AERO-Beam: An Open-architecture Test-Bed for Research and Education in Cyber-Physical Systems

Orion Lawlor*, Seta Bogosyan**, Yasin Vural***, Isaac Thompson**, Michael Moss*, Metin Gokasan**

* Department of Computer Science
University of Alaska Fairbanks
Fairbanks Alaska, USA

**Dept of Electrical & Computer Eng.
University of Alaska Fairbanks
Fairbanks Alaska, USA

***Department of Control Eng.
Istanbul Technical University
Istanbul, Turkey

***Abstract*— As cyber-physical systems (CPS) increasingly become a part of daily life, a well-prepared workforce is needed to build and maintain, calling for new approaches and test-beds to introduce CPS concepts into classrooms at all levels. The literature has many examples of CPS relevant platforms, which are used successfully in research; however, especially the commercial ones are mostly closed-architecture and do not provide an in-depth engagement for the students in all aspects of the hardware and software. This paper introduces a novel platform, known as the AERO-Beam, which conveys many basics of CPS, while also introducing the dynamics and control fundamentals of quadcopters. Quadcopters are widely used in CPS projects, yet their limited batteries, rapid control response, and tendency to crash makes them difficult to use in CPS education. The AERO-Beam addresses the need in many ways, as demonstrated through a multi-level classroom implementation.**

Keywords— cyber-physical systems, CPS, education, quadcopter

I. Introduction

Cyber-physical systems (CPS) are a new generation of smart systems that consist of computational and physical components, hardware and software seamlessly integrated to closely sense and interact with the physical world. These systems involve a high degree of complexity at numerous spatial and temporal scales and highly networked communications integrating computational and physical components [1].

CPS are rapidly becoming a part of our daily life, and are critical to industries and commerce, with applications in smart manufacturing systems, smart infrastructure (smart grids and buildings, road-weather systems), transportation (intelligent vehicles and traffic control systems), emergency response (detection and surveillance), medicine and healthcare (body area networks, telesurgery robots, assistive systems), education (telepresence robots) and defense. The increasing use and demand for CPS also increases the need for a well-prepared, well-trained workforce, hence an education system that also accommodates this need at all levels.

In recognition of this emerging need in the last few years, there have been numerous workshops and discussions as to how to bring CPS into the education system at all levels, not only at graduate level, but also at undergraduate and K-12. Recently, there have also been some reported studies addressing specific CPS education activities for different student levels: [3] presents an 8-week multi-level student project, based on an web accessible quadcopter and rover performing a coordinated search and rescue mission as an example of CPS; [4] presents the iRobot Create based UPBOT and course activities to introduce EE undergraduates to the programming and networking aspects of CPS based on a Contact Learning approach; [5] describes an obstacle avoidance Roomba project that involves high school students in the modeling and simulation aspects of CPS; [6] , [7] [8] present lecture notes combined with lab exercises to introduce CPS concepts at UC Berkeley, and [9] proposes a combination of simulation software and virtual computing to enable online teaching of CPS, also using the iRobot Create based Cal Climber. In most of these studies, the iRobot Create has been the platform of choice as it allows for a CPS composition with a robotics platform modeled as a subsystem and treated as a collection of sensors and actuators located beyond a network boundary.

While valuable educationally, the reported activities at undergraduate and K-12 level often focus more on robotics aspects and much less on CPS. However, the hardware-software integration perspective of CPS calls for more multi-dimensional research and education activities that involve actuator and sensor networks, data fusion, decision making, coordination, and networking along with embedded control, modeling, and simulation activities. [3] addresses this need to some extent with research and education activities planned around the coordinated operation of the commercial AR Drone with a rover to perform a coordinated search-and-rescue mission. However, the authors also mention the limitations posed by the closed-architecture of the quadcopter, which has not allowed for sufficient focus to be placed on sensor networks, data fusion, and control, all of which are critical components of CPS. This paper aims to address this issue.

