Hypergames and Cyber-Physical Security for Control Systems

CRAIG BAKKER, ARNAB BHATTACHARYA, SAMRAT CHATTERJEE, and DRAGUNA L. VRABIE, Pacific Northwest National Laboratory

The identification of the Stuxnet worm in 2010 provided a highly publicized example of a cyber attack that physically damaged an industrial control system. This raised public awareness about the possibility of similar attacks against other industrial targets – including critical infrastructure. In this paper, we use hypergames to analyze how strategic perturbations of sensor readings and calibrated parameters can be used to manipulate a system that employs optimal control. Hypergames form an extension of game theory that enables us to model strategic interactions where the players may have significantly different perceptions of the game(s) they are playing. Past work with hypergames has been limited to relatively simple interactions consisting of a small set of discrete choices for each player, but here, we apply hypergames to larger systems with continuous variables. We find that manipulating constraints can be a more effective attacker strategy than directly manipulating objective function parameters. Moreover, the attacker need not change the underlying system to carry out a successful attack – it may be sufficient to deceive the defender controlling the system. It is possible to scale our approach up to even larger systems, but this will depend on the characteristics of the system in question, and we identify several characteristics that will make those systems amenable to hypergame analysis.

CCS Concepts: • **Computer systems organization → Embedded and cyber-physical systems**; *Reliability*; • **Security and privacy →** *Intrusion/anomaly detection and malware mitigation.*

Additional Key Words and Phrases: Cyber-Physical Security, Optimal Control, Advanced Persistent Threats, Game Theory

ACM Reference Format:
Craig Bakker, Arnab Bhattacharya, Samrat Chatterjee, and Draguna L. Vrabie. 2018. Hypergames and Cyber-Physical Security for Control Systems. *ACM Trans. Graph.* 37, 4, Article 111 (August 2018), 42 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

1.1 Stuxnet and Cyber-Physical Security

The Stuxnet worm was identified in 2010 as a piece of malware that targeted a very specific Industrial Control System (ICS) – namely, uranium enrichment infrastructure [5, 20]. This may not have been the first cyber attack to cause physical damage to an ICS, but it was highly publicized. As such, Stuxnet brought the potential physical consequences of cyber attacks into the public eye. Stuxnet was highly sophisticated. Part of its sophistication lay in its strategy for obtaining access to its targets: it exploited four 0-day vulnerabilities, compromised two digital certificates, and propagated itself through networks and removable devices [5]. Once it reached a control system, it

Authors' address: Craig Bakker, craig.bakker@pnnl.gov; Arnab Bhattacharya, arnab.bhattacharya@pnnl.gov; Samrat Chatterjee, samrat.chatterjee@pnnl.gov; Draguna L. Vrabie, draguna.vrabie@pnnl.gov, Pacific Northwest National Laboratory, 902 Battelle Blvd. Richland, Washington, 99353.

0730-0301/2018/8-ART111 $15.00
https://doi.org/10.1145/1122445.1122456

continued to act stealthily. Stuxnet fed fake data to the ICS to disguise malicious actions [5, 11] and limited its attacks to avoid detection [14]. The goal of Stuxnet was not to cause catastrophic failure but rather to exploit the physical and cyber vulnerabilities inherent in the ICS in a stealthy fashion.

Stuxnet forced analysts to consider the risk associated with these kinds of cyber attacks. If we understand risk as the product of consequence, vulnerability, and threat, we can address each of those components separately. The potential for significant consequence is clear: many industrial processes, including critical infrastructure systems (e.g., the power grid), rely on Supervisory Control and Data Acquisition (SCADA) software and ICSs. These systems are also vulnerable. Updates can be risky because they may cause previously functional systems to produce new errors [14], and even if this is not the case, taking the system in question offline to perform the updates may be difficult or infeasible [20]. There is a tradeoff between security and ease of use, and a knowledge gap between cyber security specialists and control engineers can compound this.

There are two more factors that increase the vulnerability of ICSs to cyber attack. Firstly, industrial systems are often serviced by outside contractors, and the devices (computers, USB drives, etc.) used by those contractors can provide a malware vector that bypasses traditional cyber security measures such as air gaps [5]. Secondly, industry standardization also reduces uncertainty for potential attackers; complexity, heterogeneity, and uncertainty make it more difficult for attackers to design successful attacks. Most of the uncertainty regarding the risk of cyber attacks on ICSs has to do with threat. The old consensus was that these systems were too specialized to attack [14]. Stuxnet, for example, required a great deal of specialized knowledge about the control systems in question [5]. However, Stuxnet showed that these kinds of attacks are possible.

1.2 Hypergames

Game theory is a branch of mathematics that looks at strategic interactions between rational entities. It has seen considerable use in economic [23] and security [27] applications. A fundamental premise of strategic games in game theory is that all of the players are seeing and playing the same game. This is not always true, though. Belief manipulation plays a key role in some strategic interactions. In other cases, not all player objectives may be common knowledge. This necessitates understanding more completely players' perceptions of the game(s) they are playing; one way to model this is through hypergames [3]. Hypergames allow players to play different games and can account for differences in their perceptions of the same game without considering uncertainty probabilistically. For example, one group of players may distinguish between certain actions while another group considers those actions all to be identical. On the other hand, some players may not be aware of the existence of other players in the game (or may not be aware of all of those other players' possible actions). Hypergames essentially enable us to extend the concept of rationality to a bounded information situation. This, in turn, makes it possible for a given player to exploit another player's misperceptions. In analyzing the (potentially) different games that each player is playing, though, we are still able to apply game theoretic concepts and thus build on existing game theory research. We can describe a two-player game as $G_{A,B} = (\mathcal{P}, \mathcal{S}, \mathcal{U})$, where

$$\mathcal{P} = \{A, B\} \tag{1}$$

$$\mathcal{S} = \{S_A, S_B\} \tag{2}$$

$$\mathcal{U} = \{u_A, u_B\} \tag{3}$$

A and B are the players, S_A and S_B are those players' respective action spaces, and $u_A, u_B : S_A \times S_B \rightarrow \mathfrak{R}$ are their respective payoff functions, which provide a partial ordering over $S_A \times S_B$ for each player. We can describe a first level hypergame as

$$H_{A,B}\left(A, B, G_{A,B}\right) = \left\{p\left(A, G_{A,B}\right), p\left(B, G_{A,B}\right)\right\} \tag{4}$$

where $p\left(A, G_{A,B}\right)$ is A's perception of $G_{A,B}$ – A's subjective game. This subjective game could be Bayesian, strategic, or even a hypergame itself, and it encodes the (mis)perceptions of each player. The condition $p\left(A, G_{A,B}\right) \neq p\left(B, G_{A,B}\right)$ could be caused by discrepancies such as $p\left(A, \{A, B\}\right) = \{A\}$, which would indicate that A is not aware of B's presence. We can also describe perceptions about perceptions. For example, $p\left(AB, u_A\right)$ is A's perception of B's perception of A's utility function. In a first level hypergame, the players are not aware of their misperceptions:

$$p\left(A, G_{A,B}\right) \neq p\left(B, G_{A,B}\right) \tag{5}$$

$$p\left(AB, G_{A,B}\right) = p\left(A, G_{A,B}\right) \tag{6}$$

$$p\left(BA, G_{A,B}\right) = p\left(B, G_{A,B}\right) \tag{7}$$

In a second level hypergame, at least one player is aware of the misperceptions. For example, if A is aware of the misperceptions but B is not, we have

$$p\left(AB, G_{A,B}\right) \neq p\left(A, G_{A,B}\right) \tag{8}$$

$$p\left(BA, G_{A,B}\right) = p\left(B, G_{A,B}\right) \tag{9}$$

Player B then plays $p\left(B, G_{A,B}\right)$ while A plays the hypergame

$$H_{A,AB}\left(A, AB, G_{A,B}\right) = \left\{p\left(A, G_{A,B}\right), p\left(AB, G_{A,B}\right)\right\} \tag{10}$$

The overall solution is called a Hyper Nash Equilibrium (HNE). It can be calculated by correctly aggregating the solutions to the players' perceived (hyper)games, which are also referred to as their subjective games. In the hypergame literature, the subjective games are typically strategic finite games, and their solutions are Nash equilibria. However, in principle, the solutions could be equilibria of various kinds (Hyper Nash, Nash, Bayesian, Perfect, etc.), depending on the nature of each subjective (hyper)game. The nature of the base-level subjective game (strategic, Bayesian, etc.) has some effect on the complexity of the overall hypergame formulation and solution, but nested hypergame structures, which correspond to multi-level belief hierarchies and higher level hypergames, tend to drive the problem complexity more strongly.

In a first level hypergame, the HNE is $(\mathbf{x}_A, \mathbf{x}_B)$, where $\mathbf{x}_A$ is A's equilibrium strategy for the game $p\left(A, G_{A,B}\right)$ and $\mathbf{x}_B$ is B's equilibrium strategy for the game $p\left(B, G_{A,B}\right)$. For the second level hypergame described above, $\mathbf{x}_A$ would be A's optimal strategy for $H_{A,AB}\left(A, AB, G_{A,B}\right)$, while $\mathbf{x}_B$ would still be B's equilibrium strategy for $p\left(B, G_{A,B}\right)$. These concepts extend naturally to higher level hypergames and additional players. HNE can share properties possessed by other forms of equilibria; HNE may be unique or mixed, for instance. Some other equilibrium properties may not be as applicable, though. For example, the concept of equilibrium efficiency may not make sense for HNE because of the potential gap (or even inverse correlation) between perceived and actual payoffs. We have not seen any discussions of HNE efficiency in the hypergame literature, however. See Kovach et al. [15] and Gutierrez et al. [9] for hypergame literature reviews.

There is a connection between hypergames and bounded rationality. Bounded rationality is perhaps more commonly associated with approaches such as prospect theory [12] or quantal responses [18], which do not assume strict utility maximization, or with models that assume limited computational ability (e.g., [26]). These aspects of bounded rationality could perhaps be used with

hypergames (e.g., a Quantal HNE), but we have not seen any such work. The focus of hypergame research has instead been on levels of (mis)perceptions and systems that lack common knowledge; this lack could be considered a kind of bounded rationality.

Reflexive control [21], Mirage Equilibria [25], and k-level reasoning [4, 31] have also been applied to systems that may not have common knowledge. Despite some differences in notation and nomenclature, these all incorporate hierarchies of beliefs (e.g., Player 1's beliefs about Player 2's beliefs). However, the first two, along with hypergames, differ somewhat from k-level reasoning with respect to the accuracy of the player perceptions. In k-level reasoning, the focus is on the degree to which one player anticipates another. In principle, this approach does not rule out the possibility that a given player might misperceive the nature of the game (payoff structure, available actions, etc.), but in practice, this is not a key consideration. For hypergames, this *is* a key consideration. The concept of a subjective game (i.e., $p\left(A, G_{A,B}\right)$) is central to hypergame analysis, and belief hierarchies exist to support that; the same is true for reflexive control and Mirage Equilibria.

For example, a key hypergame result is that hypergame equilibrium solutions can be stable under misperceptions [28]. In these cases, each player does what the other players expect – which can happen even when the players' perceptions differ or are erroneous – and thus there is no motivation for players to update their perceptions. This is similar to a conjectural equilibrium [25] in that players do not know what they do not know. In a repeated hypergame context, then, these equilibria are stable, and extending belief hierarchies to higher and higher levels would not necessarily change that. Using the formalism we employed previously, a hypergame equilibrium is stable if $p\left(A, \mathbf{x}_B\right) = \mathbf{x}_B$ and $p\left(B, \mathbf{x}_A\right) = \mathbf{x}_A$, which need not imply that $p\left(A, G_{A,B}\right) = p\left(B, G_{A,B}\right)$.

Hypergames have been used to study water resource management [22], supply chain relationships [8], and cyber attacks [10]. Some research has also looked at connecting hypergames with other branches of game theory. Kanazawa et al. [13] studied an evolutionary version of hypergames. This included calculating evolutionarily stable strategies and defining hypergame replicator dynamics. Sasaki and Kijima [29, 30] showed how hypergames can be reformulated as Bayesian games in some cases. In doing so, though, they identified reasons why it may be advantageous to avoid that reformulation. Firstly, hypergames can provide a simpler and more natural epistemic representation of the game's players; the treatment of unawareness, for example, can be more convincing than in the Bayesian case. Secondly, there are some hypergame solution concepts, such as stability under misperception, that do not map to the Bayesian reformulation. For more discussion of the relationship between Bayesian games and hypergames, see Sasaki and Kijima [29, 30].

The topic of misperception has also led to research into how repeated hypergames can be used to improve or update perceptions [28]. Repeated hypergames offer the possibility of signalling and misperception correction. As in a single-stage strategic game, a single-stage hypergame may not involve signalling. However, in repeated interactions, observing unknown actions from another player, observing unexpected actions from another player, or receiving payoffs that differ from the expected value can all be 'signals' that players can use to recognize and address their misperceptions. Gharesifard and Cortés study this in some detail [7]. House and Cybenko used both hidden Markov models and a maximum entropy approach [10]. Takahashi et al., on the other hand, used a genetic algorithm [32]. Bakker et al. [1] also show how this can be applied in a control systems context. Generally speaking, the hypergame literature is relatively small, and almost all of the examples that we have seen have involved hypergames with a relatively small number of discrete choices; solving for the HNE has involved hand calculations and/or exhaustive enumeration.

1.3 Aim and Motivation

The goal of this paper is to show how hypergames can be used in optimal control where the control system in question is subject to adversarial perturbations and to demonstrate how this analysis

can apply to Stuxnet-like attacks. This research contributes to ongoing work in optimal control by showing how manipulating controller perceptions can function as an attacker strategy; the attacker actually uses the control system against itself. These analyses then highlight weaknesses in the control system – weaknesses that are vulnerable to attack even if they might not be vulnerable to random events. This research also advances hypergame research in two ways. Firstly, it brings hypergames to bear on a new application area (i.e., optimal control) – one rather different than the examples in previous papers. Secondly, it applies hypergame concepts to systems of significantly greater complexity than previous hypergame research has used. The examples in this paper have continuous variables, and the second example is a discrete-time optimal control problem with time-varying variables. Both problems, moreover, require using numerical optimization methods to find hypergame equilibria. Taking hypergames to this level of complexity makes the hypergame concept more viable as a tool for analyzing real systems and not just toy problems.

This kind of investigation is highly relevant to addressing Stuxnet-like attacks from a control perspective. The means by which an attacker might gain the network access and system knowledge necessary to carry out such an attack are not trivial, but they are not the focus of our analysis here. Rather, we assume the existence of an attacker with this kind of knowledge and access – an Advanced Persistent Threat (APT) – and we then inquire about the potential outcomes. ICSs provide examples of (potentially high-impact) cyber-physical systems where control provides the connection between the 'cyber' and 'physical' components. The idea behind this research, then, is not to replace traditional cyber security methods but rather to recognize that control systems can be used to provide another layer of robustness to attack if designed to do so and that the physical weaknesses accessible through cyber means can be analyzed by looking at the control model.

2 STATIC PROBLEM FORMULATION

To demonstrate some of the concepts of this paper, we consider a static optimization problem constrained within an operating envelope, which is represented as an inequality constraint:

$$\min_{\mathbf{u}} J(\mathbf{u}, \boldsymbol{\theta}) \tag{11}$$

$$\mathbf{g}(\mathbf{u}, \mathbf{c}) \leq \mathbf{0} \tag{12}$$

where $\mathbf{u}$ is the vector of decision variables, $\boldsymbol{\theta}$ is the vector of objective function parameters, and $\mathbf{c}$ is the vector of operating envelope parameters. Note that $\mathbf{g}$ may be a vector of constraint equations g_l, $l = 1, 2, \ldots$, in which case (12) is equivalent to $g_l(\mathbf{u}, \mathbf{c}) \leq 0 \; \forall \; l$.

2.1 Objective Function Manipulation

Here, we will consider a situation where the attacker can manipulate the defender's observation of objective function parameters; $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta} + \Delta\boldsymbol{\theta}$, where the vector $\hat{\boldsymbol{\theta}}$ denotes the quantities that the defender observes. The attacker chooses a deterministic strategy over possible $\Delta\boldsymbol{\theta}$ values, and the defender chooses a deterministic strategy over $\mathbf{u}$ values. The attacker optimization is then

$$\max_{\Delta\boldsymbol{\theta}} J(\hat{\mathbf{u}}^*, \boldsymbol{\theta}) \tag{13}$$

$$\frac{1}{2} \|\Delta\boldsymbol{\theta}\|^2 \leq \delta_{\theta,max} \tag{14}$$

$$\hat{\mathbf{u}}^* = \arg\min_{\hat{\mathbf{u}}} \left(J\left(\hat{\mathbf{u}}, \hat{\boldsymbol{\theta}}\right) : g(\hat{\mathbf{u}}, \mathbf{c}) \leq \mathbf{0} \right) \tag{15}$$

where (15) describe what the attacker expects the defender's optimization to be and (14) is a constraint on the attacker's manipulations, which is a reasonable assumption in a context of limited attack budgets or when attack detection mechanisms are present in the system. This constitutes a second level hypergame. If A represents the attacker and D represents the defender, we have

$$p(D, \boldsymbol{\theta}) = \hat{\boldsymbol{\theta}} \neq \boldsymbol{\theta} = p(A, \boldsymbol{\theta}) \tag{16}$$

$$p(D, \{A, D\}) = \{D\} = p(AD, \{A, D\}) \tag{17}$$

If the defender knows of the attacker, this leads to a higher level hypergame, where

$$p(DAD, \{A, D\}) = p(AD, \{A, D\}) = \{D\} \tag{18}$$

The defender's optimization is

$$\min_{\mathbf{u}} J\left(\mathbf{u}, \hat{\boldsymbol{\theta}} - \Delta\boldsymbol{\theta}\right) \tag{19}$$

$$\mathbf{g}(\mathbf{u}, \mathbf{c}) \leq \mathbf{0} \tag{20}$$

The true $\boldsymbol{\theta}$ values are unknown to the defender, but the defender calculates the $\Delta\boldsymbol{\theta}$ values by solving what is believed to be the attacker's problem: (13)-(15).

$$\max_{\Delta\boldsymbol{\theta}} J(\hat{\mathbf{u}}^*, \boldsymbol{\theta}) \tag{21}$$

$$\frac{1}{2}\|\Delta\boldsymbol{\theta}\|^2 \leq \delta_{\theta,max} \tag{22}$$

$$\hat{\mathbf{u}}^* = \arg\min_{\hat{\mathbf{u}}} \left(J\left(\hat{\mathbf{u}}, \hat{\boldsymbol{\theta}}\right) : \mathbf{g}(\hat{\mathbf{u}}, \mathbf{c}) \leq \mathbf{0}\right) \tag{23}$$

Given that the defender only knows $\hat{\boldsymbol{\theta}}$, not $\boldsymbol{\theta}$, solving the attacker's problem to determine $\Delta\boldsymbol{\theta}$ will require using $\boldsymbol{\theta} = \hat{\boldsymbol{\theta}} - \Delta\boldsymbol{\theta}$. As a further extension, we consider the scenario where the attacker manipulates the defender's perceptions of $\boldsymbol{\theta}$, the defender knows that the attacker is doing this, and the attacker knows that the defender is anticipating the attacker's perturbations. We refer to this as a 'double-bluff' manipulation here and in the rest of the paper. This problem leads us to a multi-level optimization problem:

$$\max_{\Delta\boldsymbol{\theta}} J(\mathbf{u}^*, \boldsymbol{\theta}) \tag{24}$$

$$\frac{1}{2}\|\Delta\boldsymbol{\theta}\|^2 \leq \delta_{\theta,max} \tag{25}$$

$$\mathbf{u}^* = \arg\min_{\mathbf{u}} \left(J\left(\mathbf{u}, \tilde{\boldsymbol{\theta}}\right) : \mathbf{g}(\mathbf{u}, \mathbf{c}) \leq \mathbf{0}\right) \tag{26}$$

subject to

$$\max_{\Delta\hat{\boldsymbol{\theta}}} J\left(\hat{\mathbf{u}}^*, \tilde{\boldsymbol{\theta}}\right) \tag{27}$$

$$\frac{1}{2}\left\|\Delta\hat{\boldsymbol{\theta}}\right\|^2 \leq \delta_{\theta,max} \tag{28}$$

$$\hat{\mathbf{u}}^* = \arg\min_{\hat{\mathbf{u}}} \left(J\left(\hat{\mathbf{u}}, \tilde{\boldsymbol{\theta}} + \Delta\hat{\boldsymbol{\theta}}\right) : \mathbf{g}(\hat{\mathbf{u}}, \mathbf{c}) \leq \mathbf{0}\right) \tag{29}$$

where $p(D,\boldsymbol{\theta}) = \tilde{\boldsymbol{\theta}} = \hat{\boldsymbol{\theta}} - \Delta\hat{\boldsymbol{\theta}}$ is the defender's estimate of the true value of $\boldsymbol{\theta}$. These nested optimizations and the use of argmin, here and in the rest of the paper, are a shorthand way to represent best responses – the solutions to different subjective games and nested hypergame levels. Note that the defender's perceived objective function value may differ from the true value in some cases. For comparison, we can also model the attacker manipulating the true value of $\boldsymbol{\theta}$:

$$\max_{\Delta\boldsymbol{\theta}} \min_{\mathbf{u}} J(\mathbf{u}, \boldsymbol{\theta} + \Delta\boldsymbol{\theta}) \tag{30}$$

$$\|\Delta\boldsymbol{\theta}\|^2 \leq \delta_{\theta,max} \tag{31}$$

$$\mathbf{g}(\mathbf{u}, \mathbf{c}) \leq \mathbf{0} \tag{32}$$

Here, there are no misperceptions; the situation is simply a zero-sum game, not a hypergame.

