of the updraft and its efficiency are both subject to significant uncertainties and are hard to formalize. The most recent development by [6], [7] demonstrated that teaming aircraft working cooperatively could improve the probability of successful detection and exploitation of thermals by splitting up the search task and sharing location data for regions of lift. Combining the ability to exploit natural lift in the environment with photo-voltaic or wind-driven energy production, the vehicles should be able to stay aloft 24/7, while still having enough additional energy to support the weight and power of meaningful payloads.

## II. INTEGRATED CONTROL DEVELOPMENT ENVIRONMENT

### A. Key Enablers of Cooperative Autonomy

Development of the collaborative capability is based on an assumption that single glider has an onboard autopilot (AP). In the soaring glider case, the AP provides the typical for UAVs stability, guidance and navigation capabilities, however with the energy management algorithms reflecting the absence of traditional propulsion. The need for multiple UAVs arises because an autonomous glider first needs to find a thermal (the area of convective air), and therefore acts as a sensor of thermals that cannot be remotely detected by the onboard sensors. A thermal can only be sensed when the glider approaches it, therefore the effectiveness of the soaring flight directly depends on the rate of detection of thermals. The detection rate can be increased [7] by utilizing multiple gliders that fly in a given air volume and share information about their findings.

\*The project has been supported over the last 3 years by a number of sponsors including the NPS Consortium for Robotics and Unmanned Systems Education and Research, the Army Research Lab, and "The Multidisciplinary Studies Support for USMC Expeditionary Energy Office."

<sup>1</sup>Senior developer in the Demand-Side Platform team, Adform Lithuania, J. Jasinskio 16C, LT-01112 Vilnius, Lithuania, dimchansky@gmail.com

<sup>2</sup>Postdoctoral researcher at the Department of Wind Energy, Technical University of Denmark, Building 114, Frederiksborgvej 399, DK-4000 Roskilde, Denmark vlfe@dtu.dk

<sup>3-5</sup>authors are with the Department of Mechanical and Aerospace Engineering, Naval Postgraduate School, Monterey, CA 93943, USA ncamacho, vldobr, jones@nps.edu

Therefore, in order to enable the collaborative flight of autonomous thermal soaring gliders, it is envisioned that each glider is equipped with online algorithms of convective thermals search, detection, and exploitation, as well as the collaborative decision making and communication methods; see the concept of collaborative flight in Figure.1. The

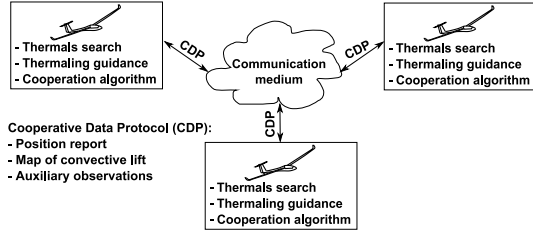


Fig. 1. Cooperation of autonomous gliders.

algorithms will run online to identify the inherent flight dynamics of the glider, which are used to detect the thermal updrafts. When flying in the updraft, the thermaling guidance algorithm is engaged to enable the maximum energy harvesting efficiency of the updrafts free energy. On the other hand it estimates the updraft geometry and motion that are used to georeference the updraft, and share its utility properties (strength) across the network of collaborative gliders. Finally, the distributed knowledge about the existing updrafts needs to be intelligently utilized to maximize the benefit of the team of gliders flying with a specific operational objective.

#### B. Architecture of High-fidelity Control Design Environment

It is clear from the preceding discussion that the desired set of enabling algorithms is significant and includes formalized knowledge from various disciplines including the flight dynamics, flight control, aerodynamics, meteorology, orography, etc. First to be developed is the high fidelity glider dynamics. Next, an advanced autopilot capable of thermals detection and autonomous soaring is to be designed and put into control of a free flying glider. A realistic weather model, including both the vertical air movements and small-scale turbulence models, has to be implemented within a considered scale of operational region. All this is to be augmented by the distributed cooperation algorithm and the communication model that both account for the distributed nature of cooperation and possibly significant losses in communication. Clearly, that even the development of high-fidelity flight dynamics of soaring glider and the realistic weather models represents a significant effort by itself. On the other hand, to verify the efficiency of the developed algorithms a number of flight tests needs to be performed. Thus, recognizing the value of flight testing and still accounting for the complexity of flight experimentation, the project builds a high-fidelity numerical simulation and control design environment that provides the convenience and rigor of control algorithm development. To this end, the project developed a realistic simulation environment that is based on tight integration of MatLab/Simulink [8] control design capabilities with the high-fidelity flight dynamics and atmospheric effects of the Condor soaring simulator, see [9].

