APPROVAL SHEET

Title of Thesis: A FRAMEWORK FOR ANALYZING THE IMPACT OF ACTUATION LIMITS ON CYBER-PHYSICAL SYSTEMS

Name of Candidate: Scott Bohon
Master of Science, 2020

Thesis and Abstract Approved: 

Ryan Robucci
Associate Professor
Department of Computer Science
and Electrical Engineering

Date Approved: July 31, 2020

ABSTRACT

Title of thesis: A FRAMEWORK FOR ANALYZING THE IMPACT OF ACTUATION LIMITS ON CYBER-PHYSICAL SYSTEMS

Scott Bohon, Master of Science, 2020

Thesis directed by: Ryan Robucci, Associate Professor
Department of Computer Science
and Electrical Engineering

Cyber-physical systems (CPS) are smart systems of networked computing and physical components. CPS are ubiquitous in industrial and consumer applications, ranging from control systems in smart power grids to phone touch screens. Unfortunately, the cyber component of CPS may introduce attack vectors by which a bad actor can cause harm to the physical system. A famous example includes the Stuxnet computer worm which inflicted physical damage to Iranian nuclear centrifuges. A cyber-physical mitigation strategy against CPS attacks is actuation limits. Actuation limits are constraints intentionally imposed on the actuators of a CPS to mitigate actuation behaviors which lead to dangerous states. Actuation limits, while able to constrain an attacker, may introduce performance penalties. In this research, a framework is presented which scores actuation limit schemes on their attack resilience and performance integrity. Attack resilience is measured by subjecting the CPS to a battery of cyber-physical attacks and observing if actuation limits were successful in mitigation. Performance integrity is measured by compar-

ing the performance of the CPS with and without actuation limits. An algorithm to combine actuation limit schemes to yield an amalgam scheme with improved scores is presented. Actuation limits for a simulated ship autopilot are scored to demonstrate the utility of the framework. Low scores were observed in two general cases. Overly constraining actuation limits scored poorly in attack resilience and performance integrity as the required operating behaviors were compromised by the limits. Overly broad actuation limits preserved performance integrity yet did not sufficiently constrain an attacker and exhibited poor attack resilience. Amalgam schemes demonstrated high scores overall by only constraining the CPS in high risk states. The results of the research indicate the proposed framework can be a useful tool in evaluating the effectiveness of actuation limits as an attack mitigation strategy in CPS.

A FRAMEWORK FOR ANALYZING THE IMPACT OF ACTUATION LIMITS ON CYBER-PHYSICAL SYSTEMS

by

Scott Bohon

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, Baltimore County in partial fulfillment
of the requirements for the degree of
Master of Science
2020

Advisory Committee:
Professor Ryan Robucci, Chair/Advisor
Professor Chintan Patel
Professor Dhananjay Phatak

Acknowledgments

I would like to thank my advisor Dr. Robucci for his knowledge, assistance, and overarching guidance throughout this research. I also thank Brien Croteau for being a mentor throughout the research process. Brien provided knowledge and insight regarding control systems, cyber-physical security, actuation limits, ship motion, and inspiration for the algorithm to combine actuation limits. Brien details many of these insights in his dissertation [1]. Finally, I want to thank Chris Boia, Jacob Kang, and Aksel Thomas for their work in designing and constructing the cyber-physical system ship testbed.

Table of Contents

List of Figures

List of Abbreviations

U	surge speed
d	rudder angle
r	reference or nominal signal
u	controller output
u_L	limited controller output
u_a	actuator output
y	plant output
$\mathbf{x}$	state vector
θ_{OS}	own ship heading angle
ϕ	yaw rate

CPA	closest point of approach
CPS	cyber-physical systems
LTI	linear time invariant
IDS	intrusion detection system
IRS	intrusion response system
NIST	National Institute for Standards in Technology
NOAA	National Oceanic and Atmospheric Administration

Chapter 1: Introduction

This chapter aims to introduce cyber-physical systems and their security, the concept of actuation limits and their use in securing CPS, the contributions of this research, and an overview of the methodology and results of the research.

1.1 Design of Physically Safe CPS in the Presence of Adversaries

The National Institute for Standards in Technology (NIST) defines cyber-physical systems (CPS) to be "smart systems that include engineered interacting networks of physical and computational components" [2]. CPS have become increasingly important as public utilities, health care, agriculture, manufacturing, and a myriad of other industries [2]. One of the goals of CPS design is physically safe operation [3]. Fault tolerance strategies play a major role in ensuring the safe physical operation of CPS. Fault tolerant control strategies, means to continue safe performance in the presence of failure, are well developed [4] and widely implemented in CPS. Unfortunately, attacks conducted against CPS have demonstrated that fault tolerant control strategies alone cannot ensure physically safe operation. Attackers bypass fault tolerant design using cyber or cyber-physical attack techniques. The success of attacks on Iranian nuclear centrifuges [5], the Ukrainian power grid [6],

and commercial aircraft [7] highlight the capabilities of dedicated adversaries. Preventing attacks against CPS requires solutions beyond fault tolerance in both the cyber and cyber-physical domain. Traditional cybersecurity solutions can be used to bolster CPS against cyber attack by providing confidentiality, integrity, and authentication properties to communications and cyber components. Cryptographically secure communication protocols can provide confidentiality and authenticity for network communications within the CPS [8]. Device attestation is a technique to authenticate the integrity of software running on CPS components [9,10]. CPS security also requires cyber-physical security solutions, solutions which protect against attackers with knowledge of the control system dynamics of the CPS. Reactive cyber-physical security solutions, solutions with the primary purpose of detecting cyber-physical attacks, include intrusion detection systems [11] and health monitoring systems [12]. Preventative cyber-physical solutions, solutions which deter or pre-emptively mitigate the effects cyber-physical attacks, include authentication via signal watermarking [13,14], reachability analysis [15], and actuation limits [16]. This research focuses on quantitative analysis of the cyber-physical mitigation strategy of actuation limits in a simulated CPS.

1.2 Actuation Limits

Cyber-physical attackers have been shown to be able to use their knowledge of the CPS dynamics to evade detection by IDS and drive the CPS to dangerous states [14]. Actuation limits, bounds on state transitions enacted by actuators, can

be used to constrain intelligent cyber-physical attackers with intimate knowledge of the CPS dynamics [16]. Artificial actuation limits, limits more constraining than the natural physical limits of the actuator, restrict the reachable set of states to contain fewer dangerous states. Restricting the actuator in this manner makes driving the CPS to a dangerous state more difficult for an attacker. The imposition of actuation limits may also restrict the normal operation of the CPS. Bounding the actuator capabilities restricts the number of operation states the system can reach [16]. The effect of actuation limits on system performance depends on the amount of restriction placed on the system's operation states. This research aims to provide a framework to evaluate the effectiveness of actuation limits on constraining attackers and the loss of performance suffered as a consequence of the restriction of operation states.

1.3 Contributions

This research makes the following contributions.

1. The definition of three metrics, attack resilience, performance integrity, and bandwidth utilization, to evaluate the effects of imposing actuation limits on a CPS.

2. The evaluation of actuation limit schemes using the aforementioned metrics to determine their performance in securing a simulated CPS.

3. The demonstration of an algorithm to combine actuation limit schemes into an amalgam scheme with the intent of improving attack resilience in danger-

ous states while also providing satisfactory performance integrity and efficient bandwidth utilization.

1.4 Experimental Methodology

The evaluation of actuation limit schemes is conducted on a simulated CPS. The simulated CPS is a ship autopilot piloting a Mariner vessel in static obstacle maps generated from real world coastline satellite imagery. The CPS is assumed to have intrusion detection and response capabilities to detect attacks and restore legitimate control within a finite time window. Time dependent actuation limits with various update rates and degrees of actuation restrictiveness are evaluated and compared according to the proposed metrics. Amalgam actuation limit schemes are generated to demonstrate how scored actuation limit schemes can be combined to provide attack resilience while keeping restriction of the actuators and bandwidth use low.

1.5 Experimental Results

Analysis of the results of the experiment indicate the proposed metrics are useful in evaluating the impact of actuation limits on the simulated CPS. Three observations are made regarding the actuation limit schemes evaluated and analyzed. Actuation limit schemes overly constraining on the required operation states of the CPS scored low on attack resilience and performance integrity. Actuation limit schemes allowing excessive extraneous operational states maintained performance

integrity but suffered from poor attack resilience. Actuation limit schemes which provided the tightest bounds without impinging on required operational states performed the best on the attack resilience and performance integrity scores. The tight actuation limit schemes often incurred relatively high bandwidth overhead. The amalgam actuation limit scheme demonstrates a solution to this bandwidth consumption by only applying tight limits in risky states while using low bandwidth loose actuation limits in low risk states.

Chapter 2: Related Work

This chapter includes a discussion of the literature in regards to a general adversarial model against CPS, CPS vulnerabilities in the cyber and cyber-physical domain, and attack detection and mitigation strategies.

2.1 Cyber Vulnerabilities and Attack Vectors

Malicious actors have a variety of motivations for attacking CPS. Motivations include criminal financial gain, espionage, cyberwar, amongst others [17]. These attackers exploit one or more vulnerabilities via an attack vector to realize their attacks. CPS have been demonstrated to be vulnerable to a wide variety of vulnerabilities and attack vectors due to their cyber and physical nature [17]. Cyber vulnerabilities in CPS include flaws in operating systems and controller software [5] as well as network communications and protocols [18]. The consequences of exploiting these vulnerabilities include compromising one or more of the CPS's properties of confidentiality, integrity, availability, privacy, and safety [17].

2.2 Cyber-Physical Attack Model

A general attack model against CPS is presented in [19]. This attack model outlines three broad types of attacks against CPS, deception, denial of service (DoS), and direct physical attacks. Figure 2.1 is an illustration of the generalized attack model against CPS. Deception attacks inject false controller or sensor data into the system. Denial of service attacks prevent the communication of controller or sensor information within the CPS. Direct physical attacks are manipulations of the actuators or plant. Each attack type presents different challenges and demands different solutions.

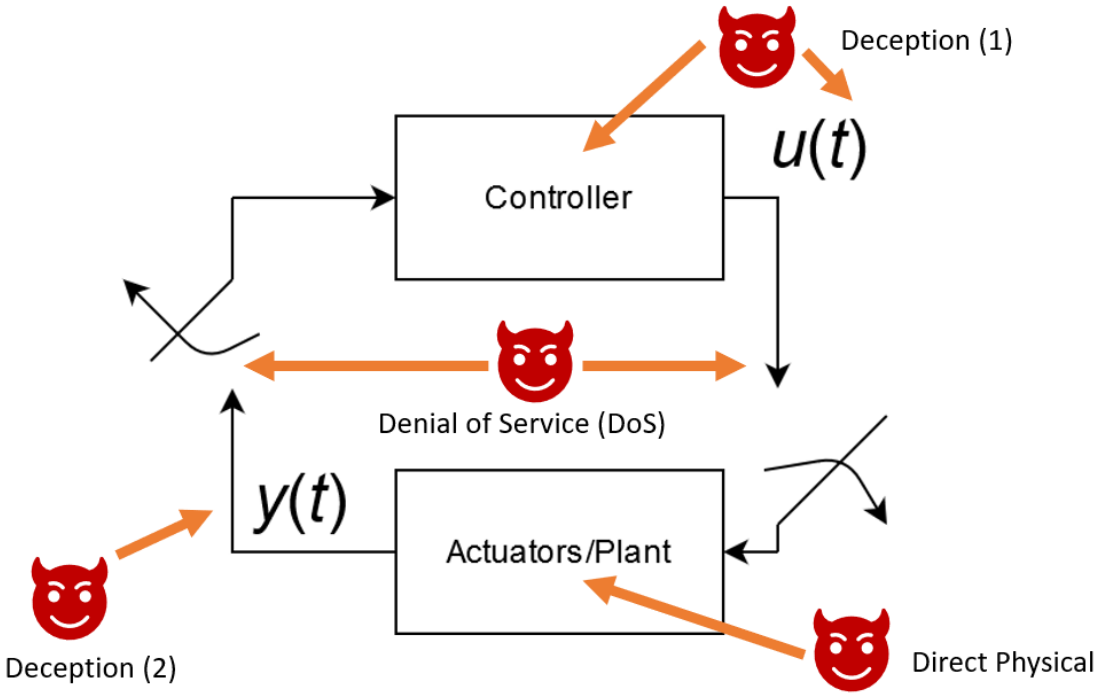


Figure 2.1: Generalized attack model against CPS [19]

This research focuses on the mitigation of deception attacks and therefore uses the more specific attack model detailed in [14]. This attack model presents

attackers of increasing sophistication able to intelligently compromise the controller and sensor outputs of a networked control system. Compromise of the CPS in this manner can drive the system to dangerous states which manifest physical harm. Emphasizes is made to distinguish between cyber and cyber-physical attackers. A cyber attacker does not use knowledge of the dynamics of a system to enhance their capabilities. The more potent cyber-physical attacker uses knowledge of the system model to craft stealthy attacks which avoid detection by masquerading as legitimate system behavior.

2.3 Cyber Security for CPS Network Communications and Software

The cyber vulnerabilities of CPS can be broadly categorized as communication or software vulnerabilities [17]. Traditional cyber solutions can be used or modified to mitigate these cyber vulnerabilities. Cryptographically secure communication channels and protocols can provide confidentiality, integrity, and authentication to CPS communications and transmitted data. A secure end-to-end communication framework, REMP, specifically designed for large scale CPS such as smart meter swarms is proposed in [8]. Device attestation can mitigate malware attacks by verifying CPS components have loaded only trusted software. An embedded device attestation scheme, SEDA, is presented in [9]. SEDA provides remote device attestation via single device and swarm attestation protocols using cryptographic primitives with minimal hardware requirements.

2.4 Reactive Cyber-Physical Security

Reactive security solutions are used to detect and respond to an attack after the effects of the attack have begun to manifest. Reactive solutions include intrusion detection systems (IDS) and health monitoring systems.

Intrusion detection systems (IDS) are a reactive solutions which detect anomalous or suspicious system behaviors. Traffic whitelists create detection rules based on normal CPS network traffic [11]. Whitelist IDS are shown to be useful in detecting cyber attackers but can be subverted by cyber-physical attackers hijacking legitimate communications. Whitelist IDS can be augmented with deep packet inspection that analyzes the system information being communicated for unusual system behaviors [11].