2.2 Constraint Manipulation

The previous section had the attacker manipulating objective function parameters. This entails a difference between manipulating the true values and the defender's perceptions. Manipulating constraints is different. If the attacker alters the constraint to be more restrictive, manipulating the real constraint or the defender's perceptions leads to the same result in either case (assuming that the defender abides by the constraint); the perceived cost is also the true cost in both cases. If the attacker alters the constraint to be less restrictive, the results are less clear. If the attacker manipulates the defender perception, the control process may hit a physical limit and/or damage the system trying to reach an infeasible state. This could be modelled with a large penalty for violations of the true constraint. Relaxing the true constraint may be impossible if the constraint is a physical limitation of the system. For this section, we specify that the attacker can manipulate the defender's perception of parameters in the constraint ($\hat{\mathbf{c}} = \mathbf{c} + \Delta\mathbf{c}$ are the quantities that the defender perceives). The attacker chooses a deterministic strategy over $\Delta\mathbf{c}$ values while the defender's strategy set remains the same. As before, attacker perturbations are subject to a constraint:

$$\frac{1}{2}\|\Delta\mathbf{c}\|^2 \leq \delta_{c,max} \tag{33}$$

2.2.1 Maximizing Cost. Manipulating the defender's perceptions to maximize cost produces a series of multi-level optimization problems, corresponding to second or higher level hypergames, like those described previously. If the attacker is deceiving an unsuspecting defender, we have

$$\max_{\Delta\mathbf{c}} J(\mathbf{u}^*, \boldsymbol{\theta}) \tag{34}$$

$$\frac{1}{2}\|\Delta\mathbf{c}\|^2 \leq \delta_{c,max} \tag{35}$$

$$\mathbf{u}^* = \arg\min_{\mathbf{u}} (J(\mathbf{u}, \boldsymbol{\theta}) : \mathbf{g}(\mathbf{u}, \hat{\mathbf{c}}) \leq \mathbf{0}) \tag{36}$$

If the defender is aware of the attack, we have

$$\min_{\mathbf{u}} J(\mathbf{u}, \boldsymbol{\theta}) \tag{37}$$

$$\mathbf{g}(\mathbf{u}, \hat{\mathbf{c}} - \Delta\mathbf{c}) \leq \mathbf{0} \tag{38}$$

subject to

$$\max_{\Delta \mathbf{c}} J\left(\hat{\mathbf{u}}^*, \boldsymbol{\theta}\right) \tag{39}$$

$$\frac{1}{2}\left\|\Delta \mathbf{c}\right\|^2 \leq \delta_{c,max} \tag{40}$$

$$\hat{\mathbf{u}}^* = \arg\min_{\hat{\mathbf{u}}} \left(J\left(\hat{\mathbf{u}}, \boldsymbol{\theta}\right) : \mathbf{g}\left(\hat{\mathbf{u}}, \hat{\mathbf{c}}\right) \leq \mathbf{0}\right) \tag{41}$$

In a situation analogous to that described in the previous section, the defender only knows $\hat{\mathbf{c}}$, not $\mathbf{c}$, so solving the attacker's problem to determine $\Delta \mathbf{c}$ will require using $\mathbf{c} = \hat{\mathbf{c}} - \Delta \mathbf{c}$. If the attacker is aware that the defender is anticipating an attack, the resulting problem is

$$\max_{\Delta \mathbf{c}} J\left(\mathbf{u}, \boldsymbol{\theta}\right) \tag{42}$$

$$\frac{1}{2}\left\|\Delta \mathbf{c}\right\|^2 \leq \delta_{c,max} \tag{43}$$

$$\mathbf{u}^* = \arg\min_{\mathbf{u}} \left(J\left(\mathbf{u}, \boldsymbol{\theta}\right) : \mathbf{g}\left(\mathbf{u}, \tilde{\mathbf{c}}\right) \leq \mathbf{0}\right) \tag{44}$$

subject to

$$\max_{\Delta \hat{\mathbf{c}}} J\left(\hat{\mathbf{u}}^*, \boldsymbol{\theta}\right) \tag{45}$$

$$\frac{1}{2}\left\|\Delta \hat{\mathbf{c}}\right\|^2 \leq \delta_{c,max} \tag{46}$$

$$\hat{\mathbf{u}}^* = \arg\min_{\hat{\mathbf{u}}} \left(J\left(\hat{\mathbf{u}}, \boldsymbol{\theta}\right) : \mathbf{g}\left(\hat{\mathbf{u}}, \tilde{\mathbf{c}} + \Delta \hat{\mathbf{c}}\right) \leq \mathbf{0}\right) \tag{47}$$

where $p\left(D, \mathbf{c}\right) = \tilde{\mathbf{c}} = \hat{\mathbf{c}} - \Delta \mathbf{c}$ is the defender's estimate of the true value of $\mathbf{c}$.

2.2.2 Breaking the System. The attacker could try to cause the defender to deviate maximally from the operating envelope constraint to cause a catastrophic failure. We refer to this as attempting to break the system. If the attacker is deceiving an unsuspecting defender, we have

$$\max_{\Delta \mathbf{c}} \boldsymbol{\gamma}^T \mathbf{g}\left(\mathbf{u}^*, \mathbf{c}\right) \tag{48}$$

$$\frac{1}{2}\left\|\Delta \mathbf{c}\right\|^2 \leq \delta_{c,max} \tag{49}$$

$$\mathbf{u}^* = \arg\min_{\mathbf{u}} \left(J\left(\mathbf{u}, \boldsymbol{\theta}\right) : \mathbf{g}\left(\mathbf{u}, \hat{\mathbf{c}}\right) \leq \mathbf{0}\right) \tag{50}$$

where $\boldsymbol{\gamma}$ weights the sum of $\mathbf{g}$'s components. If the defender is aware of the attack, we have

$$\min_{\mathbf{u}} J\left(\mathbf{u}, \boldsymbol{\theta}\right) \tag{51}$$

$$\mathbf{g}\left(\mathbf{u}, \hat{\mathbf{c}} - \Delta \mathbf{c}\right) \leq \mathbf{0} \tag{52}$$

subject to

$$\max_{\Delta \mathbf{c}} \boldsymbol{\gamma}^T \mathbf{g}\left(\hat{\mathbf{u}}^*, \mathbf{c}\right) \tag{53}$$

$$\frac{1}{2}\left\|\Delta \mathbf{c}\right\|^2 \leq \delta_{c,max} \tag{54}$$

$$\hat{\mathbf{u}}^* = \arg\min_{\hat{\mathbf{u}}} \left(J\left(\hat{\mathbf{u}}, \boldsymbol{\theta}\right) : \mathbf{g}\left(\hat{\mathbf{u}}, \hat{\mathbf{c}}\right) \leq \mathbf{0} \right) \tag{55}$$

If the attacker is aware that the defender is anticipating an attack, the resulting problem is

$$\max_{\Delta \mathbf{c}} \boldsymbol{\gamma}^T \mathbf{g}\left(\mathbf{u}^*, \mathbf{c}\right) \tag{56}$$

$$\frac{1}{2} \left\| \Delta \mathbf{c} \right\|^2 \leq \delta_{c,max} \tag{57}$$

$$\mathbf{u}^* = \arg\min_{\mathbf{u}} \left(J\left(\mathbf{u}, \boldsymbol{\theta}\right) : \mathbf{g}\left(\mathbf{u}, \tilde{\mathbf{c}}\right) \leq \mathbf{0} \right) \tag{58}$$

subject to

$$\max_{\Delta \hat{\mathbf{c}}} \boldsymbol{\gamma}^T \mathbf{g}\left(\hat{\mathbf{u}}^*, \mathbf{c}\right) \tag{59}$$

$$\frac{1}{2} \left\| \Delta \hat{\mathbf{c}} \right\|^2 \leq \delta_{c,max} \tag{60}$$

$$\hat{\mathbf{u}}^* = \arg\min_{\hat{\mathbf{u}}} \left(J\left(\hat{\mathbf{u}}, \boldsymbol{\theta}\right) : \mathbf{g}\left(\hat{\mathbf{u}}, \tilde{\mathbf{c}} + \Delta \hat{\mathbf{c}}\right) \leq \mathbf{0} \right) \tag{61}$$

where $\tilde{\boldsymbol{\theta}}$ is defined as before. There are various other possibilities in the same vein involving asymmetric information or false beliefs.

2.3 Analytical Results

2.3.1 Objective Function Perturbations. In this section, we show that the defender can be robust with respect to manipulated perceptions of θ. Let us assume that $g\left(u, c\right)$ is convex for $c \geq 0$ and

$$J\left(\mathbf{u}, \boldsymbol{\theta}\right) = \sum_k \theta_k f_k\left(\mathbf{u}\right) = \boldsymbol{\theta}^T \mathbf{f}\left(\mathbf{u}\right) \tag{62}$$

where each $f_k\left(u\right)$ is convex. The optimization is convex for $\theta \geq 0$, and the optimality conditions

$$\sum_k \frac{\partial f_k}{\partial \mathbf{u}} \theta_k + \sum_l \lambda_l \frac{\partial g_l}{\partial \mathbf{u}} = \boldsymbol{\theta}^T \frac{\partial \mathbf{f}}{\partial \mathbf{u}} + \boldsymbol{\lambda}^T \frac{\partial \mathbf{g}}{\partial \mathbf{u}} = \mathbf{0} \tag{63}$$

$$0 \leq \lambda_l \perp g_l\left(\mathbf{u}, \mathbf{c}\right) \leq \mathbf{0} \; \forall \; l \tag{64}$$

are both necessary and sufficient; $\boldsymbol{\lambda}$ is the vector of Kuhn-Tucker multipliers. Let us also define

$$R\left(\mathbf{u}\right) = \left\{ \left. \frac{\partial f_k}{\partial \mathbf{u}} \right|_{\mathbf{u}} : k = 1, 2, \ldots, n_\theta \right\} \tag{65}$$

$$S\left(\mathbf{u}\right) = \left\{ l : \lambda_l > 0 \right\} \tag{66}$$

$$S'\left(\mathbf{u}\right) = \left\{ l : g_l\left(\mathbf{u}, \mathbf{c}\right) = \mathbf{0} \right\} \tag{67}$$

$$T\left(\mathbf{u}\right) = \left\{ \left. \frac{\partial g_l}{\partial \mathbf{u}} \right|_{\mathbf{u}} : l \in S\left(\mathbf{u}\right) \right\} \tag{68}$$

$$T'\left(\mathbf{u}\right) = \left\{ \left. \frac{\partial g_l}{\partial \mathbf{u}} \right|_{\mathbf{u}} : l \in S'\left(\mathbf{u}\right) \right\} \tag{69}$$

where R and T are sets of vectors, S is a set of indices denoting the positive λ_l values at $\mathbf{u}$, and S' is a set of indices denoting the active set at $\mathbf{u}$. Note that $S(\mathbf{u}) \subseteq S'(\mathbf{u})$, and $S(\mathbf{u}) \neq S'(\mathbf{u})$ only if there are active constraints with corresponding multipliers that are zero.

Lemma 2.1. *Assume that* $\mathbf{u}^* \in \arg\min_{\mathbf{u}} (J(\mathbf{u}, \boldsymbol{\theta}) : \mathbf{g}(\mathbf{u}, \mathbf{c}) \leq \mathbf{0})$ *and that* $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta} + \Delta\boldsymbol{\theta} \geq \mathbf{0}$. *If there exists* $\Delta\boldsymbol{\lambda} \geq -\boldsymbol{\lambda}$ *such that*

$$\Delta\boldsymbol{\theta}^T \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\mathbf{u}^*} + \Delta\boldsymbol{\lambda}^T \left.\frac{\partial \mathbf{g}}{\partial \mathbf{u}}\right|_{\mathbf{u}^*} = \mathbf{0} \tag{70}$$

$$\Delta\lambda_l g_l(\mathbf{u}^*, \mathbf{c}) = 0 \; \forall \, l \tag{71}$$

then $\mathbf{u}^* \in \arg\min_{\mathbf{u}} \left(J\left(\mathbf{u}, \hat{\boldsymbol{\theta}}\right) : \mathbf{g}(\mathbf{u}, \mathbf{c}) \leq \mathbf{0}\right)$ *and* $\hat{\boldsymbol{\lambda}} = \boldsymbol{\lambda} + \Delta\boldsymbol{\lambda}$ *are the new Kuhn-Tucker multipliers.*

Proof. If

$$\boldsymbol{\theta}^T \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\mathbf{u}^*} + \boldsymbol{\lambda}^T \left.\frac{\partial \mathbf{g}}{\partial \mathbf{u}}\right|_{\mathbf{u}^*} = \mathbf{0} \tag{72}$$

$$\Delta\boldsymbol{\theta}^T \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\mathbf{u}^*} + \Delta\boldsymbol{\lambda}^T \left.\frac{\partial \mathbf{g}}{\partial \mathbf{u}}\right|_{\mathbf{u}^*} = \mathbf{0} \tag{73}$$

then for $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta} + \Delta\boldsymbol{\theta}$

$$\left(\hat{\boldsymbol{\theta}}^T - \Delta\boldsymbol{\theta}^T\right) \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\mathbf{u}^*} + \left(\boldsymbol{\lambda}^T - \Delta\boldsymbol{\lambda}^T + \Delta\boldsymbol{\lambda}^T\right) \left.\frac{\partial \mathbf{g}}{\partial \mathbf{u}}\right|_{\mathbf{u}^*} = \mathbf{0} \tag{74}$$

$$\hat{\boldsymbol{\theta}}^T \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\mathbf{u}^*} + \left(\boldsymbol{\lambda}^T + \Delta\boldsymbol{\lambda}^T\right) \left.\frac{\partial \mathbf{g}}{\partial \mathbf{u}}\right|_{\mathbf{u}^*} = \mathbf{0} \tag{75}$$

Furthermore, since $\Delta\boldsymbol{\lambda} \geq -\boldsymbol{\lambda}$ and $\Delta\lambda_l g_l(\mathbf{u}^*, \mathbf{c}) = 0 \; \forall \, l$,

$$\hat{\boldsymbol{\theta}}^T \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\mathbf{u}^*} + \hat{\boldsymbol{\lambda}}^T \left.\frac{\partial \mathbf{g}}{\partial \mathbf{u}}\right|_{\mathbf{u}^*} = \mathbf{0} \tag{76}$$

$$0 \leq \hat{\lambda}_l \perp g_l(\mathbf{u}^*, \mathbf{c}) \leq 0 \; \forall \, l \tag{77}$$

where $\hat{\boldsymbol{\lambda}} = \boldsymbol{\lambda} + \Delta\boldsymbol{\lambda}$. Since $\hat{\boldsymbol{\theta}} \geq \mathbf{0}$ and $J\left(\mathbf{u}, \hat{\boldsymbol{\theta}}\right)$ and $\mathbf{g}(\mathbf{u}, \mathbf{c})$ are convex, the optimization

$$\min_{\mathbf{u}} J\left(\mathbf{u}, \hat{\boldsymbol{\theta}}\right) \tag{78}$$

$$\mathbf{g}(\mathbf{u}, \mathbf{c}) \leq \mathbf{0} \tag{79}$$

is convex, and the optimality conditions (76)-(77) are necessary and sufficient. $\mathbf{u}^*$ satisfies these conditions, so $\mathbf{u}^* \in \arg\min_{\mathbf{u}} \left(J\left(\mathbf{u}, \hat{\boldsymbol{\theta}}\right) : \mathbf{g}(\mathbf{u}, \mathbf{c}) \leq \mathbf{0}\right)$. □

Lemma 2.2. *If* $span(R(\mathbf{u}^*)) \subseteq span(T(\mathbf{u}^*))$, *then there exists* $r > 0$ *such that for* $\|\Delta\boldsymbol{\theta}\|_p \leq r$, $p > 0$, $\mathbf{u}^* \in \arg\min_{\mathbf{u}} (J(\mathbf{u}, \boldsymbol{\theta}) : \mathbf{g}(\mathbf{u}, \mathbf{c}) \leq \mathbf{0})$ *implies that* $\mathbf{u}^* \in \arg\min_{\mathbf{u}} \left(J\left(\mathbf{u}, \hat{\boldsymbol{\theta}}\right) : \mathbf{g}(\mathbf{u}, \mathbf{c}) \leq \mathbf{0}\right)$, *where* $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta} + \Delta\boldsymbol{\theta}$.

Proof. Let us define the matrix $\mathbf{A}$ such that the rows of $\mathbf{A}$ are the vectors $\frac{\partial g_l}{\partial \mathbf{u}} \in T(\mathbf{u}^*)$. If $\text{span}(R(\mathbf{u}^*)) \subseteq \text{span}(T(\mathbf{u}^*))$, then any linear combination of $\frac{\partial f_k}{\partial \mathbf{u}} \in R(\mathbf{u}^*)$ exists within $\text{span}(T(\mathbf{u}^*))$, which is the rowspace of $\mathbf{A}$. This implies that for any $\Delta\boldsymbol{\theta}$, there exists $\Delta\boldsymbol{\lambda}$ such that

$$\sum_k \Delta\theta_k \left.\frac{\partial f_k}{\partial \mathbf{u}}\right|_{\mathbf{u}^*} + \sum_{l \in S(\mathbf{u}^*)} \Delta\lambda_l \left.\frac{\partial g_l}{\partial \mathbf{u}}\right|_{\mathbf{u}^*} = \Delta\boldsymbol{\theta}^T \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\mathbf{u}^*} + \mathbf{b}^T\mathbf{A} = \mathbf{0} \tag{80}$$

$$\Delta\lambda_l = 0,\ l \notin S(\mathbf{u}^*) \tag{81}$$

and if $\mathbf{A}^+$ is the Moore-Penrose pseudo-inverse of $\mathbf{A}$, then

$$\mathbf{b}^T = -\Delta\boldsymbol{\theta}^T \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\mathbf{u}^*} \mathbf{A}^+ \tag{82}$$

satisfies this exactly because $\Delta\boldsymbol{\theta}^T \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\mathbf{u}^*}$ is in the rowspace of $\mathbf{A}$. Define $\lambda_{min} = \min_{l \in S(\mathbf{u}^*)} \lambda_l$. By definition, $\lambda_{min} > 0$. If $\|\Delta\boldsymbol{\lambda}\|_p \leq \lambda_{min}$, then

$$\max_l |\Delta\lambda_l| = \|\Delta\boldsymbol{\lambda}\|_\infty \leq \|\Delta\boldsymbol{\lambda}\|_p \leq \lambda_{min},\ p > 0 \tag{83}$$

Therefore, $\|\Delta\boldsymbol{\lambda}\|_p \leq \lambda_{min}$ implies that $\max_l |\Delta\boldsymbol{\lambda}_l| \leq \lambda_{min}$ and thus $\Delta\lambda_l \geq -\lambda_{min} \geq -\lambda_l\ \forall\ l$. If

$$\|\Delta\boldsymbol{\theta}\|_p \leq \frac{\lambda_{min}}{\left\|\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\mathbf{A}^+\right\|_p} = r \tag{84}$$

then

$$\|\Delta\boldsymbol{\lambda}\|_p = \left\|\Delta\boldsymbol{\theta}^T \frac{\partial \mathbf{f}}{\partial \mathbf{u}}\mathbf{A}^+\right\|_p \leq \|\Delta\boldsymbol{\theta}\|_p \left\|\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\mathbf{A}^+\right\| \leq \lambda_{min} \tag{85}$$

By Lemma 2.1, $\mathbf{u}^* \in \arg\min_{\mathbf{u}} \left(J\left(\mathbf{u}, \hat{\boldsymbol{\theta}}\right) : \mathbf{g}(\mathbf{u}, \mathbf{c}) \leq \mathbf{0}\right)$.

