UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

# ON THE COMPLEXITY OF TIPPING IN SUPER-MODULAR GAMES

A DISSERTATION

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

Degree of

DOCTOR OF PHILOSOPHY

By

BRIAN L. CREMEANS

Norman, Oklahoma

2014

ON THE COMPLEXITY OF TIPPING IN SUPER-MODULAR GAMES

A DISSERTATION APPROVED FOR THE

SCHOOL OF COMPUTER SCIENCE

BY

---

Dr. S Lakshmivarahan, Co-Chair

---

Dr. S Dhall, Co-Chair

---

Dr Q Cheng

---

Dr. K Thulasaraman

---

Dr. R Landes

# Contents

# List of Tables

# List of Figures

# Abstract

The problem of finding the minimum tipping set in a super modular game is known to be NP-hard. Here, I derive an approximation algorithm to find a small tipping set in such a game. In the special case of the uniform game, the approximation provides the exact minimum tipping set. Interdependent security is a growing field. One model used for interdependent security is the airline security model. This model is used as an example for the approximation methods, and was the working model for many of the proofs and strategies developed to find tipping sets and their approximations. This algebraic approach, which makes use of group theory, is then evaluated for accuracy, It is then applied to a dynamic approach, using a simple learning function without the complete information often assumed.

This method links the non-greedy approximation to a version of SAT, and a type of influence graphs and the covering problem. The approximation fared well when finding the key players in a game, but struggled with cascades.

Chapter 1

# 1 Game Theory Concepts

Game theory is the study of decision making, usually in competitive systems. It strives to understand how rational players, be they individuals, companies, governments, or other entities, make strategic choices, when faced with a competitive situation when the results of their action is not just dependent upon their own action, but that of other involved parties the strategic choices can be difficult to understand. Even with simple interdependency these interactions, or games, between different players making different choices can exhibit complex behavior. Computer science, math, and information theory have developed systems to model these behaviors. Such systems have been used in politics, economics, computer networks, security, and many other fields to model behavior. [8][16]

To gain a greater understanding of this, we first need to explore some general concepts and terminology related to games. We will also develop the needed notation to represent the ideas.

## 1.1 Introduction

To aid in further discussion and analysis of the field of game theory, let us define some general terms. These will be used throughout this document. A game is a general system in which the involved parties are interacting, and their results are dependent upon the choices of all or some of the other parties. These parties are referred to as agents or players. Thus a game has multiple players, competing, cooperating, or at the bare minimum interacting with each other, generally with the intent to attempt to get their best results. These results are called payoffs. The negative of these payoffs are, in some cases, referred to as losses. Maximizing the payoffs (a common goal) is the same as minimizing the losses. Each player in a game has options. These options are the set of actions the player can choose to take, and are referred to as strategies. Each player can have any number of strategies. A play of a game is the set of each players choices of strategy. A play of a game determines the payoffs.

Sometimes these payoffs are computed to be the average of their possibilities, or their expected value. The payoff, or the expected value of the payoff are terms often used interchangeably. Sometimes these payoffs are referred to as utility, and the study of the properties of these utilities is called utility theory. It is worth noting that utility is inherently subjective. That is, each player sets how they value the various payoff options derived from the strategic choices. Thus the payoffs of players are inherently non-comparable.

Another important concept in games is rationality. A fundamental concept of game theory is that each player is assumed to be rational. By behaving rationally, we mean that the players seek to maximize their own payoff with no regard for the other players. This is sometimes termed selfish, but we will use the term rational to describe this behavior. [16][18]

## 1.2 Types of Games

Often, to study games it is helpful to categorize games based upon their properties and common characteristics. Then we can study the special cases separately. Here we show some common types of games.

### 1.2.1 Pure Strategy Games

Pure strategy games are games in which each of the players must choose exactly one of their strategies for a play of the game. Thus a single play of the game is a set of strategies, one from each player in the game. The payoffs are then strictly determined by the strategic choices of the players and the definition of the game. Each player has to choose their strategy deterministically, and cannot rely on any random chance or device.

Pure strategy games are more intuitive, than mixed strategy games but often just as complicated. It is conceptually similar to a discrete problem instead of a continuous problem.[16][18]

### 1.2.2 Mixed Strategy Games

Mixed strategy games are similar to pure strategy games, except that each player can determine their choice randomly, rather than deterministically. In these games, players may weight their random choices to prefer certain strategies. In some games, the strategies may have equal weight, and in others it might be biased toward one choice or another. Often this ability actually simplifies finding solutions, or their existence, when compared to pure strategy games.[16][18] In competitive situations, it is often beneficial to choose randomly. For example, when playing Paper Rock Scissors, the best strategy is to choose all options uniformly randomly.

### 1.2.3 Zero Sum Games

One common type of game is a zero sum game. This is a game in which the payoffs balance. That is, if one player gains X, the rest of the players must lose X. It is easy to think of this as a group of players competing for a fixed set of resources. In order for one player to get a resource, the rest of the player collectively must have lost that resource.

A game can be Mixed Strategy, and zero sum, or any other combination of these qualities. A special case of this is the two player zero sum game in mixed strategies. In this case, we can apply the Minimax theorem. In common terms, this theorem says that if one player can guarantee a payoff of X, then the other player will get -X. This is intuitive for a two player game in that a player, in maximizing payoff, minimizes the opponent's payoff.[16][18]

### 1.2.4 Non-Zero Sum Games

Alternatively, a game can be non-zero sum. In this case, players are not competing for a fixed resource or uniform value. In this case, one player's gain does not necessitate the other players' loss. This is often the case in large, complex interdependent systems.[16][18]

## 1.3 Solutions

There are several solution concepts in game theory. Generally a solution represents the state when all players settle. Given their choices, they have chosen the best they can, maximizing their utility. In the case of mixed strategies, this means that they choose the weights of strategies to maximize their utility. The most common type of solution used is a Nash Equilibrium (NE). In a NE, no player is willing to deviate from his choice alone. Thus it takes more than one player to move the system. NE are not necessarily unique solutions, and are not necessarily optimal. NE are still commonly used because they well represent how solution naturally evolve from competition. They also often produce feasible solutions. There can be several NE in a game, and they can be at different payoff values. [15][16][18]

## 1.4 Details of Games

Thus far we have introduced some general, intuitive explanations of games. Now we will explore the relevant precise computational elements of games and the specifics needed for further analysis. After some notes on solutions and game types, we will more closely examine a special type of game upon which most of the following work is based.