The attestation scheme outlined in [10] attests if sensors or controllers are compromised by injecting control perturbations and observing if the outputs of the CPS components match expectations. The cyber-physical attestation scheme implicitly detects component compromise at the cost of introducing process disturbances and additional trusted verification components.

Health monitoring systems are closely related to fault tolerance solutions. Both can be used to detect if system performance has degraded. Health monitoring systems, such as in [12], are useful in determining if the CPS safety has been compromised by failing subsystems. Heath monitoring solutions suffer in that they can be tricked by a cyber attacker feeding bad data to monitors as in the case of Stuxnet [5].

2.5 Preventative Cyber-Physical Security

Preventative security solutions are designed to mitigate the risk and damage of an attack before the attack occurs. Preventative solutions include signal watermarking, reachability analysis, and actuation limits.

Signal watermarking authenticates legitimate system signals by embedding an authentication challenge in the form of a watermark signal [13]. The watermark signal can be made to change frequently so as to prevent an attacker from extracting the watermark signals from the system signals [14]. Signal watermarking is designed to make arbitrary manipulations of controller and sensor signals by an attacker difficult even if the attacker has obtained access to these signals.

Reachability analysis involves estimation or exact computation of the set of states the CPS can reach. Reachability analysis can be used to determine possible compromise by detecting if the reachable set of states includes dangerous states [15]. A drawback of reachability analysis is its computational cost and requirement of knowing system state.

Actuation limits impose limitations on the reachable states of a system by constraining the actuators. Actuation limits can constrain attackers by minimizing the reachability of dangerous states [16, 20]. A trade-off of limiting the set of reachable states is the imposition of a possible performance penalty. A parameterized hardware shim to implement actuation limits between a controller and actuator is prototyped in [21]. This solution raises an alarm and defaults to a preset safe actuation behavior if dangerous actuation patterns are requested by the controller.

Chapter 3: Background

This chapter provides background information regarding feedback control systems, the CPS attack model, discrete-time state space model, and how actuation limits are implemented in the CPS.

3.1 Feedback Control Systems

In this research CPS are modeled as a feedback control system as illustrated in Figure 3.1. The control system can be represented with the cyber-physical components of a controller, actuator, sensors, and plant. The controller takes in the difference or error, $e(t)$, between a reference signal, $r(t)$, also called the nominal signal, and the current output, $y(t)$, to output the control signal, $u(t)$. The control signal, $u(t)$, is then passed to the actuator. The actuator actuates according to the command signal it receives, $u_a(t)$. The behavior of the physical world, deemed the plant, then yields an output, $y(t)$. Sensors make a measurement of the plant and encode this measurement in another signal, $y_s(t)$. In this manner the CPS tracks the desired reference signal, $r(t)$.

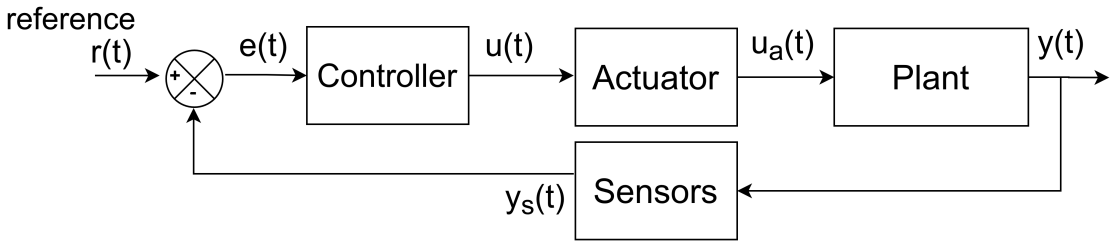


Figure 3.1: Illustration of control system with feedback. The objective of the control system is to minimize the error between the reference signal and the output signal.

3.2 Attack Model

In this research the attack model used to represent an attacker's influence on the CPS is defined in [14]. This model captures how an intelligent adversary can stealthily influence a CPS. In this model the attacker can arbitrarily modify the control signal, $u(t)$, to become $u'(t)$ and the sensor readings, $y_s(t)$, to become $y'_s(t)$. In this research the assumption is made that the attacker can influence the system for a contiguous time window, T_{attack}, before an intrusion detection and response system removes the attacker's influence on the CPS.

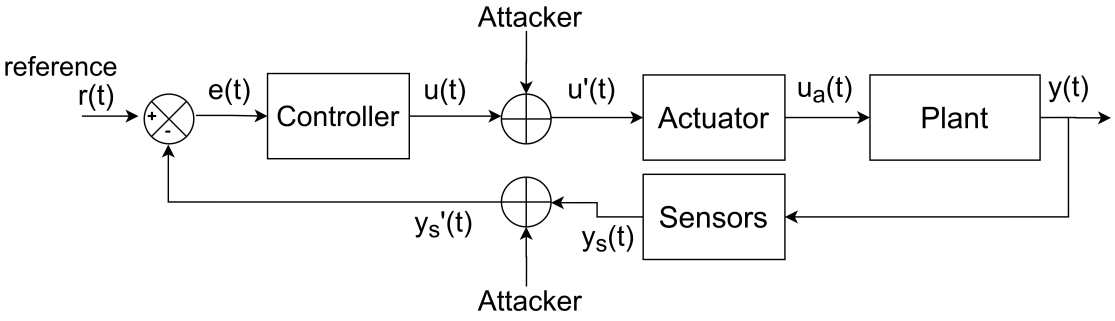


Figure 3.2: Illustration of the Deception Attack Model in Networked CPS. The attacker has complete knowledge of the control system and manipulates the controller and sensor signals to drive the CPS to a dangerous state.

3.3 State Space Model for LTI Systems

State space models may be used to model the control system. State space models utilize the concept of state, "a collection of variables that summarize the past of a system for the purpose of predicting the future" [22, p. 34]. The state variables are organized into the state vector $\mathbf{x}$.

For the purposes of simulation, the control system is modeled as a discrete-time system. If the difference equations which describe the evolution of the discrete-time system are linear and time invariant (LTI), then the control system at time step k can be modeled with the following equations [22, p. 37-38]. In this research only LTI control systems of the following form are considered:

$$\mathbf{x}[k+1] = \mathbf{A}\mathbf{x}[k] + \mathbf{B}\mathbf{u}[k]$$

$$\mathbf{y}[k] = \mathbf{C}\mathbf{x}[k] + \mathbf{D}\mathbf{u}[k],$$

where $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$ are the dynamics matrix, control matrix, sensor matrix, and direct term respectively and $\mathbf{x}$ and $\mathbf{y}$ are the state and output vectors respectively.

3.4 Actuation Limits

In this research actuation limits are implemented by two components, a planner and a shim. The actuation limit planner generates actuation limits and periodically transmits these limits to a shim component. The actuation limits are assumed to be authenticated via a cryptographic authentication protocol and transmitted over a secure secondary channel. The shim component intercepts the controller

output and applies actuation limits to produce a limited controller output, $u_L(t)$. Figure 3.3 is an illustration of the actuation limits planner and shim in the CPS control loop.

Within the shim actuation limits are applied in a cascaded manner. First, artificial limits such as a minimum and maximum controller value are applied. After all artificial limits are applied, actuation limits imposing safe physical operation of the actuator are applied. The actuation cascade and its location in the overall control loop are illustrated in Figure 3.4.

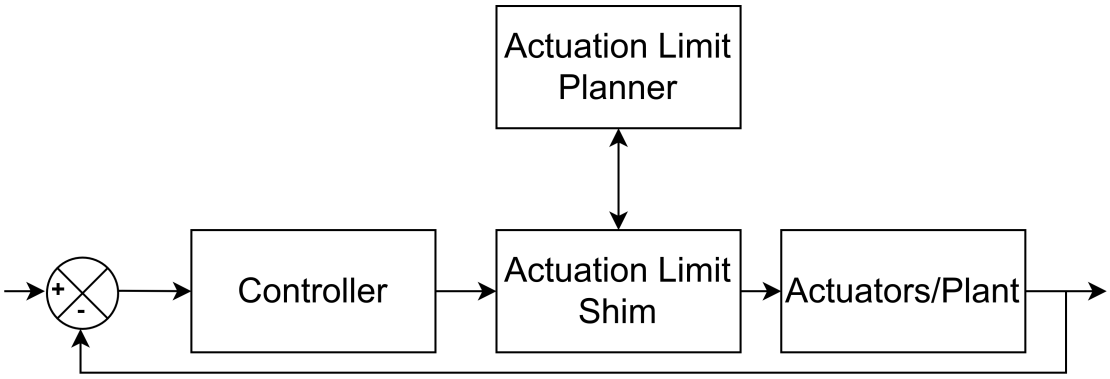


Figure 3.3: Illustration of actuation limits planner and shim in control loop. The actuation limits planner generates actuation limits and periodically transmits them to the shim. The shim intercepts the controller output and applies actuation limits.

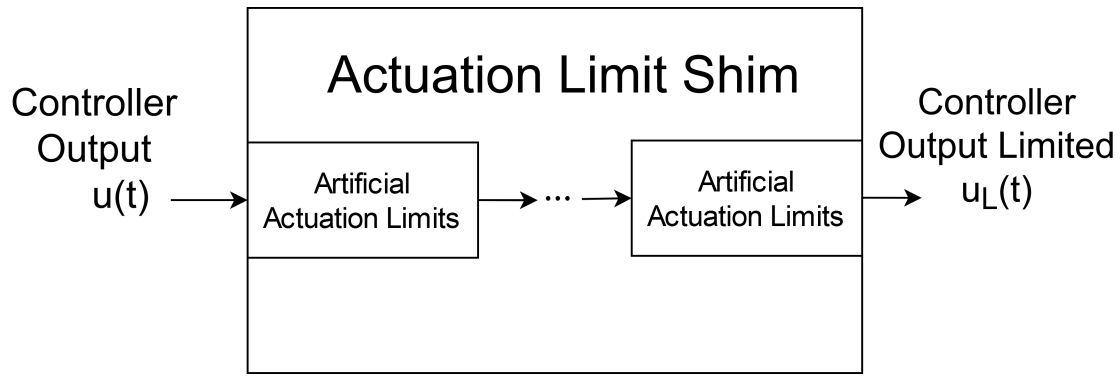


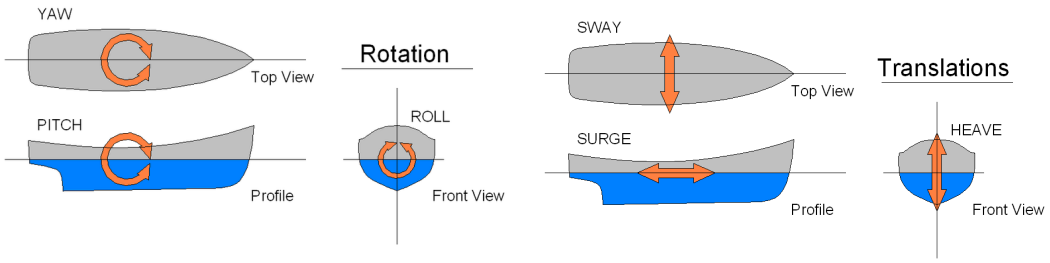
Figure 3.4: Actuation limit shim containing an actuation limit cascade. The actuation limit shim applies actuation limits to the controller output via the actuation limit cascade. The cascade applies artificial then physical limits. The limited controller output is then passed to the actuator.

Chapter 4: Experimental Setup

In this research the CPS system under study is a simulated Mariner class vessel navigating a static obstacle map. This chapter details the components involved in the simulated CPS including the ship motion model, obstacle map generation, path planning algorithm, dynamic obstacle avoidance algorithm, and path following algorithm, and process noise. The chapter concludes with a summary of the CPS under study and the algorithm used to simulate the studied CPS.

4.1 Nomoto Model of Ship Motion

The Nomoto model of ship motion is used to represent the dynamics of ship motion in water. The Nomoto model used in this research is a first order linearization of the six degree of freedom problem of ship motion. Figure 4.1 is an Illustration of the degrees of freedom associated with ship motion. The Nomoto model reduces ship motion to two dimensions (2-D) defined by the surge translation and yaw rotation. The relevant properties of the Nomoto model include surge speed U, rudder angle d, heading angle θ, yaw rate ϕ, and the two dimensional (2-D) spatial location coordinates (x, y).



(a) Ship Rotations (b) Ship Translations

Figure 4.1: Illustration of the six degrees of freedom for ship motion. The Nomoto model reduces the degrees of freedom to the surge translation and yaw rotation.

The first order Nomoto model is chosen due to its simplicity, reasonable accuracy, and desirable control properties including controllability and observability [23]. The properties of controllability and observability allow for the implementation of the state and feedback controller used in this research.

The Nomoto model in continuous state space can be represented with the following terms [23] where T is the time constant and K is the static yaw rate gain:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ 0 & -1/T \end{bmatrix} \mathbf{B} = \begin{bmatrix} 0 \\ K/T \end{bmatrix} \mathbf{C} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \mathbf{D} = \mathbf{0}$$

$$\mathbf{x} = \begin{bmatrix} \text{heading angle} \\ \text{yaw rate} \end{bmatrix} = \begin{bmatrix} \theta_{OS} \\ \phi \end{bmatrix}$$

$$y = \text{heading angle} = \theta_{OS}$$

The Nomoto model can be further extended to accept an arbitrary instantaneous surge speed U if given the length of the ship L by setting the T and K

parameters as follows [24]:

$$T = T'L/U$$

$$K = K'U/L$$

A Mariner class vessel is chosen to be the ship modeled in the simulated CPS. Mariner class vessels are well studied in the literature and their Nomoto model parameters have been empirically determined. The T and K parameters for a Mariner class vessel of length $L = 161\,\text{m}$ with constant surge speed $U = 7.7\,\text{m/s}$ are known to be $T = 107.3$ and $K = 0.185$ through maneuvering tests [23]. This allows for the derivation of the $T' = 5.13$ and $K' = 3.87$ parameters.

The continuous state space matrices shown above can then be discretized using a zero order hold method for discrete-time simulation. Using the above information, a discrete time first order model of Mariner vessel motion can be simulated.

