

Hardware in the Loop Simulator for Multi Agent Unmanned Aerial Vehicles Environment

¹Kamal S. Ali and ²Justin L. Shumaker

¹Department of Computer Engineering, Jackson State University, Jackson, Mississippi, USA

²Vehicle Technology Directorate, U.S. Army Research Laboratory, Aberdeen Proving Ground, USA

Received 2013-02-24, Revised 2013-02-27; Accepted 2013-05-10

ABSTRACT

This study describes a Multi-Agent Hardware In the Loop Simulator (HILS) for Unmanned Aerial Vehicles' (UAV) autopilots. This HILS system allows multiple autopilots to fly multiple virtual airframes in the same virtual environment allowing these agents to interact as a swarm. This system makes the exploration and evaluation of Multi Agent Systems (MAS) possible at minimal cost.

Keywords: HILS, MAV, Multi Agent, Swarm

1. INTRODUCTION

Hardware In the Loop Simulators (HILS) are systems that combine the best of software flexibility and hardware accuracy, providing the user with a powerful and realistic evaluation tool.

Currently, a monolithic HILS system is being developed at Jackson State University. This system is funded by two grants, one from The National Science Foundation (NSF) and a second from the Army Research Office (ARO). The goal of this effort is to create a HILS system for small autopilots. This system will allow the autopilot to experience all flight stimuli while in the lab. A patent for this system has been submitted and is currently pending (Serial Number 12906778). This study describes how multiple units of this system can be used to create an environment in which multiple autopilots can fly multiple airframes in a single virtual environment. This will make possible the investigation of multi-agent systems in the lab allowing for a low cost and efficient way of examining and evaluating their behavior. The study proceeds to explore the different technical problems facing such an effort and the possible solutions for such problems.

2. ONGONG RESEARCH

2.1. The Monolithic HILS System

The latest system was design and fabricated at Jackson State University and the U.S. Army Research Laboratory. Each system allows one autopilot to be placed on a Three-Degree-of-Freedom (DOF) gimbaled platform (Shumaker *et al.*, 2008). This autopilot transmits flight surface and throttle information to the platform control hardware, which relays this information to the flight simulator to control a virtual airframe. In this case X-Plane is used because it is one of the most realistic and has been approved by the FAA for training purposes. The flight simulator calculates the new airframe attitude and transmits this information to the platform control hardware, which adjusts its actuators to match the aircraft attitude accordingly. Additionally, the simulator provides Longitude, Latitude and Altitude data to the control hardware. The control hardware is interfaced via Ethernet with a GPS Simulator (Spoofers). The GPS Simulator transmits the appropriate RF to the autopilot's GPS antenna. Furthermore barometric pressure, ram air pressure and temperature are recreated by means of a vacuum pump, fan and thermoelectric element respectively.

Corresponding Author: Kamal S. Ali, Department of Computer Engineering, Jackson State University, Jackson, Mississippi, USA

