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

Dmitrij Koniajev[1], Vladimir Fedorov[2]
Nahum Camacho[3], Vladimir Dobrokhodov[4], and Kevin Jones[5]

Abstract— **This paper describes the evolutionary steps in the design of cooperative control capability of multiple autonomous soaring gliders. The paper reviews the principal components necessary to enable the "eternal" flight of a fleet of gliders, and presents initial comparison of simulated and actual flight test results that demonstrate high-potential of collaborative autonomous soaring. The paper primarily 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 in the environment to offset or remove the need for propulsion. First demonstrated by human pilots in 1900s (see [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], and [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 both 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. Concept of Operation

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 in Figure.1. The 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, and on the other hand estimates the updraft geometry and motion, that are used to georeference

*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" program.

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

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

[3−5]authors are with the Department of Mechanical and Aerospace Engineering, Naval Postgraduate School, Monterey, CA 93943, USA ncamacho, vldobr, jones@nps.edu

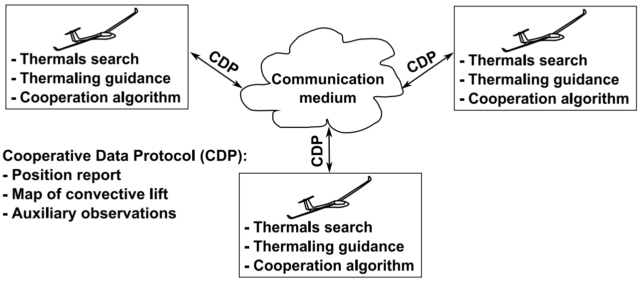

Fig. 1. Cooperation of autonomous gliders.

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 team of gliders flying with a specific operational objective.

B. Software Architecture of High-fidelity Control Design Environment

To verify the efficiency of the developed algorithms a number of flight tests needs to be performed. While 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].

The Condor soaring simulator is one of the most realistic simulators compared to real life soaring; the software has been designed focusing on the fundamental principles of aerodynamic flight, and atmosphere and weather physics. Recognized for the high fidelity of simulating of the fundamental principals of soaring flight and 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 Condor allows to advance piloting skills in different wing loading conditions and weight balancing, various flight regimes including stall and spin, and even to test pilots ability to recover the damaged aircraft. Furthermore, the pilots can experience soaring flight in different areas of the world, test their skills against other pilots in individual and team competitions. 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 physics engine (up to 500 cycles per second);
- accurate sailplane performance and handling qualities including the flight at critical and beyond critical angles of attack;
- sailplane's damage simulation including the flutter, high g stress, and aircraft collisions;
- realistic representation of the thermals life cycle that starts from the ground and expands up to the cloud base; the cloud grows bigger and more dense and later dissipates thus causing the air below it to sink;
- the location and strength of thermals is based on the exposure of terrain to sun and accounts for the ground features like forests, swamps, fields, and man-made structures like cities and villages;
- realistic daily sun travel which affects the frequency and strength of thermals;
- models up-slope wind on sunny ridges (anabatic winds);
- models ridge lift with leeward downwind and turbulence, venturi effects;
- models waves behind ridges; wavelength depends on wind speed and stability of the atmosphere;
- defines correct atmospheric pressure, density and temperature versus height;
- implements $3D$ isotropic turbulence model for thermal convection and mechanical turbulence.

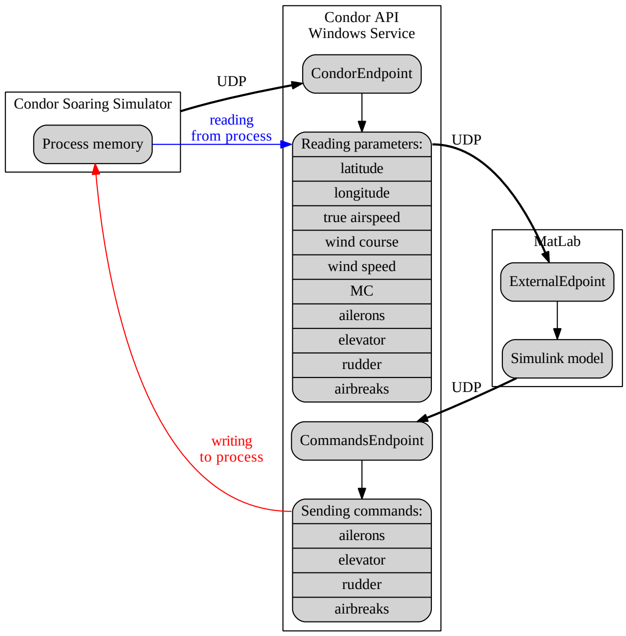

Fig. 2. The Condor API.

In order to harness the features of high-fidelity soaring simulation and connect them with advanced capabilities of control systems design provided by MatLab/Simulink, 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 communication between Simulink and Condor. Presented in Figure.2 the Condor API service significantly extends the aircraft data which Condor simulator streams to external applications. The extended data set includes data like the plane position, true airspeed, wind course, wind speed, MacCready setting (MC), control surfaces (ailerons, elevator, rudder and air-break) deflection. The Condor API then streams the extended aircraft data to an external application (MatLab/Simulink) using UDP protocol. Utilizing Condor API, an external application then 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.