□

Corollary 2.2.1. *Define the matrix* $\mathbf{A}$ *such that the rows of* $\mathbf{A}$ *are the vectors* $\frac{\partial g_l}{\partial \mathbf{u}} \in T(\mathbf{u}^*)$. *If* $\mathbf{A}$ *is invertible, then there exists* $r > 0$ *such that for* $\|\Delta\boldsymbol{\theta}\|_p \leq r, p > 0$, $\mathbf{u}^* \in \arg\min_{\mathbf{u}} (J(\mathbf{u}, \boldsymbol{\theta}) : \mathbf{g}(\mathbf{u}, \mathbf{c}) \leq \mathbf{0})$ *implies that* $\mathbf{u}^* \in \arg\min_{\mathbf{u}} \left(J\left(\mathbf{u}, \hat{\boldsymbol{\theta}}\right) : \mathbf{g}(\mathbf{u}, \mathbf{c}) \leq \mathbf{0}\right)$, *where* $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta} + \Delta\boldsymbol{\theta}$.

Proof. If $\mathbf{A}$ is invertible, then the rows of $\mathbf{A}$ are linearly independent and $\text{span}(T(\mathbf{u}^*)) = R^{n_u}$, where $\mathbf{u} \in R^{n_u}$, and thus $\text{span}(R(\mathbf{u}^*)) \subseteq \text{span}(T(\mathbf{u}^*))$. This satisfies the conditions of Lemma 2.2, and thus the same conclusions follow.

□

Lemma 2.3. *The set* $\Theta(\mathbf{u}^*) = \left\{\hat{\boldsymbol{\theta}} : \mathbf{u}^* \in \arg\min_{\mathbf{u}} \left(J\left(\mathbf{u}, \hat{\boldsymbol{\theta}}\right) : \mathbf{g}(\mathbf{u}, \mathbf{c}) \leq \mathbf{0}\right), \hat{\boldsymbol{\theta}} \geq \mathbf{0}\right\}$ *is unbounded and convex if it is non-empty.*

Proof. $J(\mathbf{u}, \boldsymbol{\theta})$ is linear in $\boldsymbol{\theta}$, so $J\left(\mathbf{u}, a\hat{\boldsymbol{\theta}}\right) = aJ\left(\mathbf{u}, \hat{\boldsymbol{\theta}}\right)$ for any positive scalar a. Optimal solutions are invariant with respect to scalar multiples of the objective function:

$$\arg\min_{\mathbf{u}}\left(J\left(\mathbf{u},\hat{\boldsymbol{\theta}}\right):\mathbf{g}\left(\mathbf{u},\mathbf{c}\right)\leq\mathbf{0}\right)=\arg\min_{\mathbf{u}}\left(aJ\left(\mathbf{u},\hat{\boldsymbol{\theta}}\right):\mathbf{g}\left(\mathbf{u},\mathbf{c}\right)\leq\mathbf{0}\right)$$
$$=\arg\min_{\mathbf{u}}\left(J\left(\mathbf{u},a\hat{\boldsymbol{\theta}}\right):\mathbf{g}\left(\mathbf{u},\mathbf{c}\right)\leq\mathbf{0}\right) \tag{86}$$

Therefore, for any $\hat{\boldsymbol{\theta}}\in\Theta\left(\mathbf{u}^*\right)$ and any positive scalar a, $a\hat{\boldsymbol{\theta}}\in\Theta\left(\mathbf{u}^*\right)$. Thus, $\Theta\left(\mathbf{u}^*\right)$ is unbounded if it is non-empty. Furthermore, for fixed $\mathbf{u}^*$, the optimality conditions

$$\hat{\boldsymbol{\theta}}^T\left.\frac{\partial\mathbf{f}}{\partial\mathbf{u}}\right|_{\mathbf{u}^*}+\hat{\boldsymbol{\lambda}}^T\left.\frac{\partial\mathbf{g}}{\partial\mathbf{u}}\right|_{\mathbf{u}^*}=\mathbf{0} \tag{87}$$
$$\hat{\lambda}_l=0\;l\notin S'\left(\mathbf{u}^*\right) \tag{88}$$
$$\hat{\lambda}_l\geq 0\;l\in S'\left(\mathbf{u}^*\right) \tag{89}$$

form a set of linear inequalities in $\hat{\boldsymbol{\lambda}}$ and $\hat{\boldsymbol{\theta}}$; because $\mathbf{u}^*$ is fixed, we can disregard $\mathbf{g}\left(\mathbf{u}^*,\mathbf{c}\right)\geq\mathbf{0}$. The space of $\hat{\boldsymbol{\lambda}}$ and $\hat{\boldsymbol{\theta}}$ that satisfy these constraints is therefore convex. Since this space is convex, for any $\left(\hat{\theta}_1,\hat{\boldsymbol{\lambda}}_1\right)$ and $\left(\hat{\theta}_2,\hat{\boldsymbol{\lambda}}_2\right)$ in this space

$$\left(\alpha\hat{\boldsymbol{\theta}}_1+\left(1-\alpha\right)\hat{\boldsymbol{\theta}}_2,\alpha\hat{\boldsymbol{\lambda}}_1+\left(1-\alpha\right)\hat{\boldsymbol{\lambda}}_2\right),\;\alpha\in[0,1] \tag{90}$$

remains in $\Theta\left(\mathbf{u}^*\right)$. Thus for any $\hat{\boldsymbol{\theta}}_1,\hat{\boldsymbol{\theta}}_2\in\Theta\left(\mathbf{u}^*\right)$, $\left(\alpha\hat{\boldsymbol{\theta}}_1+\left(1-\alpha\right)\hat{\boldsymbol{\theta}}_2\right)\in\Theta\left(\mathbf{u}^*\right)$, so $\Theta\left(\mathbf{u}^*\right)$ is convex. □

Theorem 2.4. *If span* $\left(R\left(\mathbf{u}^*\right)\right)\subseteq$ *span* $\left(T\left(\mathbf{u}^*\right)\right)$ *and* $\mathbf{u}^*\in\arg\min_{\mathbf{u}}\left(J\left(\mathbf{u},\boldsymbol{\theta}\right):\mathbf{g}\left(\mathbf{u},\mathbf{c}\right)\leq\mathbf{0}\right)$*, there exists a convex, unbounded set of* $\Delta\boldsymbol{\theta}$ *such that* $\mathbf{u}^*\in\arg\min_{\mathbf{u}}\left(J\left(\mathbf{u},\boldsymbol{\theta}+\Delta\boldsymbol{\theta}\right):\mathbf{g}\left(\mathbf{u},\mathbf{c}\right)\leq\mathbf{0}\right)$.

Proof. By Lemma 2.2, if $\text{span}\left(R\left(\mathbf{u}^*\right)\right)\subseteq\text{span}\left(T\left(\mathbf{u}^*\right)\right)$ and $\mathbf{u}^*\in\arg\min_{\mathbf{u}}\left(J\left(\mathbf{u},\boldsymbol{\theta}\right):\mathbf{g}\left(\mathbf{u},\mathbf{c}\right)\leq\mathbf{0}\right)$, then there exists $r>0$ such that for $\left\|\Delta\boldsymbol{\theta}\right\|_p\leq r$, $p>0$, $\mathbf{u}^*\in\arg\min_{\mathbf{u}}\left(J\left(\mathbf{u},\boldsymbol{\theta}+\Delta\boldsymbol{\theta}\right):\mathbf{g}\left(\mathbf{u},\mathbf{c}\right)\leq\mathbf{0}\right)$. Therefore, the set

$$\Theta\left(\mathbf{u}^*\right)=\left\{\hat{\boldsymbol{\theta}}:\mathbf{u}^*\in\arg\min_{\mathbf{u}}\left(J\left(\mathbf{u},\hat{\boldsymbol{\theta}}\right):\mathbf{g}\left(\mathbf{u},\mathbf{c}\right)\leq\mathbf{0}\right),\hat{\boldsymbol{\theta}}\geq\mathbf{0}\right\} \tag{91}$$

is non-empty. By Lemma 2.3 if $\Theta\left(\mathbf{u}^*\right)$ is non-empty, it is unbounded and convex. □

Lemma 2.5. *If span* $\left(R\left(\mathbf{u}^*\right)\right)\nsubseteq$ *span* $\left(T'\left(\mathbf{u}^*\right)\right)$ *for* $\mathbf{u}^*\in\arg\min_{\mathbf{u}}\left(J\left(\mathbf{u},\boldsymbol{\theta}\right):\mathbf{g}\left(\mathbf{u},\mathbf{c}\right)\leq\mathbf{0}\right)$*, then for any* $\epsilon>0$*, there exists* $\Delta\boldsymbol{\theta}$ *such that* $\left\|\Delta\boldsymbol{\theta}\right\|<\epsilon$ *and* $\mathbf{u}^*\notin\arg\min_{\mathbf{u}}\left(J\left(\mathbf{u},\hat{\boldsymbol{\theta}}\right):\mathbf{g}\left(\mathbf{u},\mathbf{c}\right)\leq\mathbf{0}\right)$.

Proof. Assume that for sufficiently small $\epsilon>0$, there is no $\Delta\boldsymbol{\theta}$ such that $0<\left\|\Delta\boldsymbol{\theta}\right\|<\epsilon$ and $\mathbf{u}^*\notin\arg\min_{\mathbf{u}}\left(J\left(\mathbf{u},\hat{\boldsymbol{\theta}}\right):\mathbf{g}\left(\mathbf{u},\mathbf{c}\right)\leq\mathbf{0}\right)$. Then for sufficiently small $\Delta\boldsymbol{\theta}$, there exists $\Delta\boldsymbol{\lambda}$ such that

$$\Delta\boldsymbol{\theta}^T\left.\frac{\partial\mathbf{f}}{\partial\mathbf{u}}\right|_{\mathbf{u}^*}+\Delta\boldsymbol{\lambda}^T\left.\frac{\partial\mathbf{g}}{\partial\mathbf{u}}\right|_{\mathbf{u}^*}=\mathbf{0} \tag{92}$$
$$\Delta\lambda_l\geq-\lambda_l\;l\in S'\left(\mathbf{u}^*\right) \tag{93}$$
$$\Delta\lambda_l=0\;l\notin S'\left(\mathbf{u}^*\right) \tag{94}$$

Since $\mathbf{u}^*$ is fixed, the active set cannot change. Define the matrix $\mathbf{A}$ such that the rows of $\mathbf{A}$ are vectors $\frac{\partial g_l}{\partial \mathbf{u}} \in T'(\mathbf{u}^*)$ and define the vector $\mathbf{b}$ such that the elements of $\mathbf{b}$ are $\Delta\lambda_l,\ l \in S'(\mathbf{u}^*)$. Then

$$\Delta\boldsymbol{\theta}^T \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\mathbf{u}^*} + \Delta\boldsymbol{\lambda}^T \left.\frac{\partial \mathbf{g}}{\partial \mathbf{u}}\right|_{\mathbf{u}^*} = \Delta\boldsymbol{\theta}^T \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\mathbf{u}^*} + \mathbf{b}^T\mathbf{A} = \mathbf{0} \tag{95}$$

If $\text{span}(R(\mathbf{u}^*)) \nsubseteq \text{span}(T(\mathbf{u}^*)),\ l \in S'(\mathbf{u}^*)$, then $\exists\ \Delta\boldsymbol{\theta}_0$ such that $\Delta\boldsymbol{\theta}_0^T \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \notin \text{span}(T(\mathbf{u}^*))$ and

$$\Delta\boldsymbol{\theta}_0^T \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\mathbf{u}^*} + \mathbf{b}^T\mathbf{A} \neq \mathbf{0}\ \forall\ \mathbf{b} \tag{96}$$

Moreover, for any such $\Delta\boldsymbol{\theta}_0$, there exists $\Delta\boldsymbol{\theta} = a\Delta\boldsymbol{\theta}_0$ such that for any $a > 0$

$$a\Delta\boldsymbol{\theta}_0^T \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\mathbf{u}^*} + \mathbf{b}^T\mathbf{A} \neq \mathbf{0}\ \forall\ \mathbf{b} \tag{97}$$

Since $\|a\Delta\boldsymbol{\theta}_0\| = a\|\Delta\boldsymbol{\theta}_0\|$, for any $\epsilon > 0$, there exists $\Delta\boldsymbol{\theta} = \frac{\epsilon}{\|\Delta\boldsymbol{\theta}_0\|}\Delta\boldsymbol{\theta}_0$ such that

$$\Delta\boldsymbol{\theta}^T \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\mathbf{u}^*} + \mathbf{b}^T\mathbf{A} \neq \mathbf{0}\ \forall\ \mathbf{b} \tag{98}$$

The optimality conditions are necessary and sufficient and these conditions cannot be satisfied, so $\mathbf{u}^* \notin \arg\min_{\mathbf{u}} (J(\mathbf{u}, \boldsymbol{\theta} + \Delta\boldsymbol{\theta}) : \mathbf{g}(\mathbf{u}, \mathbf{c}) \leq \mathbf{0})$, and thus the lemma is proved by contradiction. □

If small $\Delta\boldsymbol{\theta}$ values change the value of $\mathbf{u}^*$ but not the active set, it is possible to calculate the $\frac{\partial \mathbf{u}}{\partial \Delta\boldsymbol{\theta}}$ for the optimal solution by differentiating the optimality conditions. This provides us with a linear system that we can solve to calculate $\frac{\partial \mathbf{u}}{\partial \Delta\boldsymbol{\theta}}$, and $\mathbf{u}^*(\Delta\boldsymbol{\theta})$ will be smooth and well-defined as long as the active set does not change. We can therefore compare this kind of system with one that is impervious to these small changes. For such a system, the measure of the 'safe' range is conservative, but outside of it, continuous changes in $\hat{\boldsymbol{\theta}}$ could result in discrete jumps in $\mathbf{u}^*$ as the active set changes. If $J(\mathbf{u}, \boldsymbol{\theta})$ is nonlinear in $\boldsymbol{\theta}$ but still convex for all $\boldsymbol{\theta} \geq 0$, it may possible to produce similar proofs , but this would require further assumptions regarding $J(\mathbf{u}, \boldsymbol{\theta})$.

2.3.2 Constraint Function Manipulations. Unfortunately, manipulations of $\mathbf{c}$ are not subject to the same kinds of robustness that manipulations of $\boldsymbol{\theta}$ are. This is essentially a consequence of the discussion at the beginning of Section 2.2: manipulating the defender's perception of the constraints produces the same change in the decision variables as changing the true constraints would as long as the defender abides by the perceived constraints. For example,

$$g_l(\mathbf{u}, \mathbf{c}) = \mathbf{0},\ l \in S(\mathbf{u}) \tag{99}$$

$$\sum_i \frac{\partial g_l}{\partial u_i}\frac{\partial u_i}{\partial c_j} + \frac{\partial g_l}{\partial c_j} = 0 \tag{100}$$

Therefore, if $\frac{\partial g_l}{\partial c_j} \neq 0$, then $\frac{\partial u_i}{\partial c_j} \neq 0$.

2.4 Test Problem

As a demonstration, we consider minimizing power consumption for a fan in an HVAC system. A problem like this could form a component in a larger Building Automation System (BAS), possibly as a subsystem subject to repeated optimization under changing parameter values. Many large commercial buildings use sophisticated BASs to monitor and control building equipment [19], and standard communication protocols have been introduced to facilitate interoperability of connected BAS components using publically accessible networks [16]. As such, BAS components may become highly susceptible to cyber-physical attacks. The baseline defender optimization problem is

$$\min_{m,p} \theta_1 m + \theta_2 m^2 + \theta_3 p \tag{101}$$

$$\frac{1}{2}\left[(m - c_m)^2 + \left(p - c_p\right)^2 - c_r^2\right] \leq 0 \tag{102}$$

where m is the mass flow rate, p is the static pressure, the $\boldsymbol{\theta}$ values are power consumption parameters for the fan, and c_m, c_p, and c_r are parameters defining the operating envelope. The attacker can introduce perturbations $\Delta\theta_i$ such that $\hat{\theta}_i = \theta_i + \Delta\theta_i$ and $\frac{1}{2}\left\|\Delta\boldsymbol{\theta}\right\|_2^2 \leq \delta_{\theta,max}$ or perturbations $\Delta c_m, \Delta c_p, \Delta c_r$ such that $\hat{c}_m = c_m + \Delta c_m$, $\hat{c}_p = c_p + \Delta c_p$, $\hat{c}_r = c_r - \Delta c_r$, and $\frac{1}{2}\left\|\Delta\mathbf{c}\right\|_2^2 \leq \delta_{c,max}$. Note the negative sign in $\hat{c}_r$. This deviates slightly from our convention above, but it also helps to simplify later calculations in some ways, and it does not ultimately change the results. In our computations in the rest of the paper, we use $\theta_1 = \theta_2 = 1$, $\theta_3 = 2$, $c_m = c_p = 5$, and $c_r^2 = 10$. The $\frac{1}{2}$ constant in (102) does not change the mathematical properties of the optimization, but it, too, simplifies some of the calculations used later in this paper; see Appendix A for these calculations.