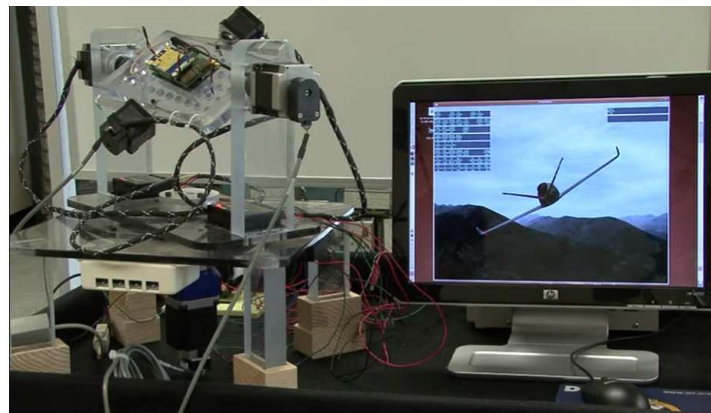


Fig. 1. Shows a prototype of the HILS system

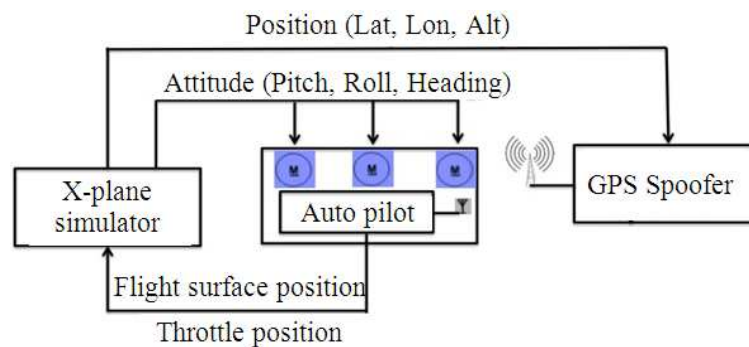


Fig. 2. Monolithic HILS system

These stimuli provide the autopilot with many of the same stimuli created under real flight conditions, while still in the laboratory. This being a virtual environment, the user can control all parameters such as wind gusts, rain, geographical location. It is also possible to repeat any scenario, making it possible to compare tune autopilots' performance as well as develop performance metrics for various sensors and algorithm combinations. **Figure 1**, shows an open-source autopilot placed on a prototype platform, flying a Cirrus jet in the X-Plane simulator environment. **Figure 2** shows a block diagram of the setup in **Fig. 1**.

The benefits of this system are not limited to creating a virtual environment for an autopilot. This system could be used to calibrate an Inertial Measurement Unit (IMU) by optimizing the contribution of gyros versus accelerometers to the final device's output. This is possible because each rotational pivot of this platform is equipped with a shaft encoder that continuously records the exact position of the platform, which can be compared to that of the IMU. This system has already demonstrated its ability improvement in IMU performance (Ali and Lamariou,

2008). It is therefore to aid in the rapid develop and optimization of an autopilot using this HILS system.

To create a Multi-Agent HILS the setup above needs to be duplicated for the necessary number of agents. The primary hurdle in doing so is the GPS Simulator usage. Having multiple GPS simulators operate in the same vicinity is physically impractical and prohibitively expensive. To use this system in a MAS setting, modifications are necessary.

2.2. The Gimbaled Platform

The gimbaled platform is the mechanical core of the monolithic HILS system. It comprises a system of stepper motors, gears and slip rings to route information from the control hardware to the autopilot while rotating without limits through any of the three degrees of freedom. The platform continuously measures heading, pitch and roll then adjusts its motors by measuring the difference between current position and commanded position. Communication with the control hardware occurs via Ethernet. The flight simulator and GPS signal generator computers connect to the control hardware through the built in Ethernet switch

ports. A four wire bus is routed from the control hardware to the autopilot, which provides custom routing of power and data specific to each situation. Only the x-plane simulation software is needed to interface the control hardware. All data translation and communication is processed by the control hardware, thereby eliminating additional computer software requirements.

2.3. The Autopilot

A primary condition for the creation of a UAV MAS environment is full and complete access to the autopilot source code, as this will eliminate the need for GPS hardware. Today, autopilot hardware and software are easily accessible and well documented in the literature. For the purposes of this MAS HILS system, an autopilot will be designed and built to use this system. The Atmel® SAM3 microcontroller will be used as the processing core of the autopilot. It is a 64Mhz 32-bit RISC processor with 256K-byte onboard flash memory. The processor is capable of handling 79 external interrupts. In addition, it comes equipped with a high-speed USB port, UARTS for RS232 and WiFi (IEEE 802.11). This processor is coupled with a GPS module (GPS-08724 antenna and a GPS-10920 Vinus GPS Logger) and a 6-DOF IMU (SEN-08726 IMU Sensor) to complete the autopilot's hardware. The autopilot software utilizes a PID controller for Throttle, Pitch and Roll, where rudder movement is correlated to that of ailerons. Although fairly simple, this autopilot works well in most conditions.

