Fall 2014

Rational Multiparty Computation

John Ross Wallrabenstein
Purdue University

Graduate School Form
(Revised /)

PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By John Ross Wallrabenstein

Entitled
RATIONAL MULTIPARTY COMPUTATION

For the degree of Doctor of Philosophy

Is approved by the final examining committee:

Christopher Clifton

Mikhail Atallah

Ninghui Li

Mathias Payer

To the best of my knowledge and as understood by the student in the Thesis/Dissertation Agreement,
Publication Delay, and Certification/Disclaimer (Graduate School Form 32), this thesis/dissertation
adheres to the provisions of Purdue University's "Policy on Integrity in Research" and the use of
copyrighted material.

Approved by Major Professor(s): Christopher Clifton

Approved by: Sunil Prabhakar 11/14/2014

Head of the Graduate Program Date

RATIONAL MULTIPARTY COMPUTATION

A Dissertation

Submitted to the Faculty

of

Purdue University

by

John Ross Wallrabenstein

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

December 2014

Purdue University

West Lafayette, Indiana

per aspera ad astra

ACKNOWLEDGMENTS

To my major professor, Prof. Chris Clifton, I am forever grateful for your mentorship on teaching, research, and traveling the world. This has been a long journey, and I could not have asked for a better guide.

To Mrs. Patricia Clifton, thank you for encouraging me to share my music and for welcoming me into your home as family.

To my committee members, thank you for the valuable time you have invested, and for your recommendations that have improved this work.

To Prof. Mikhail Atallah I owe a special thanks. Your kind words and encouragement were never forgotten.

To Dr. Keith Frikken, thank you for sparking and fostering my interest in cryptography. You changed the course of my life.

To Prof. Yevgeniy Vorobeychik, thank you for serving as my game theory mentor. You treated me as an equal, and were patient while I learned.

To Mr. Akihiro Nishida-san and Dr. David Stork of RICOH, thank you for supporting portions of this research and for introducing me to Japan.

To my colleagues at Sypris Electronics, thank you for supporting portions of this research, and for encouraging my continued education.

To my closest friend Andrew Haddad, thank you for your friendship and our discussions over a glass of single malt Scotch. Slàinte.

To Sarah Engler, thank you for loving me.

To my family, I am grateful for your enduring support and encouragement. I am fortunate to have parents that instilled in me a passion for science and mathematics. This journey would not have been possible without you.

TABLE OF CONTENTS

Page

Page

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Wallrabenstein, John Ross Ph.D., Purdue University, December 2014. Rational Multiparty Computation. Major Professor: Chris Clifton.

The field of *rational cryptography* considers the design of cryptographic protocols in the presence of rational agents seeking to maximize local utility functions. This departs from the standard secure multiparty computation setting, where players are assumed to be either honest or malicious.

We detail the construction of both a two-party and a multiparty game theoretic framework for constructing rational cryptographic protocols. Our framework specifies the utility function assumptions necessary to realize the privacy, correctness, and fairness guarantees for protocols. We demonstrate that our framework correctly models cryptographic protocols, such as rational secret sharing, where existing work considers equilibrium concepts that yield unreasonable equilibria.

Similarly, we demonstrate that cryptography may be applied to the game theoretic domain, constructing an auction market not realizable in the original formulation. Additionally, we demonstrate that modeling players as rational agents allows us to design a protocol that destabilizes coalitions. Thus, we establish a mutual benefit from combining the two fields, while demonstrating the applicability of our framework to real-world market environments.

We also give an application of game theory to adversarial interactions where cryptography is not necessary. Specifically, we consider adversarial machine learning, where the adversary is rational and reacts to the presence of a data miner. We give a general extension to classification algorithms that returns greater expected utility for the data miner than existing classification methods.

1 INTRODUCTION

Rational cryptography is the study of designing cryptographic primitives and protocols in the presence of *rational* players. By modeling all players as rational agents acting to maximize a local utility function, the security model more accurately captures how agents behave in real world settings. This departs from the standard model, where players are modeled as either *semi-honest* or arbitrarily *malicious*. The field of rational cryptography integrates results from the game theoretic literature into the security analysis of cryptographic primitives and protocols.

The standard model consists of two polar frameworks, each considering how to model the subset of players that act in an adversarial manner. The first considers semi-honest behavior, where adversaries are bound to follow the protocol specification, yet may attempt to learn additional information from the protocol transcript. The second considers malicious behavior, where adversaries may deviate arbitrarily from the protocol specification. The semi-honest model relies on the strong assumption that adversarial players will always follow the protocol specification, as this yields more efficient protocol constructions. The malicious model makes no assumptions about adversarial behavior beyond what is possible in an adversary's complexity class, but does so through increased computational cost to the protocol.

Rational cryptography provides an intermediary framework between the two poles of the standard model. First, the assumption that players will follow the protocol specification is removed. Second, the assumption that players select actions to maximize a utility function is added. Removing the first assumption admits more powerful adversaries than the semi-honest model, while the addition of the second assumption bounds adversaries to be strictly weaker than those considered in the malicious model. Acting together, the two assumptions are designed to admit realistic adversarial actions (those that benefit a player) while eliminating those that are unrealistic (those that do not).

A final departure from the standard model is in the adversarial classification of players. Both the semi-honest and malicious models divide players into an *honest* and *adversarial* class. Protocol robustness is defined with respect to the proportion of adversarial to honest players the protocol will tolerate. Rational cryptography considers only a single class of players: rational agents. That is, all players may act in an adversarial manner if doing so is a utility-maximizing strategy. This modeling choice facilitates security proofs for protocols when all players may act adversarially; a statement even the malicious model is unable to capture.

1.1 Contributions

We first present a game theoretic solution to the problem of adversarial machine learning, which demonstrates the utility of a game theoretic approach in an adversarial setting. We then demonstrate the utility of applying cryptographic primitives to a classic game theoretic problem, establishing the synergy between the two fields. We then merge cryptography and game theory, presenting two frameworks for reasoning about the security and construction of cryptographic protocols where all players are *rational*, rather than semi-honest or malicious. We first introduce the two-party framework, and demonstrate the necessary and sufficient conditions for protocols providing privacy, fairness and correctness in the presence of rational participants. Finally, we extend the two-party framework to the multiparty setting where players have access to point-to-point communication channels. We argue that our multiparty framework provides a more realistic model for collusion than prior work, and demonstrate that a broad class of non-trivial games are realizable under our model.

1.1.1 Rationality Applied to Non-Cryptographic Domains

We demonstrate that the notion of rational adversaries may be effectively applied to domains where cryptography is not necessary. Specifically, we consider the area of adversarial machine learning, and model adversaries as rational agents. We demonstrate that adjusting classifiers to the expected behavior of a rational adversary yields more accurate classification rules that are robust against future adversaries. We use this as evidence of the power a rational adversary model may bring to cryptographic constructions.

1.1.2 Applying Cryptography to Game Theory

To illustrate how cryptography can be applied to problems in game theory, we consider the Walrasian Auction Market [1]. In this setting, we consider a set of sellers and buyers that wish to determine the equilibrium price for a good *without* executing any trades. Typically, excess demand is revealed through trade, which determines the equilibrium price of a good. However, the Walrasian Auction theory holds only if no trade occurs prior to reaching equilibrium. We construct a secure protocol that takes as arguments the buyers' utility functions and the sellers' initial quantities and prices, and outputs the equilibrium price for all goods. No information about the buyers' utility functions or the sellers' supplies and initial prices are revealed, and the only information learned is the equilibrium prices and what can be deduced from knowledge of the function and its output.

1.1.3 Rational Two-Party Computation Framework

We combine cryptography with game theory to present a rational two-party computation framework. To reason about the security and construction of cryptographic protocols from a game-theoretic perspective, we build a framework to support the standard properties desirable in a cryptographic protocol: privacy, correctness, and fairness. This work illustrates how game theory can be applied to cryptography, where all players are considered *rational*, rather than semi-honest or malicious. We use this rational 2-party computation framework to construct protocols that satisfy the privacy, correctness, and fairness properties under the assumption of rational players.

1.1.4 Rational Multiparty Computation Framework

We extend the two-party framework to reason about the security of cryptographic protocols in the presence of multiple rational agents. Our framework makes no assumptions about the communication interfaces available to agents, which departs from the strong restrictions imposed by prior work. As our framework only permits those game specifications allowing point-to-point communication, our results are not general. However, we demonstrate that a non-trivial class of games (including those that restrict communication) have equivalent formulations that are admissible under our framework, and have realizable protocol constructions. We argue that our multiparty framework provides a more realistic model than prior work, as real world players typically have access to point-to-point communication channels.

1.2 Thesis Statement

In this work, we argue that there exist equilibrium concepts from the game theoretic literature that accurately capture how rational multiparty computation participants engage in cryptographic protocols. We propose a novel two-party framework for rational cryptography, proving necessary and sufficient conditions for protocols providing privacy, correctness, and fairness in the presence of rational participants. We extend our two-party framework to the multiparty setting, where players have access to point-to-point communication channels. We demonstrate that our framework captures a large class of non-trivial games, which may be realized into equivalent protocol constructions. We then demonstrate how game theoretic and cryptographic concepts may be applied to build solutions for existing problems in both fields.

1.3 Organization of Thesis

1.3.1 Cryptography Background

In Chapter 2, we describe the necessary cryptographic concepts for understanding the material presented in the Thesis. We describe homomorphic encryption, which is used to construct a privacy preserving protocol for the Walrasian Auction Market. We review standard concepts from secure multiparty computation, which are necessary to understand our rational two-party and multiparty frameworks.

1.3.2 Game Theory Background

In Chapter 3, we review the necessary concepts from the game theoretic literature. These are necessary for the proofs of incentive compatibility in our construction of the Walrasian Auction Market, as well as to understand the proofs of security for both the two-party and multiparty frameworks.

1.3.3 Rationality Applied to Non-Cryptographic Domains

In Chapter 4, we present a general method for computing optimal operational decisions in adversarial environments. We approach the problem from a machine learning perspective, where a defender deploys a machine learning algorithm and an adversary responds to the presence of the classifier. We consider the setting of spam detection, where adversaries continuously adapt to the presence of classifiers. The defender may only inspect a proper subset of all records generated, as manual inspection is costly in email and network intrusion detection settings. We show that a game theoretic approach, under the assumption of rational adversaries, yields a general solution for increasing classification accuracy and utility when inspection is bounded.

1.3.4 Applying Cryptography to Game Theory

In Chapter 5, we present a cryptographic construction of the Walrasian Auction Market. The protocol preserves the privacy of all participants, and is incentive compatible against coalitions of individuals *not* controlled by a third party. Here we demonstrate that cryptography may be successfully applied to game theoretic problems.

1.3.5 Applying Game Theory to Cryptography

In Chapter 6, we present a framework for constructing cryptographic protocols that provide privacy, correctness and fairness in the presence of rational adversaries. We demonstrate that game theoretic concepts may be applied successfully to the cryptography domain, allowing more efficient constructions of protocols to be built.

1.3.6 Realizing Rational Multiparty Protocols

In Chapter 7, we present the multiparty framework for constructing cryptographic protocols in the presence of more than two rational agents. We examine the issue of player collusion, which is critically affected by the communication resources available to players. We demonstrate that our framework is able to properly model collusion in real world protocols, and describe how ideal game specifications are translated into realizable protocols.

1.4 Summary

This work considers the intersection of cryptographic protocol design and game theoretic principles. We first present an application of game theory to adversarial domains to demonstrate the utility of the approach. Second, we present an application of cryptography to a game theoretic problem to demonstrate synergy between the two fields. We then present a two-party framework for rational cryptography, combining game theoretic principles into the framework for cryptographic protocol design. Finally, we build on the two-party framework and present a rational multi-party framework, considering an arbitrary number of rational players in standard communication models.

2 CRYPTOGRAPHY BACKGROUND

In this chapter, we review the necessary cryptographic concepts for understanding the two-party and multiparty frameworks presented in Chapters 6 and 7, as well as the cryptographic protocol construction of the Walrasian Auction Market presented in Chapter 5.

2.1 Secure Multiparty Computation

A Secure Multiparty Computation (SMPC) Protocol π is an interaction between $n \geq 2$ mutually distrustful parties $p_i \in P$. Each party p_i has a private input $x_i \in \vec{x}$, where $\vec{x}$ is drawn from some distribution $\mathcal{D}(\vec{x})$. The goal of π is to compute some functionality $f(\vec{x}) \mapsto \vec{y}$, which may be probabilistic. For randomized functionalities, a common random string $r \in \{0,1\}^*$, unknown to any party, is included as an argument and we write $f(\vec{x}, r) \mapsto \vec{y}$. Each party p_i receives output y_i at the conclusion of the protocol. The transcript τ of π contains all messages that were sent during the protocol execution. The general problem was introduced by Yao [2], where two millionaires wish to learn which one is wealthier. In a later work, Yao demonstrated that any function computable in polynomial time can be computed *securely* in polynomial time [3]. The two-party case was generalized to the multiparty case by Goldreich et al. [4].

Informally speaking, the desirable properties of a SMPC protocol include:

1. **Correctness**: Protocol π outputs $f(\vec{x})$ for all $\vec{x}$ in the domain of f.

2. **Privacy**: The transcript τ of π, and the output $\vec{y}$ reveal no additional information beyond what can be deduced from the output $\vec{y}$ and knowledge of f.

3. **Fairness**: $f(\vec{x}) = y_i$ is output by all parties $p_{i,0\leq i\leq n}$ computing π, or the probability of computing the correct output is at most negligibly different for the malicious and honest parties.

2.1.1 Adversarial Players

The standard model consists of two polar frameworks for modeling adversarial players, each of which classifies only a proper subset of the players as adversarial [5]. The two frameworks are referred to as the *semi-honest*, or "honest-but-curious" model, and the *malicious* model. In both models, the adversary is *monolithic*: a single entity that corrupts and controls a subset of the players in the protocol. The difference between the two models is the degree of power assumed of the adversary.

Semi-Honest Adversaries

In the semi-honest model, adversarial players are bound to follow the protocol specification. However, a semi-honest adversary may attempt to learn additional information by performing additional computation over the protocol transcript τ in order to learn more information. Security against semi-honest adversaries is the weakest form of security considered in the standard model, however it yields the most efficient protocol constructions.

Malicious Adversaries

Stronger adversaries, referred to as malicious adversaries, are allowed to deviate arbitrarily from the protocol specification. Thus, protocol constructions secure in this model require that players demonstrate each step of the protocol was performed correctly. This requirement imposes a considerable burden on the protocol construction, as demonstrating correctness is usually achieved through computationally expensive zero knowledge proofs [6]. However, security under the malicious model is the strongest form of security considered in the standard model.

While there exists a deterministic compiler for converting protocols secure in the semi-honest model to one secure in the malicious model [5], the resulting protocol incurs a substantial computational overhead. As this limits the likelihood that the protocol will be deployed in real world settings, the majority of protocols are constructed under the weaker but more efficient semi-honest model.

Rational Adversaries

Rational cryptography proposes an adversary model more powerful than the semi-honest framework, yet less powerful in general[1] than the malicious framework. The core of the argument towards an intermediary model is centered around the following two questions:

- Will adversarial players follow the protocol specification, even when doing so is not in their best interest?
- Must adversarial players be prevented from choosing actions that are not in their best interest?

Throughout the remainder of this thesis, we argue that the answer to both questions should be in the negative. By removing the assumption that adversarial players will follow the protocol specification, we consider more powerful adversaries than the semi-honest case. However, we consider weaker adversaries than those in the malicious model, as we do not protect against actions that are not in the best interest of the adversary (as defined by assumptions about the adversary's goals, which is expressed through a utility function [Chapter 3]).

[1] As we will demonstrate in this thesis, modeling players as rational agents yields protocol properties that are unachievable under even the malicious framework. For example, guaranteeing that rational agents will choose to provide their true input to the protocol [Chapter 5], or will continue participating in the protocol rather than aborting [Chapter 6].

A final departure of rational cryptography from the standard model concerns how players are classified. In both the semi-honest and malicious frameworks from the standard model, players are classified as either honest or adversarial. Neither model provides security guarantees in the case where all players are classified as adversarial. However, rational cryptography assumes that *all* players are rational agents, and will deviate from the protocol specification if doing so is in their best interest. In this sense, rational cryptography considers a stronger class of adversaries than either of the traditional frameworks, as all players may act adversarially.

2.1.2 The Simulation Paradigm

The security of a SMPC protocol is demonstrated by showing an equivalence between an *ideal* and a *real* model of execution under explicit assumptions. In the ideal model, each party p_i sends x_i to an incorruptible *trusted third party* (TTP) T. The output of $f(\vec{x})$ is computed by T, who then distributes y_i to p_i. The standard assumption limits the running time of all parties and adversaries to *probabilistic polynomial time* (PPT), and security is demonstrated by showing that the ideal and real distributions are *computationally indistinguishable* [5].

Definition 2.1.1 *Let X and Y be probability ensembles. We say that X and Y are* ***computationally indistinguishable****, written $X \overset{c}{\equiv} Y$, if for every non-uniform probabilistic polynomial time distinguisher $\mathcal{D}$ there exists a negligible function $\epsilon(\cdot)$ such that for all sufficiently large $\lambda \in \{0,1\}^*$:*

$$|Pr[\mathcal{D}(X) = 1] - Pr[\mathcal{D}(Y) = 1]| \leq \epsilon(\lambda) \tag{2.1}$$

Here, λ is the *security parameter* of the protocol, given in unary as 1^λ.

By demonstrating that the real view of the protocol is computationally indistinguishable from the ideal view, we guarantee that with only negligible probability PPT adversaries gain more information than the protocol specifies.

2.1.3 Secure Function Evaluation

We review Yao's constant round secure function evaluation (SFE) method for evaluating any polynomial-time function securely in the semi-honest model [3]. The goal is to take a deterministic functionality $f(x, y)$ and build a garbled circuit $\mathcal{C}$ that implements $f(x, y)$. For a complete treatment, see Lindell et al.'s proof of security [7]. The protocol occurs between a circuit *generator* and a circuit *evaluator*. The generator constructs $\mathcal{C}$ implementing $f(x, y)$, and the evaluator computes $\mathcal{C}$ by obtaining their input values through a series of 1-out-of-2 oblivious transfer[2].

To construct the garbled circuit, we will choose wire encodings k_w^σ drawn from some distribution $\mathcal{D}$, where k_w^σ is the encoding of $\sigma \in \{0, 1\}$ for wire w. As all wire encodings k_w^σ are drawn from the same distribution $\mathcal{D}$, knowledge of k_w^σ does not reveal whether $\sigma = 0$ or $\sigma = 1$. The mapping $\sigma \mapsto \{0, 1\}$ is known only to the circuit generator. The wire encodings k_w^σ are used as keys to a symmetric encryption algorithm, which decrypt to either the proper encoding for the next gate in the series, or the final output value.

We give a construction of a garbled NAND gate, as any function can be computed solely through a chain of NAND gates:

Table 2.1: Garbled NAND Gate

Input w_1	Input w_2	Output w_3	Garbled Computation Table
k_1^0	k_2^0	1	$E_{k_1^0}(E_{k_2^0}(1))$
k_1^0	k_2^1	1	$E_{k_1^0}(E_{k_2^1}(1))$
k_1^1	k_2^0	1	$E_{k_1^1}(E_{k_2^0}(1))$
k_1^1	k_2^1	0	$E_{k_1^1}(E_{k_2^1}(0))$

When gates are chained together, the output wire w_i contains k_i^σ, which is the input encoding for the next gate. As the evaluator does not know the proper encoding for its value $\sigma \in \{0, 1\}$ for wire w_i, it must ask the generator for k_i^σ. Clearly this will reveal the evaluator's input to the generator, which defeats the purpose of the

[2] See Rabin's original protocol for realizing a 1-out-of-2 oblivious transfer [8].

construction. Thus, a 1-out-of-2 oblivious transfer is used, where the evaluator learns exactly one of $\{k_i^0, k_i^1\}$ and the generator does not learn which encoding was requested.

The general SFE construction with n-ary gates yielding m outputs has complexity $2^n m$. Although fast constructions exist for the two-party case [9], the existing multiparty implementations (e.g., FairPlayMP [10]) are not practical for large n.

2.2 Homomorphic Encryption

In general, a cryptosystem supports the encryption $E_k(\cdot)$ and decryption $D_k(\cdot)$ operations such that $D_k(E_k(x)) = x$. A *homomorphic* encryption system has an additional property:

$$E_k(x) \cdot E_k(y) = E_k(x \odot y) \tag{2.2}$$

where $\odot$ is a binary operator, such as addition or multiplication. By the definition of multiplication, we can observe that the following property also holds:

$$E_k(x)^c = E_k(x \cdot c) \tag{2.3}$$

In this thesis, we consider Paillier's cryptosystem [11], where the homomorphic operation $\odot$ is additive.

3 GAME THEORY BACKGROUND

In this chapter, we review the necessary game theoretic concepts required for the proofs of security for the two-party and multiparty frameworks presented in Chapters 6 and 7, and the incentive compatibility argument for the Walrasian Auction Market presented in Chapter 5. Katz gives an excellent summary of basic game theory notions and their applications to cryptography [12].

3.1 Equilibrium Concepts

We now review the equilibrium concepts and game settings considered by existing work. Our goal is to demonstrate the shortcomings of equilibrium concepts considered by existing work, and to motivate our choice of perfect Bayesian equilibrium as the solution concept for our framework.

3.1.1 Normal Form Games

We begin by introducing *normal form*, or strategic, games. Normal form representation of games is ideal for modeling simultaneous interaction, rather than sequential moves. We review the formal definition from Osborne [13]:

Definition 3.1.1 *A **normal form game** Γ consists of:*

1. *A finite set N of players.*
2. *A nonempty set A_i of actions available for each player $i \in N$.*
3. *A preference relation $\precsim_i$ on $A = \times_{j \in N} A_j$ for each player i.*

Frequently, the preference relation $\precsim_i$ is represented by a *utility function* $\mu_i : A \rightarrow \mathbb{R}$, such that $\mu_i(a) \geq \mu_i(b)$ when $b \precsim_i a$. The normal form game is then denoted by $\Gamma = \langle N, (A_i), (\mu_i) \rangle$.

Normal form games are well-suited to modeling one-shot protocols where players move simultaneously[1]. In a computational setting, this is equivalent to assuming the existence of a broadcast channel. However, it is desirable to remove the assumption of simultaneous moves (and, thus, the assumption of a broadcast channel) so that players may move sequentially. We will return to this goal when we consider extensive-form dynamic games.

Nash Equilibrium

We first review the standard solution concept in game theory, the Nash equilibrium [14]. The definition does not account for players in a computational setting. Frequently, a deterministic choice of an action $a_i \in A_i$ will not yield a Nash equilibrium. Thus, we allow players to choose a *strategy* σ_i, a probability distribution over A_i.

Definition 3.1.2 *A* ***Nash equilibrium*** *of a normal form game* $\Gamma = \langle N, (A_i), (\mu_i) \rangle$ *is a strategy profile* $\vec{\sigma}$ *such that for every player* $i \in N$*:*

$$\mu_i(\sigma_i) \geq \mu_i(\sigma_i', \sigma_{-i}) \forall \sigma_i' \tag{3.1}$$

where $\sigma_{-i} \overset{\text{def}}{=} (\sigma_j)_{j \in N \setminus \{i\}}$

Intuitively, no player i has an incentive to deviate from strategy σ_i given that every other player j selects their equilibrium strategy σ_j. We now consider Nash equilibria in the computational setting.

[1] Technically, the notion of simultaneity only requires that players *commit* to their strategies before moving. However, this is still an assumption we would like to remove.

Computational Nash Equilibrium

The computational Nash equilibrium is the most widely used solution concept for rational cryptography [15–19]. The intuition is to account for strategies that, although optimal, occur with only negligible probability. In a cryptographic setting, an optimal strategy may be to break the underlying cryptosystem. However, for players bound to PPT, this strategy succeeds with only negligible probability. Consequently, Nash equilibrium has been refined into a *computational* variant that states players only switch strategies if the gain is not negligible with respect to the security parameter λ. The original definition of a computational Nash equilibrium was given by Dodis et al. [16]:

Definition 3.1.3 *A* ***computational Nash equilibrium*** *of a two-party extensive-form game* Γ *is an independent strategy profile* (σ_1^*, σ_2^*)*, such that*

1. *both* σ_1^*, σ_2^* *are PPT computable.*
2. *for any other PPT computable strategies* σ_1', σ_2'*, we have*

$$\mu_1(\sigma_1', \sigma_2^*) \leq \mu_1(\sigma_1^*, \sigma_2^*) + negl(\lambda) \tag{3.2}$$

and

$$\mu_2(\sigma_1^*, \sigma_2') \leq \mu_2(\sigma_1^*, \sigma_2^*) + negl(\lambda) \tag{3.3}$$

Nash equilibria are well-suited to normal form games, where players move simultaneously and have full knowledge of the game state and payoffs. However, in the computational setting we must consider extensive form dynamic games of imperfect information, where players move sequentially and may be unaware of the game state or the payoffs of other players. Nash equilibria are known to be difficult to compute, and the problem is in PPAD [20, 21].

Strict Nash Equilibrium

A common refinement of Nash equilibria is to require the equilibrium to be *strict*, where the optimal strategy yields strictly greater utility than the alternative strategies. The computational variant of strict Nash equilibrium requires the introduction of a non-negligible gain in utility. This is necessary, as the computational Nash equilibrium assumes players will not switch strategies for a negligible gain, despite the alternate strategy yielding more (although negligible) utility. For example, a player bound to PPT succeeds in breaking a cryptographic primitive with at most negligible probability. Thus, the computational variant of the equilibrium concept assumes rational players will not switch strategies for this negligible utility gain. We use the definition of a strict computational Nash equilibrium from Fuchsbauer et al. [22].

Let k be the security parameter. The function $\epsilon : \mathbb{N} \to \mathbb{R}$ is *negligible* if for all $c > 0$ there is an $N_c > 0$ such that $\epsilon(k) < 1/k^c$ for all $k > N_c$. Let ρ_i be the Turing machine that implements strategy σ_i in the protocol Π. We write $\rho_i \not\approx \Pi$ if ρ_i does not yield equivalent play with respect to Π.

Definition 3.1.4 *Protocol* Π *induces a computational strict Nash equilibrium if*

1. Π *induces a computational Nash equilibrium*

2. *For any PPT strategy* σ_1' *with* $\sigma_1' \not\approx \Pi$*, there is a* $c > 0$ *such that* $\mu_1(\sigma_1, \sigma_2) \geq \mu_1(\sigma_1', \sigma_2) + 1/k^c$ *for infinitely many values of* k *(with an analogous requirement for a deviating player* P_2*).*

Kol and Naor [17] argue that the requirement of a strict (i.e., *unique*) equilibrium is problematic in the computational setting.

Iterated Deletion of Weakly Dominated Strategies

Refining the Nash equilibrium concept through the iterative deletion of weakly dominated strategies has been proposed to select a strategy when multiple Nash equilibria exist [23–25]. The intuition is that if a given strategy σ_i always yields equal or greater utility than a strategy $\sigma_j, j \neq i$, the strategy σ_j can be removed from consideration. Stated more formally, we follow the definition given by Katz [24]:

Definition 3.1.5 *Given a game $\Gamma = (\{A_i\}, \{\mu_i\})$, we say that action $a_i \in A_i$ is weakly dominated with respect to $A_{-i}(A_{-i} \overset{\text{def}}{=} \times_{j \neq i} A_j)$ if there exists a randomized strategy $\sigma_i \in \delta(A_i)$ such that:*

1. $\mu_i(\sigma_i, \vec{a_{-i}}) \geq \mu_i(a_i, \vec{a_{-i}}) \forall \vec{a_{-i}} \in A_{-i}$
2. $\exists \vec{a_{-i}} \in A_{-i}$ *such that* $\mu_i(\sigma_i, \vec{a_{-i}}) > \mu_i(a_i, \vec{a_{-i}})$

Definition 3.1.6 *Given a game $\Gamma = (\{A_i\}, \{\mu_i\})$ and $\hat{A} \subseteq A$, let $DOM_i(\hat{A})$ denote the set of strategies in $\hat{A}_i$ that are weakly dominated with respect to $\hat{A}_{-i}$. For $k \geq 1$, set $A_i^k \overset{\text{def}}{=} A_i^{k-1} \setminus DOM_i(A^{k-1})$. Set $A_i^\infty \overset{\text{def}}{=} \cap_k A_i^k$. A Nash equilibrium $\vec{\sigma}$ of Γ survives iterated deletion of weakly dominated strategies if $\sigma_i \in \delta(A_i^\infty)$ for all i.*

The primary issue with using iterative deletion of weakly dominated strategies to refine Nash equilibria is that *the order of deletion crucially affects the result.* That is, different equilibria may result depending on the order in which strategies are deleted [17, 26].

Stability with respect to Trembles

Fuchsbauer et al. [22] extend their definition of computational Nash equilibrium for the refinement of stability with respect to trembles. Informally, a *tremble* is the unlikely event where a rational agent chooses a strategy σ_i' rather than the optimal strategy σ_i. By accounting for trembles, nodes in the game tree that are off the equilibrium path may be dealt with more appropriately. Following the formal definition by Fuchsbauer et al., we say that ρ_i is δ-close to σ_i if ρ_i takes the following form: with probability $1 - \delta$ party i plays σ_i, while with probability δ it follows an arbitrary PPT strategy σ_i'. Thus, i will play the optimal strategy σ_i, but may deviate and play an arbitrary strategy σ_i' with some small probability δ.

Definition 3.1.7 *Protocol* Π *induces a computational Nash equilibrium that is stable with respect to trembles if*

1. Π *induces a computational Nash equilibrium*

2. *There is a noticeable function* δ *such that for any PPT strategy* ρ_2 *that is* δ*-close to* σ_2*, and any PPT strategy* ρ_1*, there exists a PPT strategy* $\sigma_1' \approx \Pi$ *such that* $\mu_1(\rho_1, \rho_2) \leq \mu_1(\sigma_1', \rho_2) + negl(k)$ *(with an analogous requirement for the case of deviations by player* P_2*)*

A primary concern with the concept of resiliency to trembles is that there may be multiple alternatives to the optimal strategy. How δ is divided amongst these sub-optimal strategies affects the final equilibrium. However, the notion of trembles can accurately model the case where a player is able to break the underlying cryptographic primitive. This is assumed to happen only with negligible probability, and is considered off the equilibrium path as a non-credible threat.

Correlated Equilibrium

A strong case for the use of *correlated* equilibrium can be made from the fact that a mediator is able to “recommend” a set of actions to the players. Thus, the action set follows a *joint* probability distribution, where each player learns the conditional distribution over the actions of other players. Correlated equilibria are commonly used in *signaling games*, where the ideal equilibrium is induced by an external signal available to all players. The standard example is the game of *chicken*, where a player may choose from $a_i \in \{\text{fast}, \text{slow}\}$. The mediator may be a traffic light, which signals a recommended strategy to all players. This equilibrium may be outside the convex hull of (mixed) Nash equilibria, which yields greater expected utility for players.

By recommending actions to players, greater utility may be achieved when players follow the mediator’s advice. Further, correlated equilibria are computationally less expensive (in *strategic games*) to compute than general Nash equilibrium. That is, computing Nash equilibria is NP-Hard, while computing correlated equilibria can be done in polynomial time by solving a linear program [27]. Correlated equilibria were considered in a computational setting by Urbano et al. [28], and specifically in the context of rational cryptography by Dodis et al. [16], Atallah et al. [29], and Gradwohl et al. [30]. Our objection to correlated equilibria is that they are defined only for strategic form games, rather than the more expressive extensive form games. The extension of correlated equilibria to extensive form games was considered by von Stengel et al. [31], but they demonstrated finding the optimal equilibria is NP-Hard. We give the definition for a correlated equilibrium of Dodis et al. [16]:

Definition 3.1.8 *A correlated equilibrium is a strategy profile* $s^* = s^*(A_1 \times A_2) = (s_1^*, s_2^*)$*, such that for any* (a_1^*, a_2^*) *in the support of* s^**, any* $a_1 \in A_1$ *and any* $a_2 \in A_2$*, we have* $\mu_1(a_1^*, s_2^*|a_1^*) \geq \mu_1(a_1, s_2^*|a_1^*)$ *and* $\mu_2(s_1^*, a_2^*|a_2^*) \geq \mu_2(s_1^*, a_2|a_2^*)$*.*

Bayesian Nash Equilibrium

Bayesian Nash equilibria (BNE) consider uncertainty with respect to a player's *type*, chosen by the fictitious player *Nature*. Thus, the optimal strategy for a player is conditioned on the probability of the other players' types. Bayesian Nash equilibria result in implausible equilibria in extensive form dynamic games as non-credible threats are not accounted for. The rational secret sharing problem was considered by Groce et al. [32] without assuming broadcast channels, using BNE as the solution concept. As BNE requires players fix their strategies before the game, they are unable to update their strategies based on information observed throughout the game.

Definition 3.1.9 *A* ***Bayesian game*** *consists of:*

1. *A finite set* N *of players.*

2. *A finite set* Ω *of states.*

3. *A set* A_i *of actions available to player* i*.*

4. *A finite set T_i of types for player i, and a function $\tau_i : \Omega \to T_i$ that assigns types to players.*

5. *A probability measure p_i on Ω for which $p_i(\tau_i^{-1}(t_i)) > 0 \forall t_i \in T_i$.*

6. *A preference relation $\precsim_i$ on the set of probability measures over $A \times \Omega$, where $A = \times_{j \in N} A_j$.*

From this, we are able to define a Bayesian Nash equilibrium:

Definition 3.1.10 *A* ***Bayesian Nash equilibrium*** *of a game $\Gamma = \langle N, \Omega, (A_i), (T_i), (\tau_i), (p_i), (\mu_i) \rangle$ for player i is an action set $a_i \in A_i$ such that:*

$$E[\mu_i(\sigma_i|\sigma_{-i}, t_i)] \geq E[\mu_i(\sigma_i'|\sigma_{-i}, t_i)] \tag{3.4}$$

where $\sigma_i : T_i \to A_i$ is the strategy mapping type space to action space.

Bayesian Nash equilibria are sufficient for strategic games, but lack the notion of *sequential rationality* necessary for application in extensive form games. We introduce the refinement of Bayesian Nash equilibria, namely perfect Bayesian equilibria, in Section 3.1.2.

3.1.2 Extensive Form Games

We now leave the setting of normal form games, and consider extensive form dynamic games where players move sequentially. Extensive form games are defined by Osborne et al. [13] as follows:

Definition 3.1.11 *An* ***extensive form game*** *Γ consists of:*

1. *A finite set N of players.*

2. *A (finite) set of sequences $\mathcal{H}$. The empty sequence $\emptyset$ is a member of $\mathcal{H}$. We let k denote the current decision node. If $(a^k)_{k=1,\ldots,K} \in \mathcal{H}$ and $L < K$ then*

$(a^k)_{k=1,\ldots,L} \in \mathcal{H}$. If an infinite sequence $(a^k)_{k=1}^{\infty}$ satisfies $(a^k)_{k=1,\ldots,L} \in \mathcal{H}$ for every positive integer L then $(a^k)_{k=1}^{\infty} \in \mathcal{H}$. A history $(a^k)_{k=1,\ldots,K} \in \mathcal{H}$ is a terminal history if it is infinite or if there is no a^{K+1} such that $(a^k)_{k=1,\ldots,K+1} \in \mathcal{H}$. The set of actions available after the nonterminal history h is denoted $A(h) = \{a : (h, a) \in \mathcal{H}\}$ and the set of terminal histories is denoted $\mathcal{Z}$. We let $\mathcal{H}^k$ denote the history through round k.

3. *A player function P that assigns to each nonterminal history (each member of $\mathcal{H} \setminus \mathcal{Z}$) a member of $N \cup \{Nature\}$. When $P(h) = Nature$, then Nature determines the action taken after history h.*

4. *For each player $i \in N$ a partition $\mathcal{I}_i$ of $\{h \in \mathcal{H} : P(h) = i\}$ with the property that $A(h) = A(h')$ whenever h and h' are in the same member of the partition. For $I_i \in \mathcal{I}_i$ we denote by $A(I_i)$ the set $A(h)$ and by $P(I_i)$ the player $P(h)$ for any $h \in I_i$. Thus, $\mathcal{I}_i$ is the information partition of player i, while the set $I_i \in \mathcal{I}_i$ is an information set of player i.*

5. *For each player $i \in N$ a preference relation $\precsim_i$ on lotteries[2] over $\mathcal{Z}$ that can be represented as the expected value of a payoff function defined on $\mathcal{Z}$.*

Throughout, we replace the preference relation $\precsim_i$ by a *utility function* $\mu_i : A \to \mathbb{R}$, such that $\mu_i(a) \geq \mu_i(b)$ when $b \precsim_i a$.

Information

In an extensive-form game, the notion of an *information set* I_i in an information partition $\mathcal{I}_i$ is used to describe the information available to player p_i at round k. If all players observe all moves by every other player, then the current node in the game tree is known with probability 1 and all information sets are called *singleton*, as they apply to one specific node in the tree. If some moves are unobserved, then

[2] Even if all actions are deterministic, moves by *Nature* can induce a probability distribution over the set of terminal histories.

players may only know they are in a *set* of possible game tree nodes. In this case, the information set is *non-singleton*, as the information set applies to more than one game tree node.

The degree of information available to a player in a game is characterized as *perfect* vs. *imperfect*:

1. Games of *perfect* information reveal the moves made by all players, and contain only singleton information sets; all players observe the actions of others.

2. Games of *imperfect* information have non-singleton information sets, where players do not observe a non-empty subset of the moves by other players.

From a cryptographic perspective, it is often the case that players in the protocol are unaware of the actions of other parties. Thus, such protocols are games of *imperfect* information[3], and any equilibrium concept used to model cryptographic protocols must address this uncertainty. As noted by Halpern et. al. [34], current frameworks (including their own) must incorporate an equilibrium concept that incorporates a player's beliefs about the computational abilities of other players. The *perfect Bayesian equilibrium* seems well-suited for addressing the uncertainties about the current game state, as well as the computational abilities of the other players, which we discuss further in Section 6.4.

