Mission-Aware Cyber-Physical Optimization on a Tabletop Satellite

Justin M. Bradley,[*] Meghan L. Clark,[†] Ella M. Atkins,[‡] and Kang G. Shin[§]
University of Michigan, Ann Arbor, MI, USA

As digital and physical systems become more tightly integrated, multi-disciplinary design will be necessary to maximize total-system efficiency. Mission objectives and success of the system as a whole are becoming increasingly dependent on appropriate allocation of computational resources balanced against demands of the physical actuation systems. In this paper we adapt and apply a cooptimization scheme considering tradeoffs between costs associated with physical actuation effort required for control and computational effort required to acquire and process incoming information. We use TableSat, a tabletop satellite, as a real-world testbed to investigate specifics of cyber-physical cost terms and their tradeoffs. A multi-disciplinary cost function minimizes energy and maximizes mission efficiency and effectiveness. We examine simulated results generated using numerical methods and demonstrate that excluding either cyber or physical cost terms results in reduced performance for the holistic system over the course of the mission. These theoretical results are then verified using experimental data from the TableSat platform.

I.   Introduction

Modern systems require sensors, actuators, algorithms, and real-time digital systems to coordinate their activities with a physical system to achieve designated goals. Often each of these individual subsystems are designed independently to meet performance objectives. Important system design properties such as compositionality and composability can suffer without co-design techniques that account for limitations and strengths of each subsystem as well as the physical objects with which the system interacts. Additionally, as systems become smaller, requiring less energy for actuation and sensing, computational (cyber) resources begin to demand energy comparable to that of the physical system.

Control systems engineers attempt to optimize physical system trajectories by the proper application of force over time. Physics-based models of system dynamics including saturation constraints and other nonlinearities are used to design control laws that achieve designed trajectories - most often in the continuous time domain. On the other hand, real-time systems engineers, using discrete mathematical tools, optimize task allocation and scheduling over processor, communication, and I/O resources to guarantee performance deadlines for reliability and robustness. Good design of the task schedule may provide enough slack so that energy can be conserved through variable speed processors. Alternatively, slack in the task schedule may allow for aperiodic and sporadic task guarantees thereby providing event-driven capabilities or simply just increased service of individual tasks.

For systems which must more carefully manage all their physical and cyber resources together to achieve their objectives, globally-optimal (minimum-energy, minimum-time, maximum-information) performance can only be achieved by identifying and exploiting coupling between cyber and physical resources during the design process. Cyber resources provide the means for guidance, navigation, and control of the physical system, as well as the estimation of states, communication, and processing of information. The physical system, in turn, provides the ability to acquire information, survey an area, or take important measurements.

---

[*]PhD Student, Aerospace Engineering Dept., AIAA Student Member, justyn@umich.edu
[†]PhD Student, Electrical Engineering and Computer Science Dept., AIAA Student Member, mclarkk@umich.edu
[‡]Associate Professor, Aerospace Engineering Dept., AIAA Associate Fellow, ematkins@umich.edu
[§]Kevin and Nancy O'Connor Professor of Computer Science, Electrical Engineering and Computer Science Dept., kgshin@umich.edu

American Institute of Aeronautics and Astronautics

We have been conducting research to try and accomplish such optimal cyber-physical co-design. In recent work we developed a multidisciplinary approach for optimizing over both cyber and physical resources, including mission goals and objectives.[1,2] Using metrics encompassing physical system energy, time, surveillance information, and cyber utilization we showed that we can more appropriately balance overall system performance. We also used Pareto front analysis to examine some of the coupling between cyber and physical resource use.

In this paper we take this research a step further by trying to experimentally validate this technique. To this end we have adopted TableSat,[3] a one-degree of freedom rotating platform emulating a small satellite to demonstrate this method of design. TableSat, shown in Figure 1, uses computer fan actuators, rate gyros, and accelerometers controlled by an on-board Gumstix computing platform to control its rotation. We have
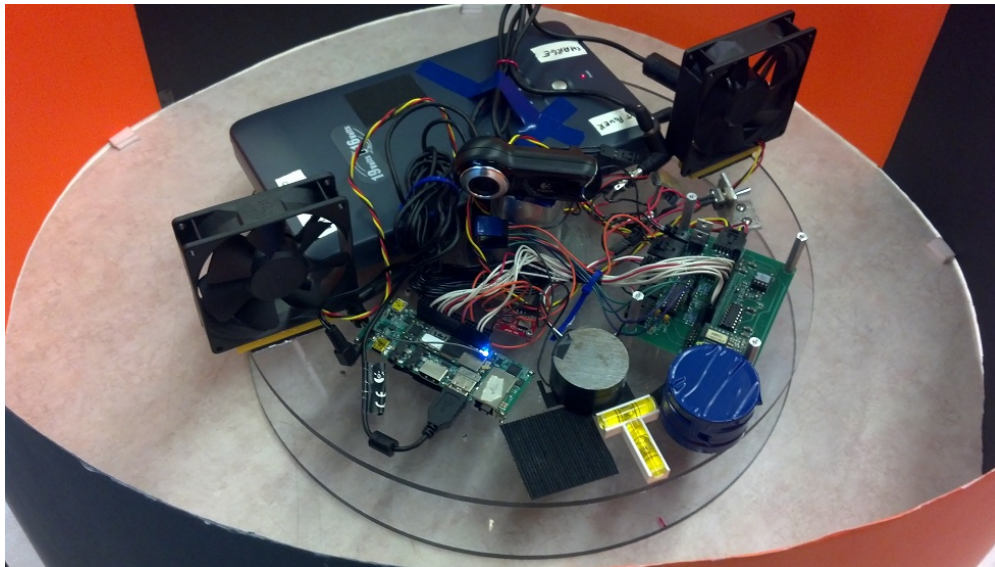

Figure 1. TableSat

fitted TableSat with a camera and propose an example mission to emulate a satellite gathering important data. The example mission is to collect the maximum amount of "information" which we define to be number of orange pixels viewed by the camera. The colored paper setup can also be seen in Figure 1.