As also evident from the existing literature, autonomous aerial (UAV) and ground vehicles (UGVs) are among the most popular platforms for research and education activities in CPS. Quadcopters, which are of particular interest for young students at all age groups, make great test-beds for many different aspects of CPS, especially when their autonomous control systems are involved. However, for a thorough experience with the multitude of sensors and actuators in the control system, the quadcopter should offer an open-architecture configuration, and be robust to control errors. On the other hand, besides the fact that such systems offer benefits in terms of providing the full range of engineering and computer science experience for the students, even simple experiments of tuning control parameters could result in damaging crashes, hence, call for alternative measures to protect the system. Whether closed- or open-architecture, another issue to consider with the quadcopter as an educational system is the complexity of its dynamics and control. This often results in educational approaches that engage the students minimally in science and physics, with the system dynamics often taken into account as a black box. The limited involvement in system dynamics together with the closed-architecture of such systems also results in control exercises that are limited to the outer loop only, with the inner loop control taken as another black box.

The platform of choice to introduce students to CPS, then, would be a simple, robust, and unbreakable open-architecture platform that delivers the main concepts of the larger, more complicated platforms. Such educational platforms for CPS are very few, if any. A unique characteristic of the AERO-Beam is that the students could be involved in all aspects of its design, starting with the construction of the beam, using 3D-printing. The main aim of this paper is to introduce this novel test-bed and discuss how it could be used for integrated research and education activities of CPS in a level appropriate manner. The platform is designed engage the students adequately in the modeling, simulation, sensor networks, data fusion, and control fundamentals of CPS, while also acting as an introduction to the dynamics and control basics of quadcopters and UAVs.

A similar structure known as Aeropendulum [10],[11], [12], [13], [14] also uses thrust force for the actuation of the arm and is controlled directly through a computer via a Matlab/SIMULINK interface. However, the Aeropendulum is designed mostly for education in control theory, uses only an encoder for feedback, and does not involve sensor networks or data fusion. The AERO-Beam and the associated balancing control, on the other hand, uses an Arduino as the embedded controller, is equipped with an accelerometer, gyro, and an encoder, allowing for their fusion, hence, provides a structure that is also adequate for CPS research and training. The AERO-Beam is also designed to convey the basic physics and control concepts of quadcopters, with the strong correlation between balancing control of the beam, and stabilization control of the quadcopter. The system could also offer a real-time animation and user interface to enter parameters on-line as well as for the monitoring of system performance. The modeling and simulation approaches, sensor/data fusion schemes and control algorithms could be designed and implemented on the system to accommodate research and education for students at multiple levels, i.e. graduate, undergraduate, and high school.

This paper is organized as follows: following this Introduction in Section I, Section II provides an overview for the use of the Aero-Beam in CPS research and education. This is followed by a Case Study in Section III where an undergraduate level classroom implementation is discussed with the results. Finally, a brief conclusion is provided in Section IV.

II. AERO-BEAM AND ITS USE FOR MULTI-LEVEL CPS RESEARCH AND EDUCATION

In this section, the structure of the AERO-Beam will be presented and its use for a multi-level research and training will be discussed for modeling, simulation, sensor networks, data fusion, and control aspects of CPS.

The AERO-Beam is composed of a metal (or plastic) beam, an Arduino Uno as the controller, a brushless DC motor that creates the torque, a propeller that converts the torque into a thrust, and a gyro and accelerometer to provide angular feedback. Fig 1 depicts the constructed beam and the numbers indicate the above mentioned components.

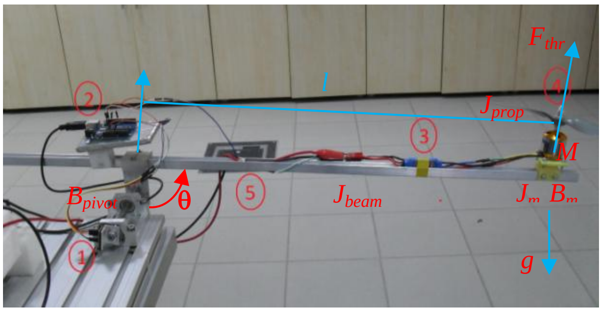

Fig. 1. Aerobeam and components

Components of the system:

1. MPU-6050 sensor package (3-axis accelerometer, 3-axis gyroscope)
2. Arduino Uno microcontroller
3. ESC (Electronic Speed Controller)
4. Brushless DC motor (BLDCM) and Propeller
5. Beam (aluminium, 75 cm between center of rotation and motor)
6. The power supply is 8 volt and runs a current of up to 10 A.