The study of game theory strives to understand the behavior of agents, be they people, companies, governments, or other entities in systems. We denote these players by labeling them as player $i$. A single player game reduces to an optimization problem. Since we are interested in games with multiple players, we use $n$ to denote the number of players, and index the players $i : 1, 2, ..., n$ for reference. Generally speaking, these systems are competitive, or non-cooperative. We assume that these players are rational, that is, that they do what is in their best interest to maximize their gain and minimize their loss from a situation. Games can be as simple as Paper, Rock, Scissors, or as complex as derivative markets. These systems arise in may areas, including advertising, networks, negotiations, and security. [16][17]

Given a situation or play, each of the players has a set of strategies to follow. We denote these strategies as $s_i$ for each player $i$. The set of strategies can be any action or combined action taken (or not taken) by a player. Herein, we are primarily concerned with the cases when $s_i \in \{0,1\}$ where 0 represents the choice of a player not to take action, and 1 the choice to act or invest. Sometimes, it is in a player's best interest to choose randomly (with some defined weight to each strategy) between these strategies, which is a mixed strategy. although this research is concerned with pure strategy games, wherein the players are limited to picking one strategy exclusively.

A play $s$ is defined by the n-tuple $s = (s_{1,}, s_2, ..., s_n)$ where $s_i \in \{0,1\}$ denotes the choice of pure strategy by a player $i$ and $1 \leq i \leq n$.[16] A player $i$'s payoff or loss function is denoted as $u_i(s)$. This is often the average payoff given the actions of the players. We use $S$ to denote the set of all possible distinct plays. There are a total of $2^n$ distinct plays denoted by $S = \{s | s = (s_1, s_2, ..., s_n), s_i \in \{0,1\}\}$in the games we discussed herein.

Of course, the analysis of such complex systems is somewhat divorced from reality, thus we must develop a model of the payoff functions. It is constructed as a function $U : S \rightarrow R^n$ and defined by:$U(s) = (u_1(s), ..., u_n(s)$ where each $u_i$ is the payoff function for player $i$ and $s$ is the strategic choice of that player. Thus $U$ maps $S \rightarrow R^n$, the payoffs of each player, given the play $s = (s_1, ...s_n) \in S$ to the payoff vector for all players. Thus any game is completely defined by $(S, U)$.

## 1.5 Super-modular Games

One important class of games is super-modular games. This class of games stems from a class of functions by the same name. Super-modular functions are functions with a property known as 'increasing differences'.[16]. Intuitively, 'increasing differences' means that as more players decide to use a strategy, the more incentivized others are to do use the same strategy. Mostly,this quality is utilized in cases with only two strategies, represented by 0 and 1. We use the notation $s_{i-i}$ to represent the set of strategy choices $S$ for every player except $i$. To characterize this, we often examine the difference in a player's payoffs resulting from their choice. To examine tipping, we often consider the difference quantity:

$u_i(1_i, s_{-i}) - u_i(0_i, s_{-i})$

Intuitively, this quantifies the proposition of the player $i$ to choose strategy 1 over 0 for any given distinct partial play $s_{-i}$. If this value is positive, the player will choose 1 instead of 0. In this paper, this will often be referred to as the choice to 'invest' (1) or 'not invest' (0). For a game to be super-modular, it must conform to the following inequality:

$$u_i(1_i, s'_{-i}) - u_i(0_i, s'_{-i}) \geq u_i(1_i, s_{-i}) - u_i(0_i, s - i) \quad (1)$$

if $s'_{-i} \geq s_{-i}$with strict inequality if $s'_{-i} > s_{-i}$[10][21]

In our case, the underlying binary lattice is defined by the $2^n$ binary strings of 1's and 0's under the partial order defined over the binary strings. It turns out this lattice is also a hypercube of dimension $n$.[7]

## 1.6 Tipping

The focus of this work is on tipping. As mentioned above, NE are not unique, and can vary in value. One NE can be considered superior, or more efficient, than another. This prompts the interesting question of "How do we move from one NE, presumably an inefficient one, to another one?" This movement can be achieved by externally motivating a set of players to change and letting the rest follow suit, ending in another NE. A tipping set is a set of players that motivate such a change.

A minimum tipping set is the minimum of such tipping sets. The primary interest of this work is in finding the minimum tipping set, or the smallest amount of players to move from one NE to another.[8][10][3]The phenomenon we are interested in is tipping. This is the transition from one NE to another, and typically from an inefficient NE to an efficient one. The above super-modularity condition 1 is sufficient to produce this tipping phenomenon in binary games (games with two strategies). This has many applications, from market coverage, social network saturation, and security or technology investment. Generally, this type of dynamic arises when a single player rationally won't force transition of the whole system to a new state alone, but if everyone chooses the new state (or a sufficient number of players) it would be better for everyone, or at least the investing players. Then, the right players together choose to change, and everyone, including the initially abstaining players follows suit rationally.

A similar phenomenon is called cascading, in which the players change one at a time like dominoes, but in a perfect information system, with abstracted transitional times, cascades don't come into play because each player would know that the cascade would happen and skip to the investment strategy at the end.[8][9][10]

### 1.6.1 Definitions

The study of games require us to have a set nomenclature. As discussed above, $(S, U)$ defines a game. We let $S = \{s | s = (s_1, s_2, ..., s_n), s_i \in \{0, 1\}\}$ so it is a set of possible strategies for each play. Each $s_i$ is the strategic choice of the $i^{th}$player. Here, $n$ is the number of players, or agents in the game, indexed $1, ..., n$. In our case, where the strategies are 0 or 1, this set of all actions has cardinality $|S| = 2^n$. Clearly, this is a n-tuple of n zeros or ones, thus the cardinality. We use $U$ to be the payoff, equivalently gain or loss, by each of the players.