We first introduce some related work in the fields of real-time systems and aerospace systems. We then describe the cost function including the metrics describing the competing mission objectives. A description of our experimental setup follows, after which we present analytical and subsequently experimental results.

II.   Related Work

The design of control systems under the constraints of cyber resources is not a new area of research. Anytime control[4–6] tries to improve control accuracy as cyber resources become available. Feedback scheduling[7–9] is a technique wherein cyber resources are modified according to the needs of the cyber system. However, only recently in the context of aerospace, satellites, avionics systems, surveillance, and UASs has there been increasing work attempting to address this important issue.

Agrawal et. al. explore some of the reasons why more advanced control algorithms are not used in modern avionics systems. They conclude that a Quality of Service (QoS) approach is needed to address the problem and they propose an adaptive resource management scheme for a real-time avionics system using anytime control and accompanying nontraditional task scheduling.[10] Russ and Stütz proposed a higher-level style of resource management that includes task-based guidance and navigation and perception plans. Their method focuses on finding algorithmic solutions adapting to perceptual demands that vary during flight as well as balancing those demands with sensory and computational resources.[11] Narayan et al. present a novel computationally adaptive trajectory decision optimization system that can dynamically manage, calculate, and schedule task execution parameters.[12] An offline and online component work together to increase overall mission efficiency.

American Institute of Aeronautics and Astronautics

For satellite systems information gathering by imaging systems generally happens in a relatively short time window (in low earth orbits around three min) during which the system must maximize its efforts to collect the data. There is generally a 10-15 minute window during which the system can prepare resources for intense data collection. Traditionally, such task scheduling problems have been addressed by teams of planners on the ground using write and check procedures.[13] However, automated methods have been proposed and used with success. Bataille et. al. examine and design for physical constraints, fairness, and efficiency for different agents using a shared resource (an earth observing satellite).[14] Bresina et. al. combine together two techniques - GenH which generates a specialized search heuristic, and HBSS which employs the heuristic within a stochastic sampling method - to automatically generate high-quality schedules with respect to an objective function.[15]

Our work complements existing research by providing a mechanism to optimize over cyber-physical resources as well as mission objectives for the holistic system while leveraging optimization theory, and more particularly, optimal control. Existing solutions to dynamically adjust parameters can provide the tools by which a system could use our methodology to produce more efficient missions according to the individual metrics chosen in the cost function.

III.   Cost Functions

We desire to optimize over both physical and cyber characteristics to improve efficiency and performance for our designated mission. We do this by developing *cost* terms for physical system energy, cyber utilization, as well as mission-critical task performance. This means developing cost terms for the energy consumed by the fan actuators as function of desired angular velocity, a term representing the energy consumed by the computing system, and a metric measuring the amount of information we gather from the on-board camera.

For the physical system, we wish to determine the optimal angular velocity of TableSat for our proposed mission. Because the on-board camera is fixed and rotating with TableSat, and since we only allow for rotation in one direction, we need only consider simplified motion about a single axis. For the cyber system, our objective is to determine the optimal execution rate of a mission-specific task. This task is a real-time surveillance task in which we wish to collect data on orange colored segments of construction paper. A single complete job of this surveillance task consists of acquisition, processing, and storage of an image. We assume that other system-critical cyber tasks, including the control task, have been allocated a fixed amount of resources. Our surveillance task therefore operates within the remaining non-critical bandwidth available in the cyber system.

We build our cost function from the ground-up, integrating physical and cyber cost terms and associated functions. "Physical" in the context of a TableSat includes items related to rotation of the system, for example, the rotating table, fan actuators, and sensors. "Cyber" relates to items required for image collection, data processing, computation of control inputs, etc.

A.   Physical System Terms

To balance the goals of the physical system we seek to minimize total energy consumption by the fan actuators while also minimizing time required for mission completion.

1.   Physical System Energy

In TableSat all the energy required to rotate the table is consumed by the fan actuators. The metric then for physical system energy becomes

$$E_p = \int P(\omega) \, dt. \tag{1}$$

Where P is power as a function of the angular velocity, ω. Because we use simple computer fans controlled by a Pulse Width Modulation (PWM) signal we assume power consumed is simply

$$P(V) = IV \tag{2}$$

where I is the constant current draw (a parameter of the fan), and V the time averaged voltage "seen" by the fan via the PWM duty cycle. As voltage is a function of PWM signal which is our control input, we can

American Institute of Aeronautics and Astronautics

rewrite Equation 2 as

$$P(d) = IV_{\max} \frac{d - d_{\min}}{d_{\max} - d_{\min}}$$

where d is the PWM duty cycle and $d_{\min}$ and $d_{\max}$ are the constant minimum and maximum duty cycles respectively as experimentally determined. $V_{\max}$ is a constant parameter of the fan and is the maximum voltage the fan can accept.