The Arduino Uno, chosen for its simplicity, provides a servo-style PWM signal to the ESC. This control signal is proportional to the angular velocity reference of the BLDCM, which actuates the beam. The required force (thrust force) to balance the beam against gravity is provided through a motor-propeller system. Based on the control input signal, the ESC drives the motor at an appropriate angular velocity (dictated by the control signal received from Arduino). The generated motor torque is then converted to a thrust force through the propeller, which moves the beam up.

For the user interface, a desktop or laptop computer is used, which communicates with the Ardunio controller. The web-accessible user interface allows for reference angle and control parameter entries by the user, also providing feedback data to monitor the performance. via real-time plots of the system variables, as well as animations of the beam. The interactive user interface is another important component of CPS and its web accessibility makes the AERO-Beam a highly versatile test-bed for remote labs.

Having introduced the basic structure of the AERO-beam system, we can now discuss how it could address some CPS fundamentals, and act as an introduction to the dynamics and control of aerial vehicles, especially quadcopters.

The AERO-Beam can be used for integrated research and education activities in the following components of CPS:

A. *Modeling and Simulation*

While the quadcopter design might be complicated for students without the appropriate physics and dynamics background, the AERO-Beam is particularly designed as a simplified version of a quadcopter to provide a smooth transition for beginner level students into dynamics concepts, while also accommodating more advanced students for their research and education needs in dynamics modeling and simulation.

The beam configuration and its balancing control cover almost all fundamentals of Newtonian physics and dynamics, from torque and thrust force, to weight, mass, and acceleration concepts. At beginner or high school level, Excel could be used as the modeling and simulation environment and students could also be engaged in activities to determine the many physical relationships in the system; i.e. commanded ESC signal pulse width versus resulting motor RPM (measured with a tachometer or encoder), thrust (measured using stacks of pennies as weights, and increasing PWM until the thrust balances the weights), and the torque-thrust and propeller dimension relationships. This level of engagement in physics and dynamics would not be possible with the actual quadcopter system, and would force the instructors to take more of a black-box approach for the system dynamics.

Derivation of a dynamics model for the beam is an exercise for more advanced students, and would be essential for the design of model-based controllers as the next step. Students at this level would also be potentially prepared to validate the derived system dynamics through a MATLAB/Simulink based simulation, rather than using Excel. At this level, the modeling exercise could be further enhanced by combining it with adaptive estimation and system identification research. At that point, a moving mass could also be added to the beam to create the changing inertia effects of the quadcopter.

Both student levels could also be involved in the design of a 2D animation for the AERO-Beam system, or the animation could be provided as a feature, and the students involved in the process at a more minor level; i/e. changing certain parameters etc.

B. Feedback via Multiple Sensors and Data Fusion

The current design of the AERO-Beam presented in Fig.1 allows for experimentation with the accelerometer, or gyroscope separately, as each sensor could be used individually to provide angular position feedback for the stabilization control. Hence, the individual use of these sensors would serve as a valuable exercise for beginner students to understand the vibration and drift problems associated with accelerometers and gyroscopes, respectively, and experience the benefits of fusing different sensors via simple data fusion algorithms. An encoder is also connected to the rotational center of the beam to give a better understanding of gyro drift and its effect on the physical position of the beam.

The Complementary Filter scheme is simple and highly intuitive for the concept of fusion, hence would serve as an informative introduction to data fusion. This approach could be used both for high school and beginner level college students. The more advanced students could use more sophisticated filtering and fusion schemes, such as Kalman Filters.