When studying these games, it is often useful to directly examine the impact a player has on another. The common notation for this is as follows:

$$\triangle_j(s_{-j}) = u_j(s_j, 1) - u_j(s_{-j}, 0)$$

Where $\triangle_j(s_{-j})$, is defined to be the incentive of player $j$ to invest, given that all the other players use their strategic choices in $s$. If this is positive for payoff, or negative for loss, it is in player $j$ 's best interest to invest, choosing strategy 1. To examine the impact of a player $i$ on player $j$, we use the following extension of this notation:

$$\triangle_j(s_{-i-j}, 0_i) = u_j(s_{-i-j}, 0_i 1_j) - u_j(s_{-i-j}, 0_i, 0_j) \quad (2)$$

$$\triangle_j(s_{-i-j}, 1_i) = u_j(s_{-i-j}, 1_i 1_j) - u_j(s_{-i-j}, 1_i, 0_j) \quad (3)$$

These show the incentive for player $j$ first if player $i$ chose 0, then 1. To quantify player $i$ 's impact on player $j$, we use the following:

$$\Delta_{ij} = \triangle_j(s_{-i-j}, 1_i) - \triangle_j(s_{-i-j}, 0_i) =$$

$$(u_j(s_{-i-j}, 1_i, 1_j) - u_j(s_{-i-j}, 1_i, 0_j)) - (u_j(s_{-i-j}, 0_i 1_j) - u_j(s_{-i-j}, 0_i, 0_j) \quad (4)$$

In a super-modular (or monotone) system, this value is always positive (or negative) by definition for all $i$ and $j$. Thus as more players invest, it can only incentivized other players to invest as well. Intuitively, if there is an NE at $1^n$ there is some subset of players who will make the other players invest as well. A trivial example of this is would be $n-1$ players. Since $1^n$ is a NE, the remaining 1 player would be sufficiently incentivized to invest. This, coupled with the presence of a NE at $0^n$and another at $1^n$is sufficient to have a tipping or cascading phenomenon.

### 1.6.2 Complexity of Tipping

Finding a solution to these kinds of problems is generally possible, but we are interested in doing this efficiently, thus we desire a minimum tipping set. This is much more difficult given the combinatorial nature of player interactions. The minimum tipping set is then to be derived from those sets. Hence, the problem of finding the minimum tipping set is generally NP-complete. This has been shown via reductions to SAT class problems.[12]

## 1.7 Interdependent Airline Security Game (IDS)

### 1.7.1 History of Airline Security Game

While the study of game theory has become more popular in recent years, it has a long history of study and applications. Some ancient systems of law were based on what were later determined to be cooperative game systems, and min-max solutions first were documented in the early 1700's.[2] Game theory, as a discipline, has borrowed from fields of information theory, computer science, and mathematics.

IDS games have been developed and modeled to solve many problems throughout their history. After the Pan Am incident, these tools were turned to IDS models for the airline industry. Since then, the field has expanded quickly both in its applications and depth of its academic study.[18]

In particular, Heal and Kunreuther (hereafter H-K) developed a model, using game theory, to explain the adoption of of security methods. In this case they were concerned about entry points to baggage security, but the structure applies to a much wider set of applications, from technology investment, to subsidies or marketing.[13][18]

### 1.7.2 Airline Security Model

Let us formally define the model used for analysis of airline security by H-K. There are n players labeled 1 through $n$, each endowed with two pure strategies denoted by 1 (invest) and 0 ( not invest). A play $s$ is defined by the n-tuple $s = (s_{1,}, s_2, ..., s_n)$ where $s_i \in \{0, 1\}$ denotes the choice of pure strategy by player $1 \leq i \leq n$. Clearly $0 \leq 1$. There are a total of $2^n$ distinct plays denoted as follows:

$$S = \{s | s = (s_1, s_2, ..., s_n), s_i \in \{0, 1\}\}$$

For $a, b \in S$, define a binary relation $\leq$ as follows:

$a \leq b$ (or $b \geq a$ ) when $a_i \leq b_i$ for $1 \leq i \leq n$.

$(S, \leq)$ is a poset and is indeed a complete lattice[21]. Let $u : s \to R^n$ where $u(s) = (u_1(s), u_2(s), ..., u_n(s))$ denote the n-tuple of utility functions with $u_i : S \to R$ denoting the utility of the player $i$.

Let $S_{-i}$ denote a play $(s_1, ..., s_{i-1}, *, s_{i+1}, ..., s_n)$ and $(s_{-i}, 1_i)$ denote the play $(s_{1,}, ..., s_{i-1}, 1_i, s_{i+1}, ..., s_n)$.

We now define the airline security game which is predicated on the natural assumption that no one can die no more than once[9]. The utility, or the payoff, is defined in terms of losses and the goal of the players is to minimize these losses. Let $c_i > 0$ be the cost of investment (choice of strategy 1) in security by player $i$, $L_i > 0$ be the cost or loss due to a catastrophic incident, $p_i > 0$ be the probability that player $i$ will suffer a catastrophic loss due to his own inaction (choice of strategy 0) and $q_{ij}$ $(0 \leq q_{ij} \leq 1)$ be the probability that player $j$ will suffer a catastrophic loss due to inaction of player $i$. For later reference, define an $n \times n$ matrix $Q = [q_{ij}]$ with $q_{ii} = 0$. Clearly, the off-diagonal elements of Q define the interdependency among the n players. It can be shown [13]that, where $u_i^{(1)}$ is the average cost due to self action and $u_i^{(2)}$is the average cost due to the action of others, the total expected cost is given by:

$$u_i(s) = u_i^{(1)}(s) + u_i^{(2)}(s) \tag{5}$$

$$u_i^{(1)}(s) = s_i c_i + (1 - s_i) p_i L_i \tag{6}$$

$$u_i^{(2)} = (1 - (1 - s_i) p_i)(1 - \Pi_{j \neq i,}(1 - (1 - s_j) q_{ji})) L_i \tag{7}$$

where $\Pi_{i=1}^{k} a_i = a_1 a_2 ... a_k$ refers to the product of the $a_i$. Equation 7 gives the expected loss for a two player IDS game. An example of a three person airline game is given in Table 1 .