Because we require a power function that takes angular velocity as a parameter for optimization we conducted a series of experiments commanding a PWM duty cycle, awaiting approximately steady state angular velocity, and subsequently timing the rotational speed. We used MATLAB's `cftool` to determine a linear curve fit as seen in Figure 2. We note that higher order polynomials would provide a better fit. We explicitly chose a linear fit due to the uncertainty surrounding our friction coefficients between experiments.
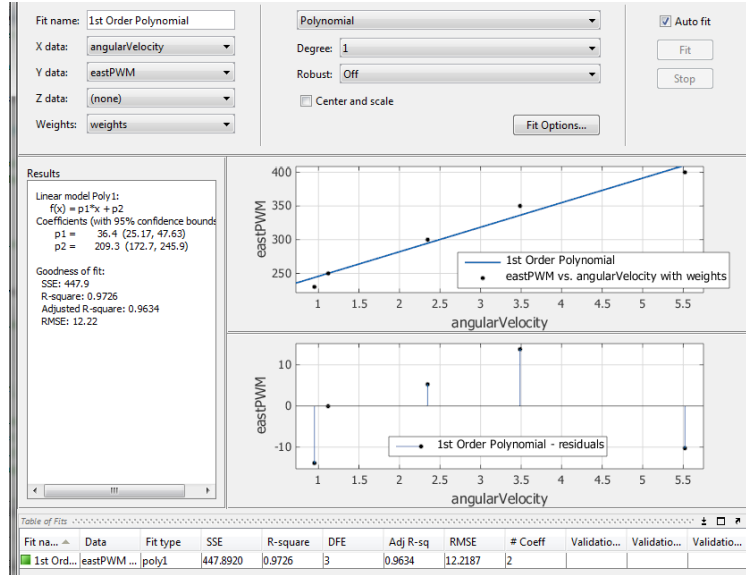

Figure 2. Linear Curve Fit Using MATLAB's `cftool`

Functionally, we have power as a function of angular velocity given by

$$P(\omega) = IV_{\max}\left(\frac{p_1\omega + p_2 - d_{\min}}{d_{\max} - d_{\min}}\right) \tag{3}$$

Using the coefficients experimentally determined we can reduce Equation 3 to

$$P(\omega) = 0.514\omega - 0.292 \tag{4}$$

The experimentally determined power curve and associated energy curve can be seen in Figure 3. We note the flattening out of the energy curve as angular velocity increases. This suggests that at some point it does not cost us significantly more to rotate much faster. This observation is seen later on in the final cost function.

2. Time

In addition to minimizing power consumption, we would also like to minimize the amount of time required to accomplish our mission. Such time-minimal optimization cost terms are common, and are simply given by
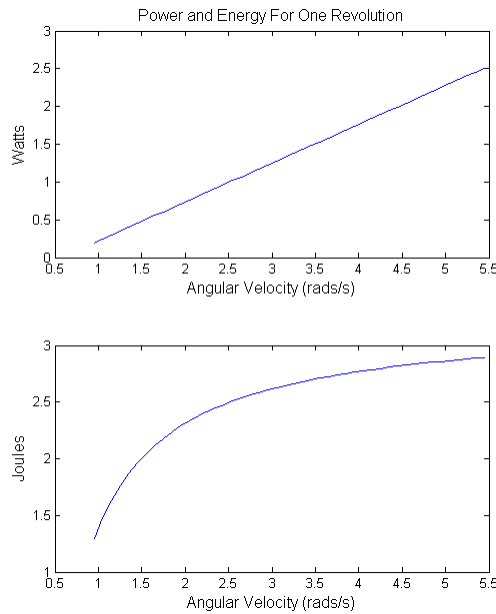
$$T = \int dt. \tag{5}$$

American Institute of Aeronautics and Astronautics

Figure 3. Power and Energy Curves for TableSat Physical System

3. Cost Function for Physical System

The two competing cost metrics, E_p and T, comprise the cost function for the physical system

$$J_p(\omega) = \beta_{p1} \int P(\omega)\, dt + \beta_{p2} \int dt \tag{6}$$

where β_{p1}, β_{p2} are weighting terms. Optimizing J_p alone is what a traditional trajectory or path planner would do if no costs are attributed to the cyber system.

B. Cyber System Terms

For a modern autonomous system the cyber portion becomes the center point for data collection, actuation inputs, communication, I/O, path planning, control input calculation, etc. In many real-time systems task execution rates are determined *a priori* based on requirements of the system and/or mission. For example, the sampling rate of the control task may be selected based on digital control analysis. In this work we do not interfere with such high-priority task assignments and focus instead on the tuning of a lower-priority mission-critical task giving us the ability to optimize over mission and cyber parameters without interfering with safety-critical tasks. We assume that we can not only conserve energy by optimally selecting execution rates of lower-priority tasks, but we can also increase mission effectiveness by developing costs that relate task execution rates to mission efficiency.