Cryptographic protocols usually consider players with some private information that serves as their input to the protocol. The game theoretic literature views such inputs as a random move by Nature that determines the player's *type*:

Definition 3.1.12 *Let t_i denote the* ***type*** *of player p_i, where $T = T_1 \times \cdots \times T_n$ is the type space. Nature makes an initial move by sampling the type space distribution $\Delta(T)$ and assigning a type to each player. Player p_i's utility function is now defined as $\mu_i : (\vec{s}, \vec{t})$, where $\vec{s} = \{s_i\}_{1 \leq i \leq n}$ is the strategy profile and $\vec{t} = \{t_i\}_{1 \leq i \leq n}$ is the type profile.*

[3] The well-known Harsanyi transformation [33] allows any game of incomplete information to be transformed into a game of complete and imperfect information by introducing an initial move by Nature that assigns a *type* to each player.

Sequential Equilibrium

Sequential equilibrium was considered by Gradwohl et al. [30], and to a more full extent by Zhang et al. [35]. We use the definition of sequential equilibrium from Osborne et al. [13]:

Definition 3.1.13 *An assessment (β, μ) is a sequential equilibrium of an extensive game $\Gamma = \langle N, H, P, f_c, (\mathcal{I}_i), f(U_i) \rangle$, if it satisfies the following two conditions:*

1. *(β, μ) is sequentially rational: For every player $i \in N$ and every information set $I_i \in \mathcal{I}_i$, there holds: $U_i(\beta, \mu | I_i) \geq U_i((\beta_{-i}, \beta'_i), \mu | I_i)$ for every strategy β'_i of player i, where (β_{-i}, β'_i) is a strategy profiles that all players stick to the strategy β except that player i turns to the strategy β'_i, and $U_i((\beta_{-i}, \beta'_i), \mu | I_i)$ denotes player i's utility induced by this strategy profile and the belief system μ conditional on I_i being reached.*

2. *(β, μ) is consistent: There exists a sequence $((\beta^n, \mu^n))_{n=1}^{\infty}$ of assessments that converges to (β, μ) in Euclidian space and has the properties that each strategy profile β^n is completely mixed and that each belief system μ^n is derived from β^n using Bayes' rule.*

In the case of Zhang et al. [35], the authors consider extensive-form games with *simultaneous* moves. Specifically, they assume the existence of a broadcast channel. In this work, we make no such assumption and allow players to move sequentially. Sequential equilibria were originally proposed by Kreps and Wilson [36], and are a refinement of perfect Bayesian equilibria. However, the consistency requirement requires any assessment where an action is assigned zero probability to approximate an assessment where all actions have non-zero probability, and the definition has been considered overly stringent [13].

Perfect Bayesian Equilibrium

Formal definitions of perfect Bayesian equilibria (PBE) are usually not generalizable to all extensive form games, and contain the vague requirement that beliefs be updated according to Bayes' rule "whenever possible". Bonanno [37] gives a definition of PBE that is applicable for general extensive form games, but we will use the definition by Diaz et al. [38], as they go further by extending to general extensive form games as well as clarifying the ambiguous "whenever possible" updating requirement.

We first require that, for player $i \in N$, their *assessment* (σ_i, β_i) consisting of a strategy σ_i and a *belief* β_i about the game state, be sequentially rational:

Definition 3.1.14 *An assessment (σ_i, β_i) is (computationally)* ***sequentially rational*** *if, for every player $i \in N$ and every information set $I_i \in \mathcal{I}_i$, there holds:*

$$\mu_i(\sigma_i, \beta_i | I_i) + \epsilon(\lambda) \geq \mu_i((\sigma_{-i}, \sigma_i'), \beta_i | I_i) \tag{3.5}$$

for every strategy σ_i', a probability distribution over actions, of player i, where (σ_{-i}, σ_i') is a strategy profile where all players select strategy $\vec{\sigma}$ except that player i selects strategy σ_i', and $\mu_i((\sigma_{-i}, \sigma_i'), \beta_i | I_i)$ denotes player i's utility induced by this strategy profile and the belief system β_i, a probability distribution over game states, conditional on I_i being reached. The term $\epsilon(\lambda)$ denotes a negligible utility gain with respect to the security parameter λ, and σ_i is an efficiently computable strategy for player i with complexity $\mathscr{C}$.

Next, we give the definition of a *weak perfect Bayesian equilibrium*, which we build on to construct the final equilibrium concept that applies to general extensive form games:

Definition 3.1.15 *Let Γ be an extensive form game. An assessment (σ, β) is a* ***weak perfect Bayesian equilibrium*** *if it is sequentially rational and, on the path of σ, β is derived from σ from Bayes' rule.*

Building on the definition of a weak perfect Bayesian equilibrium, we reach the definition of a $\mathscr{C}$*-simple perfect Bayesian equilibrium*:

Definition 3.1.16 *Let* Γ *be an extensive form game. An assessment* (σ, β) *is a* ***$\mathscr{C}$-simple perfect Bayesian equilibrium*** *if, for each regular information set* I_i^k*, the restriction of* (σ, β) *to* $\Gamma_{I_i^k}(\sigma, \beta)$ *is sequentially rational and* β *is obtained by* conditional updating *from* σ *(i.e., the restriction of* (σ, β) *to* $\Gamma_{I_i^k}(\sigma, \beta)$ *is a weak perfect Bayesian equilibrium), where* σ *is efficiently computable by an interactive Turing machine (ITM) with complexity* $\mathscr{C}$*.*

For a proper introduction to game theory, Katz [12] describes the current effort to combine game theoretic and cryptographic concepts, while Osborne et al. [13], Nisan et. al. [39], and Fudenberg et. al. [40] give a complete introduction to game theory.

4 RATIONALITY APPLIED TO NON-CRYPTOGRAPHIC DOMAINS

Before applying game theory to a cryptographic setting, we first demonstrate the utility of a game theoretic approach to adversarial settings outside of a cryptographic context. Specifically, we consider *adversarial machine learning*, where a defender deploys a machine learning algorithm against an active adversary, whose strategy reacts to the presence of the algorithm. Adversarial machine learning covers a broad set of real world scenarios. For example, both spam and fraud detection consider adversaries that react adaptively to the presence of a machine learning algorithm.

In our setting we consider spam detection, where a rational adversary (spammer) has full knowledge of a defender's utility function and strategy. This choice is to give the adversary the best possible advantage against the defender, and to demonstrate that a solution exists yielding an advantage for the defender even in this extreme case. We model the interaction as a Stackelberg game, where the defender moves first by deploying the machine learning algorithm. We stress that Stackelberg games are a subset of those games expressible under our frameworks presented in Chapters 6 and 7, and the simplifying assumption that the adversary has full knowledge of the defender's strategy is to consider the worst possible scenario.

4.1 Introduction

Classical supervised learning assumes that training data is representative of the data expected to be observed in the future. This assumption is clearly violated when an intelligent adversary actively tries to deceive the learner by generating instances very different from those previously seen. The literature on adversarial machine learning aims to address this problem, but often assumes constraints that sophisticated and determined adversaries need not abide by. We model the adversarial machine learning problem by considering an unconstrained, but utility-maximizing, adversary. In addition, rather than modifying the learning algorithm to increase its robustness to adversarial manipulation, we use an output of an arbitrary probabilistic classifier (such as Naïve Bayes) in a linear optimization program that computes optimal randomized operational decisions based on machine learning predictions, operational constraints, and our adversarial model. Our approach is simpler than its predecessors, highly scalable, and we experimentally demonstrate that it outperforms the state of the art on several metrics.

4.2 Motivation

In a classical supervised learning setting one starts with a data set of instances generated according to some fixed distribution, and learns a function which (one hopes) effectively evaluates new instances generated from the same distribution. While this assumption is often reasonable, it is clearly violated in adversarial settings. For example, if machine learning is used for network intrusion detection, an intelligent adversary will try to avoid detection by deliberately changing behavior to appear benign.

We study the problem of *adversarial machine learning*, which we view as a game between a *defender* (*learner*), who uses past data to predict and respond to potential threats, and a collection of *attackers* who aim to bypass defensive response to their activities while achieving some malicious end. The issue of learning in adversarial environments has been addressed from a variety of angles, ranging from robustness to data corruption [41], to analysis of the problem of manipulating a learning algorithm [42]. The perspective we take here is that the training data accurately reflects current threats (i.e., has not been tampered with by an adversary), but the way we discriminate between benign and malicious behavior will influence the adversary to change its future actions. A classic example of this is spam detection, where it is quite clear that spammers deliberately manipulate email templates that they use in order to circumvent filters. Indeed, spam detection has been the prime motivator for the progress in adversarial machine learning, in large part because there is abundant public email data on which algorithms can be evaluated. However, spam, while a nuisance, is hardly the most pernicious of adversarial activities. Spear phishing, or sending targeted emails to groups of individuals in a specific organization in order to either exfiltrate information or introduce malware, is both more malicious than typical spam, and qualitatively different. Regular spam is untargeted, and it is, perhaps, reasonable to posit an adversarial model in this context where the attacker (spammer) manipulates the distribution of future instances in some relatively limited way (a common assumption is that the spammer chooses a linear transformation) [43–46]. Such a model, however, is clearly too limited when an attacker deliberately targets an organization: in this case, the attacker will go to great lengths to circumvent detection systems, and as long as machine learning is not perfect (which it never is), it is in all likelihood vulnerable. Additionally, while the adversary may well be constrained, it is highly unlikely that it is constrained in precisely the way modeled; indeed, it seems more reasonable that the attacker faces a cost-benefit tradeoff in a given attack setting, rather than hard constraints.

There is another important feature of most of the literature on machine learning techniques aimed at adversarial settings, such as network anomaly and intrusion detection. Almost universally, the predictions produced by the application of learning are conflated with operational decisions based on these predictions, even though these are conceptually distinct [43–49]. This has important consequences. First, the adversary does not respond to predictions per se, but to operational actions based on these. For example, even if the prediction identified an input as malicious, as long as this prediction is not actualized in operations, the adversary would have no reason to change behavior. Second, learned predictions typically do not account for operational costs and constraints (although the literature on cost-sensitive learning attempts to do so to some degree [50]). For example, even if an input is identified as malicious, it may not be worth the cost to act on it if the damage is minimal (say, if an input is a

Windows virus and you have only Linux machines). Third, machine learning is best suited for the task of prediction, and in that sense we should focus on trying to develop the most predictive algorithm given the available data. Operational decisions, on the other hand, are most meaningfully an outcome of an optimization problem under uncertainty, and the adversary's response is naturally captured in this context (unless there is specific data about how an adversary may respond to operational decisions, which there rarely is).

The main conceptual contribution of this work is to separate the problem of prediction, for which machine learning is used, and the problem of computing optimal operational decisions based on such predictions in the face of a sophisticated adversary. To this end, we introduce a linear optimization problem in which the objective is the defender's expected utility, balancing the value of good traffic that is allowed (e.g., non-malicious or non-spam email), and the cost due to missed malicious activity, as well as constraints on the fraction of all traffic that can be operationally acted on. We assume that "acting" on a particular observed input (e.g., a suspicious email or network packet) is inherently costly, representing, for example, deep packet inspection, or in-depth investigation by cyber security professionals prompted by an alert. Throughout, we use the word *inspect* to mean any costly operational activity on a suspected malicious input. Overall, we presume a two-stage process: first, a machine learning tool is trained on historical data, and second, the linear program is solved to compute an optimal operational policy, using the predictions produced by the learning algorithm. In our context, it is especially significant to use highly informative learning methods, i.e., those which produce, for each input x, a probability that x is malicious. As a direct consequence of our approach, operational decisions are in general randomized, unless it is either too costly to inspect anything, or inspection is cheap so that everything plausibly malicious is inspected. Thus, our approach can be viewed as a principled instance of moving target (or dynamic) defense in the context of intrusion detection, which is also where such randomized defensive methods may be relatively easy to deploy. Finally, because our model amounts to solving a linear program, the approach we present is very simple, and highly scalable. Nevertheless, we show experimentally that it outperforms current art in adversarial machine learning on public spam data, as well as data generated using an artificial utility-maximizing adversary.

4.3 Model

We consider the problem of adversarial binary classification over a space $\mathcal{X}$ of inputs, where each input feature vector $\vec{x} \in \mathcal{X}$ can be categorized as benign or malicious. The defender, $\mathcal{D}$, starts with a data set of labeled instances, $\mathcal{I}$, such that $\mathcal{I} = \{(\vec{x}_1, y_1), \ldots, (\vec{x}_m, y_m)\}$. We assume to accurately represent the current distribution of input instances and corresponding categories.[1] $\mathcal{D}$ then uses an algorithm of choice, such as Naive Bayes, to obtain a probabilistic classifier $p(\vec{x})$ which assigns to an arbitrary input vector a probability that it (or, rather, a producer of it) is malicious. In traditional application of machine learning, adversarial or not, one would then use a threshold, θ, and classify an instance $\vec{x}$ as malicious if $p(\vec{x}) \geq \theta$, and benign otherwise, with adversarial aspects of the problem folded into the algorithm that derives the function $p(\cdot)$. It is on this point that our approach diverges from current art. Specifically, we introduce a function $q(\vec{x}, p(\cdot)) \in [0, 1]$ which prescribes a possibly randomized operational decision (e.g., the probability of filtering an email or manually investigating an observed network access pattern) for an instance $\vec{x}$ given a prediction $p(\vec{x})$. Clearly, the threshold function typically used is a special case, but we will productively consider alternative possibilities. To simplify notation, where $p(\cdot)$ is clear from context, we use instead $q(\vec{x})$, keeping in mind its implicit dependence on the prediction made by the learning algorithm.

We model the adversarial machine learning setting as a Stackelberg game between a defender and a population of attackers. In this game, the defender moves first, choosing $q(\cdot)$. Next, the attackers learn $q(\cdot)$ (for example, through extensive

[1] The problem of adversarial tampering of such *training* data is outside the scope of our work, and can be viewed as an extension of our setup.

probing), and each attacker subsequently chooses an input vector $\vec{x}$ (e.g., a phishing email) so as to maximize their expected return (a combination of bypassing defensive countermeasures and achieving a desired outcome upon successfully penetrating the defense, such as a high response rate to a phishing attack). Our assumption that the operational policy $q(\cdot)$ is known to attackers reflects threats that have significant time and/or resources to probe and respond to defensive measures, a feature characteristic of advanced cyber criminals [51].

We view the data set $\mathcal{I}$ of labeled malware instances as representing *revealed preferences* of a sample of attackers, that is, their preference for input vectors $\vec{x}$ (if an attacker prefered another input $\vec{x}'$, we assume that this attacker would have chosen $\vec{x}'$ instead of $\vec{x}$). To appreciate this modeling choice, it is worth noting that much variation in malware is due either to differences in perpetrators themselves, or differences in their goals (even for the same attackers), and labeled data provides information, albeit indirectly, about these differences. Therefore, in our framework $p(\vec{x})$ takes on a dual-meaning: first, it is the probability that $\vec{x}$ reflects a malicious action, and second, if malicious, $\vec{x}$ represents an attacker's "type", or ideal method of attack. Insofar as we view an attack $\vec{x}$ as ideal for an attacker, it is just as natural to posit that an attacker would prefer attack patterns that are close to $\vec{x}$ in feature space to those distant from it. For example, a model in which an attacker would minimize the number of feature values to alter in order to bypass defensive activities has this characteristic, as do models which use a regularization term to reduce the scale of attack manipulation of data [43, 45, 46, 52, 53].

Suppose that if an attack $\vec{x}$, succeeds, the attacker gains $V(\vec{x})$, which is also the value lost to the defender. On the other hand, if an attack is filtered or caught by the defender, both receive 0. Finally, if the attacker with a preference for $\vec{x}$ chooses an alternative attack vector $\vec{x}'$, his utility from successfully bypassing defenses becomes $V(\vec{x})Q(\vec{x}, \vec{x}')$, where

$$Q(\vec{x}, \vec{x}') = e^{-\delta||\vec{x}-\vec{x}'||}, \tag{4.1}$$

with $||\cdot||$ a norm (we use Hamming distance, as our feature vector is binary), and δ corresponding to importance of being close to the preferred $\vec{x}$. Observe that when $\delta = 0$, the attacker is indifferent among attack vectors, and all that matters is success at bypassing defensive action, while $\delta \to \infty$ results in an attacker who does not react to defensive action at all, either because it is too costly to change, or because this attacker simply does not have the capability of doing so (e.g., someone who merely reuses attack templates previously developed by others). The full utility function of an attacker with type $\vec{x}$ for choosing another input $\vec{x'}$ when the defense strategy is $q(\cdot)$ is then

$$\mu(\vec{x}, \vec{x'}; q) = V(\vec{x})Q(\vec{x}, \vec{x'})(1 - q(\vec{x'})), \tag{4.2}$$

since $1 - q(\cdot)$ is the probability that the attacker successfully bypasses the defensive action.

While the above attacker model admits considerable generality, we assume that attackers fall into two classes: adaptive, as described above, and static, corresponding to the limiting case of $\delta \to \infty$. Let $v_t(\vec{x}; q)$ be the value function of an attacker with class (type) t and preference for $\vec{x}$, when the defender chooses a policy q. $v_t(\vec{x}; q)$ represents the maximum utility that the attacker with type t can achieve given q. For a static attacker, the value function is

$$v_S(\vec{x}; q) = V(\vec{x})(1 - q(\vec{x})), \tag{4.3}$$

that is, a static attacker always uses his preferred input $\vec{x}$, and receives his corresponding value for it whenever the defender (operator) does not take action upon observing $\vec{x}$. For an adaptive attacker, the value function is

$$v_A(\vec{x}; q) = \max_{\vec{x'} \in \mathcal{X}} \mu(\vec{x}, \vec{x'}; q), \tag{4.4}$$

that is, the maximum utility that the attacker obtains from using an *arbitrary* input $\vec{x'}$ (that is, we assume that the adaptive attacker is unconstrained). Finally, let P_A

be the probability that an arbitrary malicious input was generated by an adaptive adversary; the probability that the adversary was static is then $P_S = 1 - P_A$.

Having described in some detail our model of the adversarial response to defensive choice of $q(\cdot)$, we now turn to the objective of the defender. At the high level, a natural goal for the defender is to maximize expected value of benign traffic that is classified as benign, less the expected losses due to attacks that successfully bypass the operator (i.e., incorrectly classified as benign). Presently, we show that a special case of this is equivalent to maximizing accuracy or minimizing loss. To formalize, we make two assumptions. First, we assume that the set of all possible instances $\mathcal{X}$ is finite, and use $\vec{q}$ and $\vec{p}$ as vectors corresponding to $q(\vec{x})$ and $p(\vec{x})$ respectively, using some fixed arbitrary ordering over $\mathcal{X}$. This assumption is clearly unrealistic (even if $\mathcal{X}$ is technically finite, it will typically be intractably large), but will help with exposition below. We subsequently (in Section 4.4) describe how to apply our approach in practice, when this assumption will not hold. Second, we assume that the defender gains a positive value $G(\vec{x})$ from a benign input $\vec{x}$ only if it is not inspected. In the case of email traffic, this is certainly sensible if our action is to filter a suspected email. More generally, inspection can be a lengthy process, in which case we can interpret $G(\vec{x})$ as the value of time lost if $\vec{x}$ is, in fact, benign, but is carefully screened before it can have its beneficial impact. Formally, we suppose that the defender maximizes $U_{\mathcal{D}}(\vec{q}, \vec{p}, \mathcal{X})$, defined as

$$\begin{aligned} U_{\mathcal{D}}(\vec{q}, \vec{p}, \mathcal{X}) = \sum_{\vec{x} \in \mathcal{X}} & [(1 - q(\vec{x}))G(\vec{x})(1 - p(\vec{x})) - \\ & p(\vec{x})(P_S v_S(\vec{x}; q) + P_A v_A(\vec{x}; q))] \,. \end{aligned} \tag{4.5}$$

To appreciate that this formal definition of the defender's objective is sensible, let us first rewrite it for a special case when $V(\vec{x}) = G(\vec{x}) = 1$ and $P_S = 1$, reducing the utility function to

$$\sum_{\vec{x} \in \mathcal{X}} (1 - q(\vec{x}))(1 - p(\vec{x})) - p(\vec{x})(1 - q(\vec{x})). \tag{4.6}$$

Since $p(x)$ is constant, this is equivalent to minimizing

$$\sum_{\vec{x} \in \mathcal{X}} q(\vec{x})(1 - p(\vec{x})) + p(\vec{x})(1 - q(\vec{x})), \tag{4.7}$$

or, for each $\vec{x}$, the sum of probability that it is benign and misclassified as malicious, and probability that it is malicious but misclassified as benign; which is to say, the expected loss.

The final aspect of our model is a resource constraint on the defender. Sommer and Paxson [48] identify the cost of false positives and the gap between the output of machine learning algorithms and its use in operational decisions as two of the crucial gaps that prevent widespread use of machine learning in network intrusion detection. Our framework directly addresses the latter point, and we now turn focus to the former. False positives are quite costly because following up on an alert is a very expensive proposition, involving the use of a scarce resource, a security expert's time understanding the nature of the alert. In practice, it is simply not feasible to follow up on every alert, and there is a need for a principled approach that accounts for such budget constraints. An additional cost of false positives comes from the fact that, depending on the nature of operational decision, it results in some loss of value, either because a valuable email gets filtered, or because important communication is delayed due to deeper inspection it needs to undergo. In fact, $G(\vec{x})$ in our model already serves to quantify this loss of value. We handle the typically harder constraint on defensive resources by introducing a budget constraint, where we ensure that our solution inspects at most a fraction c of events, on average.

4.4 Computing Optimal Operational Decisions

Now that we have described our model of adversarial machine learning, the natural next question is: how do we solve it? Since our objective and constraints are linear (using the assumption that the attacker's gains translate directly into defender's losses), we can formulate our optimization problem as the following linear program (LP):

$$\max_{\vec{q}} \quad U_{\mathcal{D}}(\vec{q}, \vec{p}, \mathcal{X}) \tag{4.8a}$$

$$\text{s.t. :} \quad 0 \leq q(\vec{x}) \leq 1 \qquad \forall\, \vec{x} \in \mathcal{X} \tag{4.8b}$$

$$v_A(\vec{x}; q) \geq \mu(\vec{x}, \vec{x'}; q) \qquad \forall\, \vec{x}, \vec{x'} \in \mathcal{X} \tag{4.8c}$$

$$v_S(\vec{x}; q) = V(\vec{x})(1 - q(\vec{x})) \qquad \forall\, \vec{x} \in \mathcal{X} \tag{4.8d}$$

$$\sum_{\vec{x}} q(\vec{x}) \leq c|\mathcal{X}|. \tag{4.8e}$$

Since the number of variables in this LP is linear in $|\mathcal{X}|$, while the number of constraints is quadratic in this quantity, clearly we cannot hope to use this when the space of all possible inputs is large (let alone infinite). Note, however, that we only need to compute the decisions $q(\vec{x})$ for inputs $\vec{x}$ we actually see in reality. Therefore, in practice we can batch observations into manageable sets $\bar{\mathcal{X}} \subset \mathcal{X}$, and solve this optimization program using inputs restricted to $\bar{\mathcal{X}}$.[2]

A natural sanity check that our formulation is reasonable is that the solution is particularly intuitive when there is no budget constraint or adaptive adversary. We now show that in this case, the policy $q(\vec{x})$ which uses a simple threshold on $p(\vec{x})$ (as commonly done) is, in fact optimal.

[2] It may seem that this setup violates our assumption that the attacker observes $q(\vec{x})$. However, our assumption amounts to an attacker observing many instances of solutions to this optimization problem, allowing the attacker to infer $q(\vec{x})$ for an arbitrary input $\vec{x}$ under consideration. Thus, it is still accurate to characterize an attacker as responding to the policy $q(\vec{x})$.

Proposition 4.4.1 *Suppose that $P_A = 0$ and $c = 1$ (i.e., no budget constraint). Then the optimal policy is*

$$q(\vec{x}) = \begin{cases} 1 & \text{if } p(\vec{x}) \geq \frac{G(\vec{x})}{G(\vec{x})+V(\vec{x})} \\ 0 & o.w. \end{cases} \tag{4.9}$$

Proof Since we consider only static adversaries and there is no budget constraint, the objective becomes

$$\max_{\vec{q}} \sum_{\vec{x} \in \mathcal{X}} \left[(1 - q(\vec{x}))G(\vec{x})(1 - p(\vec{x})) - p(\vec{x})v_S(\vec{x})\right], \tag{4.10}$$

and the only remaining constraint is that $q(\vec{x}) \in [0, 1]$ for all $\vec{x}$. Since now the objective function is entirely decoupled for each $\vec{x}$, we can optimize each $q(\vec{x})$ in isolation for each $\vec{x} \in \mathcal{X}$. Rewriting, maximizing the objective for a given $\vec{x}$ is equivalent to minimizing $q(\vec{x})[G(\vec{x}) - p(\vec{x})(G(\vec{x}) + V(\vec{x}))]$. Whenever the right multiplicand is negative, the quantity is minimized when $q(\vec{x}) = 1$, and when it is positive, the quantity is minimized when $q(\vec{x}) = 0$. Since $p(\vec{x}) \geq \frac{G(\vec{x})}{G(\vec{x})+V(\vec{x})}$ implies that the right multiplicand is negative (more accurately, non-positive), the result follows. ■

While traditional approaches threshold an odds ratio (or log-odds) rather than the probability $p(\vec{x})$, the two are, in fact equivalent. To see this, let us consider the generalized (cost-sensitive) threshold on odds ratio used by the Dalvi et al. [52] model. In their notation, $U_{\mathcal{C}}(+,+)$, $U_{\mathcal{C}}(+,-)$, $U_{\mathcal{C}}(-,+)$, and $U_{\mathcal{C}}(-,-)$ denote the utility of the defender (classifier) when he correctly identifies a malicious input, incorrectly identifies a benign input, incorrectly identifies a malicious input, and correctly identifies a benign input, respectively. In our setting, we have $U_{\mathcal{C}}(+,+) = 0$ (i.e., no loss), $U_{\mathcal{C}}(+,-) = 0$ (and capture the costs of false positives as operational constraints instead), $U_{\mathcal{C}}(-,+) = -V(\vec{x})$, and $U_{\mathcal{C}}(-,-) = G(\vec{x})$ (note that we augment the utility functions to depend on input vector $\vec{x}$). The odds-ratio test used by Dalvi et al. therefore checks

$$\frac{p(\vec{x})}{1-p(\vec{x})} \geq \frac{U_{\mathcal{C}}(-,-)-U_{\mathcal{C}}(+,-)}{U_{\mathcal{C}}(+,+)-U_{\mathcal{C}}(-,+)} = \frac{G(x)}{V(x)}. \tag{4.11}$$

and it is easy to verify that inequality 4.11 is equivalent to the threshold test in Proposition 4.4.1.

Consider now a more general setting where $P_A = 0$, but now with a budget constraint. In this context, we now show that the optimal policy is to first set $q(\vec{x}) = 0$ for all $\vec{x}$ with $p(\vec{x})$ below the threshold described in Proposition 4.4.1, then rank the remainder in descending order of $p(\vec{x})$, and assign $q(\vec{x}) = 1$ in this order until the budget is exhausted.

Proposition 4.4.2 *Suppose that $P_A = 0$ and $c|\mathcal{X}|$ is an integer. Then the optimal policy is to let $q(\vec{x}) = 0$ for all $\vec{x}$ with*

$$p(\vec{x}) < \frac{G(\vec{x})}{G(\vec{x})+V(\vec{x})}. \tag{4.12}$$

Rank the remaining $\vec{x}$ in descending order of $p(\vec{x})$ and set $q(\vec{x}) = 1$ for the top $c|\mathcal{X}|$ inputs, with $q(\vec{x}) = 0$ for the rest.

Proof The LP can be rewritten so as to minimize

$$\sum_{\vec{x}} q(\vec{x})[G(\vec{x}) - p(\vec{x})(G(\vec{x})+V(\vec{x}))] \tag{4.13}$$

subject to the budget constraint. By the same argument as above, whenever $p(\vec{x})$ is below the threshold, the optimal $q(\vec{x}) = 0$. Removing the corresponding $\vec{x}$ from the objective, we obtain a special knapsack problem in which the above greedy solution is optimal, since the coefficient on the budget constraint is 1. ■

In a nutshell, Proposition 4.4.2 suggests an intuitive policy that whenever the budget constraint binds, we should simply inspect the highest priority items. Therefore, randomization becomes important only when there is an adversary actively responding to our inspection efforts.

4.5 Experiments

Experimentally validating a scheme for adversarial machine learning is inherently difficult using publicly available data, such as spam. The reason is that insofar as this data captures evolution of spam, it is in response to the ecology of spam filters, and, in addition, the precise nature of the actually deployed filters is not a part of such public databases. In addition, spam is in itself a rather benign attack, as compared to, say, a spear phish aimed at stealing intellectual property. The latter is clearly much more targeted, much more costly to the organizations, and involves far more sophisticated and adaptive adversaries. All of the previous attempts to address machine learning in adversarial settings struggled with this problem, and evaluation is typically either (a) nevertheless involving public spam data [43–46], or (b) generating synthetic data according to their model of the adversary [46, 52]. We do *both*: evaluate our approach on *actual public spam data* (Section 4.5.2), and using synthetically generated attacks (Section 4.5.3). There is a clear limitation of using one's own model for validation: it naturally favors the proposed approach if the model is assumed to be an accurate description of attacker's behavior. We address this limitation by evaluating the robustness of our approach to errors in the adversarial model it uses (Section 4.5.4).

4.5.1 Setup

In all our experiments we use the TREC spam corpora from 2005 – 2008. In Section 4.5.2, we use this data as is to compare the performance of our approach in a spam filtering task, compared to state-of-the-art alternatives. In Sections 4.5.3 and 4.5.4 we use this data only for training, and simulate adversarial behavior according to our model (as done, for example, by Dalvi et al. [52]). Throughout, we performed 10-fold cross-validation and analyzed the results using the approach outlined by Demšar [54]. We compare our approach against using a classifier it is based upon (i.e., $p(\vec{x})$) directly using pairs of the form $\{\mathcal{C}, E[OPT(\mathcal{C})]\}$, where $\mathcal{C}$ is the classifier providing $p(\vec{x})$ for our model, and $E[OPT(\mathcal{C})]$ denotes our approach using $\mathcal{C}$. We use Friedman's test to compute the p-values, using $N = 4$ data sets and $k = 2$ classifiers, as we are only concerned with the performance of our approach with respect to the corresponding classifier. We use the post-hoc Bonferroni test, which does not alter α as $\alpha/(k-1) = \alpha$ when $k = 2$, as in all of our comparisons. As detailed by Salzberg [55], the feature criteria were chosen to optimize the performance of Naïve Bayes on the TREC 2005 spam corpus. Feature vectors were generated from the raw emails, and the same criteria were used for each corpus. None of the algorithms have been optimized or tuned on *future* data. In Section 4.5.2, we train on a fold of the TREC 2005 data, evaluate the performance over the test fold for the TREC 2005 corpus, and test over the entire set of future corpora.

Our approach uses predictions $p(\vec{x})$ obtained using three existing classifiers: Naïve Bayes (our non-adversarial baseline), and the adversarial classifiers developed by Bruckner and Scheffer [45] and Dalvi [52], which are state-of-the-art alternatives.[3] We denote the expected utility of our approach as $E[OPT(\cdot)]$, where the argument is an existing classifier that provides $p(\vec{x})$. We solve the LP (Equations 4.8a-4.8e) using CPLEX version 12.2.

Our optimization approach explicitly bounds the number of instances that can be inspected. We consider two principled ways of imposing the same restriction on existing classifiers:

1. Let $\bar{\mathcal{X}}$ be all $\vec{x}$ with $p(\vec{x})$ above a threshold from Proposition 4.4.1. Then set $q(\vec{x}) = 1$ if $|\bar{\mathcal{X}}| \leq c|\mathcal{X}|$, while $q(\vec{x}) = c$ otherwise. This policy is optimal when there are only stationary attackers and $p(\vec{x}) \in \{0, 1\}$. We use this as the default.

2. Rank the instances in descending order of $p(\vec{x})$, and set $q(\vec{x}) = 1$ for the first $c|\mathcal{X}|$ of these (as long as $p(\vec{x})$ exceeds the threshold from Proposition 4.4.1).

[3] We use the variant of Bruckner and Scheffer's classifier with the linear loss function.

This policy is optimal when there are only stationary attackers, as we showed in Proposition 4.4.2. We call this “Naïve Ranking”.

We used the **ifile** tool by Rennie [47] to select tokens for the feature vectors. Many of the desirable tokens for the TREC 2005 corpus are specific to the company where the emails were collected. Since our experiments evaluate performance on future TREC data which includes emails collected from different sources, we selected a subset of tokens that are environment invariant. Specifically, we restricted attention to the 14 tokens shown in Figure 4.1, and created a binary feature for each that indicates its presence in an email.

$$\begin{aligned}\vec{x} = \{ \quad & font, td, http, nbsp, span, color, content, \\ & div, face, net, src, www, charset, strong \quad \}\end{aligned}$$

Figure 4.1.: The Boolean Feature Vector Tokens

We compare the algorithms below using an empirical utility function, which we normalize to facilitate comparison across different cost settings (this utility is a generalization of accuracy that accounts for costs $V(\vec{x})$ and $G(\vec{x})$). Specifically, given data with true labels, $y(\vec{x})$, we can express total accuracy on training/test data, weighted with the corresponding costs of false positives and false negatives, as

$$\sum_{\vec{x}}(1-y(\vec{x}))(1-q(\vec{x}))G(x) + \sum_{\vec{x}} y(\vec{x})q(\vec{x})V(\vec{x}). \tag{4.14}$$

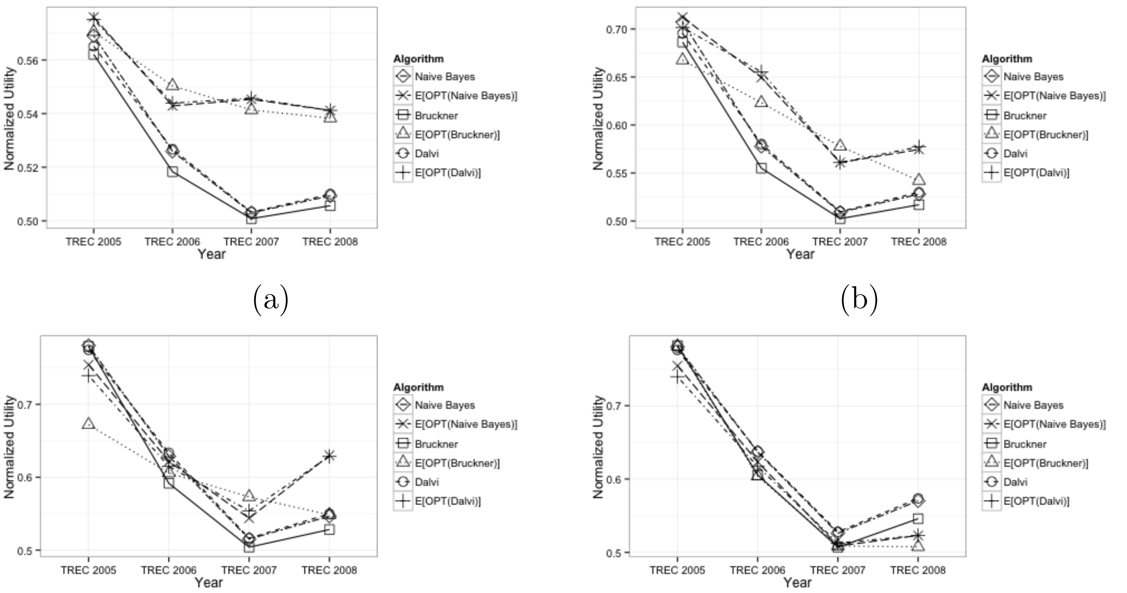
In our experiments, we fix $G(\vec{x}) = G$, and $V(\vec{x}) = V$ for all $\vec{x}$. Let $|\mathcal{X}_{TN}|$ denote the number of true negatives, $|\mathcal{X}_{TP}|$ be the number of true positives, $|\mathcal{X}^-| = \sum_{\vec{x}} y(\vec{x})(1-q(\vec{x}))$ be the expected number of false negatives, and $|\mathcal{X}^+| = \sum_{\vec{x}}(1-y(\vec{x}))q(\vec{x})$ be the expected number of false positives. After dropping the terms that do not depend on $q(\vec{x})$, we can rewrite Equation 4.14 as $U_{\mathcal{D}} = w(|\mathcal{X}_{TN}| - |\mathcal{X}^+|) + (|\mathcal{X}_{TP}| - |\mathcal{X}^-|)$, where $w = \frac{G}{V}$ (note that when $w = 1$, this measure becomes exactly the total expected accuracy achieved by $q(\vec{x})$). In the reported results, we normalize this by the total utility of a perfect classifier, obtaining the empirical *normalized utility*

$$\tilde{U}_{\mathcal{D}} = 1 - \frac{w|\mathcal{X}^+| + |\mathcal{X}^-|}{w|\mathcal{X}_{TN}| + |\mathcal{X}_{TP}|}. \tag{4.15}$$

4.5.2 Performance on Public Spam Data

Our first set of experiments is a direct comparison of the performance of our model as compared to state-of-the-art alternatives described above evaluated on public spam data. In this experiment, we use TREC 2005 data to train the classifiers, compute the optimal $q(\vec{x})$ for our approach while using the other alternatives as prescribed, and evaluate (by computing the expected normalized utility shown in Equation 4.15) on TREC data for years 2005-2008. As in all past evaluations of adversarial machine learning algorithms [43–46] we do not retrain the classifiers, since our intent is not merely to demonstrate value on spam data, but to anticipate far more actively adversarial environments in which attackers adapt to defense decisions quickly, and the defender wishes to have success in *anticipating adversarial response.*

Our first set of results, shown in Figure 4.2, compares our optimization-based approach to alternatives when $V(\vec{x}) = G(\vec{x}) = 1$ for all $\vec{x}$ and $P_A = 0.5$ (this choice was made somewhat arbitrarily and not optimized to data), under a variety of budget constraints. Since our optimization can take as input an arbitrary $p(\vec{x})$, we compare the results of using the alternative machine learning approaches as input. From considering the four plots in Figure 4.2, each corresponding to a different budget constraint, it is apparent that the relative advantage of our approach (using any of the alternative $p(\vec{x})$ in the optimization problem) is pronounced (exhibiting 10-20% improvement over baseline) when the budget is relatively tight. Additionally, as we would intuitively expect, our approach performs better than alternatives as we move further into the future (giving the spammers more time to react to countermeasures from 2005). With a sufficiently generous budget constraint, it is also interesting to observe the tradeoff one would expect: the accuracy of our approaches is inferior



(a)

(b)

(c)

(d)

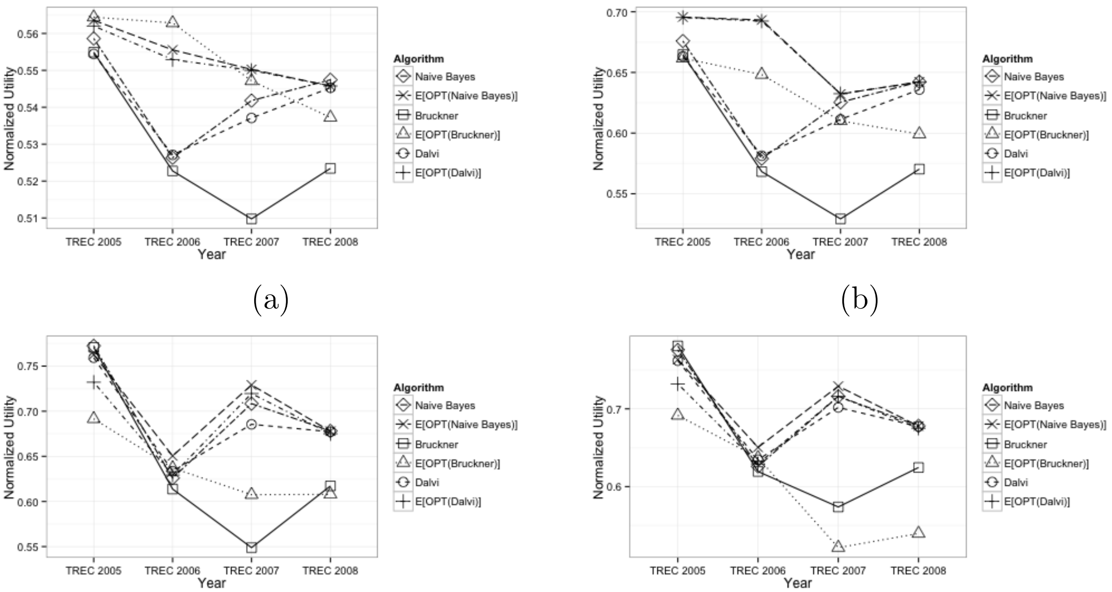
Figure 4.2.: Comparison of algorithms on TREC data, trained on year 2005, and tested on years 2005-2008. Our approach is labeled as $E[OPT(\cdot)]$, where the parameter is the classifier that serves as our $p(\vec{x})$. We use the following parameters: $\delta = 1$, $V(x) = G(x) = 1, P_A = 0.5$. (a) $c = 0.1$; (b) $c = 0.3$; (c) $c = 0.5$; (d) $c = 0.9$.

to alternatives on training data, but the decisions are more robust to adversarial manipulation embedded in future data.