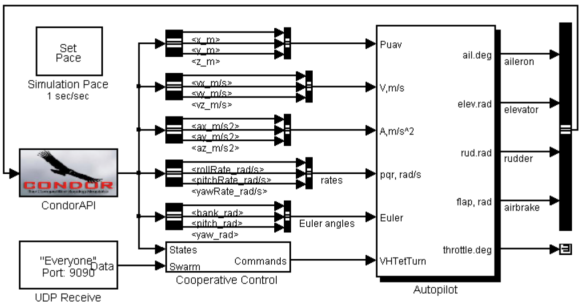

Fig. 3. Integration of Simulink and Condor capabilities.

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 for the researchers that allows them to quickly connect to the simulator and focus on the novel algorithm development.

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. Moreover, this capability enables a competitive flight of multiple soaring gliders inside one local network.

At the same time the UDP communication has also its flaws. Since the UDP multicast over the Internet is not possible in general because the multicast packages are not routed, the cooperative competitive flight of multiple autonomous gliders with human pilots over Internet is not possible. Another limitation is that multiple groups of autonomous agents located on the same local network at the same time will interfere with each other.

The future version of the Condor API service could be improved by building distributed Condor API system, which would consist of a number of Condor API services communicating with each other. The distribution mechanism could be implemented using TCP/IP sockets like in the distributed Erlang systems ([10]) or Akka actors ([11]). This will solve the aforementioned problems and allow to create more powerful distributed agents.

Verification and validation 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 "FlightRadar" external application that provides the capability to represent multiple evolving trajectories in one $3D$ map. The

To visualize the strategies of multiple competing soaring gliders in real-time the FlightRadar streams information to Google Maps [12], see an example of a half hour flight in Figure.4.

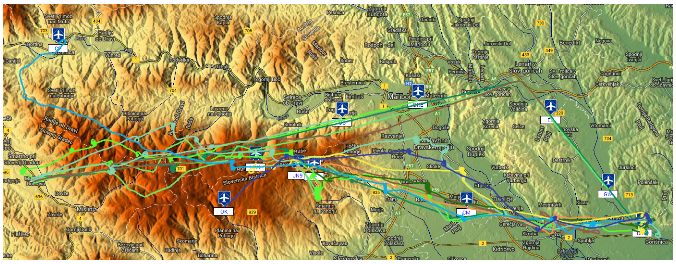

Fig. 4. Evolution of cooperative strategies.

III. Principles of Convective Lift Detection and Cooperative Exploitation

...

A. *Thermals Detection and Exploitation by Single Glider*

Sink polar
Total Energy approach
Guidance of a single glider

B. *Cooperative Strategies*

....

IV. Experimental Results

Present results of 3 gliders searching for a single thermal in a bounded box. One of them finds the thermal, estimates its position and shares knowledge with the rest of the team. The other 2 gliders come to the same spot and start climbing.

As an illustration of the achieved capabilities, Figure.5 represents the cooperative flight of three gliders in a simplified scenario introduced above. 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 gain height all together. Time history of the altitude of three cooperative gliders is presented next in Figure.6. The result clearly demonstrates the benefits and significant potential of collaborative strategies in harvesting the convective updraft energy from the environment.

Appendix

Appendixes can be used to show the snippets of configuration files.

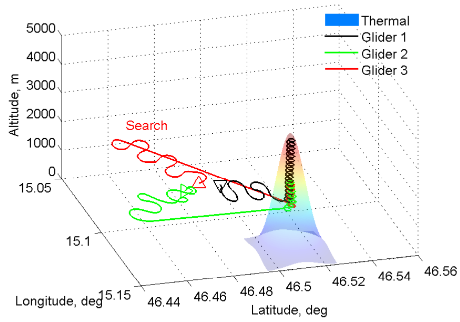

Fig. 5. Cooperative flight of three gliders; starting at different locations they all converge to the same updraft when glider #1 finds it and shares its estimated location.

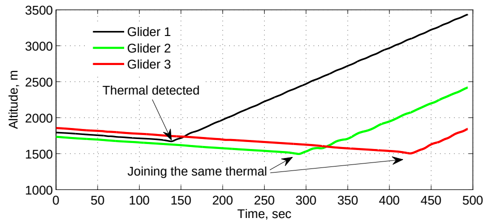

Fig. 6. An example of cooperative flight of three gliders.

ACKNOWLEDGMENT

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" program. The authors would like to mention the contribution of graduate students Andrew Streenan and Joshua Weiss for providing results of their final project and contributing to the extensive simulation research.

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] Erlang/OTP. [Online]. Available: http://www.erlang.org/

[11] Akka toolkit. [Online]. Available: http://akka.io/

[12] Google maps api. [Online]. Available: http://maps.google.com