C. Controller Design

The balancing problem of the AERO-Beam could be used to introduce a wide variety of control approaches to students at different levels. After demonstrating the benefits of feedback control over open-loop control, high school and beginner level college students could be taken through the full range of P, PD, PI, and PID control design. This experience could further be enhanced by demonstrating that the PID control is not the perfect solution to the position control problem in this case, getting the students attention to many nonlinearities in the system as the cause of the problem, and providing a smooth transition into model-based control design. A PID+ controller (based on trial-and-error) could be a good introduction to model-based control for such students, after demonstrating the inadequacy of the PID control for perfect balance.

Advanced level students could further dive into the causes and modeling of the nonlinearities; i.e nonlinear thrust-angular velocity relationship, nonlinear load, stiction friction, dead-zone created by the slow build up of thrust from torque etc. Starting with PID and PID+, adaptive estimation approaches could also be developed to support the design of more sophisticated model-based control, and feedback linearization schemes.

Finally, advanced students could also be engaged in the design of robust control schemes, such as sliding mode control.

III. CASE STUDY FOR CLASSROOM AERO-BEAM IMPLEMENTATION

In this section, we will present a 2-month long pilot classroom project that involved the construction, and control of the AERO-beam through the collaboration of two teams that involved a mixed (undergraduate and graduate) group of students from University of Alaska Fairbanks (UAF) of USA and Istanbul Technical University (ITU) of Turkey. The project brought together a group of Computer Science and Electrical Engineering students from UAF, and ITU. Each team designed and controlled a separate AERO-Beam, however shared the design and collaborated in different aspects of the project. The UAF team also mentored a high school engineering classroom of 20 students in a 8-week long CPS Challenge preparation, for which the AERO-Beam test bed was used for an introduction to CPS, as well as to the fundamentals of quadcopter dynamics and control. The CPS challenge involved the development of a UAV and UGV based search-and-rescue system. This challenge activity is an extension of the NSF funded CyberAlaska grant, which is the first known grant [3] in USA for the preparation of the future workforce in CPS. This section will provide some details about the design and implementation of the AERO-Beam, as well as some results obtained through data fusion and control activities with the system.

A. Modeling of the AERO-Beam

The project started with the derivation of the AERO-beam dynamics model with the use of Newtonian principles:

$$F_{thr}\, l = J_{beam}\, \theta + B_{pivot}\, \theta + T_c + M\, g\, l \sin\theta \qquad (1)$$

Dynamics directly affecting to the motor:

$$J_m + J_{prop} \;\; \omega_m + B_m \omega_m = T_m = K_t i_q \qquad (2)$$

The voltage equations of the BLDCM,

$$v_q = R\, i_q + e_b = R\, i_q + K_b\, \omega_m \quad (3)$$

where T_m is the motor torque, i_q is the input current, Kt is the torque proportionality constant, v_q is the voltage drop across the motor, R is the motor resistance, ω and ω_m are the angular velocities of the beam and motor, respectively and K_b is a proportionality constant (indicating back-EMF, e_b generated per RPM). J_{beam} ,J_m,J_{prop} are the inertia of the beam, motor and propeller, respectively; B_{pivot},B_m are the viscous friction constants of the pivot and the motor, respectively; T_c is the Coulomb friction torque the power can be obtained as,

$$P = e_b\, i_q = \frac{K_b\, T_m\, \omega_m}{K_t} \quad (4)$$

This power will keep the beam balanced. By conservation of energy, we know that the energy the motor expends in a given time period is equal to the force generated on the propeller times the displacement of the air it moves [15],

(P ·d t = F_{thr} ·d x). Equivalently, the *power* is equal to the thrust times the air velocity

$$P = F_{thr}\, \frac{dx}{dt} = F_{thr}\, v \quad (5)$$

We assume vehicle speeds are low, so v is the air velocity when hovering. We also assume that the free stream velocity, $v\infty$, is zero (the air in the surrounding environment is stationary relative to the AERO-beam). Momentum theory gives us the equation for hover velocity as a function of thrust,

$$v = \sqrt{\frac{F_{thr}}{2\, \rho\, A}} \quad (6)$$

where ρ is the density of the surrounding air and A is the area swept out by the rotor. Note that in the general case, $T = r \times F$; in this case, the torque is proportional to the thrust $T_m = F_{thr}\, K_{thr}$.