In Figure 4.3, our second set of results demonstrate that even after retraining the classifier on all years prior to and including the current year, we typically outperform the alternatives.

In Figure 4.4, we consider a higher cost of malware relative to benign instances, fixing $G(x) = 1$ and considering $V(x) = 2$ and 10. Perhaps the most surprising finding in these plots is that here Naïve Bayes outperforms Dalvi et al. and Bruckner and Scheffer in several instances, even though these are specifically tailored to adversarial situations. Our approaches, however, perform consistently better than the alternatives.

We performed a statistical comparison between our approach and a corresponding classifier $p(\vec{x})$ on which it is based using Friedman's test with the post-hoc Bon-



(a) (b)

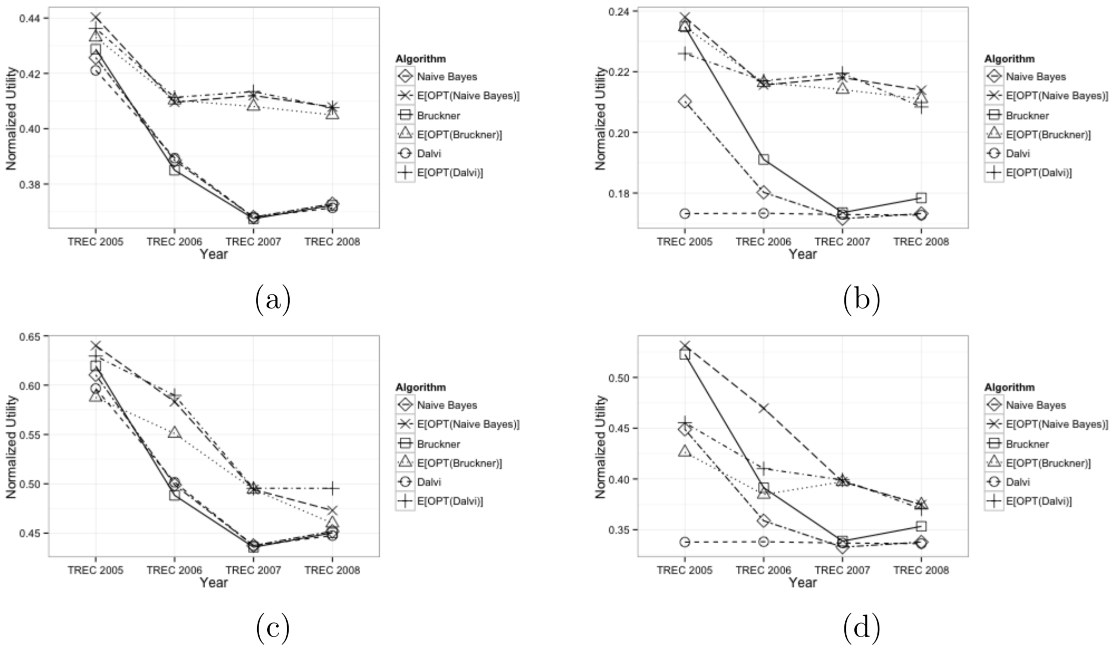
(c) (d)

Figure 4.3.: Comparison of algorithms on TREC data, trained on all years prior to and including the test year. Our approach is labeled as $E[OPT(\cdot)]$, where the parameter is the classifier that serves as our $p(\vec{x})$. We use the following parameters: $\delta = 1$, $V(x) = G(x) = 1, P_A = 0.5$. (a) $c = 0.1$; (b) $c = 0.3$; (c) $c = 0.5$; (d) $c = 0.9$.

ferroni correction [54]. For all classifier pairs of the form $\{\mathcal{C}, E[OPT(\mathcal{C})]\}$ with $\mathcal{C} \in \{\textit{Naïve Bayes}, \textit{Bruckner}, \textit{Dalvi}\}$ and for $c \in \{0.1, 0.3\}, V(x) \in \{1, 2, 10\}$, our approach is statistically better than the alternative at the $\alpha = 0.05$ confidence level.

4.5.3 Performance with an Optimizing Attacker

Evaluating performance on future TREC data as done above is fundamentally limited since this data set represents spam, where adversaries generally do not target a specific classifier or organization but a relatively large population of spam filters. In contrast, our approach is tailored to highly sophisticated and targeted attacks. The problem is that data of this nature is highly sensitive and not publicly available. Indeed, the ideal, infeasible, experiment is to observe adversarial response to our model as well as other alternatives and evaluate the approaches with respect to such



(a)

(b)

(c)

(d)

Figure 4.4.: Comparison of algorithms on TREC data, trained on year 2005, and tested on years 2005-2008. Our approach is labeled as $E[OPT(\cdot)]$, where the parameter is the classifier that serves as our $p(\vec{x})$. We fix $\delta = 1, G(x) = 1, P_A = 0.5$, and vary $V(x)$ and c. (a) $V(x) = 2$, $c = 0.1$; (b) $V(x) = 10$, $c = 0.1$; (c) $V(x) = 2$, $c = 0.3$; (d) $V(x) = 10$, $c = 0.3$.

adversarial response. As the next best alternative which has become relatively standard [46,52], we complement the spam evaluation in Section 4.5.2 with an alternative set of experiments aimed at modeling highly adaptive adversaries who maximize their expected utility in response to operational decisions $q(\vec{x})$. Specifically, we assume that a machine learning algorithm provides an accurate assessment of current or near-term threats, $p(\vec{x})$, and that all of the attackers are adaptive (i.e., that $P_A = 1$). Moreover, we assume that the learner/defender has correct knowledge of these parameters, as well as the parameter of the adaptive attacker's objective function, δ (we relax this assumption in Section 4.5.4). Finally, we let $V(x) = G(x) = 1$ for all $\vec{x}$. For each year Y in the TREC data set (e.g., $Y = 2005$), we perform 10-fold cross-validation. However, rather than computing the utility directly using the test fold, we compute the expected utility, assuming the adaptive attacker described in Section 4.3. Equivalently, we can think of this as the following exercise: for each $\vec{x}$ in the test fold, we assign it a benign label with probability $1 - p(\vec{x})$, assign a malicious label with probability $p(\vec{x})P_S$, and with probability $p(\vec{x})P_A$ generate a new malicious input $\vec{x}'$ that maximizes the attacker's expected utility given $q(\vec{x})$ computed by our algorithm.

In the first set of experiments, we choose $p(\vec{x})$ as generated by each alternative learning model that we consider (i.e., Naïve Bayes, Dalvi et al., and Bruckner and Scheffer). Figure 4.5 shows the results comparing the direct use of the three classifiers, and as a part of our optimization program, when $B = c|\mathcal{X}|$ with $c = 0.1$ and $c = 0.3$. This figure exhibits several findings. First, all three alternatives, including the two state-of-the-art approaches to adversarial classification, are exploitable by a sophisticated adversary. By comparison, all three of our optimization-based counterparts are more robust and beat their respective classifiers in paired comparisons. Second, the classifier of Dalvi et al. is in all cases far more robust to adversarial manipulation than the one derived from Bruckner and Scheffer. Finally, we did not display the

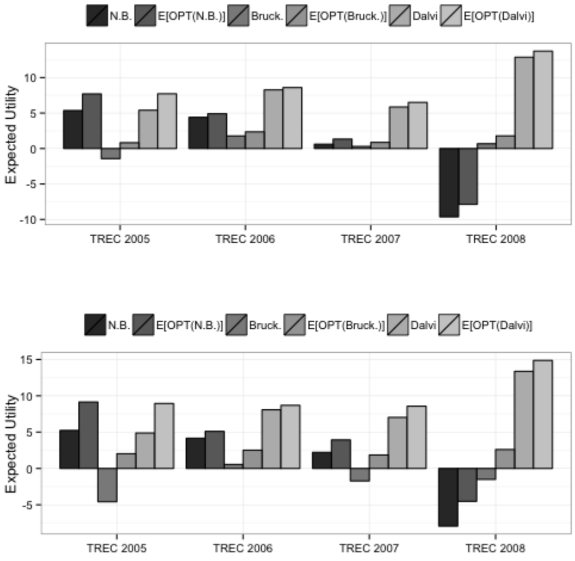


Figure 4.5.: The expected utilities, assuming $P_A = 1$ and that our attacker model is correct; top: $c = 0.1$; bottom: $c = 0.3$.

results of using Naïve Ranking here, as it performs far worse; clearly, randomization is crucial when facing a sophisticated adversary.

In another set of experiments, we use Naïve Bayes as $p(\vec{x})$, and evaluate the quality of Dalvi et al., Bruckner and Scheffer, and our optimized approach (still using a synthetic attacker). Figure 4.6 shows the results. As in the previous set of experiments, our model outperforms all of the alternatives. Surprisingly, however, Dalvi et al. and Bruckner and Scheffer do not improve much upon the baseline Naïve Bayes in this setting, and in some cases are even slightly worse.

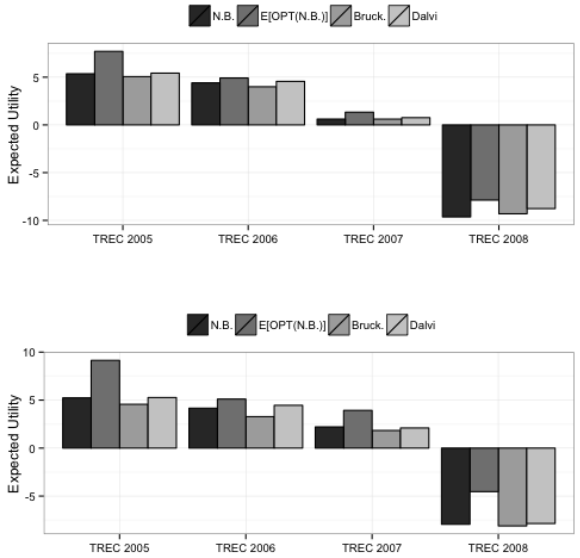


Figure 4.6.: The expected utilities, assuming $P_A = 1$ and that our attacker model is correct, where $p(x)$ is provided by Naïve Bayes; top: $c = 0.1$; bottom: $c = 0.3$.

In our final set of experiments in this section, we consider the impact of varying $V(x)$. The results are shown in Figure 4.7. Again, our model consistently outperforms alternatives in paired comparisons, at times by a considerable margin (up to 50% improvement). In all experiments in this section, we verified that our approach is statistically better than alternatives at the $\alpha = 0.05$ confidence level.

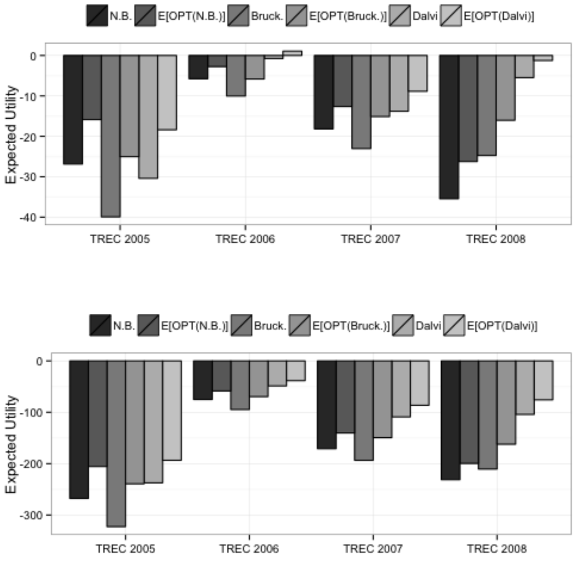


Figure 4.7.: The expected utilities, assuming $P_A = 1$ and that our attacker model is correct. Top: $V(x) = 2$; bottom: $V(x) = 10$. $c = 0.3$.

4.5.4 Robustness to Modeling Errors

A clear limitation of our evaluation in Section 4.5.3 is that the comparison which simulates attacker behavior according to our modeling assumptions unduly favors our approach. In this section, we relax this restriction in two ways: first, we introduce a significant error into the attacker model used in the LP that our approach solves, and evaluate by simulating attacker's response according to the "correct" model; and second, we solve the LP as before, but simulate the attacker's response according to an entirely different utility model. These experiments evaluate the sensitivity of our approach to parameter selection and model correctness. Below, we observe that our model is highly robust to both of these manipulations.

In our first set of robustness tests, we introduce an error $\eta \in [-1, 1]$ into the estimate of P_A and the parameter of the adversarial objective function δ. Specifically, given the true value of a parameter $\phi^T = \{P_A, \delta\}$, we add the error as $\phi = \phi^T + \eta$. We train our optimization problem using the erroneous parameter values, and evaluate the results by simulating attacker's response using the correct parameters.

Figure 4.8 displays the expected utility of the algorithms, using the error $\eta = 0.3$ which changes the true parameters $P_A^* = 1, \delta^* = 1$ to erroneous $P_A = 0.7, \delta = 0.7$ estimates. While our approach is certainly harmed by the inaccuracy in the parameter estimates, it is surprisingly robust to it, and we still outperform the state-of-the-art alternatives even in this rather unfavorable context. Next we consider the expected defender's utility when $p(x)$ is determined by Naïve Bayes, and $q(x)$ is determined by the classifier, and introduce the error of $\eta = 0.3$ into our estimates of P_A, δ as before. The results, shown in Figure 4.9, again demonstrate that our model is relatively robust to parametric errors, and still outperforms the competition.

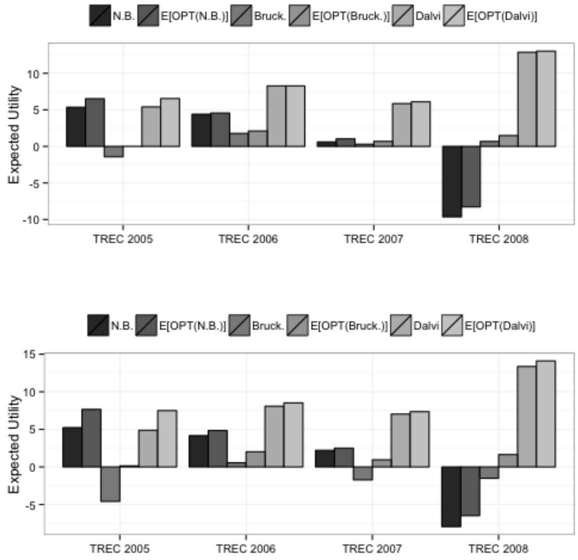


Figure 4.8.: The expected utilities, assuming $P_A = 1$ and that our attacker model is correct, but allowing for errors in parameter estimates; top: $c = 0.1$; bottom: $c = 0.3$.

In the next set of experiments, presented in Figure 4.10, we vary $V(x)$ to consider its effect on the defender's utility in the context of modeling errors ($\eta = 0.3$). Yet again, despite the errors, our model outperforms alternative approaches.

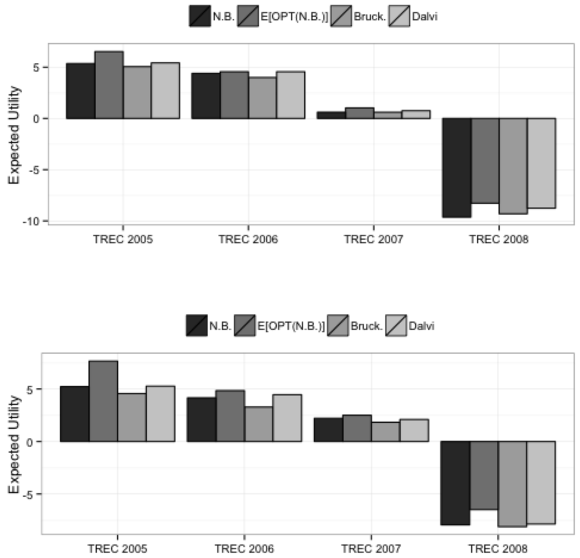


Figure 4.9.: The expected utilities, assuming $P_A = 1$, that our attacker model is correct, but allowing for errors in parameter estimates, and $p(x)$ is provided by Naïve Bayes; top: $c = 0.1$; bottom: $c = 0.3$.

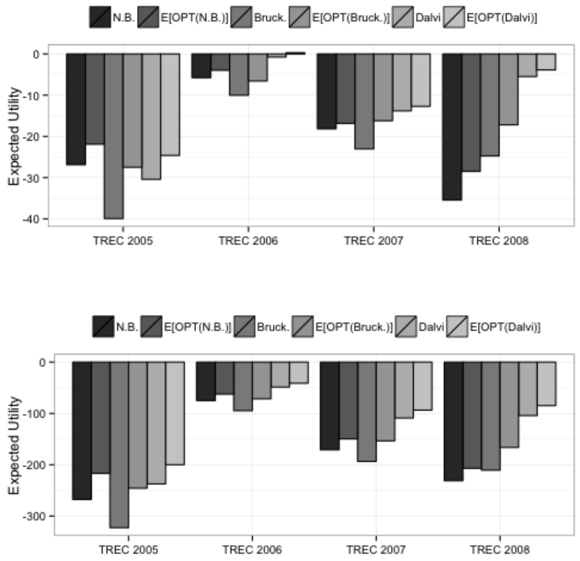


Figure 4.10.: The expected utilities, assuming $P_A = 1$ and that our attacker model is correct, but allowing for errors in parameter estimates. Top: $V(x) = 2$; bottom: $V(x) = 10$. $c = 0.3$.

In the final set of experiments in this section, we verify robustness of *our very model* of the adversary's utility. Arguably the most fundamental component of our model is exponential decay of the adversary's utility for using any but the most preferred input $\vec{x}$. To check robustness to this construction, we solve our model (the LP) as before, but evaluate the solutions $q(\vec{x})$ by simulating an adversary whose utility actually decays polynomially, i.e.,

$$Q_{poly}(\vec{x}, \vec{x'}) = \frac{1}{1 + \delta||\vec{x} - \vec{x'}||}. \tag{4.16}$$

The results are shown in Figure 4.11, and demonstrate that our model is quite robust even when the assumption about the attackers' utility functions is fundamentally incorrect, and handily outperforms the alternatives.

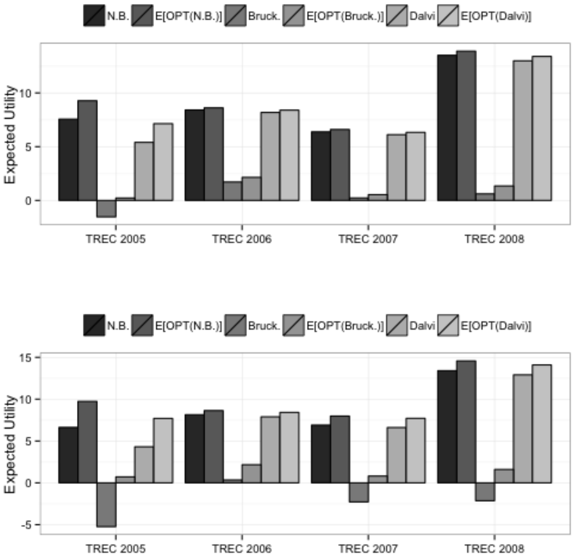


Figure 4.11.: The expected utilities, assuming $P_A = 1$ and that our attacker utility model is incorrect, and the actual utility decays for non-preferred input vectors according to Equation 4.16; top: $c = 0.1$; bottom: $c = 0.3$.

In all experiments in this section, we verified that our approach is statistically better than the alternatives at the $\alpha = 0.05$ confidence level.

4.6 Related Work

There have been a number of related methods that apply game theoretic reasoning in the context of network security in general, as well as specifically to harden a machine learning algorithm against adversarial manipulation [43–46, 52]. Dalvi et al. [52] consider an adaptive utility-maximizing adversary who is unaware of the data miner's strategy and reacts to the presence of a baseline (i.e., not hardened) classifier. The adversary's utility function is defined with respect to the minimum cost camouflage (MCC) of a given feature vector, which represents the least costly modification that allows a true positive feature vector to be classified as benign. This model of an adversary thus bears close similarity to ours. The defender therefore designs a classifier that is a best response to such attacks. The authors evaluate their algorithm by synthetically generating an adversarial response according to their model using public spam corpus as "original" data (i.e., prior to adversarial response). We perform a similar evaluation in a subset of our experiments. Bruckner and Scheffer [43, 45] consider building a classifier against *future* records generated by an adaptive adversary. Similar to our approach, they formalize this interaction first as a one-shot game [43], and in a later effort as a Stackelberg game in which the learner (defender) moves first, choosing a classification algorithm, and an adversary optimally responds to it by applying an optimal linear transformation to future data [45]. In both of these efforts, the goal of the defender is to minimize loss, and the attacker's goal is to maximize defender's loss, although the two players may weigh the loss differently (and, thus, the game need not be zero-sum). The evaluation in this work consists training on past spam records and testing on future records, just as we do in the first set of experiments we report. In a similar vein, Liu and Chawla [44] and Colbaugh and Glass [46] consider a constant-sum game between an adversary and a data miner, where the adversary aims to maximize loss on test data incurred by the learner by adding a fixed vector to every input, with a regularization term aimed to minimize the norm of the manipulation vector. The regularization term in the adversary's objective common to all of these captures, like our approach, that the adversary wishes to minimally manipulate the data to achieve his goal. Lowd and Meek [53] study the classifier manipulation problem in isolation, and using a model very similar to ours in which an adversary strives to choose a single input vector $\vec{x}$ that is misclassified by the learner (rather than, say, expected loss as in many other approaches described above), but with the objective of minimizing weighted l_1 distance to the *base* instance $\vec{x}^a$ (in our terminology, the attacker's preferred method of attack). In a more recent work, Huang et al. [42] give a taxonomy of attacks against machine learning algorithms. Biggio et al. [56] are among the first to consider adding randomness to a classifier in an effort to harden it against adversarial manipulation, but only propose adding a small amount of noise, rather than a computational framework for optimal randomization. Colbaugh and Glass [46] propose a uniform randomization scheme among several equally good classifiers, showing that this scheme is more robust to manipulation than a solution to a zero-sum game between the learner and manipulator which is similar in structure to the model of Liu and Chawla [44]. Using a somewhat different model, Kantarcioglu et al. [49] consider a Stackelberg game in which the adversary moves first, followed by the data miner. The authors give conditions under which an equilibrium exists in this game, and present stochastic search methods for approximating it. A number of related approaches exist that aim to make learning robust to specific data manipulations. For example, Globerson et al. [41] use quadratic programming to ensure robustness against feature deletion.

Game theory has been proposed for use in network security settings more broadly [57–61], but without considering specifically the adversarial aspects in machine learning. Most of this work considers variations on the problem of a defender trying to block attack paths, for example, by inspecting packets at a subset of nodes.

4.7 Conclusion

We have presented a general approach for finding the optimal inspection policy against both static and adaptive adversaries. We showed that in the special case when an adversary is static and with no operational budget constraints, our model is equivalent to traditional likelihood ratio approaches (equivalently, using a threshold on the probability of malware/spam). Our experiments demonstrated that our model consistently outperforms both a baseline, non-adversarial machine learning approach, as well as several state-of-the-art adversarial classification alternatives. Overall, our approach demonstrates a clear advantage when inspection is costly, events have weighted importance, and when there are sophisticated, adaptive attackers. From a practical perspective, our approach is very simple, highly scalable (it involves solving a linear program), and can use an arbitrary classifier as input (indeed, a better classifier would improve the performance of our optimization method). Our model is, of course, a severe simplification of reality, and in future work one could consider attackers that strategically manipulate training data, and/or multi-stage games in which defender and attackers move in sequence. Despite the apparent simplificity of our model, however, we demonstrate that it outperforms alternatives on *actual* data and, thus, is a good starting point for future, more complex, modeling advances, which would need to demonstrate sufficient added value to compensate for additional complexity.

5 APPLYING CRYPTOGRAPHY TO GAME THEORY

After demonstrating the utility of a game theoretic approach to adversarial problems in Chapter 4, we now demonstrate the utility of applying cryptography to solve a game theoretic problem. We consider the theoretical *Walrasian Auction Model* [1], where equilibrium is reached through a process called *tâtonnement*. The market presents a paradoxical scenario where trade cannot occur until equilibrium is reached, yet it is trade that determines excess demand. Secure multiparty computation is capable of acting as a fictitious mediator, handling pseudo-trades between buyers and sellers consistent with their private utility functions until equilibrium is reached. When the protocol is finished, the final equilibrium price is revealed to all players, thus avoiding the aforementioned conundrum.

After constructing a cryptographic protocol capable of realizing the Walrasian Auction Market in practice, we demonstrate that the protocol is designed to destabilize coalitions. As we will demonstrate through our multiparty framework in Chapter 7, applying game theory to cryptographic protocol design yields properties that cannot be achieved under the standard model. Destabilizing player coalitions is one such powerful property, which we demonstrate concretely in this chapter.

5.1 Introduction

Léon Walras' theory of general equilibrium put forth the notion of *tâtonnement* as a process by which equilibrium prices are determined [1]. Recently, Cole and Fleischer provided tâtonnement algorithms for both the classic One-Time and Ongoing Markets with guaranteed bounds for convergence to equilibrium prices. However, in order to reach equilibrium, trade must occur outside of equilibrium prices, which violates the underlying Walrasian Auction model. We propose a cryptographic solution to this game theoretic problem, and demonstrate that a secure multiparty computation protocol for the One-Time Market allows buyers and sellers to jointly compute equilibrium prices by *simulating* trade outside of equilibrium. This approach keeps the utility functions of all parties private, revealing only the final equilibrium price. Our approach has a real world application, as a similar market exists in the Tokyo Commodity Exchange where a trusted third party is employed. We prove that the protocol is inherently *incentive compatible*, such that no party has an incentive to use a dishonest utility function. We demonstrate security under the standard semi-honest model, as well as an extension to the stronger Accountable Computing framework.

5.2 Motivation

Open markets balance supply and demand by converging to a price where the two are equal. For example, oil is a commodity where increasing supply becomes progressively more expensive, and increasing price reduces demand. Absent other disturbing factors, oil supply and demand would eventually stabilize. However, this takes time, and in the meantime prices rise and fall, leading to unnecessary investment in uneconomical production based on an expectation of high prices, or investment in consumption based on expectation of low prices. Faster convergence or lower volatility in prices can have significant benefits.

Economic models generally accepted as valid representations of real-world market behavior tend to have underlying computationally tractable algorithms. It follows naturally to propose that these algorithms could be evaluated by parties to arrive at the result deemed to accurately reflect the outcome of a given market phenomenon. The work of Cole and Fleischer studies the market equilibrium problem from an algorithmic perspective, and they give tractable price update algorithms that do not rely on global information [62].

The algorithms of Cole and Fleischer [62] follow the Walrasian Auction model: prices are adjusted according to a tâtonnement process, where prices iteratively rise or fall in response to changes in demand [1]. In the Walrasian Auction model, *trade occurs only once equilibrium has been established.* In real-world markets, it is trade that dictates demand and, thus, how prices are adjusted to converge toward equilibrium. However, Cole and Fleischer's algorithms allow trade outside of equilibrium.

As specified, the Walrasian Auction model is limited to the theoretical domain unless a trusted third party is invoked to serve as a mediator between the buyers and sellers. Not only must the mediator be trusted to faithfully represent the interests of all parties involved, it must be trusted with substantial information about each party's private utility function. As a utility function defines a party's preferences over goods with respect to both quantity and price, it reveals valuable information that parties would prefer to keep private. Further, there are no guarantees that the parties will truthfully report their valuations of the good. This problem becomes particularly pronounced when independent buyers collude to reduce the final equilibrium price.

The recent work of Dodis et al. [16] considered a similar game theoretic problem: implementing the mediator for rational players to arrive at a *correlated equilibrium.* In game theory, a correlated equilibrium is selected when a mediator recommends a strategy to each player such that, given the recommended strategy, no player can improve their utility[1] by choosing a different strategy. Further, the payoff may exist outside the convex hull of standard Nash equilibria, yielding more utility than when a mediator is not present. Dodis et al. demonstrate that secure multiparty computation (SMPC) can replace the mediator with a protocol among the players, removing the necessity of a trusted third party. In this work, we use SMPC to find Walrasian equilibria *without* invoking a mediator or allowing trade to occur prior to arriving at a stable price.

Further, we are able to make strong claims of *incentive compatibility.* In the standard security model, a monolithic adversary $\mathcal{A}$ corrupts a subset of the participants. In rational cryptography, each player acts solely in their own self-interest, and thus have an associated *local* adversary controlling their deviations [63]. The move to lo-

[1] A utility function describes an agent's preferences over outcomes, and can informally be considered a mapping between events and agent happiness.

cal adversaries has important consequences on the stability of coalitions for rational players. Not even protocols secure in the malicious model cannot guarantee that a malicious party will not manipulate its input to the protocol, as a monolithic adversary may force the equilibrium price to be deflated through centralized control of corrupted parties. We demonstrate that our protocols are resilient against this behavior in the presence of local, independent rational adversaries seeking to maximize their utility.

5.3 Our Contribution

Drawing on recent work from both the cryptographic and game theoretic literature [12,15,26,30,34,64,65], we propose a privacy preserving protocol that allows buyers and sellers to arrive at an equilibrium price using the tâtonnement process *without* trade occurring outside of equilibrium. This approach has the auxiliary benefit of keeping the utility functions of all parties private; only the final equilibrium price is revealed. Further, we show that our construction is *incentive compatible*: the strategy of reporting truthful private valuations weakly dominates all other strategies for both buyers and seller.

A protocol that arrives at the equilibrium price for a good is beneficial to both the buyers and sellers involved. A participant's utility function must be evaluated many times throughout the tâtonnement process in order for appropriate price updates to occur. This is a potential disincentive to engaging in the protocol, as the participant's utility function contains their preferences for a good, and many individual points from their utility function are evaluated and publicly disclosed. A malicious agent could use this information to alter their behavior for personal gain. SMPC allows two or more mutually distrustful parties to engage in a collaborative protocol to compute the result of a function securely [3,4]. Our approach allows the tâtonnement process to be evaluated privately, revealing *only* the final equilibrium price.

SMPC has had real-world use, very much in the scenario we suggest. Bogetoft et al. [66] deploy a privacy preserving protocol to evaluate a double auction model for Danish commodity trading. However, they assume that all parties behave honestly in using the system, and do not explore the possibility that a malicious party could manipulate the equilibrium price to its advantage. In fact, they state *"we did not explicitly implement any security against cheating bidders"*, although they were only discussing semi-honest vs. malicious behavior in the traditional sense. Further, the authors surveyed the farmers' views on the privacy of their utility functions, and found that nearly all preferred that information to remain private.

We go well beyond this, exploring *lying about the input to the protocol itself*: a behavior that even the malicious model does not prevent. Previous work has demonstrated this idea, although the authors only consider a two-party protocol, and showed incentive compatibility only for an approximation of the real-world problem [67]. We show that this approach can be used to enable SMPC to address the full range of malicious behavior in a real-world, multiparty problem.

As another example, the Tokyo Commodity Exchange uses the *itayose* mechanism, similar to tâtonnement, to reach equilibrium. In fact, this existing market circumvents the restriction of disallowed trade until equilibrium is reached by invoking a trusted third party: an auctioneer that adjusts prices based on excess demand [68]. Our approach requires no trusted third party, resulting in the minimum possible disclosure of information regarding each party's utility function. Thus, there is clear real-world application and tangible benefit from our results, similar to those of Bogetoft et al. [66].

Note that our model makes a stronger statement than that of a Bayes-Nash equilibrium, where participants have an incentive to be truthful if and only if others are acting truthfully as well. We show that acting honestly is the dominant strategy in our protocol *regardless of the actions of the other players.* The work by Eaves et al. [68] provides further evidence for our claims of incentive compatibility, based on the fact that agents engage in the protocol repeatedly. However, our results hold without the assumption of repeated interaction.

To ensure parties deviating from the protocol will be caught, it is secure under the accountable computing (AC) framework proposed by Jiang and Clifton [69]. Note that we first show security under the standard semi-honest model, and then extend this to the AC-framework. The AC-framework provides the ability to verify that a party correctly followed the protocol; contractual penalties can then be used to ensure that correctly following the protocol is incentive compatible. Typical semi-honest protocols provide no such guarantee; a malicious party may be able to manipulate the protocol to their benefit. Protocols secure under the malicious model (forcing participants to correctly follow the protocol) typically have much greater computational cost. By demonstrating security under the AC-framework, detected deviations are punishable by other participants forcing the *minmax utility*[2] on the deviating parties [16]. We also use commitments to ensure that parties use their true utility function with the protocol; this prevents parties from supplying one input to the protocol (e.g., a low demand) to give an artificially beneficial price, then purchasing greater quantities at the resulting price.

We show that the utility functions and actions of all agents remain private, with the equilibrium price revealed to all agents at the conclusion of the protocol. The knowledge gain is only the information that can be derived from the result of the function, and knowledge of the function itself. This satisfies the standard definition of semi-honest security in that the protocol emulates the existence of a trusted third party, *without* actually requiring such an entity [5]. This property is ideal, as a universally trusted third party rarely exists for a given set of parties. Our work considers only the case of the *oblivious One-Time Market* setting. That is, we consider the market where all parameters are assumed *not* to be global information. Rather, agents compute the price updates based solely on local information.

[2] The *minmax* punishment approach forces the outcome yielding the minimum utility to the deviator, while maximizing the utility of the other participants.

We begin by defining the market problem and reviewing the oblivious One-Time Market algorithm in Section 5.4. We review the cryptographic primitives used in Section 5.5, and give a construction[3] based on an additively homomorphic cryptosystem in Section 5.6. Finally, we demonstrate that the resulting protocol is incentive compatible in Section 5.7. All proofs are provided in Section 5.9.

5.4 The Market Problem

Our SMPC protocol computes the equilibrium for a single seller offering a single good to a set of buyers, which we extend to the general definition of the problem following the notation from Cole and Fleischer [62]. The market under consideration contains a set of infinitely divisible goods G, where $|G| = n$, and a set of agents A, where $|A| = m$. Agent l has quantity w_{il} of good i at the start of the protocol and has a corresponding utility function $\mu_l(x_{1l}, \ldots, x_{nl})$ that gives their preferences for all goods $i \in G$. Note that the initial allocation w_{il} may consist solely of currency; it is a measure of the agent's wealth. We make the simplifying assumption that $\mu_l(x_{1l}, \ldots, x_{nl}) = \Sigma_{i=1}^{n} \mu(x_{il})$; the utility of a basket of goods is the sum of the utility of each individual good. Each good i has a collection of prices $p_i, 1 \leq i \leq n$. Each agent l selects a basket with x_{il} units of good i so that u_l is a maximum and is affordable given their initial allocation. That is: $\sum_{i=1}^{n} x_{il} p_i \leq \sum_{i=1}^{m} w_{il} p_i$. The prices $p = (p_1, p_2, \ldots, p_n)$ are in equilibrium if the demand for all goods $i \in G$ is bounded by the supply for good i: $\sum_{l=1}^{m} x_{il} \leq \sum_{l=1}^{m} w_{il}$.