Condor Soaring Simulator (Condor) is a Microsoft Windows application developed around a decade ago. It is a very well-known soaring flight simulator within the glider pilot community around the globe. The Condor soaring simulator is one of the most realistic simulators compared to real life soaring; the software has been designed focusing on the fundamentals of aerodynamic flight, and the atmosphere and weather physics. Recognized also for the high realism of visualization, this simulator has been used by human glider pilots to acquire initial skills and to maintain their proficiency during off-season. The key features of the Condor soaring simulator that are important enablers of the autonomous soaring are provided below, see [9] for more details:

- advanced 6DOF *flight dynamics* model run by real-time high-fidelity computational engine (up to 500 cycles per second); it includes the sailplane's *damage simulation* with an account of the flutter, high  $g$  – load stress, and aircraft collisions;
- accurate sailplane *performance and handling qualities* including the flight at critical and beyond critical angles of attack;
- defines the *weather model* with an outstanding level of details where such phenomena as convectional thermal updraft, slope updraft, ridge updraft, lee-side waves, exposure to sun that travels at side-real rate are simulated; for example, thermal updrafts reflect all phases of the real thermals including the formation, upraise and drifting with the wind, cloud forming, increase in power near the cloud base due to the condensation process in the cloud and finally the dissolving phase;
- the location and strength of thermals *accounts for the ground features* like variable terrain (ridges and valleys), forests, swamps, fields, and man-made structures like cities and villages; wavelength depends on the wind speed and stability of the atmosphere;
- implements *3D isotropic turbulence model* for thermal convection and mechanical turbulence.

The simulator provides real-time output of the glider state vector, which is extremely useful for the present project. The output can be done using a serial RS232 serial interface or via UDP protocol. The possibility of on-line flights together with other glider pilots is one of the main advantages of the Condor simulator in application to the automation of the soaring flight. In order to harness the features of high-fidelity soaring simulation and connect them with advanced capabilities of control systems design provided by the MatLab/Simulink software, an unobtrusive and invisible for the original Condor software service has been developed. The key objective for this “Condor API” service is to provide low-level interoperability between the Simulink and Condor. This interoperability<sup>1</sup> has been made possible by using reverse engineering without modifying any of the licensed software. Presented in Figure.2, the Condor API service significantly extends the aircraft data,

<sup>1</sup>EU Computer Program Directive [10] and Sec. 103(f) of the DMCA (17 U.S.C. 1201 (f)) [11] allow reverse engineering for the purposes of interoperability.

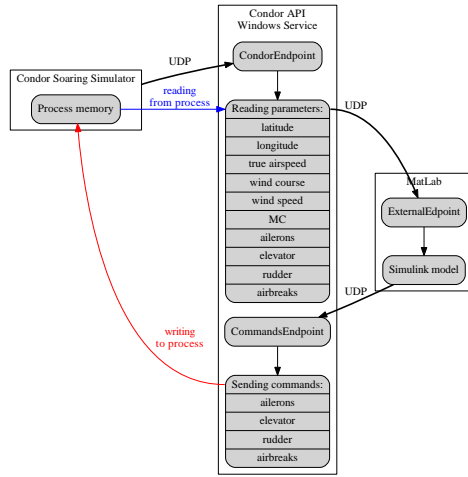


Fig. 2. The Condor API architecture.

which Condor simulator streams to external applications. The data set includes parameters like the plane position, true airspeed, wind course, wind speed, MacCready setting, and the control surfaces deflection. The Condor API then streams the aircraft data to an external application using the UDP protocol. Utilizing this API, an external application can send the control surface commands back to the Condor API service via UDP, thus effectively establishing the software in the loop (SIL) environment, see Figure.3.