3 DYNAMIC OPTIMIZATION

3.1 Model Formulation

We now bring hypergames to bear on a Model Predictive Control (MPC) problem, where the control objective is to minimize a cost function subject to state dynamics constraints and operational constraints over a time horizon of length τ:

$$\min_{\mathbf{u}^t} \sum_{t=1}^{\tau} J\left(\mathbf{u}^t, \mathbf{x}^t, \boldsymbol{\theta}\right) \tag{103}$$

$$\mathbf{x}^t = \mathbf{f}\left(\mathbf{x}^{t-1}, \mathbf{u}^t, \boldsymbol{\alpha}^t, \boldsymbol{\beta}\right) \tag{104}$$

$$\mathbf{x}^\tau - \mathbf{x}^0 = \mathbf{0} \tag{105}$$

$$\mathbf{g}\left(\mathbf{x}^t, \mathbf{u}^t, \boldsymbol{\alpha}^t, \boldsymbol{\beta}\right) \leq \mathbf{0} \tag{106}$$

where $\mathbf{u}^t$ are the control decision variables, $\mathbf{x}^t$ are the states of the system, $\boldsymbol{\alpha}^t$ are the system disturbances, and $\boldsymbol{\beta}$ are the model parameters. We assume that $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}^t$ can be affected by adversarial perturbations. The attacker can either perturb the defender's perception of parameters $\boldsymbol{\beta}$ to maximize cost ('Static Attack') or perturb the defender's perception of $\boldsymbol{\alpha}^t$ to maximize cost ('Dynamic Attack'). The perturbations denoted $\Delta\boldsymbol{\beta}$, and $\Delta\boldsymbol{\alpha}^t$ are bounded by constraints, normalized as appropriate if they have different orders of magnitude; such constraints are then with respect to *relative* perturbations on those parameters.

$$\frac{\Delta\boldsymbol{\beta}}{\boldsymbol{\beta}} \equiv \left[\frac{\Delta\beta_1}{\beta_1} \;\; \frac{\Delta\beta_2}{\beta_2} \;\; \dots\right]^T \tag{107}$$

$$\frac{1}{2}\left\|\frac{\Delta\boldsymbol{\beta}}{\boldsymbol{\beta}}\right\|^2 \leq \delta_{\beta,max} \tag{108}$$

$$\frac{1}{2}\sum_{t=1}^{\tau}\|\Delta\boldsymbol{\alpha}^t\|^2 \leq \delta_{\alpha,max} \tag{109}$$

The static attack problem is

$$\max_{\Delta\boldsymbol{\beta}} \sum_{t=1}^{\tau} J(\mathbf{u}^t, \mathbf{x}^t, \boldsymbol{\theta}) \tag{110}$$

$$\frac{1}{2}\left\|\frac{\Delta\boldsymbol{\beta}}{\boldsymbol{\beta}}\right\|^2 \leq \delta_{\beta,max} \tag{111}$$

$$\mathbf{x}^t = \mathbf{f}\left(\mathbf{x}^{t-1}, \mathbf{u}^t, \boldsymbol{\alpha}^t, \boldsymbol{\beta}\right) \tag{112}$$

$$\hat{\mathbf{x}}^0 = \mathbf{x}^0 \tag{113}$$

subject to

$$\min_{\mathbf{u}^t} \sum_{t=1}^{\tau} J(\mathbf{u}^t, \hat{\mathbf{x}}^t, \boldsymbol{\theta}) \tag{114}$$

$$\hat{\mathbf{x}}^t = \mathbf{f}\left(\hat{\mathbf{x}}^{t-1}, \mathbf{u}^t, \boldsymbol{\alpha}^t, \hat{\boldsymbol{\beta}}\right) \tag{115}$$

$$\hat{\mathbf{x}}^\tau - \hat{\mathbf{x}}^0 = \mathbf{0} \tag{116}$$

$$\mathbf{g}\left(\hat{\mathbf{x}}^t, \mathbf{u}^t, \boldsymbol{\alpha}^t, \hat{\boldsymbol{\beta}}\right) \leq \mathbf{0} \tag{117}$$

This is a second level hypergame where $p\left(D, \boldsymbol{\beta}\right) = \hat{\boldsymbol{\beta}} \neq \boldsymbol{\beta}$. The defender optimization is with respect to perceived values, not real values; the attacker perturbations mean that $p\left(D, \mathbf{x}^t\right) = \hat{\mathbf{x}}^t \neq \mathbf{x}^t$ even though the attacker does not directly manipulate the state variables. The static attack problem has the attacker choose a deterministic strategy over $\Delta\boldsymbol{\beta}$ values, while the attacker chooses a deterministic strategy over $\Delta\boldsymbol{\alpha}^t$ values in the dynamic attack; in both cases the defender chooses a deterministic strategy over $\mathbf{u}^t$ values.

$$\max_{\Delta\boldsymbol{\alpha}^t} \sum_{t=1}^{\tau} J\left(\mathbf{u}^t, \mathbf{x}^t, \boldsymbol{\theta}\right) \tag{118}$$

$$\frac{1}{2}\sum_{t}\left\|\Delta\boldsymbol{\alpha}^t\right\|^2 \leq \delta_{\alpha,max} \tag{119}$$

$$\mathbf{x}^t = \mathbf{f}\left(\mathbf{x}^{t-1}, \mathbf{u}^t, \boldsymbol{\alpha}^t, \boldsymbol{\beta}\right) \tag{120}$$

$$\hat{\mathbf{x}}^0 = \mathbf{x}^0 \tag{121}$$

subject to

$$\min_{\mathbf{u}^t} \sum_{t=1}^{\tau} J\left(\mathbf{u}^t, \hat{\mathbf{x}}^t, \boldsymbol{\theta}\right) \tag{122}$$

$$\hat{\mathbf{x}}^t = \mathbf{f}\left(\hat{\mathbf{x}}^{t-1}, \mathbf{u}^t, \hat{\boldsymbol{\alpha}}^t, \boldsymbol{\beta}\right) \tag{123}$$

$$\hat{\mathbf{x}}^\tau - \hat{\mathbf{x}}^0 = \mathbf{0} \tag{124}$$

$$\mathbf{g}\left(\hat{\mathbf{x}}^t, \mathbf{u}^t, \hat{\boldsymbol{\alpha}}^t, \boldsymbol{\beta}\right) \leq \mathbf{0} \tag{125}$$

This, similarly, is a second level hypergame where $p\left(D, \boldsymbol{\alpha}^t\right) = \hat{\boldsymbol{\alpha}}^t \neq \boldsymbol{\alpha}^t$. As before, we could consider many variations on the dynamic and static attacks, but we will only look at these two scenarios here.

3.2 Analytical Results

The analytical results derived for the static optimization problem are applicable here as well. If the dynamic optimization is convex, there are analogous results for perturbations to $\boldsymbol{\theta}$, and it can similarly be shown that constraint perturbations (to $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}^t$, in this case) cannot exhibit the same kind of local robustness as objective function perturbations.

3.3 Test Problem

Our MPC test problem is a single-zone HVAC system with a fan, heater, and chiller. The objective is to minimize power consumption subject to physical constraints (e.g., the zonal temperature evolution) and operational constraints (e.g., remaining within comfort-defined temperature limits). The baseline optimal control problem for the system is

$$\begin{aligned}\min \sum_{t=1}^{\tau} \Big[& \theta_1 m^t + \theta_2 \left(m^t\right)^2 + v_h c_p m^t \left(T_i^t - d^t T_0^t - \left(1 - d^t\right) T_n^t\right) \\ & + v_n c_p m^t \left(T_{s,n}^t - T_s^t\right) + v_c c_p m^t \left(T_i^t - T_s^t\right) \Big] \end{aligned} \tag{126}$$

$$T_n^t = (1-\gamma)\, T_n^{t-1} + \beta m^t \left(T_{s,n}^t - T_n^t\right) + \gamma T_0^t + Q_n^t \tag{127}$$

$$T_n^\tau - T_n^0 = 0 \tag{128}$$

$$m_l \leq m^t \leq m_u \tag{129}$$

$$T_{s,n}^t - T_s^t \geq 0 \tag{130}$$

$$T_n^l \leq T_n^t \leq T_n^u \tag{131}$$

$$d_l \leq d^t \leq d_u \tag{132}$$

$$T_{s,n}^l \leq T_{s,n}^t \leq T_{s,n}^u \tag{133}$$

$$T_i^t - d^t T_0^t - \left(1 - d^t\right) T_n^t \geq 0 \tag{134}$$

$$T_i^t - T_s^t \geq 0 \tag{135}$$

where m^t is the mass flow rate, T_i^t is the internal duct temperature, T_s^t is the temperature of the air put out by the chiller, $T_{s,n}^t$ is the temperature of the air supplied to the zone, T_n^t is the temperature of the zone, and d^t is the damper position. All of these are control variables. T_0^t is the external temperature (set to 25°C in this instantiation of the model); β and γ are scalar parameters that capture the room thermal properties. Other quantities listed in the problem description are parameters that are not affected by any adversarial perturbations. See Appendix B for more details. The fan, heater, and chiller power consumption levels at each time step are

$$\theta_1 m^t + \theta_2 \left(m^t\right)^2 \tag{136}$$

$$v_h c_p m^t \left(T_i^t - d^t T_0^t - \left(1 - d^t\right) T_n^t\right) v_n c_p m^t \left(T_{s,n}^t - T_s^t\right) \tag{137}$$

$$v_c c_p m^t \left(T_i^t - T_s^t\right) \tag{138}$$

respectively. In this model, the static pressure is almost constant, and thus we omit it from the fan component of the model. The static attack manipulates the defender perception of β and γ. The attacker goal is to maximize power consumption over the entire time horizon given that the defender observes $\hat{\beta} = \beta + \Delta\beta$ and $\hat{\gamma} = \gamma + \Delta\gamma$ and the attacker is constrained by

$$\frac{1}{2}\left[\left(\frac{\Delta\beta}{\beta}\right)^2 + \left(\frac{\Delta\gamma}{\gamma}\right)^2\right] \leq \delta_{max} \tag{139}$$

subject to the defender optimization of the original baseline problem. Because β and γ are of different magnitudes, using relative perturbations, not absolute ones, avoids some potential problems. We also highlight the previously mentioned differences between the perceived and actual state variables values. For example, the true zone temperature, T_n^t, and the defender perception of the zone temperature, $\hat{T}_n^t$, will evolve according to the equations, respectively,

$$T_n^t = (1 - \gamma) T_n^{t-1} + \beta m^t \left(T_{s,n}^t - T_n^t\right) + \gamma T_0^t + Q_n^t \tag{140}$$

$$\hat{T}_n^t = (1 - \hat{\gamma}) \hat{T}_n^{t-1} + \hat{\beta} m^t \left(T_{s,n}^t - \hat{T}_n^t\right) + \hat{\gamma} T_0^t + Q_n^t \tag{141}$$

There will be a similar discrepancy between T_i^t and $\hat{T}_i^t$. The dynamic attack manipulates the defender's perception of T_0^t so that $\hat{T}_0^t = T_0^t + \Delta T_0^t$ and $\frac{1}{2}\sum_t \left(\Delta T_0^t\right)^2 \leq \Delta T_{max}$. As in the static parameter manipulation case, the defender will misperceive both T_n^t and T_i^t. The full formulations for the static and dynamic manipulation problems are provided in Appendix B.

4 COMPUTATIONAL IMPLEMENTATION

The specific calculations to turn each hypergame problem into a tractable nonlinear program (NLP) are provided in Appendices A and B. We summarize our general approach here. Each hypergame produces a multi-level optimization problem. To solve this, we write the optimality conditions of the lower level problems as complementarity conditions. In the case of the fan optimization, we can transform these complementarity conditions into equality constraints and then solve the resulting problem as an NLP. For the HVAC problem, we cannot do this, and this leaves us with a Mathematical Program with Equilibrium Constraints (MPEC) [24]. We can solve the MPEC as a series of NLPs by relaxing the complementarity constraints and penalizing the relaxation with a progressively increasing weight. For the work described in this paper, this was both reliable and efficient. To implement our approach, we derived the necessary optimality conditions by hand, coded up the NLPs in MATLAB [17], and solved the NLPs using *fmincon*.

5 RESULTS

5.1 Fan Optimization

Table 1 shows the results for the attacker manipulation of the objective function parameters; power consumption values in parentheses indicate the power usage perceived by the defender where it differs from the actual usage. Manipulating the true θ_i values produced a notable increase in power

Table 1. Objective Function Manipulation Results ($\delta_{\theta,max} = 0.1$)

Case	m	p	$\Delta\theta_1$	$\Delta\theta_2$	$\Delta\theta_3$	Power
Baseline	2.06	3.85	-	-	-	13.97
True Manipulation	2.02	3.94	0.150	0.303	0.292	16.68
Perception Manipulation	2.29	3.38	-0.090	-0.411	0.151	14.26 (12.42)
Faulty Defender Anticipation	1.95	4.16	-	-	-	14.08 (14.71)
Double-Bluff Manipulation	1.89	4.42	0.00684	0.259	-0.358	14.30 (13.76)

consumption compared with the baseline. Manipulating defender perceptions, though, proved less effective. For example, when the attacker manipulated the perceptions of an unsuspecting defender (Perception Manipulation), the gap between the perceived and actual power usage was noticeable, but the actual increase in power relative to the baseline case was small. Similarly, if the defender erroneously thought that the attacker was manipulating the perceived values of θ_i (Faulty Defender Anticipation), the true power usage was almost identical to the baseline case, though the perceived power consumption was somewhat higher. When manipulating the defender's perceptions, the attacker got the defender to increase m and decrease p (relative to the baseline case) by decreasing the perceived value of θ_1 and θ_2 ($\Delta\theta_1 < 0$, $\Delta\theta_2 < 0$) and increasing the perceived value of θ_3 ($\Delta\theta_3 > 0$). This approach is more beneficial for the attacker than decreasing m and increasing p because the objective is quadratic in m but only linear in p. In the double-bluff situation, however, the defender expects the attacker to employ this optimal strategy, and so the attacker does the exact opposite (i.e., encourages the defender to increase p and decrease m), which provides a slight additional benefit over the simple manipulation case.

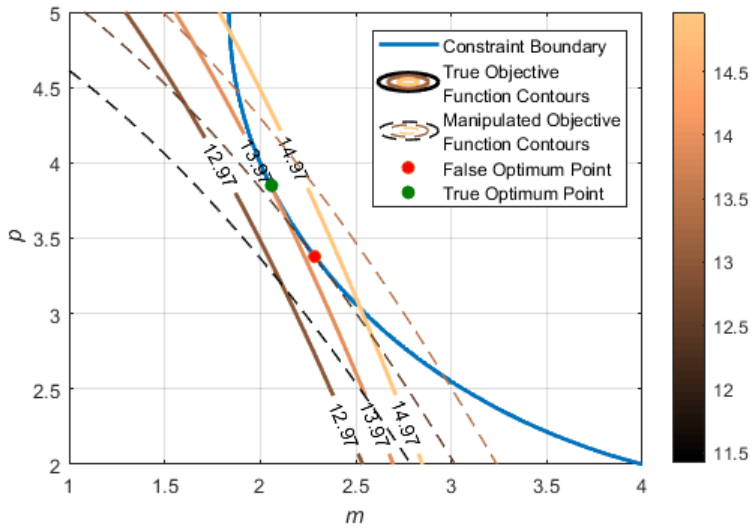


Fig. 1. Visualization of 'Perception Manipulation' attack.

Fig. 1 shows the 'Perception Manipulation' case and why it produces so little payoff for the attacker. The perceived objective function contours are essentially a rotated version of the original objective function contours. That rotation, produced by changes in the relative magnitudes of the θ_i

parameters, produces a perceived (i.e., false) optimum point that is noticeably different from the true optimum point. However, even a significant difference in the solution location does not translate to a large difference in the true objective function value because neither the constraint nor the objective function contours have large curvatures near the true optimum – most of the translation between the two points is parallel to the contours of the true objective function. Manipulating constraints gave the attacker more options than manipulating the objective function parameters. As Table 2 shows, constraint manipulation was also much more effective as an attacker strategy. For example, when the attacker attempted to maximize power consumption against a defender who did not believe an attack was underway (Power Max, Normal), the attacker was able to increase power consumption by almost 30% compared with the baseline. Attempting to maximize the constraint violation (Break System, Normal) resulted in a significant level of violation, too.

Table 2. Constraint Manipulation Results ($\delta_{c,max} = 0.1$)

Attacker Action	Defender Belief	m	p	Power	Violation
No Attack	Normal	2.06	3.85	13.97	-
Power Max	Normal	2.59	4.22	17.76	-
No Attack	Power Max	1.57	3.37	10.79	4.92
No Attack	Break System	2.59	2.24	17.76	-
Break System	Power Max	1.17	2.78	8.11	4.85
Power Max	Break System	3.16	4.53	22.21	-
Break System	Normal	1.58	3.36	10.79	2.20
Power Max (Double-Bluff)	Power Max	2.16	3.94	14.71	0.406
Break System (Double-Bluff)	Break System	2.05	3.87	13.97	0.003

Table 3. Constraint Manipulation Results ($\delta_{c,max} = 0.1$)

Attacker Action	Defender Belief	Δc_m	Δc_p	Δc_r
Power Max	Normal	0.301	0.097	0.316
Break System	Power Max	-0.285	-0.137	-0.316
Power Max	Break System	0.301	0.097	0.316
Break System	Normal	-0.285	-0.137	-0.316
Power Max (Double-Bluff)	Power Max	0.419	0.157	0.000
Break System (Double-Bluff)	Break System	-0.295	-0.113	-0.316

In the case of constraint manipulations, there were also major consequences for wrongly anticipating an attack. Anticipating a power maximization attack when there was no attack resulted in a worse constraint violation than when the attacker was deliberately trying to break the system. Conversely, anticipating a 'break system' attack when the actual attack was a 'power max' attack led to an increase in power consumption of almost 60% compared with the baseline. Note that in these false anticipations, the attacker is assuming that the defender is just playing normally (i.e., the attacker is not taking advantage of the defender's mistake). The double-bluff strategies did not provide much benefit to the attacker, though. Table 3 also shows the perturbations used by the attacker. We can see that the attacker strategies for maximizing power consumption and breaking the system are almost exactly mirror opposites, which makes sense.

The double-bluff strategies are not that much different than the regular strategies that they correspond to, though, so it is not surprising that the double-bluff approach is not very effective.

Switching attack modes would be a better option if the defender is anticipating an attack, and though we did not calculate this here, it would be possible to calculate an optimal attack for one mode given that the defender is expecting the other mode. Given how the two modes produce almost exactly opposite attacker strategies, the attacker strategy would likely be quite similar to the same attack mode employed against an unsuspecting defender. In general, changes in constraint parameters may result in larger objective function changes than changes in objective function parameters for two reasons. Firstly, the changes in constraints will be multiplied by the dual variables (Lagrange or Kuhn-Tucker) associated with those constraints to produce a final change in the objective function. Secondly, changing constraint values may result in the active set at the optimum also changing, and that could produce large, nonlinear changes in the objective function. All in all, this likely makes constraint manipulation a much more attractive target for a would-be attacker than objective function manipulation.

5.2 Single-Zone HVAC Control

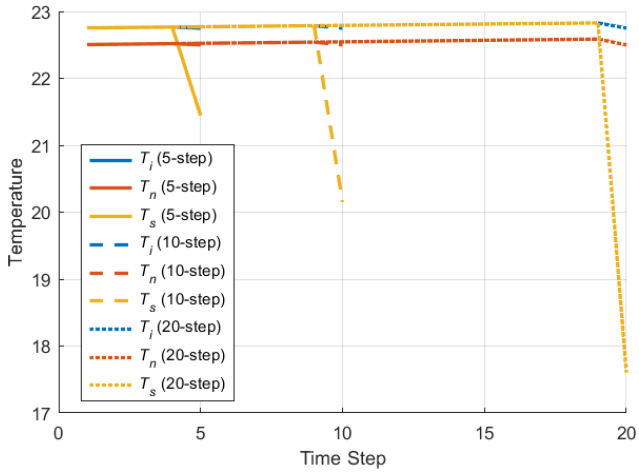


Fig. 2. Baseline temperature results.

In the baseline case, and for all of the adversarial perturbations, m^t and d^t were both at their lower bounds for the entire optimization. Fig. 2 shows the defender strategy in more detail for different optimization horizon lengths. There, we see that the defender essentially allows the zone to evolve without manipulation until the last time step. Because $T_0^t > T_n^t$, this means that the zone warms over time, but because γ is very small, this happens slowly. At the last time step, the defender then chills the zone back to the initial temperature. We can see this in the sudden drop in T_s^t at the end of each time horizon; note that the optimization produces $T_s^t = T_{s,n}^t$ for each optimization. This general behaviour is seen when the attacker manipulates defender perceptions, too. The longer the optimization time horizon, the larger the drop in T_s^t at the last time step. If the length of the time horizon were increased sufficiently, eventually the system would require multiple steps of cooling, because T_s^t would hit its lower bound. T_n^t never hit its upper bound, but if it did, this would also require additional cooling prior to the end of the optimization horizon.