We define $w_i = \sum_l w_{il}$ to be the supply of good i, and $x_i = \sum_l x_{il}$ to be the corresponding demand. We define $z_i = x_i - w_i$ to be the excess demand of good i. At a given set of prices p, the wealth of agent l is $v_l(p) = \sum_i w_{il} p_i$. By definition, w is from the market specification while v, x and z are computed with respect to the vector of prices. The wealth of an agent l is computed directly from a given price

[3] Our protocol can also be implemented using frameworks for the GMW protocol [4], such as FairPlayMP [10], VIFF [70] or SEPIA [71].

vector p, whereas x and z are computed by agents maximizing their utility functions under the constraints imposed by v.

The model put forth by Cole and Fleischer is based upon a series of iterative price and demand updates. We omit discussion of the proofs of bounded convergence time and refer the reader to their original work [62]. In each iteration r, the price of a good $i \in G^r$ is updated by its price setter using knowledge of only p_i, z_i, and their history. Here, a price setter is a virtual entity that governs the price adjustments. However, the price adjustments are governed by changes in demand in the algorithms. After the price setters have released the new prices p^r, the buying agents compute the set of goods that maximizes their utility under the constraint of their wealth given the current prices, $v_l(p)$. We consider only the oblivious One-Time Market price update rule, which is as follows:

$$p_i \leftarrow p_i \cdot (1 + \frac{1}{2^{\lceil \log_4 r_i \rceil}} \cdot min\{1, \frac{z_i}{w_i}\}) \tag{5.1}$$

The current round r is bounded prior to the start of the protocol by fixing the terminal round r^*. At the conclusion of the protocol, we will have computed the equilibrium price and demand, p^* and x^*, respectively.

To construct a privacy preserving protocol, we show how buyers compute their demand based on the current price p_i, and how sellers compute the price update given the demand x_i from the buyers. In our privacy preserving protocol, the buyers compute the update for each round locally to prevent the seller from learning intermediate prices. Symmetrically, neither the price nor the demand is known to either the buyers or seller until the conclusion of the protocol. Finally, we must account for the fact that $\frac{z_i}{w_i}$ may be less than 1, which cannot be represented properly in the field $\mathbb{Z}_n$. To handle this, prices are represented in integer units corresponding to the minimum increment (e.g., cents). We use the division protocol δ of Dahl et al. [72] to compute $\frac{z_i}{w_i}$, which we discuss further in Section 5.5.1. As the degree of Walrasian auction utility functions is 1 with overwhelming probability [1], all buyers are modeled as having Cobb-Doublas[4] utility functions. As noted by Cole and Fleischer, under these conditions the price update rule converges in a single round [62], so $r^* \leftarrow 1$.

Our work is certainly not the first to apply SMPC principles to economic and game theory. Previous work has shown that SMPC removes potential disincentives from bartering to auctions [73, 74]. Additionally, recent work has shown the potential of combining cryptography with game theoretic principles [12, 15, 26, 30, 34, 64, 65]. However, no attempt has been made to remedy the paradox of the Walrasian Auction model using SMPC techniques. In this way, we not only remove disincentives from engaging in the protocol, we allow the model to exist in reality. That is, our protocol allows the participants to evaluate the iterative price update function on the basis of the buyers' demand without actually revealing the demand through trade or invoking a trusted third party. Additionally, we show that our construction constitutes an incentive compatible market with respect to both buyers and sellers.

We review the One-Time Market Oblivious tâtonnement algorithm proposed by Cole and Fleischer [62]. The original algorithm is a protocol between a set of buyers $b_l \in B$ and a set of sellers $s_l \in S$. We assume that for each buyer $b_l \in B$ they have an associated utility function $\mu_{b_l}(i)$, where i is the good offered for sale from S. Recall that the seller S has knowledge of their supply of i, given by w_i. The task of the set of buyers B is to compute the excess demand for good i, given by $z_i = x_i - w_i$, where $x_i = \Sigma_l x_{il}$ is the sum of the demand of all buyers $b_l \in B$. The original protocol by Cole and Fleischer is given formally by Algorithm 1.

The algorithm fixes a price p_i for the good, uses the utility functions of the buyers to determine the excess demand x_i at that price, and sets the price for the next round. The key contribution of Cole and Fleischer is to prove that the given update rule gives a guaranteed convergence rate. Beyond simply bounding the number of required rounds, as Walrasian markets typically have Cobb-Douglas utility functions, the algorithm converges in one round [62].

[4] That is, a utility function whose parameter is an exponential function of the quantity of a good received.

Algorithm 1 Model by Cole and Fleischer

```
for $r_i = 0; r_i < r; ++n_i$ do
    for $s_l \in S$ do
        $p_i \leftarrow p_i + \frac{1}{2^{\lceil \log_4 r_i \rceil}} p_i \cdot min\{1, \frac{z_i}{w_i}\}$
    end for
    for all $b_l \in B$ do
        $x_i \leftarrow x_i + \mu_{b_l}(p_i)$
    end for
    $z_i \leftarrow x_i - w_i$
end for
$p^* = p_i$
$x^* = x_i$
return $(p^*, x^*)$
```

5.5 Building Blocks

To build the privacy preserving protocol, we build on a collection of cryptographic primitives.

We require an additively homomorphic public-key encryption scheme $\mathcal{E}$, with the additional property of semantic security [75]. Such a scheme was proposed by Paillier [11]. We denote the encryption of some plaintext x with Bob's public key as $E_b(x)$, and the decryption of some ciphertext $c = E_b(x)$ as $D_b(c)$. We require that our cryptosystem's *homomorphic property* is additive, which means that the following operations are supported:

$$E_b(x) \cdot E_b(y) = E_b(x+y), \qquad (E_b(x))^c \equiv E_b(x)^c = E_b(x \cdot c) \tag{5.2}$$

Here, c is an unencrypted plaintext constant. Note that we omit the enclosing parentheses and treat $E_b(x)$ as a distinct term. The construction of the additively homomorphic encryption scheme allows mathematical operations over encrypted data to be performed, and provides the foundation for our protocol.

5.5.1 Division Protocol δ

The price update rule requires computing the quotient of the excess demand and the supply, $\frac{x_i - w_i}{w_i}$. Dahl et al. give a protocol for securely computing integer division under the Paillier cryptosystem *without* requiring a bit-decomposition [72]. For l-bit values, the constant round protocol requires $O(l)$ arithmetic operations in $O(1)$ rounds.

5.6 Protocol Construction

We consider a set of k buyers $b_l \in B$ interacting with a single seller S of a good i. The protocol π securely implements the functionality $f(\mu_1, \cdots, \mu_k, p_S) \mapsto \langle p^*, x^* \rangle$. Here, μ_l is the utility function of buyer $b_l \in B$. The full Walrasian Market (composed of more than a single seller and good) is modeled by instantiating an instance of Protocol 5.6.1 for each pair of seller and good (S, i), and the associated set of buyers. Note that our protocol centers around specific utility functions known as Marshallian or Walrasian demand functions. That is, the participant's utility function is modeled as a polynomial, and defines the quantity demanded for a single good over all possible prices. Overwhelmingly, the degree of a Walrasian demand function will be one [1]. Thus, a buyer's utility function μ_{b_i} has the form $\mu_{b_i}(p_i) = cp_i$ where the coefficient c is a constant, satisfying the definition of a Cobb-Douglas utility function. The final argument to the functionality is the initial price p_i specified by the seller. A Paillier-based algorithm for computing the Walrasian equilibrium is given by Protocol 5.6.1. This protocol could also be implemented using a state-of-the-art framework for the GMW protocol [4], such as FairPlayMP [10], VIFF [70] or SEPIA [71].We defer the proof of security to Section 5.9.

Buyers $1 \leq l \leq k$: All buyers issue commitments (e.g., Pedersen [76]) to their private utility function coefficients. This is necessary for the verification stage of the AC-Framework [69].

Seller S: Set p_i as the Seller's initial price for good i.
Set w_i as the supply of good i.
Send $E_S(p_i)$ to all buyers.

Buyer 1 : The first buyer computes the initial demand as $E_S(x_i) \leftarrow \mu_{b_1}(E_S(p_i))$[§], where μ_{b_1} is the initial buyer's utility function.
The first buyer forwards $E_S(x_i)$ to the next buyer, so that they can update the demand x_i based on their utility function.

Buyers $1 < l \leq k$: Each buyer updates the demand at the current price p_i based on their utility function μ_{b_l} by computing $E_S(x_i) \leftarrow \mu_{b_l}(E_S(p_i))$[§].

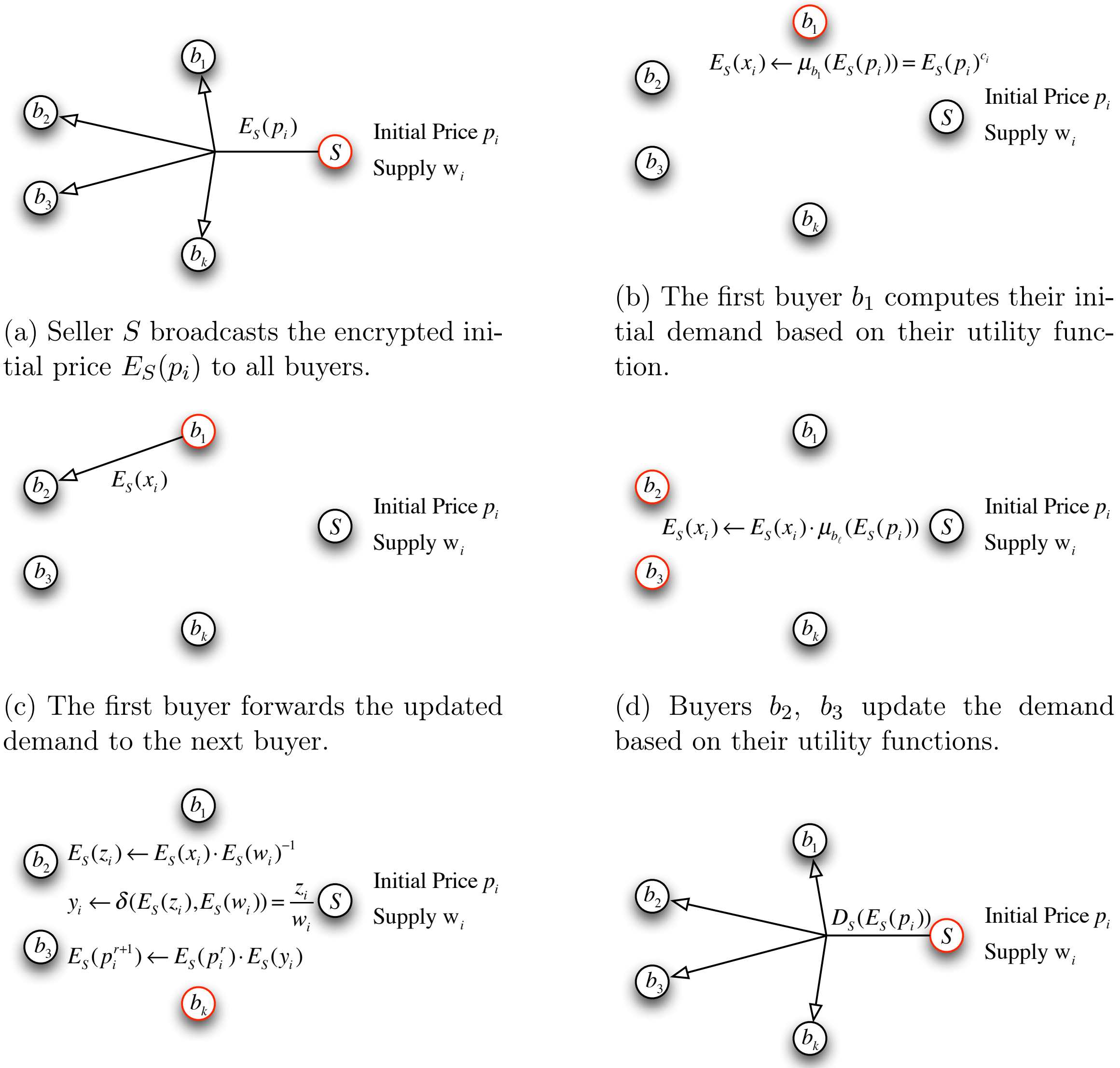
Buyer k: The final buyer b_k must perform additional updates before sending the results of the current round to either buyer 1 (if $r < r^*$) or the seller (if the terminal round r^* has been reached).
The final buyer updates the excess demand z_i by computing $E_S(z_i) \leftarrow E_S(x_i) \cdot E_S(w_i)^{-1}$.
The final buyer computes the price update coefficient $y_i := \frac{z_i}{w_i}$, the fraction of excess demand to supply, using the division protocol of Dahl et al. [72]: $y_i \leftarrow \delta(E_S(z_i), E_S(w_i))$.
The final buyer updates the current round price p_i^r to p_i^{r+1} by computing $E_S(p_i^{r+1}) \leftarrow E_S(p_i^r) \cdot E_S(y_i)$.
If $r = r^*$, where r^* is the final round, buyer b_k sends $\langle E_S(p_i), E_S(x_i) \rangle$ to the seller. Otherwise, this tuple is forwarded to buyer 1 and the next round begins.

Seller S: After receiving $\langle E_S(p_i), E_S(x_i) \rangle$ in the final round, the seller computes the equilibrium price $p^* \leftarrow D_S(E_S(p_i))$ and the final demand $x^* \leftarrow D_S(E_S(x_i))$.
The seller forwards p^* to all of the buyers.

Protocol 5.6.1: Additively Homomorphic Encryption Algorithm for Tâtonnement

In the next section, we prove that if a player is unable to deviate from the protocol without being caught (e.g., a protocol secure in the AC-Framework), then the dominant strategy is for parties to provide their true utility functions.

[§] Here, we evaluate $\mu_{b_l}(E_S(p_i))$ as $E_S(p_i) \cdot E_S(c)$, where c is the buyer's coefficient term in μ_{b_l}.


b_1
b_2
b_3
b_k
$E_S(p_i)$
S
Initial Price p_i
Supply w$_i$
$E_S(x_i) \leftarrow \mu_{b_1}(E_S(p_i)) = E_S(p_i)^{c_i}$
$E_S(x_i)$
$E_S(x_i) \leftarrow E_S(x_i) \cdot \mu_{b_\ell}(E_S(p_i))$
$E_S(z_i) \leftarrow E_S(x_i) \cdot E_S(w_i)^{-1}$
$y_i \leftarrow \delta(E_S(z_i), E_S(w_i)) = \frac{z_i}{w_i}$
$E_S(p_i^{r+1}) \leftarrow E_S(p_i^r) \cdot E_S(y_i)$
$D_S(E_S(p_i))$

(a) Seller S broadcasts the encrypted initial price $E_S(p_i)$ to all buyers.

(b) The first buyer b_1 computes their initial demand based on their utility function.

(c) The first buyer forwards the updated demand to the next buyer.

(d) Buyers b_2, b_3 update the demand based on their utility functions.

(e) The final buyer b_k uses the demand to calculate the final price, which is forwarded to the seller for decryption.

(f) The seller S distributes the final price to all buyers.

Figure 5.1.: Illustrated Homomorphic Tâtonnement Protocol

5.7 Incentive Compatibility

We claim that Protocol 5.6.1 is inherently *incentive compatible* with respect to protocol inputs from the perspectives of both buyers and sellers. That is, each player has no incentive to maliciously modify their actual input (utility function). We assume that malicious buyers have the option to either inflate or deflate their demand for a given price relative to their actual utility function. We show that while this can influence the price, it works to their detriment. We demonstrate that a seller only sets the initial price, and that their choice does not affect the final equilibrium price, so deviating provides no utility gain.

5.7.1 Utility Function Assumptions

In order to simplify the game theoretic analysis of the protocol, we write μ^+ to denote positive utility, μ^- to denote negative utility, and μ^0 to denote neutral utility gain. We assume that the magnitude of preference for all μ_i are equal (i.e., $\mu^+ + \mu^- = \mu^0$). Similarly, we assume that μ^ϵ represents only a marginal utility gain. That is, $\mu^+ > \mu^\epsilon > \mu^0$.

Additionally, we assume that $(p_i - p_i^*) \in \{\mu^+, \mu^-, \mu^\epsilon\}$, although this value depends on how much the reported utility function μ_l^* differs from an agent b_l's actual utility function μ_l. Clearly there is an inverse relationship between how much an agent can under-inflate μ_l^* (which subsequently reduces the equilibrium price p_i^*), and the likelihood of a trade occurring between the agent and the seller. As the agent is involved in the protocol, we assume that they prefer a trade occur. If not, they would have abstained from the protocol entirely. Thus, it is natural to assume the agent's utility function assigns the same range to both of these preferences. This assumption does not affect our analysis, and is solely to ease the exposition.

Definition 5.7.1 *Let r_l be the* ***reward*** *that a buyer b_l gains by reporting μ_l^* in lieu of their actual utility function μ_l. Where p_i^* (resp. p_i) is the resulting equilibrium price when μ_l^* (resp. μ_l) is reported, b_l's reward is given by:*

$$r_l = \begin{cases} (p_i - p_i^*) < 0 & : \mu_l^* > \mu_l \\ 0 & : \mu_l^* = \mu_l \\ (p_i - p_i^*) > 0 & : \mu_l^* < \mu_l \end{cases} \tag{5.3}$$

We make the natural assumption that each buyer prefers some (possibly large) quantity of the seller's good to their initial allocation, otherwise they would not engage in the protocol.

Definition 5.7.2 *Define the utility gained through trade as μ_τ:*

$$\mu_\tau = \begin{cases} \mu_\tau^+ & : \textit{trade occurs} \\ \mu_\tau^- & : \textit{trade does not occur} \end{cases} \tag{5.4}$$

Similarly, a buyer offering a higher price has increased control over the *quantity* of the good they can demand, subject to the seller's supply w_i. That is, the seller prefers to sell to the set of buyers $\{b_l | p_i^l \geq p_i^m, l \neq m\}$ offering the highest price. Thus, a highest price buyer b_m can command $min(w_i, w_m)$ units of good i, where w_i is the seller's supply and w_m is the initial allocation of resources for buyer b_m.

Definition 5.7.3 *Define buyer b_l's utility gained from control over quantity received, $\mu_{q,l}$, as follows:*

$$\mu_{q,l} = \begin{cases} \mu_{q,l}^+ & : \forall j, p_i > p_j, j \neq i \\ \mu_{q,l}^- & : \forall j, p_i \leq p_j, j \neq i \end{cases} \tag{5.5}$$

That is, b_l receives μ_q^+ if b_l is offering the highest price p_i, and μ_q^- otherwise.

Definition 5.7.4 *Let r_l be the reward for buyer b_l, let $\mu_{\tau,l}$ be b_l's trade utility, and let $\mu_{q,l}$ be b_l's quantity control utility. We define b_l's* ***total reward*** *ρ_l as follows:*

$$\rho_l = r_l + \mu_{\tau,l} + \mu_{q,l} \tag{5.6}$$

Without loss of generality, consider a coalition of buyers with utility functions satisfying the above constraints. Let $a_l = \{a_u, a_t, a_o\}$ denote b_l's action set, where a_u denotes under-inflating, a_o denotes over-inflating, and a_t denotes reporting the buyer's true utility function u_l rather than a modified utility function u_l^*.

We assume that a rational seller will agree to sell their entire allocation of goods to the buyer whose utility function u_b gives the highest valuation for the good, thus maximizing their profit. Thus, for all buyers $b_k \notin \{b_l | p_i^l \geq p_i^m, l \neq m\}$, we have that $\mu_{\tau,k} = \mu_{q,k} = \mu^-$. Note the following:

- A buyer playing a_u in the presence of a buyer playing $\{a_t, a_o\}$ does not have quantity control
- A buyer playing a_u in the presence of a buyer playing $\{a_t, a_o\}$ does not receive any goods
- A unique buyer playing $\{a_t, a_o\}$ in the presence of buyers playing only a_u has quantity control

We begin by reviewing the formal definition for *weakly dominated* strategies as given by Katz [12], where a player can never increase their utility by playing a weakly dominated strategy.

Definition 5.7.5 *Given a game $\Gamma = (\{A_l\}_{l=1}^k, \{u_l\}_{l=1}^k)$, where $A = A_1 \times \cdots \times A_k$ is a set of actions, with $a = (a_1, \ldots, a_k) \in A$ being a strategy and $\{\mu_l\}$ is a set of utility functions, we say that action $a_l' \in A_l$ is* ***weakly dominated*** *by $a_l \in A_l$ if*

$\mu_l(a_l) \geq \mu_l(a_l')$. That is, player P_l never improves their payoff by playing a_l', but can sometimes improve their payoff by playing a_l.

To show that our construction is *incentive compatible*, we iteratively delete weakly dominated strategies to arrive at the stable Nash equilibrium [14]. The process of iteratively deleting weakly dominated strategies is criticized because, in some cases, the *order* of deletion affects the final result [26]. In this analysis, weakly dominated strategies can be removed in an arbitrary order without affecting the result.

We present a simplified payoff matrix in Table 5.1. The strategy a_o of over-inflating the utility function is removed for clarity, as a_u, the strategy of under-inflating, is a much more intuitive deviation for maximizing utility. However, we formally demonstrate that a_o is weakly dominated in lemma 5.7.1.

Table 5.1: Total Payoff Matrix

	a_u'	a_t'
a_u	(μ^+,μ^+)	$(\mu^-,2\mu^+)$
a_t	$(2\mu^+,\mu^-)$	(μ^+,μ^+)

Lemma 5.7.1 *The strategy a_o of reporting an over-inflated utility function u_i^* is weakly dominated by a_t.*

Proof We show that the action of over-inflating the buyer's true utility function is weakly dominated by truthfully reporting the utility function, demonstrating that a_o is weakly dominated by a_t. Recall that buyer b_l's total reward is defined as $\rho_l = r_l + \mu_{\tau,l} + \mu_{q,l}$. For convenience, we will parameterize $\rho_l(\cdot)$ with the action being played. This notation is convenient for comparing the total payoff yielded from different actions.

We begin by deriving the maximum utility that could be gained by playing a_o, the action of over-inflating the true utility function. As buyer b_l is playing a_o, we have that $\mu_l^* > \mu_l$. From Equation 5.3, we have $\rho_l(a_o) = (p_i - p_i^*) + \mu_{\tau,l} + \mu_{q,l}$. As $(p_i - p_i^*) < 0$, we write μ^- for concreteness. Given that b_l is over-inflating their true utility function μ_l, they are more likely to effect a trade. Clearly the seller S prefers the higher price p_i^* to b_l's true valuation, p_i. By Equation 5.7.2, we have that $\rho_l(a_o) = \mu^- + \mu_{\tau,l}^+ + \mu_{q,l}$. Similarly, by over-inflating their true utility function, b_l is more likely to have control over the quantity of the good they receive, as they are offering a higher price. By Equation 5.7.3, we have that: $\rho_l(a_o) = \mu^- + \mu_{\tau,l}^+ + \mu_{q,l}^+ = \mu^+$. Thus, we have that $max(\mu_l(a_o)) = \mu^+$. We now derive the maximum utility that could be gained by playing a_t, where buyer b_l reports the true utility function μ_l. By Equation 5.3, we have that $\rho_l(a_t) = \mu^0 + \mu_{\tau,l} + \mu_{q,l}$ as $p_i = p_i^*$ so $(p_i - p_i^*) = \mu^0$. Buyer b_l maximizes their utility when a trade occurs, and they can control the quantity of the good they receive. Following the same derivation that was used for a_o, we have from Equation 5.7.2 that $\rho_l(a_t) = \mu^0 + \mu_{\tau,l}^+ + \mu_{q,l}$. Similarly, by Equation 5.7.3 we have that $\rho_l(a_t) = \mu^0 + \mu_{\tau,l}^+ + \mu_{q,l}^+ = 2\mu^+$. We have that $max(\mu_l(a_t)) = 2\mu^+$, and it follows that $max(\mu_l(a_t)) > max(\mu_l(a_o))$. Thus, a buyer always does *at least as well or better* by playing a_t, and we say that a_t weakly dominates strategy a_o. ■

Lemma 5.7.2 *The strategy a_u of reporting an under-inflated utility function u_l^* is weakly dominated by a_t.*

Proof We demonstrate that the action a_u is weakly dominated by a_t when considering both individual buyers and members of a buyer coalition that collude to lower the equilibrium price p^*.

Consider an individual buyer b_l that is not a member of a coalition. As b_l reports $\mu_l^*, \mu_l^* < \mu_l$, by Equation 5.3 we have that $\rho_l(a_u) = (p_i - p_i^*) + \mu_{\tau,l} + \mu_{q,l}$. Again, as $(p_i - p_i^*) > 0$, we assume $(p_i - p_i^*) = \mu^+$ for concreteness. Similarly, we assume that under-inflating μ_l reduces the chances of b_l effecting a trade with S, as b_l is offering a lower price. By Equation 5.7.2, we have that $\rho_l(a_u) = \mu^+ \mu_{\tau,l}^- + \mu_{q,l}$. Playing action a_u

also reduces the chances of b_l having control over the quantity of the good received, if any is received at all. By Equation 5.7.3, we have that $\rho_l(a_u) = \mu^+\mu^-_{\tau,l} + \mu^-_{q,l} = \mu^-$. Thus, $max(\mu_l(a_u)) = \mu^-$, and it follows that $max(\mu_l(a_t)) > max(\mu_l(a_u))$. Thus, a (non-coalition) buyer always does *at least as well or better* by playing a_t, and we say that a_t weakly dominates strategy a_u.

We now consider a coalition of *unique* buyers under-reporting μ_l as $\mu^*_l < \mu_l$, colluding to decrease the resulting equilibrium price p^* of the good. That is, the coalition is *not* controlled by a monolithic adversary as is common in the standard security model: they are independent buyers in competition, modeled under the *local adversary* framework of Canetti [63]. In the game theoretic literature, this is referred to as the cartel problem. Note that the best response of any member of the coalition is to report $\mu^*_l + \epsilon$ for any positive ϵ. In doing so, they receive the goods at a price $p' < p^*$ while the other coalition members receive no goods. Applying backward induction, we demonstrate that the best response of all buyers in a coalition is to report μ_l, as $\mu^*_l + \epsilon$ converges to their true utility function μ_l.

Suppose all coalition members agree to collude by reporting $\mu^*_l < \mu_l$, and all members play this strategy. For any buyer b_l in the coalition, we have that $\mu^*_l < \mu_l$ and by Equation 5.3 we have that $\rho_l(a_u) = (p_i - p^*_i) + \mu_{\tau,l} + \mu_{q,l}$. As $(p_i - p^*_i) > 0$, we set $(p_i - p^*_i) = \mu^+$ to denote a positive utility gain. As the coalition consists of more than a single buyer, all members of the coalition are more likely to effect a trade. From Equation 5.7.2, we have that $\rho_l(a_u) = \mu^+ + \mu^+_{\tau,l} + \mu_{q,l}$. However, as all members of the coalition are offering the same price for the good, they have no control over the quantity of the good they receive. By Equation 5.7.3, we have that $\rho_l(a_u) = \mu^+ + \mu^+_{\tau,l} + \mu^-_{q,l} = \mu^+$. Thus, $max(\mu_l(a_u)) = \mu^+$ for all coalition members. However, consider the case where a coalition member reports a utility function $\mu'_l = \mu^*_l + \epsilon, \epsilon > 0$. That is, some b_l in the coalition increases the price they are willing to pay for the good by any positive amount ϵ. From Equation 5.3, we have that

$$\rho_l(a_u + \epsilon) = ((p_i - (p_i^* + \epsilon)) + \mu_{\tau,l} + \mu_{q,l} = \mu^{(+)-\epsilon} + \mu_{\tau,l} + \mu_{q,l} \tag{5.7}$$

However, now b_l is more likely to effect a trade, as $p_i^* + \epsilon > p_i^*$. By Equation 5.7.2, we have that $\rho_l(a_u+\epsilon) = \mu^{(+)-\epsilon} + \mu^+_{\tau,l} + \mu_{q,l}$. Similarly, b_l has control over the quantity of the good received as b_l is offering ϵ more than the coalition members. From Equation 5.7.3, we have

$$\rho_l(a_u + \epsilon) = \mu^{(+)-\epsilon} + \mu^+_{\tau,l} + \mu^+_{q,l} > 2\mu^+ > max(\mu_l(a_u)) \tag{5.8}$$

Thus, $max(\mu_l(a_u + \epsilon)) > max(\mu_l(a_u))$, as $\mu^{(+)-\epsilon} = \mu^+ + \mu^{-\epsilon} > \mu^0$. However, all coalition members are aware of this fact. Applying backward induction, it is not difficult to see that action a_u converges to a_t by increasing ϵ until $\mu_l^* = \mu_l$, and that a_t weakly dominates a_u. ■

Corollary 5.7.3 *The strategy a_t of reporting the true utility function u_l weakly dominates $\{a_u, a_o\}$ for all buyers.*

Proof A buyer's action set is defined as $a_l \in \{a_u, a_t, a_o\}$. By lemma 5.7.1, we have that a_o is a weakly dominated strategy, and can be eliminated. By lemma 5.7.2, we have that a_u is a weakly dominated strategy, and can be eliminated. Thus, reporting the true utility function μ_l as denoted by action a_t is a stable Nash equilibrium. ■

Theorem 5.7.4 *The strategy a_t of reporting the true utility function u_l weakly dominates $\{a_u, a_o\}$ for the seller.*

Proof As noted in the original paper, the update protocol converges on the equilibrium price p^* from any *arbitrary* initial price p_i [62]. Given that the seller's only influence on the equilibrium price is through setting the initial price p_i, there is no incentive to report some $p_i' \neq p_i$, as p^* is unaffected in doing so. ■

5.8 Semi-Honest Proof of Security

The proof of security under the semi-honest model is the first step in demonstrating security under the AC-Framework. We formally prove that Algorithm 5.6.1 is secure under the standard semi-honest model in Section 5.9, when demonstrating that the **Basic Security** condition holds for the AC-Framework.

5.9 Security under the AC-Framework

The Accountable Computing (AC) -framework [69] considers adversaries in the gap between the semi-honest and malicious models. The AC-framework guarantees that an honest party *can* catch malicious behavior (unlike Aumann's covert model, which requires that such behavior be caught); honest parties can choose not to verify that behavior is correct (thus saving computation), verify if they do not trust the results, or probabilistically verify sufficiently often to ensure incentives for correct behavior. We now show that our protocol satisfies the conditions necessary under the AC-framework. As part of this, we formally prove that the protocol is secure under the semi-honest model (Theorem 5.9.1), as security under the standard semi-honest model is a requirement for satisfying security under the AC-Framework.

The definition as given by Jiang and Clifton [69] is as follows:

Definition 5.9.1 *(AC-protocol) An AC-protocol Φ must satisfy the following three requirements:*

1. ***Basic Security:*** *Without consideration of the verification process, Φ satisfies the security requirements of a SSMC-protocol (a SMC-protocol secure under the semi-honest model).*

2. ***Basic Structure:*** *The execution of Φ consists of two phases:*

   - ***Computation phase:*** *Compute the prescribed functionality and store information needed for the verification process.*

   - ***Verification phase:*** *An honest party (we name such a party as a prover hereafter) can succeed in verifying an accountable behavior.*

3. ***Sound Verification:*** *Φ is sound providing that the verification phase cannot be fabricated by a malicious party.*

We now demonstrate that Φ satisfies all requirements of the AC-framework.

Theorem 5.9.1 ***Basic Security*** *Given an adversary $\mathcal{A}$'s private inputs $I_{\mathcal{A}}$ and output $O_{\mathcal{A}}$, $\mathcal{A}$'s view of the protocol can be efficiently simulated.*

Proof We follow the simulation proof of semi-honest security characterized by Goldreich [5]. Consider the case where $\mathcal{A}$ is a buyer. With the exception of $\mathcal{A}$'s private input and the result of Φ, all messages are encrypted with the seller's public key of an additively homomorphic encryption scheme $\mathcal{E}$. It follows naturally that a simulator could generate and send a series of random elements in $\mathbb{Z}^*_{n^2}$ to $\mathcal{A}$. The encryption scheme $\mathcal{E}$ is semantically secure, which implies that $\mathcal{A}$ is unable to distinguish the random elements of $\mathbb{Z}^*_{n^2}$ from true encryptions. Thus, $\mathcal{A}$'s view of Φ is efficiently simulatable. Consider next the case where $\mathcal{A}$ is the seller. $\mathcal{A}$ sees only the final message $E_S(p_i)$, which is the output of the protocol. Thus, $O_{\mathcal{A}} = E_S(p_i)$ can be efficiently simulated by encrypting the final result p_i with the seller's public key (known to the seller/simulator) to get $E_S(p_i)$. Thus, Φ does not reveal any additional information to $\mathcal{A}$ through the intermediary messages. ■

Lemma 5.9.2 *(Basic Structure: Computation) Φ stores sufficient information to support the verification phase.*

Proof In the case of the seller S, the initial price $p_{initial}$ as well as all internal coin tosses used for encryption are stored. In the case of a buyer, the committed (e.g., Pedersen's scheme [76]) coefficients, all encrypted price updates, as well as all internal coin tosses are stored. ■

Lemma 5.9.3 *(Basic Structure: Verification) An honest party in Φ can succeed in verifying an accountable behavior while revealing only that information in β.*

Proof Let T_Φ represent the entire protocol transcript. Consider the case where an honest buyer b_l wishes to demonstrate accountable behavior. In this case, all intermediate prices p_i are revealed. A verifier uses the internal coin tosses of b_l to reconstruct $E_S(\mu_{b_l}(p_i))$. For each committed coefficient c_l, we reconstruct $E_S(\mu b_l(p_i)) \in T_\Phi$ by computing $\Pi_{j=1}^{t} E_S(c_l)^{p_i}$ using the internal coin tosses of b_l. The encryptions of $E_S(\mu_{b_l}(p_i))$ will have *identical representations* in $\mathbb{Z}^*_{n^2}$, as they were generated with the same randomness. Thus, the encrypted elements can be compared bitwise for equality. If the price updates of $b_l \in T_\Phi$ match the reconstructed values, b_l demonstrates accountable behavior. Consider the case of the seller S. A seller needs to demonstrate that the final decrypted price $p^r = D_S(E_S(p^r))$ in the final round is equal to the *reported* final price p^*_r. Any verifier can compute a seller verification value $V_S = E_S(R_2 \cdot (R_1 - p_r)) = (E_S(p_r) \cdot E_S(-R_1))^{R_2}$, where R_1, R_2 are chosen uniformly at random from $\mathbb{Z}_n$, and ask S to decrypt the value. If $R_2 \cdot (R_1 - p_r) = R_2 \cdot (R_1 - p^*_r)$, the seller demonstrates accountable behavior. Each buyer signs $E_S(p_r)$ to prevent a dishonest buyer from recanting in order to falsely implicate an honest seller. ■

Theorem 5.9.4 Φ *satisfies the sound verification phase.*

Proof Consider the case of a malicious buyer b_m. If any of b_m's price updates were not computed using the committed coefficients of b_m's utility function, the reconstructed encrypted update will not match the update in T_Φ. Further, there does not exist a series of coin tosses that allow b_m to represent an altered update $E_S(\mu^*_{b_m}(p_i))$ as the actual update $E_S(\mu_{b_m}(p_i)) \in T_\Phi$, as this would prevent deterministic decryption. Thus, no malicious buyer b_m can forge a legitimate verification. In the case of a malicious seller S_m, the blinded value of p_r prevents S_m from constructing a response $V'_S \neq V_S$ such that some p^*_r can be reported in lieu of the actual equilibrium price p_r.
■

Theorem 5.9.5 ***Basic Structure (buyer)*** *Let Φ represent Protocol 5.6.1 for the Walrasian Auction problem. Assuming an honest majority, an honest buyer can be verified by any honest party (including an independent verifier) other than the seller.*

Proof The verifier is provided with the commitment of coefficients by all buyers (with the majority agreeing). The buyer b_l being verified provides their input and output values of each round; the following buyer b_{l+1} also provides their input for each round. b_l also provides the random value used in encryption during each round. The verifier can then duplicate the calculations of b_l, ensuring that the output of each round is consistent with the committed coefficients. If not, b_l is dishonest.

If the output reported by b_l does not match the input reported by b_{l+1}, then either b_l is dishonest, or b_{l+1} is reporting an incorrect value to the verifier. In the latter case, b_{l+1} can be required to verify, if it succeeds, then b_l is dishonest. ■

Theorem 5.9.6 ***Sound Verification (buyer)*** *A rational malicious buyer b_l cannot fabricate verification provided b_{l+1} is honest.*

Proof If b_{l+1} correctly reports the value received from b_l, then b_l must provide the same value to the verifier, and this must be the value generated from b_l's input. Generating this input from the output violates the assumption that the encryption is semantically secure. If b_l uses an incorrect input in the protocol (thus generating a matching output, but not following the protocol), the actual value and thus the impact on the outcome is completely unpredictable due to the security of the encryption, violating the assumption of a rational party. ■

Lemma 5.9.7 *Φ computes the equilibrium value of the Walrasian Auction model and stores sufficient information for verification to occur.*

Proof Note that given the set $V = \{E_S(p_{initial}), E_S(w_{initial})\}$ and the seller S's private decryption key D_S, the entire protocol can be executed by a participating-party. By revealing D_S, the seller only exposes the verification set V and no other private data. Given this, the participating-party can verify the correctness of the output of Φ by retrieving the demand $x_i - x_p$ from the remaining buyers through a trivial protocol (where x_p is the demand of the participating-party performing the verification). The participating-party is thus able to execute Φ to verify the correctness of the equilibrium price p^*. ■

Theorem 5.9.8 ***Accountability (seller)*** *A rational seller S will not behave dishonestly in Φ.*

Proof This follows from the proof of Theorem 5.7.4, as the seller's input has no effect on the final equilibrium price. ■

Given the previous two lemma's, we can conclude that Φ satisfies the *Basic Structure* condition.