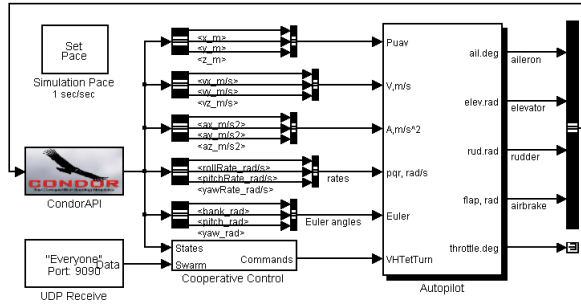


Fig. 3. SIL environment integrates the capabilities of Simulink and Condor.

Verification and validation (V&V) of cooperative control strategies that take significant time to evolve in a distributed settings can be very difficult. To facilitate early identification of correct strategies and save time a visualization of cooperative strategies might be very effective. To this end, the Condor API service has been expanded with the “FlightRadar” external application that provides the capability to represent multiple evolving trajectories in one 3D map. To visualize the strategies of multiple competing soaring gliders in real-time the FlightRadar streams the telemetry information to Google Maps [12], see an example of a half-hour flight in Figure.4.

An example of possible setup for the design and V&V of the cooperative soaring algorithms is given in Figure. 5 that represents a scenario where the autonomous gliders and the human pilots can fly together. Current weather conditions

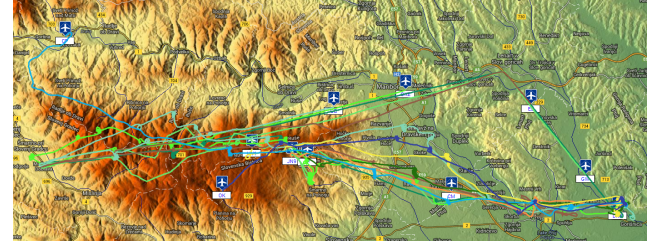


Fig. 4. Evolution of cooperative strategies in Google Maps.

and interaction between the gliders in Condor simulator is handled by the Condor server application that can be deployed either in the same local network as the Condor Soaring Simulator or a dedicated Internet server provided by the company can be utilized; a set of servers is publicly available with their IP addresses and settings available to the registered users. A modelled glider communicates with the server via the Condor protocol over the Internet/network as in a typical computer game in multi-player mode. The Condor-Simulink SIL environment runs on the same (or a separate) computer and interacts via UDP protocol with the Condor API service. The multiple instances of Condor-Simulink SIL environments share knowledge among themselves via the co-operation protocol (CDP) within the local computer network. In addition, this setup also allows gliders to be controlled by human pilots operating within the same network by using the original Condor simulator client application.

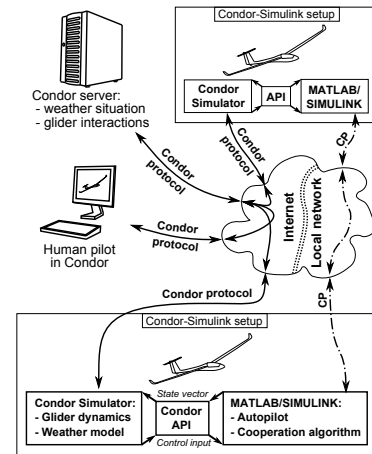


Fig. 5. Possible setup of two SIL environments and a glider operated by a human pilot; Condor simulator server is running on a dedicated PC.

Harnessing the features of Condor soaring simulator allows leveraging the advanced capabilities of the MatLab/Simulink control design environment, thus resulting in a high-fidelity control design setup for cooperative soaring flight. Thus, the developed setup provides a convenient interface to researchers that allows them to quickly connect to the simulator and focus on the development of novel cooperative control.