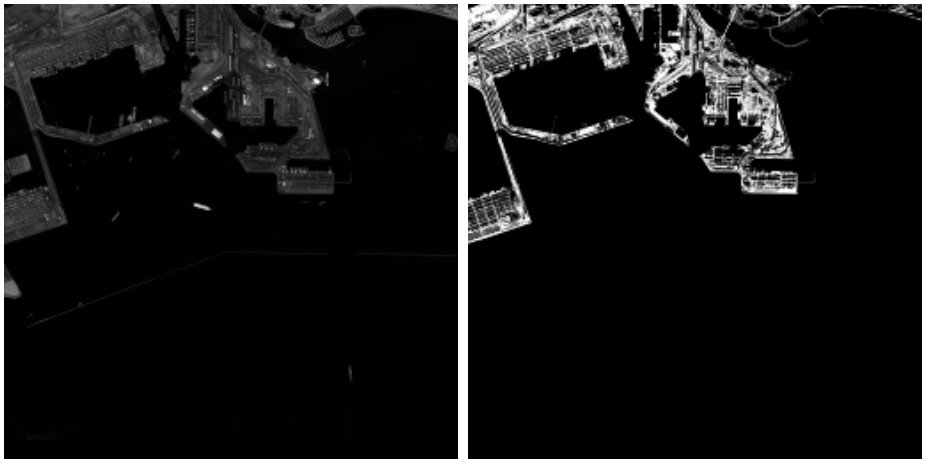
4.2 Static Obstacle Maps

The Mariner vessel requires a static obstacle map in which to navigate. The National Oceanic and Atmospheric Administration (NOAA) provides high resolution 8-bit greyscale coastline satellite imagery with known spatial resolution through its Data Access Viewer tool [1]. Various ports and sections of coastline are selected as maps on which the Mariner vessel can navigate.

The greyscale imagery contains more information than required. For the purposes of this research, only a binary image is needed where a value of 1 indicates land

[1] `https://coast.noaa.gov/dataviewer`

as an obstacle and a value of 0 indicates unoccupied water. The greyscale imagery is thresholded and morphologically eroded to eliminate noise. Four 16 384 pixels by 16 384 pixels binary maps are produced with a spatial resolution of 0.5 m/s using this process. An example of one such map before and after processing is shown in Figure 4.2.



(a) Greyscale Image (b) Binary Image

Figure 4.2: Satellite imagery of the port of San Francisco transformed into a binary static obstacle map. The greyscale image, Figure 4.2(a), is thresholded and eroded to yield a binary map, Figure 4.2(b).

4.3 Path Planning

Path planning functionality over the binary map environment is provided by the D* algorithm. D* is chosen as a proven graph traversal algorithm able to efficiently plan in the presence of dynamic obstacles. D* has the desirable properties of providing the optimal traversal of a graph in addition to relatively fast re-planning compared to its predecessor A* [25]. To prevent the D* algorithm from providing paths too close to land or near the edges of the map, the static obstacle map

is dilated with a circular structuring element with a radius of 880 pixels (440 m), roughly triple the length of the Mariner vessel. After dilation, the binary image is downsampled to 128 pixels by 128 pixels for faster D* performance. Paths are generated by D* between random open start and end pixels with a minimum distance of 1280 m to minimize the generation of trivial paths. Figure 4.3 shows a dilated and downsampled static obstacle map used as input to the D* path planner. Figure 4.4 shows two example paths generated by the D* path planner.

Figure 4.3: Dilated and downsampled binary image of static obstacle map for D* path generation. The static obstacle map is dilate to prevent the generation of paths too close to the static obstacles. The static obstacle map is downscaled for faster D* performance.

(a) Example 1 of D* Generated Path (b) Example 2 of D* Generated Path

Figure 4.4: Illustration of two examples of D* Generated Paths. Random start and end pixels a minimum distance apart are chosen and the D* algorithm generates the optimal path from start to end.

4.4 Dynamic Obstacle Avoidance

Bounding circles are an effective method by which CPS may avoid stationary obstacles [26]. In this method, assuming the obstacle is represented by occupied spaces on a grid, the obstacle is dilated with a circular structuring element to generate a circular clearance by which to avoid the obstacle. The circle formed by this dilation is the bounding circle. A new path which avoids the obstacle is then planned by utilizing points tangential to the bounding circles. This method is chosen due to its requirement of only simple mathematical operations and fast runtime. In our case the radius of the bounding circle is chosen to be 528 m or roughly 3.3 times the length of the ship.

At the start of select ship navigation simulations, 40 m by 40 m obstacles may be inserted at points along the nominal path. The vessel is assumed to detect the

obstacle upon starting the simulation and plans two avoidance paths, one for each tangent point on the bounding circle. If the ship starts the simulation within the bounding circle, then the avoidance paths cannot be generated and the original path is maintained. The avoidance paths correspond to the two possible circumnavigations of the bounding circle, either clockwise or counterclockwise. The vessel is assumed to possess a computational system that selects the path that maximizes the closest point of approach (CPA). This computational system is emulated by simulating the following of both avoidance paths and selecting the path with the higher CPA. Figure 4.5 is an illustration of the two possible avoidance paths circumnavigating the bounding circle around an example obstacle.

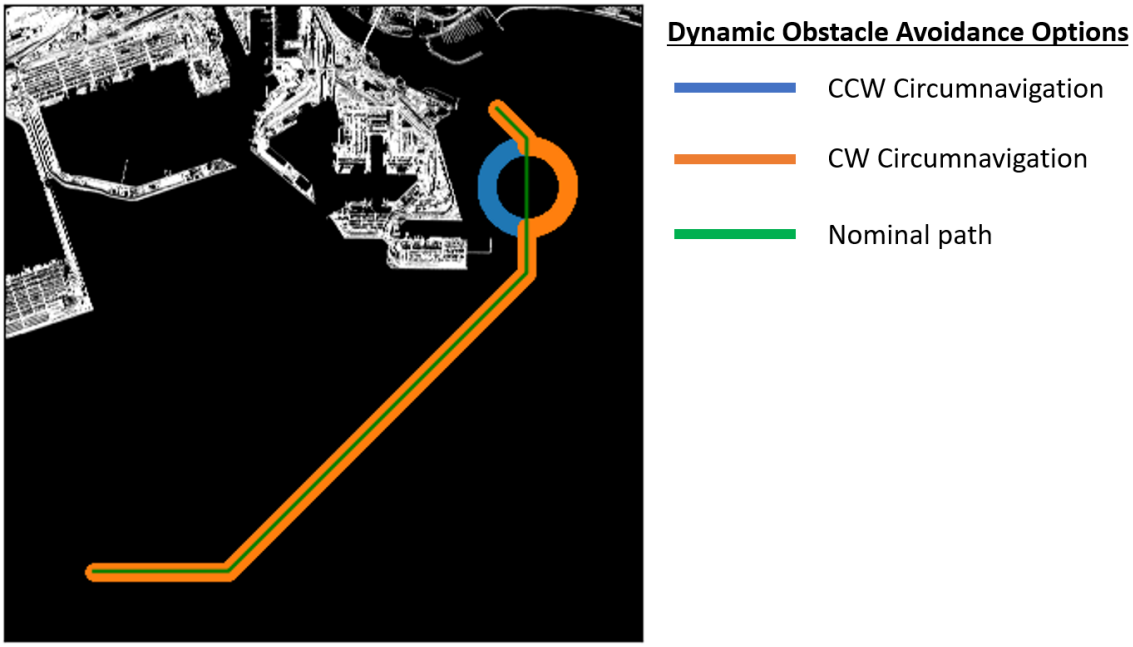


Figure 4.5: Illustration of the two avoidance paths circumnavigating the bounding circle of a dynamically inserted obstacle. The avoidance path with the greater CPA is chosen by the CPS's computational system.

4.5 Path Following

The rudder angle of the simulated Mariner vessel is controlled by a proportional derivative (PD) controller in a feedback control system. The surge speed is assumed to be held constant at $U = 6\,\mathrm{m/s}$. The vessel, referred to as the own ship (OS), maintains course by computing the angle $\Delta\theta$ between its current heading angle $\theta_{\mathrm{OS}}(t)$ and a target point on the path approximately 200 m ahead of its current position. Figure 4.6 shows a diagram of the course keeping components. The target point is picked approximately 200 m from the current position of the OS to minimize ringing in the path following. The parameters of the PD controller are $K_P = 0.5$ and $K_D = -20$ for proportional and derivative control respectively. The K_P determines the influence of the proportional difference from the reference and K_D determines the influence of the derivative of the output on the control signal. The K_D coefficient is large in our case to counteract the large turning inertia of the Mariner vessel. The equation for the controller output $u(t)$ is given in Equation 4.1.

$$u(t) = K_P \Delta\theta(t) + K_D \frac{\partial \Delta\theta(t)}{\partial t} \tag{4.1}$$

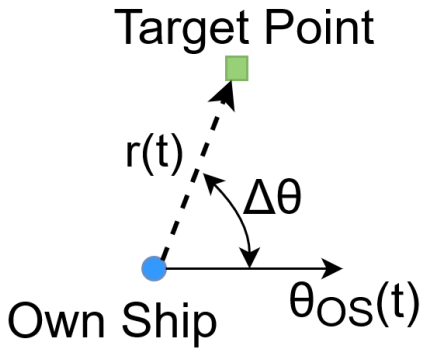


Figure 4.6: Diagram of course keeping components. The own ship PD controller steers the rudder by processing the proportional and differential difference between the heading angle and the target point on the path.

4.6 Process Noise

Noise is added to the system to simulate the presence of noise such as unpredictable waves and ocean currents. Additive white gaussian noise is added to the nominal controller output and surge speed to yield an effective rudder angle and surge speed at each time step in the simulation. The rudder angle noise is defined as $n_u \sim \mathcal{N}(0^\circ, (2.5^\circ)^2)$ and surge speed noise as $n_U \sim \mathcal{N}(0\,\mathrm{m/s}, (0.5\,\mathrm{m/s})^2)$.

4.7 Simulation of Mariner Vessel

A summary of the simulated CPS and the simulation algorithm are described as follows. The CPS system under study is a simulation of a Mariner class vessel with a first order ship motion model. Navigation takes place on maps derived from NOAA coastal satellite imagery with a spatial resolution of 0.5 m/pixel to attempt to capture navigation in a real world environment. Simulation of the CPS occurs in the following steps. First, a target point on the path approximately 200 m from the

own ship is selected as the target point. The difference in own ship heading angle and the target point is calculated. The angle difference is passed to the PD controller to generate the own ship rudder angle u. The controller output u is then passed to the actuation shim. The actuation shim applies actuation limits to the controller output to generate u_L. The output of the actuation shim u_L and the nominal surge speed U are made noisy via the additive noise terms n_u and n_U to capture the effect of process noise. The discrete-time Nomoto model is then stepped forward in time with the noisy inputs. The simulation continues until the vessel is within 483 m (3 lengths of the vessel) of the path end point, the vessel collides with an obstacle, or the simulation times out after 5000 s. The simulation algorithm described above is illustrated in Figure 4.7.

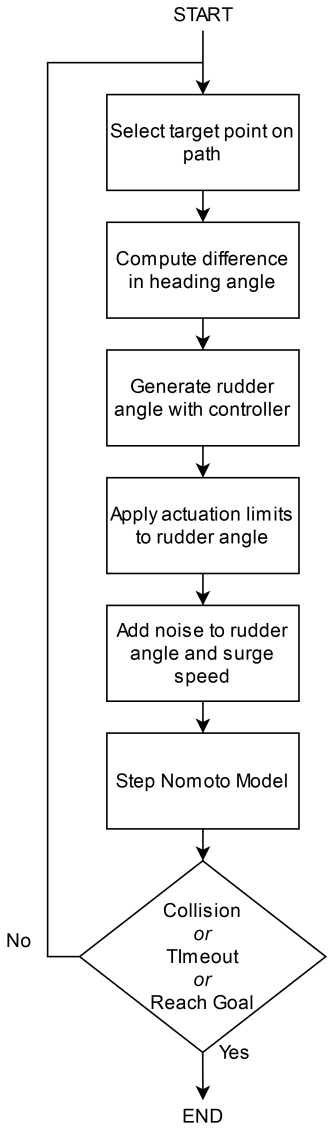


Figure 4.7: Algorithm to simulate mariner vessel path following on static obstacle map. The vessel starts at the start state and follows the path to the end state. The simulation terminates if the vessel collides with an obstacle, exceeds the timeout time, or reaches a state near the end state.

Chapter 5: Actuation Limits

This chapter details the generation of time-based actuation limits. Three metrics, attack resilience, performance integrity, and bandwidth utilization, are introduced to score the actuation limits. Figure 5.1 is an illustration of the CPS inputs used to generate these metrics. An algorithm to combine actuation limits to provide an amalgam limit scheme with desirable properties is detailed. The chapter concludes with a demonstration of how limit schemes are scored on an example path.

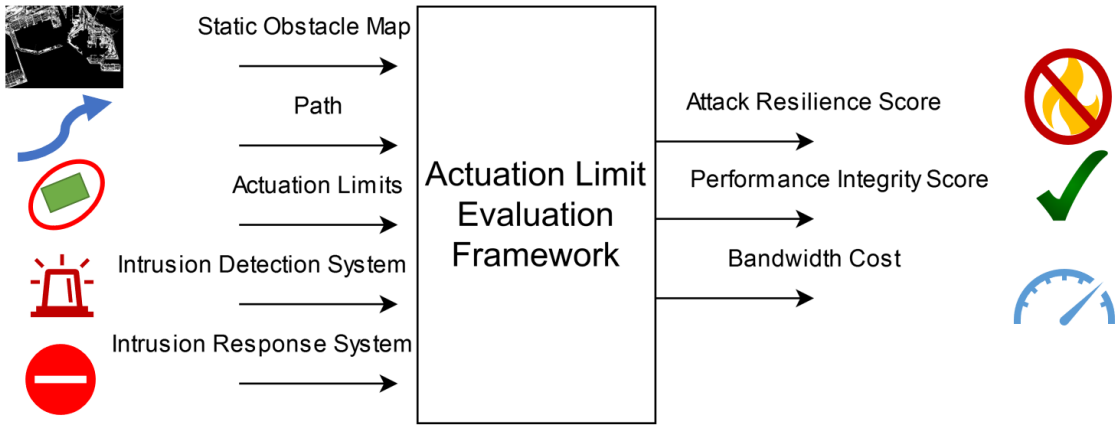


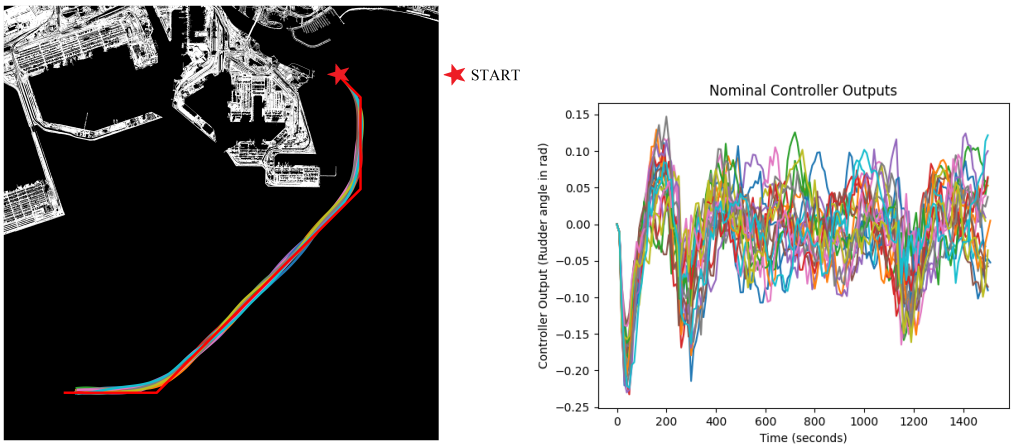
Figure 5.1: CPS inputs to framework and corresponding metrics output

5.1 Generation of Actuation Limits

In this research actuation limits are implemented as a function of time. Actuation limits may also be implemented as a function of state depending on the application.