Theorem 5.9.9 ***Sound Verification*** *The verification phase of Φ cannot be fabricated by a malicious party.*

Proof At the beginning of Φ, the seller S distributes the set V, such that $V = \{E_S(p_{initial}), E_S(w_{initial})\}$ to all buyers $b \in B$. It follows naturally that once this commitment is made, the seller is unable to alter the commitments. Should the seller provide an erroneous decryption key $D_S^* \neq D_S$, the commitments will decrypt to values $p_{initial}^* \neq p_{initial}$ and $w_{initial}^* \neq w_{initial}$ which defeats the seller's intention to fabricate the verification. Thus, we can conclude that the seller cannot succeed in fabricating the result of the verification process. ■

With this, we can conclude that our protocol is secure under the AC-framework, thus enabling malicious behaviour to be caught and contractual incentives put into place to ensure that semi-honest behavior is incentive compatible.

5.10 Conclusion

We have presented a privacy preserving, incentive compatible market construction that is secure against malicious parties, going beyond the standard security model to protect against malicious input to the protocol. To do this, we demonstrated that by securely computing the Oblivious One-Time Market protocol given by Cole and Fleischer [62], no agent has an incentive to report false valuations of the goods in the market. Thus, SMPC solves a long-standing problem in economic theory, as it allows Léon Walras' tâtonnement process for arriving at equilibrium to be computed while conforming to the constraints of the Walrasian Auction model. In this way, trade does not occur outside of equilibrium, and yet the final equilibrium price is computed and made available to all agents in the market.

6 APPLYING GAME THEORY TO CRYPTOGRAPHY

In Chapter 4, we demonstrated the utility of a game theoretic approach to adversarial settings. Chapter 5 demonstrated the utility of applying cryptographic primitives to classic game theoretic problems. In this chapter, we will merge cryptography and game theory to construct a two-party framework for reasoning about the security of cryptographic protocols. Our framework presents the standard notions of security through equivalent game theoretic concepts.

We build upon previous results to strengthen the equilibrium concept for rational two-party computation. Only rational players acting to maximize their utility functions are considered. Games are analyzed as extensive form dynamic games of imperfect information, using a computational variant of perfect Bayesian equilibrium as the solution concept. We argue that the perfect Bayesian equilibrium is a more appropriate solution concept than current solutions, as in cryptographic protocols information is often imperfect by design. Further, the perfect Bayesian equilibrium concept is able to address dynamic games, where players move sequentially rather than simultaneously. By considering players that move sequentially, we are able to remove the assumption of a broadcast channel. Finally, we give novel definitions of privacy, correctness and fairness solely in terms of game theoretic constructs.

6.1 Introduction

A recent focus of the cryptographic literature has been to formulate a framework for analyzing the security of protocols from a game theoretic perspective. The notion of rational multiparty computation considers only a single class of players: those that are rational, seeking to maximize their utility functions. A survey of the intersection of cryptography and game theory is given by Katz [12].

Most previous work towards a general game theoretic framework for reasoning about security in rational multiparty computation has been limited to those functions that are *non-cooperatively computable* (NCC), as defined by Shoham et al. [77]. In addition to being restricted to NCC, most existing work uses computational variants of Nash, Correlated or Bayesian equilibrium [15,16,23,39,65] as the solution concept for games. The exception is work by Gradwohl et al. [30], where the authors consider a relaxed version of computational sequential rationality that removes non-credible threats, called *threat-free Nash equilibrium*. However, all of these solution concepts consider only games of perfect information. We argue that the notion of perfect Bayesian equilibrium (PBE), a solution concept for extensive form dynamic games of imperfect information, is preferable for modeling cryptographic protocols. As players commonly cannot observe the moves made by others in cryptographic protocols, PBE offers a natural method for modeling this uncertainty. Further, it formally models observable actions and auxiliary information available to players that affects their strategy selection. As extensive form games may contain non-credible threats, we give a modified version of Gradwohl et al.'s [30] definition that is intuitive for games in the computational setting. Finally, we give novel definitions capturing the cryptographic concepts of privacy, correctness and fairness in terms of game theoretic constructs and prove the necessary and sufficient conditions under which they hold.

The goal of a rational multiparty computation framework is to relax the requirements of the malicious and semi-honest models in secure multiparty computation. The malicious model must protect against *all* deviations from the protocol specification, including actions that do not give an adversary an advantage. Protocols secure in the semi-honest model achieve greater efficiency, but suffer from the strong assumption that parties will not deviate from the protocol even if they benefit from doing so. As we describe in Section 6.6, our framework requires only that parties follow the protocol if such action constitutes *rational behavior*. We argue that the assumption of rationality is far weaker than the blind obedience required in the semi-honest model, and the resulting protocols will be more efficient than their malicious model counterparts that must prevent arbitrary (i.e., *non-rational*) actions. Perhaps most critically, even protocols secure under the malicious model do not prevent a party from lying about their input. Rational behavior provides a means to incorporate this into the discussion through *incentive compatibility*, ensuring that results reflect the true data.

First we review existing work applying game theory to cryptographic protocols. Section 6.3 discusses limitations with prior work, in particular showing that existing approaches do not fully model the rational secret sharing problem. We argue that modeling imperfect information, beliefs about the game state, and non-credible threats are desirable qualities of a candidate equilibrium concept for rational multiparty computation. From the game theory background of Section 3.1, we move to the computational setting in Section 6.4. A computational variant of the perfect Bayesian solution concept is defined in Section 6.5, and finally our new game theoretic framework for analyzing cryptographic protocols is presented in Section 6.6.

6.2 Related Work

The impetus for this work is largely due to a recent survey by Katz describing ongoing research into potential links between cryptographic and game theoretic notions [12]. We attempt to formulate our definitions in the same manner with the hope of consensus.

Many of the current frameworks for rational multiparty computation are limited in scope to functions that are *non-cooperatively computable* (NCC) [15, 16, 23, 39, 65]. Such a restriction is necessarily imposed under the assumption that parties desire *exclusivity* of the function output. That is, they prefer to learn[1] the *correct* result of the function, while preventing other parties from learning the result. This assumption is not necessarily valid for all games. In market scenarios, at least two parties must learn the result of the function in order to complete a trade. That is, an adversary needs the result of the function to be known to other participants in order to achieve their goal. Thus, the functions in *NCC* are a proper subset of those supported in this framework.

Halpern and Teague study rational multiparty computation under the assumptions of *correctness* and *exclusivity* [23]. They show the impossibility of secret sharing and general multiparty computation for any *deterministic* mechanism under these assumptions. However, they give randomized algorithms that terminate in expected constant time for both problems, and show that they satisfy their framework. We note that we remove the *exclusivity* requirement from our general framework, although this can be modeled through a natural extension. Further, we do not consider the notion of *iterated deletion of weakly dominated strategies*, as this equilibrium concept is sensitive to the order of strategy deletion [26]. Kol and Naor expand on the work of Halpern and Teague to give protocols that are not susceptible to *backward induction*, even in the presence of exponentially many iterations [26]. These solutions assume the existence of a broadcast channel, and they give solutions for both the non-simultaneous and simultaneous cases. The authors choose the notion of a *computational Nash equilibrium*, and leave extensions to subgame perfection open. Subgame perfection requires

[1] That is, each party prefers to have the ability to *derive* the correct result of the function. This may require additional computation after the function output is revealed to the party. This occurs when, for example, the party provides any input x' that differs from the party's true input x.

optimal play at each decision node in the game tree, and thus refines Nash equilibria in games of perfect information in the same manner that PBE refines Nash equilibria in games of imperfect information. We argue that even the extension to subgame perfection is inadequate, as it assumes players are aware of the moves made by others. The goal of cryptographic interactions is often to prevent learning others' information or actions. Instead, we consider extensive form dynamic games of imperfect information, where players' information sets are not guaranteed to be singleton nodes and players move sequentially rather than simultaneously. Nojoumian et al. [78] introduced *socio-rational secret sharing*, where rational and malicious players engage in the same protocol more than once. A public trust network is assumed, which stores a player's believed honesty based on past protocol interactions. We go beyond this model by modeling all players as rational, rather than creating a separate class of malicious players. Further, we do not assume the existence of a public trust network, nor do we assume that players necessarily value future interaction.

This work is not the first to attempt a game theoretic framework for constructing rational multiparty computation protocols. Our goal is to unify the existing frameworks by providing a stronger equilibrium solution concept well-suited for cryptographic protocols, while introducing notions of privacy, correctness, and fairness defined in game theoretic terms.

The first framework to consider players' motivations was given by Aumann et al. [79]. Although the framework does not explicitly consider rationality, the authors assume that adversaries are *covert*. That is, they are willing to cheat so long as they will not be caught doing so. Implicitly, an adversary is assumed to have a utility function defined solely with respect to their aversion to detection. The authors give three protocol constructions that provide security gradients between the standard semi-honest and malicious models of secure multiparty computation. The recent direction in the literature is towards a game theoretic framework for constructing protocols with *weaker* security guarantees than the standard multiparty computation frameworks. The most complete game theoretic framework to date was given by Halpern et al. [34]. They consider how agents play games when computation has an associated cost and affects agents' utility functions directly. The authors formalize the notion of a computational Nash equilibrium, and demonstrate that mixed computational Nash equilibria are guaranteed to exist for the set of computational games where randomization is free. However, the framework considers only Bayesian games of perfect information. Bayesian Nash equilibrium can result in implausible equilibria, as it does not exclude non-credible threats. In the setting of cryptography, threatening to break the underlying cryptosystem would constitute a non-credible threat for a player bound to probabilistic polynomial time (PPT), despite the action's optimality for an unbounded player. We build on their framework to provide a computational model of extensive form dynamic games of imperfect information.

The most complete framework from a cryptographic perspective that integrates game theoretic concepts was given by Groce et al. [32], which builds on the framework by Asharov et al. [15]. Asharov et al. demonstrate how standard cryptographic notions of security can be framed in a game theoretic view when considering malicious fail-stop adversaries. The authors demonstrate that privacy, correctness and fairness can be met using a game theoretic simulation-based framework. However, the framework only considers computational Nash equilibrium in extensive-form games of perfect information. We argue that a computational variant of PBE is preferable for constructing cryptographic protocols in a game theoretic framework, where players may not know the actions of other players when their computational abilities are bounded. The authors limit a player's strategy set to $\{\sigma^{\text{continue}}, \sigma^{\text{abort}}\}$, where at each node to follow σ^{continue} requires following the protocol specified by the mechanism designer precisely. From this the authors argue that non-credible threats in *fail-stop* games are meaningless, as a party that aborts cannot be punished. The work of Groce et al. [32] demonstrates that fairness can be achieved for a much broader class of utility functions than those specified by Asharov et al. [15]. Further, Groce et al. consider the *byzantine* case, where deviations are not limited to the fail-stop model. However, the equilibrium concept considered by Groce et al., namely Bayesian strict Nash equilibrium, does not *explicitly* model players' beliefs about the game state. Rather, this concept captures only uncertainty about the *types* of the other players. However, the players' beliefs about the current game state are modeled exogenously in Groce et al.'s setting. In cryptographic settings, a player's uncertainty about the current state is of critical importance, and we demonstrate the shortcomings of other equilibria concepts in Section 6.3. Our framework builds directly on Asharov et al.'s work, and as in Groce et al.'s setting, we allow for *arbitrary* deviation from the protocol beyond simple aborts.

6.3 Motivation

We motivate our approach by demonstrating cryptographic interactions where players' information is imperfect, and their beliefs must be formally modeled. Specifically, we show that a simple change to the rational secret sharing protocol used in the Groce et al. [32] framework results in a protocol where a rational player would cheat, but existing work predicts the player behaves honestly.

Cryptographic protocols proceed in a series of rounds, where at each round some subset of the parties select and play an action. Game theory models such interactions as extensive form dynamic games, where players move sequentially through a series of rounds, rather than normal form static games that model a single simultaneous interaction.

6.3.1 Imperfect Information

The information available to a player in a cryptographic protocol is of critical importance. The notion of computational security relies on the fact that players can be modeled as asymptotically bounded algorithms, and are only able to gain certain information with negligible probability. Consider for instance the *ciphertext indistinguishability* (IND-CPA) game [80]. In this game, an adversary $\mathcal{A}$ bound to probabilistic polynomial time (PPT) has two plaintext messages $\{m_0, m_1 : |m_0| = |m_1|\}$, and the challenger $\mathcal{C}$ has an asymmetric key pair $\{E_{\mathcal{C}}, D_{\mathcal{C}}\}$ from a public key cryptosystem. $\mathcal{C}$ publicizes $E_{\mathcal{C}}$, and $\mathcal{A}$ performs up to polynomially many encryptions before sending $\{m_0, m_1\}$ to $\mathcal{C}$. $\mathcal{C}$ selects a bit $b \in \{0, 1\}$ uniformly at random, and returns $E_{\mathcal{C}}(m_b)$ to $\mathcal{A}$. After performing at most polynomially many operations, $\mathcal{A}$ outputs a *guess* $b' \in \{0, 1\}$, and succeeds when $b' = b$. The cryptosystem is said to be IND-CPA secure if, for all PPT adversaries $\mathcal{A}$

$$|Pr[\mathcal{A}(E_{\mathcal{C}}(m_b)) = 1] - Pr[\mathcal{A}(E_{\mathcal{C}}(m_{1-b})) = 1]| \leq \epsilon(\lambda) \tag{6.1}$$

where $\epsilon(\cdot)$ is a negligible function and λ is the security parameter. Clearly this property reflects the inability of a computationally bounded adversary to distinguish between two cases. From a game theoretic perspective, we argue that this lack of knowledge is properly modeled as an extensive form dynamic game of *imperfect* information. When some player p_0 does not observe a previous action by another player p_1, we say that the game has imperfect information and p_0's *information set* is *non-singleton*. That is, p_0 only knows that p_1 has moved, and does not know which action was played.

In the IND-CPA ciphertext indistinguishability game, $\mathcal{A}$ has imperfect information as it does not observe $\mathcal{C}$'s action $b \mapsto \{0, 1\}$. Thus, $\mathcal{C}$'s information set contains both the left ($b \mapsto 0$) and right ($b \mapsto 1$) nodes of the game tree Γ under the assumption that $\mathcal{C}$ is bound to PPT. Current rational multiparty computation frameworks consider solution concepts that require perfect information, and do not formally model players' information and beliefs. For instance, if $\mathcal{A}$ had some auxiliary information (e.g., $\mathcal{C}$'s random seed), it may be able to predict $\mathcal{C}$'s choice for b with probability non-negligibly greater than $\frac{1}{2}$. Thus, any solution concept must explicitly model the fact that moves in cryptographic interactions are frequently unobserved, and also that players may have auxiliary information or beliefs that influence their strategy selection.

6.3.2 Updating Beliefs

Additionally, players typically update their *beliefs* throughout cryptographic protocols based on observed events. Consider the case of interactive zero-knowledge proof systems [6]. This game is an interaction between a prover $\mathcal{P}$ in possession of a secret, and a verifier $\mathcal{V}$ that is to learn *only* whether or not $\mathcal{P}$ does, in fact, know the secret. In each round, a prover not in possession of the secret succeeds with probability $0 < p < 1$. Thus, $\mathcal{V}$ must interact with $\mathcal{P}$ through k rounds until $1 - p^k$ is acceptably close to 1. If at any round $\mathcal{P}$ fails the test, then $\mathcal{V}$ knows with certainty that $\mathcal{P}$ does not possess the secret and the game terminates. However, the likelihood that $\mathcal{P}$ *does* know the secret approaches 1 as $k \to \infty$. Thus, $\mathcal{V}$ is consistently updating a belief about $\mathcal{P}$ throughout the protocol.

6.3.3 Dynamic Games

In game theory, games may be either *static* or *dynamic*. In the former, actions are played simultaneously, while in the latter actions may be played sequentially. In a computational setting, this is equivalent to deciding between whether or not to assume the existence of a *broadcast channel.* As broadcast channels are a relaxation of real world interactions, removing this assumption is desirable as it allows players to act in a specified order. This introduces non-trivial issues into protocols that may be very basic in the semi-honest model, such as the recovery protocol for secret sharing. This protocol was modeled as an extensive form dynamic game by Groce et al. [32], who give a solution when players must move sequentially in a known order.

6.3.4 Non-credible Threats

Recently, Halpern et al. [34] showed that a Nash equilibrium is guaranteed to exist for all finite machine games under the assumption that randomization is free. However, their framework considers only Bayesian Nash equilibrium: an equilibrium concept susceptible to implausible equilibria through non-credible threats. A threat is not considered credible if it is "off the equilibrium path" for a player. That is, action a is not credible if player i receives a greater *expected* utility by playing action $a' \neq a$. We consider a *computational* non-credible threat to be any action a where there exists another action a' that yields negligibly less utility and is computable subject to the player's complexity bound $\mathscr{C}$. Our definition assumes that a player will choose the optimal strategy whenever their complexity $\mathscr{C}$ allows such action to be performed.

6.3.5 Rational Secret Sharing

The necessity of modeling imperfect information, and the difficulty imposed when broadcast channels cannot be assumed, is easily illustrated using the most common example of rational cryptographic protocols to date: *rational secret sharing* [22, 24, 25, 32, 35, 81]. First introduced by Halpern and Teague [23], the authors consider a set of purely rational players, seeking only to maximize their respective utility functions. This departs from the standard security models in cryptography, which assume two distinct types: semi-honest players that follow the protocol specification while possibly analyzing the protocol transcript in an attempt to learn more information, and malicious players that may deviate arbitrarily. The goal of general secret sharing is to split a secret among n parties such that any k shares are sufficient to recover the secret value, using a scheme such as the polynomial interpolation approach proposed by Shamir [82]. Rational secret sharing, introduced by Halpern and Teague [23], is particularly concerned with the process of *recovering* the secret from the shares. As noted by Halpern et al. [23], rational players' utility functions are assumed to value *exclusivity*, where preference is given to learning the output of the function while preventing other players from doing so. Assume that a player's strategy set σ is limited to $\sigma \in \{H, \perp\}$, where H denotes the honest strategy of revealing the player's share, and $\perp$ denotes the action of not revealing the share. Without loss of generality, assume μ^+ denotes positive utility, μ^- denotes negative utility, and μ^0 denotes no net change in utility. We formally define a utility function $\mu^f(\sigma_0, \sigma_1)$ for fairness where exclusivity is valued in Definition 6.3.1:

Definition 6.3.1 *Let π be a two-party protocol, f be a two-party function, and $\sigma \in \{H, \perp\}$. Then, for every x_0, x_1 as above the **utility function for fairness valuing exclusivity** for party p_i, denoted μ_i^f, is defined as:*

$$\mu_0^f(\sigma_0, \sigma_1) \mapsto \begin{cases} \mu^+ & : \text{output}_{\pi,0} = f(x_0, x_1) \wedge \text{output}_{\pi,1} \neq f(x_0, x_1) \\ \mu^0 & : \text{output}_{\pi,0} = f(x_0, x_1) \wedge \text{output}_{\pi,1} = f(x_0, x_1) \\ \mu^- & : \text{output}_{\pi,0} \neq f(x_0, x_1) \end{cases}$$

Under this assumption, no party has any incentive to distribute their share to the other parties. Rather, the equilibrium is to wait for other players to distribute their shares, as this is the only action that increases a player's utility function. The authors demonstrate that this implies no deterministic protocol exists where rational parties

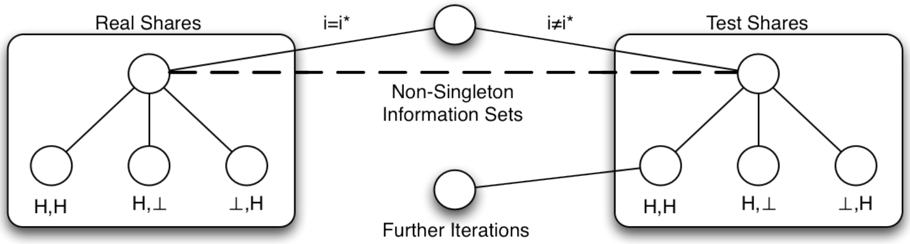


Figure 6.1.: Imperfect Information Sets in the Rational Secret Sharing Game

are willing to disseminate their shares to other players. However, their randomized protocol relies on the fact that parties are unaware whether the current state is terminal (allowing the secret to be recovered), or merely a "test" state (where the secret cannot be recovered, but players who do not distribute shares are caught as cheaters).

This fundamental lack of information constitutes an extensive form game of *imperfect* information, for which the Nash equilibrium (and computational variants thereof) are insufficient equilibria concepts.

Figure 6.1 illustrates the two-party rational secret sharing game Γ, which proceeds in a series of rounds. At round i, player p_0's share x_0^i may be a legitimate share, such that combined with p_1's share the secret may be recovered. However, p_0's share may also be illegitimate, such that the shares combine to a pre-determined test value that is not the original secret. Players are not aware whether the given round i is the terminal round i^* where the secret may be recovered, or a test round $i \neq i^*$ where no information may be learned from the shares. Assume that a player's strategy set σ is limited to $\sigma \in \{H, \perp\}$, where H denotes the honest strategy of revealing the player's share, and $\perp$ denotes the action of not revealing the share. By choosing i^* from a geometric distribution, as in Groce et al. [32], cheating players that choose strategy $\sigma = \perp$ when $i \neq i^*$ are caught and the game may be terminated.

Thus, players now have an incentive to distribute their share, as playing $\perp$ only yields μ^+ when $i = i^*$. When the perfect Bayesian equilibrium was proposed for modeling extensive form games of imperfect information by Harsanyi [33], the author specifically cautioned against using the standard Nash equilibrium concept. This view was echoed by Estevez-Tapiador et al. [83], specifically in the context of rational exchange. The presence of non-singleton information sets in the rational secret sharing game is illustrated by the dashed line between the two possible game states in Figure 6.1.

The difference between the Bayesian strict Nash equilibrium (BNE), used in the rational secret sharing setting of Groce et al. [32], and the perfect Bayesian equilibrium (PBE) concept we consider in our setting, bears clarification. If all moves were simultaneous, BNE and PBE would yield the same equilibria. However, in extensive form games of imperfect information, a player may not be able to observe all moves by other players. This results in non-singleton information sets, which BNE is unable to model, as it only considers uncertainty about players' types. Consequently, this uncertainty about the game state should be explicitly modeled into their expected utility. The PBE concept is able to "cut through" the non-singleton information sets present in the rational secret sharing game, as it considers players' beliefs about the type of other players *as well as beliefs about the current game state.* Thus, PBE avoids implausible equilibria that result from the presence of non-singleton information sets.

	H_1	$\perp_1$
H_0	(a_0, a_1)	(c_0, b_1)
$\perp_0$	(b_0, c_1)	(d_0, d_1)

Figure 6.2.: The Payoff Table for the Rational Secret Sharing Game

As in the setting of Groce et al. [32], assume that exclusivity holds. Thus, each player orders their preferences according to $b_i > a_i \geq d_i \geq c_i$, so that player p_i prefers to learn the correct result while p_{1-i} learns an erroneous result.

We review the share distribution and reconstruction phase considered by Groce et al. [32].

Share Generation:

- A value $i^* \in \{1, \dots\}$ is chosen according to a geometric distribution, and represents the iteration (unknown to the parties) in which both parties will learn the correct output.
- Values $r_1^0, r_1^1, \dots, r_n^0, r_n^1$ are chosen, with the $\{r_i^0\}_{i=1}^n$ intended for p_0 and the $\{r_i^1\}_{i=1}^n$ intended for p_1. For $i \geq i^*$, we have $r_i^0 = f_0(x_0, x_1)$ and $r_i^1 = f_1(x_0, x_1)$, while for $i < i^*$ the $\{r_i^0\}$ (resp., $\{r_i^1\}$) values depend on p_0's (resp. p_1's) input only.
- Each r_i^b value is randomly shared as s_i^b and t_i^b (with $r_i^b = s_i^b \oplus t_i^b$), where s_i^b is given to p_0 and t_i^b is given to p_1.

Share Recovery: For n iterations, do as follows:

- p_1 sends t_i^0 to p_0, enabling p_0 to learn r_i^0
- p_0 sends t_i^1 to p_1, enabling p_1 to learn r_i^1

When the protocol ends, a party outputs the most recently learned value of r_i.

We now review the guessing strategies employed by Groce et al. [32], which players use when the other player aborts the protocol prematurely. The guessing distribution W_i are chosen such that the strategy vector $\{(\mathit{cooperate}, W_0), (\mathit{cooperate}, W_1)\}$ is a Bayesian strict Nash Equilibrium. For all $i < i^*$, the r_i^j values are chosen from W_j, which is assumed to assign non-zero probability to all elements in the range of f. The critical issue with the Groce et al. [32] approach is that the expectation for utility *exogenously* considers the probability that $i \overset{?}{=} i^*$, rather than making this belief explicit in the equilibrium concept. Thus, they restrict a player to fix their strategy at the start of the game for consistency with BNE, even as a mediator introduces auxiliary information.

Consider a game where p_0 is given auxiliary information about whether $i \overset{?}{=} i^*$ after the game has started. Suppose the share generator reveals to p_0 that the current round i is, in fact, the terminal round i^*. This information crucially affects p_0's expected utility function under PBE, as p_0's beliefs about i^* have changed from the start of the game. This information should be explicitly factored into the calculation of expected utility, but the definition of Bayesian Nash equilibria ignores this, focusing on uncertainty only about the player's types. Thus, even in the case where p_0 *knows* the correct value of i^* at some round k, the BNE for the above game predicts that the player will play honestly and reveal their share. However, PBE allows p_0 to update their belief about i^* as the game progresses, and requires that all subsequent play be optimal with respect to their beliefs. Thus, PBE predicts that p_0 should not reveal their share, and instead collect p_1's share to recover the secret. Given p_0's beliefs about the game state, this clearly maximizes p_0's expected utility. The equilibrium predicted by BNE, namely for p_0 to distribute their share, is implausible given the auxiliary information provided to p_0 and the fact that p_0 values exclusivity. This implausible equilibrium is avoided when the PBE concept is used.

Formally, assume that players have reached round i of the rational secret sharing recovery game Γ. There are at most $n > i$ rounds in Γ, and the terminal round i^* is chosen from a geometric distribution. In the setting of Groce et al. [32], incentive compatibility must hold for all players *a priori*. To accomplish this, we must set the parameter α, the probability of success. Since in each round i there are two distinct possibilities (namely, $i = i^*$ and $i \neq i^*$), we must set α such that no party has an incentive to abort during the game. Groce et al. [32] show that such an α exists. All that remains to be shown is that the uncertainty of a player about the *types* of the other players does not provide an incentive to abort. Assume that a player type $t_i \in \{continue, abort\}$. The fact that uncertainty about types does not induce a player to abort follows from the assumption of incentive compatibility. That is, all players know i is chosen from a geometric distribution parameterized by α, so no player has an incentive to abort. At round i, each player has observed all other players continue the protocol, otherwise the game would have terminated. Thus, all players are convinced that the remaining players are of type *continue*. This is the BNE equilibrium for Γ, *even when uncertainty about the game state is introduced*, as the BNE concept does not consider game state beliefs.

The PBE equilibrium concept weights strategies according to a player's beliefs about other players' types *and the current game state*. These beliefs are updated throughout the game based on observed actions and auxiliary information provided to the player. Recall that player p_i receives payoff b_i when p_i selects abort ($\perp$) while all other players select *continue* (H). The payoff $b_i > a_i$, where a_i denotes the case where all players select H. The probability that $i = i^*$ is given by the CDF for a geometric distribution parameterized with α:

$$Pr[i = i^*] = 1 - (1 - \alpha)^i \tag{6.2}$$

Thus, in PBE players would weight b_i by $Pr[i = i^*]$ and a_i by $1 - Pr[i = i^*]$. As Groce et al. have demonstrated, α can be chosen to guarantee that $a_i(1 - \alpha)^i > b_i(1 - (1 - \alpha)^i)$, so BNE and PBE yield the same equilibrium for Γ. However, if we introduce *auxiliary information* about i^* to player p_i, then the equilibriums diverge. Assume at round i a mediator informs p_i that $i = i^*$. Now, we have:

$$\begin{aligned}
\mu_i(abort) &= Pr[i = i^*]b_i && (6.3)\\
&= (1)b_i && (6.4)\\
&> Pr[i \neq i^*]a_i && (6.5)\\
&= (0)a_i && (6.6)\\
&= \mu_i(continue) && (6.7)
\end{aligned}$$

Thus, the auxiliary information provided to p_i concerning the game state affects μ_i such that *abort* provides greater utility than *continue*. This is intuitive, as p_i values exclusivity, so knowing i^* induces a decision to abort. However, BNE does *not* weight strategies by uncertainty about the game state. Even though p_i is aware $i = i^*$, BNE ranks $\mu_i(continue) > \mu_i(abort)$, which is an implausible equilibrium.

Given these observations, we will argue that the notion of PBE is a more appropriate solution concept than those previously proposed, as it explicitly models games of imperfect information.

6.4 Computational Setting

Cryptographic protocol construction necessarily requires the computational ability of players to be explicitly modeled. Game theory makes no such assumptions; the computational abilities of the players are considered unlimited, and do not affect their utility functions. Thus, any game theoretic framework for building cryptographic protocols requires that computational limitations be taken into account.

Interactive Turing Machines

In order to transform the standard game theoretic definitions to the computational setting, we must redefine functionalities to be computable by an interactive Turing machine (ITM)[2], and explicitly model the complexity of players' ITMs following the work of Halpern et al. [34].

Definition 6.4.1 *Let $\vec{M} = M_1 \times \cdots \times M_n$ denote the set of ITMs that terminate with probability 1. $\forall M \in \vec{M}$, we have that M consists of a finite read-only input tape M_I, a finite read-only random tape M_R with elements drawn uniformly at random from $\{0,1\}^*$, a finite read-write work tape M_W, a finite read-only communication tape M_C, and a finite write-only output tape M_O.*

As players are now modeled as ITMs, actions and types are represented as elements drawn from $\{0,1\}^*$ and correspond to M's input and output.

Following Halpern et al.'s [34] definitions, we define a *view* as the pair $v = (t, r) \in (\{0,1\}^*, \{0,1\}^*)$, where t is the *type* of the player read from M_I, and r is the finite bit string M reads from M_R. We define $M(v)$ to be the finite output written to M_O.

Each player's ITM is bounded by an associated *complexity function* $\mathscr{C} : \vec{M} \times \{0,1\}^* \to \mathbb{N}$. When considering our framework's application to cryptography, it is useful to define the complexity in terms of a globally known *security parameter* λ, as in Asharov et al.'s work [15]. For any machine M with $\mathscr{C}(M, \lambda) = 0$, we require that $M = \perp$, where $\perp$ denotes the ITM that does not read M_I, write to M_O, or change states.

[2] Modeling players as ITMs is the approach taken by Halpern et al. [34] and Asharov et al. [15], as this is a foundational model in the cryptographic literature [5, 6].

6.5 Perfect Bayesian Equilibrium

Formal definitions of perfect Bayesian equilibria (PBE) are usually not generalizable to general extensive form games, and contain the vague requirement that beliefs be updated according to Bayes' rule "whenever possible". Bonanno [37] gives a definition of PBE that is applicable for general extensive form games, but we will use the definition by Diaz et al. [38], as they go further by extending to general extensive form games as well as clarifying the ambiguous "whenever possible" updating requirement.

We first require that, for player $i \in N$, their *assessment* (σ_i, β_i) consisting of a strategy σ_i and a *belief* β_i about the game state, be sequentially rational:

Definition 6.5.1 *An assessment (σ_i, β_i) is (computationally)* ***sequentially rational*** *if, for every player $i \in N$ and every information set $I_i \in \mathcal{I}_i$, there holds:*

$$\mu_i(\sigma_i, \beta_i|I_i) + \epsilon(\lambda) \geq \mu_i((\sigma_{-i}, \sigma_i'), \beta_i|I_i) \tag{6.8}$$

for every strategy σ_i', a probability distribution over actions, of player i, where (σ_{-i}, σ_i') is a strategy profiles that all players stick to the strategy $\vec{\sigma}$ except that player i turns to the strategy σ_i', and $\mu_i((\sigma_{-i}, \sigma_i'), \beta_i|I_i)$ denotes player i's utility induced by this strategy profile and the belief system β_i, a probability distribution over game states, conditional on I_i being reached. The term $\epsilon(\lambda)$ denotes a negligible utility gain with respect to the security parameter λ, and σ_i is an efficiently computable strategy for player i with complexity $\mathscr{C}$.

Next, we give the definition of a *weak perfect Bayesian equilibrium*, which we build on to construct the final equilibrium concept that applies to general extensive form games:

Definition 6.5.2 *Let Γ be an extensive form game. An assessment (σ, β) is a* ***weak perfect Bayesian equilibrium*** *if it is sequentially rational and, on the path of σ, β is derived from σ from Bayes' rule.*

With this, we reach the definition of a *$\mathscr{C}$-simple perfect Bayesian equilibrium*:

Definition 6.5.3 *Let Γ be an extensive form game. An assessment (σ, β) is a* ***$\mathscr{C}$-simple perfect Bayesian equilibrium*** *if, for each regular information set I_i^k, the restriction of (σ, β) to $\Gamma_{I_i^k}(\sigma, \beta)$ is sequentially rational and β is obtained by* conditional updating *from σ (i.e., the restriction of (σ, β) to $\Gamma_{I_i^k}(\sigma, \beta)$ is a weak perfect Bayesian equilibrium), where σ is efficiently computable by an interactive Turing machine (ITM) with complexity $\mathscr{C}$.*

6.6 Framework

In order to show the application of game theoretic models to cryptography, a proper security model must be introduced. Thus, we consider appropriate game theoretic definitions of privacy, correctness and fairness.

Our framework is an extension of Asharov et al.'s [15] model of security under *fail-stop* games. The original work considered two players with action sets limited to $\{\sigma^{\text{abort}}, \sigma^{\text{continue}}\}$, where σ^{abort} implied that the ITM output a special signal $\perp$ observed by all players and stopped playing the game, and σ^{continue} is the strategy of following the game specification *without deviation.* Thus, the only deviating strategy is to abort the protocol, which is similar to the standard semi-honest security model. We extend this model to assume that σ^{continue} is precisely the vector of strategies of *not aborting*, regardless of whether or not the chosen action is the honest choice. Similarly, $\sigma^{\text{deviate}} = \{\sigma^{U}/\{\sigma^{\text{honest}}, \sigma^{\text{abort}}\}\}$ is the *set* of all possible strategies that are *dishonest*, taking σ^{U} to be the universe of strategies. That is, σ^{deviate} corresponds to choosing a strategy σ that deviates from the prescribed protocol. Without loss of generality, we assume that $\sigma^{\text{continue}} = \{\sigma^{\text{honest}}, \sigma^{\text{deviate}}\}$, where σ^{honest} is equivalent to following the prescribed protocol. As multiparty computation players are assumed to be mutually distrustful in the cryptographic literature, we assume they are *risk-averse* in the game theoretic sense. Thus, when an honest player cannot distinguish between the probability of $\mathcal{A}$ selecting $\sigma_{\mathcal{A}}^{\text{deviate}}$ or $\sigma_{\mathcal{A}}^{\text{honest}}$, the honest party assumes that $\sigma_{\mathcal{A}}^{\text{deviate}}$ was selected. We consider only the two-party case, as the extension to multiple parties requires modeling player collusion. Throughout, we let μ^{+} represents positive utility gain, μ^{-} represent negative utility, and μ^{0} represents neutral utility. We now give novel definitions of privacy, correctness and fairness in purely game theoretic terms, considering a more expressive model where players may deviate *arbitrarily* from the protocol beyond simply aborting.

6.6.1 Privacy

We follow Asharov et al.'s [15] intuition and require that parties' utility functions reflect the loss of privacy with negative utility. This requires no assumptions about other players' utility functions with respect to the *gain* of information; *the burden is player specific* and known, as we assume players are aware of their own utility functions. Thus, players may choose to require that any subset of privacy, correctness and fairness are satisfied by the protocol.

We first introduce a new notion of indistinguishability defined in terms of a $\mathscr{C}$-bounded distinguisher $\mathcal{D}$'s ability to differentiate between information sets. We first introduce notation for an ITM's local history:

Definition 6.6.1 *Let $\pi = (M_0, M_1)$ be a two-party protocol between a pair of ITMs (M_0, M_1). Then we write*

$$\mathcal{H}^k_{\pi,i}(x_0, x_1, \lambda) = (x_i, M_R, m^i_1, \dots, m^i_k) \tag{6.9}$$

to denote the local history of M_i at round k, with input x_i, random tape M_R, security parameter λ and m^i_j represents the j^{th} message.

We consider the set of infinitely many input tuples $(x_0, x^0_1, x^1_1, \lambda)$ where we have that $|x_0| = |x^0_1| = |x^1_1| = \lambda$, and party p_0's input is fixed at x_0 while p_1's input is in the set $\{x^0_1, x^1_1\}$.

Definition 6.6.2 *We say that a finite extensive form computational game Γ^λ has **indistinguishable initial information sets in the presence of $\mathscr{C}$-bounded adversaries** if:*

$$|Pr[(\mathcal{H}^0_{\pi,\mathcal{D}}(x_0, x^0_1, \lambda) \in I_0) = 1] - Pr[(\mathcal{H}^0_{\pi,\mathcal{D}}(x_0, x^1_1, \lambda) \in I_0) = 1]| \leq \epsilon(\lambda) \tag{6.10}$$

for some negligible function $\epsilon(\cdot)$.