### C. Potential Capabilities and Future Improvements

In the current implementation the intelligent agents, which are implemented by the Simulink model, share the states of a glider and its local onboard knowledge of thermaling convective lift via multicast UDP messages. This communication mechanism allows to run the Condor soaring simulator and Matlab/Simulink on different machines connected over the local network. As a result, this capability enables a competitive flight of multiple soaring gliders inside one local network. In turn, the availability of the multi-player mode in the Condor opens the possibilities for testing of cooperation algorithm of the unmanned gliders in the realistic environment, and validation of the algorithm against the behavior of a group of human pilots in a wide range of weather conditions.

At the same time the UDP communication has also its flaws. Since, in general, the UDP multicast (it is used to share the knowledge among gliders) over the Internet is not possible because the multicast packages are not routed, the cooperative competitive flight of multiple autonomous gliders placed in different networks is not possible; it is possible only inside one local network. Another limitation is that multiple independent groups of autonomous agents located on the same local network at the same time can mislead each other.

The future version of the Condor API service could be improved by building a distributed Condor API system, which would consist of a number of Condor API services communicating with each other using gossip protocol [13]. The distribution mechanism could be implemented using TCP/IP sockets like in the distributed Erlang systems [14] or Akka actors [15]. This will solve the aforementioned problems and allow to create more powerful distributed agents.

### III. PRINCIPLES OF CONVECTIVE LIFT DETECTION AND COOPERATIVE EXPLOITATION

To illustrate the utility of the developed SIL integrated environment a brief outline of the algorithms [7] developed to enable the cooperative soaring is presented next.

The onboard algorithm of detecting a thermal is based on two complementary approaches. The first approach utilizes the inherent sink polar, and the second one is based on the total energy of the system. However, conceptually they are similar as they compare the natural metrics of the system with the same metrics actually measured in flight.

The *sink polar* is the function of vertical sink rate versus the true airspeed (TAS) - of a particular glider can be practically achieved in extensive experimentation. Sink polars are roughly quadratic in nature, and a least-squares approach yields suitable coefficients based on the historical data. In the developed approach, the estimates of the sink-polar were first made by post-processing a collection of experimental flight results obtained in low-wind, low-lift conditions. Further in flight, a recursive linear least square estimator is used in real-time to account for specific variation in the platform and atmospheric conditions at that moment. The detection

of a thermal and estimation of its intensity, that contributes to the recursive identification of its parameters, are based on comparison of the currently measured sink rate with the sink rate predicted by the polar for a measured TAS. If the measured sink rate is smaller than predicted, then there is a thermal.

The *total energy* approach is based on the concept that the mechanical energy  $E_{tot}$  of the soaring glider combines the potential energy,  $E_p = mgh$ , and kinetic energy,  $E_k = \frac{m \cdot V^2}{2}$ , of the airframe minus the "leakage" of the energy due to the work of the parasitic and induced aerodynamic drag,  $E_D$ . For an "aerodynamically clean" glider designed with an objective to minimize the total energy loss, the control commands of its autopilot will necessarily result in mild variations of the angle of attack, thus leading to the relatively constant parasitic drag and  $\dot{E}_D \approx 0$ . Consequently, for the total energy and its rate of change over sufficiently long time intervals one can consider the following:

$$E_{tot} = mgh + \frac{m \cdot V^2}{2} - E_D, \quad E = \frac{E_{tot}}{mg},$$

$$\dot{E} = \dot{h} + \frac{V \cdot \dot{V}}{g}, \quad \ddot{E} = \frac{\dot{V}^2 + V \cdot \ddot{V}}{g} + \ddot{h}, \quad (1)$$

where  $m$  is the mass of the airframe,  $g$  is the gravitational constant,  $E$  is the normalized total mechanical energy of the system (also called the specific energy),  $h$  is the height, and  $V$  is the inertial speed. Therefore, the longitudinal long period oscillations represent the natural tradeoff of kinetic and potential energy while their sum remains nearly constant. As a consequence, in no updraft conditions the rate of change of the total energy  $\dot{E} \approx 0$ . Therefore, if there is a significant variation of the total energy, then the energy rate will be significantly away from zero thus indicating the energy variation due to updraft or downdraft airflow. Both equations in (1) are included into one Kalman filter along with the inertial and barometric sensors outputs. The resulting energy rate-based solution provides another accurate indication of the updraft event. A comparison of the output of the total energy approach with the output of the total energy compensated variometer  $\dot{h}_{TEK}$  [16] is presented next in Figure.6.