Once the monolithic HILS system is operational, it will be used to verify the functionality of and fine-tune the new open-source autopilot optimizing its performance prior to using it in the MAS HILS system. Having access to the source code of the autopilot makes it possible to deliver GPS data directly to it, bypassing the GPS antenna and eliminating the need for a GPS simulator/spoofers.

3. THE MAS HILS SYSTEM

3.1. Available MAS Simulators

With the exponentially increasing power of computers, the task of achieving intelligence through the collective behavior of decentralized agents has once more gained popularity. Researchers have created many algorithms that mimic the natural behavior of a swarm. These include "The Ant Colony Optimization" algorithm (Dorigo and Stutzle, 2004), "The Intelligent Water Drop" algorithm (Shah-Hosseini, 2009) and "The Krill Herd" algorithm (Gandomi and Alavi, 2012), all of which are swarm-based nature-inspired algorithms.

These algorithms can be used to govern the behavior of multi agent systems such as a swarm of UAVs. Verifying the efficacy of such algorithms through real flight can be quite challenging. It is therefore commonplace that these algorithms are verified through simulation.

Today, there is a multitude of Multi Agent Simulation packages, all of which are pure software. These packages include Net Logo (Sklar, 2007), MASON (Luke *et al.*, 2005), ABGPU (Railsback *et al.*, 2006) and FABS (Richmond and Romano, 2008), among many others. These packages create a virtual environment in which virtual agents may interact. This type of pure simulation presents one extreme of algorithm evaluation, with the other extreme being actual agents interacting in an actual environment. A middle ground that takes advantage of the best of both worlds is Hardware in the Loop Simulation, where the actual hardware (UAV autopilots) interacts with each other in a virtual environment. It has been suggested (Niazi and Hussain, 2011) that such a system can be of benefit; yet, to date no such system could be found in the literature.

3.2. The MAS-HILS

The MAS HILS system is a collection of monolithic HILS systems that are made to communicate with each other and share the same virtual environment. An autopilot on the monolithic HILS system has access to 4 circuits for communication with the outside world through slip-rings. This coupled with the fact that the autopilot's source code is completely accessible makes it possible to reconfigure the system as shown in **Fig. 3**. Here the Longitude, Latitude and Altitude data generated by X-Plane is fed directly to the autopilot eliminating the need for GPS simulation. In other words, modify autopilot's code so that rather than receiving GPS data from the GPS module, it gets that information from X-Plane data. This system can then be duplicated to create a MAS HILS. In such a MAS each instance/autopilot will communicate its parameters, such as position, velocity, to a central controller that will share that information with other agents according to the algorithm being executed. Therefore, an agent in this system will be aware of all the necessary neighboring agents' information allowing these agents to perform as swarm. **Figure 4** shows a platform for a single agent with a vacuum chamber attached to accommodate autopilots with pressure altimeters. This chamber will not be needed when the JSU autopilot is used since such autopilot uses GPS for altitude measurements. **Figure 5** shows a 4-Agent MAS HILS system.

Each agent in this system runs its own instance of the virtual environment (X-Plane), a separate screen is used to show the world as seen by that agent. An extra screen is connected to the central controller to show the relative position of all agents.