1. Cyber Utilization

In real-time system scheduling theory online schedules can be created by examining relative deadlines of independent periodic tasks as in the Earliest Deadline First (EDF) scheduling algorithm. Such optimal scheduling algorithms are dynamic by assigning task priority as jobs are released to the operating system for scheduling.[16] They therefore have the ability to respond to changing deadlines and periodic rates.

In this paper, we assume that at least part of the cyber utilization is fixed based on selected and scheduled periodicity of mission-critical tasks, and instead focus on maximizing use of the remaining resources. To that end, we assume a single task, τ, achieves the important mission goal of capturing and processing an image of our orange and black cylindrical backdrop (see Figure 1). The task runs at execution rate r_τ Hz and has a maximum execution rate of $r_{\tau,max}$ Hz stemming from restrictions based on available cyber resources. That is, we ensure schedulability of the task based on $r_{\tau,\max}$ but allow that period to change resulting in freed

cyber resources to be devoted to increased service of other tasks or simply to conserve energy. We introduce the cyber utilization term

$$U_\tau = \frac{r_\tau}{r_{\tau,max}}. \tag{7}$$

We note that r_τ is the rate of execution of task τ throughout the mission. We assume that cyber utilization is proportional to energy consumed by the cyber system, and as a result, minimizing it is the cyber analog to the energy minimization term of the physical system in Equation (1).

2. Mission Information

In developing our cost metrics, we seek to relate mission efficiency to cyber and physical parameters. Our specific mission objective is to aquire images of orange squares on a poster board backdrop. We can consider this mission equivalent to collecting as many orange pixels as possible. To appropriately model the collection of orange pixels we make the simplifying assumption that the orange squares of the backdrop are laid out continuously (as opposed to an alternating pattern with black) covering θ_{orange} radians of the rotation. We then find the number of frames F collected during the interval which is a function of both angular velocity and cyber task execution rate

$$F = \frac{r_\tau \theta_{\text{orange}}}{\omega}. \tag{8}$$

We note that this is a continuous function and does not truly represent the discrete number of frames taken. However, this approximation allows us to use existing mathematical tools to solve the problem. We then determine the amount of sweep made by the camera in F frames and including the camera's footprint γ in radians

$$\theta_F = \gamma + \theta_{\text{orange}}. \tag{9}$$

We assume that the camera can **only** see the colored backdrop and that all pixels will be counted as either orange or black. We can then combine Equations (8) and (9) to determine the number of orange pixels per radian seen by the camera which we call mission "information"

$$I(\omega, r_\tau) = \frac{FP\theta_{\text{orange}}}{2\pi\theta_F} \tag{10}$$

where P is the total number of pixels in an image (for a 640×480 camera this is 307,200 pixels).

Naturally, this metric as we have described it, should be maximized to produce the greatest mission success. Traditionally, we would simply minimize $-\int I(\omega, r_\tau)\, dt$ for the equivalent. This results in the metric shown in Figure (4). Such a metric becomes problematic in the total system cost function because it is concave rather than convex. This in turn means the final cost function has a saddle point and falls off sharply toward the boundaries. To be able to solve a well-behaved convex shaped optimization problem we need a mission success metric that, when minimized, produces maximum mission success. Therefore we propose

$$M = \int \frac{1}{I(\omega, r_\tau)} dt \tag{11}$$

which can be seen in Figure (5). We note the nonlinear dependence of M on both angular velocity, and the cyber rate. We also note that the dependence on cyber rate falls off as a steep exponential, and falls off more gradually with aircraft velocity. This means we expect optimal solutions to achieve higher mission success by increasing cyber rate than by going slower.

3. Cost Function for Cyber System

The expressions in Equations (7) and (11) together comprise the cost function for the cyber system

$$J_c(\omega, r_\tau) = \beta_{c1}\frac{r_\tau}{r_{\tau,max}} + \beta_{c2}\int \frac{1}{I(\omega, r_\tau)} \tag{12}$$