by some constant ratio, K_{thr} determined by the blade configuration and parameters.

$$P = \frac{F_{thr}^{3/2}}{\sqrt{2\, \rho\, A}} = \frac{K_b}{K_t}\, T_m\, \omega_m = \frac{K_b\, K_{thr}}{K_t}\, F_{thr}\, \omega_m \quad (7)$$

Using our simplified equation for power, we can then write,

$$F_{thr} = \left(\frac{K_b\, K_{thr}}{K_t}\right)^2 2\, \rho\, A\;\; \omega_m^2 \quad (8)$$

Solving for the thrust magnitude F_{thr}, we obtain that thrust-angular velocity relationship as below:

$$F_{thr} = k\, \omega_m^2 \quad (9)$$

where k is some appropriately dimensioned constant.

As previously mentioned, the derivation of the thrust-angular velocity relationship at high school classrooms could be performed via the empirical plots between these variables with the use of Excel, and by taking current, RPM, and PWM readings for different weights placed on the beam.

B. PID and PID+ Design for AERO-Beam Balance Control

The $Mg \sin\theta$ type gravitational load of the beam (M consisting of the mass of the beam, motor, propellers etc) becomes a constant value for a given angular position reference (90 degrees in this case). Hence, as in all constant load systems, a PID type controller should be sufficient theoretically to achieve a given position with zero steady state error. However, the actuation of the aerobeam system depends on a thrust-velocity relationship which is very nonlineer. There is also a dead zone effect caused upon initial actuation due the latency between the motor torque being converted to upward thrust. Intuitively, it becomes necessary to add a certain value to the PID output to compensate for these nonlinearities. While this intuitive first-step to PID+ could be performed roughly through trial-and- error, more systematic approaches can also be taken.

Upon the derivation of the angular velocity-thrust relationship, we design the control scheme, starting with PID and then, PID+ to further improve the control performance. To calculate the plus term, u_+ of the control signal, we use the relationship between the thrust force and the load of the beam

$$F_{thr} = \frac{J_{beam}\, \theta + B_{pivot}\, \theta + T_c}{l} + Mgsin\theta = k\, \omega_m^2 \quad (10)$$

Combining these equations,

$$\omega_m = \sqrt{\frac{mg\, l\, sin\, \theta}{k\, l}} + \sqrt{\frac{J_{beam}\, \theta + B_{pivot}\, \theta + T_c}{k\, l}} \quad (11)$$

Considering that we are controlling the motor velocity in this problem, (11) gives the configuration of the PID+ control, with

$$u_+ = \sqrt{\frac{mg\, l\, sin\, \theta}{k\, l}} \quad (12)$$

While the second part of (11) is compansated by the PID control. The dead zone effect may also be compansated by the addition of a constant term, determined by trial and error at this stage.

C. Demonstrating Performance Improvement via Multiple Sensors and Data Fusion

- **Performance with Accelerometer Only**

To demonstrate the benefit of multiple sensors and data fusion for improved feedback and control, our experiments start with the use of accelerometer only for the determination of angular position. As the accelerometer is affected by external forces, such as vibrations, where it is placed on the beam becomes very important to reduce the vibrations. The sensor location can be seen in Fig 2.

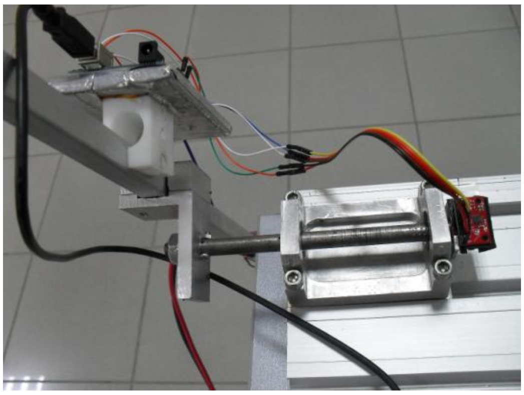

Fig. 2. Accelerometer location

At this given location, the sensor is exposed to minimum vibration. This position makes the sensor move along the y-axis, hence the y-axis will be used in the project. The determination of the angular position, theta, through the accelerometer readings can be explained with the following example:

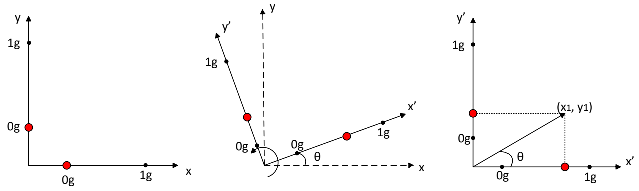

Fig. 3. Calculation of angular position using accelerometer

In the left most plot of Fig.3, the accelerometer is at rest. When the accelerometer under goes a rotation of θ_a, this value can be calculated using:

$$\theta_a = \arctan \frac{y_1}{x_1} \qquad (13)$$

With the feedback control applied to the beam based on accelerometer feedback, the system performance is as below:

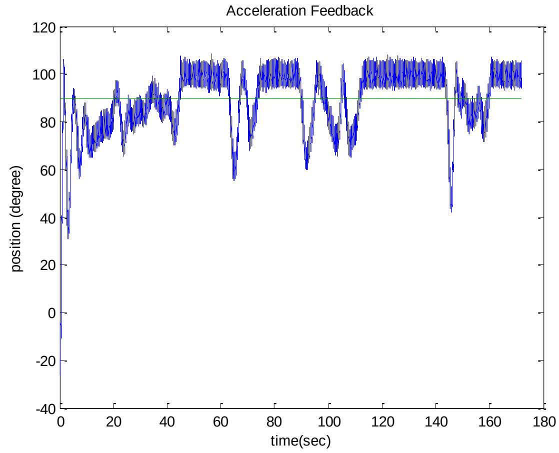

Fig. 4. Example for calculation of angular position using accelerometer

This performance depicted in Fig.4 reflects the well-known effects of vibrations and noise associated with accelerometers.

- **Performance with gyroscope only**

Similar to the accelerometer, the gyroscope can not provide angular information directly, and instead provides angular velocity information, which is converted to angular position through an integration process.

$$\theta_g\ t = \ \omega_g(t)dt \qquad (14)$$

θ_g: Angular position ,ω_g: Angular velocity of gyroscope
The control performance resulting from gyro feedback alone is depicted in Fig 5.

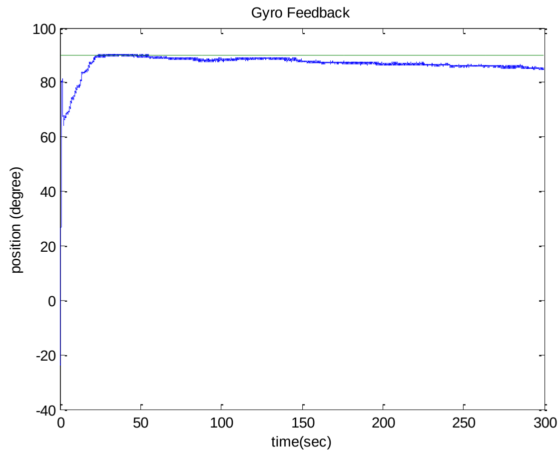

Fig. 5. Control results with gyroscope feedback alone

The drift of the output observed in the long run is an expected outcome of the integration process. An encoder was used to capture the drift of the physical beam.

- **Performance with Fused Accelerometer and Gyro Feedback:**

In this section, the benefits of data fusion (in this case performed by fusing accelerometer and gyroscope data) for an improved control performance will be demonstrated through the design of a simple Complementary Filter. The filter combines the accelerometer and gyro outputs in a weighted manner, with weights determined by trial-and-error to minimize noise as below:

$$\theta_c(k+1) = \alpha\left(\theta_c(k) + \theta_g(k)\right) + (1-\alpha)\,\theta_c(k) \qquad (15)$$