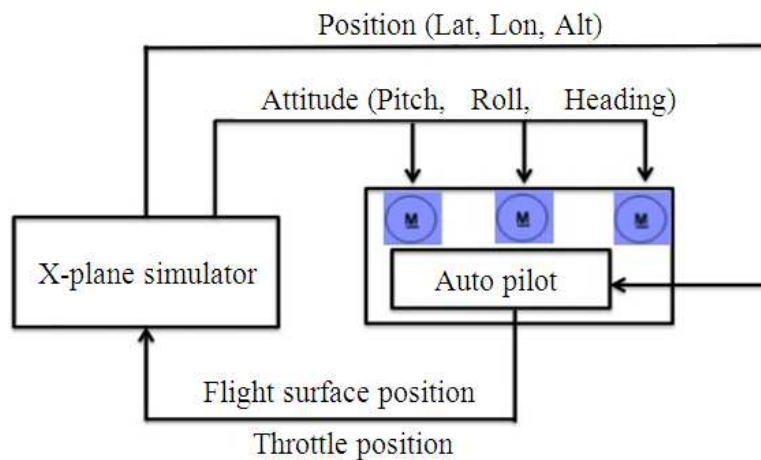


Fig. 3. A block diagram of a single agent in a MAS HILS system

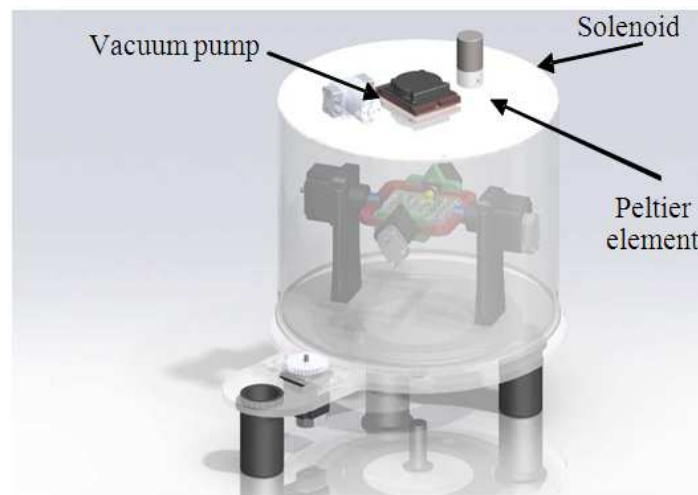


Fig. 4. Platform for a single agent

Another challenge facing the MAS-HILS is visualization. That is, to allow all instances of X-Plane to share the same visual environment.

It is possible to have two instances of X-Plane interface with one another, so that a screen of one agent can show the second agent as well. The question is how would the system behave when several instances of the simulator are made to share the visual space and would the benefits of such an effort justify the cost?

The way the system will operate is as follows: for a single instance/single agent, an autopilot placed on the 3-DOF platform (Atmel SAM3 based autopilot in this case) will provide the gimbal control hardware with actuator position data, which will relay transmit this

information to the X-Plane simulation software while gimbal control hardware adjusts the attitude of the gimbaled platform to reflect that of the simulated flight vehicle. Multiple agents (autopilots) will communicate with one another through Ethernet. Each autopilot will have a unique IP address. Communications between agents are not relayed through the gimbaled control hardware, but function as point to point nodes networked through an external switch.

The MAS HILS system described above is unique. It combines the flexibility of software simulation with the accuracy of hardware performance. Today there are a number of MAS simulators, all of which are solely based on software.