They are super-modular (monotone), which is a commonly exhibited behavior in real world applications. This also means that given extreme NE, the NE at $1^n$will be more efficient than at $0^n$. The equations above separate nicely into a dependent part and an interdependent part. The dependent part is the loss do to the action or inaction of the player, while the inter-dependent part is the loss due to the actions of other players.[9][10][13]

### 1.7.3 Three Player Game

It is helpful to examine this general model via an example. Examine the three player case, for players $i$, $j$, $k$. The above payoff formulas are:

$$u_i(s_{-i}, 0_i) = p_i L_i + (1 - p_i)(1 - \Pi_{j \neq i,}(1 - (1 - s_j)q_{ji}))L_i$$

$$u_i(s_{-i}, 1_i) = c_i + (1 - \Pi_{j \neq i,}(1 - (1 - s_j)q_{ji}))L_i$$

In the case when no one is investing, for player one these become:

$$u_1(s_{-1}, 0_1) = p_1 L_1 + (1 - p_1)(q_{21} + q_{31} - q_{21}q_{31})L_1$$

$$u_1(s_{-1}, 1_1) = c_1 + (q_{21} + q_{31} - q_{21}q_{31})L_1$$

Evaluating these equations for each player gives us the equations in Table .

### 1.7.4 Structural Significance of Airline Security Game

Binary games are interesting due to their relation with common mathematical structures. The state space of a game can be represented as a lattice and a hypercube. Here each state of a game, that is an element of S , is a vertex whose coordinates are the ordered strategic choices of the players. [7][14] Equilibria can be explained using topological fixed point theorems.[4] This can show connections between fields of topology and graph theory that may be further exploited to solve difficult problems.

Super-modular games provide structure to the game that manifest in many common problems.[7][19][20][21] These problems are still difficult enough to be of interest, and often reflective of realistic structures in competitive systems. Those which involve moving from one uniform state or standard to another, such as adoption of a policy, standard, or practice can have very real impact on society. This can also be seen in estimations of how temporary subsidies impact practices, which impacts economics and politics. This research is interested in the class of models arising from the airline interdependent security investment models [5][7][10][13][17]

## 1.8 Organizational note

Thus far we have explored some general game concepts and developed the basic tools needed to analyze relevant systems.

In chapter 2 we use these tools, and expand on them to analyze the problem of finding a minimum tipping sets. First we explore the details of the required background information. We then conduct an analysis of the game, including special cases. We then present a non-greedy approximation method to approximate the minimum tipping set.

In chapter 3 We present some empirical information on different games and explore the performance of the approximation algorithm.

Chapter 2

# 2 Literature

## 2.1 Introduction and Background

A central concept in game theory is equilibrium. There are several types of equilibrium, but the most common is Nash Equilibrium (NE). [7][15] An equilibrium point occurs when the selected play is a play from which no player is willing to deviate from his strategy alone. These NE are not unique, a game can have several. They are also not necessarily equal, as NE can exist with different levels of value may exist in which all the players, including NE that are strictly better, or dominate, other NE. We are interested in the transition from one equilibrium to another. Given that in a NE no player is willing to deviate alone, we are interested in finding a set of players who can move (or tip) the game to another NE. We are interested in a phenomenon called tipping. Tipping can be thought of as the movement of a game from one equilibrium state to another. The set of players needed to commit to the new state in order to move the entire system with them is called a tipping set, and the smallest such set, if it exists, is the minimum tipping set, which is a focus of this research.[10]

While the study of game theory has become more popular in recent years, it has a long history of study and applications. Game theory as been applied across many fields, solving many problems arising from both cooperative and competitive systems. Some ancient systems of law were based on what were later determined to be cooperative game systems. One simple type of solution, Min-max solutions, were first documented in the early 1700's. [2] John von Neumann's and John Nash's work in the early and mid-1900's launched game theory as a separate discipline with wide applications. Since then, the field has expanded quickly both in its applications and depth of its academic study.[18]

Games in general have been used to model a wide range of competitive systems with rational behavior.[8][9][10][13][16] They provide a framework for understanding an influencing large, seemingly chaotic systems with independent agents acting in their own best interest. While not always a reliable mechanism for predicting human behavior, game theory has proven useful in the prediction and optimization of rational agents, investment and business decisions, to auctions and evolutionary systems.[8] [16]

When we discuss games we are discussing system in which independent agents act in their own best interest and the results, or payoffs, of the system are dependent on the combinations of the players' choices. Recall from section 1.1 that such a system, or game, is said to be in an equilibrium state if it is in no player's best interest to change strategy alone. A play of a game is the players' choice of strategies from each player's strategy set. In this paper, the strategy set is $\{0,1\}$, where 0 means not investing or not taking action, and 1 means investing or taking action. A state of a game is the set of plays, one from each player. A tipping set is a set of players who, when motivated by external influences such as bribery or coercion, to change their strategy and move the game to another equilibrium state. [10]

A common way of evaluating these states is to treat them as nodes of a lattice, with dimension equal to the number of players. They can also be represented as a hypercube with a dimension equal to the number of players. This allows us to also leverage tools from established fields, such as mathematics, to analyze games.[7][14]

## 2.2 Super-modular games

With ever growing dependency through commerce and communication between various parts of the globe, there is a greater need for the analysis and understanding of interdependent security (IS) among interacting agents. IS has been successfully modeled using the framework of n-person (non-cooperative) game theory[15]. In a series of seminal papers[9][10][13], H-K describe these models and their applications to IDS games for airline security, vaccination games to prevent the spread of infectious diseases, etc. The special features of this model include the following: (a) each player is endowed with only two pure strategies or actions, invest (1), and not invest (0) in security measures; (b) the players are not allowed to use randomized strategies, which in turn implies that we are interested only in the Nash equilibria (NE) in pure strategies[13]; (c) the utility (negative of the cost) function for each player has two components. The first component is due to one's own action (1 or 0) and the second component is due to the action (1 or 0), of the other players. This second component is called the externality component, which in turn decides the level of interdependency among the players; and (d) a subset of players acting in collusion, by the clever choice of their own actions, can influence the externalities of other players not in the coalition so as to force them to change their choice of actions. This phenomenon, whereby one subset can exert influence over another is called tipping[7][15] or cascading.[10]

A condition for an n-person game to exhibit the tipping / cascading property is that the payoff (loss) function of the players must satisfy the increasing (or decreasing) differences property. This property is intimately associated with the super-modularity of the utility functions[10][16]. In this paper we work with the differences in the loss which are the negation of the payoff differences. Another common reference to this increasing differences property is to describe the game as monotone.