To generate the actuation limits as a function of time, controller output traces under nominal conditions over time are gathered. To this end a number of nominal runs are simulated and the controller outputs recorded. Numerous nominal runs are collected to account for different controller output traces due to noise. Figure 5.2 is a collection of twenty path following simulations under nominal conditions. Figure 5.2(a) is an illustration of the Mariner vessel following the path over time. Figure 5.2(b) is a plot of the controller output traces corresponding to the nominal runs.

This research studies both physical and artificial limit schemes. In the artificial limit schemes there are a maximum and minimum allowed instantaneous actuation value per each update period. Contrast the artificial schemes to the omnipresent physical limit scheme which has a maximum and minimum instantaneous actuation value in addition to a maximum absolute actuation differential. The physical limit scheme is implemented to restrict the actuator to only physically realizable actuation values that do not damage the actuator. In the case of the Mariner vessel, the physical actuation limits are instantaneous rudder values d where $d \in [-45°, 45°]$ and a maximum absolute differential value of $5\,°/\text{s}$.

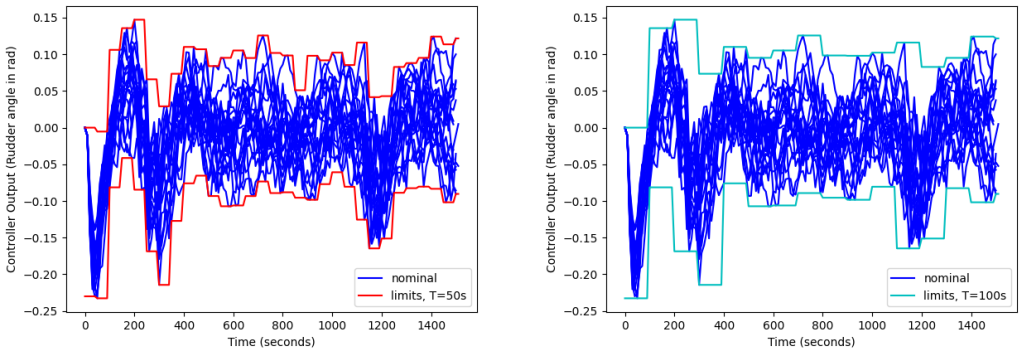


(a) Nominal Path Following Locations over Time

(b) Nominal Controller Outputs over Time

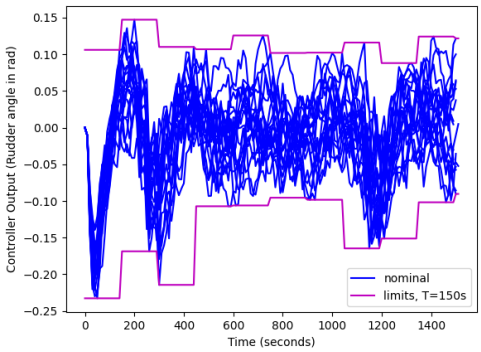
Figure 5.2: Collection of controller output traces over time under nominal conditions. Figure 5.2(a) plots the location of the vessel during the nominal runs over time. Figure 5.2(b) plots the controller output traces corresponding to the nominal runs.

One of the defining properties of the artificial actuation limits is the update period. Periods of 50 s, 100 s, 150 s, 200 s and 400 s are considered in this research. Within each update window the maximum and minimum value of the collection of nominal controller outputs become the artificial actuation limits max and min respectively. An example of the generation of actuation limits for various update periods is shown in Figure 5.3. It is important to note that actuation limits with faster update rates more tightly bound the envelope of the controller outputs. Each artificial limit also has an associated margin. This margin is an additive value to the maximum and minimum controller values within the update period. This margin increases the number of operational states if positive and decreases the number if negative. Actuation limits with the update period held constant with varying margins are shown in Figure 5.4.

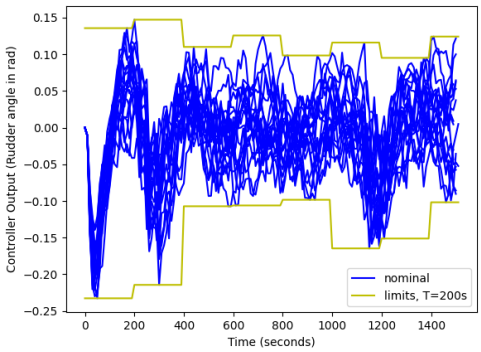


(a) Limits Update Period of 50 s

(b) Limits Update Period of 100 s

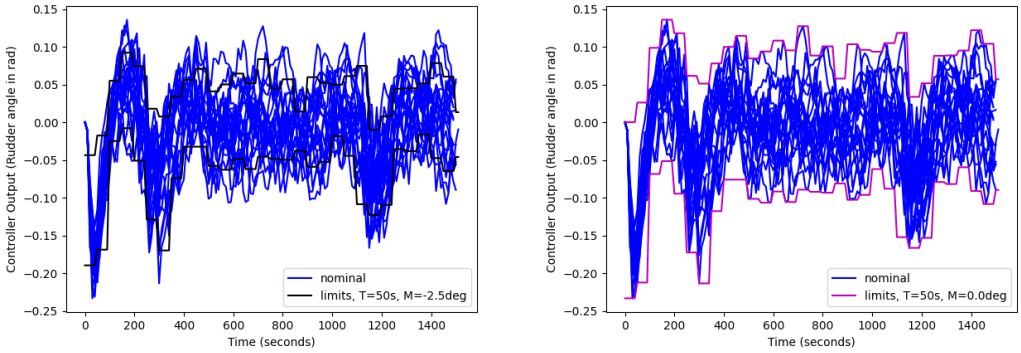


(c) Limits Update Period of 150 s



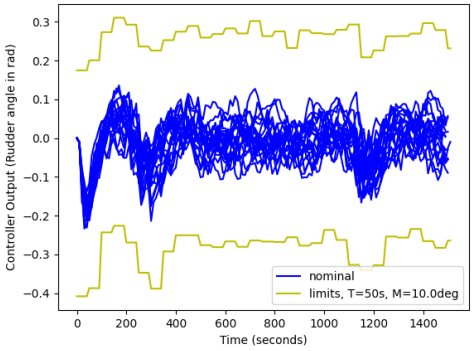
(d) Limits Update Period of 200 s

Figure 5.3: Examples of actuation limits generated with various update periods. The maximum and minimum controller output values within a period are set as the maximum and minimum actuation limit values for that period respectively.

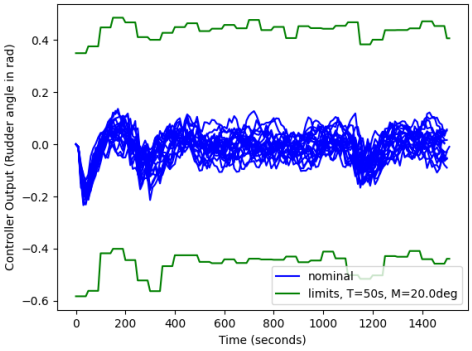


(a) Limits with margin -2.5°

(b) Limits with margin 0°



(c) Limits with margin 10°



(d) Limits with margin 20°

Figure 5.4: Example of actuation limits with constant update period and varying margins. The margin is an additive constant to the actuation limits within each update period.

5.2 Discussion of an State-Based Actuation Limit Scheme

A state-based actuation limit scheme presented in [1] is contrasted with the purely time-based actuation limit scheme used in this research. In the state-based actuation limit scheme, zones around static obstacles are placed wherein actuation is limited to mitigate risk of collision. Larger zones correspond to less risk of collision. The actuation limits in the zones are formulated such that the CPS has roughly one minute to recover in the event of deviation from the nominal path.

The implementation of the state-based actuation limits is similar to the time-based actuation limit scheme used in this research. One of the most significant similarities is that both can be implemented in a periodic fashion. Both the time-based and state-based limit schemes can be enforced by an actuation limit shim periodically updated by a planner. Specifically in the case of the state-based limits, the planner periodically polls the current limit zone to determine the actuation limits. Another similarity is that both schemes depend on the length of the update period for timely and effective implementation of actuation limits. A shorter update period benefits the time-based limits by allowing for tighter bounding of the controller output envelope. A shorter update period also benefits the state-based limits by ensuring the effected actuation limits correspond to the current state of the system. Consider the worst case scenario for an update in a state-based limit scheme in which an update occurs immediately preceding a transition into a more restrictive limit zone. In this case the actuation limits will be stale for almost an entire period before updating to the more constraining limits. In such a case an attacker can exploit the less restrictive limits for a time about equal to the update period. Thus, the update period must be carefully considered when implementing both the time and state-based actuation limit schemes.

A noteworthy difference in the actuation limit schemes is the information required for their implementation. The state-based scheme utilizes expert knowledge of the CPS to craft the size of the zones and specific limits associated with each zone. In contrast, the time-based actuation limit scheme uses simulations of the CPS along the nominal path to generate limits. The state-based limits have the

beneficial properties of being applicable independent of time aside from the update period and independent of the nominal path. In contrast, the time-based limits are only applicable to a specific nominal path as a function of time.

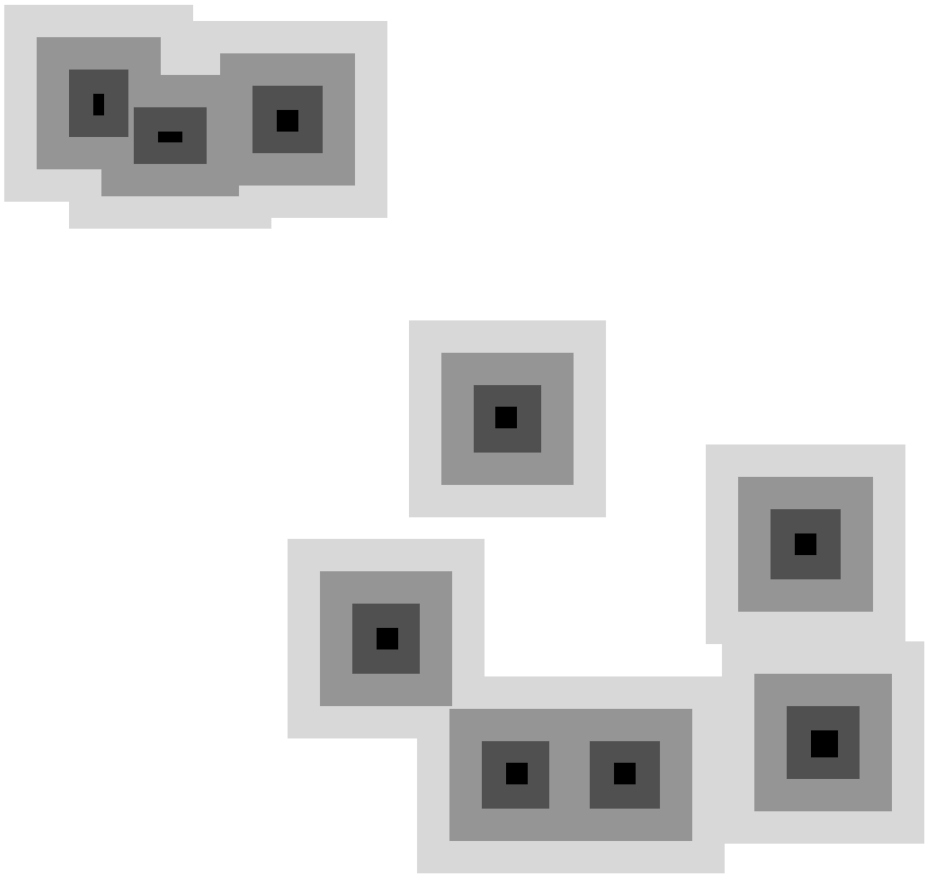


Figure 5.5: Illustration of state-based actuation limits scheme described in [1]. Black represents obstacles and shades of gray represent various actuation limit zones. Actuation limit zones are placed around obstacles to mitigate the risk of collision. In the event of attack, larger zones are considered less risky and smaller zones more risky. Larger zones correspond to less constraining actuation limits.

5.3 Metrics to Evaluate Actuation Limits

This research is concerned with assessing the benefits and penalties associated with the imposition of actuation limits on a CPS system. In order to assess these impacts, three metrics are devised. The first metric is attack resilience to capture how well the limits protect the CPS in the event of a cyber-physical attack. The second metric is performance integrity to capture how the actuation limits affect the operational capability of the CPS. The third metric is bandwidth utilization to capture the cost of implementing actuation limits in terms of network resources.

5.3.1 Attack Resilience Metric

The goal of the attack resilience metric is to measure if the actuation limits are useful in constraining a cyber-physical attacker. The attack model assumes that an attacker has a contiguous, finite time window in which the attacker can arbitrarily modify the controller output value. This represents a cyber-physical attacker conducting a stealthy attack against the system until an IDS or human operator detects the attack and eliminates the attacker's influence on the CPS.