That is, no $\mathscr{C}$-bounded distinguisher $\mathcal{D}$ can distinguish the *type* (i.e., private input) of party p_1 with probability non-negligibly greater than $\frac{1}{2}$. With this notion formally defined, we now give a definition for players' utility functions with respect to privacy:

Definition 6.6.3 *Let π be a two-party protocol and f be a two-party function. Then, for every x^0_0, x^1_0, x_1 such that $f(x^0_0, x_1) = f(x^1_0, x_1)$, and for every $\mathscr{C}$-bounded distinguisher $\mathcal{D}$, the **utility function for privacy** μ^p for party p_i, on input $x_0 \in \{x^0_0, x^1_0\}$, is defined by*

- $\mu^p_0(\mathcal{H}^{\varnothing}_{\pi,i}) = 0$ *when p_0 aborts immediately*
- $\mu^p_0(\mathcal{H}^k_{\pi,0}(x^b_0, x_1, \lambda)) \mapsto \begin{cases} \mu^- & : \text{guess}_\pi((\mathcal{H}^k_{\pi,\mathcal{D}}(x^b_0, x_1, \lambda) \in I_\mathcal{D})) = b', x^b_0 = x^{b'}_0 \\ \mu^+ & : \textit{otherwise} \end{cases}$

where we write guess : $\mathcal{H} \rightarrow T_i$ *to denote a function mapping a player's history to other players' types.*

Initially π is run, then $\mathcal{D}$ is given as input the local state of π w.r.t. p_i and two auxiliary values (x_0^0, x_0^1). $\mathcal{D}$ outputs a guess $b' \in \{0,1\}$, where $\mathcal{D}$ succeeds whenever $x_0^b = x_0^{b'}$.

For all rational players with utility functions $\mu \in \mu^p$, we have that $\mu(\sigma^{\text{continue}}) > \mu(\sigma^{\text{abort}})$ *iff*:

$$Pr[\text{guess}_\pi((\mathcal{H}^k_{\pi,\mathcal{D}}(x_0^b, x_1, \lambda) \in I_{\mathcal{D}})) = 1] = \frac{1}{2} + \epsilon(\lambda) \tag{6.11}$$

That is, a rational party with a utility function preferring privacy ($\mu \in \mu^p$) only continues participating in the protocol (i.e., by selecting a strategy in σ^{continue}) if for all $\mathscr{C}$-bounded adversaries, the probability of success is at most negligibly greater than $\frac{1}{2}$. We let σ^{deviate} imply that

$$Pr[\text{guess}_\pi((\mathcal{H}^k_{\pi,\mathcal{D}}(x_0^b, x_1, \lambda) \in I_{\mathcal{D}})) = 1] > \frac{1}{2} + \epsilon(\lambda) \tag{6.12}$$

That is, by playing σ^{deviate} the adversary has an advantage at breaking the privacy of the protocol with probability non-negligibly greater than $\frac{1}{2}$. Any other strategy $\sigma \notin \sigma^{\text{deviate}}$ will not affect *privacy* under this assumption, although *it may affect correctness or fairness.* We restrict our attention to privacy at the moment.

Definition 6.6.4 *Let f and π be as above. Then, π is* ***$\mathscr{C}$-Game-Theoretic Private*** *for party p_i if $\mu_i(\sigma_0^{\text{honest}}, \sigma_1^{\text{honest}})$ is a $\mathscr{C}$-PBE with respect to $\mu^p_{i,i\in\{0,1\}}, \beta_{i,i\in\{0,1\}}$ and all $\mathscr{C}$-bounded distinguishers $\mathcal{D}$.*

Our next theorem proves how to define a protocol π such that π will satisfy Definition 6.6.4:

Theorem 6.6.1 *Let f be a deterministic two-party function, and let π be a two-party protocol that computes f correctly. Then, π is $\mathscr{C}$-**Game-Theoretic Private** w.r.t. p_0 (resp. p_1) iff π has indistinguishable initial information sets in the presence of $\mathscr{C}$-bounded adversaries.*

Proof [Theorem 6.6.1] We first demonstrate that if π is $\mathscr{C}$-Game-Theoretic Private w.r.t. p_0, then π has indistinguishable initial information sets w.r.t. p_0 in the presence of $\mathscr{C}$-bounded adversaries.

If π is $\mathscr{C}$-Game-Theoretic Private w.r.t. p_0, then by definition we have that:

$$\mu_0(\sigma_0^{\text{honest}}|\beta_0, \mathcal{H}_0) + \epsilon(\lambda) \geq \mu_0(\sigma_0', \sigma_0^{\neg\text{honest}}|\beta_0, \mathcal{H}_0) \tag{6.13}$$

That is, if π is $\mathscr{C}$-Game-Theoretic Private, then players receive more utility by playing strategy σ^{honest} than any other strategy $\sigma^{\neg\text{honest}} = \{\sigma^U/\sigma^{\text{honest}}\}$. Assume by contradiction that π does not have indistinguishable initial information sets w.r.t. p_0. Without loss of generality, we assume $\mathcal{A}$ corrupts p_1. Then a $\mathscr{C}$-bounded adversary $\mathcal{A}$ is able to choose a strategy in the set $\sigma_1^{\text{deviate}}$, where $\mathcal{A}$ invokes a $\mathscr{C}$-bounded distinguisher $\mathcal{D}$ which succeeds in differentiating p_0's information set with probability

$$Pr[\text{guess}_\pi((\mathcal{H}^k_{\pi,\mathcal{D}}(x_0^b, x_1, \lambda) \in I_{\mathcal{D}})) = 1] > \frac{1}{2} + \epsilon(\lambda) \tag{6.14}$$

as given by Equation 6.12, which is a non-negligible advantage. Thus, we have that:

$$\begin{aligned}
\mu_0(\sigma_0^{\text{honest}}, \sigma_1^{\text{deviate}}) &= Pr[\text{guess}_\pi((\mathcal{H}^k_{\pi,\mathcal{D}}(x_0^b, x_1, \lambda) \in I_{\mathcal{D}})) = 1] \cdot \mu^- && (6.15)\\
&+ Pr[\text{guess}_\pi((\mathcal{H}^k_{\pi,\mathcal{D}}(x_0^b, x_1, \lambda) \in I_{\mathcal{D}})) = 0] \cdot \mu^+ && (6.16)\\
&< \mu^0 < \mu_0(\sigma_0^{\text{abort}}, \sigma_1^{\text{deviate}}) = \mu^0 && (6.17)
\end{aligned}$$

thus contradicting the assumption that σ_0^{honest} is a $\mathscr{C}$-PBE w.r.t. p_0, and that π is $\mathscr{C}$-Game-Theoretic Private by Definition 6.6.4, as σ^{abort} yields more utility for p_0 than σ^{honest}.

Next, we show that if π has indistinguishable initial information sets w.r.t. p_0, then π is $\mathscr{C}$-Game-Theoretic Private. By definition, if π has indistinguishable initial information sets w.r.t. p_0, then there *does not* exist a strategy in the set $\sigma_{\mathcal{A}}^{\text{deviate}}$ such that, for any $\mathscr{C}$-bounded distinguisher $\mathcal{D}$ invoked by $\mathcal{A}$

$$Pr[\text{guess}_\pi((\mathcal{H}^k_{\pi,\mathcal{D}}(x_0^b, x_1, \lambda) \in I_{\mathcal{D}})) = 1] > \frac{1}{2} + \epsilon(\lambda) \tag{6.18}$$

Assume by contradiction that σ_0^{honest} is not a $\mathscr{C}$-PBE w.r.t. p_0. Then, we must have that:

$$\mu_0(\sigma_0^{\text{abort}}, \sigma_1^{\neg\text{deviate}}) > \mu_0(\sigma_0^{\text{honest}}, \sigma_1^{\neg\text{deviate}}) \tag{6.19}$$

Clearly we have that $\mathcal{A}$'s strategies are limited to $\sigma_1^{\neg\text{deviate}} = \{\sigma_1^{\text{honest}}, \sigma_1^{\text{abort}}\}$, as by assumption π has indistinguishable initial information sets w.r.t. p_0, so no strategy in the set $\sigma_1^{\text{deviate}}$ exists by Equation 6.12. Consider first the strategy pair $(\sigma_0^{\text{abort}}, \sigma_1^{\text{abort}})$:

$$\mu_0(\sigma_0^{\text{abort}}, \sigma_1^{\text{abort}}) = \mu^0 = \mu_0(\sigma_0^{\text{honest}}, \sigma_1^{\text{abort}}) \tag{6.20}$$

Thus, σ_0^{honest} is a $\mathscr{C}$-PBE w.r.t. p_0, contradicting the assumption. Similarly, consider the strategy pair $(\sigma_0^{\text{abort}}, \sigma_1^{\text{honest}})$:

$$\mu_0(\sigma_0^{\text{abort}}, \sigma_1^{\text{honest}}) = \mu^0 < \mu_0(\sigma_0^{\text{honest}}, \sigma_1^{\text{honest}}) = \mu^+ \tag{6.21}$$

Thus, σ_0^{honest} is a $\mathscr{C}$-PBE w.r.t. p_0, contradicting the assumption.

■

6.6.2 Correctness

Asharov et al.'s [15] notion of correctness is similar to their notion of privacy: party p_i prefers to learn the correct output of the function f to learning an incorrect output. We modify their definition with respect to the utility gained from aborting before the protocol starts. Rather than specify this utility as $\mu_i^c(\mathcal{H}_{\pi,i}^{\varnothing}) = \mu^+$, we say that a party that does not participate in the protocol receives $\mu_i^c(\mathcal{H}_{\pi,i}^{\varnothing}) = \mu^0$, so that parties prefer to participate in the protocol. As defined in the original work, players receive the same utility for not participating as they do for receiving the correct output of the function. As we assume computation is costly, it seems more natural to assign greater utility to receiving the correct output of the function.

As previously specified when considering privacy, we consider the set of infinitely many input tuples $(x_0, x_1^0, x_1^1, \lambda)$ where we have that $|x_0| = |x_1^0| = |x_1^1| = \lambda$, and party p_0's input is fixed at x_0 while p_1's input is in the set $\{x_1^0, x_1^1\}$.

Definition 6.6.5 *Let f be a deterministic two-party function, and let π be a two-party protocol that computes f correctly. Then, for every x_0, x_1 as above the* ***utility function for correctness*** *for party p_i, denoted μ_i^c, is defined as:*

- $\mu_i^c(\mathcal{H}_{\pi,i}^{\varnothing}) = \mu^0$
- $\mu_i^c(\text{output}_{\pi,i}, x_0, x_1) \mapsto \begin{cases} \mu^+ & : \text{output}_{\pi,i} = f(x_0, x_1) \\ \mu^- & : \textit{otherwise} \end{cases}$

We consider σ^{honest} to represent the strategy that follows the protocol specification of π, which by definition computes f correctly. Similarly, any other strategy $\sigma \in \{\sigma^{\text{deviate}}, \sigma^{\text{abort}}\}$ is assumed to compute f *incorrectly*. That is, we limit σ^{deviate} to those strategies that yield an incorrect output. Other strategies certainly exist in σ^{deviate} that will not alter the result, but these are handled when privacy and fairness are required. To satisfy the correctness condition, we need only consider those strategies in σ^{deviate} that yield incorrect outputs of f.

Definition 6.6.6 *Let f and π be as above. Then, π is $\mathscr{C}$-**Game-Theoretic Correct** for party p_i if $\mu_i^c(\sigma_0^{\text{honest}}, \sigma_1^{\text{honest}})$ is a $\mathscr{C}$-PBE with respect to $\mu^c_{i,i\in\{0,1\}}, \beta_{i,i\in\{0,1\}}$ and all $\mathscr{C}$-bounded adversaries $\mathcal{A}$.*

We now prove a theorem defining how protocol π may satisfy Definition 6.6.6:

Theorem 6.6.2 *Let f be a deterministic two-party function, and let π be a two-party protocol that computes f correctly. Then, π is $\mathscr{C}$-**Game-Theoretic Correct*** w.r.t. *p_0 (resp. p_1) if*

$$\forall \beta_0, \sigma_1^{\text{deviate}} \in I_0(\mathcal{H}^k) \implies I_0(\mathcal{H}^k) = \{\sigma_1^{\text{deviate}}\} \tag{6.22}$$

That is, all information sets containing strategy $\sigma_1^{\text{deviate}}$ are singleton nodes, distinguishable by any distinguisher $\mathcal{D}$ of bounded complexity $\mathscr{C}$.

Proof [Theorem 6.6.2] We demonstrate that if π is $\mathscr{C}$-Game-Theoretic Correct w.r.t. p_0, then $\forall \beta_0, \sigma_1^{\text{deviate}} \in I_0(\mathcal{H}^k) \implies I_0(\mathcal{H}^k) = \{\sigma_1^{\text{deviate}}\}$. Intuitively, this means that if π satisfies Definition 6.6.6, then p_0 must be able to differentiate p_1 selecting σ^{deviate} rather than $\sigma^{\neg\text{deviate}}$.

If π is $\mathscr{C}$-Game-Theoretic Correct w.r.t. p_0, then by definition we have that:

$$\mu_0^c(\sigma_0^{\text{honest}} | \beta_0, \mathcal{H}_0) + \epsilon(\lambda) \geq \mu_0^c(\sigma_0', \sigma_0^{\neg\text{honest}} | \beta_0, \mathcal{H}_0) \tag{6.23}$$

That is, if π is $\mathscr{C}$-Game-Theoretic Correct, then players receive greater utility by playing strategy σ^{honest} than any other strategy $\sigma^{\neg\text{honest}} = \{\sigma^U/\sigma^{\text{honest}}\}$. Assume by contradiction that

$$\forall\beta_0, \sigma_1^{\text{deviate}} \in I_0(\mathcal{H}^k) \not\Rightarrow I_0(\mathcal{H}^k) = \{\sigma_1^{\text{deviate}}\} \tag{6.24}$$

$$: \exists I_0(\mathcal{H}^k) = \{\sigma_1^{\text{abort}}, \sigma_1^{\text{honest}}, \sigma_1^{\text{deviate}}\} \tag{6.25}$$

That is, $\sigma_1^{\text{deviate}}$ exists in *non-singleton* information sets for p_0. Thus, for some previous history $\mathcal{H}_0^j, j < k$, we have that p_0 cannot distinguish between $\mathcal{H}^j = \{\sigma_1^{\text{deviate}}\}$ and $\mathcal{H}^j = \{\sigma_1^{\text{honest}}\}$, where we do not consider $\mathcal{H}^j = \{\sigma_1^{\text{abort}}\}$ as p_1 would output $\perp$, and p_0 would know with probability 1 that this strategy was used. Recall that risk-averse participants assume σ^{deviate} when information sets are non-singletons. We have that

$$\mu_0^c(\sigma_0^{\text{honest}}, \sigma_1^{\text{deviate}}) = \mu^- < \mu_0^c(\sigma_0^{\text{abort}}, \sigma_1^{\text{deviate}}) = \mu^0 \tag{6.26}$$

which contradicts the assumption that σ_0^{honest} is a $\mathscr{C}$-PBE, as aborting yields more utility than engaging in the protocol, and that π is $\mathscr{C}$-Game-Theoretic Correct w.r.t. p_0 by Definition 6.6.6. ■

6.6.3 Fairness

In Asharov et al.'s [15] original definitions for fairness, players are implicitly assumed to abide by the *exclusivity* property: a player prefers to be the *only* party to learn the output over a fair distribution of the function result. We argue that this assumption does not always hold.

Any framework constructed under the assumption of exclusivity is limited to the set of *non-cooperatively computable* (NCC) [77] functions. Let $f(\cdot,\cdot)$ be a two-party function, with party p_i holding input x_i, $i \in \{0,1\}$. If party p_i provides an alternate input $x_i' \neq x_i$ to f, a fair protocol outputs $f(x_i', x_{1-i})$ to all parties. However, if p_i can compute $g(f(x_i', x_{1-i}), x_i) = f(x_i, x_{1-i})$, then p_i has no *rational* incentive to provide their true input x_i as p_i alone can now deduce the correct output of the function $f(x_i, x_{1-i})$ from the output $f(x_i', x_{1-i})$. Thus, any framework requiring the *exclusivity* requirement is limited to functions for which the correct output cannot be produced given knowledge of the function and its output on a different input.

As an example, consider auction scenarios. Clearly, any adversary requires that all parties learn the output of the protocol *even if it is not the correct output*, as the result induces others to perform the actual goal of the protocol: distributing goods or services to the winner. If an adversary was the only party to receive the output, no distribution occurs and the effort was pointless.

We modify Asharov et al.'s [15] original utility function for fairness to reflect the fact that the exclusivity assumption does not always hold. Let E denote the set of players whose utility functions for fairness μ^f value *exclusivity*:

Definition 6.6.7 *Let π be a two-party protocol and f be a two-party function. Then, for every x_0, x_1 as above the* ***utility function for fairness*** *for party p_i, denoted μ_i^f, is defined as:*

$$\mu_0^f(\sigma_0, \sigma_1) \mapsto \begin{cases} \mu^+ & : \text{output}_{\pi,0} = f(x_0,x_1) \wedge \text{output}_{\pi,1} \neq f(x_0,x_1) \wedge p_0 \in E \\ \mu^+ & : \text{output}_{\pi,0} = f(x_0,x_1) \wedge \text{output}_{\pi,1} = f(x_0,x_1) \wedge p_0 \notin E \\ \mu^- & : \text{output}_{\pi,0} = f(x_0,x_1) \wedge \text{output}_{\pi,1} \neq f(x_0,x_1) \wedge p_0 \notin E \\ \mu^- & : \text{output}_{\pi,0} \neq f(x_0,x_1) \wedge \text{output}_{\pi,1} = f(x_0,x_1) \\ \mu^0 & : \textit{otherwise} \end{cases}$$

We consider σ^{honest} to represent the strategy that follows the protocol specification of π. Similarly, fairness is only compromised when a party selects σ^{abort}, which deprives other players of information necessary to compute the output.

Definition 6.6.8 *Let f and π be as above. Then, π is $\mathscr{C}$**-Game-Theoretic Fair** for party p_i if $\mu_i^f(\sigma_0^{\text{honest}}, \sigma_1^{\text{honest}})$ is a $\mathscr{C}$-PBE with respect to $\mu^f_{i,i\in\{0,1\}}, \beta_{i,i\in\{0,1\}}$ and all $\mathscr{C}$-bounded adversaries $\mathcal{A}$.*

We now prove a theorem defining how protocol π may satisfy Definition 6.6.8:

Theorem 6.6.3 *Let f be a deterministic two-party function, and let π be a two-party protocol that computes f correctly. Then, π is $\mathscr{C}$**-Game-Theoretic Fair** w.r.t. p_0 (resp. p_1) iff $\forall \mathcal{H}^k$*

$$|Pr[\text{output}_{\pi,0}(\mathcal{H}^k) = f(x_0, x_1)] - Pr[\text{output}_{\pi,1}(\mathcal{H}^k) = f(x_0, x_1)]| \leq \epsilon(\lambda) \tag{6.27}$$

That is, at any round k, the strategy σ^{abort} yields a player at most a negligible advantage over other players at determining the correct function output $f(x_0, x_1)$.

Proof [Theorem 6.6.3] We first demonstrate that if π is $\mathscr{C}$-Game-Theoretic Fair w.r.t. p_0, then p_1 has a negligible advantage over p_0 at determining the correct function output $f(x_0, x_1)$ when playing strategy σ_1^{abort}.

If π is $\mathscr{C}$-Game-Theoretic Fair w.r.t. p_0, then by definition we have that:

$$\mu_0(\sigma_0^{\text{honest}}|\beta_0, \mathcal{H}_0) + \epsilon(\lambda) \geq \mu_0(\sigma_0', \sigma_0^{\text{abort}}|\beta_0, \mathcal{H}_0) \tag{6.28}$$

That is, if π is $\mathscr{C}$-Game-Theoretic Fair, then players receive more utility by playing strategy σ^{honest} than aborting and attempting to recover $f(x_0, x_1)$ on their own. Assume by contradiction that p_1 has a non-negligible advantage over p_0 at determining the correct function output $f(x_0, x_1)$ when playing strategy σ_1^{abort}. Without loss of generality, we assume $\mathcal{A}$ corrupts p_1. Then we have that

$$Pr[\text{output}_{\pi,1}(\mathcal{H}^k) = f(x_0, x_1)] > \frac{1}{2} + \epsilon(\lambda) \tag{6.29}$$

which is a non-negligible advantage. Thus, we have that:

$$\begin{aligned}
\mu_0(\sigma_0^{\text{honest}}, \sigma_1^{\text{abort}}) &= Pr[\text{output}_{\pi,1}(\mathcal{H}^k) = f(x_0, x_1)] \cdot \mu^- && (6.30)\\
&+ Pr[\text{output}_{\pi,0}(\mathcal{H}^k) = f(x_0, x_1)] \cdot \mu^+ && (6.31)\\
&< \mu^0 < \mu_0(\sigma_0^{\text{abort}}, \sigma_1^{\text{abort}}) = \mu^0 && (6.32)
\end{aligned}$$

thus contradicting the assumption that σ_0^{honest} is a $\mathscr{C}$-PBE w.r.t. p_0, and that π is $\mathscr{C}$-Game-Theoretic Fair by Definition 6.6.8, as σ^{abort} yields more utility for p_0 than σ^{honest}.

Next, we show that if p_1 has at most a negligible advantage over p_0 at determining the correct function output $f(x_0, x_1)$ when playing strategy σ_1^{abort}, then π is $\mathscr{C}$-Game-Theoretic Fair. By definition, we have that

$$|Pr[\text{output}_{\pi,0}(\mathcal{H}^k) = f(x_0, x_1)] - Pr[\text{output}_{\pi,1}(\mathcal{H}^k) = f(x_0, x_1)]| \leq \epsilon(\lambda) \tag{6.33}$$

Assume by contradiction that σ_0^{honest} is not a $\mathscr{C}$-PBE w.r.t. p_0. Then, we must have that:

$$\mu_0(\sigma_0^{\text{abort}}, \sigma_1^{\text{abort}}) > \mu_0(\sigma_0^{\text{honest}}, \sigma_1^{\text{abort}}) \tag{6.34}$$

Consider first the strategy pair $(\sigma_0^{\text{abort}}, \sigma_1^{\text{abort}})$:

$$\mu_0(\sigma_0^{\text{abort}}, \sigma_1^{\text{abort}}) = \mu^0 = \mu_0(\sigma_0^{\text{honest}}, \sigma_1^{\text{abort}}) \tag{6.35}$$

Thus, σ_0^{honest} is a $\mathscr{C}$-PBE w.r.t. p_0, contradicting the assumption. Similarly, consider the strategy pair $(\sigma_0^{\text{abort}}, \sigma_1^{\text{honest}})$:

$$\mu_0(\sigma_0^{\text{abort}}, \sigma_1^{\text{honest}}) = \mu^0 < \mu_0(\sigma_0^{\text{honest}}, \sigma_1^{\text{honest}}) = \mu^+ \tag{6.36}$$

Thus, σ_0^{honest} is a $\mathscr{C}$-PBE w.r.t. p_0, contradicting the assumption.

■

6.7 Conclusion

We have presented an expressive two-party framework for reasoning about the security of cryptographic protocols in game theoretic terms, where all players are only assumed to be rational. We have demonstrated the ability of the perfect Bayesian equilibrium concept to model the inherent uncertainty and auxiliary information in cryptographic protocols, and translated this into the computational domain. Finally, we have provided novel definitions of privacy, correctness, and fairness in game theoretic terms, and demonstrated the conditions under which they hold.

We expand this two-party framework to the multiparty setting in Chapter 7. Using the multiparty model, we apply our framework to a series of classic games from the game theoretic literature. Additionally, we apply our framework to rational secret sharing, the most commonly examined rational cryptographic protocol.

7 REALIZING RATIONAL MULTIPARTY PROTOCOLS

We continue by building on the two-party rational framework of Chapter 6, and extend these results into the multiparty setting. A core difficulty when considering more than two players is the issue of collusion: players forming a coalition to undermine the ideal goal of the protocol. Collusion is enabled through a player's ability to communicate with others, both within and outside of the protocol. Thus, existing frameworks all impose restrictions on the communication interface in order to prevent players from colluding, and to preserve equilibria between game descriptions and realized protocols. In this work, we approach the issue of collusion from the opposite direction, asking if a meaningful notion of rational security can be achieved when players have access to point-to-point communication channels. We will demonstrate how to realize rational cryptographic protocols in practice from abstract game specifications. We argue that for real world protocols, it must be assumed that players have access to point-to-point communication channels. Thus, allowing signaling and strategy correlation becomes unavoidable. We argue that ideal world game descriptions of realizable protocols should include such communication resources as well, in order to facilitate the design of protocols in the real world. Our results specify a modified ideal and real world model that account for the presence of point-to-point communication channels between players, where security is achieved through the simulation paradigm.

7.1 Introduction

The field of rational cryptography departs from modeling players as either honest or malicious, and instead models all players as rational utility-maximizing agents:

each player chooses those actions that maximize their utility function $\mu(\cdot)$, which expresses their preferences over outcomes. All players may arbitrarily depart from the protocol specification if doing so is a utility-maximizing strategy. This approach to modeling removes the strong assumption of the semi-honest model: that honest players follow the protocol specification, regardless of whether or not it is in their best interest. By considering all players as rational agents, the standard properties of cryptographic protocols (e.g., privacy, correctness and fairness) are modeled through the utility functions of the players. Security of the protocol is then deduced from whether or not the stable equilibrium of the original game specification is reachable given the players' utility functions.

In secure multiparty computation (SMPC), the security of protocols are demonstrated through the *simulation paradigm*. Define an *ideal* protocol for computing a functionality f that invokes an incorruptible and universally trusted third party (TTP). Similarly, define a *real* protocol π for computing f where no TTP exists. Security is established if an adversary $\mathcal{A}$ in the real model has no advantage over a simulator $\mathcal{S}$ in the ideal model [4].

A major obstacle when defining security for rational multiparty protocols is the potential for players to form *coalitions*, colluding to undermine the security of the protocol. The strongest result, by Izmalkov et al. [64], allows any function to be computed securely by rational players using the approach of Goldreich et al. [4]. Although a universal result, it relies on strong assumptions including forced actions and physical primitives. A weaker notion, referred to as *collusion-free* computation [84–86], removes the ability of players to communicate additional information subliminally through the protocol communication resources. The result relies on a trusted mediator at the center of a star network topology, where all messages pass through the mediator and are re-randomized in order to prevent steganographic communication between the players. This result relies on adversarial independence, where simulators and adversaries are disallowed communication in the protocol. However, a collusion-free protocol may still cause issues when executed as part of a larger protocol. For example, the collusion-free protocols of Izmalkov et al. [64, 86] provide no guarantees when all players are malicious. This observation led to the work of Alwen et al. [87], where communication restrictions are further weakened to achieve *collusion-preserving* computation, which preserves any potential for collusion present in the original game specification. Although this result removes the requirement of a trusted mediator, it rules out a large class of communication resources (e.g., point-to-point and broadcast channels). Kamara et al. [88] consider a setting where adversaries have the capability to communicate additional information during protocol execution, yet choose to be *non-colluding*. Fuchsbauer et al. [22] give constructions under standard communication channels by forcing parties to send only unique messages as part of the protocol. Thus, collusion within the protocol is avoided, but communication outside of the protocol execution still facilitates collusion.

From this collection of work, addressing the issue of collusion appears to require strong limitations on the type of communication resources granted to players. As the general goal of rational cryptography is to provide a more realistic view of how players behave in cryptographic protocols, we consider what can be achieved when players have access to point-to-point communication channels - an unavoidable aspect in real world applications. Thus, in this work we define a security model where players may communicate information over point-to-point channels both inside and outside the protocol execution.

Our work proposes a new security framework for rational agents that models player access to point-to-point communication channels in the ideal world model. From this, we describe how to demonstrate the security of protocols in a real world model that implements games specified in our modified ideal world model. We note that imposing restrictions on the ideal world to capture unavoidable behavior exists currently in the cryptographic literature: it is a core feature of the malicious model, which extends the semi-honest model to consider more powerful adversaries. In the malicious setting, the ideal world must capture the ability of an adversary to coordinate the actions and inputs of players it corrupts, and force aborts during protocol execution; these actions are unavoidable in the presence of a monolithic malicious adversary. Our model necessarily limits the class of games that may be modeled in the ideal world formulation of our framework, as point-to-point communication channels must exist in the original game. Our work differs from existing formulations, which attempt to realize all games at the expense of restricting the communication interface available to players.

Throughout the remainder of the introduction, we argue that when point-to-point communication channels are unavoidable, it is meaningful to consider what games are realizable in their presence. We demonstrate that a non-trivial class of games constructed in our modified ideal world model have realizable implementations in the real world model through the Signaling game in Subsection 7.1.2, and the classic prisoner's dilemma in Section 7.2. Our technical contribution, a security model for realizing protocols from game specifications in the presence of point-to-point communication channels, is given in Section 7.3. The power of our model relative to others is first demonstrated on the Prisoners' Dilemma in Section 7.4. Finally, a full proof of security for the rational secret sharing protocol of Halpern and Teague [23] is given in Section 7.5, which is inadmissible under existing frameworks due to the presence of point-to-point communication channels. These examples demonstrate the key contribution of our model, which is less restrictive than prior work yet is able to correctly model the games' equilibria when played in the real world.

7.1.1 Local Adversaries

Translating the standard simulation paradigm to the game theoretic setting of rational cryptography requires addressing how adversaries should be modeled. In the original formulation, a centralized semi-honest or malicious adversary corrupts a subset of the players. However, rational cryptography makes no such distinction[1] between honest and corrupted players, and assumes all players are rational and acting to maximize their local utility function. Thus, translating the concept of an adversary is not immediate. Alwen et al. [87] give a collusion preserving framework where each player has an associated *local* adversary. Thus, the monolithic adversary of the standard model is shattered into an adversary for each individual player. Canetti et al. [63] argue that a local adversary should be defined for each ordered pair of players, as this provides a more granular model of the flow of information. Canetti et al. then demonstrate that the *local universal composition* (LUC) model can preserve the incentive structure in games.

We follow this modeling trend of shattering the monolithic adversary $\mathcal{A}$ into a *set* of local adversaries $\mathcal{A} = \{\mathcal{A}_i\}_{i\in[1\ldots n]}$ such that each player $\mathtt{P}_i \in \mathcal{P}$ is associated with adversary $\mathcal{A}_i$. Rather than considering local adversaries that "corrupt" their associated player $\mathtt{P}_i$, we simply require that the adversary selects the actions of $\mathtt{P}_i$ to maximize their local utility function μ_i. Thus, we preserve the assumption in rational cryptographic protocols that all players are purely rational and bound to a utility function, rather than remaining honest unless corrupted by a monolithic adversary.

[1] A mixed model has been proposed by Lysyanskaya et al. [65] where one subset of players are arbitrarily malicious, and the other subset are utility-maximizing rational agents.

7.1.2 Communication Resources

A core issue with existing work is how communication resources are modeled in game descriptions. In order to prevent players from signaling information or coordinating their actions, available communication resources are tightly restricted. For example, Izmalkov et al. [64] propose *rational secure computation* where only those equilibria in the game description exist in the realized protocol. However, this result comes at the cost of requiring forced actions and physical primitives such as opaque envelopes and ballot boxes[2]. Although not impossible to realize, in practice it has limited applicability.

In the ideal world model of secure multiparty computation, a protocol is viewed as an interaction between a set of players and a universally trusted third party (TTP). An ideal computation of a function has each player send their private input to the TTP, who computes the function and returns the results to each player. Restricting communication resources is not necessary, as players are assumed to be mutually distrustful. Further, any collusion between players is modeled through a monolithic adversary $\mathcal{A}$ that coordinates the actions of the players it corrupts.

In order to implement *arbitrary* games as protocols, strict notions of privacy preservation and the prevention of signaling and correlation must be satisfied. Arbitrary game specifications may impose restrictions on the communication resources available to players. Thus, the corresponding protocol implementation must not allow players to communicate more information than is possible in the ideal game specification. We briefly review the characteristics a model for implementing arbitrary games must satisfy[3]. We make the argument that even if a protocol satisfies all of these characteristics, it is likely to fall short of satisfying the ideal world model: communi-

[2] This result is a direct application of the GMW protocol [4].

[3] The ECRYPT summary report [89] on rational cryptographic protocols provides background on modeling techniques used to address privacy, signaling, and correlated actions.

cation between players outside of the protocol is unavoidable in real world settings. Thus, the model we present is not bound to satisfy these restrictions, and is a more accurate representation of what is achievable for protocols executed in the real world.

Privacy

A protocol π implementing an arbitrary game Γ must preserve both *pre-game privacy* and *post-game privacy* in addition to preserving the equilibrium of Γ. The notion of pre-game privacy ensures that the private input of each party is not revealed, as this will affect the actions of other parties. However, protocols implementing arbitrary games must also preserve the notion of post-game privacy, where nothing beyond the intended result (and what can be inferred from this) is revealed. This notion is necessary so that the equilibria of future games are not perturbed by information revealed in previous games.

Signaling

Similar to the notions of pre- and post-game privacy are the notions of *pre-game signaling* and *post-game signaling.* The ability to signal other players allows protocol participants to coordinate their actions to achieve a higher payoff. For example, consider two players $\mathtt{A}$ and $\mathtt{B}$ with inputs a and b. The payoff function is defined as $\Pi(\Gamma) \coloneqq a \oplus b$, and described in Table 7.1.

If $\mathtt{A}$ or $\mathtt{B}$ can signal even a single bit to the other, each will receive a payoff of 1 as opposed to an expected payoff of $\frac{1}{2}$. Thus, similar to the restriction on privacy, preventing pre- and post-game signaling is necessary to preserve the equilibria of individual and future games when constructing protocols for *arbitrary* games.

Table 7.1: Signaling Game

	`A` **sets** $a = 1$	`A` **sets** $a = 0$
`B` **sets** $b = 1$	(0,0)	(1,1)
`B` **sets** $b = 0$	(1,1)	(0,0)

The signaling game specification can be formulated under existing frameworks as a protocol, and demonstrated to preserve the mixed equilibrium of the original game. Yet by ignoring the ability of players to communicate outside of the protocol, the protocol formulation is invalidated in real world settings: players will collude to achieve a payoff of 1, rather than the expected payoff of $\frac{1}{2}$ of the original game specification.

We only consider those game specifications that allow point-to-point communication, as these channels are unavoidable in the real world. Thus, our model correctly predicts a payoff of 1 for players in the signaling game, as point-to-point communication channels allow signaling.

Correlated Actions

Correlated actions are similar to signaling, but allow parties to coordinate actions without exchanging information. This is usually accomplished through a shared value, such as a *common reference string* (CRS). The parties need not distribute information, but rather rely on the shared CRS to coordinate their actions. As with signaling, protocol constructions for arbitrary games must prevent pre- and post-game correlation to preserve equilibria in local as well as future games.

Table 7.2: Prisoner's Dilemma Game

	A **Remains Silent**	A **Confesses**
B **Remains Silent**	(-1,-1)	(0,-3)
B **Confesses**	(-3,0)	(-2,-2)

7.2 Prisoner's Dilemma

As a classic example, we consider the Prisoner's Dilemma[4]: a game between two suspects A and B that have been accused of committing both a principal and lesser crime. The Authority has sufficient evidence to convict both A and B on the lesser crime, punishable by 1 year in prison. However, there is insufficient evidence to convict A or B on the principal crime. The Authority separates A and B, and offers the following proposal: confess and serve no time while your partner serves 3 years in prison. Players A and B are then subject to the following dilemma:

1. If both A and B remain silent, they will each be convicted on the lesser crime and serve 1 year in prison.

2. If one confesses while the other remains silent, the confessor is set free while the other serves 3 years in prison.

3. If both A and B confess, each will serve 2 years in prison.

From the player payoffs listed in Table 7.2, note that each player maximizes their utility by confessing to the principal crime regardless of the strategy of their partner. We use this example to illustrate the necessity of removing monolithic adversaries, as well as how communication assumptions should be formulated in the ideal game

[4] The concept was originally proposed by Flood and Dresher while working at the RAND corporation, and is described in detail by Poundstone [90].

description. Note that the original ideal game specification of the prisoner's dilemma requires that the suspects $\mathtt{A}$ and $\mathtt{B}$ are physically separated: thus unable to communicate or otherwise coordinate their actions. However, we will construct a modified formulation in the presence of point-to-point communication channels with an *equivalent equilibrium* to the original formulation under our proposed model.

7.2.1 Monolithic Adversaries

Traditionally, cryptographic protocols are analyzed with respect to their resilience to a monolithic adversary $\mathcal{A}$ corrupting some subset of the players. Protocol resilience to adversarial corruption is quantified by the fraction of players that may be corrupted before the protocol security is violated.

In the game theoretic setting of rational cryptography, this model has been called into question by Alwen et al. [87] and Canetti et al. [63]. The goal of rational cryptography is to model each player as bound to their local utility function, rather than controlled by a monolithic adversary with a global utility function. The monolithic adversary in both of their models is shattered into a set of *local* adversaries unique to each player. Removing the monolithic adversary in favor of a set of local adversaries is critical to preserving game theoretic equilibria. In the running example of the Prisoner's Dilemma, consider the case where $\mathcal{A}$ corrupts both $\mathtt{A}$ and $\mathtt{B}$. As $\mathcal{A}$ controls both players, $\mathtt{A}$ and $\mathtt{B}$ may be forced to remain silent and achieve payoff $(-1, -1)$. However, consider the case where $\mathtt{A}$ (resp. $\mathtt{B}$) has a *local* adversary $\mathcal{A}_{\mathtt{A}}$ (resp. $\mathcal{A}_{\mathtt{B}}$): as $\mathcal{A}_{\mathtt{A}}$ is bound to the utility function $\mu_{\mathtt{A}}(\cdot)$ of $\mathtt{A}$, $\mathcal{A}_{\mathtt{A}}$ maximizes $\mu_{\mathtt{A}}(\cdot)$ by confessing as in the ideal specification of the game. An identical argument holds for $\mathcal{A}_{\mathtt{B}}$ as well. Thus, a monolithic adversary is capable of introducing a stable collusion equilibrium that does not exist in the ideal game specification, whereas the local adversary model preserves the original incentive structure.

7.2.2 Realistic Communication Model

To prevent pre- and post-game signaling and strategy correlation, many rational cryptographic frameworks impose strong restrictions on the communication resources available to players. This issue is most pronounced in the multiparty setting, where communication resources may enable collusion. To prevent communication resources from perturbing the equilibria of the ideal world game, existing constructions require forced player action and physical primitives [64], trusted mediators and forced broadcast channels [85], as well as the cooperation of adversarial players to deliver messages [87].