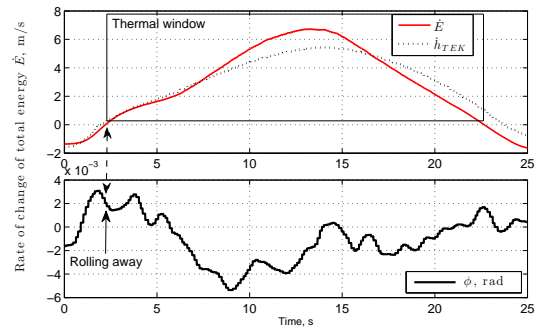


Fig. 6. Energy-based detection of updrafts simulated by the software [9].

When a thermal updraft is detected, the glider needs to automatically maneuver to enable staying in the thermal

with the objective of increasing the glider's potential energy through a rapid increase of the height. The thermal centering guidance law produces a turn rate command  $\dot{\psi}_c$  to the autopilot, and is based on the feedback control law that takes into account the desire to get closer to the updraft center (defined by the  $\rho_d$ ), where its intensity (the positive vertical speed) is the highest. On the other hand, the control balances the height increase and the turn-induced sink by a measure proportional to the rate of increase of the total energy (defined by the  $\ddot{E}$  in (1)), see the geometry of the guidance task in Figure.7 and the resulting guidance law in (2):

$$\dot{\psi}_c = \frac{V}{\rho_d} - k_1 \cdot \ddot{E}, \quad (2)$$

where  $\rho$  and  $\rho_d$  are the current distance and the desired orbital radius around the center of the thermal updraft, and  $k_1$  is the feedback gain determined by the stability and performance requirements. The physical meaning of the

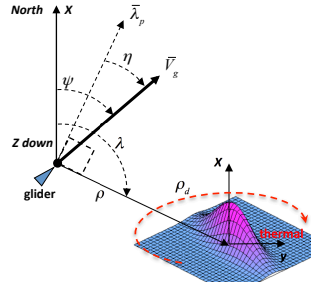


Fig. 7. Kinematics of guidance around a stationary thermal updraft; the desired orbit is represented by the red dashed line defined by  $\rho_d$ .

guidance law (2) is to increase the commanded turn rate until the rate of climb in the latched updraft is compensated by the sink resulted from the steep banking; most traditional autopilots implement bank-to-turn control laws.

The initial approach to the estimation of thermals (2D coordinates of the center vs. the glider altitude) by utilizing the measurements of single glider in soaring mode was based on two classical nonlinear filtering techniques: first - the nonlinear Kalman filter with the "bearings only measurements", and second - the kinematic relation  $\dot{\rho} = -V_g \cdot \sin(\eta)$  between the speed over ground  $V_g$  and the "navigation error"  $\eta$ , see Figure.7. In both formulations the bearing to the updraft center was assumed to be constant at  $\pi/2$  ( $\lambda - \psi = \pi/2$ ) with respect to the direction of turning flight; the turn is defined toward the center of the updraft. When entering a strong updraft, the performance of either filter was slow but reasonable, resulting in a converging solution in about 2 full orbits and precision of the thermal center estimation of 75m.

To further improve the efficiency of updraft estimation the solution should integrate the knowledge gained by multiple gliders and the prior meteorological observations, see [17], [18]), which might be available for the area of operation. The latter data can be conveniently interpreted as a map of probability density of convective air activity with respect to the geographic latitude and longitude, see a conceptual

example in Figure.8. As a first step toward cooperative iden-

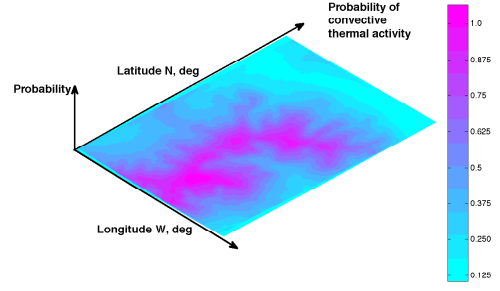


Fig. 8. The "heat map" as a probability density function.

tification and mapping of convective thermals over extended areas with an account of prior methodological observation, the probabilistic recursive Bayesian approach was adopted, see details of the approach in [19].