where we have weighting terms β_{c1}, and β_{c2}. We might independently optimize over such a cost function if we were interested solely in trading cyber resource utilization cost against reward from accomplishing mission objectives.
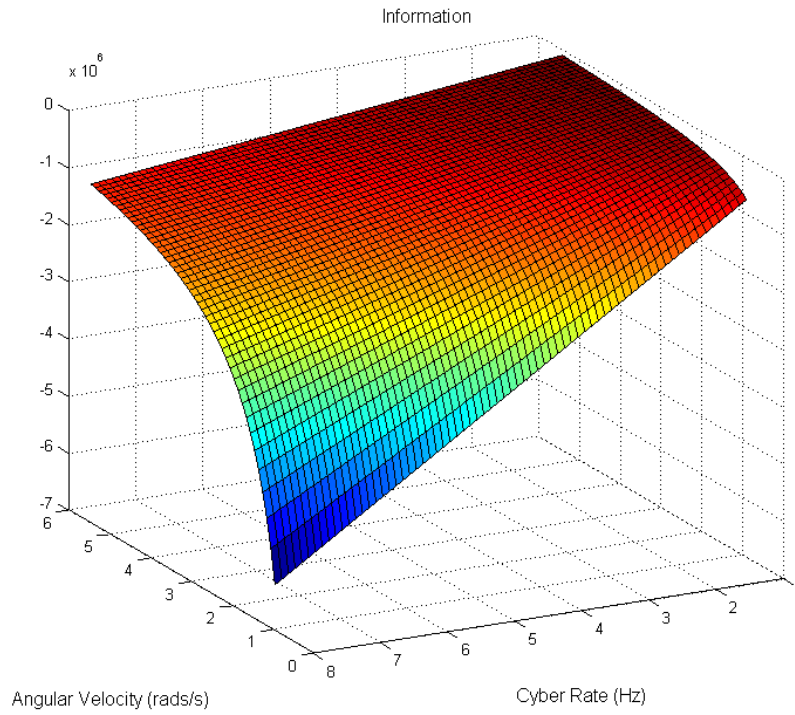
American Institute of Aeronautics and Astronautics
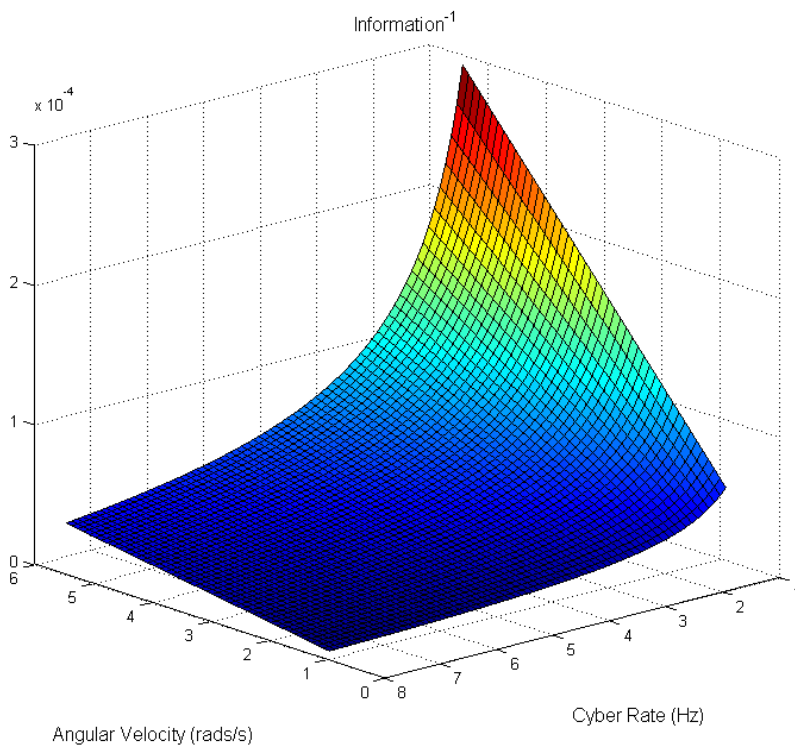
Figure 4. $-I\left(\omega, r_{\tau}\right)$

Figure 5. Information Metric

American Institute of Aeronautics and Astronautics

C. CPS Cost Function

We desire, however, a holistic cost function that gives us optimal values of ω, and r_τ for the total system including mission objectives. We therefore combine J_p and J_c to obtain a CPS cost function

$$J\left(\omega, r_\tau\right) = \beta_{p1} \int P\left(\omega\right) dt + \beta_{p2} \int dt + \beta_{c1} \frac{r_\tau}{r_{\tau,max}} + \beta_{c2} \int \frac{1}{I\left(\omega, r_\tau\right)} dt \tag{13}$$

In Section V we will manipulate the weights to compare physical-only optimization, cyber-only optimization, and total system optimization, to illustrate the efficacy of our co-design methodology.

1. Simplified Cost Function

Although Equation (13) could be solved for the optimal trajectories for ω, and r_τ, as in our previous work[1] we limit our solution to finding the static ω, and r_τ that minimize the cost function. If we know the total "distance" D of the mission (for TableSat we use the rotational distance of 2π radians), we can replace the integrals in (13) with more straightforward sums, simplifying the equation to

$$J\left(\omega, r_\tau\right) = \beta_{p1} \frac{DP\left(\omega\right)}{\omega} + \beta_{p2} \frac{D}{\omega} + \beta_{c1} \frac{r_\tau}{r_{\tau,max}} + \beta_{c2} \frac{1}{DI\left(\omega, r_\tau\right)}. \tag{14}$$

Note that this function is convex with an unconstrained minimum, so finding a solution is straightforward.

Choosing appropriate weights is a difficult part of the design process, and is made much harder when different cost metrics take on different ranges of values. If we normalize the terms, the weights we choose will make intuitive sense and can be meaningfully compared between terms. Using the maximum possible values for each term, we are able to normalize Equation (14) to

$$J\left(\omega, r_\tau\right) = \beta_{p1} \frac{DP\left(\omega\right)}{\omega \max\left\{E_p\right\}} + \beta_{p2} \frac{D}{\omega \max\left\{T\right\}} + \beta_{c1} \frac{r_\tau}{r_{\tau,max}} + \beta_{c2} \frac{1}{DI\left(\omega, r_\tau\right) \max\left\{M\right\}} \tag{15}$$

where $\max\left\{E_P\right\}$ is found by assuming the maximum angular velocity for the mission, $\max\left\{T\right\}$ is found from the slowest angular velocity, and $\max\left\{M\right\}$ from the fastest angular velocity and slowest cyber rate. We solve the problem with numerical methods, which we describe more in Section A.