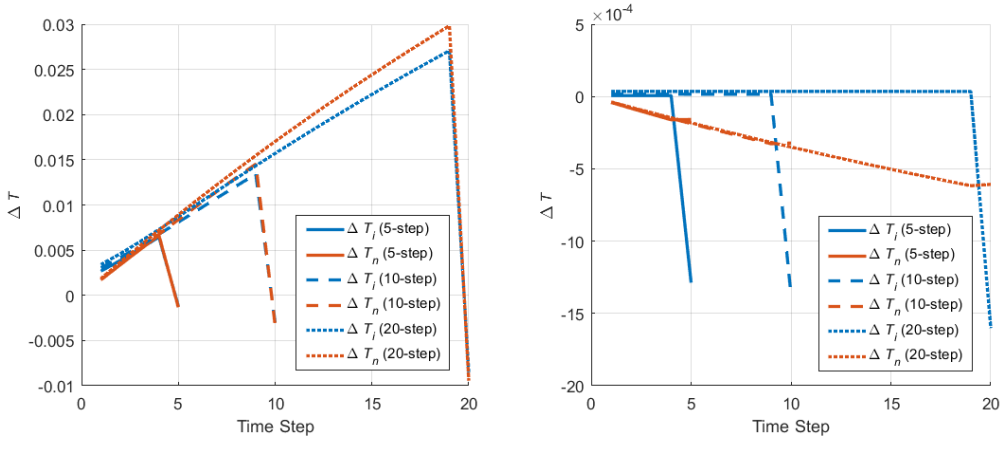
Table 4 shows that manipulating the defender's perception of β and γ resulted in small power increases, relative to the baseline, and small discrepancies between the actual and perceived power use. The perturbations themselves also change slightly as the length of the time horizon changes;

Table 4. Static Parameter Manipulation Results ($\delta_{max} = 0.1$)

	5-step	10-step	20-step
Baseline Power	14.76	29.48	58.77
Actual Power	15.08	30.27	60.95
Defender Perceived Power	15.00	29.97	59.80
$\Delta\beta$	-1.81e-3	-1.84e-3	-1.94e-3
$\Delta\gamma$	1.64e-5	1.52e-5	9.74e-6
λ_{mean}	367	370	383

there is a greater emphasis on $\Delta\beta$ as the time horizon gets longer. In this model, β essentially measures how hard it is to change the zone temperature with the HVAC system. Setting $\Delta\beta < 0$ makes the defender think that the zone is harder to adjust than it actually is. The γ parameter then captures the heat transfer between the zone and the outside environment. Setting $\Delta\gamma > 0$ makes the defender think that there is more heat transfer than there actually is. All of this combines to increase the amount of cooling that the defender thinks is necessary at the end. The ΔT plots in Figs. 3a and 3b show this kind of behaviour: the defender thinks that the temperatures are higher than they actually are and therefore overcompensates at the end. This overcompensation leads to an increase in power use and a final T_n^t value that is actually slightly lower than it should be.

Next, we can look at the λ_{mean} values given in Table 4. λ_{mean} is the average of the Lagrange multipliers associated with (141) and therefore provides a measure of how the $\Delta\beta$ and $\Delta\gamma$ perturbations get multiplied. This value increases as the time horizon lengthens, which makes sense: as the time horizon lengthens, the importance of the thermal evolution process increases. An attacker perturbing β and γ would want this value to be as large (positive or negative) as possible.



(a) Static attack.

(b) Dynamic attack.

Fig. 3. Temperature deviations, $\Delta T = \left(T_{true} - T_{perceived}\right)$.

Table 5 shows that manipulating T_0^t provided a much larger increase in power consumption as well as a larger difference between the perceived and actual power consumption. λ_{mean} is also much smaller, and these phenomena are related. The static parameters could only affect the power consumption indirectly through the temperature evolution equation. T_0^t, however, shows up in

Table 5. Dynamic Attack Results ($\Delta T_{max} = 0.1n$ for n-step problem)

	5-step	10-step	20-step
Baseline Power	14.76	29.48	58.77
Actual Power	16.35	32.85	65.68
Defender Perceived Power	15.58	31.20	62.27
λ_{mean}	219	218	216

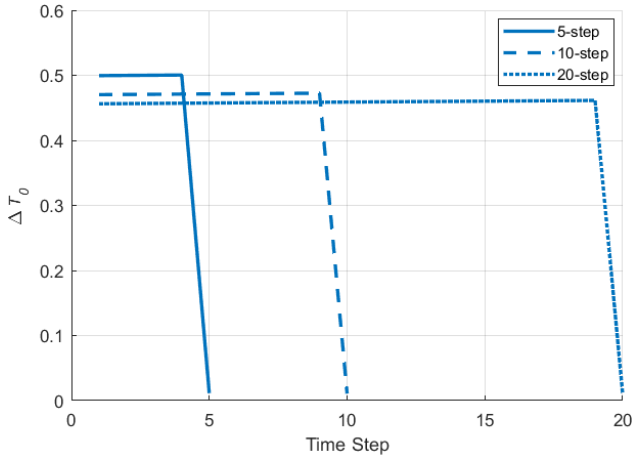


Fig. 4. Dynamic parameter manipulation temperature perturbations.

Table 6. Power Consumption Comparisons relative to Baseline (%)

	5-step	10-step	20-step
Static Attack (Perceived)	1.6	1.7	1.8
Static Attack (Actual)	2.2	2.7	3.7
Dynamic Attack (Perceived)	5.6	5.8	6.0
Dynamic Attack (Actual)	10.1	11.4	11.8

the objective function and another constraint in addition to the temperature evolution equation, so increasing λ_{mean} becomes less important. In this case, misperceptions of T_i^t and T_n^t become smaller (see Figs. 3a and 3b) and less important to the attacker. Instead, the attacker uses $\Delta T_0^t > 0$ to get the defender to increase T_i^t, and thus the defender ends up engaging the heater (because $T_i^t - d^t T_0^t - \left(1 - d^t\right) T_n^t > 0$ even though $\hat{T}_i^t - d^t \hat{T}_0^t - \left(1 - d^t\right) \hat{T}_n^t = 0$) as well as the chiller. The perturbations themselves follow a clear pattern, as shown in Fig. 4. They increase very slightly over time until the last time step, at which point they drop to nearly zero. The last step is less valuable to the attacker because there are no more thermal evolution steps left in the optimization at that point. Table 6 provides an overall summary of the power consumption results. Generally speaking, the relative payoff for the attacker increases with the length of the time horizon. The actual power consumed in the static attack scenario, relative to the baseline, is roughly proportional to the length of the time horizon, but the other three cases in Table 6 all seem to plateau.

6 DISCUSSION

6.1 Stuxnet-like Attacks and Hypergames

In this paper, we showed how an attacker with system access and knowledge could manipulate the optimization processes of that system. These problems were relatively small but showed how the analysis works. Hypergames are about strategic interactions when there are misperceptions and/or information asymmetries. In this case, we were able to show how those asymmetries or misperceptions could affect system performance. For example, getting the defender to respond to a non-existent threat could actually prove to be a very effective attacker strategy. Conversely, it is possible for the defender system to have a natural robustness to perturbations (though that was not the case in these test problems). We could consider more complex interactions, and we intend to do so in future work, but that future work will need to build upon the basics outlined here.

When we look at Stuxnet as a motivating example for this work, we can see that there are many similarities as well as some key differences between Stuxnet and the cases considered here. In both Stuxnet and our case studies, the attacker employed limited deviations to avoid detection; we modelled this using the concept of an attacker budget. Both also involved fake sensor signals (ΔT_0^t) and manipulated calibration values ($\Delta\boldsymbol{\theta}$, $\Delta\mathbf{c}$, $\Delta\beta$, $\Delta\gamma$). Our examples each had two different kinds of attack modes, and for the fan optimization, there were two different attack objectives for one of the modes, but these all involved negatively impacting the defender's control system in some way. Finally, Stuxnet and the attacks considered in this paper all utilized deep knowledge of an automated decision-making system to determine how to perform the attack.

There are two primary sets of differences between this paper's case studies and Stuxnet. Firstly, to the best of our knowledge, Stuxnet was not optimization-based, and the centrifuge control systems did not employ optimal control, so the decision-making processes for both the attacker and the defender were different than in our paper. Secondly, Stuxnet actually overrode the control signals and software to manipulate the centrifuges [5], whereas our attacks only altered sensor and calibration data. If we were trying to model the Stuxnet attack itself, these discrepancies would be problematic. Given the more general nature of our investigation here, though, this is less of an issue, and the key similarities identified above are ones we believe to be relevant to a wide range of control systems that might be threatened by cyber attacks in general and APTs in particular.

6.2 Scalability Considerations

A big question in applying these techniques to real-world problems is scalability. These problems were relatively small; even the 20-step HVAC problem had only 120 variables (six per time step) in the baseline problem. How easy would it be to propagate the optimality conditions and solve the resulting MPECs for larger systems? The answer has two parts. Firstly, if the optimality conditions are necessary but not sufficient, as in general continuous NLP problems, propagating the optimality conditions to turn the multi-level optimization into an MPEC may run into difficulties; multiple optima would be one example of this. That being said, the single-zone HVAC system presented here was a nonconvex problem, and it had no such problems. If there are more than two levels to the optimization, that can also cause difficulties, as the lower level optimality conditions compound.

This then leads into the question of tractability. Adding the dual variables of lower level optimizations to the problem description in order to solve the system as an MPEC can greatly increase the number of variables involved; having multiple levels may exacerbate the issue. However, it is sometimes possible to simplify the optimality conditions and thereby remove some of the dual variables (as was done for the fan optimization problem). The NLP sequential relaxation of the MPEC also scales well and handles the complementarity constraints efficiently. On the whole, the scalability of this approach will depend on the problem in question and how many levels of

(mis)perception are of interest. Hypergames where the individual players' games are differentiable, convex optimization problems are likely to have the greatest amount of success with this approach. Problems with known or constant active constraint sets will also generally be more amenable to the multi-level optimizations than problems with varying active sets.

6.3 Future Work

Some authors writing on Stuxnet suggest the use of heuristics to identify attacks [2, 14]. One area of future work would be to take existing research on learning in repeated hypergames [6, 32] and apply it to this context. For this, we would consider the defender's ability to detect attacks as well as the attacker's behaviour when the non-detection constraint is endogenous rather than exogenous; the attacker budget imposed here would be an example of an exogenous detection constraint. Another area of interest would be the defender's decision-making more generally. Given the possibility of attack and the potential consequences (as calculated in this paper), how should a defender respond if an attack is undetectable beforehand? Hypergame results here should enable us to to evaluate and prescribe control policies more broadly. Finally, we intend to extend this work to larger, real-world systems. Working on such systems may then also involve more complicated attacker manipulations, but we anticipate being able to use the same techniques demonstrated here.

7 CONCLUSIONS

In this paper, we showed how hypergames can be extended to situations with continuous and time-varying variables. That extension allowed us to consider the effects of adversarial perturbations in an optimal control context, which can give us insights into the control aspects of a Stuxnet-like attack. Manipulating constraints can be a more effective attacker strategy than directly manipulating objective function parameters; our analytical results showed why we would expect this to be true more generally. Moreover, the attacker need not change the underlying system in any way to attack successfully – it may be sufficient to deceive the defender controlling the system. It is possible to scale our approach up to larger systems, but the ability to do so will depend on the characteristics of the system in question, and we identified several characteristics that will make larger systems amenable to hypergame analysis.

ACKNOWLEDGMENTS

This work was supported by the DOE EERE Building Technologies Office, Sensors and Controls program.

REFERENCES

[1] Craig Bakker, Arnab Bhattacharya, Samrat Chatterjee, and Draguna L Vrabie. 2019. Learning and Information Manipulation: Repeated Hypergames for Cyber-Physical Security. *IEEE Control Systems Letters* 4, 2 (2019), 295–300.

[2] Boldizsár Bencsáth, Gábor Pék, Levente Buttyán, and Mark Felegyhazi. 2012. The cousins of stuxnet: Duqu, flame, and gauss. *Future Internet* 4, 4 (2012), 971–1003.

[3] Peter G Bennett. 1980. Hypergames: developing a model of conflict. *Futures* 12, 6 (1980), 489–507.

[4] Colin F Camerer, Teck-Hua Ho, and Juin-Kuan Chong. 2004. A cognitive hierarchy model of games. *The Quarterly Journal of Economics* 119, 3 (2004), 861–898.

[5] Nicolas Falliere, Liam O Murchu, and Eric Chien. 2011. W32. stuxnet dossier. *White paper, Symantec Corp., Security Response* 5, 6 (2011), 29.

[6] Bahman Gharesifard and Jorge Cortés. 2010. Evolution of the perception about the opponent in hypergames. In *Decision and Control (CDC), 2010 49th IEEE Conference on*. IEEE, 1076–1081.

[7] Bahman Gharesifard and Jorge Cortés. 2011. Exploration of misperceptions in hypergames. In *Communication, Control, and Computing (Allerton), 2011 49th Annual Allerton Conference on*. IEEE, 1565–1570.

[8] Ian Graham, Fiona O'Doherty, Alan McKinnon, and Lynne Baxter. 1992. Hypergame analysis of the stability of relationships between computerbased logistics systems. *International Journal of Production Economics* 26, 1-3 (1992), 303–310.

[9] Christopher N Gutierrez, Saurabh Bagchi, H Mohammed, and Jeff Avery. 2015. Modeling Deception In Information Security As A Hypergame–A Primer. In *Proceedings of the 16th Annual Information Security Symposium*. CERIAS-Purdue University, 41.

[10] James Thomas House and George Cybenko. 2010. Hypergame theory applied to cyber attack and defense. In *Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense IX*, Vol. 7666. International Society for Optics and Photonics, 766604.

[11] Gerry Howser and Bruce McMillin. 2014. A modal model of stuxnet attacks on cyber-physical systems: A matter of trust. In *2014 Eighth International Conference on Software Security and Reliability (SERE)*. IEEE, 225–234.

[12] Daniel Kahneman and Amos Tversky. 1979. Prospect Theory: An Analysis of Decision under Risk. *Econometrica* 47, 2 (1979), 263–292.

[13] Takafumi Kanazawa, Toshimitsu Ushio, and Tatsushi Yamasaki. 2007. Replicator dynamics of evolutionary hypergames. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans* 37, 1 (2007), 132–138.

[14] Stamatis Karnouskos. 2011. Stuxnet worm impact on industrial cyber-physical system security. In *IECON 2011-37th Annual Conference on IEEE Industrial Electronics Society*. IEEE, 4490–4494.

[15] Nicholas S Kovach, Alan S Gibson, and Gary B Lamont. 2015. Hypergame theory: a model for conflict, misperception, and deception. *Game Theory* 2015 (2015).

[16] W. Lin, Y. Huang, and Q. Wu. 2010. Study on monitor system of heating ventilation air conditioning based on LonWorks technology. In *Proceedings of the International Technology and Innovation Conference*. 1–5.

[17] MATLAB. 2017. *version 9.2.0 (R2017a)*. The MathWorks Inc., Natick, Massachusetts.

[18] Richard D McKelvey and Thomas R Palfrey. 1995. Quantal response equilibria for normal form games. *Games and economic behavior* 10, 1 (1995), 6–38.

[19] M. Mirsky, Y.and Guri and Y. Elovici. 2017. HVACKer: Bridging the Air-Gap by Attacking the Air Conditioning System. *eprint arXiv: 1703.10454* (2017).

[20] Arash Nourian and Stuart Madnick. 2018. A systems theoretic approach to the security threats in cyber physical systems applied to stuxnet. *IEEE Transactions on Dependable and Secure Computing* 15, 1 (2018), 2–13.

[21] Dmitry A Novikov and Alexander G Chkhartishvili. 2014. *Reflexion and control: mathematical models*. CRC Press.

[22] Norio Okada, Keith W Hipel, and Yoshiharu Oka. 1985. Hypergame analysis of the Lake Biwa conflict. *Water Resources Research* 21, 7 (1985), 917–926.

[23] A. Roth. 2002. The Economist as Engineer: Game Theory, Experimentation, and Computation as Tools for Design Economics. *Econometrica* 70, 4 (2002), 1341–1378.

[24] Carlos Ruiz, Antonio J Conejo, J David Fuller, Steven A Gabriel, and Benjamin F Hobbs. 2014. A tutorial review of complementarity models for decision-making in energy markets. *EURO Journal on Decision Processes* 2, 1-2 (2014), 91–120.

[25] József Sákovics. 2001. Games of incomplete information without common knowledge priors. *Theory and decision* 50, 4 (2001), 347–366.

[26] Tuomas W Sandholm and Victor RT Lesser. 1997. Coalitions among computationally bounded agents. *Artificial intelligence* 94, 1-2 (1997), 99–137.

[27] Todd Sandler. 2003. Terrorism & game theory. *Simulation & Gaming* 34, 3 (2003), 319–337.

[28] Yasuo Sasaki. 2008. preservation of misperceptions–stability analysis of hypergames. In *Proceedings of the 52nd Annual Meeting of the ISSS-2008, Madison, Wisconsin*, Vol. 3.

[29] Yasuo Sasaki and Kyoichi Kijima. 2012. Hypergames and bayesian games: A theoretical comparison of the models of games with incomplete information. *Journal of Systems Science and Complexity* 25, 4 (2012), 720–735.

[30] Yasuo Sasaki and Kyoichi Kijima. 2016. Hierarchical hypergames and Bayesian games: A generalization of the theoretical comparison of hypergames and Bayesian games considering hierarchy of perceptions. *Journal of Systems Science and Complexity* 29, 1 (2016), 187–201.

[31] Dale O. Stahl and Paul W. Wilson. 1995. On Players' Models of Other Players: Theory and Experimental Evidence. *Games and Economic Behavior* 10, 1 (1995), 218 – 254. https://doi.org/10.1006/game.1995.1031

[32] Shingo Takahashi, Naotaka Hinago, Takehiro Inohara, and Bumpei Nakano. 1999. Evolutionary approach to three-person hypergame situation. In *Systems, Man, and Cybernetics, 1999. IEEE SMC'99 Conference Proceedings. 1999 IEEE International Conference on*, Vol. 4. IEEE, 254–259.