Super-modular functions and functions with increasing differences are defined on lattices[16]. In our case, the underlying binary lattice is defined by the $2^n$ binary strings of 1's and 0's under the partial order defined over the binary strings. It turns out this lattice is also a hypercube of dimension $n$.[7]

## 2.3 Airline Security Games

As discussed in section 1.8, H-K have written substantially regarding IDS models and their applications to specific fields of study, notably terrorist risks and airline security.[9][10][13]

H-K present a model (see below) for analyzing these risks and their impacts on adoption, player behavior, and tipping. In particular, they draw distinctions between individual and industry optimal behavior based on the model. This model allows them to apply tools such as, graphical and algebraic, of game theory to solve these real world problems. Based on airline industry data, they determine that acting on game theoretic analysis, the industry as a whole could save money through the strategic adoption of security procedures. This was represented by moving from the inefficient equilibrium without investment to an efficient equilibrium with universal investment. This improvement suggests substantial uses in policy, as illustrated by the Pan Am incident reaction[9].

Later, in 2006, H-K explored mathematical approaches to finding the tipping set in this problem.[10] Connecting the monotone nature of the game system with the super modularity of the system allows the increasing differences property to become of use. Letting each player have a strategy of either 0 or 1, and a position vector $S$, they examined the increasing differences property. Let

$u_i$ be the payoff function of the player $i$. These assumptions, along with the assumption of multiple, usually extreme equilibria, imply that there must exist a tipping subset. Making use of the natural order on the hypercube of strategy vectors, H-K have shown that for each agent $i$, the increasing differences yields the following equation, previously stated in equation 1:

$u_i(1_i, s'_{-i}) - u_i(0_i, s'_{-i}) \geq u_i(1_i, s_{-i}) - u_i(0_i, s_{-i})$

with strict inequality if $s'_{-i} > s_{-i}$

Intuitively, this means that the payoff of a player choosing to invest (moving from 0 to 1) increases, or does not decrease, if other agents choose to invest. This is quantified, as mentioned previously in equations 234 as:

$$\triangle_j(s_{-i-j}, 0_i) = u_j(s_{-i-j}, 0_i 1_j) - u_j(s_{-i-j}, 0_i, 0_j) \tag{8}$$

and

$$\triangle_j(s_{-i-j}, 1_i) = u_j(s_{-i-j}, 1_i, 1_j) - u_j(s, 1_i, 0_j) \tag{9}$$

Equations 8 and 9 combine as follows:

$$\Delta_{ij} = \triangle_j(s_{-i-j}, 1_i) - \triangle_j(s_{-i-j}, 0_i) =$$
$$(u_j(s_{-i-j}, 1_i, 1_j) - u_j(s_{-i-j}, 1_i, 0_j)) - (u_j(s_{-i-j}, 0_i 1_j) - u_j(s_{-i-j}, 0_i, 0_j)) \tag{10}$$

Equations 8, 9 and 10 provide a measure of the incentive to player $i$ when $j$ changes from 0 to 1.

They then assume that the systems have at least 2 NE, one at $0^n$ and one at $1^n$, with the latter dominating (payoffs are better for every player in $1^n$). To find the (minimum) tipping set, they attempt to find subsets of players who, when their only strategy is 1, motivates all other players to choose 1, then find the minimum such set.

Admitting the difficulty and complexity of this problem, assumptions are made to simplify the problem. H-K call it A1:

$\triangle_{ij}(s) = \triangle_{ik}(s) = \triangle_i(s) = \triangle_i$

They then give, in the form of Proposition 2, a method for ranking the players by this quantity to find the smallest tipping set.

Proposition 2: Given A1, if a smallest T-set exists, then for some integer F it consists of the first F agents when agents are ranked by the value of $\triangle_i$.

The first part of our recent work focuses on exploring the consequences of these two assumptions on the game structure, how to better exploit that structure to find minimum tipping sets, and separating A1 into its two assumptions and addressing them independently.

## 2.4 Model Analysis

The game $(S, u)$ is super-modular[21] if, for every player $i$, the following condition is satisfied:

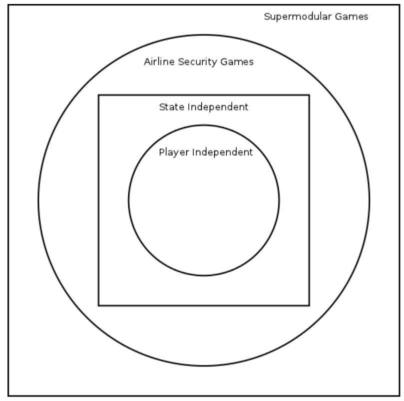
$$u_i(s'_{-i}, 1_i) - u_i(s'_{-i}, 0_i) \geq u_i(s_{-i}, 1_i) - u_i(s_{-i}, 0_i) \tag{11}$$

whenever $s'_{-i} \geq s_{-i}$. Intuitively, this decreasing difference condition states that the loss for player $i$ to change from strategy 0 to 1 does not increase when a subset of the other players has already moved from 0 to 1. Recall that the utility for two players is as follows:

$$u_i(s) = u_i^{(1)}(s) + u_i^{(2)}(s) \tag{12}$$

$$u_i^{(1)}(s) = s_i c_i + (1 - s_i) p_i L_i \tag{13}$$

Figure 1: Game Chart




$$u_i^{(2)} = (1 - (1 - s_i)p_i)(1 - \Pi_{j \neq i,}(1 - (1 - s_j)q_{ji}))L_i \tag{14}$$

where $\Pi_{i=1}^{k} a_i = a_1 a_2 ... a_k$ refers to the product of the $a_i$. An example of a three person airline game is given in Table 1.

which measures the change in the incentive for player j to change from 0 to 1 resulting from a similar change by player $i$. It can be verified that the super-modularity condition[5]implies that $\triangle_{ij}(s_{-i-j}) \geq 0$.