2. Optimization Problem

This leads to the constrained optimization problem we wish to solve

$$\text{Minimize} \quad J\left(\omega, r_\tau\right) \tag{16a}$$

$$\text{subject to} \quad \omega \leq \omega_{\max} \tag{16b}$$

$$\omega \geq \omega_{\min} \tag{16c}$$

$$r_\tau \leq r_{\tau,\max} \tag{16d}$$

$$r_\tau \geq r_{\tau,\min} \tag{16e}$$

where $\omega_{\min}$, $\omega_{\max}$, $r_{\tau,\min}$, and $r_{\tau,\max}$ are determined based on limitations of the physical system along with diminishing returns of success, and are

$$0.95\,\mathrm{rads/s} \leq \omega \leq 5.52\,\mathrm{rads/s}$$

$$1\,\mathrm{Hz} \leq r_\tau \leq 8\,\mathrm{Hz}$$

IV. Experiment Setup

Our objective is to survey orange segments of an orange and black paper wall surrounding a rotating table-top satellite, TableSat, using a fixed camera mounted at the center. The surveying mission requires two configurable settings, angular velocity and task rate. Below is a description of the hardware and software components of our application, as well as the assumptions and model parameters that arose from the components.

A. TableSat Overview

TableSat is a table-top satellite with one degree of rotational freedom. It has a base with a central post that comes to a conical point. The board itself balances on the post via a screw located at the middle of the board. Because the contact is reduced to that single point, TableSat has a very low coefficient of friction. It rotates around the base using two standard computer cooling fans which take PWM signals to dictate fan rotations per minute (RPMs). The computer-on-module (COM) is a Gumstix Overo with a Tobi expansion board that provides GPIO, such as that used for PWM output. TableSat also has a camera mounted at the center that connects to the COM through the USB port.

B. Hardware

1. Gumstix

The TableSat COM is a Gumstix Overo with a 1GHz ARMv7 processor and 512 MB DDR RAM. The Gumstix runs Linux 2.6.36, which is not a real-time operating system (RTOS). That means that because our software timers are run at the user-level, as opposed to the kernel-level, the timers may be limited in their precision. The slow speed of the processor also means that the maximum frame capture and process rate is 8 frames per second (FPS). Finding an efficient means of employing this limited resource is therefore crucial.

2. Fans

The actuators are two Sunon KD1209PTB2 computer cooling fans with a max rated speed of 2600 RPM. The operating voltage ranges between 5 and 13.8 volts. They are rated to draw 0.2 amps of direct current and 2.4 watts of power.

The fan speed is set by sending a PWM signal over the control line. The characteristics of the PWM signal is specified in code by sending a byte sequence to the control line. We needed to discover which bytes corresponded to the minimum and maximum PWM signals, and ensure that the range of intermediate byte values mapped linearly to the signal. We connected the PWM line to an oscilloscope and determined what bytes corresponded to which PWM signals. In particular, we figured out what bytes corresponded to max speed. In this way, we could tell what fraction of max fan speed we were signaling. By running a series of timed experiments and fitting a curve to the resulting data, we were able to map PWM signal to angular velocity. As described in 1, to model the energy usage of the fans, we assumed that the power draw is proportional to the average amount of time the PWM signal is up.

3. Camera

The webcam is a Logitech QuickCam Pro 9000 that connects to the board via USB. It has a max frame rate of 30 FPS and a resolution of 640×480. The camera also has an automatic brightness/contrast adjustment feature that must be considered and accounted for when attempting computer vision tasks.

C. Software

1. Frame Capture

Camera access and frame processing was done using OpenCV libraries. We wrote a function that captured a 640×480 frame from the webcam, converted it to a HSV (Hue-Saturation-Value) colorspace and then thresholded it on orange. Thresholding produces a black and white image where white pixels indicate the presence of orange on the original image. Mathematically, the resulting image has $640 \times 480 = 307,200$ pixels, each of which takes on a value, v, of either 255 or 0, where 255 indicates orange detection. The total number of orange pixels in the original image, N, is thus given by

$$N = \frac{\sum_{i=1}^{307,200} v_i}{255}$$

Once the camera is initialized, this capture-and-process task can be called whenever desired.

2. Cyclic Executive

Our program takes two parameters: angular velocity and the task rate for our periodic task. Since we only consider the static case in this project, we set the fan speed in the beginning that corresponds to the desired angular velocity, and the only periodic task is then capturing frames from the camera.

After the PWM signal is initialized, we wait three minutes for the TableSat to spin up to steady state angular velocity. Then, when the experimenter presses <ctrl>-c, the program starts capturing frames at the specified frame rate. This is done using a cyclic executive that consists of one task, the frame capturing/pixel counting task. When the experimenter presses <ctrl>-c again after the desired number of revolutions, the system stops and some metrics are displayed - the number of white pixels seen, the number of total pixels captured, and the amount of time that the cyclic executive executed for.

D. Assumptions

We assume that friction is constant, or at least consistent. This is not necessarily true, because even when the TableSat is perfectly balanced at rest, as it rotates it begins to precess due to the heterogeneous mounted components applying non-uniform forces to the board. As the TableSat precesses, it changes position on its base, which changes the coefficient of friction over time.