A STATIC FAN OPTIMIZATION CALCULATIONS

A.1 Baseline Problem

The baseline defender optimization is

$$\min_{m,p} \theta_1 m + \theta_2 m^2 + \theta_3 p \tag{142}$$

$$\frac{1}{2}\left[(m - c_m)^2 + \left(p - c_p\right)^2 - c_r^2\right] \leq 0 \tag{143}$$

Note that we include the 1/2 factor in the constraint to cancel out factors of 2 that appear when taking the derivative of that constraint. The objective function and inequality constraint are both convex functions, so the optimization is a convex problem and the KKT conditions are necessary and sufficient to define problem optima. If we define the Lagrangian as L and use λ as the dual variable associated with the inequality constraint, we get the following optimality conditions:

$$\frac{\partial L}{\partial m} = \theta_1 + 2\theta_2 m + (m - c_m)\,\lambda = 0 \tag{144}$$

$$\frac{\partial L}{\partial p} = \theta_3 + \left(p - c_p\right)\lambda = 0 \tag{145}$$

$$\frac{1}{2}\left[(m - c_m)^2 + \left(p - c_p\right)^2 - c_r^2\right]\lambda = 0 \tag{146}$$

For these equations to be satisfied, $\lambda \neq 0$. Since $\lambda \geq 0$, this ensures that $p < c_p$. Moreover, if c_r is sufficiently small, $m > 0$, and thus $m < c_m$. We can then get rid of λ by substitution, and we are left with

$$\left(p - c_p\right)(\theta_1 + 2\theta_2 m) - (m - c_m)\,\theta_3 = 0 \tag{147}$$

$$\frac{1}{2}\left[(m - c_m)^2 + \left(p - c_p\right)^2 - c_r^2\right] = 0 \tag{148}$$

A.2 Objective Function Manipulation

A.2.1 Attacker Manipulates True/Physical Properties and Defender Knows. The min-max problem is

$$\min_{m,p} \max_{\Delta\theta_i} (\theta_1 + \Delta\theta_1)\, m + (\theta_2 + \Delta\theta_2)\, m^2 + (\theta_3 + \Delta\theta_3)\, p \tag{149}$$

$$\frac{1}{2}\left[(m - c_m)^2 + \left(p - c_p\right)^2 - c_r^2\right] \leq 0 \tag{150}$$

$$\frac{1}{2}\sum_i \Delta\theta_i^2 \leq \delta_{\theta,max} \tag{151}$$

We can use the attacker's KKT conditions to transform the min-max problem into a pure optimization problem. Define L as the Lagrangian and σ as the dual variable associated with the attacker budget constraint. Then

$$\frac{\partial L}{\partial \Delta\theta_1} = m - \sigma\Delta\theta_1 = 0 \Rightarrow \Delta\theta_1 = \frac{1}{\sigma}m \tag{152}$$

$$\frac{\partial L}{\partial \Delta\theta_2} = m^2 - \sigma\Delta\theta_2 = 0 \Rightarrow \Delta\theta_2 = \frac{1}{\sigma}m^2 \tag{153}$$

$$\frac{\partial L}{\partial \Delta\theta_3} = p - \sigma\Delta\theta_3 = 0 \Rightarrow \Delta\theta_1 = \frac{1}{\sigma}p \tag{154}$$

For finite $\Delta\theta_i$, we require $\sigma \neq 0$. Since we know, by definition, that $\sigma \geq 0$, then $\sigma > 0$. We can therefore parameterize the attacker's decisions in terms of $\tau = 1/\sigma$:

$$\min_{m,p} \max_{\tau} (\theta_1 + m\tau)\, m + \left(\theta_2 + m^2\tau\right) m^2 + (\theta_3 + p\tau)\, p \tag{155}$$

$$\frac{1}{2}\left[(m - c_m)^2 + \left(p - c_p\right)^2 - c_r^2\right] \leq 0 \tag{156}$$

$$\frac{1}{2}\tau^2\left(m^2 + m^4 + p^2\right) \leq \delta_{\theta,max} \tag{157}$$

Given that the last constraint will always be active ($\sigma \neq 0$), we can solve for τ:

$$\tau = \left[\frac{2\delta_{\theta,max}}{m^2 + m^4 + p^2}\right]^{\frac{1}{2}} \tag{158}$$

We are then left with the following defender optimization:

$$\min_{m,p} \theta_1 m + \theta_2 m^2 + \theta_3 p + \left[2\delta_{\theta,max}\left(m^2 + m^4 + p^2\right)\right]^{\frac{1}{2}} \tag{159}$$

$$\frac{1}{2}\left[(m - c_m)^2 + \left(p - c_p\right)^2 - c_r^2\right] \leq 0 \tag{160}$$

A.2.2 Attacker Manipulates Defender Perceptions, Defender Unaware. The attacker is solving the problem

$$\max_{\Delta\theta_i} \theta_1 m + \theta_2 m^2 + \theta_3 p \tag{161}$$

$$\frac{1}{2}\sum_i \Delta\theta_i^2 \leq \delta_{\theta,max} \tag{162}$$

subject to the defender optimization

$$\min_{m,p} (\theta_1 + \Delta\theta_1)\, m + (\theta_2 + \Delta\theta_2)\, m^2 + (\theta_3 + \Delta\theta_3)\, p \tag{163}$$

$$\frac{1}{2}\left[(m - c_m)^2 + \left(p - c_p\right)^2 - c_r^2\right] \leq 0 \tag{164}$$

The optimality conditions of the defender problem are the same as in the baseline case except that we replace θ_i with $\hat{\theta}_i = \theta_i + \Delta\theta_i$:

$$\left(p - c_p\right)\left(\hat{\theta}_1 + 2\hat{\theta}_2 m\right) - \left(m - c_m\right)\hat{\theta}_3 = 0 \tag{165}$$

$$\frac{1}{2}\left[\left(m - c_m\right)^2 + \left(p - c_p\right)^2 - c_r^2\right] = 0 \tag{166}$$

This then results in the optimization problem for the attacker:

$$\max_{\Delta\theta_i, m, p} \ \theta_1 m + \theta_2 m^2 + \theta_3 p \tag{167}$$

$$\frac{1}{2}\left[\left(m - c_m\right)^2 + \left(p - c_p\right)^2 - c_r^2\right] = 0 \ \ (\rho) \tag{168}$$

$$\frac{1}{2}\sum_i \Delta\theta_i^2 \leq \delta_{\theta,max} \ \ (\mu) \tag{169}$$

$$\left(p - c_p\right)\left(\hat{\theta}_1 + 2\hat{\theta}_2 m\right) - \left(m - c_m\right)\hat{\theta}_3 = 0 \ \ (\lambda) \tag{170}$$

where the dual variable for each constraint is shown in brackets next to that constraint. We can solve this directly as an optimization, but we can also use the optimality conditions to calculate $\Delta\theta_i$. Define L as the optimization's Lagrangian. Then

$$\frac{\partial L}{\partial \Delta\theta_1} = -\mu\Delta\theta_1 + \left(p - c_p\right)\lambda = 0 \tag{171}$$

$$\frac{\partial L}{\partial \Delta\theta_2} = -\mu\Delta\theta_2 + 2\left(p - c_p\right)m\lambda = 0 \tag{172}$$

$$\frac{\partial L}{\partial \Delta\theta_3} = -\mu\Delta\theta_3 - \left(m - c_m\right)\lambda = 0 \tag{173}$$

If we use $\tau = \lambda/\mu$, we get

$$\Delta\theta_1 = \tau\left(p - c_p\right) \tag{174}$$

$$\Delta\theta_2 = 2\tau\left(p - c_p\right)m \tag{175}$$

$$\Delta\theta_3 = -\tau\left(m - c_m\right) \tag{176}$$

$$\tau = \left[\frac{2\delta_{\theta,max}}{\left(p - c_p\right)^2 + \left(2\left(p - c_p\right)m\right)^2 + \left(m - c_m\right)^2}\right]^{\frac{1}{2}} = \left[\frac{2\delta_{\theta,max}}{4\left(p - c_p\right)^2 m^2 + c_r^2}\right]^{\frac{1}{2}} \tag{177}$$

We know that $\mu > 0$, but in principle λ could be positive or negative. When we solve the optimization directly (using the parameter values specified in the main body of the paper), we find that $\lambda > 0$. Given that $p - c_p < 0$ and $m - c_m < 0$, this means that the attacker decreases the defender-perceived values of θ_1 and θ_2 while raising the defender-perceived value of θ_3. This in turn results in an increased value of m and a decreased value of p (relative to the unperturbed case). The case where $\lambda < 0$ would correspond to the opposite behaviour.

Both options produce local maxima, for the attacker, but in general, we would expect the $\lambda > 0$ option to produce a higher payoff: the objective is linear in p but quadratic in m, so increasing m would often provide a greater payoff than increasing p. We do not have a proof delineating when this is the case, but we would expect this not to be the case only for small values of θ_1 and θ_2 (relative to θ_3). For the c_m, c_p, c_r, and $\delta_{\theta,max}$ values considered in this paper, we can empirically

verify that for $\theta_1 \in [0.5, 3.5]$, $\theta_2 \in [0.5, 3.5]$, and $\theta_3 \in [0.5, 3.5]$, the $\lambda > 0$ option provides a larger attacker payoff. This domain encompasses all of the true θ_i values that an attacker could manipulate to produce the $\hat{\theta}_i$ values observed by the defender. Since the defender knows the attacker budget, if the defender believes that the attacker is attempting to perturb θ_i, the defender can know that the attacker is employing the attack where $\tau > 0$.

A.2.3 Attacker Manipulates Defender Perceptions, Defender is Aware. Using the results from the previous section, the defender can reverse engineer the true θ_i values from the perceived values $\hat{\theta}_i$ if the defender is aware of an attack. The defender believes that $\hat{\theta}_i$ has been calculated by an attacker solving the problem in Appendix A.2.2. Therefore the defender's optimization is

$$\min_{m,p} \left(\hat{\theta}_1 - \Delta\theta_1\right) m + \left(\hat{\theta}_2 - \Delta\theta_2\right) m^2 + \left(\hat{\theta}_3 - \Delta\theta_3\right) p \tag{178}$$

$$\frac{1}{2}\left[(m - c_m)^2 + \left(p - c_p\right)^2 - c_r^2\right] \leq 0 \tag{179}$$

$$\Delta\theta_1 = \tau\left(\hat{p} - c_p\right) \tag{180}$$

$$\Delta\theta_2 = 2\tau\left(\hat{p} - c_p\right)\hat{m} \tag{181}$$

$$\Delta\theta_3 = -\tau\left(\hat{m} - c_m\right) \tag{182}$$

$$\tau = \left[\frac{2\delta_{\theta,max}}{4\left(\hat{p} - c_p\right)^2 \hat{m}^2 + c_r^2}\right]^{\frac{1}{2}} \tag{183}$$

$$\frac{1}{2}\left[(\hat{m} - c_m)^2 + \left(\hat{p} - c_p\right)^2 - c_r^2\right] = 0 \tag{184}$$

$$\left(\hat{p} - c_p\right)\left(\hat{\theta}_1 + 2\hat{\theta}_2\hat{m}\right) - \left(\hat{m} - c_m\right)\hat{\theta}_3 = 0 \tag{185}$$

where $\hat{m}$ and $\hat{p}$ are the decision variable values that the defender thinks that the attacker expects the defender to employ. Note that it is possible to solve

$$\frac{1}{2}\left[(\hat{m} - c_m)^2 + \left(\hat{p} - c_p\right)^2 - c_r^2\right] = 0 \tag{186}$$

$$\left(\hat{p} - c_p\right)\left(\hat{\theta}_1 + 2\hat{\theta}_2\hat{m}\right) - \left(\hat{m} - c_m\right)\hat{\theta}_3 = 0 \tag{187}$$

once with the known $\hat{\theta}_i$ values and then use those to calculate $\Delta\theta_i$ – these do not depend on m or p. Once this calculation has been performed, we are left with the original convex defender optimization problem.

A.2.4 Attacker Manipulates Defender Perceptions, Defender is Aware, Attacker Knows that Defender is Aware. This problem leads us to a multi-level optimization problem. At level 1, we have the attacker optimization

$$\max_{\Delta\theta_i} \theta_1 m + \theta_2 m^2 + \theta_3 p \tag{188}$$

$$\frac{1}{2}\sum_i \Delta\theta_i \leq \delta_{\theta,max} \tag{189}$$

$$\hat{\theta}_i = \theta_i + \Delta\theta_i \tag{190}$$

At the next level (level 2), we have the defender optimization. The defender performs his optimization based on the belief that the values he perceives, $\hat{\theta}_i$ has been perturbed by an attacker solving the problem in Appendix A.2.2. Therefore the defender's optimization is

$$\min_{m,p} \left(\hat{\theta}_1 - \Delta\hat{\theta}_1\right) m + \left(\hat{\theta}_2 - \Delta\hat{\theta}_2\right) m^2 + \left(\hat{\theta}_3 - \Delta\hat{\theta}_3\right) p \tag{191}$$

$$\frac{1}{2}\left[(m - c_m)^2 + \left(p - c_p\right)^2 - c_r^2\right] \leq 0 \tag{192}$$

$$\Delta\theta_1 = \tau\left(\hat{p} - c_p\right) \tag{193}$$

$$\Delta\theta_2 = 2\tau\left(\hat{p} - c_p\right)\hat{m} \tag{194}$$

$$\Delta\theta_3 = -\tau\left(\hat{m} - c_m\right) \tag{195}$$

$$\tau = \left[\frac{2\delta_{\theta,max}}{4\left(\hat{p} - c_p\right)^2 \hat{m}^2 + c_r^2}\right]^{\frac{1}{2}} \tag{196}$$

$$\frac{1}{2}\left[(\hat{m} - c_m)^2 + \left(\hat{p} - c_p\right)^2 - c_r^2\right] = 0 \tag{197}$$

$$\left(\hat{p} - c_p\right)\left(\hat{\theta}_1 + 2\hat{\theta}_2\hat{m}\right) - \left(\hat{m} - c_m\right)\hat{\theta}_3 = 0 \tag{198}$$

The defender's optimality conditions (level 2) are then:

$$\frac{1}{2}\left[(\hat{m} - c_m)^2 + \left(\hat{p} - c_p\right)^2 - c_r^2\right] = 0 \tag{199}$$

$$\left(\hat{p} - c_p\right)\left[\left(\theta_1 + \Delta\theta_1\right) + 2\left(\theta_2 + \Delta\theta_2\right)\hat{m}\right] - \left(\hat{m} - c_m\right)\left(\theta_3 + \Delta\theta_3\right) = 0 \tag{200}$$

$$\frac{1}{2}\left[(m - c_m)^2 + \left(p - c_p\right)^2 - c_r^2\right] = 0 \tag{201}$$

$$\begin{aligned}\left(p - c_p\right)\Big[\left(\theta_1 + \Delta\theta_1 - \tau\left(\hat{p} - c_p\right)\right) + 2\left(\theta_2 + \Delta\theta_2 - 2\left(\hat{p} - c_p\right)\hat{m}\tau\right)m\Big] \\ - \left(m - c_m\right)\left(\theta_3 + \Delta\theta_3 + \tau\left(\hat{m} - c_m\right)\right) = 0\end{aligned} \tag{202}$$

$$\tau = \left[\frac{2\delta_{\theta,max}}{4\left(\hat{p} - c_p\right)^2 \hat{m}^2 + c_r^2}\right]^{\frac{1}{2}} \tag{203}$$

The attacker's optimization (level 1) is then

$$\max_{\Delta\theta_i} \theta_1 m + \theta_2 m^2 + \theta_3 p \tag{204}$$

$$\frac{1}{2}\sum_i \Delta\theta_i \leq \delta_{\theta,max} \tag{205}$$

$$\frac{1}{2}\left[(\hat{m} - c_m)^2 + \left(\hat{p} - c_p\right)^2 - c_r^2\right] = 0 \tag{206}$$

$$\left(\hat{p} - c_p\right)\left[\left(\theta_1 + \Delta\theta_1\right) + 2\left(\theta_2 + \Delta\theta_2\right)\hat{m}\right] - \left(\hat{m} - c_m\right)\left(\theta_3 + \Delta\theta_3\right) = 0 \tag{207}$$

$$\frac{1}{2}\left[(m - c_m)^2 + \left(p - c_p\right)^2 - c_r^2\right] = 0 \tag{208}$$

$$\begin{aligned}\left(p-c_p\right)\left[\left(\theta_1+\Delta\theta_1-\tau\left(\hat{p}-c_p\right)\right)+2\left(\theta_2+\Delta\theta_2-2\left(\hat{p}-c_p\right)\hat{m}\tau\right)m\right]\\ -\left(m-c_m\right)\left(\theta_3+\Delta\theta_3+\tau\left(\hat{m}-c_m\right)\right)=0\end{aligned} \tag{209}$$

$$\tau=\left[\frac{2\delta_{\theta,max}}{4\left(\hat{p}-c_p\right)^2\hat{m}^2+c_r^2}\right]^{\frac{1}{2}} \tag{210}$$

The attacker optimization may not be convex, but each $\Delta\theta_i$ value corresponds to a single $(\hat{m},\hat{p},m,p)$ tuple. We can show by via a sequential analysis. The equations

$$\frac{1}{2}\left[\left(\hat{m}-c_m\right)^2+\left(\hat{p}-c_p\right)^2-c_r^2\right]=0 \tag{211}$$

$$\left(\hat{p}-c_p\right)\left[\left(\theta_1+\Delta\theta_1\right)+2\left(\theta_2+\Delta\theta_2\right)\hat{m}\right]-\left(\hat{m}-c_m\right)\left(\theta_3+\Delta\theta_3\right)=0 \tag{212}$$

define a unique solution $(\hat{m},\hat{p})$ to an instance of the unaware defender optimization. By the logic employed in the previous section, we can calculate $\Delta\hat{\theta}_i$ values from that, which then in turn defines m and p as the unique solution to

$$\frac{1}{2}\left[\left(m-c_m\right)^2+\left(p-c_p\right)^2-c_r^2\right]=0 \tag{213}$$

$$\begin{aligned}\left(p-c_p\right)\left[\left(\theta_1+\Delta\theta_1-\tau\left(\hat{p}-c_p\right)\right)+2\left(\theta_2+\Delta\theta_2-2\left(\hat{p}-c_p\right)\hat{m}\tau\right)m\right]\\ -\left(m-c_m\right)\left(\theta_3+\Delta\theta_3+\tau\left(\hat{m}-c_m\right)\right)=0\end{aligned} \tag{214}$$

$$\tau=\left[\frac{2\delta_{\theta,max}}{4\left(\hat{p}-c_p\right)^2\hat{m}^2+c_r^2}\right]^{\frac{1}{2}} \tag{215}$$

A.3 Constraint Manipulation

In this section, for the sake of simplicity, we assume that the attacker is only manipulating the constraint parameters (not the objective function parameters). These constraint manipulations take the form of

$$\hat{c}_m=c_m+\Delta c_m \tag{216}$$

$$\hat{c}_p=c_p+\Delta c_p \tag{217}$$

$$\hat{c}_r=c_r-\Delta c_r \tag{218}$$

The attacker is also subject to an attack budget of

$$\frac{1}{2}\left(\Delta c_m^2+\Delta c_p^2+\Delta c_r^2\right)=\frac{1}{2}\sum_i\Delta c_i^2\leq\delta_{c,max} \tag{219}$$

A.3.1 Attacker Manipulates Defender Perceptions, Defender Unaware. The attacker's optimization is

$$\max_{\Delta c_i}\theta_1 m+\theta_2 m^2+\theta_3 p \tag{220}$$

$$\frac{1}{2}\sum_i\Delta c_i^2\leq\delta_{c,max} \tag{221}$$

subject to the defender optimization

$$\min_{m,p} \theta_1 m + \theta_2 m^2 + \theta_3 p \tag{222}$$

$$\frac{1}{2}\left[(m - c_m - \Delta c_m)^2 + \left(p - c_p - \Delta c_p\right)^2 - (c_r - \Delta c_r)^2\right] \leq 0 \tag{223}$$

The defender optimality conditions are

$$\left(p - c_p - \Delta c_p\right)(\theta_1 + 2\theta_2 m) - (m - c_m - \Delta c_m)\,\theta_3 = 0 \tag{224}$$

$$\frac{1}{2}\left[(m - c_m - \Delta c_m)^2 + \left(p - c_p - \Delta c_p\right)^2 - (c_r - \Delta c_r)^2\right] = 0 \tag{225}$$

and we are left with the attacker optimization

$$\max_{\delta_i} \theta_1 m + \theta_2 m^2 + \theta_3 p \tag{226}$$

$$\frac{1}{2}\sum_i \Delta c_i^2 \leq \delta_{c,max} \tag{227}$$

$$\left(p - c_p - \Delta c_p\right)(\theta_1 + 2\theta_2 m) - (m - c_m - \Delta c_m)\,\theta_3 = 0 \tag{228}$$

$$\frac{1}{2}\left[(m - c_m - \Delta c_m)^2 + \left(p - c_p - \Delta c_p\right)^2 - (c_r - \Delta c_r)^2\right] = 0 \tag{229}$$