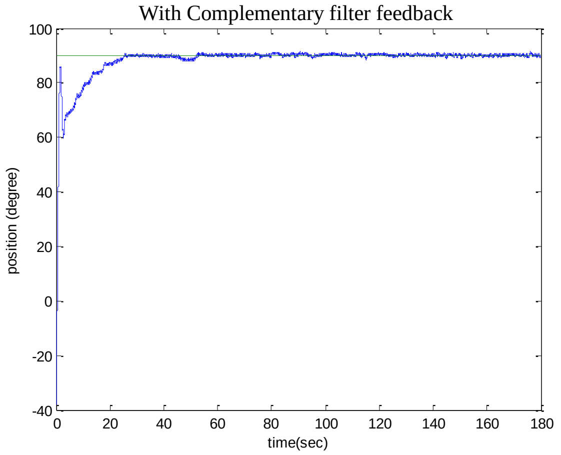

Fig. 6. Results with complementary filter

By inspecting Fig 6, the improvement in the control performance can be noted. This is due to the fused use of data, which has reduced the issues associated with each sensor.

D. Aero Beam Simulation

A key part of our approach to teaching cyber-physical systems is the use of simulated hardware. Simulations provide much better read/write access to the true internal state of the system than physical hardware, where the internal state can only be read by reconstructing it from a limited set of sensors, and the internal state is can only modified via a limited set of actuators. The ability to use simulated hardware is quite useful for scaling out hardware exercises across an entire classroom, where physical components would be expensive to purchase, and time consuming to assemble, inventory, and maintain. Simulated hardware also facilitates the development of the user interface and initial construction of the control system, where physical hardware could be dangerous to operate with unfinished software.

For the AERO-Beam, we have built a high-fidelity graphical simulator in JavaScript that runs in a web browser, see Figure 7. This simulator includes a live online charting interface, with the simulation moving to the right down the time axis as simulated time progresses, and simulation variables increasing and decreasing onscreen, leaving colored chart lines. The beam's height is illustrated as a UAV moving up and down onscreen, the fan thrust is illustrated by changing the length of an upward-facing red arrow indicating the net force vector, and the derivative terms are described as rates for students who are not comfortable with calculus.

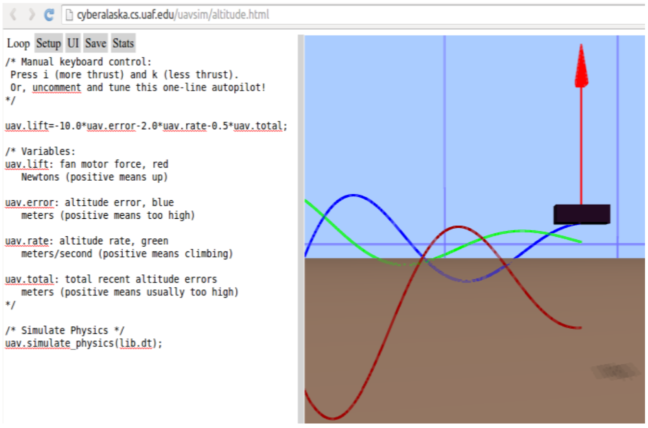

Fig. 7. AERO-Beam web simulator interface. The "error", "rate", and "total" terms correspond to the proportional, derivative, and integral terms of a PID controller.

A recurring theme in CPS is the tight interplay between the cyber and physical aspects of the system. We have experienced a similar interplay between our simulations and physical training models—as we improve the user interface on our simulation, we find students would prefer to use the same interface to interact with the physical model, rather than the simpler interface we had initially presented.

We have found it is useful to use the same human-machine interface and online control system to control either a simulation or a physical system, by using the same interchangeable programming interface for both simulated and physical components, and only deciding between them at runtime. Our implementation of this uses a web-accessible NoSQL storage system called SuperStar, where the pilot user interface continually posts control commands which are queried by the low-level hardware, and the hardware posts sensor data which are queried by the pilot user interface. Because neither the pilot user interface nor the hardware interacts directly, they can run on different machines across the network, be written in different languages such as JavaScript for a web user interface and C++ for the hardware, and each one can be exchanged without affecting the other.