Following H-K[13] the airline security game is state independent (SI) if $\triangle_{ij}(s_{-i-j})$ is the same for all $s_{-i-j}$ where $s_{-i-j}$ denotes any one of the $2^{n-2}$ strategies by all other players except $i$ and $j$. Similarly, the above game is said to be player independent (PI) if $\triangle_{ij}$ is independent of the choice of $i$.

Here we only analyze the impact of SI on the tipping set. Our analysis is predicated on an assumption, namely we consider an $n$ person super-modular airline security game where there are exactly two NE, one at $0^n$and another at $1^n$. Recently, Dhall et al[7] have shown that a strict super-modular game can have up to $2^{n/2}$ NE. Analysis of tipping in super-modular games with more than two NE is wide open at this time.

Figure 2: Three Player Lattice

Binary Lattice of dimension 4

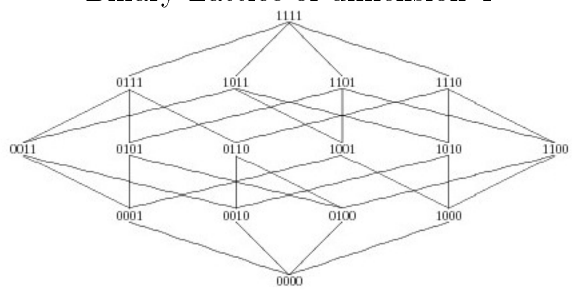



Table 1: Three Player Losses

Airline Game matrix for 3 players

| Play \ Cost | Player 1 | Player 2 | Player 3 |
|---|---|---|---|
| (000) | $p_1L_1 + (1-p_1)*(q_{21}+q_{31}-q_{21}q_{31})L_1$ | $p_2L_2 + (1-p_2)*(q_{12}+q_{32}-q_{12}q_{32})L_2$ | $p_3L_3 + (1-p_3)*(q_{13}+q_{23}-q_{13}q_{23})L_3$ |
| (001) | $p_1L_1 + (1-p_1)q_{21}L_1$ | $p_2L_2 + (1-p_2)q_{12}L_2$ | $c_3+(q_{13}+q_{23}-q_{13}q_{23})L_3$ |
| (010) | $p_1L_1 + (1-p_1)q_{31}L_1$ | $c_2+(q_{12}+q_{32}-q_{12}q_{32})L_2$ | $p_3L_3 + (1-p_3)q_{13}L_3$ |
| (011) | $p_1L_1$ | $c_2 + q_{12}L_2$ | $c_3 + q_{13}L_3$ |
| (100) | $c_1+(q_{21}+q_{31}-q_{21}q_{31})L_1$ | $p_2L_2 + (1-p_2)q_{32}L_2$ | $p_3L_3 + (1-p_3)q_{23}L_3$ |
| (101) | $c_1 + q_{21}L_1$ | $p_2L_2$ | $c_3 + q_{23}L_3$ |
| (110) | $c_1 + q_{31}L_1$ | $c_2 + q_{32}L_2$ | $p_3L_3$ |
| (111) | $c_1$ | $c_2$ | $c_3$ |

## 2.5 Analysis

Our recent work focuses on two approaches to exploring this model. First, we further analyze the assumptions proposed by HK and focus on tipping in the State Independent cases of the Airline Security game. We present a three player example and a four player example before analyzing the general case of SI games. We then turn to the more complex general case games, and an algebraic method for approximating the minimum tipping set.

## 2.6 Analysis of SI

To illustrate the key features of our approach to finding the minimum tipping set in a super-modular airline security game satisfying the SI condition, we consider the n=3 and n=4 player cases.

### 2.6.1 Three player game

Consider the case when i=2 and j=1. Since there are only three players, $s_{-i-j}$ takes on only two values, 0 and 1. The SI condition stated above becomes:

$$\triangle_{21}(0_3) = \triangle_{21}(1_3) \tag{15}$$

Using the payoff values from Table 1 and the expressions (8) - (10) the above equality (15) becomes:

$$\begin{aligned}[c_1 + q_{31}L_1] - [p_1L_1 + (1-p_1)q_{21}L_1] = [c_1 + (q_{21} + q_{31} - q_{21}q_{31})L_1] \\ - [p_1L_1 + (1-p_1)(q_{21} + q_{31} + q_{21}q_{31})]\end{aligned} \tag{16}$$

When simplified, the SI condition (16) becomes

$$(p_1L_1)q_{21}q_{31} = 0 \; or \; q_{21}q_{31} = 0 \tag{17}$$

because $p_1L_1 > 0$ by assumption.

Clearly, there are $3 * 2 = 6$ pairs of $(i, j)$. Considering the other five possible conditions imposed by the SI condition, we get $\triangle_{12}(0) = \triangle_{12}(1)$; $\triangle_{13}(0) =$

$\triangle_{13}(1)$; $\triangle_{21}(0) = \triangle_{21}(1)$; $\triangle_{23}(0) = \triangle_{23}(1)$ *and* $\triangle_{32}(0) = \triangle_{32}(1)$. Again using the relations (8) - (10) and the entries in Table 1 we obtain the following three conditions:

$$q_{21}q_{31} = 0,\ q_{12}q_{32} = 0,\ q_{13}q_{23} = 0 \tag{18}$$

(18) Verbally, the SI condition requires the product of the off-diagonal elements in each column of the 3x3 matrix Q to be zero. Suppose we set both $q_{12} = 0 = q_{32}$, then the second column of Q is zero, which implies that player 2 does not have externality and is not affected by any other player. This choice defines a degenerate case where player 2 can be dropped from the consideration for the tipping set. To avoid this degeneracy, it is assumed that in each column of Q, there is exactly one non zero off-diagonal entry in each column of Q.

Guided by this non-degeneracy requirement, it follows that the condition (18) can be enforced modulo a permutation in one of two ways:

$$q_{21} = q_{12} = q_{13} = 0\ and\ q_{31} \neq 0,\ q_{32} \neq 0,\ q_{23} \neq 0 \tag{19}$$

and

$$q_{21} = q_{32} = q_{13} = 0\ and\ q_{31} \neq 0,\ q_{32} \neq 0,\ q_{12} \neq 0 \tag{20}$$

Conditions (19) and (20) can be visually depicted as a directed graph with 3 nodes and 3 directed edges where the in degree of each node is one, as shown in Figures 3a and 3b respectively. It is fitting to call these influence graphs. Clearly any node in the 2-cycle in Figure 3a and any node in the 3 cycle in Figure 3a is a minimum tipping set.