To properly maintain a constant speed, we would need to adjust the fan signal in response to the changing friction. However, in our experiments, we send a constant PWM signal and assume that after some time the TableSat will reach a steady state speed. As noted in 2, we mapped PWM signals to steady state angular velocities using the results from a set of timed trials. Assuming constant (or consistent) friction is necessary for the assumption that the same PWM signal will always result in the same steady-state velocity. Since we are using open-loop control, this assumption is crucial.

V.   Results

We first investigated the simulated impact and tradeoffs between objectives from both the cyber and physical systems with the goal of minimizing energy use and time while maximizing mission success. We hope to demonstrate that consideration given to both physical, cyber, and mission objectives can yield more well-rounded, efficient results. We first examine and analyze simulated results obtained by optimizing the cost function in Equation (15) to gain insight into the tradeoffs from competing objectives. We optimize over the physical system alone, the cyber system alone, and both together by selecting appropriate weights β_{p1}, β_{p2}, β_{c1}, and β_{c2}. Then we compare real-world performance of the three different optimization techniques as indicated by TableSat mission data.

A. Simulated Results

We solve the optimization problem in Equation (16) using numerical methods. We use MATLAB's solver `fmincon` and setting lower and upper bounds on the design parameters. This MATLAB function uses active-set optimization which utilizes sequential quadratic programming (SQP) and estimates the Hessian of the Lagrangian using the well-known BFGS algorithm.[17]

Often there are auxiliary reasons for favoring one cost term over another such as length of time since the last mission, or a cloudy day resulting in poorer image quality. Since we wish to investigate the comparison of holistic CPS optimization with independent physical and cyber system optimization we allow corresponding weights to go to zero as indicated in the 1st and 2nd rows of Table 1. In the 3rd row we use some *a priori* intuition to select weights that focus slightly more on time and less on cyber utilization. The analytical results of our optimization scheme and simulation are in Table 1. These results suggest that the lowest cost solution comes by focusing on cyber parameters. This indicates there may be a problem with the weights that we are applying to the cost function. We note the specific weighting we used that slightly favors time over cyber utilization, which means we won't collect as many orange pixels. Despite these analytical results, in the next section we show experimental results that indeed validate this design methodology.

B. Experimental Results

We present in Table 2 the results from our experiments with TableSat. As we had hoped, the lowest cost solution was indeed the total system optimization that accounts for cyber, physical, and mission objectives.

American Institute of Aeronautics and Astronautics

Table 1. Comparison of Simulated Results

Parameters	Solution Type	E_p	T	U_τ	M	Total	% Worse than Lowest Cost Soln.
$\omega = 5.2\,\mathrm{rads/s}$ $r_\tau = 2\,\mathrm{Hz}$	J with $\beta_{p1} = \beta_{p2} = 0.5$ $\beta_{c1} = \beta_{c2} = 0.0$	$2.87\,\mathrm{J}$	$1.21\,\mathrm{s}$	0.25	2.21×10^{-5}	1.0963	100.8%
$\omega = 0.95\,\mathrm{rads/s}$ $r_\tau = 3\,\mathrm{Hz}$	J with $\beta_{p1} = \beta_{p2} = 0.0$ $\beta_{c1} = \beta_{c2} = 0.5$	$1.30\,\mathrm{J}$	$6.61\,\mathrm{s}$	0.38	2.70×10^{-6}	0.5457	N/A
$\omega = 2.42\,\mathrm{rads/s}$ $r_\tau = 2\,\mathrm{Hz}$	J with $\beta_{p1} = 1$, $\beta_{p2} = 1.2$ $\beta_{c1} = 0.8$, $\beta_{c2} = 1$	$2.47\,\mathrm{J}$	$2.60\,\mathrm{s}$	0.25	1.03×10^{-6}	0.7179	31.55%

Table 2. Comparison of Experimental Results

Parameters	Solution Type	E_p	T	U_τ	M	Total	% Worse than Lowest Cost Soln.
$\omega = 5.2\,\mathrm{rads/s}$ $r_\tau = 2\,\mathrm{Hz}$	J with $\beta_{p1} = \beta_{p2} = 0.5$ $\beta_{c1} = \beta_{c2} = 0.0$	$2.88\,\mathrm{J}$	$1.21\,\mathrm{s}$	0.25	2.75×10^{-6}	0.4483	9.09%
$\omega = 0.95\,\mathrm{rads/s}$ $r_\tau = 3\,\mathrm{Hz}$	J with $\beta_{p1} = \beta_{p2} = 0.0$ $\beta_{c1} = \beta_{c2} = 0.5$	$1.29\,\mathrm{J}$	$6.61\,\mathrm{s}$	0.38	2.36×10^{-7}	0.4635	12.79%
$\omega = 2.42\,\mathrm{rads/s}$ $r_\tau = 2\,\mathrm{Hz}$	J with $\beta_{p1} = 1$, $\beta_{p2} = 1.2$ $\beta_{c1} = 0.8$, $\beta_{c2} = 1$	$2.39\,\mathrm{J}$	$2.52\,\mathrm{s}$	0.25	1.392×10^{-6}	0.4109	N/A

VI.   Conclusions and Future Work

The impact of the digital revolution will continue to reverberate across many fields for some time. As cyber and physical systems become more tightly integrated, the need for strong multidisciplinary codesign techniques becomes increasingly urgent. In this paper, we've implemented one such proposed optimization scheme - a cost function combining physical and cyber terms - on real hardware. We used a TableSat with a mounted camera to undertake a mission to survey orange segments of an orange and black paper landscape. We experimentally determined system-specific parameters for our equations and used the resulting optimization problem to produce an optimal angular velocity and task rate. We were then able to use these results to configure a real mission, and compare real-world performance between the traditional optimization method and the cost-function method.