While these results provide strong guarantees under restrictive communication resource assumptions, the security guarantees are with respect to the protocol only. That is, assuming players may only interact through the protocol and its communication resources, the equilibria of the ideal world game is preserved. However, we argue that this results in a false sense of security for protocols realized in the real world, where players typically have access to point-to-point communication channels - undermining the strict communication assumptions of the protocol.

Our example of the prisoner's dilemma illustrates a salient point: the necessary and sufficient condition for preserving the equilibrium of the original formulation is the ability of $\texttt{A}$ and $\texttt{B}$ to *privately communicate* with the $\texttt{Authority}$. The original game specification requires the two players $\texttt{A}$ and $\texttt{B}$ to be physically separated, and thus unable to communicate. However, the key to preserving the equilibrium $(\mathsf{confess}, \mathsf{confess})$ of the original game Γ only requires preventing $\texttt{A}$ and $\texttt{B}$ from observing their interaction with the $\texttt{Authority}$. Consider a modified game $\bar{\Gamma}$ where all players $\{\texttt{A}, \texttt{B}, \texttt{Authority}\} \in \mathcal{P}$ have access to a point-to-point communication resource $\mathcal{R}$. As long as the communication links $\mathcal{R}_{\texttt{A},\texttt{Authority}}$, $\mathcal{R}_{\texttt{B},\texttt{Authority}}$ are private, the original equilibrium is preserved despite the presence of point-to-point communication channels. In game theoretic terms, communication between A and B through $\mathcal{R}_{\mathtt{A},\mathtt{B}}$ is considered *cheap talk*, as both A and B will claim to play silent, yet as utility maximizing agents they choose to confess, which strictly dominates silent. As neither A nor B can observe the message sent by the other to Authority, the coalition is unstable and disintegrates despite the presence of point-to-point communication channels.

7.3 Our Contribution

We argue that ideal world protocols should assume that players have the ability to communicate over point-to-point channels. As in the standard SMPC ideal world model, players may not wish to communicate due to mutual distrust. However, the option to do so should be part of the model, as this is unavoidable in the real world. Thus, we present a modified ideal world model capturing the presence of point-to-point communication channels between all players. Specifically, we answer the following questions:

1. How is security formalized when all players are rational and have access to point-to-point communication channels?
2. What benefits result from weakening the security guarantees of the standard malicious model by considering rational players with local adversaries?

7.3.1 Unstable Coalitions

A powerful aspect of the rational cryptographic setting with local adversaries is the ability to design protocols where coalitions are unstable. As each player has a local adversary that selects their actions in order to maximize a utility function, protocols may be designed to incentivize players to *leave* coalitions [91]. This benefit of modeling each player as an independently rational agent is frequently overlooked, and allows game equilibria to be preserved despite the presence of point-to-point communication channels. We have illustrated the power of unstable coalitions through our example of the prisoner's dilemma. We now consider coalition stability in the setting of *rational secret sharing*, as it is the most familiar example of a rational cryptographic protocol.

Rational Secret Sharing

Candidate definitions for achieving security against rational agents should accurately model well-studied problems in rational cryptography. The most familiar rational cryptographic protocol is *rational secret sharing* [22, 24, 25, 32, 35, 81]. The goal of *threshold* secret sharing is to split a secret among n parties such that any k shares are sufficient to recover the secret value, using a scheme such as the polynomial interpolation approach proposed by Shamir [82]. Rational secret sharing, introduced by Halpern and Teague [23], is particularly concerned with the process of recovering the secret from the shares[5]. As noted by Halpern et al. [23], rational players' utility functions are assumed to value *exclusivity*, where preference is given to learning the output of the function while preventing other players from doing so. Under this assumption, no party has any incentive to distribute their share to the other parties, which destabilizes coalition formation. The equilibrium is to wait for other players to distribute their shares, as this is the only action that increases a player's utility function. Thus, a player that does not distribute their share has the potential to be the exclusive player to recover the secret.

The authors demonstrate that this implies no deterministic protocol exists where rational parties are willing to disseminate their shares to other players. Their ran-

[5] Maleka et al. [92] consider rational secret sharing in the context of repeated games, and Nojoumian et al. [78] consider the repeated game setting from a socio-rational perspective where player reputation is important.

domized protocol is a modified game where players are distributed a *set* of shares, where only one share is correct. In each round k, players distribute their shares which evaluate to either the secret or a default value $\perp$. The solution relies on the fact that parties are unaware whether the current round k is terminal (k^*, allowing the secret to be recovered), or merely a "test" round $k \neq k^*$ (where the secret cannot be recovered, but players who do not distribute shares are caught as cheaters). By choosing k^* from a geometric distribution, as in Groce et al. [32], cheating players that choose strategy $\sigma =\perp$ when $k \neq k^*$ are caught and the game may be terminated. Thus, players now have an incentive to distribute their share, as playing $\perp$ only yields positive utility when $k = k^*$.

A candidate security definition should accept this probabilistic protocol for rational secret sharing as secure against rational agents. However, the strong restrictions on communication channels imposed by existing work preclude the above protocol from satisfying their security definitions, despite refinements considering the problem under standard communication models [17, 19, 22, 26]. That is, the rational secret sharing protocol of Halpern and Teague [23] assumes players have access to a non-rushing broadcast channel. This clearly violates the assumptions of models assuming physical primitives [64], and even fails to satisfy the weakest security definition that has been proposed: collusion-preserving computation [87]. Ideally, the original rational secret sharing protocol of Halpern and Teague should be demonstrably secure against rational agents under a general security framework. Our framework allows point-to-point communication in the ideal model, and thus is able to accurately model the original solution to rational secret sharing, which we demonstrate in Section 7.5.

7.3.2 Adversarial Model

Traditionally, an adversary $\mathcal{A}$ is viewed as a monolithic entity with a specified computational complexity and ability to "corrupt" players in a static or dynamic fashion. In our model, we consider all players to have the ability to act in an adversarial manner. Thus, rather than considering a monolithic adversary $\mathcal{A}$, we endow each player $\mathtt{P} \in \mathcal{P}$ with a local adversary $\mathcal{A}_{\mathtt{P}}$. The adversary is bound to the player's utility function $\mu_{\mathtt{P}}(\cdot)$ and selects actions for $\mathtt{P}$ in order to maximize $\mu_{\mathtt{P}}(\cdot)$. Note that as we bind player actions to a local adversary seeking to maximize a utility function, we cannot bound the number of players that deviate from the protocol. This is an unavoidable consequence of modeling players as rational agents; they select strategies to maximize a local utility function and follow the protocol only when doing so is advantageous. As cryptographic protocols typically require a number of rounds of interaction, we allow the rational players to update their strategy based on observations throughout the game Γ. Thus, we assume each local adversary is *mobile* [93], and may choose to deviate or follow the protocol at each round in a dynamic fashion. Additionally, players may choose probabilistic strategies[6], so we must introduce a random tape $r_{\mathtt{P}}$ for each player $\mathtt{P}$. Thus, each local adversary is adaptive, mobile, probabilistic, malicious, runs in *probabilistic polynomial-time* (PPT) and is presumed rational: bound to the player's local utility function.

Given the above definition of adversaries, the following actions are unavoidable:

- **Refusal to Participate:** Players may refuse to participate in the protocol.
- **Input Substitution:** Players may supply an input to the protocol different from their true input.
- **Premature Abort:** Players may abort the protocol prior to completion.

[6] In a game theoretic setting, such strategies are referred to as *mixed.*

- **Collusion:** Players may privately communicate over point-to-point communication channels, and collude to influence the protocol execution.

Constructions satisfying our definition thus assume that it is advantageous for players to engage in the protocol, and that this constitutes a utility maximization strategy with respect to their local utility function.

7.3.3 Ideal World Model

We now formalize the *ideal world* model, under which an ideal game specification Γ is constructed. We assume familiarity with standard game theoretic concepts in our exposition[7]. We first define the game specification of Γ under the *extensive form game* representation. In the game theoretic literature, *normal form game* representation is generally used for single round games where actions are played simultaneously. As cryptographic protocols typically proceed in a series of rounds where actions are played asynchronously, we prefer *extensive form game* representation, where the ideal game specification Γ is represented as a tree. At each node in the game tree, a subset $\mathtt{P} \subseteq \mathcal{P}$ of the players select and simultaneously play an action.

Definition 7.3.1 *An* ***extensive form game*** Γ *consists of:*

1. *A finite set $\mathcal{P} = \{P_i\}_{i=1}^n$ of players.*

2. *A (finite) set of sequences $\mathcal{H}$ called the* history*. The empty sequence $\emptyset$ is a member of $\mathcal{H}$. We let k denote the current decision node. If $(a^k)_{k=1,\ldots,K} \in \mathcal{H}$ and $L < K$ then $(a^k)_{k=1,\ldots,L} \in \mathcal{H}$. If an infinite sequence $(a^k)_{k=1}^{\infty}$ satisfies $(a^k)_{k=1,\ldots,L} \in \mathcal{H}$ for every positive integer L then $(a^k)_{k=1}^{\infty} \in \mathcal{H}$. A history*

[7] For a proper introduction to the subject, Katz [12] describes the current effort to combine game theoretic and cryptographic concepts, while Osborne et al. [13] and Fudenberg et al. [40] give a complete introduction to game theory.

$(a^k)_{k=1,\ldots,K} \in \mathcal{H}$ is a terminal history if it is infinite or if there is no a^{K+1} such that $(a^k)_{k=1,\ldots,K+1} \in \mathcal{H}$. The set of actions available after the nonterminal history h is denoted $A(h) = \{a : (h, a) \in \mathcal{H}\}$ and the set of terminal histories is denoted $\mathcal{Z}$. We let $\mathcal{H}^k$ denote the history through round k.

3. *A player function P that assigns to each nonterminal history (each member of $\mathcal{H}/\mathcal{Z}$) a member of $\mathcal{P} \cup \{nature\}$. When $P(h) = nature$, then nature determines the action taken after history h.*

4. *For each player $P_i \in \mathcal{P}$ a partition $\mathcal{I}_i$ of $\{h \in \mathcal{H} : P(h) = i\}$ with the property that $A(h) = A(h')$ whenever h and h' are in the same member of the partition. For $I_i \in \mathcal{I}_i$ we denote by $A(I_i)$ the set $A(h)$ and by $P(I_i)$ the player $P(h)$ for any $h \in I_i$. Thus, $\mathcal{I}_i$ is the information partition of player i, while the set $I_i \in \mathcal{I}_i$ is an information set of player i.*

5. *For each player $P_i \in \mathcal{P}$ a preference relation $\precsim_i$ on lotteries[8] over $\mathcal{Z}$ that can be represented as the expected value of a payoff function defined on $\mathcal{Z}$.*

Throughout, we replace the preference relation $\precsim_i$ by a *utility function* $\mu_i : A \to \mathbb{R}$, such that $\mu_i(a) \geq \mu_i(b)$ when $b \precsim_i a$.

We make the following modeling choices:

- **Extensive Form Games:** The ideal game specification Γ is described by a game tree in extensive form representation.

- **Imperfect Information:** A game specification is said to have *imperfect information* if players may have non-singleton information sets $I_i \in \mathcal{I}_i$. That is, at a given round in the game, players may be unaware of the move selected by

[8] Even if all actions are deterministic, moves by *nature* can induce a probability distribution over the set of terminal histories.

the previous player(s). Thus, their information set may contain more than one node in the game tree at any given round.

- **Local Simulators:** Each player $\mathtt{P}_i \in \mathcal{P}$ in the ideal model has a *local* simulator $\mathcal{S}_i$ that forces $\mathtt{P}$ to play those actions that maximize $\mu_i(\cdot)$, the utility function of player $\mathtt{P}_i$. Each simulator $\mathcal{S}_i$ has an associated adversary $\mathcal{A}_i$ in the real world execution model, denoted $\mathcal{S}_i = \mathsf{Sim}(\mathcal{A}_i)$.

- **Point-to-Point Communication Resources:** Each player pair $(\mathtt{P}_i, \mathtt{P}_j)_{i \neq j} \in \mathcal{P}$ has a secure point-to-point communication resource $\mathcal{R}_{ij}$.

As we consider all players to be rational agents, we model the ideal world protocol as a game specification Γ that aims to achieve an equilibrium. The ideal game specification is an interaction between a set of n players $\mathcal{P} = \{\mathtt{P}_i\}_{i=1}^{n}$, their local utility functions $\vec{\mu} = \{\mu_i\}_{i=1}^{n}$ and action sets A_i, which contains those actions playable by player $\mathtt{P}_i$. Frequently, a deterministic choice of an action $a \in A_i$ will not yield a Nash equilibrium. Thus, we allow players to choose a *strategy* σ_i: a probability distribution over A_i. The standard equilibrium concept in the rational cryptographic literature is a *computational* Nash equilibrium [15–19], given in Definition 3.1.3. Intuitively, no player $\mathtt{P}_i$ has an incentive to deviate from strategy σ_i given that every other player $\mathtt{P}_j$ selects their equilibrium strategy σ_j. The definition of a computational Nash equilibria adds a negligible term $\mathit{negl}(\lambda)$ with respect to a security parameter λ. This is necessary in the computational setting, as security rests on the premise that breaking cryptographic primitives occurs with only negligible probability. Thus, this notion must be incorporated into the equilibrium definition. Although computational Nash equilibria are the weakest of the equilibrium concepts described in the rational cryptographic literature, preserving only computational Nash equilibria in our framework is sufficient for extensions to more powerful equilibrium concepts.

The standard ideal world model has players interact with an incorruptible trusted third party (TTP) that accepts player inputs, computes the ideal functionality f, and distributes the output to players. In the setting of rational cryptography, we will consider a `Mediator` that enforces the ideal game specification.

Input Distribution: Each player $\mathtt{P}_i \in \mathcal{P}$ receives its input x_i, random coins r_i and auxiliary input[a] z_i. Each player has the option of inputting a different input $\bar{x}_i \neq x_i$, as this is unavoidable.

Game Execution: The `Mediator` allows the subset of players $\mathtt{P} \subseteq \mathcal{P}$ specified at each node of the game specification Γ to simultaneously play their actions. Note that games where only a single player moves at each node (asynchronous play) are fully supported, as this is modeled by setting the subset $\mathtt{P} = \{\mathtt{P}_i\}$.

Payoff Assignment: If the current node k is terminal (i.e., $k \in \mathcal{Z}$), then the `Mediator` distributes the payoffs associated with k to all players $\mathtt{P}_i \in \mathcal{P}$.

[a] An auxiliary input is provided to all players to model additional information available to them [5].

Protocol 7.3.1: Ideal World Game Execution

Definition 7.3.2 *Let Γ represent the ideal game specification in extensive form representation, $\mathcal{R}$ a point-to-point communication resource available between all pairs of players in $\mathcal{P}$, $\mathcal{S}$ the set of local simulators, $\vec{\mu}$ the set of player utility functions and z any auxiliary information provided to a player. We denote by $\vec{\bar{x}}$ the set of inputs for players (which may differ from the set of their true inputs $\vec{x}$) and by r the random coins provided to a player. We then define the i^{th} output of an* `ideal world execution` *for players $\mathcal{P}$ in the presence of local simulators $\mathcal{S}$ as:*

$$\left\{\mathrm{IDEAL}^{(i\in[1\ldots n])}_{\Gamma,\mathcal{R},\mathcal{P},\mathcal{S},\vec{\mu},z}(\lambda,\vec{\bar{x}};r)\right\}_{\lambda\in\mathbb{N},\vec{\bar{x}},r\in\{0,1\}^*} \triangleq \{\vec{\sigma^*},\mathcal{I}\} \tag{7.1}$$

where $\vec{\sigma^*}$ *is the equilibrium in the ideal game specification* Γ*,* $\mathcal{S} = \{\mathcal{S}_i\}_{i \in [1 \ldots n]}$ *is the set of simulators such that* $\mathcal{S}_i = \mathsf{Sim}(\mathcal{A}_i)$*,* $\mathcal{I}$ *is the information partition set for* $\mathcal{P}$*,* $|\bar{x}_i| = |\bar{x}_j| \forall i \neq j$ *and* $|z| = \text{poly}(|\bar{x}_i|)$*.*

This ideal world model necessarily limits the class of games that may be realized, as any game specification that disallows point-to-point communication channels between all parties cannot be modeled in the presence of $\mathcal{R}$. However, we will demonstrate that a broad class of games that initially appear inadmissible under our model are realizable through minor modifications to the game specification, and which preserve the equilibria of the original game.

7.3.4 Real World Model

We now introduce the real world model protocol Π that implements the ideal game specification Γ. In order to translate ideal game specifications into realizable protocols, we assume the existence of a public key infrastructure (PKI) in the real world model. That is, we must translate the ideal world point-to-point communication resource $\mathcal{R}$ into an implementation allowing point-to-point private communication between all players $\mathtt{P}_i, \mathtt{P}_j \in \mathcal{P}$ during the execution of Π. We denote the real world PKI communication resource by $\mathcal{C}$, where $\forall (\mathtt{P}_i, \mathtt{P}_j)_{i \neq j} \in \mathcal{P}, \exists \mathcal{C}_{ij} \in \mathcal{C}$.

In the real world execution, each player $\mathtt{P}_i$ has an associated local adversary $\mathcal{A}_i$, rather than a simulator $\mathcal{S}_i$ as in the ideal world game. The local adversary $\mathcal{A}_i$ selects the actions of $\mathtt{P}_i$ to maximize the player's local utility function μ_i. Similarly, in the real world execution there is no `Mediator`, as the goal is to remove reliance on trusted third parties.

Input Distribution: Each player $\mathtt{P}_i \in \mathcal{P}$ receives its input x_i, random coins r_i and auxiliary input z_i. Each player has the option of inputting a different input $\bar{x_i} \neq x_i$, as this is unavoidable.

Protocol Execution: The execution of Π proceeds in a series of rounds, where at each round a subset of players $\mathtt{P} \subseteq \mathcal{P}$ specified at each node play their actions. Each player pair $(\mathtt{P}_i, \mathtt{P}_j)_{i \neq j} \in \mathcal{P}$ is connected by a private authenticated point-to-point communication channel $\mathcal{C}_{ij}$, and may exchange messages throughout the protocol execution.

Payoff Assignment: If the current node k is terminal (i.e., $k \in \mathcal{Z}$), then each player $\mathtt{P}_i \in \mathcal{P}$ receives its associated payoff.

Protocol 7.3.2: Real World Protocol Execution

Definition 7.3.3 *Let* Π *represent the real world protocol implementing* Π*,* $\mathcal{C}$ *a point-to-point authenticated and private PKI communication resource available between all pairs of players in* $\mathcal{P}$*,* $\mathcal{A}$ *the set of local adversaries,* $\vec{\mu}$ *the set of player utility functions and* z *any auxiliary information provided to a player. We denote by* $\vec{\bar{x}}$ *the set of inputs for players (which may differ from the set of their true inputs* $\vec{x}$*) and by* r *the random coins provided to a player. We then define the* i^{th} *output of a* `real world execution` *for players* $\mathcal{P}$ *in the presence of local adversaries* $\mathcal{A}$ *as:*

$$\left\{\mathrm{REAL}^{(i\in[1\ldots n])}_{\Pi,\mathcal{C},\mathcal{P},\mathcal{A},\vec{\mu},z}(\lambda,\vec{\bar{x}};r)\right\}_{\lambda\in\mathbb{N},\vec{\bar{x}},r\in\{0,1\}^*} \triangleq \{\vec{\sigma^*},\mathcal{I}\} \tag{7.2}$$

where $\vec{\sigma^*}$ *is the equilibrium in the real world protocol* Π*,* $\mathcal{I}$ *is the information partition set for* $\mathcal{P}$*,* $|\bar{x}_i| = |\bar{x}_j| \forall i \neq j$ *and* $|z| = \mathrm{poly}(|\bar{x}_i|)$*.*

7.3.5 Establishing the Security of Realized Protocols

The security of protocols is established by demonstrating that the real and ideal world distribution ensembles are computationally indistinguishable[9]. This guarantees that any attack available to an adversary $\mathcal{A}$ in the real model is also available to a simulator $\mathcal{S}$ in the ideal model.

Definition 7.3.4 *(Security against Rational Adversaries) Let* Γ *be an* n*-player ideal game specification and* Π *be an* n*-party real world protocol. We say that* Π *securely realizes* Γ *if there exists a set* $\{\mathsf{Sim}_i\}_{i\in[1\ldots n]}$ *of PPT transformations admissible in the ideal model such that for all PPT rational adversaries* $\mathcal{A} = \{\mathcal{A}_i\}_{i\in[1\ldots n]}$ *admissible in the real model, for all* $\vec{x} \in (\{0,1\}^*)^n$ *and* $\vec{z} \in (\{0,1\}^*)^n$*, and for all* $i \in [1\ldots n]$*,*

$$\left\{\mathrm{IDEAL}^{(i\in[1\ldots n])}_{\Gamma,\mathcal{R},\mathcal{P},\mathcal{S},\vec{\mu},z}(\lambda,\vec{\vec{x}};r)\right\}_{\lambda\in\mathbb{N},\vec{\vec{x}},r\in\{0,1\}^*} \overset{c}{\equiv} \left\{\mathrm{REAL}^{(i\in[1\ldots n])}_{\Pi,\mathcal{C},\mathcal{P},\mathcal{A},\vec{\mu},z}(\lambda,\vec{\vec{x}};r)\right\}_{\lambda\in\mathbb{N},\vec{\vec{x}},r\in\{0,1\}^*}$$

where $\mathcal{S} = \{\mathcal{S}_i\}_{i\in[1\ldots n]}$ *is the set of simulators such that* $\mathcal{S}_i = \mathsf{Sim}(\mathcal{A}_i)$*,* $\mathcal{I}$ *is the information partition set for* $\mathcal{P}$ *and* r *is chosen uniformly at random.*

Thus, to establish the security of a realized protocol Π, we must construct a simulator $\mathcal{S}_i$ for all players $\mathsf{P}_i \in \mathcal{P}$ such that for all probabilistic polynomial-time distinguishers $\mathcal{D}$, the distributions of $\mathcal{S}$ in the ideal world and $\mathcal{A}$ in the real world can only be differentiated with probability negligibly greater than $\frac{1}{2}$.

7.4 Demonstrating the Model on the Prisoner's Dilemma

To illustrate the power of our model, we return to our running example of the prisoner's dilemma. We demonstrate that, despite the requirement of physical sep-

[9] That is, any probabilistic polynomial-time (PPT) distinguisher $\mathcal{D}$ cannot distinguish between an execution of Γ in the ideal world model and an execution of Π in the real world model with probability non-negligibly greater than $\frac{1}{2}$.

aration (and, consequently, lack of communication between players) in the original game specification, we are able to construct a modified game specification that is admissible in the ideal world model, and realizable in the real world model under our security definition.

7.4.1 Ideal World Game Specification

The ideal world game Γ is an interaction between a set of players $\mathcal{P}$ such that $\mathcal{P} = \{\mathtt{A}, \mathtt{B}, \mathtt{Authority}\}$, where $\mathtt{A}$ (resp. $\mathtt{B}$) has access *only* to a communication resource $\mathcal{R}_{\mathtt{A},\mathtt{Authority}}$ (resp. $\mathcal{R}_{\mathtt{A},\mathtt{Authority}}$). That is, $\mathtt{A}$ and $\mathtt{B}$ are physically separated and, thus, unable to communicate. In the original game Γ, the strategy $\vec{\sigma^*} = \{\sigma^*_{\mathtt{A}} = \mathsf{confess}, \sigma^*_{\mathtt{B}} = \mathsf{confess}\}$ is the sole Nash equilibrium. However, consider a modified game specification $\bar{\Gamma}$ where there exists a communication resource $\mathcal{R}_{\mathtt{A},\mathtt{B}}$ enabling $\mathtt{A}$ and $\mathtt{B}$ to communicate. We now demonstrate that $\bar{\Gamma}$ is admissible in our ideal world definition.

Let $\bar{\Gamma}$ be an ideal game specification, with player set $\mathcal{P} = \{\mathtt{A}, \mathtt{B}, \mathtt{Authority}\}$ and associated set of local simulators $\mathcal{C} = \{\mathcal{C}_{\mathtt{A}}, \mathcal{C}_{\mathtt{B}}, \mathcal{C}_{\mathtt{Authority}}\}$ that select actions for players to maximize their local utility functions. We define the resource set $\mathcal{R}$ as $\mathcal{R} = \{\mathcal{R}_{\mathtt{A},\mathtt{Authority}}, \mathcal{R}_{\mathtt{B},\mathtt{Authority}}, \mathcal{R}_{\mathtt{A},\mathtt{B}}\}$, and players $\mathtt{A}$ and $\mathtt{B}$ have identical utility functions defined as follows:

$$\mu_i(\sigma_i, \sigma_j) \mapsto \begin{cases} -1 & : \quad \sigma_i = \mathsf{silent}, \sigma_j = \mathsf{silent} \\ 0 & : \quad \sigma_i = \mathsf{silent}, \sigma_j = \mathsf{confess} \\ -3 & : \quad \sigma_i = \mathsf{confess}, \sigma_j = \mathsf{silent} \\ -2 & : \quad \sigma_i = \mathsf{confess}, \sigma_j = \mathsf{confess} \end{cases} \tag{7.3}$$

Clearly $\bar{\Gamma}$ is admissible under the ideal world model, as all players have access to a point-to-point communication resource $\mathcal{R}$. The modified ideal world game $\bar{\Gamma}$ of the prisoner's dilemma proceeds as follows:

Input Distribution: Each player $\mathtt{P}_i \in \mathcal{P}$ receives its input x_i, random coins r_i and auxiliary input z_i. Each player has the option of inputting a different input $\bar{x}_i \neq x_i$ or aborting the protocol at any time, as this is unavoidable.

"Cheap Talk": Players A and B are free to communicate over $\mathcal{R}_{\mathtt{A},\mathtt{B}}$, and each may try to convince the other that they will set $\sigma = \mathsf{silent}$. However, as $\mathcal{R}_{\mathtt{A},\mathtt{Authority}}$ (resp. $\mathcal{R}_{\mathtt{B},\mathtt{Authority}}$) is private, neither is able to observe the message sent to Authority. Thus, this communication is considered "cheap talk", in that it *does not affect* the strategy selection of the player. The local simulator $\mathcal{S}_i$ for each player selects $m_i = \mathsf{confess}$, as this maximizes μ_i.

Game Execution: The Mediator instructs A and B to send a message m to Authority with their decision, where the message $m \in \{\mathsf{silent}, \mathsf{confess}\}$.

Payoff Assignment: After Authority has received $m_{\mathtt{A}}$ and $m_{\mathtt{B}}$, Mediator distributes the payoffs to A and B.

Protocol 7.4.1: Ideal World Game Execution

It is not difficult to see that the equilibrium in the modified ideal game specification $\bar{\Gamma}$ is identical to the equilibrium in the original game specification Γ. That is, despite the presence of a point-to-point communication channel $\mathcal{R}_{\mathtt{A},\mathtt{B}}$, we achieve the desired equilibrium of $\vec{\sigma^*} = \{\sigma_{\mathtt{A}} = \mathsf{confess}, \sigma_{\mathtt{B}} = \mathsf{confess}\}$.

7.4.2 Real World Protocol Construction

Input Distribution: Each player $\mathtt{P}_i \in \mathcal{P}$ receives its input x_i, random coins r_i and auxiliary input z_i. Each player has the option of inputting a different input $\bar{x_i} \neq x_i$ or aborting at any time, as this is unavoidable. The payoff for abort is equivalent to silent, and other players may continue the protocol execution in the presence of aborts.

"Cheap Talk": Players $\mathtt{A}$ and $\mathtt{B}$ are free to communicate over $\mathcal{C}_{\mathtt{A,B}}$, and each may try to convince the other that they will set $\sigma = \mathsf{silent}$. However, as $\mathcal{C}_{\mathtt{A,Authority}}$ (resp. $\mathcal{C}_{\mathtt{B,Authority}}$) is private, neither is able to observe the message sent to $\mathtt{Authority}$. Thus, this communication is considered "cheap talk", in that it *does not affect* the strategy selection of the player. The local adversary $\mathcal{A}_i$ for each player selects $m_i = \mathsf{confess}$ as this maximizes μ_i.

Game Execution: $\mathtt{A}$ and $\mathtt{B}$ send a message m to $\mathtt{Authority}$ with their decision, where $m \in \{\mathsf{silent}, \mathsf{confess}\}$. Although aborting (setting $m =\perp$) is an option, it is equivalent to setting $m = \mathsf{silent}$. As players are controlled by an adversary seeking to maximize their utility function μ as defined by Equation 7.3, this strategy is never played; setting $m = \mathsf{silent}$ is strictly dominated by setting $m = \mathsf{confess}$.

Payoff Assignment: After $\mathtt{Authority}$ has received $m_{\mathtt{A}}$ and $m_{\mathtt{B}}$, $\mathtt{A}$ and $\mathtt{B}$ receive the payoffs specified by their local utility functions as defined in Equation 7.3.

Protocol 7.4.2: Real World Protocol Execution

We now translate the ideal game specification $\bar{\Gamma}$ to a real world protocol Π, and demonstrate that there exist simulators such that the distribution of the ideal world game is computationally indistinguishable from the distribution of the real world protocol execution.

In the real world model, the communication resource $\mathcal{R}$ is replaced with a public key infrastructure $\mathcal{C}$. Each pair of players $(\mathtt{P}_i, \mathtt{P}_j) \in \mathcal{P}$ has access to a private and authenticated point-to-point communication channel $\mathcal{C}_{ij}$. Let Π be a real world protocol, with player set $\mathcal{P} = \{\mathtt{A}, \mathtt{B}, \mathtt{Authority}\}$ and associated set of local adversaries

$\mathcal{A} = \{\mathcal{A}_{\texttt{A}}, \mathcal{A}_{\texttt{B}}, \mathcal{A}_{\texttt{Authority}}\}$ that select actions for players to maximize their local utility functions, communication channel set $\mathcal{C} = \{\mathcal{C}_{\texttt{A,Authority}}, \mathcal{C}_{\texttt{B,Authority}}, \mathcal{C}_{\texttt{A,B}}\}$, and players $\texttt{A}$ and $\texttt{B}$ have identical utility functions defined as in Equation 7.3.

Clearly Π is admissible under the real world model, as the PKI infrastructure $\mathcal{C}$ facilitates the point-to-point communication channels between all players. The real world protocol Π of the prisoner's dilemma proceeds as in Protocol 7.4.2. Again, the original equilibrium of $\vec{\sigma^*} = \{\sigma_{\texttt{A}} = \textsf{confess}, \sigma_{\texttt{B}} = \textsf{confess}\}$ is preserved despite the presence of the communication channel $\mathcal{C}$.

7.4.3 Demonstrating Protocol Π Security

We use the simulation paradigm [5] to demonstrate the security of the construction by proving the distribution of the real world protocol is computationally indistinguishable from the ideal world distribution.

Theorem 7.4.1 *(Security of Π against Rational Adversaries) Let $\bar{\Gamma}$ be the n-party ideal world game specification of Protocol 7.4.1 and let Π be the n-party real world execution of Protocol 7.4.2. There exists a set $\{\textsf{Sim}_i\}_{i\in[1\ldots n]}$ of PPT transformations admissible in the ideal model such that for all PPT rational adversaries $\mathcal{A} = \{\mathcal{A}_1, \ldots, \mathcal{A}_n\}$ admissible in the real model, for all $\vec{x} \in (\{0,1\}^*)^n$ and $\vec{z} \in (\{0,1\}^*)^n$, and for all $i \in [1 \ldots n]$,*

$$\left\{\mathrm{IDEAL}^{(i\in[1\ldots n])}_{\bar{\Gamma},\mathcal{R},\mathcal{P},\mathcal{S},\vec{\mu},z}(\lambda, \vec{\bar{x}}; r)\right\}_{\lambda\in\mathbb{N},\vec{\bar{x}},r\in\{0,1\}^*} \overset{c}{\equiv} \left\{\mathrm{REAL}^{(i\in[1\ldots n])}_{\Pi,\mathcal{C},\mathcal{P},\mathcal{A},\vec{\mu},z}(\lambda, \vec{\bar{x}}; r)\right\}_{\lambda\in\mathbb{N},\vec{\bar{x}},r\in\{0,1\}^*}$$

establishing that Π securely realizes $\bar{\Gamma}$.

Proof To prove the security of Π against the set of rational adversaries $\mathcal{A} = \{\mathcal{A}_{\texttt{A}}, \mathcal{A}_{\texttt{B}}, \mathcal{A}_{\texttt{Authority}}\}$ we must construct a set of simulators $\mathcal{S} = \{\mathcal{S}_{\texttt{A}}, \mathcal{S}_{\texttt{B}}, \mathcal{S}_{\texttt{Authority}}\}$

whose output in the ideal game specification $\bar{\Gamma}$ is indistinguishable from the output of $\mathcal{A}$ in the real world execution. To achieve this, we construct simulators $\mathcal{S}_i = \mathsf{Sim}(\mathcal{A}_i)$ that simulate all messages and the output of $\mathcal{A}_i$ in the real world execution of Π, and is thus able to return these as its own. The simulated messages and output returned by $\mathcal{S}_i$ must be computationally indistinguishable such that, for all probabilistic polynomial-time distinguishers $\mathcal{D}$, the probability of differentiating the ideal world and real world distributions is at most negligibly greater than $\frac{1}{2}$.

Each simulator $\mathcal{S}_i$ will rely on the private communication resource $\mathcal{R}$ to simulate the messages exchanged and final output produced by $\mathcal{A}_i$ acting to maximize the utility function μ_i for player $\mathtt{P}_i$. The simulator $\mathcal{S}_i$ given in Construction 7.4.1 holds for all players $\mathcal{P} = \{\mathtt{A}, \mathtt{B}, \mathtt{Authority}\}$.

The construction relies on the computational indistinguishability of the real world communication channel $\mathcal{C}$ from the ideal world private and authenticated communication resource $\mathcal{R}$. All messages sent by simulators $\mathcal{S}_i \in \mathcal{S}$ in the ideal world model are passed over $\mathcal{R}$. In the real world execution, messages are encrypted between players using the PKI communication resource $\mathcal{C}$. Thus, all probabilistic polynomial-time distinguishers $\mathcal{D}$ are able to distinguish the view of the ideal world execution from the real world execution with at most probability negligibly greater than $\frac{1}{2}$ by the security of the PKI communication resource $\mathcal{C}$. $\blacksquare$

Input Distribution: The simulator $\mathcal{S}_i \in \mathcal{S}$ is given input x_i, random coins r_i and auxiliary input z_i

"Cheap Talk": The simulator $\mathcal{S}_i$ is free to communicate over $\mathcal{R}_{\mathcal{S}_i,\mathcal{S}_j}$ where $i \neq j$. If $i, j \in \{\texttt{A}, \texttt{B}\}$, it must simulate the "cheap talk" between the other player's adversary $\mathcal{A}_j$. $\mathcal{S}_i$ uses its random coins r_i to construct a random message m, and sends m over resource $\mathcal{R}_{\mathcal{S}_i,\mathcal{S}_j}$. By definition, $\mathcal{R}$ is a private and authenticated point-to-point communication resource. Thus, the messages sent by the simulator are computationally indistinguishable from those sent in the real world execution, which are encrypted under the public key infrastructure communication resource $\mathcal{C}$. The local simulator $\mathcal{S}_i$ for each player selects $m_i = \mathsf{confess}$, as this maximizes μ_i regardless of the messages exchanged during this phase.

Game Execution: The simulator $\mathcal{S}_i$ sends a message m to $\mathcal{S}_{\texttt{Authority}}$ over $\mathcal{R}_{\mathcal{S}_i,\mathcal{S}_{\texttt{Authority}}}$ with their decision, where the message $m \in \{\mathsf{silent}, \mathsf{confess}\}$. By definition, $\mathcal{R}$ is a private and authenticated point-to-point communication resource. Thus, the messages sent by the simulator to $\mathcal{S}_{\texttt{Authority}}$ are computationally indistinguishable from those sent in the real world execution, which are encrypted under the public key infrastructure communication resource $\mathcal{C}$.

Payoff Assignment: After $\mathcal{S}_{\texttt{Authority}}$ has received $m_{\mathcal{S}_i}$ and $m_{\mathcal{S}_j}$, each simulator receives the payoff associated with the outcome.

Construction 7.4.1: Construction of Simulator $\mathcal{S}_i$

7.5 Demonstrating the Model on Rational Secret Sharing

To illustrate the power of our model, we return to the example of rational secret sharing. We demonstrate that, despite the presence of point-to-point communication channels, the original game specification is admissible in our ideal world model, and realizable in the real world model. This violates the assumptions of existing security frameworks, which disallow point-to-point communication either within the protocol execution, outside of the protocol execution, or both.

7.5.1 Ideal World Game Specification

The ideal world game Γ is an interaction between a set of players $\mathcal{P} = \{\mathtt{P}_i\}_{i \in [1 \ldots n]}$, where $\mathtt{P}_i$ has access to a point-to-point communication resource $\mathcal{R}_{\mathtt{P}_i, \mathtt{P}_j} \forall j \neq i$. That is, $\mathtt{P}_i$ may privately communicate with any other player $\mathtt{P}_j$. We now demonstrate that Γ is admissible in our ideal world definition.

Input Distribution: Each player $\mathtt{P}_i \in \mathcal{P}$ receives its input share x_i, random coins r_i and auxiliary input z_i. Each player has the option of inputting a different share $\bar{x_i} \neq x_i$ or aborting the protocol at any time, as this is unavoidable.

"Cheap Talk": Player $\mathtt{P}_i$ is free to collaborate with all players $\mathtt{P}_j \in \hat{\mathcal{P}}$ over $\mathcal{R}_{\mathtt{P}_i, \mathtt{P}_j}$, where $\hat{\mathcal{P}}$ is the set of colluding players. Proposition 7.5.1 demonstrates that communication over $\mathcal{R}$ is considered "cheap talk" (it *does not affect* the strategy selection of the player), and that the local simulator $\mathcal{S}_i$ for each player will select $a_i = \mathsf{reveal}$, as this maximizes μ_i.