A.3.2 Attacker Manipulates Defender Perceptions, Defender is Aware. The defender's optimization is

$$\min_{m,p} \theta_1 m + \theta_2 m^2 + \theta_3 p \tag{230}$$

$$\frac{1}{2}\left[(m - \hat{c}_m + \Delta c_m)^2 + \left(p - \hat{c}_p + \Delta c_p\right)^2 - (\hat{c}_r + \Delta c_r)^2\right] \leq 0 \tag{231}$$

where $\hat{c}_m$, $\hat{c}_p$, and $\hat{c}_r$ are the quantities that the defender perceives (which the defender believes to have been manipulated by the attacker). The true parameter values are unknown, but the Δc_i values are calculated by solving the attacker problem from the previous section:

$$\max_{\hat{m},\hat{p},\Delta c_i} \theta_1 \hat{m} + \theta_2 \hat{m}^2 + \theta_3 \hat{p} \tag{232}$$

$$\frac{1}{2}\sum_i \Delta c_i^2 \leq \delta_{c,max} \;\; (\mu) \tag{233}$$

$$\left(\hat{p} - c_p - \Delta c_p\right)(\theta_1 + 2\theta_2 \hat{m}) - (\hat{m} - c_m - \Delta c_m)\,\theta_3 = 0 \;\; (\sigma) \tag{234}$$

$$\frac{1}{2}\left[(\hat{m} - c_m - \Delta c_m)^2 + \left(\hat{p} - c_p - \Delta c_p\right)^2 - (c_r - \Delta c_r)^2\right] = 0 \;\; (\rho) \tag{235}$$

where the dual variables for each constraint are shown in brackets beside the equation Define L as the Lagrangian for this problem. The optimality conditions are then

$$\frac{\partial L}{\partial \hat{m}} = \theta_1 + 2\theta_2\hat{m} - \sigma\left(2\left(\hat{p} - c_p - \Delta c_p\right)\theta_2 - \theta_3\right) - \rho\left(\hat{m} - c_m - \Delta c_m\right) = 0 \tag{236}$$

$$\frac{\partial L}{\partial \hat{p}} = \theta_3 - \sigma\left(\theta_1 + 2\theta_2\hat{m}\right) - \rho\left(\hat{p} - c_p - \Delta c_p\right) = 0 \tag{237}$$

$$\frac{\partial L}{\partial \Delta c_m} = -\mu\Delta c_m - \sigma\theta_3 + \rho\left(\hat{m} - c_m - \Delta c_m\right) = 0 \tag{238}$$

$$\frac{\partial L}{\partial \Delta c_p} = -\mu\Delta c_p + \sigma\left(\theta_1 + 2\theta_2\hat{m}\right) + \rho\left(\hat{p} - c_p - \Delta c_p\right) = 0 \tag{239}$$

$$\frac{\partial L}{\partial \Delta c_r} = -\mu\Delta c_r - \rho\left(c_r - \Delta c_r\right) = 0 \tag{240}$$

If we take the first two equations and simplify using $\hat{c}_i$, we get

$$\theta_1 + 2\theta_2\hat{m} - \sigma\left(2\left(\hat{p} - \hat{c}_p\right)\theta_2 - \theta_3\right) - \rho\left(\hat{m} - \hat{c}_m\right) = 0 \tag{241}$$

$$\theta_3 - \sigma\left(\theta_1 + 2\theta_2\hat{m}\right) - \rho\left(\hat{p} - \hat{c}_p\right) = 0 \tag{242}$$

We can set this up to solve for σ and ρ:

$$\begin{bmatrix} 2\left(\hat{p} - \hat{c}_p\right)\theta_2 - \theta_3 & \hat{m} - \hat{c}_m \\ \theta_1 + 2\theta_2\hat{m} & \hat{p} - \hat{c}_p \end{bmatrix} \begin{Bmatrix} \sigma \\ \rho \end{Bmatrix} = \begin{Bmatrix} \theta_1 + 2\theta_2\hat{m} \\ \theta_3 \end{Bmatrix} \tag{243}$$

We can get closed-form expressions for σ and ρ by solving this 2x2 system analytically, and we can then use these expressions to calculate our Δc_i values in terms of $\tau = 1/\mu$:

$$\Delta c_p = \tau\theta_3 \tag{244}$$

$$\Delta c_m = \tau\left[\rho\left(\hat{m} - \hat{c}_m\right) - \sigma\theta_3\right] \tag{245}$$

$$\Delta c_r = -\tau\rho\hat{c}_r \tag{246}$$

The constraint on the sum of squared Δc_i values then lets us calculate a value for τ:

$$\tau^2\left[\theta_3^2 + \left(\rho\left(\hat{m} - \hat{c}_m\right) - \sigma\theta_3\right)^2 + \rho^2\hat{c}_r^2\right] = 2\delta_{c,max} \tag{247}$$

$$\tau = \left[\frac{2\delta_{c,max}}{\theta_3^2 + \left[\rho\left(\hat{m} - \hat{c}_m\right) - \sigma\theta_3\right]^2 + \rho^2\hat{c}_r^2}\right]^{\frac{1}{2}} \tag{248}$$

and thus we have closed-form expressions for the Δc_i values that can then be plugged back into the original defender optimization without needing to know the true c_i values. Note that the defender can perform these calculations without knowing the true c_i ahead of time – it is sufficient to know $\hat{c}_i$.

A.3.3 Attacker Manipulates Defender Perceptions, Defender is Aware, Attacker Knows that Defender is Aware. The attacker's optimization is

$$\max_{\Delta c_i} \theta_1 m + \theta_2 m^2 + \theta_3 p \tag{249}$$

$$\frac{1}{2}\sum_i \Delta c_i^2 \leq \delta_{c,max} \tag{250}$$

$$\hat{c}_m = c_m + \Delta c_m \tag{251}$$

$$\hat{c}_p = c_p + \Delta c_p \tag{252}$$

$$\hat{c}_r = c_r - \Delta c_r \tag{253}$$

subject to the defender optimization from the previous section. The optimality conditions for the defender's optimization are

$$\left(p - \hat{c}_p + \Delta\hat{c}_p\right)\left(\theta_1 + 2\theta_2 m\right) - \left(m - \hat{c}_m + \Delta\hat{c}_m\right)\theta_3 = 0 \tag{254}$$

$$\frac{1}{2}\left[\left(m - \hat{c}_m + \Delta\hat{c}_m\right)^2 + \left(p - \hat{c}_p + \Delta\hat{c}_p\right)^2 - \left(\hat{c}_r + \Delta\hat{c}_r\right)^2\right] = 0 \tag{255}$$

where

$$\Delta\hat{c}_p = \tau\theta_3 \tag{256}$$

$$\Delta\hat{c}_m = \tau\left[\rho\left(\hat{m} - \hat{c}_m\right) - \sigma\theta_3\right] \tag{257}$$

$$\Delta\hat{c}_r = -\tau\rho\hat{c}_r \tag{258}$$

$$\tau = \left[\frac{2\delta_{c,max}}{\theta_3^2 + \left[\rho\left(\hat{m} - \hat{c}_m\right) - \sigma\theta_3\right]^2 + \rho^2\hat{c}_r^2}\right]^{\frac{1}{2}} \tag{259}$$

$$\begin{bmatrix} 2\left(\hat{p} - \hat{c}_p\right)\theta_2 - \theta_3 & \hat{m} - \hat{c}_m \\ \theta_1 + 2\theta_2\hat{m} & \hat{p} - \hat{c}_p \end{bmatrix}\begin{Bmatrix} \sigma \\ \rho \end{Bmatrix} = \begin{Bmatrix} \theta_1 + 2\theta_2\hat{m} \\ \theta_3 \end{Bmatrix} \tag{260}$$

$$\left(\hat{p} - \hat{c}_p\right)\left(\theta_1 + 2\theta_2\hat{m}\right) - \left(\hat{m} - \hat{c}_m\right)\theta_3 = 0 \tag{261}$$

$$\frac{1}{2}\left[\left(\hat{m} - \hat{c}_m\right)^2 + \left(\hat{p} - \hat{c}_p\right)^2 - \hat{c}_r^2\right] = 0 \tag{262}$$

A.3.4 Attacker Manipulates Defender to Break System, Defender is Unaware. In this case, the attacker wants to cause the defender to deviate maximally from the constraint $\frac{1}{2}\left[\left(m - c_m\right)^2 + \left(p - c_p\right)^2 - c_r^2\right] \leq 0$ in the interest of causing a catastrophic failure. The attacker's optimization is

$$\max_{\Delta c_i} \frac{1}{2}\left[\left(m - c_m\right)^2 + \left(p - c_p\right)^2 - c_r^2\right] \tag{263}$$

$$\frac{1}{2}\sum_i \Delta c_i^2 \leq \delta_{c,max} \tag{264}$$

$$\left(p - c_p - \Delta c_p\right)\left(\theta_1 + 2\theta_2 m\right) - \left(m - c_m - \Delta c_m\right)\theta_3 = 0 \tag{265}$$

$$\frac{1}{2}\left[\left(m - c_m - \Delta c_m\right)^2 + \left(p - c_p - \Delta c_p\right)^2 - \left(c_r - \Delta c_r\right)^2\right] = 0 \tag{266}$$

A.3.5 Attacker Manipulates Defender to Break System, Defender Knows. The defender's optimization is

$$\min \theta_1 m + \theta_2 m^2 + \theta_3 p \tag{267}$$

$$\frac{1}{2}\left[(m - c_m)^2 + \left(p - c_p\right)^2 - c_r^2\right] \leq 0 \tag{268}$$

where the defender only observes $\hat{c}_i$ and needs to calculate Δc_i. The defender knows that the attacker is solving the problem

$$\max_{\Delta c_i} \frac{1}{2}\left[(\hat{m} - c_m)^2 + \left(\hat{p} - c_p\right)^2 - c_r^2\right] \tag{269}$$

$$\frac{1}{2}\sum_i \Delta c_i^2 \leq \delta_{c,max} \ \ (\mu) \tag{270}$$

$$\left(\hat{p} - c_p - \Delta c_p\right)\left(\theta_1 + 2\theta_2 \hat{m}\right) - \left(\hat{m} - c_m - \Delta c_m\right)\theta_3 = 0 \ \ (\sigma) \tag{271}$$

$$\frac{1}{2}\left[(\hat{m} - c_m - \Delta c_m)^2 + \left(\hat{p} - c_p - \Delta c_p\right)^2 - (c_r - \Delta c_r)^2\right] = 0 \ \ (\rho) \tag{272}$$

where the dual variables for each constraint are shown in brackets beside their respective equations. If we define L as the Lagrangian for that problem, the optimality conditions for this problem are

$$\frac{\partial L}{\partial \hat{m}} = \hat{m} - c_m + \sigma\left(2\theta_2\left(\hat{p} - \hat{c}_p\right) + \theta_3\right) - \rho\left(\hat{m} - \hat{c}_m\right) = 0 \tag{273}$$

$$\frac{\partial L}{\partial \hat{p}} = \hat{p} - c_p - \sigma\left(\theta_1 + 2\theta_2 \hat{m}\right) - \rho\left(\hat{p} - \hat{c}_p\right) = 0 \tag{274}$$

$$\frac{\partial L}{\partial \Delta c_m} = -\mu \Delta c_m - \sigma\theta_3 + \rho\left(\hat{m} - \hat{c}_m\right) = 0 \tag{275}$$

$$\frac{\partial L}{\partial \Delta c_p} = -\mu \Delta c_p + \sigma\left(\theta_1 + 2\theta_2 \hat{m}\right) + \rho\left(\hat{p} - \hat{c}_p\right) = 0 \tag{276}$$

$$\frac{\partial L}{\partial \Delta c_r} = -\mu \Delta c_r - \rho \hat{c}_r = 0 \tag{277}$$

We can solve for σ, ρ, and $\tau = 1/\mu$ to get expressions for Δc_i.

$$\Delta c_m = \tau\left(\hat{m} - c_m + 2\theta_2 \sigma\left(\hat{p} - \hat{c}_p\right)\right) \tag{278}$$

$$\Delta c_p = \tau\left(\hat{p} - c_p\right) \tag{279}$$

$$\Delta c_r = -\tau \rho \hat{c}_r \tag{280}$$

$$\begin{Bmatrix} \sigma \\ \rho \end{Bmatrix} = \frac{1}{-\theta_3\left(\hat{p} - \hat{c}_p\right) - \left(\hat{m} - \hat{c}_m\right)\left(\theta_1 + 2\theta_2 \hat{m}\right)} \begin{bmatrix} -\left(\hat{p} - \hat{c}_p\right) & \hat{m} - \hat{c}_m \\ \theta_1 + 2\theta_2 \hat{m} & \theta_3 \end{bmatrix} \begin{Bmatrix} \hat{m} - c_m \\ \hat{p} - c_p \end{Bmatrix} \tag{281}$$

$$\tau = \left[\frac{2\delta_{c,max}}{\left(\hat{m} - c_m + 2\theta_2 \sigma\left(\hat{p} - \hat{c}_p\right)\right)^2 + \left(\hat{p} - c_p\right)^2 + \rho^2 \hat{c}_r^2}\right]^{\frac{1}{2}} \tag{282}$$

Unlike the result in the power maximization case, solving for Δc_i requires knowing c_i, not just $\hat{c}_i$. The defender then has to solve

$$\min \theta_1 m + \theta_2 m^2 + \theta_3 p \tag{283}$$

$$\frac{1}{2}\left[(m - c_m)^2 + \left(p - c_p\right)^2 - c_r^2\right] \leq 0 \tag{284}$$

$$\hat{c}_m = c_m + \tau\left(\hat{m} - c_m + 2\theta_2\sigma\left(\hat{p} - \hat{c}_p\right)\right) \tag{285}$$

$$\hat{c}_p = c_p + \tau\left(\hat{p} - c_p\right) \tag{286}$$

$$\hat{c}_r = c_r + \tau\rho\hat{c}_r \tag{287}$$

$$\left\{ \begin{array}{c} \sigma \\ \rho \end{array} \right\} = \frac{1}{-\theta_3\left(\hat{p} - \hat{c}_p\right) - \left(\hat{m} - \hat{c}_m\right)\left(\theta_1 + 2\theta_2\hat{m}\right)} \left[\begin{array}{cc} -\left(\hat{p} - \hat{c}_p\right) & \hat{m} - \hat{c}_m \\ \theta_1 + 2\theta_2\hat{m} & \theta_3 \end{array} \right] \left\{ \begin{array}{c} \hat{m} - c_m \\ \hat{p} - c_p \end{array} \right\} \tag{288}$$

$$\tau = \left[\frac{2\delta_{c,max}}{\left(\hat{m} - c_m + 2\theta_2\sigma\left(\hat{p} - \hat{c}_p\right)\right)^2 + \left(\hat{p} - c_p\right)^2 + \rho^2\hat{c}_r^2}\right]^{\frac{1}{2}} \tag{289}$$

$$\left(\theta_1 + 2\theta_2\hat{m}\right)\left(\hat{p} - \hat{c}_p\right) - \left(\hat{m} - \hat{c}_m\right)\theta_3 = 0 \tag{290}$$

$$\frac{1}{2}\left[\left(\hat{m} - \hat{c}_m\right)^2 + \left(\hat{p} - \hat{c}_p\right)^2 - \hat{c}_r^2\right] = 0 \tag{291}$$

where $\hat{c}_i$ is known. This is actually less complicated than it appears, though. We can calculate $\hat{m}$ and $\hat{p}$ only knowing θ_i and $\hat{c}_i$ (which are fixed) and using

$$\left(\theta_1 + 2\theta_2\hat{m}\right)\left(\hat{p} - \hat{c}_p\right) - \left(\hat{m} - \hat{c}_m\right)\theta_3 = 0 \tag{292}$$

$$\frac{1}{2}\left[\left(\hat{m} - \hat{c}_m\right)^2 + \left(\hat{p} - \hat{c}_p\right)^2 - \hat{c}_r^2\right] = 0 \tag{293}$$

With $\hat{m}$ and $\hat{p}$ known, σ and ρ are just linear functions of c_i, and we have another closed-form expression for τ. We are then left with three equations in three unknowns: solving (285)-(287) for c_i. These unknowns, moreover, do not depend on m or p.

A.4 Attacker Manipulates Defender to Break System, Defender Knows, Attacker Knows that Defender is Aware

The attacker optimization is

$$\max \frac{1}{2}\left[(m - c_m)^2 + \left(p - c_p\right)^2 - c_r^2\right] \tag{294}$$

$$\frac{1}{2}\sum_i \Delta c_i^2 \leq \delta_{c,max} \tag{295}$$

$$\hat{c}_m = c_m + \Delta c_m \tag{296}$$

$$\hat{c}_p = c_p + \Delta c_p \tag{297}$$

$$\hat{c}_r = c_r - \Delta c_r \tag{298}$$

subject to the defender optimization

$$\min \theta_1 m + \theta_2 m^2 + \theta_3 p \tag{299}$$

$$\frac{1}{2}\left[(m - \tilde{c}_m)^2 + \left(p - \tilde{c}_p\right)^2 - \tilde{c}_r^2\right] \leq 0 \tag{300}$$

where

$$\hat{c}_m = \tilde{c}_m + \tau\left(\hat{m} - \tilde{c}_m + 2\theta_2\sigma\left(\hat{p} - \hat{c}_p\right)\right) \tag{301}$$

$$\hat{c}_p = \tilde{c}_p + \tau\left(\hat{p} - \tilde{c}_p\right) \tag{302}$$

$$\hat{c}_r = \tilde{c}_r + \tau\rho\hat{c}_r \tag{303}$$

$$\left\{ \begin{array}{c} \sigma \\ \rho \end{array} \right\} = \frac{1}{-\theta_3\left(\hat{p} - \hat{c}_p\right) - \left(\hat{m} - \hat{c}_m\right)\left(\theta_1 + 2\theta_2\hat{m}\right)} \left[\begin{array}{cc} -\left(\hat{p} - \hat{c}_p\right) & \hat{m} - \hat{c}_m \\ \theta_1 + 2\theta_2\hat{m} & \theta_3 \end{array} \right] \left\{ \begin{array}{c} \hat{m} - c_m \\ \hat{p} - c_p \end{array} \right\} \tag{304}$$

$$\tau = \left[\frac{2\delta_{c,max}}{\left(\hat{m} - c_m + 2\theta_2\sigma\left(\hat{p} - \hat{c}_p\right)\right)^2 + \left(\hat{p} - c_p\right)^2 + \rho^2\hat{c}_r^2} \right]^{\frac{1}{2}} \tag{305}$$

$$\left(\theta_1 + 2\theta_2\hat{m}\right)\left(\hat{p} - \hat{c}_p\right) - \left(\hat{m} - \hat{c}_m\right)\theta_3 = 0 \tag{306}$$

$$\frac{1}{2}\left[\left(\hat{m} - \hat{c}_m\right)^2 + \left(\hat{p} - \hat{c}_p\right)^2 - \hat{c}_r^2\right] = 0 \tag{307}$$

The quantities with tildes on them indicate that these values are what the defender *believes* to be the true values. Given that (301)-(307) not depend on m or p, the defender optimality conditions are

$$\left(\theta_1 + 2\theta_2 m\right)\left(p - \tilde{c}_p\right) - \left(m - \tilde{c}_m\right)\theta_3 = 0 \tag{308}$$

$$\frac{1}{2}\left[\left(m - \tilde{c}_m\right)^2 + \left(p - \tilde{c}_p\right)^2 - \tilde{c}_r^2\right] = 0 \tag{309}$$

The full attacker optimization is then

$$\max \frac{1}{2}\left[\left(m - c_m\right)^2 + \left(p - c_p\right)^2 - c_r^2\right] \tag{310}$$

$$\frac{1}{2}\sum_i \Delta c_i^2 \leq \delta_{c,max} \tag{311}$$

$$\hat{c}_m = c_m + \Delta c_m \tag{312}$$

$$\hat{c}_p = c_p + \Delta c_p \tag{313}$$

$$\hat{c}_r = c_r - \Delta c_r \tag{314}$$

$$\hat{c}_m = \tilde{c}_m + \tau\left(\hat{m} - \tilde{c}_m + 2\theta_2\sigma\left(\hat{p} - \hat{c}_p\right)\right) \tag{315}$$

$$\hat{c}_p = \tilde{c}_p + \tau\left(\hat{p} - \tilde{c}_p\right) \tag{316}$$

$$\hat{c}_r = \tilde{c}_r + \tau\rho\hat{c}_r \tag{317}$$

$$\left\{ \begin{array}{c} \sigma \\ \rho \end{array} \right\} = \frac{1}{-\theta_3\left(\hat{p} - \hat{c}_p\right) - \left(\hat{m} - \hat{c}_m\right)\left(\theta_1 + 2\theta_2\hat{m}\right)} \left[\begin{array}{cc} -\left(\hat{p} - \hat{c}_p\right) & \hat{m} - \hat{c}_m \\ \theta_1 + 2\theta_2\hat{m} & \theta_3 \end{array} \right] \left\{ \begin{array}{c} \hat{m} - c_m \\ \hat{p} - c_p \end{array} \right\} \tag{318}$$

$$\tau = \left[\frac{2\delta_{c,max}}{\left(\hat{m} - c_m + 2\theta_2\sigma\left(\hat{p} - \hat{c}_p\right)\right)^2 + \left(\hat{p} - c_p\right)^2 + \rho^2\hat{c}_r^2} \right]^{\frac{1}{2}} \tag{319}$$

$$\left(\theta_1 + 2\theta_2\hat{m}\right)\left(\hat{p} - \hat{c}_p\right) - \left(\hat{m} - \hat{c}_m\right)\theta_3 = 0 \tag{320}$$

$$\frac{1}{2}\left[\left(\hat{m} - \hat{c}_m\right)^2 + \left(\hat{p} - \hat{c}_p\right)^2 - \hat{c}_r^2\right] = 0 \tag{321}$$