The metric is computed as follows. First, attack data points at equally spaced times are selected by dividing the maximum time for a nominal run by the number of desired attack data points. Twenty attack data points are collected for the metric in this research. At each attack data point the CPS simulation state is saved and secondary simulations are forked with each simulating a different possible attack type. This research considers five types of attack listed in Table 5.1. A demonstration of

how each attack type impacts the CPS is shown in Figure 5.6. If any of the attack types result in a collision then a collision is reported for the attack data point. The selection of the most devastating attack emulates the attacker having knowledge of how to drive the system to an unsafe state. The mean number of collisions is then taken to be the attack resilience metric. For instance, if three collisions are reported in twenty attack data points then the attack resilience metric is computed to be 0.15. Equation 5.1 is the formula for computing the attack resilience score S_{AR} where B_i is the collision value of attack data point i and N is the number of attack data points.

$$S_{\mathrm{AR}} = \sum_{i=1}^{N} B_i/N \tag{5.1}$$

Table 5.1: Attacks simulated in attack resilience metric

Attack Type	**Controller Output**	**Attack Duration** [s]
high	actuator physical maximum	80
high half	half of the actuator physical maximum	80
low	actuator physical minimum	80
low half	half of the actuator physical minimum	80
hold	hold the last previous legitimate output	80

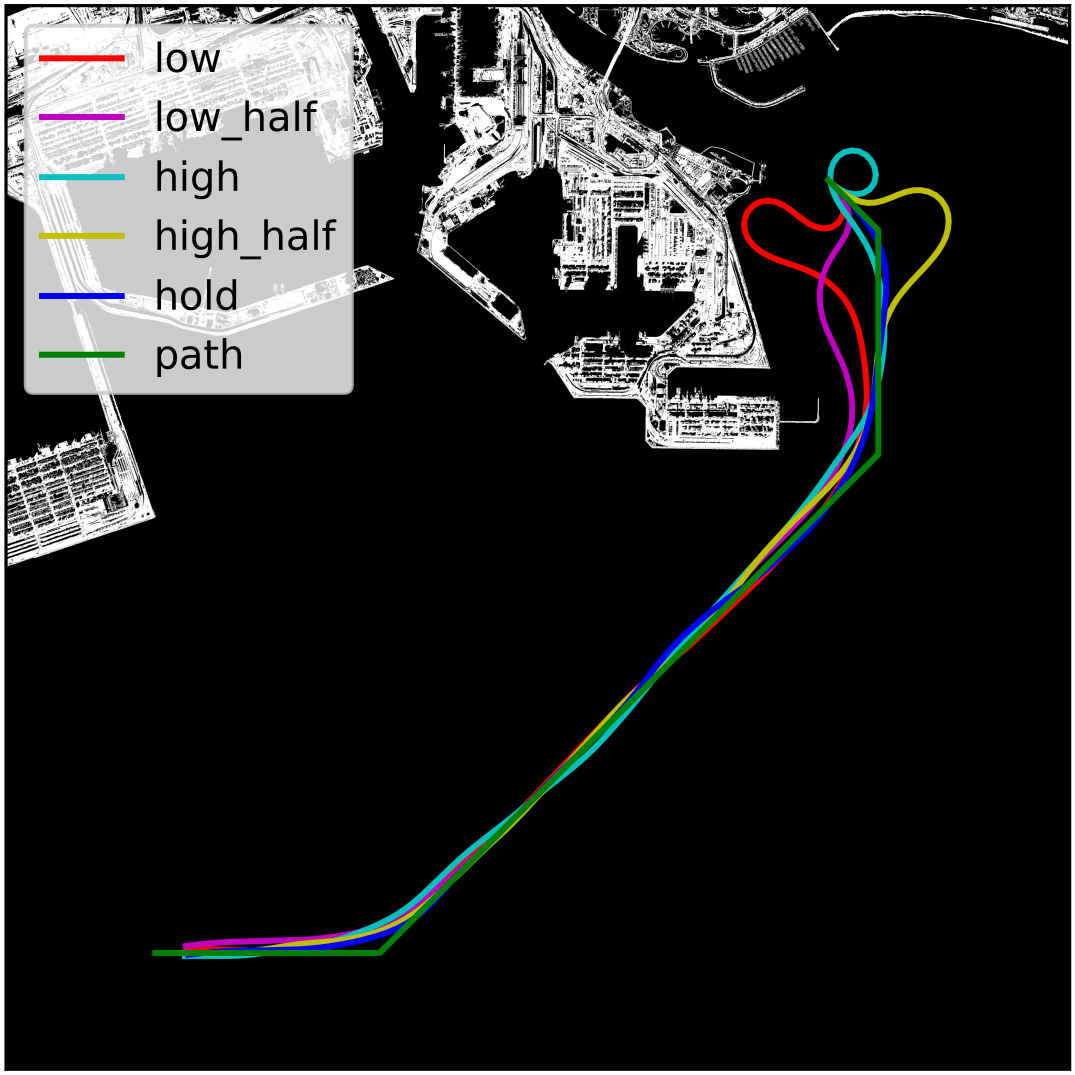


Figure 5.6: Demonstration of the Impact of Attacks on Mariner Vessel Navigation. The attack starts at the start of the simulation when the vessel is in the upper right corner. Note the low attack almost causes a collision.

5.3.2 Intrusion Response System Resolution of Attack

The modeling of an intrusion response system (IRS) is an augmentation to the attack resilience metric. The IRS provides a more nuanced method of resolving attacks. The IRS may be presented multiple possible resolutions to an attack. For

example, an attack against the own ship may be considered resolved if the own ship can complete a 180° turn maneuver. Completing such a maneuver likely indicates the complete mitigation of the effects of the attack. Alternatively, the own ship could return to the nominal path and reach the target end state. Figure 5.7 is an illustration of three possible attack resolution options available after an attack on the own ship. Given multiple attack resolution options, the framework can evaluate one or many possible attack resolution options during the evaluation of attack resilience. In this research, reaching the target end state is considered the only valid resolution of an attack.

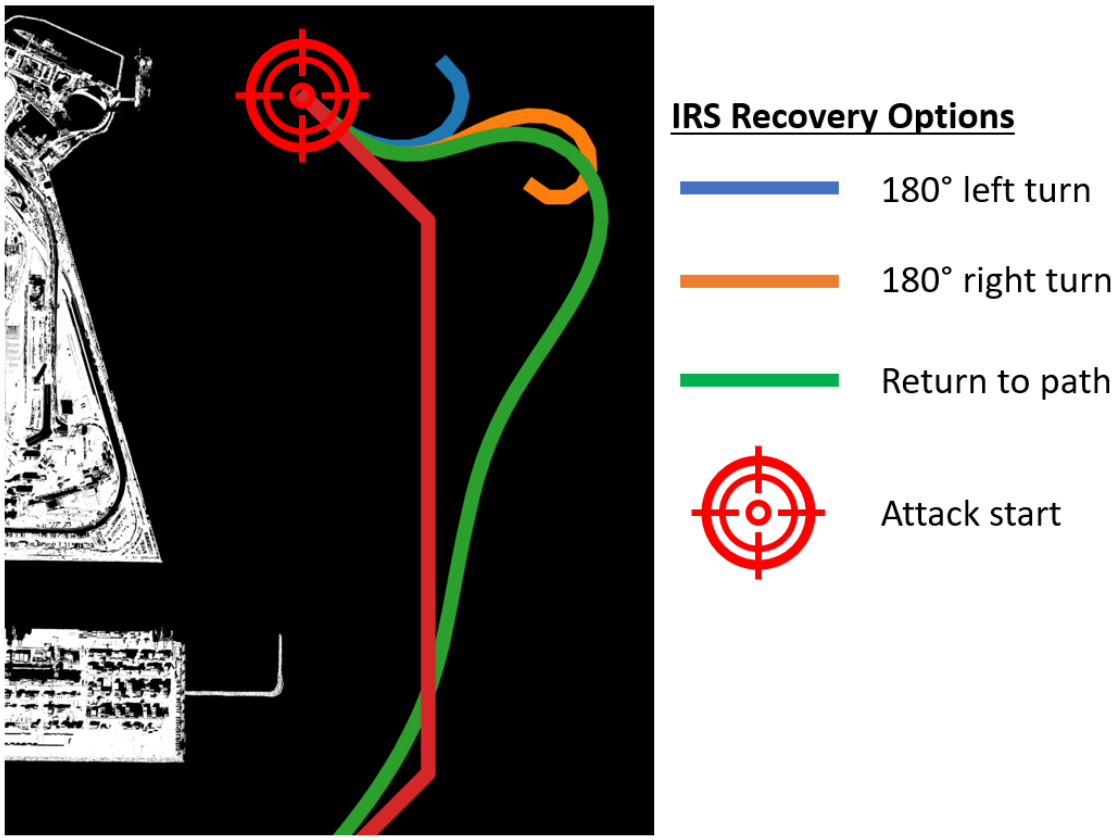


Figure 5.7: Attack resolution options available to the IRS. After regaining control, the attack may be considered resolved if any one of the options can be satisfied. In this research, only the "return to path" option is considered.

5.3.3 Performance Integrity Metric

The goal of the performance integrity metric is to measure degradation in system performance caused by the imposition of actuation limits. The aim of the metric is to determine if the CPS can accommodate demanded deviations from nominal behavior while constrained by actuation limits.

The performance integrity metric attempts to qualitatively capture loss of performance by injecting obstacles on the nominal path. These obstacles represent previously safe states that have unexpectedly become dangerous. The obstacles are injected one per simulation run at equally spaced points along the nominal path. The CPS is challenged to avoid these obstacles by maintaining a maximum closest point of approach (CPA) while still attempting to follow the nominal path. The metric is then computed as the mean of the CPAs scaled by the mean CPA achieved under the physical limits to yield a score from 0 to 1. Scores above 1 are set to 1. Equation 5.3 is the formula for computing the performance integrity score S_{PI} where CPA_i is the CPA for the ith obstacle, N is the number of obstacles, $\mathbf{E}[\text{CPA}_{\text{physical}}]$ is the mean physical CPA.

$$\mathbf{E}[\text{CPA}_{\text{physical}}] = \sum_{i=1}^{N} \text{CPA}_i / N \tag{5.2}$$

$$S_{\text{PI}} = \sum_{i=1}^{N} \frac{\text{CPA}_i}{\mathbf{E}[\text{CPA}_{\text{physical}}]} \tag{5.3}$$

5.3.4 Bandwidth Utilization Metric

The objective of the bandwidth utilization metric is to estimate the amount of network resources the actuation limit scheme demands. The bandwidth utilization metric is computed by counting the number of transmissions between the actuation limit planner and the actuation limit shim during runtime.

5.3.5 Greedy Algorithm for Amalgam Actuation Limits

One use of the attack resilience and performance integrity scores is to algorithmically combine actuation limit schemes to constrain attackers while minimizing bandwidth usage and loss of performance integrity. We develop an algorithm which first groups limits by update period. Each limit scheme within each update period group is then ranked by performance integrity score. During each update period, each update period group starting with the largest period is searched for a limit scheme which has an acceptable attack resilience score. If an acceptable attack resilience score is found, the search stops and that scheme is picked to be transmitted to the shim for this update period. In this way, a scheme is picked with acceptable attack resilience score that also prefers low bandwidth and high performance integrity. For our purposes, an acceptable attack resilience score is 0. In this fashion an amalgam actuation limit scheme is formed that prefers high performance integrity and low bandwidth utilization with acceptable attack resilience is created. The amalgam scheme has the benefit of a dynamic update period which prefers to use bandwidth efficiently. Bandwidth use is generally high when the CPS in

near dangerous states to provide good attack resilience, but this bandwidth use can be lowered when in safer states where maintaining attack resilience requires fewer updates.

5.4 Evaluation of Actuation Limits in Example Ship Scenario

In this section, an example of how actuation limits are scored in terms of attack resilience and performance integrity is presented. In the example, time-based actuation limit schemes with various margins and a fixed update rate of 100 s are scored. Observations of general trends regarding attack resilience and performance integrity scores on an individual path basis are also discussed.

5.4.1 Attack Resilience Scoring Example

In this example twenty attack data points evenly spaced in time are gathered to compute the attack resilience score for each considered actuation limit scheme. Table 5.2 contains the attack resilience scores for the example path.

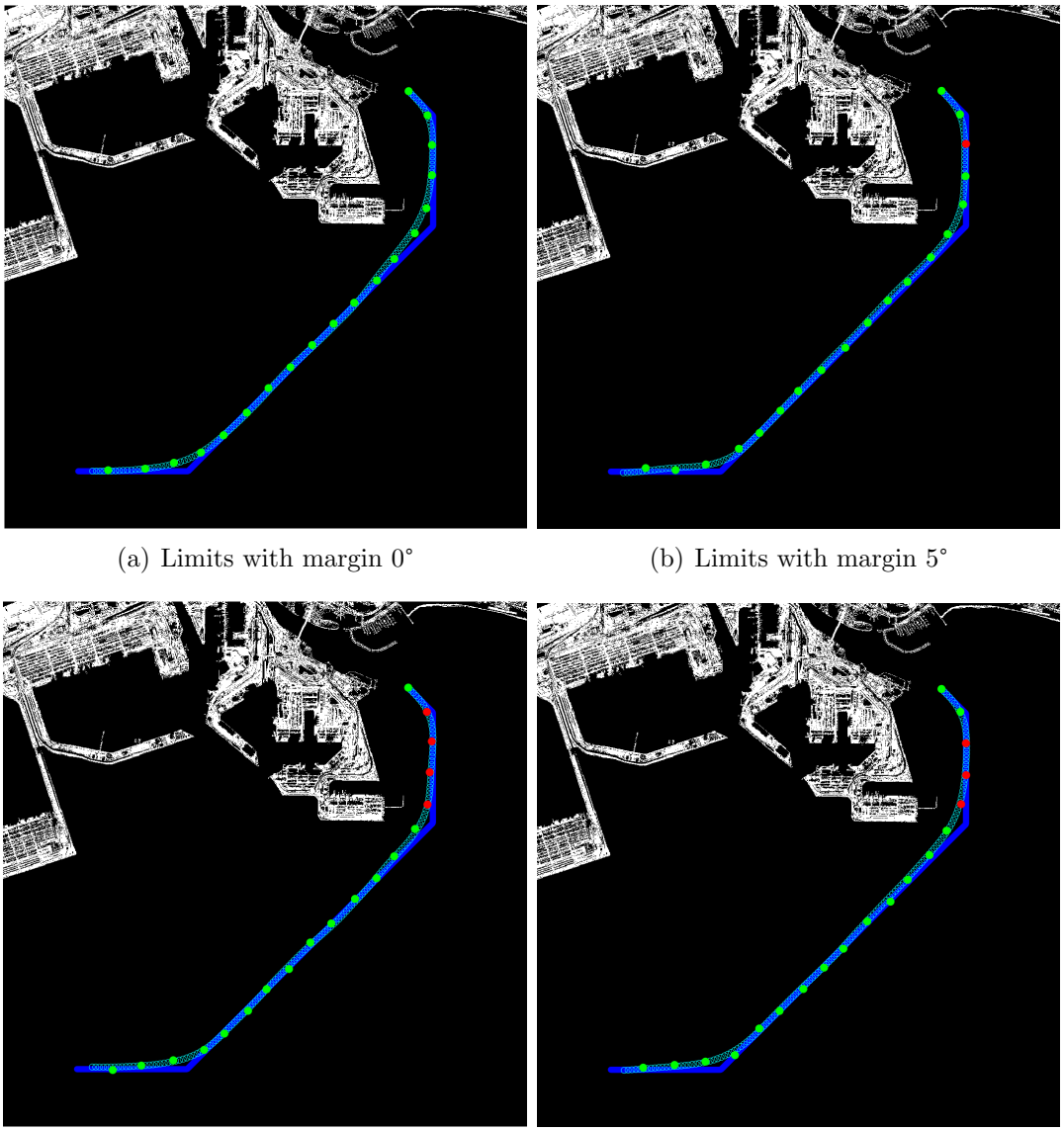
The physical limits set a baseline against which the actuation limit schemes are compared. Note that the physical limits suffer from a 0.16 attack resilience or on average 16 % of the attack data points result in a collision. The limits with a margin of 0° are noteworthy in that its configuration is the most restrictive without impeding on the nominal behavior. In this scenario the limits with margin 0° are able to fully mitigate the attacker, boasting a attack resilience score of 0. If the limits are too stringent as in the case of a margin of −5°, then the system behaves poorly

and an attacker can readily cause a collision if the system does not fail outright. As the margins increase the beyond 0° the scores become worse, with the larger margins of 10° and 20° roughly equal to the physical limit score. If the number of attack data points were increased, the physical limits and limits with large margins would likely converge to similar values. This increase in score is explained by the attacker exploiting an increasing number of extraneous operation states that enhance the attacker's influence on the CPS.