**Remark: Impact of SI on influence graphs:** Without the SI condition, the influence graph is a complete directed graph. In this case one must consider the set of all $n!$ possible orderings of $\triangle_j$'s in (8) to find the smallest tipping set.

**Remark: Impact of SI on Nash equilibria:** For the general 3-player

Figure 3: Three Player Graphs

(a) Three Player Branch

(b) Three Player Cycle


game in Table 1, conditions for NE at 000 and 111 are given by

$$0 \leq 1 - \frac{c_1}{p_1 L_1} \leq q_{21} + q_{31} - q_{21}q_{31};\ 0 \leq 1 - \frac{c_2}{p_2 L_2} \leq q_{12} + q_{32} - q_{12}q_{32}$$

*and*

$$0 \leq 1 - \frac{c_3}{p_3 L_3} \leq q_{13} + q_{23} - q_{13}q_{23}$$

(21)

Under the SI conditions in (19) , the above condition for NE in (21) becomes

$$0 \leq 1 - \frac{c_1}{p_1 L_1} \leq q_{31},\ 0 \leq 1 - \frac{c_2}{p_2 L_2} \leq q_{32},\ 0 \leq 1 - \frac{c_3}{p_3 L_3} \leq q_{23} \qquad (22)$$

Clearly (22) implies (21) but not vice versa. Because our analysis is predicated on the condition that we have two NE at 000 and 111, the conditions for the tipping set under SI in general do not carry over to general airline games. In other words, finding the minimum tipping set in general super-modular games is more complex than those with SI. Similar arguments can be made using the SI condition (20) in (21).

Table 2: Four Player Interdependence Matrix

| 12 | 13 | 14 | 21 | 23 | 24 | 31 | 32 | 34 | 41 | 42 | 43 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $q_{12}$ | $q_{13}$ | $q_{14}$ | $q_{21}$ | $q_{23}$ | $q_{24}$ | $q_{31}$ | $q_{32}$ | $q_{34}$ | $q_{41}$ | $q_{42}$ | $q_{43}$ |
| or | or | or | or | or | or | or | or | or | or | or | or |
| $q_{32}$ | $q_{23}$ | $q_{24}$ | $q_{31}$ | $q_{13}$ | $q_{14}$ | $q_{21}$ | $q_{12}$ | $q_{14}$ | $q_{21}$ | $q_{12}$ | $q_{13}$ |
| $q_{42}$ | $q_{43}$ | $q_{34}$ | $q_{41}$ | $q_{43}$ | $q_{34}$ | $q_{41}$ | $q_{42}$ | $q_{24}$ | $q_{31}$ | $q_{32}$ | $q_{23}$ |

### 2.6.2 The Four Player Game

Setting i=2 and j=1, since $s_{-i-j}$ has four possible values, 00, 01, 10, and 11. The SI conditions becomes:

$$\triangle_{21}(00) = \triangle_{21}(01) = \triangle_{21}(10) = \triangle_{21}(11) \tag{23}$$

By combining the relations (8) - (10), with the payoff for the 4 player airline games[7], the SI condition takes the following form:

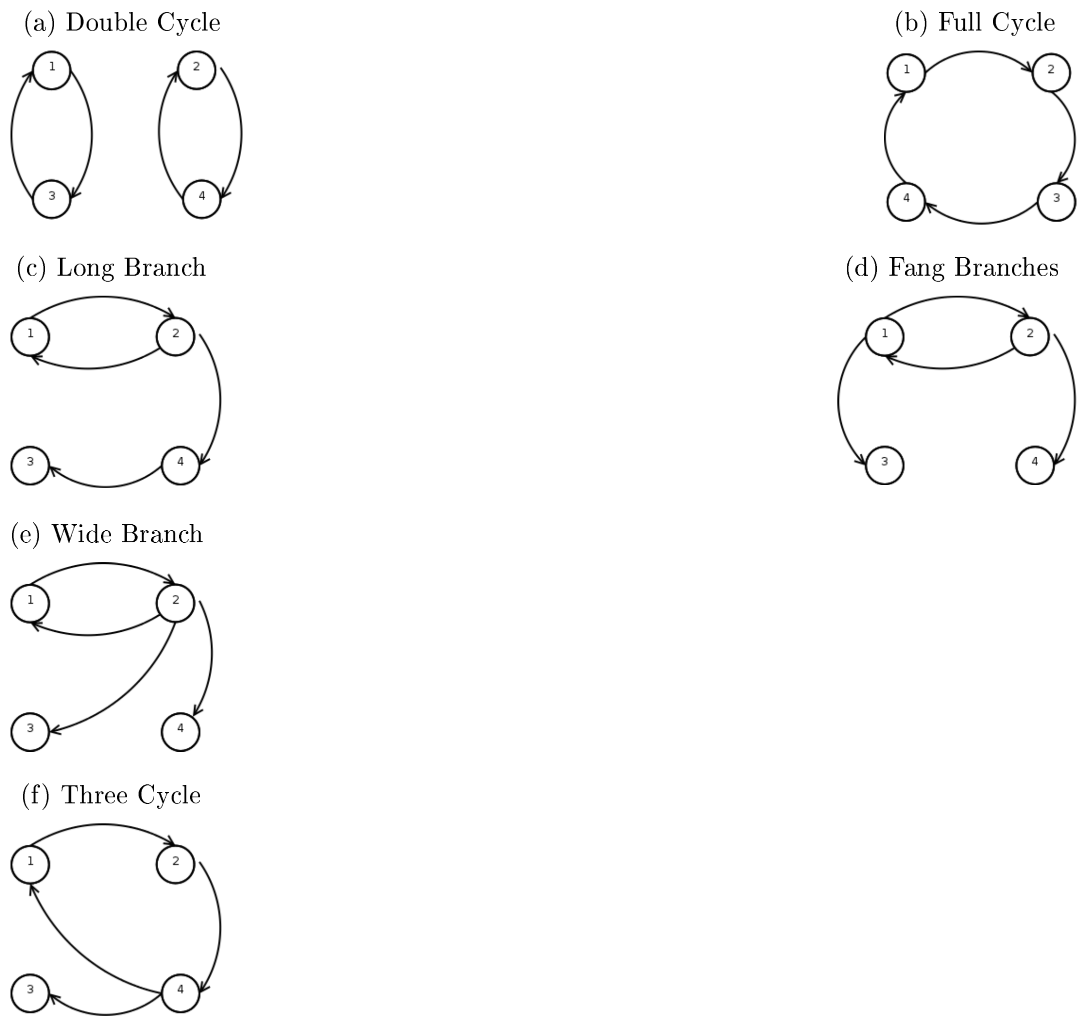
$$either \;\; q_{12} = 0 \; or \; q_{32} = q_{42} = 0 \tag{24}$$