An example of the Bayesian approach for the case of three simulated gliders cooperatively flying and estimating parameters of a single stationary updraft in a given area modeled by a Gaussian model is presented in Figure.9; note, there is no horizontal component of airflow in the model. The task is

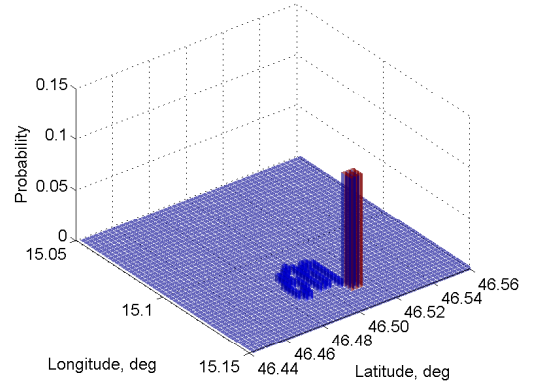


Fig. 9. Estimation of an updraft obtained onboard of glider #1 from the cooperative sampling of environment.

to find the updraft in a bounded area and to converge to the same thermal by utilizing the detection algorithms discussed above; the task mimics the setup and the objectives of the first cooperative flight test of two gliders reported earlier in [7]. In the demonstrated result the prior probability density is initialized by a uniform function over the entire area of operation. The result corresponds to the progression of the probability density function estimated onboard of glider #1 along its flight path, see the corresponding cooperative trajectories of gliders #2 and #3 in Figure.10. The gliders start their flight simultaneously at the same altitude, and initially spend some time in search for thermals. When glider #1 detects an updraft utilizing either of the thermal detection approaches and shares the information about the thermal, the other two gliders arrive to the same thermal, and successfully



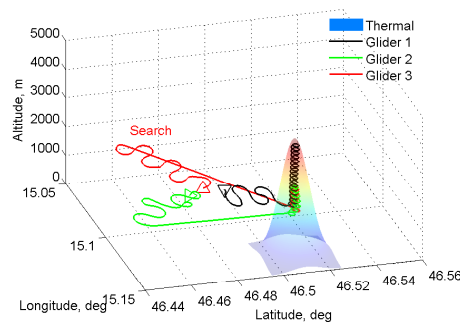


Fig. 10. Cooperative flight of three gliders.

gain height all together. The time history of the altitude gain is presented next in Figure.11. The following result illustrates

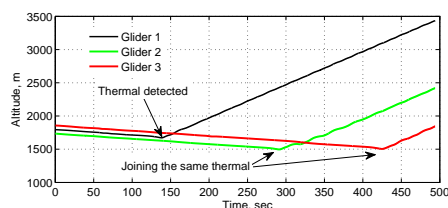


Fig. 11. Cooperative gain of altitude.

the capabilities of gliders to make independent decisions in gaining the altitude. The initial settings, the setup, and the goal remain the same. Figure.12 illustrates that independent decision making allows to utilize the thermals of opportunity by sharing their geographic location and strength found by other gliders. The decision to stay in a newly found thermal is made if its strength is comparable with the strength of nearby "known" thermal. The obvious gain of height confirms the efficiency of this strategy. The results clearly demonstrate

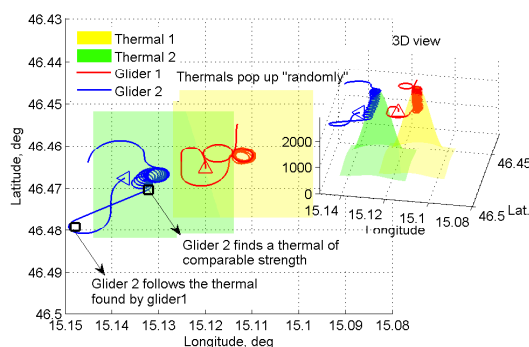


Fig. 12. Independent decision making.

the benefits of collaborative strategies in harvesting the convective updraft energy from the environment.