Figure 5.8 is an illustration of how the attack data points map to the position of the ship during path following in time and space. Red dots indicate attack data points that resulted in collisions while green dots indicate no collision occurred. Note the collisions occur when the ship is near a land mass and is navigating a turn such that the vessel is gaining yaw rate towards the nearby land mass. Intuitively this may be recognized as a precarious situation as the ship is gaining momentum towards a dangerous state without much ability to recover in the event the ship goes off course. An attacker can exploit this situation to cause a collision as indicated by the attack data points. Actuation limits help remedy this situation by constraining the attacker's ability to deviate the CPS from nominal to dangerous states.

Table 5.2: Attacks resilience scores for select actuation limits. Note that attack resilience suffers as the margins become more extreme. Extreme negative margins introduce deviation from the nominal by compromising controller capability. Extreme positive margins permit numerous extraneous states that may be exploited by an attacker.

Update Period [s]	**Margin** [°]	**Attack Resilience Score (lower is better)**
Physical	Physical	0.16
100	-5	0.25
100	-2.5	0
100	0	0
100	5	0.05
100	10	0.21
100	20	0.21



(a) Limits with margin 0°

(b) Limits with margin 5°

(c) Limits with margin 10°

(d) Physical Limits

Figure 5.8: Attack resilience plots varying limit margin. Red dots indicate the attacker can cause a collision at that location. Note that as limit margins increase the attack resilience score becomes worse as the vessel is exposed to more dangerous states. Actuation limits which tightly bound the nominal behavior, such as a those with 0° margin, minimize the number of dangerous states and prevent collisions.

5.4.2 Performance Integrity Scoring Example

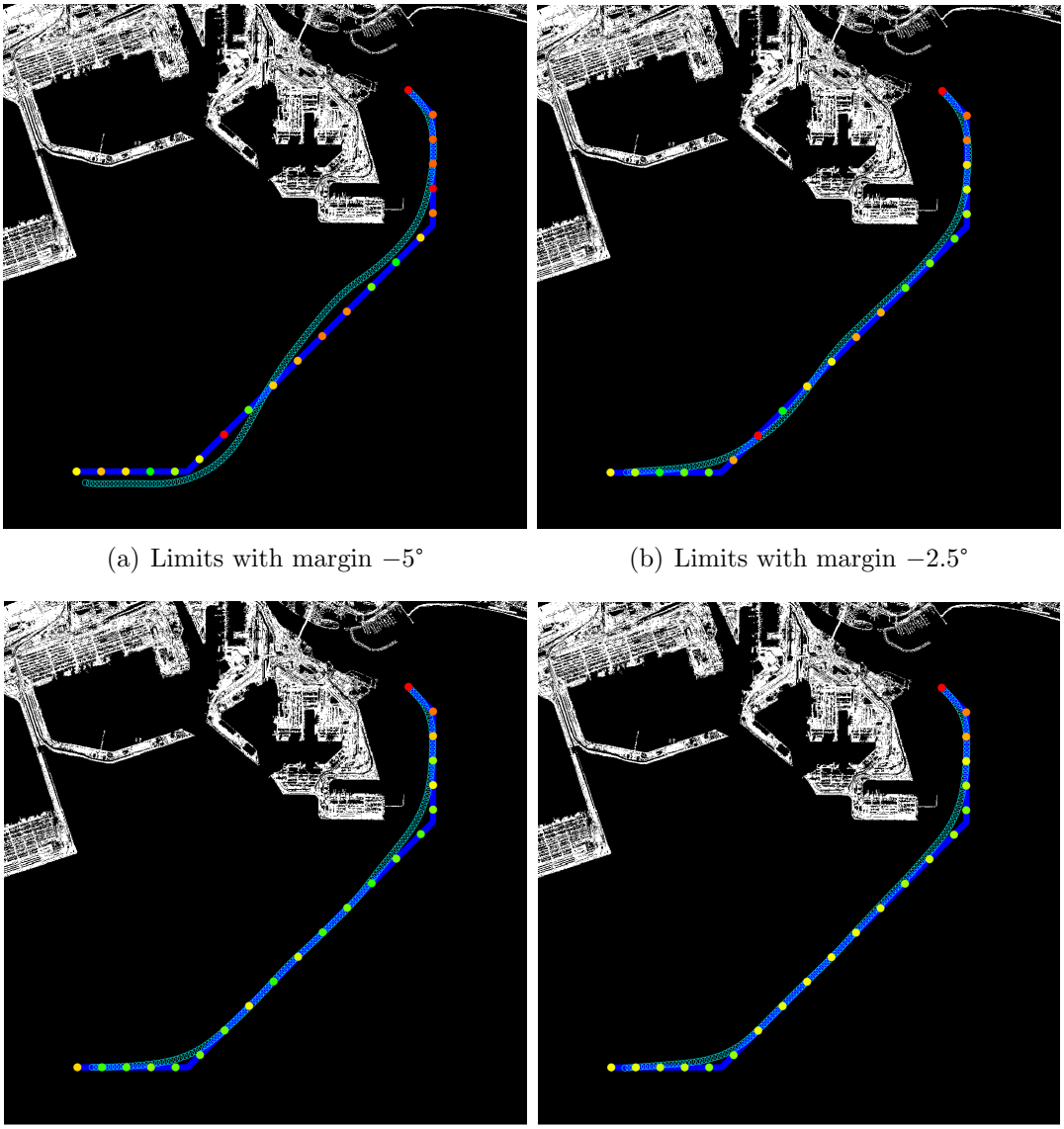
In this example twenty obstacles are injected one per run at evenly spaced intervals throughout the path to compute the performance integrity score for each actuation limit scheme considered. During each run, the own ship attempts to avoid the dynamic obstacle by following the avoidance path circumnavigating the bounding circle around the obstacles. The physical limits serve as a baseline by which to judge the actuation limit schemes under study. A score below the physical limit's score is indicative of a loss of performance integrity.

Table 5.3 contains the performance integrity scores for the limit schemes considered in this demonstration. In general, we expect only negative limit margins to have any loss of performance integrity. This is reflected in the results as a loss of performance integrity is only seen at a limit margin of -5°.

Figure 5.9 is an illustration of the CPA associated with each obstacle inserted along the nominal path. Each obstacle is colored on the spectrum between red, yellow, and green, where red indicates a collision and green indicates a performance integrity score of ≥ 1. Generally obstacles inserted close to the start state will result in a collision or low CPA as the CPS has no or little time to react. Unless performance integrity is compromised as in the case of negative limit margins, performance integrity is expected to be very similar to the physical limit score.

Table 5.3: Performance integrity scores for select actuation limits. Note that performance integrity generally only declines if the margin is negative. Negative margins impede on the required operational states of the controller and thus impose performance penalties. Zero and positive margins do not generally incur performance penalties on controller behavior.

Update Period [s]	**Margin** [°]	**Performance Integrity Score (higher is better)**
Physical	Physical	1
100	-5	0.85
100	-2.5	1
100	0	1
100	5	1
100	10	1
100	20	1



(a) Limits with margin -5°

(b) Limits with margin -2.5°

(c) Limits with margin 0°

(d) Physical Limits

Figure 5.9: Performance integrity plots varying limit margin. Dots mark the locations of inserted dynamic obstacles. Dot color ranges from red to greed corresponding to the performance integrity scores associated with that obstacle. Red indicates the minimum score of 0 while green indicates the maximum score of 1. Note the actuation limits with a margin of -5° suffer from poor performance integrity as these limits egregiously impede required controller behavior.

Chapter 6: Experimental Results and Discussion

This chapter opens by presenting the mean scores for a variety of actuation limit schemes over a range of static obstacle maps. A discussion of these scores follows which details the trade-offs of constraining an attacker in terms of performance integrity and bandwidth utilization. The general applicability of the framework is then discussed and demonstrated by example. The chapter concludes by discussing ongoing work to integrate the proposed framework into a real world CPS testbed.

6.1 Discussion of Results

In this section the aggregated data is presented and discussed. Four maps with one hundred paths each were evaluated with artificial limit schemes with update rates ranging from 50 s to 400 s and margins of -5° to 20°. Testing multiple maps with numerous paths provides support that the properties of actuation limits hold across different static obstacle map topologies. Figure 6.1 contains the mean attack resilience scores. Figure 6.2 contains the mean performance integrity scores. Figure 6.3 contains the mean number of transmissions per path.

The results indicate a general trend that actuation limits are a useful tool in constraining attackers if the limits do not impose on the nominal controller behavior

and are not too broad. When the actuation limits are at a margin of 0° the actuation limits display the greatest ability to constrain the attacker while not compromising performance. This is supported by the performance integrity metric results which are 1 for margins greater than or equal to 0° but indicate a loss of performance when less than 0°. If the limits impose on the nominal controller behavior as in the case of negative margins, then the CPS deviates from the nominal even when not under attack. Then, in the event of an attack, the attacker can amplify the pre-existing deviation to cause catastrophic deviations from the nominal path. Poor attack resilience is exhibited at high margins as well but for a different reason. With a high margin, the attacker has an increased capability to influence the system as the attacker has access to a greater number of states with which to drive the CPS to a catastrophic state.

Another observation is that the the update period of the limits plays an important role in constraining attackers. A smaller update period allows the limits to better conform to the shape of the controller output envelope. Taking the shape of the envelope allows the limits to eliminate extraneous operation states and thus mitigate states the attacker can exploit. However, in the case of negative margins smaller update periods incurred a greater loss of performance. This is likely due to larger update periods generally containing a larger range of controller values and therefore usually setting higher limits.

The amalgam actuation limit scheme is also shown to have efficient bandwidth utilization without a significant loss of attack resilience or performance integrity. The amalgam scheme scores could be further improved by using an algorithm that

is not greedy in its scope. Since the amalgam algorithm only considers the current update period, the algorithm may miss important controller interactions that deleteriously affect the operation of the CPS. Global optimization techniques would likely improve upon the performance of the greedy algorithm.

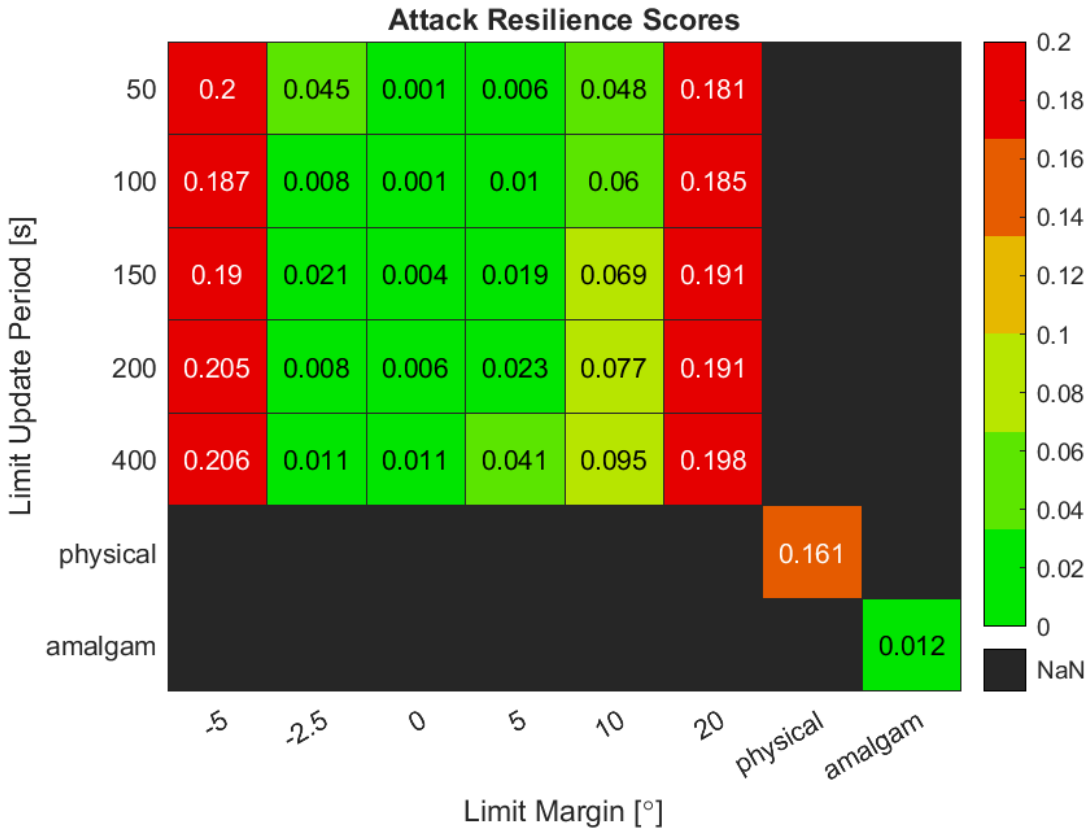


Figure 6.1: Mean attack resilience score over four maps with 100 runs per map. Smaller update periods and limit margins around 0° exhibit the best scores. Margins at the extremes exhibit the poorest scores.