B SINGLE-ZONE HVAC CONTROL CALCULATIONS

B.1 Baseline Problem

The baseline problem is a power minimization problem for a heater, chiller, and fan together affecting a single zone of interest:

$$\min \sum_{t=1}^{\tau} \Big[\theta_1 m^t + \theta_2 \left(m^t\right)^2 + \nu_h c_p m^t \left(T_i^t - d^t T_0^t - \left(1 - d^t\right) T_n^t\right) + c_p \nu_n m^t \left(T_{s,n}^t - T_s^t\right) + \nu_c c_p m^t \left(T_i^t - T_s^t\right) \Big] \tag{322}$$

$$-T_n^t + (1-\gamma) T_n^{t-1} + \beta m^t \left(T_{s,n}^t - T_n^t\right) + \gamma T_0^t + Q_n^t = 0 \ \left(\lambda^t\right) \tag{323}$$

$$T_n^\tau - T_n^0 = 0 \ \left(\mu_\tau\right) \tag{324}$$

$$m^t - m_l \geq 0 \ \left(\sigma_{m,l}^t\right) \tag{325}$$

$$m_u - m^t \geq 0 \ \left(\sigma_{m,u}^t\right) \tag{326}$$

$$T_{s,n}^t - T_s^t \geq 0 \ \left(\sigma_s^t\right) \tag{327}$$

$$T_n^t - T_n^l \geq 0 \ \left(\sigma_l^t\right) \tag{328}$$

$$T_n^u - T_n^t \geq \ \left(\sigma_u^t\right) \tag{329}$$

$$d^t - d_l \geq 0 \ \left(\sigma_{d,l}^t\right) \tag{330}$$

$$d_u - d^t \geq 0 \ \left(\sigma_{d,u}^t\right) \tag{331}$$

$$T_{s,n}^t - T_{s,n}^l \geq 0 \ \left(\sigma_{snl}^t\right) \tag{332}$$

$$T_{s,n}^u - T_{s,n}^t \geq 0 \ \left(\sigma_{snu}^t\right) \tag{333}$$

$$T_i^t - d^t T_0^t - \left(1 - d^t\right) T_n^t \geq 0 \ \left(\sigma_{in}^t\right) \tag{334}$$

$$T_i^t - T_s^t \geq 0 \ \left(\sigma_{is}^t\right) \tag{335}$$

where the quantities in brackets after each equation are the dual variables corresponding to those equations. Descriptions of the model variables and the model parameters are given in Tables 7 and 8, respectively. This is a single-zone version of a multi-zone HVAC model. The goal of the system is to manage the temperature in that single zone. To do this, it takes in a mixture of air from the zone and from the environment, heats that air (if necessary) at a central heating unit, cools the air (if necessary) with a chiller, and uses a fan to send the air through HVAC ducting. In a multi-zone model, there would be a local heater for each zone to provide any zone-specific heating; for our single-zone model, we retain the local heater in the interest of maintaining the same model structure.

All of the other parameters with l or u in them correspond to lower or upper bounds on their respective variables.

At each time step t, the fan consumes power $\theta_1 m^t + \theta_2 \left(m^t\right)^2$ to move air through the system, the chiller consumes power $\nu_c c_p m^t \left(T_i^t - T_s^t\right)$, and the central heating unit consumes power $\nu_h c_p m^t \left(T_i^t - d^t T_0^t - \left(1 - d^t\right) T_n^t\right)$ and the zonal heater consumes power $c_p \nu_n m^t \left(T_{s,n}^t - T_s^t\right)$. Most of the constraints are variable upper and lower bounds or physical constraints on the system (e.g., the temperature evolution of the room, the heater outputting air that is at least as warm as the air it takes in). However, there is an endpoint constraint $T_n^\tau = T_n^0$ that is essentially a design constraint: at the end of the optimization horizon, the zone needs to be at the same temperature it was at

Table 7. HVAC Control Variables

Quantity	Description
m^t	Mass flow rate
T_i^t	Temperature of air put out by central heating unit
d^t	Fraction of HVAC input air coming from environment
T_n^t	Zone temperature
$T_{s,n}^t$	Temperature of air supplied to zone
T_s^t	Output air temperature of chiller

Table 8. HVAC Model Parameters

Quantity	Value	Description
θ_1	0.1	Fan power consumption parameter
θ_2	0.1	Fan power consumption parameter
ν_h,ν_n,ν_c	0.99	Heater and chiller efficiencies
c_p	1	Specific heat of air
T_0^t	25	Environment air temperature at time t
β	0.0045	Parameter describing temperature evolution
γ	8.4e-6	Parameter describing temperature evolution
Q_n^t	0	Thermal load at time t
τ	varies	Length of optimization horizon
d_l,d_u	0.2, 0.5	Lower and upper bounds on d^t
m_l,m_u	3.93, 13.1	Lower and upper bounds on m^t
T_n^l,T_n^u	21.1, 23.9	Lower and upper bounds on T_n^t
$T_{s,n}^l$,$T_{s,n}^u$	12.7, 35	Lower and upper bounds on $T_{s,n}^t$

the beginning of the horizon. If we define the Lagrangian for this problem as L, the optimality conditions for this problem are

$$\frac{\partial L}{\partial m^t} = \theta_1 + 2\theta_2 m^t + \nu_h c_p \left(T_i^t - d^t T_0^t - \left(1-d^t\right) T_n^t\right) + c_p \nu_n \left(T_{s,n}^t - T_s^t\right) + \nu_c c_p \left(T_i^t - T_s^t\right) + \lambda^t \beta \left(T_{s,n}^t - T_n^t\right) + \sigma_{m,u}^t - \sigma_{m,l}^t = 0 \tag{336}$$

$$\frac{\partial L}{\partial d^t} = \nu_h c_p m^t \left(T_n^t - T_0^t\right) + \sigma_{d,u}^t - \sigma_{d,l}^t - \sigma_{in}^t \left(T_n^t - T_0^t\right) = 0 \tag{337}$$

$$\frac{\partial L}{\partial T_n^t} = \nu_h c_p m^t \left(d^t - 1\right) + \lambda^t \left(-1 - \beta m^t\right) - \delta_{t\tau}\mu_\tau + (1-\gamma)\lambda^{t+1} - \sigma_{in}^t \left(d^t - 1\right) - \sigma_l^t + \sigma_u^t = 0 \tag{338}$$

$$\frac{\partial L}{\partial T_{s,n}^t} = c_p \nu_n m^t + \lambda^t \beta m^t - \sigma_s^t - \sigma_{snl}^t + \sigma_{snu}^t = 0 \tag{339}$$

$$\frac{\partial L}{\partial T_s^t} = -c_p \nu_n m^t - \nu_c c_p m^t + \sigma_s^t + \sigma_{is}^t = 0 \tag{340}$$

$$\frac{\partial L}{\partial T_i^t} = \nu_h c_p m^t + \nu_c c_p m^t - \sigma_{in}^t - \sigma_{is}^t = 0 \tag{341}$$

plus the optimization problem constraints listed above; note that $\delta_{t\tau}$, is a Kronecker delta, so it is 1 if $t = \tau$ and 0 otherwise. These derivative conditions can simplify down to

$$0 \leq \theta_1 + 2\theta_2 m^t + \nu_h c_p \left(T_i^t - d^t T_0^t - \left(1 - d^t\right) T_n^t\right) + c_p \nu_n \left(T_{s,n}^t - T_s^t\right) + \nu_c c_p \left(T_i^t - T_s^t\right)$$
$$+\lambda^t \beta \left(T_{s,n}^t - T_n^t\right) + \sigma_{m,u}^t \perp m^t - m_l \geq 0 \tag{342}$$
$$0 \leq \sigma_{m,u}^t \perp m_u - m^t \geq 0 \tag{343}$$
$$0 \leq d^t - d_l \perp \left(\sigma_{is} - \nu_c c_p m^t\right)\left(T_n^t - T_0^t\right) + \sigma_{d,u}^t \geq 0 \tag{344}$$
$$0 \leq \sigma_{d,u}^t \perp d_u - d^t \geq 0 \tag{345}$$
$$0 \leq \left(\sigma_{is} - \nu_c c_p m^t\right)\left(d^t - 1\right) - \lambda^t \left(1 + \beta m^t\right) - \delta_{t\tau}\mu_\tau + \left(1 - \gamma\right)\lambda^{t+1} + \sigma_u^t \perp T_n^t - T_n^l \geq 0 \tag{346}$$
$$0 \leq \sigma_u^t \perp T_n^u - T_n^t \geq 0 \tag{347}$$
$$0 \leq \lambda^t \beta m^t + \sigma_{is} - \nu_c c_p m^t + \sigma_{snu}^t \perp T_{s,n}^t - T_{s,n}^l \geq 0 \tag{348}$$
$$0 \leq \sigma_{snu}^t \perp T_{s,n}^u - T_{s,n}^t \geq 0 \tag{349}$$
$$0 \leq \nu_h c_p m^t - \left(\sigma_{is} - \nu_c c_p m^t\right) \perp T_i^t - d^t T_0^t - \left(1 - d^t\right) T_n^t \geq 0 \tag{350}$$
$$0 \leq \nu_n c_p m^t - \left(\sigma_{is} - \nu_c c_p m^t\right) \perp T_{s,n}^t - T_s^t \geq 0 \tag{351}$$
$$0 \leq \sigma_{is}^t \perp T_i^t - T_s^t \geq 0 \tag{352}$$

where $x \perp y$ indicate the complementarity constraint $xy = 0$. In general, this problem is nonconvex. However, the parameter values specified above result in $m^t = m_l$ and $d^t = d_l$ for all t. If we take these variables as constants, then the objective function and constraints are all linear in the model variables, so the optimization is a linear program, and the optimality conditions are then necessary and sufficient. More generally, as long as the fan consumes most of the power (as it does in this case), it will be advantageous to keep m^t as small as possible, and as long as the environment temperature differs from the zone temperature, the controller will always be incentivized to minimize the amount of outside air brought in (air that will have to be heated or cooled to reach the zone temperature).

B.2 Attacker Manipulates Defender Perceptions of Static Parameters

The attacker can manipulate the defender's perception of β and γ to maximize power consumption over the entire time horizon:

$$\max \sum_{t=1}^{\tau} \Big[\theta_1 m^t + 2\theta_2 \left(m^t\right)^2 + \nu_h c_p m^t \left(T_i^t - d^t T_0^t - \left(1 - d^t\right) T_n^t\right)$$
$$+c_p \nu_n m^t \left(T_{s,n}^t - T_s^t\right) + \nu_c c_p m^t \left(T_i^t - T_s^t\right)\Big] \tag{353}$$
$$T_n^t = \left(1 - \gamma\right) T_n^{t-1} + \beta m^t \left(T_{s,n}^t - T_n^t\right) + \gamma T_0^t + Q_n^t \tag{354}$$
$$\hat{\beta} = \beta + \Delta\beta \tag{355}$$
$$\hat{\gamma} = \gamma + \Delta\gamma \tag{356}$$

$$\frac{1}{2}\left[\left(\frac{\Delta\beta}{\beta}\right)^2 + \left(\frac{\Delta\gamma}{\gamma}\right)^2\right] - \delta_{max} \leq 0 \tag{357}$$
$$0 \leq T_i^t - T_s^t \perp T_i^t - d^t T_0^t - \left(1 - d^t\right) T_n^t \geq 0 \tag{358}$$

subject to the defender optimality conditions

$$0 \leq \theta_1 + 2\theta_2 m^t + v_h c_p \left(\hat{T}_i^t - d^t T_0^t - \left(1 - d^t\right) \hat{T}_n^t\right) + c_p v_n \left(T_{s,n}^t - T_s^t\right) + v_c c_p \left(\hat{T}_i^t - T_s^t\right)$$
$$+\lambda^t \hat{\beta} \left(T_{s,n}^t - \hat{T}_n^t\right) + \sigma_{m,u}^t \perp m^t - m_l \geq 0 \tag{359}$$
$$0 \leq \sigma_{m,u}^t \perp m_u - m^t \geq 0 \tag{360}$$
$$0 \leq d^t - d_l \perp \left(\sigma_{is} - v_c c_p m^t\right) \left(\hat{T}_n^t - T_0^t\right) + \sigma_{d,u}^t \geq 0 \tag{361}$$
$$0 \leq \sigma_{d,u}^t \perp d_u - d^t \geq 0 \tag{362}$$
$$0 \leq \left(\sigma_{is} - v_c c_p m^t\right) \left(d^t - 1\right) - \lambda^t \left(1 + \hat{\beta} m^t\right) - \delta_{t\tau} \mu_\tau + \left(1 - \hat{\gamma}\right) \lambda^{t+1} + \sigma_u^t \perp \hat{T}_n^t - T_n^l \geq 0 \tag{363}$$
$$0 \leq \sigma_u^t \perp T_n^u - \hat{T}_n^t \geq 0 \tag{364}$$
$$0 \leq \lambda^t \hat{\beta} m^t + \sigma_{is} - v_c c_p m^t + \sigma_{snu}^t \perp T_{s,n}^t - T_{s,n}^l \geq 0 \tag{365}$$
$$0 \leq \sigma_{snu}^t \perp T_{s,n}^u - T_{s,n}^t \geq 0 \tag{366}$$
$$0 \leq v_h c_p m^t - \left(\sigma_{is} - v_c c_p m^t\right) \perp \hat{T}_i^t - d^t T_0^t - \left(1 - d^t\right) \hat{T}_n^t \geq 0 \tag{367}$$
$$0 \leq v_n c_p m^t - \left(\sigma_{is} - v_c c_p m^t\right) \perp T_{s,n}^t - T_s^t \geq 0 \tag{368}$$
$$0 \leq \sigma_{is}^t \perp \hat{T}_i^t - T_s^t \geq 0 \tag{369}$$
$$-\hat{T}_n^t + \left(1 - \hat{\gamma}\right) \hat{T}_n^{t-1} + \hat{\beta} m^t \left(T_{s,n}^t - \hat{T}_n^t\right) + \hat{\gamma} T_0^t + Q_n^t = 0 \tag{370}$$
$$\hat{T}_n^T - T_n^0 = 0 \tag{371}$$

Note that the defender conditions are with respect to perceived/perturbed values, not real values (hence the ˆ on certain quantities). The defender directly controls most of the variables (e.g., m^t, T_s^t) but does not directly control T_i^t or T_n^t. These variables are essentially functions of processes governed by other variables. As such, $\hat{T}_i^t$ and $\hat{T}_n^t$ are the defender's perceived values for these variables. The true equations governing the evolution of T_n^t and T_i^t are, respectively,

$$T_n^t = \left(1 - \gamma\right) T_n^{t-1} + \beta m^t \left(T_{s,n}^t - T_n^t\right) + \gamma T_0^t + Q_n^t \tag{372}$$
$$0 \leq T_i^t - T_s^t \perp T_i^t - d^t T_0^t - \left(1 - d^t\right) T_n^t \tag{373}$$

The complementarity constraint ensures that T_i^t is the minimum of T_s^t and $d^t T_0^t + \left(1 - d^t\right) T_n^t$. If $T_i^t > T_s^t$, the defender spends energy to cool the air and if $T_i^t > d^t T_0^t + \left(1 - d^t\right) T_n^t$, the defender spends energy to heat the air.

B.3 Attacker Manipulates Defender Perceptions of Time-Varying Parameters

The attacker can also manipulate the defender's perception of T_0^t to maximize power consumption over the entire time horizon:

$$\max_{\Delta T_0^t} \sum_t \Big[\theta_1 m^t + \theta_2 \left(m^t\right)^2 + \nu_h c_p m^t \left(T_i^t - d^t T_0^t - \left(1-d^t\right) T_n^t\right) + c_p \nu_n m^t \left(T_{s,n}^t - T_s^t\right) + \nu_c c_p m^t \left(T_i^t - T_s^t\right) \Big] \tag{374}$$

$$\frac{1}{2} \sum_t \left(\Delta T_0^t\right)^2 \leq \Delta T_{max} \tag{375}$$

$$\hat{T}_0^t = T_0^t + \Delta T_0^t \tag{376}$$

$$-T_n^t + (1-\gamma) T_n^{t-1} + \beta m^t \left(T_{s,n}^t - T_n^t\right) + \gamma T_0^t + Q_n^t = 0 \tag{377}$$

$$0 \leq T_i^t - T_s^t \perp T_i^t - d^t T_0^t - \left(1-d^t\right) T_n^t \geq 0 \tag{378}$$

subject to the defender optimality conditions

$$\hat{T}_n^T - T_n^0 = 0 \tag{379}$$

$$0 \leq \theta_1 + 2\theta_2 m^t + \nu_h c_p \left(\hat{T}_i^t - d^t \hat{T}_0^t - \left(1-d^t\right) \hat{T}_n^t\right) + c_p \nu_n \left(T_{s,n}^t - T_s^t\right) + \nu_c c_p \left(\hat{T}_i^t - T_s^t\right) + \lambda^t \beta \left(T_{s,n}^t - \hat{T}_n^t\right) + \sigma_{m,u}^t \perp m^t - m_l \geq 0 \tag{380}$$

$$0 \leq \sigma_{m,u}^t \perp m_u - m^t \geq 0 \tag{381}$$

$$0 \leq d^t - d_l \perp \left(\sigma_{is} - \nu_c c_p m^t\right) \left(\hat{T}_n^t - \hat{T}_0^t\right) + \sigma_{d,u}^t \geq 0 \tag{382}$$

$$0 \leq \sigma_{d,u}^t \perp d_u - d^t \geq 0 \tag{383}$$

$$0 \leq \left(\sigma_{is} - \nu_c c_p m^t\right) \left(d^t - 1\right) - \lambda^t \left(1 + \beta m^t\right) - \delta_{t\tau} \mu_\tau + (1-\gamma) \lambda^{t+1} + \sigma_u^t \perp \hat{T}_n^t - T_n^l \geq 0 \tag{384}$$

$$0 \leq \sigma_u^t \perp T_n^u - \hat{T}_n^t \geq 0 \tag{385}$$

$$0 \leq \lambda^t \beta m^t + \sigma_{is} - \nu_c c_p m^t + \sigma_{snu}^t \perp T_{s,n}^t - T_{s,n}^l \geq 0 \tag{386}$$

$$0 \leq \sigma_{snu}^t \perp T_{s,n}^u - T_{s,n}^t \geq 0 \tag{387}$$

$$0 \leq \nu_h c_p m^t - \left(\sigma_{is} - \nu_c c_p m^t\right) \perp \hat{T}_i^t - d^t \hat{T}_0^t - \left(1-d^t\right) \hat{T}_n^t \geq 0 \tag{388}$$

$$0 \leq \nu_n c_p m^t - \left(\sigma_{is} - \nu_c c_p m^t\right) \perp T_{s,n}^t - T_s^t \geq 0 \tag{389}$$

$$0 \leq \sigma_{is}^t \perp \hat{T}_i^t - T_s^t \geq 0 \tag{390}$$

$$-\hat{T}_n^t + (1-\gamma) \hat{T}_n^{t-1} + \beta m^t \left(T_{s,n}^t - \hat{T}_n^t\right) + \gamma \hat{T}_0^t + Q_n^t = 0 \tag{391}$$

$$\hat{T}_n^T - T_n^0 = 0 \tag{392}$$