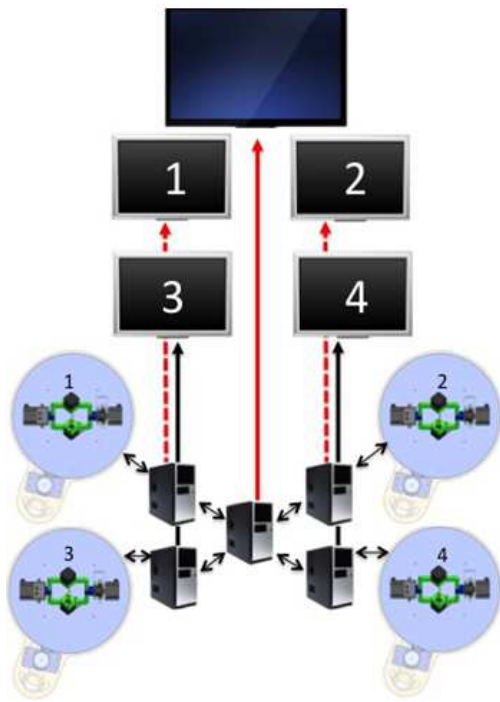


Fig. 5. a 4 agent MAS HILS system

The only other alternative to such simulators is actual flight, which is expensive and time consuming. The system described above, on the other hand, tests the real hardware with the software running on the autopilot, as if in real flight. This will result in a testing environment that is much closer to the real thing. This system is analogous to loading MAS code on actual UAVs and allowing them to fly. The only difference is that here all activities are carried out in the lab, where a crash is instantaneously recoverable.

This system is unique in that it will be the only HILS system for multiple agents, allowing real hardware agent to interact with each other in a virtual environment. It is expected that, when finished, a patent for this system will be sought.

3.3. Anticipated Hurdles

The primary components of the HILS platform (platform for a single agent) are 5 stepper motors, 3 slip rings and 3 shaft encoders, plus microcontroller boards and mechanical parts. The existence of this many motors in close proximity, coupled with the fact that signals and power will have to come close to one another in slip rings makes for a potentially noisy RF environment. Because of this, differential signaling

(Ethernet) is utilized to mitigate this problem, but may prove to be insufficient.

Developing the protocol to start, stop and reset the systems in tandem developing new software may prove to be time consuming. The Ethernet hardware in the current system operates at 10Mb/sec and therefore may introduce undesirable latency issues with so much traffic on the same bus.

3.4. Possible Solutions

The best way to handle the first hurdle is to minimize noise. Noise is generated by the Electromagnetic radiation (EM) from the stepper motors and from the induction due to the proximity of power and signal lines at slip-rings. If necessary, the first solution to be attempted is to upgrade from a 3 slip-ring system to a 5 slip-ring system. Here pitch and roll will be accessed by two slip-rings each. One slip-ring will be used for power and the other for signal. This leaves the heading slip ring, for which it is proposed to cut circular concentric grooves in the bottom side of the base gear, far from the center, where the heading slip-ring is, as possible. In these grooves copper rings with brushes will be placed to transport power up to the gimbaled platform. It is anticipated that this will reduce the noise levels to ones that will allow for wide band communications. If more noise reductions are needed, the stepper motors will have to be shielded. This will reduce the EM radiated and hence reduce noise. This will only be done if increasing the number of slip rings does not produce desired effects.

To minimize the effect of latency, an attempt will be made to characterize the latency of each component in the system, be that software or hardware. All data must reach its destination in less than the period of each simulation tick. A restriction on the number of nodes, a faster Ethernet link, or compression algorithms may be utilized to overcome any latency related issues.

3.5. Evaluation Techniques

Initially, multi-agent algorithms with known performance metrics will be selected to run on the central and system computers. All these computers being PCs will allow flexibility loading and running of the MAS software packages. These packages will communicate the desired agent's position, which is bundled with X-Plane's new platform's attitude and transmitted to the platform's controller and then to the autopilot that will in turn calculate new flight surface information.

In the second evaluation stages, the multi-agent control algorithm will need to be loaded on the

autopilots' hardware. For example, if a swarm was to fly in formation, it should be able to do so autonomously, with out communicating with ground stations. Once control algorithm is loaded on actual autopilots the system's performance will be contrasted with software simulation results. Finally, the performance of the MAS-HILS will be evaluated through real flight. Here, small UAVs will be flown while running the MAS control algorithm. Results of these experiments will be published in appropriate literature.