Game Execution: The $\texttt{Mediator}$ instructs $\mathtt{P}_i, \forall i \in n$ to play their action a_i at each round k, where $a_i \in \{\mathsf{silent}^a, \mathsf{reveal}\}$.

Payoff Assignment: At the terminal round k^* where the shares yield the secret, $\texttt{Mediator}$ distributes the payoffs to $\mathtt{P}_i \in \mathcal{P}$.

[a] Note that selecting $a_i = \mathsf{silent}$ is equivalent to aborting.

Protocol 7.5.1: Ideal World Game Γ Execution

Let Γ be the ideal game specification for rational secret sharing, with player set $\mathcal{P} = \{\mathtt{P}_i\}_{i \in [1 \ldots n]}$ and associated set of local simulators $\mathcal{S} = \{\mathcal{S}_i\}_{i \in [1 \ldots n]}$ that select actions for players to maximize their local utility functions, resource set $\mathcal{R} = \{\mathcal{R}_{\mathtt{P}_i, \mathtt{P}_j}\}_{\forall i, i \neq j}$, and all players $\mathtt{P}_i \in \mathcal{P}$ have utility functions defined as

$$\mu_i(\sigma_i) \mapsto \begin{cases} (\mu^{++})(p) & : \quad \sigma_i = \mathsf{silent}, k = k^* \\ (\mu^{-})(1-p) & : \quad \sigma_i = \mathsf{silent}, k \neq k^* \\ (\mu^{+}) & : \quad \sigma_i = \mathsf{reveal} \end{cases} \tag{7.4}$$

where μ^{+} represents positive utility, μ^{-} represents negative utility, and $\mu^{++} > \mu^{+}$ as players value exclusivity.

Proposition 7.5.1 *For all players $P_i \in \mathcal{P}$ in Γ with utility function defined as $\mu_i(\sigma_i)$ in Equation 7.4, strategy $\{\sigma^*_{P_i} = \textsf{reveal}\}_{\forall i \in n} > \{\sigma_{P_i} = \textsf{silent}\}_{\forall i \in n}$ when $p < \frac{\mu^{+}}{\mu^{++}}$.*

Proof In the original rational secret sharing protocol, the strategy $\vec{\sigma^*} = \{\sigma^*_{\mathsf{P}_i} = \textsf{reveal}\}_{\forall i \in n}$ is the only Nash equilibrium, as the true final round k^* (where combining shares reveals the shared secret) is chosen from a geometric distribution. As the probability of correctly guessing the final round k^* is the parameter p, the expected utility for $\sigma_{\mathsf{P}_i} = \textsf{silent}$ is at most $(\mu^{++})(p)$. We set $\mu^{++} > \mu^{+}$, as players are assumed to value exclusivity (recovering the secret while preventing other players from doing so). If a player remains silent in any round $k < k^*$, they are caught by the other players as a cheater and excluded from future rounds (receiving negative utility μ^{-}). By choosing p such that $p < \frac{\mu^{+}}{\mu^{++}}$, we have $(\mu^{++})(p) < \mu^{+}$ which implies $\mu_{\mathsf{P}_i}(\textsf{silent}) < \mu_{\mathsf{P}_i}(\textsf{reveal})$. Thus revealing the share for each round strictly dominates remaining silent. Players in our ideal model Γ may communicate over $\mathcal{R}$ and attempt to convince other players that they will select $\textsf{silent}$. This provides a greater degree of exclusivity, as only those colluding players in $\hat{\mathcal{P}} \subseteq \mathcal{P}$ will recover the secret. However, this communication is considered cheap talk, as each player maximizes μ_i by selecting $\sigma_i = \textsf{reveal}$ regardless of the messages sent over $\mathcal{R}$ when $p < \frac{\mu^{+}}{\mu^{++}}$. ■

7.5.2 Real World Protocol Construction

We now translate the ideal game specification Γ to a real world protocol Π, and demonstrate that there exist simulators such that the distribution of the ideal world game is computationally indistinguishable from the distribution of the real world protocol execution.

In the real world model, the communication resource $\mathcal{R}$ is replaced with a public key infrastructure $\mathcal{C}$. Each pair of players $(\mathsf{P}_i, \mathsf{P}_j) \in \mathcal{P}$ has access to a private and authenticated point-to-point communication channel $\mathcal{C}_{ij}$. Let Π be a real world protocol, with player set $\mathcal{P} = \{\mathsf{P}_i\}_{i \in [1 \ldots n]}$ and associated set of local adversaries $\mathcal{A} = \{\mathcal{A}_i\}_{i \in [1 \ldots n]}$ that select actions for players to maximize their local utility functions, communication channel set $\mathcal{C} = \{\mathcal{C}_{ij}\}_{\forall i \neq j}$, and all players have identical utility functions defined as in Equation 7.4.

Input Distribution: Each player $\mathsf{P}_i \in \mathcal{P}$ receives its input share x_i, random coins r_i and auxiliary input z_i. Each player has the option of inputting a different share $\bar{x}_i \neq x_i$ or aborting the protocol at any time, as this is unavoidable.

"Cheap Talk": Player P_i is free to collaborate with all players $\mathsf{P}_j \in \hat{\mathcal{P}}$ over $\mathcal{C}_{\mathsf{P}_i,\mathsf{P}_j}$, where $\hat{\mathcal{P}}$ is the set of colluding players. Proposition 7.5.1 demonstrates that communication over $\mathcal{C}$ is considered "cheap talk" (it *does not affect* the strategy selection of the player), and that the local adversary $\mathcal{A}_i$ for each player selects $a_i = \mathsf{reveal}$, as this maximizes μ_i.

Game Execution: Each player $\mathsf{P}_i \in \mathcal{P}$ selects and plays their action a_i at each round k, where $a_i \in \{\mathsf{silent}$[a]$, \mathsf{reveal}\}$.

Payoff Assignment: At the terminal round k^* where the shares yield the secret, each player $\mathsf{P}_i \in \mathcal{P}$ receives its associated payoff.

[a] Note that selecting $a_i = \mathsf{silent}$ is equivalent to aborting.

Protocol 7.5.2: Real World Protocol Π Execution

Clearly Π is admissible under the real world model, as the PKI infrastructure $\mathcal{C}$ facilitates the point-to-point communication channels between all players. The real world protocol Π for rational secret sharing proceeds as in Protocol 7.5.2. Again, the original equilibrium of $\vec{\sigma^*} = \{\sigma_{\mathsf{P}_i} = \mathsf{reveal}\}$ is preserved despite the presence of the communication channel $\mathcal{C}$.

7.5.3 Demonstrating Protocol Π Security

We use the simulation paradigm [5] to demonstrate the security of the construction by proving the distribution of the real world protocol is computationally indistinguishable from the ideal world distribution.

Theorem 7.5.1 *(Security of* Π *against Rational Adversaries) Let* Γ *be the* n*-party ideal world game specification of Protocol 7.5.1 and let* Π *be the* n*-party real world execution of Protocol 7.5.2. There exists a set* $\{\mathsf{Sim}_i\}_{i\in[1\ldots n]}$ *of PPT transformations admissible in the ideal model such that for all PPT rational adversaries* $\mathcal{A} = \{\mathcal{A}_i\}_{i\in[1\ldots n]}$ *admissible in the real model, for all* $\vec{x} \in (\{0,1\}^*)^n$ *and* $\vec{z} \in (\{0,1\}^*)^n$*, and for all* $i \in [1\ldots n]$*,*

$$\left\{\mathrm{IDEAL}^{(i\in[1\ldots n])}_{\Gamma,\mathcal{R},\mathcal{P},\mathcal{S},\vec{\mu},z}(\lambda,\vec{\vec{x}};r)\right\}_{\lambda\in\mathbb{N},\vec{\vec{x}},r\in\{0,1\}^*} \overset{c}{\equiv} \left\{\mathrm{REAL}^{(i\in[1\ldots n])}_{\Pi,\mathcal{C},\mathcal{P},\mathcal{A},\vec{\mu},z}(\lambda,\vec{\vec{x}};r)\right\}_{\lambda\in\mathbb{N},\vec{\vec{x}},r\in\{0,1\}^*}$$

establishing that Π *securely realizes* Γ*.*

Proof To prove the security of Π against rational adversaries $\mathcal{A} = \{\mathcal{A}_i\}_{i\in[1\ldots n]}$ we must construct a set of simulators $\mathcal{S} = \{\mathcal{S}_i\}_{i\in[1\ldots n]}$ whose output in the ideal game specification Γ is indistinguishable from the output of $\mathcal{A}$ in the real world execution.

Input Distribution: The simulator $\mathcal{S}_i \in \mathcal{S}$ is given input share x_i, random coins r_i and auxiliary input z_i

"Cheap Talk": The simulator $\mathcal{S}_i$ is free to communicate over $\mathcal{R}_{\mathcal{S}_i,\mathcal{S}_j}$ where $i \neq j$. $\mathcal{S}_i, \forall i \neq j$ must simulate the "cheap talk" between the other player's adversary $\mathcal{A}_j$. $\mathcal{S}_i$ uses its random coins r_i to construct a random message m, and sends m over resource $\mathcal{R}_{\mathcal{S}_i,\mathcal{S}_j}$. By definition, $\mathcal{R}$ is a private and authenticated point-to-point communication resource. Thus, the messages sent by the simulator are computationally indistinguishable from those sent in the real world execution, which are encrypted under the public key infrastructure communication resource $\mathcal{C}$. The local simulator $\mathcal{S}_i$ for each player selects $m_i = \mathsf{reveal}$, as this maximizes μ_i regardless of the messages exchanged during this phase.

Game Execution: The simulator $\mathcal{S}_i$ sends a message m to $\mathcal{S}_j, \forall j \neq i$ over $\mathcal{R}_{\mathcal{S}_i,\mathcal{S}_j}$ with their decision, where $m \in \{\mathsf{silent}, \mathsf{reveal}\}$. By definition, $\mathcal{R}$ is a private and authenticated point-to-point communication resource. Thus, the messages sent by the simulator to $\mathcal{S}_j$ are computationally indistinguishable from those sent in the real world execution, which are encrypted under the public key infrastructure communication resource $\mathcal{C}$.

Payoff Assignment: After $\mathsf{P}_j \in \mathcal{P}, \forall j \neq i$ has received $m_{\mathcal{S}_i}$, each simulator receives the payoff associated with the outcome.

Construction 7.5.1: Construction of Simulator $\mathcal{S}_i$

To achieve this, we construct simulators $\mathcal{S}_i = \mathsf{Sim}(\mathcal{A}_i)$ that simulate all messages and the output of $\mathcal{A}_i$ in the real world execution of Π, and is thus able to return these as its own. The simulated messages and output returned by $\mathcal{S}_i$ must be computationally indistinguishable such that, for all probabilistic polynomial-time distinguishers $\mathcal{D}$, the probability of differentiating the ideal world and real world distributions is at most negligibly greater than $\frac{1}{2}$.

Each simulator $\mathcal{S}_i$ will rely on the private communication resource $\mathcal{R}$ to simulate the messages exchanged and final output produced by $\mathcal{A}_i$ acting to maximize the utility function μ_i for player $\mathtt{P}_i$. The simulator $\mathcal{S}_i$ given in Construction 7.5.1 holds for all players $\mathcal{P} = \{\mathtt{P}_i\}_{i\in[1\ldots n]}$.

The construction relies on the computational indistinguishability of the real world communication channel $\mathcal{C}$ from the ideal world private and authenticated communication resource $\mathcal{R}$. All messages sent by simulators $\mathcal{S}_i \in \mathcal{S}$ in the ideal world model are passed over $\mathcal{R}$. In the real world execution, messages are encrypted between players using the PKI communication resource $\mathcal{C}$. Thus, all probabilistic polynomial-time distinguishers $\mathcal{D}$ are able to distinguish the view of the ideal world execution from the real world execution with at most probability negligibly greater than $\frac{1}{2}$ by the security of the PKI communication resource $\mathcal{C}$. ■

7.6 Conclusion

In this chapter, we have proposed a security definition capturing rational cryptographic protocols in the presence of standard point-to-point communication resources. Rather than limit the communication resources available to players, we answer the question of how game specifications admissible in an ideal model allowing point-to-point communication channels may be realized in practice. Thus, the ideal world model necessarily limits the class of games that are admissible and is not a general result. However, we have argued that point-to-point communication channels are unavoidable in real-world settings, and consequently must be incorporated into the definition of security. Further, we have demonstrated that not all game specifications forbidding point-to-point communication are inadmissible under our model. We presented the transformation for the classic prisoner's dilemma, which disallows point-to-point communication through physical assumptions, into a modified game that is admissible under our model and preserves the original equilibrium. Similarly, we have demonstrated that the signaling game has an expected payoff of 1 when executed in the presence of point-to-point channels, rather than an expected payoff of $\frac{1}{2}$, a distinction not captured by models that disallow communication outside of the protocol execution. Finally, we have presented a full security proof for rational secret sharing under our proposed framework. Although our results are not universal, we have demonstrated a powerful benefit of our model: assigning local adversaries may aid mechanism design in destabilizing the formation of coalitions. Thus, there are tangible benefits from adopting our definition of security against local rational adversaries in the presence of point-to-point communication resources.

8 SUMMARY

In this thesis, we have presented a rational cryptographic framework for both the two-party and multiparty settings. We have demonstrated the necessary and sufficient utility assumptions to achieve privacy, correctness and fairness in game theoretic terms, as well as removed all restrictions on the communication resources available to players. We have argued the necessity of allowing point-to-point communication channels between players, which prior work restricts in an attempt to address collusion. Although we do not restrict the communication resources, our frameworks accept a non-trivial class of ideal game specifications.

8.1 Summary of Main Results

8.1.1 Separated Classification and Inspection

In Chapter 4, we separate the tasks of classification and inspection in the adversarial classification problem. Although closely related, separating the task of inspection from the task of classification yields an advantage to a defender. Working against an active adversary, we apply game theory to make optimal operational decisions for the inspection policy in the presence of limited resources.

8.1.2 Resolved Game Theoretic Dilemma

In Chapter 5, we resolve a game theoretic dilemma where an auction model's foundational assumption prevented deployment in real world settings. We demonstrate that realizable protocols for the auction model are possible by employing cryptographic primitives, and that the realized protocols satisfy the theoretical model's underlying assumptions.

8.1.3 Applied Stronger Equilibrium Concept

In Chapter 6, we propose the *computational perfect Bayesian equilibrium* (PBE) concept as a replacement for the widely-used computational Nash equilibrium. We have argued that PBE is a more realistic equilibrium concept for cryptographic protocols, which typically proceed in a series of rounds and necessarily represent games of imperfect information.

8.1.4 Game Theoretic Security Definitions

In Chapter 6, we give definitions of privacy, correctness and fairness purely in game theoretic terms. From these definitions, we demonstrate the necessary and sufficient conditions for player utility functions in order to achieve them in protocol design.

8.1.5 Removed Restrictions on Communication Resources

In Chapter 7, we introduce a rational multiparty computation framework that places no restrictions on the communication resources available to players. Prior work imposed strong restrictions on the communication resources available to players, as arbitrary communication enables collusion. However, we have demonstrated that even ideal game specifications that restrict communication may be translated into equivalent formulations allowing point-to-point communication. Once admissible in our proposed ideal model, the ideal game specifications are translated into real world protocol constructions.

LIST OF REFERENCES

LIST OF REFERENCES

[1] Leon Walras. *Élements d'Economie Politique or Elements of Pure Economics; translated by William Jaffe*. Routledge, London and New York, 1874.

[2] Andrew C. Yao. Protocols for Secure Computations. In *SFCS '82: Proceedings of the 23rd Annual Symposium on Foundations of Computer Science*, pages 160–164, Washington, DC, USA, 1982. IEEE Computer Society.

[3] Andrew C. Yao. How to Generate and Exchange Secrets. In *SFCS '86: Proceedings of the 27th Annual Symposium on Foundations of Computer Science*, pages 162–167, Washington, DC, USA, 1986. IEEE Computer Society.

[4] O. Goldreich, S. Micali, and A. Wigderson. How to Play ANY Mental Game. In *STOC '87: Proceedings of the 19th Annual ACM Symposium on Theory of Computing*, pages 218–229, New York, NY, USA, 1987. ACM.

[5] O. Goldreich. *Foundations of Cryptography*, volume 2. Cambridge University Press, 2004.

[6] S. Goldwasser, S. Micali, and C. Rackoff. The Knowledge Complexity of Interactive Proof Systems. *SIAM Journal on Computing*, 18:186–208, February 1989.

[7] Yehuda Lindell and Benny Pinkas. A Proof of Security of Yao's Protocol for Two-Party Computation. *Journal of Cryptology*, 22:161–188, 2009.

[8] Michael O Rabin. How to Exchange Secrets by Oblivious Transfer. *Exchange Organizational Behavior Teaching Journal*, pages 1–5, 1981.

[9] Yan Huang, David Evans, Jonathan Katz, and Lior Malka. Faster Secure Two-Party Computation Using Garbled Circuits. In *USENIX Security Symposium*. USENIX Association, 2011.

[10] Assaf Ben-David, Noam Nisan, and Benny Pinkas. FairplayMP: a System for Secure Multi-party Computation. In *Proceedings of the 15th ACM Conference on Computer and Communications Security*, CCS '08, pages 257–266, New York, NY, USA, 2008. ACM.

[11] Pascal Paillier. Public-key Cryptosystems based on Composite Degree Residuosity Classes. In *EUROCRYPT'99: Proceedings of the 17th International Conference on Theory and Application of Cryptographic Techniques*, pages 223–238. Springer-Verlag, Berlin, Heidelberg, 1999.

[12] Jonathan Katz. Bridging Game Theory and Cryptography: Recent Results and Future Directions. In Ran Canetti, editor, *TCC*, volume 4948 of *Lecture Notes in Computer Science*, pages 251–272. Springer-Verlag, Berlin, Heidelberg, 2008.

[13] Martin J. Osborne and Ariel Rubinstein. *A Course in Game Theory*, volume 1 of *MIT Press Books*. The MIT Press, 1994.

[14] John Nash. Non-Cooperative Games. *The Annals of Mathematics*, 54(2):286–295, September 1951.

[15] Gilad Asharov, Ran Canetti, and Carmit Hazay. Towards a Game Theoretic View of Secure Computation. In *Proceedings of the 30th Annual International Conference on Theory and Applications of Cryptographic Techniques: Advances in Cryptology*, EUROCRYPT'11, pages 426–445. Springer-Verlag, Berlin, Heidelberg, 2011.

[16] Yevgeniy Dodis, Shai Halevi, and Tal Rabin. A Cryptographic Solution to a Game Theoretic Problem. In Mihir Bellare, editor, *Advances in Cryptology, CRYPTO 2000*, volume 1880 of *Lecture Notes in Computer Science*, pages 112–130. Springer-Verlag, Berlin, Heidelberg, 2000.

[17] Gillat Kol and Moni Naor. Cryptography and Game Theory: Designing Protocols For Exchanging Information. In *Proceedings of the 5th Conference on Theory of Cryptography*, TCC'08, pages 320–339. Springer-Verlag, Berlin, Heidelberg, 2008.

[18] Peter Bro Miltersen, Jesper Buus Nielsen, and Nikos Triandopoulos. Privacy-Enhancing Auctions Using Rational Cryptography. In *Proceedings of the 29th Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '09, pages 541–558. Springer-Verlag, Berlin, Heidelberg, 2009.

[19] Zhifang Zhang and Mulan Liu. Unconditionally Secure Rational Secret Sharing in Standard Communication Networks. In *Proceedings of the 13th International Conference on Information Security and Cryptology*, ICISC'10, pages 355–369. Springer-Verlag, Berlin, Heidelberg, 2011.

[20] Christos H. Papadimitriou. On the Complexity of the Parity Argument and Other Inefficient Proofs of Existence. *Journal of Computer and System Sciences*, 48(3):498–532, June 1994.

[21] Constantinos Daskalakis, Paul W. Goldberg, and Christos H. Papadimitriou. The Complexity of Computing a Nash Equilibrium. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, STOC '06, pages 71–78, New York, NY, USA, 2006. ACM.

[22] Georg Fuchsbauer, Jonathan Katz, and David Naccache. Efficient Rational Secret Sharing in Standard Communication Networks. In Daniele Micciancio, editor, *Theory of Cryptography*, volume 5978 of *Lecture Notes in Computer Science*, pages 419–436. Springer-Verlag, Berlin, Heidelberg, 2010.

[23] Joseph Halpern and Vanessa Teague. Rational Secret Sharing and Multiparty Computation: Extended Abstract. In *Proceedings of the 36th Annual ACM Symposium on Theory of Computing*, STOC '04, pages 623–632, New York, NY, USA, 2004. ACM.

[24] S. Dov Gordon and Jonathan Katz. Rational Secret Sharing, Revisited. Cryptology ePrint Archive, Report 2006/142, 2006. `http://eprint.iacr.org/`.

[25] Silvio Micali and Abhi Shelat. Purely Rational Secret Sharing (Extended Abstract). In *Proceedings of the 6th Theory of Cryptography Conference on Theory of Cryptography*, TCC '09, pages 54–71. Springer-Verlag, Berlin, Heidelberg, 2009.

[26] Gillat Kol and Moni Naor. Games For Exchanging Information. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing*, STOC '08, pages 423–432, New York, NY, USA, 2008. ACM.

[27] Itzhak Gilboa and Eitan Zemel. Nash and Correlated Equilibria: Some Complexity Considerations. Discussion Papers 777, Northwestern University, Center for Mathematical Studies in Economics and Management Science, June 1988.

[28] Amparo Urbano and Jose E. Vila. Computationally Restricted Unmediated Talk under Incomplete Information. *Economic Theory*, 23(2):283–320 (2004), 2004.

[29] Mikhail J. Atallah, Marina Blanton, Keith B. Frikken, and Jiangtao Li. Efficient Correlated Action Selection. In Giovanni Di Crescenzo and Avi Rubin, editors, *Financial Cryptography and Data Security*, volume 4107 of *Lecture Notes in Computer Science*, pages 296–310. Springer-Verlag, Berlin, Heidelberg, 2006.

[30] Ronen Gradwohl, Noam Livne, and Alon Rosen. Sequential Rationality in Cryptographic Protocols. In *Proceedings of the 2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, FOCS '10, pages 623–632, Washington, DC, USA, 2010. IEEE Computer Society.

[31] Bernhard von Stengel and Françoise Forges. Extensive-Form Correlated Equilibrium: Definition and Computational Complexity. *Mathematics of Operations Research*, 33(4):1002–1022, 2008.

[32] Adam Groce and Jonathan Katz. Fair Computation with Rational Players. In David Pointcheval and Thomas Johansson, editors, *Advances in Cryptology - EUROCRYPT 2012*, volume 7237 of *Lecture Notes in Computer Science*, pages 81–98. Springer-Verlag, Berlin, Heidelberg, 2012.

[33] John C Harsanyi. Games with Incomplete Information Played by Bayesian Players, I-III. Part II. Bayesian Equilibrium Points. *Management Science*, 14(5):320–334, 1968.

[34] Joseph Y Halpern and Rafael Pass. Game Theory with Costly Computation. *Proceedings of the Behavioral and Quantitative Game Theory on Conference on Future Directions BQGT 10*, pages 1–1, 2008.

[35] ZhiFang Zhang and MuLan Liu. Rational Secret Sharing as Extensive Games. *Science China Information Sciences*, 56:1–13, 2013.

[36] David M Kreps and Robert Wilson. Sequential Equilibria. *Econometrica*, 50(4):863–94, July 1982.

[37] Giacomo Bonanno. AGM-Consistency and Perfect Bayesian Equilibrium. Part I: Definition and Properties. *International Journal of Game Theory*, pages 1–26, 2011.

[38] Julio González-Díaz and Miguel Meléndez-Jiménez. On the Notion of Perfect Bayesian Equilibrium. *TOP*, pages 1–16, 2011. 10.1007/s11750-011-0239-z.

[39] Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay V. Vazirani. *Algorithmic Game Theory.* Cambridge University Press, New York, NY, USA, 2007.

[40] Drew Fudenberg and Jean Tirole. *Game Theory.* MIT Press, August 1991.

[41] Amir Globerson and Sam Roweis. Nightmare at Test Time: Robust Learning by Feature Deletion. In *ICML '06: Proceedings of the 23rd International Conference on Machine Learning*, pages 353–360, New York, NY, USA, 2006. ACM Press.

[42] Ling Huang, Anthony D. Joseph, Blaine Nelson, Benjamin I.P. Rubinstein, and J. D. Tygar. Adversarial Machine Learning. In *Proceedings of the 4th ACM Workshop on Security and Artificial Intelligence*, AISec '11, pages 43–58, New York, NY, USA, 2011. ACM.

[43] Michael Brückner and Tobias Scheffer. Nash Equilibria of Static Prediction Games. In *Advances in Neural Information Processing Systems*, pages 171–179, 2009.

[44] Wei Liu and Sanjay Chawla. Mining Adversarial Patterns via Regularized Loss Minimization. *Machine Learning*, 81(1):69–83, 2010.

[45] Michael Brückner and Tobias Scheffer. Stackelberg Games for Adversarial Prediction Problems. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, pages 547–555, New York, NY, USA, 2011. ACM.

[46] Richard Colbaugh and Kristin Glass. Predictive Defense against Evolving Adversaries. In *IEEE International Conference on Intelligence and Security Informatics*, pages 18–23, 2012.

[47] Jason D. M. Rennie. ifile: An Application of Machine Learning to E-Mail Filtering. In *Proceedings of the KDD Workshop on Text Mining*, 2000.

[48] Robin Sommer and Vern Paxson. Outside the Closed World: On Using Machine Learning for Network Intrusion Detection. In *IEEE Symposium on Security and Privacy*, pages 305–316, 2010.

[49] Murat Kantarcıoğlu, Bowei Xi, and Chris Clifton. Classifier Evaluation and Attribute Selection against Active Adversaries. *Data Mining and Knowledge Discovery*, 22(1-2):291–335, January 2011.

[50] Hamed Masnadi-Shirazi and Nuno Vasconcelos. Risk Minimization, Probability Elicitation, and Cost-Sensitive SVMs. In *International Conference on Machine Learning*, pages 759–766, 2010.

[51] Stas Filshtinskiy. Cybercrime, Cyberweapons, Cyber Wars: Is There Too Much of It in the Air? *Communications of the ACM*, 56(6):28–30, 2013.

[52] Nilesh Dalvi, Pedro Domingos, Mausam, Sumit Sanghai, and Deepak Verma. Adversarial classification. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pages 99–108, New York, NY, USA, 2004. ACM.

[53] Daniel Lowd and Christopher Meek. Adversarial Learning. In *ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, pages 641–647, 2005.

[54] Janez Demšar. Statistical Comparisons of Classifiers over Multiple Data Sets. *Journal of Machine Learning Research*, 7:1–30, December 2006.

[55] Steven L. Salzberg. On Comparing Classifiers: Pitfalls to Avoid and a Recommended Approach. *Data Mining and Knowledge Discovery*, 1(3):317–328, January 1997.

[56] Battista Biggio, Giorgio Fumera, and Fabio Roli. Adversarial Pattern Classification Using Multiple Classifiers and Randomisation. In *Structural, Syntactic, and Statistical Pattern Recognition*, volume 5342 of *Lecture Notes in Computer Science*, pages 500–509. Springer-Verlag, Berlin, Heidelberg, 2008.

[57] T. Alpcan. *Network Security: A Decision and Game-Theoretic Approach*. Cambridge University Press, 2010.

[58] L. Chen and J. Leneutre. A Game Theoretical Framework on Intrusion Detection in Heterogeneous Networks. *IEEE Transactions on Information Forensics and Security*, 4(2):165–178, 2009.

[59] M. Kodialam and T. Lakshman. Detecting Network Intrusions via Sampling: A Game Theoretic Approach. In *22nd IEEE Annual Computer and Communications Conference*, pages 1880–1889, 2003.

[60] H. Otrok, M. Mehrandish, C. Assi, M. Debbabi, and P. Bhattacharya. Game Theoretic Models for Detecting Network Intrusions. *Computer Communications*, 31(10):1934–1944, 2008.

[61] Ondrej Vanek, Zhengyu Yin, Manish Jain, Branislav Bosansky, Milind Tambe, and Michal Pechoucek. Game-Theoretic Resource Allocation for Malicious Packet Detection in Computer Networks. In *11th International Conference on Autonomous Agents and Multiagent Systems*, 2012.

[62] Richard Cole and Lisa Fleischer. Fast-Converging Tatonnement Algorithms for One-time and Ongoing Market Problems. In *STOC '08: Proceedings of the 40th Annual ACM symposium on Theory of Computing*, pages 315–324, New York, NY, USA, 2008. ACM.

[63] Ran Canetti and Margarita Vald. Universally Composable Security with Local Adversaries. In Ivan Visconti and Roberto Prisco, editors, *Security and Cryptography for Networks*, volume 7485 of *Lecture Notes in Computer Science*, pages 281–301. Springer-Verlag, Berlin, Heidelberg, 2012.

[64] Sergei Izmalkov, Silvio Micali, and Matt Lepinski. Rational Secure Computation and Ideal Mechanism Design. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '05, pages 585–595, Washington, DC, USA, 2005. IEEE Computer Society.

[65] Anna Lysyanskaya and Nikos Triandopoulos. Rationality and Adversarial Behavior in Multi-party Computation. In Cynthia Dwork, editor, *CRYPTO*, volume 4117 of *Lecture Notes in Computer Science*, pages 180–197. Springer-Verlag, Berlin, Heidelberg, 2006.

[66] Peter Bogetoft, DanLund Christensen, Ivan Damgård, Martin Geisler, Thomas Jakobsen, Mikkel Krøigaard, JanusDam Nielsen, Jesper Buus Nielsen, Kurt Nielsen, Jakob Pagter, Michael Schwartzbach, and Tomas Toft. Secure Multiparty Computation Goes Live. In Roger Dingledine and Philippe Golle, editors, *Financial Cryptography and Data Security*, volume 5628 of *Lecture Notes in Computer Science*, pages 325–343. Springer-Verlag, Berlin, Heidelberg, 2009.

[67] Chris Clifton, Ananth Iyer, Richard Cho, Wei Jiang, Murat Kantarcıoğlu, and Jaideep Vaidya. An Approach to Identifying Beneficial Collaboration Securely in Decentralized Logistics Systems. *Management & Service Operations Management*, 10(1):108–125, Winter 2008.

[68] James Eaves and Jeffrey C. Williams. Walrasian Tâtonnement Auctions on the Tokyo Grain Exchange. *Review of Financial Studies*, 20(4):1183–1218, 2007.

[69] Wei Jiang and Chris Clifton. AC-Framework for Privacy-Preserving Collaboration. In *Proceedings of the 7th SIAM International Conference on Data Mining, April 26-28, 2007, Minneapolis, Minnesota, USA*. SIAM, 2007.

[70] Ivan Damgård, Martin Geisler, Mikkel Krøigaard, and Jesper Buus Nielsen. Asynchronous Multiparty Computation: Theory and Implementation. In Stanislaw Jarecki and Gene Tsudik, editors, *Public Key Cryptography*, volume 5443 of *Lecture Notes in Computer Science*, pages 160–179. Springer-Verlag, Berlin, Heidelberg, 2009.

[71] Martin Burkhart, Mario Strasser, Dilip Many, and Xenofontas Dimitropoulos. SEPIA: Privacy-Preserving Aggregation of Multi-Domain Network Events and Statistics. In *Proceedings of the 19th USENIX Conference on Security*, USENIX Security'10, pages 15–15, Berkeley, CA, USA, 2010. USENIX Association.

[72] Morten Dahl, Chao Ning, and Tomas Toft. On Secure Two-Party Integer Division. In Angelos D. Keromytis, editor, *Financial Cryptography and Data Security*, volume 7397 of *Lecture Notes in Computer Science*, pages 164–178. Springer-Verlag, Berlin, Heidelberg, 2012.

[73] Keith Frikken and Lukasz Opyrchal. PBS: Private Bartering Systems. In *Financial Cryptography and Data Security: 12th International Conference, FC 2008, Cozumel, Mexico, January 28-31, 2008. Revised Selected Papers*, pages 113–127. Springer-Verlag, Berlin, Heidelberg, 2008.

[74] Moni Naor, Benny Pinkas, and Reuban Sumner. Privacy Preserving Auctions and Mechanism Design. In *EC '99: Proceedings of the 1st ACM Conference on Electronic Commerce*, pages 129–139, New York, NY, USA, 1999. ACM.

[75] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28(2):270 – 299, 1984.

[76] Torben Pryds Pedersen. Non-Interactive and Information-Theoretic Secure Verifiable Secret Sharing. In Joan Feigenbaum, editor, *Advances in Cryptology, CRYPTO 1992*, volume 576 of *Lecture Notes in Computer Science*, pages 129–140. Springer-Verlag, Berlin, Heidelberg, 1992.

[77] Yoav Shoham and Moshe Tennenholtz. Non-Cooperative Computation: Boolean Functions with Correctness and Exclusivity. *Theory of Computer Science*, 343:97–113, October 2005.

[78] Mehrdad Nojoumian and DouglasR. Stinson. Socio-Rational Secret Sharing as a New Direction in Rational Cryptography. In Jens Grossklags and Jean Walrand, editors, *Decision and Game Theory for Security*, volume 7638 of *Lecture Notes in Computer Science*, pages 18–37. Springer-Verlag, Berlin, Heidelberg, 2012.

[79] Yonatan Aumann and Yehuda Lindell. Security Against Covert Adversaries: Efficient Protocols for Realistic Adversaries. In *Proceedings of the 4th Conference on Theory of Cryptography*, TCC'07, pages 137–156. Springer-Verlag, Berlin, Heidelberg, 2007.

[80] Shafi Goldwasser and Silvio Micali. Probabilistic Encryption. *Journal of Computer and System Sciences*, 28(2):270–299, 1984.

[81] Adam Groce, Jonathan Katz, Aishwarya Thiruvengadam, and Vassilis Zikas. Byzantine Agreement with a Rational Adversary. In *Proceedings of the 39th International Colloquium Conference on Automata, Languages, and Programming*, ICALP'12, pages 561–572. Springer-Verlag, Berlin, Heidelberg, 2012.

[82] Adi Shamir. How to Share a Secret. *Communications of the ACM*, 22(11):612–613, November 1979.

[83] Juan M. Estevez-Tapiador, Almudena Alcaide, Julio C. Hernandez-Castro, and Arturo Ribagorda. Bayesian Rational Exchange. *International Journal of Information Security*, 7:85–100, 2008.

[84] Joël Alwen, Abhi Shelat, and Ivan Visconti. Collusion-Free Protocols in the Mediated Model. In David Wagner, editor, *Advances in Cryptology, CRYPTO 2008*, volume 5157 of *Lecture Notes in Computer Science*, pages 497–514. Springer-Verlag, Berlin, Heidelberg, 2008.

[85] Joël Alwen, Jonathan Katz, Yehuda Lindell, Giuseppe Persiano, Abhi Shelat, and Ivan Visconti. Collusion-Free Multiparty Computation in the Mediated Model. In Shai Halevi, editor, *Advances in Cryptology, CRYPTO 2009*, volume 5677 of *Lecture Notes in Computer Science*, pages 524–540. Springer-Verlag, Berlin, Heidelberg, 2009.

[86] Matt Lepinksi, Silvio Micali, and Abhi Shelat. Collusion-Free Protocols. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, STOC '05, pages 543–552, New York, NY, USA, 2005. ACM.

[87] Joël Alwen, Jonathan Katz, Ueli Maurer, and Vassilis Zikas. Collusion-Preserving Computation. In Reihaneh Safavi-Naini and Ran Canetti, editors, *Advances in Cryptology, CRYPTO 2012*, volume 7417 of *Lecture Notes in Computer Science*, pages 124–143. Springer-Verlag, Berlin, Heidelberg, 2012.

[88] Seny Kamara, Payman Mohassel, and Mariana Raykova. Outsourcing Multi-Party Computation. Cryptology ePrint Archive, Report 2011/272, 2011. `http://eprint.iacr.org/`.

[89] Joël Alwen, Christian Cachin, Jesper Buus Nielsen, Olivier Pereira, Ahmad-Reza Sadeghi, Berry Schomakers, Abhi Shelat, and Ivan Visconti. Summary Report on Rational Cryptographic Protocols, 2007. `http://www.ecrypt.eu.org`.

[90] William Poundstone. *Prisoner's Dilemma: John Von Neumann, Game Theory and the Puzzle of the Bomb*. Doubleday, New York, NY, USA, 1st edition, 1992.

[91] John Ross Wallrabenstein and Chris Clifton. Privacy Preserving Tatonnement: A Cryptographic Construction of an Incentive Compatible Market. In *Financial Cryptography and Data Security*, Lecture Notes in Computer Science. Springer-Verlag, Berlin, Heidelberg, 2014.

[92] Shaik Maleka, Amjed Shareef, and C.Pandu Rangan. Rational Secret Sharing with Repeated Games. In Liqun Chen, Yi Mu, and Willy Susilo, editors, *Information Security Practice and Experience*, volume 4991 of *Lecture Notes in Computer Science*, pages 334–346. Springer-Verlag, Berlin, Heidelberg, 2008.

[93] Rafail Ostrovsky and Moti Yung. How to Withstand Mobile Virus Attacks (Extended Abstract). In *Proceedings of the 10th Annual ACM Symposium on Principles of Distributed Computing*, PODC '91, pages 51–59, New York, NY, USA, 1991. ACM.

VITA

VITA

John Ross Wallrabenstein was born in 1988. He was admitted to Purdue University as a Ph.D. student in 2009 after receiving a B.S. in Computer Science from Miami University. He received a M.S. in Computer Science from Purdue University in 2012. During his time at Purdue, he worked for Ricoh, Sandia National Laboratories and Sypris Electronics. In December 2014 he received the degree of Doctor of Philosophy under the supervision of Prof. Chris Clifton. His research interests include theoretical and applied cryptography, secure multiparty computation, as well as game theory.