Our results showed that our multi-disciplinary approach indeed resulted in better efficiency than optimizing the physical terms without considering the cyber term (or optimizing the cyber term without considering the physical term). This supports the theoretical results found by previous work.

However, these results have limited statistical significance, as they are results from only one run for each configuration. The TableSat platform was very noisy and had extremely high variance in steady state speed. We tried to calibrate it before every run such that the behavior roughly matched the experimental results we obtained earlier when mapping PWM signals to angular velocity, but this was time-consuming and statistically unsatisfying. Another result of this is that we were not able to do a parameter sweep of cost-function weights. Our information metric is also not very sophisticated and may not be representative of all the complexities present in many real applications.

This work highlighted many of the difficulties associated with working with physical systems, one of which is that some physical phenomena are extremely sensitive to initial conditions. At a macro-level this looks like inconsistency (e.g. varying steady-state speeds). To be able to use simple equations that cannot account for all possible disturbances, the system must display consistent, stable states at whatever scale the system is modeled. Then, control theory is applied to reject (most) disturbances usually via feedback. For the scope of this project, however, we used open-loop control, which made data collection difficult. For those reasons, further work on this particular application would require writing a closed-loop controller for the TableSat platform. This would also allow us to do a broad parameter sweep of our cost function.

Though further work in this area presents challenges, the effort may be well worth it. As cyber-physical

systems become a more and more ubiquitous presence in our lives, we must embrace the integration of previously disparate paradigms and begin working towards a similarly integrated multidisciplinary approach.

References

[1] J. M. Bradley and E. M. Atkins, "Multi-Disciplinary Cyber-Physical Optimization for Unmanned Aircraft Systems," in *Infotech@Aerospace.* AIAA, June 2012.

[2] ——, "Cyber-physical optimization for unmanned aircraft systems," *Journal of Aerospace Information Systems*, vol. To Appear, 2013.

[3] M. E. F. Vess, "System Modeling and Controller Design for a Single Degree of Freedom Spacecraft Simulator," Master's thesis, University of Maryland, May 2005.

[4] D. Fontanelli, L. Greco, and A. Bicchi, "Anytime Control Algorithms for Embedded Real-time Systems," *Hybrid Systems: computation and control*, pp. 158–171, 2008.

[5] V. Gupta, "On an anytime algorithm for control," in *Proceedings of the 48th IEEE Conference on Decision and Control.* IEEE, 2010, pp. 6218–6223.

[6] R. Bhattacharya and G. Balas, "Anytime control algorithm: Model reduction approach," *Journal of Guidance, Control, & Dynamics*, vol. 27, no. 5, 2004.

[7] K. Arzén, A. Cervin, J. Eker, and L. Sha, "An introduction to control and scheduling co-design," in *Proc. of the 39th IEEE Conf. on Decision and Control*, vol. 5. IEEE, 2002, pp. 4865–4870.

[8] L. Sha, T. Abdelzaher, K. Årzén, A. Cervin, T. Baker, A. Burns, G. Buttazzo, M. Caccamo, J. Lehoczky, and A. Mok, "Real-time Scheduling Theory: A Historical Perspective," *Real-time Systems*, vol. 28, no. 2, pp. 101–155, 2004.

[9] J. Eker, P. Hagander, and K. Årzén, "A feedback scheduler for real-time control tasks," *Control Engineering Practice*, vol. 8, no. 12, pp. 1369–1378, 2000.

[10] M. Agrawal, D. Cofer, and T. Samad, "Real-time adaptive resource management for advanced avionics," *Control Systems, IEEE*, vol. 23, no. 1, pp. 76 – 86, feb 2003.

[11] M. Russ and P. Stutz, "Airborne sensor and perception management: A conceptual approach for surveillance uas," in *Information Fusion (FUSION), 2012 15th International Conference on*, july 2012, pp. 2444 –2451.

[12] P. Narayan, D. Campbell, and R. Walker, "Computationally adaptive multi-objective trajectory optimization for uas with variable planning deadlines," in *Aerospace conference, 2009 IEEE*, march 2009, pp. 1 –8.

[13] S. Harrison, M. Price, and M. Philpott, "Task scheduling for satellite based imagery," in *Proceedings of the Eighteenth Workshop of the UK Planning and Scheduling Special Interest Group*, vol. 78. University of Salford, UK, 1999, pp. 64–78.

[14] N. Bataille, M. Lemaitre, and G. Verfaillie, "Efficiency and fairness when sharing the use of a satellite," in *Artificial Intelligence, Robotics and Automation in Space*, vol. 440, 1999, p. 465.

[15] J. L. Bresina, R. A. Morris, and E. W. R., "Optimising Observation Scheduling Objectives," in *Proceedings of the 1997 NASA Workshop on Planning and Scheduling for Space*, 1997.

[16] C. Krishna, *Real-Time Systems.* Wiley Online Library, 1999.

[17] D. Liu and J. Nocedal, "On the limited memory bfgs method for large scale optimization," *Mathematical programming*, vol. 45, no. 1, pp. 503–528, 1989.