IV. Conclusion

This paper presents the design of the AERO-Beam, a novel platform to convey CPS concepts to students at all levels. The main benefit of the beam over other existing platforms is that it offers a low-cost open-architecture configuration that provides access for students to all aspects of the system's hardware and software, and to CPS fundamentals, from modeling and simulation to sensor networks, data fusion, and control. The platform also provides a well-rounded introduction to quadcopter dynamics and control, providing a robust, crash-free experimentation tool. The system is also equipped with a web-accessible user interface, and could be used in remote labs.

ACKNOWLEDGEMENTS

This work was supported by NSF Award #1045601.

REFERENCES

[1] Foundations for Innovation in CPS, Workshop Report, Prepared for NIST, Workshop record, [Online]. January 2013.

[2] S. C. Suh et al. (eds.), Applied Cyber-Physical Systems, DOI: 10.1007/978-1-4614-7336-7_1,_ Springer Science+Business Media New York 2014

[3] Orion Lawlor, Mike Moss, Steven Kibler, Charles Carson, Shaun Bond,Seta Bogosyan, "Search-and Rescue Robots for Integrated Research and Education in Cyber-Physical Systems" IEEE ICELIE, Nov, 2013.

[4] Tanya Lee Ann Crenshaw, "Using Robots and Contract Learning to Teach Cyber-Physical Systems to Undergraduates", IEEE TRANSACTIONS ON EDUCATION, VOL. 56, NO. 1, February, 2013

[5] Vijay Gadepally, Ashok Krishnamurthy, Umit Ozguner, "A Hands-on Education Program on Cyber Physical Systems for High School Students", 3 Aug 2014

[6] J. C. Jensen, E. A. Lee, and S. A. Seshia, "Teaching Embedded Systems the Berkeley Way," in Workshop on Embedded Systems Education (in conjunction with ESWeek), Tampere, Finland, October 2012.

[7] ——, "An Introductory Capstone Design Course on Embedded Systems," in International Symposium on Circuits and Systems (ISCA S).Rio de Janeiro, Brazil: IEEE, 2011, pp. 1199–1202.

[8] ——, An Introductory Lab in Embedded and Cyber-Physical Systems. Berkeley, CA: LeeSeshia.org, 2012. [Online]. Available: http://LeeSeshia.org/lab.

[9] Jeff C.Jensen, Edward A.Lee, Sanjit A. Seshia, "Virtualizing Cyber-Physical Systems: Bringing CPS to Online Education, [online]

[10] Enikov, E.T. and Campa, G. "Mechatronic Aeropendulum: Demonstration of Linear and Non-Linear Feedback Control Principles with Matlab/Simulink Real-Time Windows Target," IEEE Transactions on Education, v. 55(4), pp. 538-545, 2012.

[11] Giuseppe Habib, Akos Miklos, Eniko T. Enikov, Gabor Stepan, and Giuseppe Rega, "Experimental Validation of the Act-and-Wait Control Concept Through the Aeropendulum", Proceedings of 11 International Conference on Vibration Problems,Lisbon, Portugal, 9–12 September 2013

[12] Enikov. E.T., Eke, E. "Teaching Classical Control System Course with Portable Student-Owned Mechatronic Kits", Proceedings of ASME International Mechanical Engineering Congress & Exposition, Houston, November 9-15, 2012,Paper IMECE2012-86700.

[13] Enikov, E.T., Campa, G. "USB-powered Portable Experiment for Classical Control with Matlab Real-time Windows Target," Proceedings 119th ASEE Annual Congress, San Antonio, TX. June 10-13, 2012

[14] Enikov. E.T., Polyzoev, V., Gill J. "Hardware Demonstration of Classical Undergraduate Control Design Methods Using Matlab Real-Time Windows Target Environment", Proceedings of ASME International Mechanical Engineering Congress & Exposition, Denver, November 11-17, 2011, Paper IMECE2011-62491.

[15] http://andrew.gibiansky.com/downloads/pdf/Quadcopter%20Dynamics,%20Simulation,%20and%20Control.pdf