3.6. Novel Algorithm Exploration

Nature has long adopted the swarm technique enabling small unintelligent creatures to band together forming superior power and intelligence. There are a number of mathematical models that describe swarm behavior (Seung-Hwa *et al.*, 2010; Dervis, 2010). Many of these models can be adapted to a multi agent UAV system and be evaluated using the MAV-HILS system proposed. The advantage is that using this system a new idea can be loaded, evaluated, modified, re-evaluated and optimized in the laboratory making for quick progress. It is therefore expected that the second half of this project will be devoted to selecting mathematical models (nature inspired or otherwise) that can be adapted to a swarm of UAVs, rendering them more powerful and intelligent. These algorithms will be evaluated using the MAS-HILS system. This study will be done in collaboration with researchers from Pennsylvania State University (See support letter).

3.7. Anticipated Results

Upon the completion of this project a patentable, expandable 9-12 Agent HILS system will be developed. The system will be publicized and made available to other educational and industrial entities. Duplicate systems will be created and shared with collaborating entities. A detailed evaluation of some of the existing MAS algorithms will be conducted using this system. This system will be made available to students and faculty of JSU for teaching and research.

Since this system allows for the rapid evaluation of MAS algorithms, it is possible to explore the efficacy of novel MAS algorithms.

4. REFERENCES

- Ali, K. and C. Lamariou, 2008. Miniature-autopilot evaluation system. *J. Comput. Sci.*, 4: 30-35.
- Dervis, K., 2010. Artificial bee colony algorithm. *Scholarpedia*, 5: 6915-6915.
- Dorigo, M. and T. Stutzle, 2004. *Ant Colony Optimization*. 1st Edn., MIT Press, ISBN-10: 0262042193, pp: 305.
- Gandomi, A.H. and A.H. Alavi, 2012. Krill herd: A new bio-inspired optimization algorithm. *Commun. Nonlinear Sci. Num. Simulat.*, 17: 4831-4845. DOI: 10.1016/j.cnsns.2012.05.010
- Luke, S., C. Cioffi-Revilla, L. Panait, K. Sullivan and G. Balan, 2005. MASON: A multiagent simulation environment. *SIMULATION*, 81: 517-527. DOI: 10.1177/0037549705058073
- Niazi, M.A. and A. Hussain, 2011. A novel agent-based simulation framework for sensing in complex adaptive environments. *IEEE Sensors J.*, 11: 404-412. DOI: 10.1109/JSEN.2010.2068044
- Railsback, S.F., S.L. Lytinen and S.K. Jackson, 2006. Agent-based simulation platforms: Review and development recommendations. *Simulation*, 82: 609-623. DOI>10.1177/0037549706073695
- Richmond, P. and D. Romano, 2008. Agent based GPU, a real-time 3d simulation and interactive visualisation framework for massive agent based modelling on the GPU. *Proceedings of the International Workshop on Super Visualizationm, (SV' 08)*, Citeseerx Press, pp: 1-8.
- Seung-Hwa, S., K. Doo-Hyun and C. Chun-Hyon, 2010. Experimental reliability analysis of multi-UAV simulation with TMO-based distributed architecture and global time synchronization. *Proceedings of the 13th IEEE International Symposium on Object/Component/Service-Oriented Real Time Distributed Computing Workshop*, May 4-7. IEEE Xplore Press, Carmona, Seville, pp: 212-218. DOI: 10.1109/ISORCW.2010.23
- Shah-Hosseini, H., 2009. The intelligent water drops algorithm: A nature-inspired swarm-based optimization algorithm. *Int. J. Bio-Inspired Comput.* 1: 71-79. DOI>10.1504/IJBIC.2009.022775
- Shumaker, J., K.S. Ali and L. Carter, 2008. A gimbaled platform for MAV autopilot simulation and calibration. *Proceedings of the 27th Digital Avionics Systems Conference*, IEEE Xplore Press, Oct. 26-30, St. Paul, MN, pp: 4.C.4-1-4.C.4-10. DOI: 10.1109/DASC.2008.4702832
- Sklar, E., 2007. Software review: Netlogo, a multiagent simulation environment. *Artif. Life*, 13: 303-3011. DOI: 10.1162/artl.2007.13.3.303