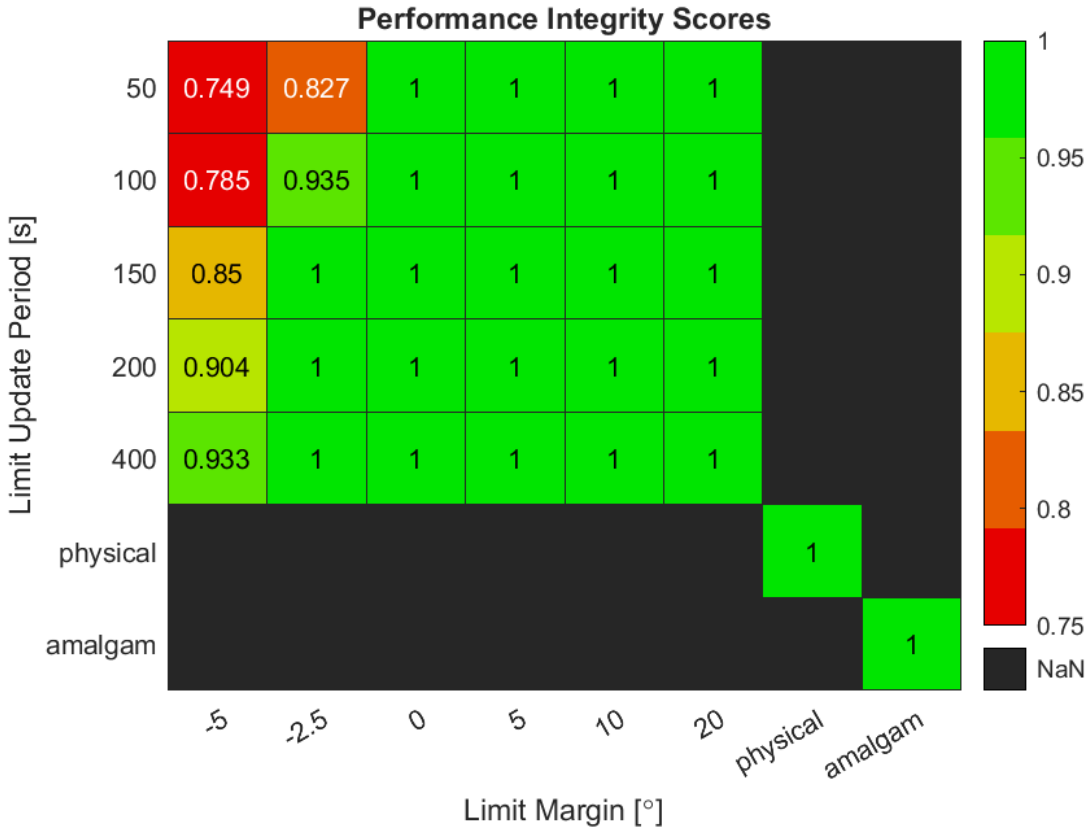


Figure 6.2: Mean performance integrity score over four maps with 100 runs per map. Performance integrity is lost as nominal controller behavior is compromised as in the case of negative margins.

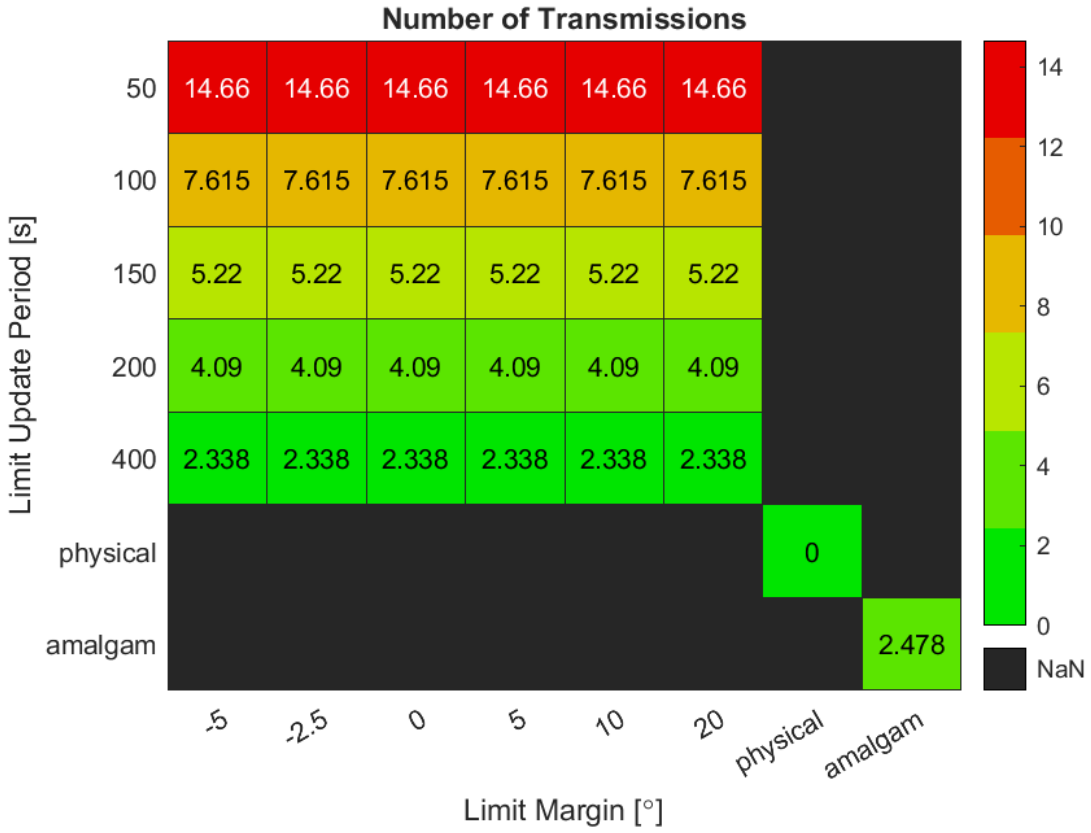


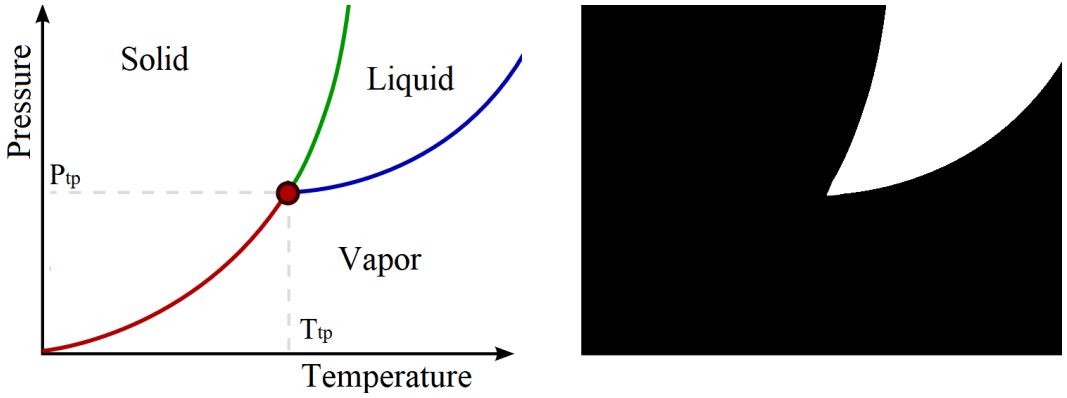
Figure 6.3: Mean bandwidth utilization score over four maps with 100 runs per map. As expected, higher update periods are more bandwidth efficient. Note that the amalgam limits take on a bandwidth utilization score similar to the highest period but slightly higher. The amalgam limit bandwidth use is slightly higher as the amalgamation algorithm attempts to increase the bandwidth in dangerous states.

6.2 General Applicability of Framework

The actuation limit evaluation framework detailed in this research is applicable to many control CPS. Control CPS follow a nominal state trajectory or path that may have dangerous nearby states an attacker can use to harm the system. This framework can be used to show if the imposition of actuation limits are useful in constraining an attacker, if the actuation limits compromise the performance

integrity of the system, and the network bandwidth required to implement actuation limits.

Consider the dehydration process of freeze-drying in which water is sublimated from material by controlling the pressure and temperature. Freeze-drying follows the process of freezing the material to yield ice, reducing pressure, and then sublimating the ice [27]. Sublimation occurs when the water changes its state directly from solid to gas. During the sublimation process the liquid state of water is undesirable. The phase diagram for water cab be used to determine the pressure and temperature state requirements such that the water remains either solid or vapor. In a similar process to extracting land masses from the coastal NOAA satellite imagery, the desirable states can be extracted from the phase diagram shown in Figure 6.4(a) to yield a binary static obstacle map such as Figure 6.4(b). Given the dynamics of the pressure and temperature system, the actuation limits framework can analyze how an attacker may drive the CPS to undesirable states while following the state path from start (solid) to end (vapor) state, the effectiveness of actuation limits in mitigating an attacker's influence, if the actuation limits compromise performance integrity, and the amount of network bandwidth required.



(a) Pressure-Temperature phase diagram (b) Phase diagram as binary map

Figure 6.4: Dehydration process water phase diagram transformed into binary static obstacle map. Figure 6.4(a) illustrates the phase of water as a function of pressure and temperature. The phase diagram can be converted into a binary static obstacle map as shown in Figure 6.4(b) for use in the framework to evaluate actuation limits on the process of sublimation. The liquid phase is undesirable in sublimation and thus marked as an obstacle in the static obstacle map.

6.3 Integration of Framework into CPS Testbed

The actuation limit schemes discussed in this research can be readily integrated into real CPS as a lightweight, transparent hardware solution. The limit schemes are well suited for implementation in the lowest level of a hierarchical CPS near the actuators and plant. The implementation of a hardware shim as middleware to implement actuation limits between current controllers and actuators eases integration with existing CPS hardware. Higher levels in the CPS hierarchy can formulate the actuation limit values and periodically transmit cryptographically authenticated updates over a low bandwidth secondary channel to the shim. Cryptographic authentication using HMAC is readily accelerated in hardware. Device attestation may be used to ensure the integrity of the shim software on boot. The required

shim operating system can be minimal or nonexistent in dedicated hardware implementations, cutting back on exploitable operating system features and requiring minimal patch maintenance.

Work is currently ongoing to integrate a secure hardware shim into a CPS testbed to verify the results obtained by simulation. The CPS testbed is a model ship controlled by an array of microcontrollers interfaced with a high level MATLAB planning software module and user interface [28]. The hardware shim is planned to be a system on a chip (SoC) including a field-programmable gate array (FPGA) implementing an actuation limit cascade state machine, an HMAC authentication module for hardware accelerated cryptographic authentication of actuation limits, and a trusted platform module (TPM) to provide remote attestation. The shim is to be placed between the rudder controller and rudder actuator. The shim shall attest itself via the TPM, and subsequently receive and authenticate via HMAC actuation limits. Figure 6.5 illustrates the overall layout of the CPS testbed. Figure 6.6 illustrates the secure communication channels and hardware components of the hardware shim. Figure 6.7 shows select physical components of the testbed.

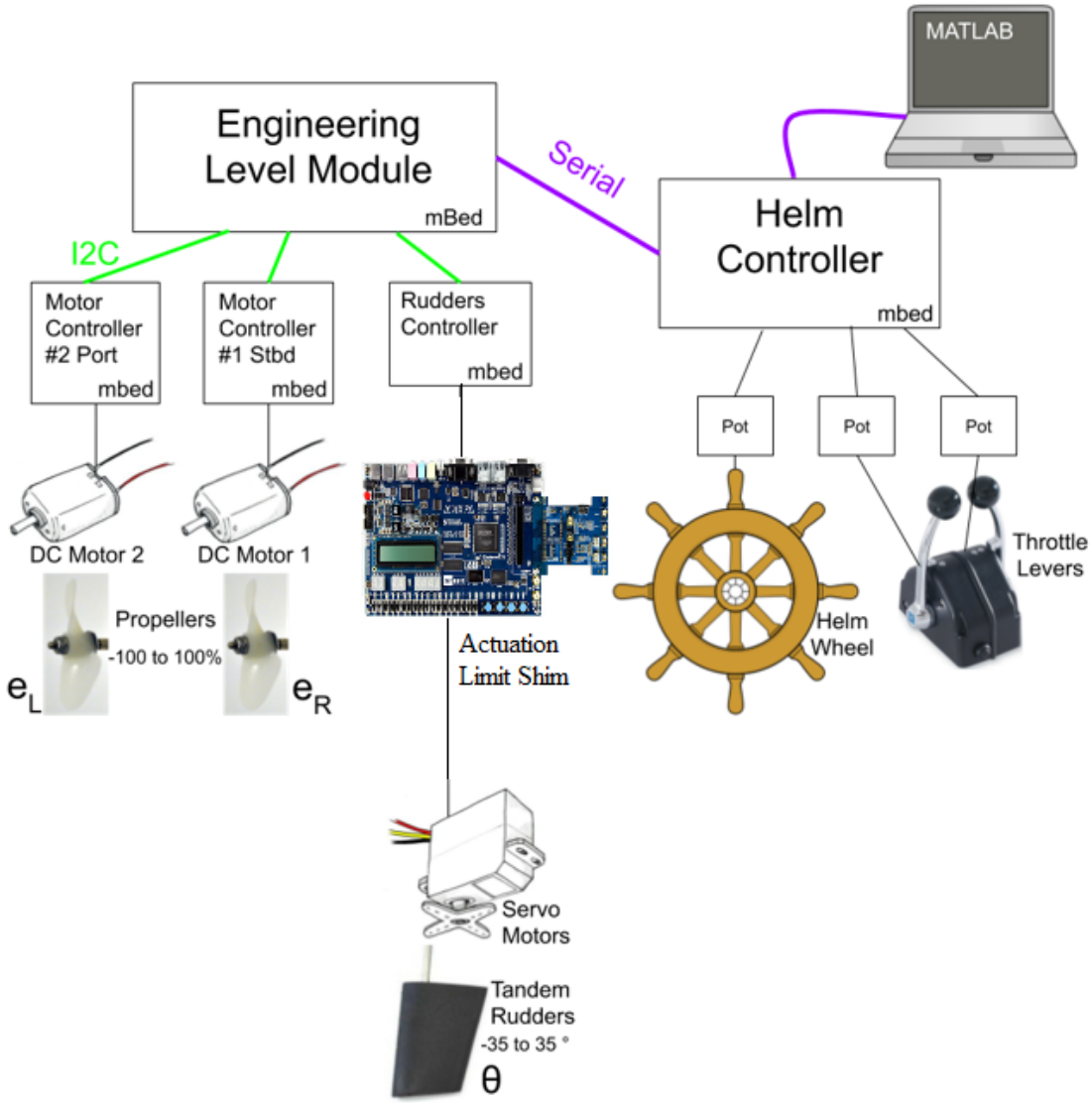


Figure 6.5: Diagram of CPS testbed adapted from [28] augmented with actuation limit shim. The four actuators in the system include two propellers and two tandem rudders. The actuation shim augment would intercept the rudder control signal and apply actuation limits received from the MATLAB actuation limit planner.