## CONCLUSIONS

The paper presents the development of autonomous convective thermal soaring capability of multiple cooperative gliders. The discussion focuses on the software architecture

that allows to design the intelligent cooperative soaring autonomy, and to verify the designed control algorithms in a high-fidelity simulation prior to the actual flight. The key benefits of the software include the ability to realistically represent the atmospheric convective airflow and its interaction with 3D terrain, the high-fidelity flight dynamics of a variety of soaring gliders, and the advance MathWorks' tools of the control development and flight data analysis. The key technologies developed and verified in the novel SIL environment include the online characterization of the inherent flight dynamics of soaring gliders, the convective thermals detection, and the collaborative environment sensing by utilizing recursive Bayesian estimation. Ultimately, the developed system allows V&V of advanced cooperative control strategies and their comparison against the best practices of human piloted soaring flight.

## REFERENCES

- [1] M. Simons and P. A. Schweizer, *Sailplanes by Schweizer: A History*. Crowood Press, 1998.
- [2] J. Wharington, "Autonomous control of soaring aircraft by reinforcement learning," Ph.D. dissertation, Royal Melbourne Institute of Technology (Australia), 1998.
- [3] D. J. Edwards, "Implementation details and flight test results of an autonomous soaring controller," in *AIAA Guidance, Navigation and Control Conference and Exhibit*. North Carolina State University, 2008, aIAA 2008-7244.
- [4] M. J. Allen, "Updraft model for development of autonomous soaring uninhabited air vehicles," in *44th AIAA Aerospace Sciences Meeting and Exhibit*, 2006.
- [5] M. J. Allen and V. Lin, "Guidance and control of an autonomous soaring vehicle with flight test results," in *AIAA Aerospace Sciences Meeting and Exhibit, AIAA Paper*, vol. 867, 2007.
- [6] K. Andersson, K. Jones, V. Dobrokhodov, and I. Kaminer, "Thermal highs and pitfall lows-notes on the journey to the first cooperative autonomous soaring flight," in *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*. IEEE, 2012, pp. 3392–3397.
- [7] K. Andersson, I. Kaminer, V. Dobrokhodov, and V. Cichella, "Thermal centering control for autonomous soaring: stability analysis and flight test results," *AIAA Journal of Guidance, Control, and Dynamics*, vol. 35, no. 3, pp. 963–975, 2012.
- [8] MATLAB, version 8.2.0.701 (R2013b). Natick, Massachusetts: The MathWorks Inc., 2013.
- [9] Condor, "The competition soaring simulator," October 2013. [Online]. Available: <http://www.condorsoaring.com/>
- [10] Directive 2009/24/EC of the European Parliament of 23 April 2009 on the legal protection of computer programs. [Online]. Available: <http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2009:111:0016:01:EN:HTML>
- [11] Digital Millennium Copyright Act (Title 17 of the United States Code 1201). [Online]. Available: <http://www.law.cornell.edu/uscode/text/17/1201>
- [12] Google maps api. [Online]. Available: <http://maps.google.com>
- [13] I. Gupta, A.-M. Kermarrec, and A. J. Ganesh, "Efficient and adaptive epidemic-style protocols for reliable and scalable multicast," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 17, no. 7, pp. 593–605, 2006.
- [14] Erlang/OTP. [Online]. Available: <http://www.erlang.org/>
- [15] Akka toolkit. [Online]. Available: <http://akka.io/>
- [16] PitLab, "Skyassistant variometer," October 2013. [Online]. Available: <http://www.pitlab.com>
- [17] C. Pennycuik, "Field observations of thermals and thermal streets, and the theory of cross-country soaring flight," *Journal of Avian Biology*, vol. 29, no. 1, pp. 33–43, March 1998.
- [18] E. Hindman, S. M. Saleeby, O. Liechti, and W. R. Cotton, "A meteorological system for planning and analyzing soaring flights in colorado usa," *Technical Soaring*, vol. 31, no. 3, pp. 68–82, 2007.
- [19] N. Bergman, "Recursive bayesian estimation," Ph.D. dissertation, Department of Electrical Engineering, Linköping University, Sweden, 1999.