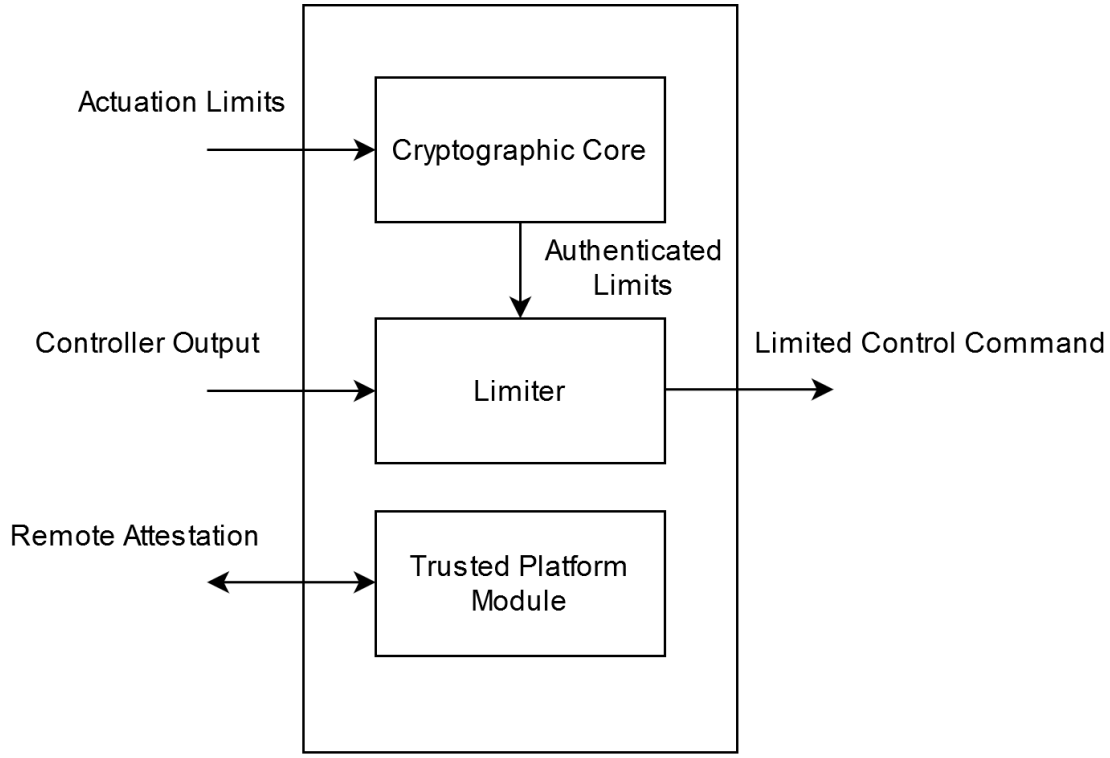


Figure 6.6: Diagram of actuation limit shim with authentication hardware modules. Before actuation limits are sent, the actuation limit planner verifies that the shim is running trusted software via remote attestation assisted by shim's TPM. Subsequently, actuation limits are sent with an HMAC. The shim verifies that the actuation limits are authentic by verifying the HMAC. The shim then imposes the authenticated actuation limits on the controller.

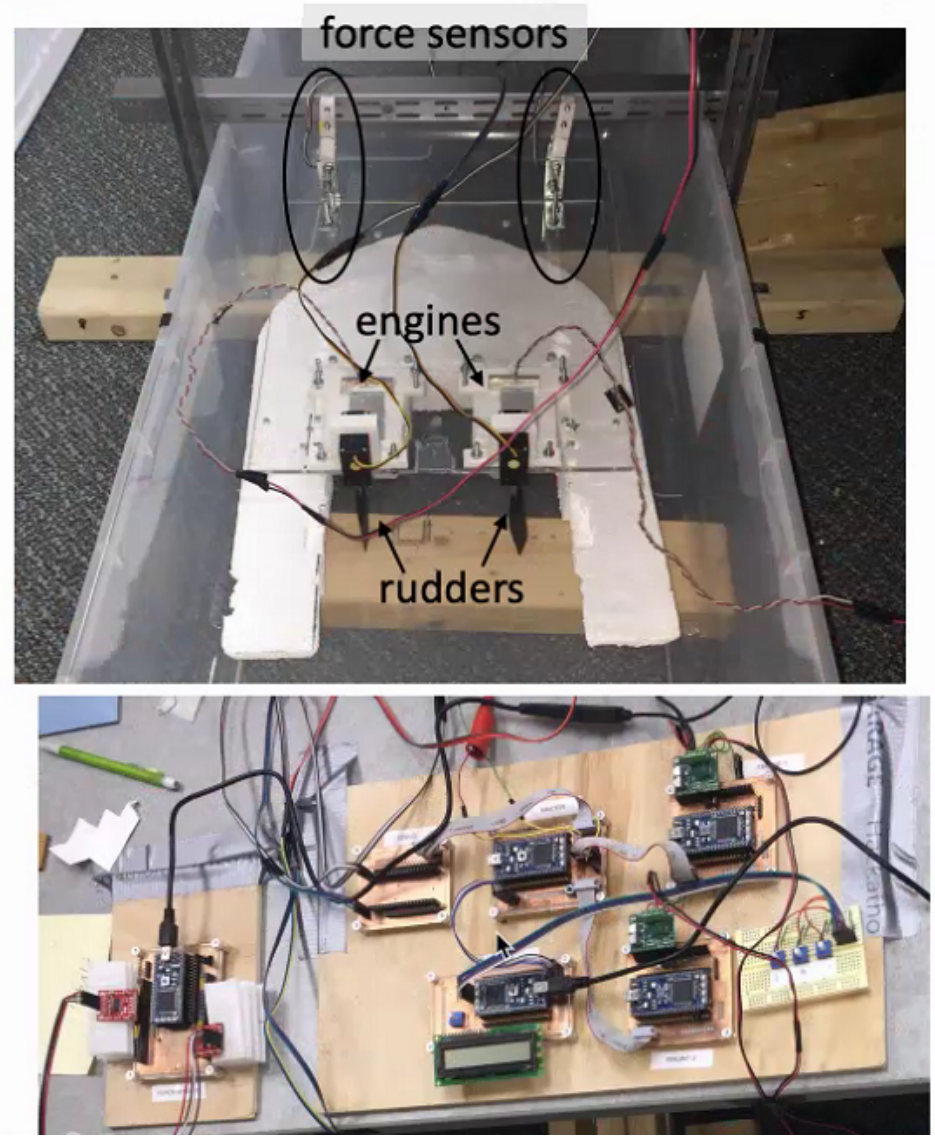


Figure 6.7: Picture of select components of the CPS Testbed adapted from [28]. The upper image shows the force sensors, engines, and tandem rudders mounted on the vessel frame. The lower image shows the helm, engineering, and actuator controllers.

6.4 Future Work

Future work includes design and deployment of a hardware shim which supports remote attestation and implementation of authenticated actuation limits for testing in real world CPS. Further analysis of the attack resilience and performance

integrity scores can be done in the context of identifying defining characteristics of dangerous states. Processing of the actuation limit scores with neural networks and machine learning techniques may yield useful results in quickly identifying dangerous paths or dangerous portions of paths. Global optimization of actuation limits to yield improved metric scores is also an avenue for future research.

6.5 Conclusion

Actuation limits are limitations imposed on a CPS to mitigate dangerous actuation behaviors. Actuation limits, while constraining an attacker, may introduce performance penalties. In this research, we introduce a framework for evaluating the benefits and costs associated with imposing actuation limits on a CPS. Metrics of attack resilience, performance integrity, and bandwidth utilization are devised to capture the impact of applying actuation limits to CPS. Actuation limits imposed on a simulated CPS in the form of a Mariner vessel navigating static obstacle maps is scored. Analysis of the actuation limit scores reveals that actuation limits can be effective in constraining cyber-physical attacks at acceptable performance and bandwidth costs. An algorithm is also introduced that combines actuation limit schemes to trade-off performance integrity and bandwidth usage in exchange for attack resilience when the CPS is in a dangerous state.

Bibliography

[1] B. Croteau, *Attack-Resilient Cyber-Physical Systems*. PhD thesis, University of Maryland, Baltimore County, 2020.

[2] C. Greer, M. Burns, D. Wollman, and E. Griffor, "Cyber-physical systems and internet of things," Tech. Rep. NIST Special Publication 1900-202, National Institute of Standards and Technology, Gaithersburg, Maryland, 2019.

[3] I. Horvath and B. Gerritsen, "Cyber-physical systems: Concepts, technologies and implementation principles," 05 2012.

[4] M. Blanke, "Fault tolerant control systems," 01 1999.

[5] R. Langner, "Stuxnet: Dissecting a cyberwarfare weapon," *IEEE Security Privacy*, vol. 9, no. 3, pp. 49–51, 2011.

[6] E. I. Sharing and A. Center, "Analysis of the cyber attack on the ukrainian power grid," tech. rep., 2016.

[7] C. Biesecker, "Dhs led team demonstrates that commercial aircraft can be remotely hacked," Nov. 2017.

[8] Y. Kim, V. Kolesnikov, and M. Thottan, "Resilient end-to-end message protection for large-scale cyber-physical system communications," in *2012 IEEE Third International Conference on Smart Grid Communications (SmartGridComm)*, pp. 193–198, 2012.

[9] N. Asokan, F. F. Brasser, A. Ibrahim, A.-R. Sadeghi, M. Schunter, G. Tsudik, and C. Wachsmann, "Seda: Scalable embedded device attestation," in *CCS '15*, 2015.

[10] J. Valente, C. Barreto, and A. Cardenas, "Cyber-physical systems attestation," pp. 354–357, 05 2014.

[11] T. Nakai, S. Ichikawa, N. Kobayashi, K. Hata, and K. Sawada, "Whitelisting cyber attack detection according to estimated operational states for cps," in *2019 IEEE 17th International Conference on Industrial Informatics (INDIN)*, vol. 1, pp. 440–445, 2019.

[12] L. Shangguan and S. Gopalswamy, "Health monitoring for cyber physical systems," *IEEE Systems Journal*, vol. 14, no. 1, pp. 1457–1467, 2020.

[13] Y. Mo, S. Weerakkody, and B. Sinopoli, "Physical authentication of control systems: Designing watermarked control inputs to detect counterfeit sensor outputs," *Control Systems, IEEE*, vol. 35, pp. 93–109, 02 2015.

[14] J. Rubio-Hernán, L. De Cicco, and J. García-Alfaro, "Revisiting a watermark-based detection scheme to handle cyber-physical attacks," in *2016 11th International Conference on Availability, Reliability and Security (ARES)*, pp. 21–28, 2016.

[15] C. Kwon and I. Hwang, "Reachability analysis for safety assurance of cyber-physical systems against cyber attacks," *IEEE Transactions on Automatic Control*, vol. 63, no. 7, pp. 2272–2279, 2018.

[16] S. Hadizadeh Kafash, N. Hashemi, C. Murguia, and J. Ruths, "Constraining attackers and enabling operators via actuation limits," in *2018 IEEE Conference on Decision and Control (CDC)*, pp. 4535–4540, 2018.

[17] A. Humayed, J. Lin, F. Li, and B. Luo, "Cyber-physical systems security—a survey," *IEEE Internet of Things Journal*, vol. 4, pp. 1802–1831, Dec. 2017.

[18] J. Wang and G. Yang, "Data-driven methods for stealthy attacks on tcp/ip-based networked control systems equipped with attack detectors," *IEEE Transactions on Cybernetics*, vol. 49, no. 8, pp. 3020–3031, 2019.

[19] A. A. Cardenas, S. Amin, and S. Sastry, "Secure control: Towards survivable cyber-physical systems," in *2008 The 28th International Conference on Distributed Computing Systems Workshops*, pp. 495–500, 2008.

[20] S. H. Kafash, J. Giraldo, C. Murguia, A. A. Cardenas, and J. Ruths, "Constraining attacker capabilities through actuator saturation," in *2018 Annual American Control Conference (ACC)*, pp. 986–991, 2018.

[21] P. Kabiri and M. Chavoshi, "Destructive attacks detection and response system for physical devices in cyber-physical systems," in *2019 International Conference on Cyber Security and Protection of Digital Services (Cyber Security)*, pp. 1–6, 2019.

[22] K. Åström and R. Murray, *Feedback Systems*. Princeton University Press, 2010.

[23] C.-Y. Tzeng and J.-F. Chen, "Fundamental properties of linear ship steering dynamic models," *Journal of Marine Science and Technology*, vol. 7, no. 2, pp. 79–88, 1999.

[24] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley and Sons, 2011.

[25] A. Stentz, "Optimal and efficient path planning for partially-known environments," in *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pp. 3310–3317 vol.4, 1994.

[26] A. Rashid, Z. Yahia, and A. Marhoon, "Path planning with polygonal obstacles avoidance based on the virtual circles of the visible vertices," 03 2017.

[27] P. Fellows, *Food Processing Technology*. Woodhead Publishing, 2009.

[28] B. Croteau, R. Robucci, C. Patel, N. Banerjee, K. Kiriakidis, T. Severson, and E. Rodriguez-Seda, "Alternative actuation paths for ship applications in the presence of cyber-attacks," in *2019 Resilience Week (RWS)*, vol. 1, pp. 91–97, 2019.