Clearly there are 4x3=12 pairs $(i, j)$ with $1 \leq i, j \leq 4$. Considering each of these pairs in turn we get a condition similar to (24). For completeness these 12 conditions are given in Table 2 where in each column either the element in the first row is zero or those in the second and third rows are zeros.

Combining these with the non-degeneracy requirement, it turns out that there are exactly six distinct patterns modulo a permutation for the SI conditions. The resulting six influence graphs are given in Figure 4. In Figure 4a there are two 2-cycles and hence the minimum tipping set has two players, one from each two cycles. In all the other five cases in Figure 4, the graph is weakly connected. In each of those cases, any player who is a member of the directed cycle constitutes a minimum tipping set.

**Remark:** In any given realization of the game satisfying the SI condition, only one of these influence graphs will be applicable. So, in general one could use the standard depth first search to identify the cycles in the influence graph. The minimum tipping set is then made of one node from each cycle. This idea readily generalizes to the n-person game.

Figure 4: Four Player Influence Graphs



## 2.7 Analysis of SI - General case

### 2.7.1 Extreme Equilibria

Many of the tools used to analyze tipping in super modular games are to be applied only when tipping from one extreme equilibria ($0^n$) to another ($1^n$). Many game cases do not exhibit this behavior. Here we give methods for reducing SI games without extreme equilibria to games with extreme equilibria in order to apply our well known tools.

Given a SI super-modular game without an equilibria at $1^n$, we can use repeated play from $1^n$ or some other method to find an equilibria at the highest level of the the lattice. That is, an equilibria with the largest number of investing players. By super-modularity, the nodes that will not tip when all others investing will never invest. Thus they can be removed from the analysis and assumed to never invest for the payoff calculations of other players. This could disconnect other players, who would be removed in the same fashion. This yields a new game with an equilibrium at $1^{n-v}$ where $v$ is the number of players not investing in the top equilibrium.

Against the back drop of these examples, we now state the general results. We begin with a definition. An n-person airline super-modular game is non-degenerate if there is exactly one non-zero off-diagonal element in each column of the nxn matrix $Q = [q_{ij}]$ with $q_{ii} = 0$. Our first result relates the SI condition to the non-degeneracy defined above.

### 2.7.2 General Airline Security

**Lemma 1:** SI holds iff each player is effected by at most one other player.

**Proof**:

First we show that State Independence implies if each player is effected by at most one other player.

Let $S$ be the state of a super modular game for which the State Independence holds. That is,

$\triangle_{ij}(s_{-i-j}) = p_j q_{ij} \prod_{k \notin \{S\}, k \neq i,j}(1-q_{kj}) = \triangle_{ij}(Q_{-i-j}) = p_j q_{ij} \prod_{k \notin \{Q\}, k \neq i,j}(1-q_{kj}) \forall Q$

Now, let $q_{ij} \neq 0$. Given any other player, called $l$, we see that the following must hold:

$\triangle_{ij}(1_{-i-j-l}, 1_l) = p_j q_{ij} = \triangle_{ij}(1_{-i-j}, 0_l) = p_j q_{ij}(1 - q_{lj})$

The $k$ is excluded so that we can see that it is included in the second product, but not the first. Thus we have:

$p_j q_{ij} = p_j q_{ij} - q_{lj}(p_j q_{ij})$

Which, along with the previous assumptions that $p_j \neq 0$ implies:

$-q_{lj}(p_j q_{ij}) = 0 \implies q_{lj} = 0$

Thus State Independence implies the Independence Condition.

Now to show that if each player is effected by at most one other player, this implies State Independence. Assume we have a game as defined above for which the above condition holds. Let us examine

$\triangle_{ij}(s_{-i-j}) = p_j q_{ij} \prod_{k \notin \{S\}, k \neq i,j}(1 - q_{kj})$

If $q_{ij} = 0$ then all $\triangle_{ij} = 0$ and the Independence is trivial, so we consider the case when $q_{ij} \neq 0$. That and the condition that each player is effected by at most one other player implies that all other $q_{kj} = 0 \forall k \neq i$ which means that the product terms must be the multiples of 1's regardless of state, so the game has State Independence. Therefore State Independence and the condition that each player is effected by at most one other player are equivalent.

$\square$

Recall from above that the non-degeneracy condition can be visually depicted by the influence graphs (IG). Formally, an IG is a directed graph $G = (V, E)$ with $|V| = n$ and $|E|$ =n with the added requirement that the in degree of each node is one. Thus every pattern of $q_{ij}$ satisfying the non-degeneracy condition fits within an equivalence class of influence graphs where the members of an equivalence class are obtained by graph automorphisms.

Thus, for n=3, there are exactly two equivalence classes (see Figure 3) and there are only six equivalence elements for n=4 (see Figure 4). It can be verified that for n=5, there are exactly thirteen equivalence classes as shown in Table 3.

At this time, enumerating the number of equivalence classes of influence graphs for any n is open. However, we can readily catalog many of the key structural properties of these graphs. 1) Given any integer $n \geq 3$, there are exactly $(n-1)$ subsets of influence graphs, one corresponding to each value of the parameter $2 \leq k \leq n$. 2) For a given k in this range, the $k^{th}$ subset contains graphs with k-cycles. The number $n_k$ of k-cycles in a graph in this subset can vary in the range $1 \leq n_k \leq \lfloor \frac{n}{k} \rfloor$ 3) In a graph with $n_k$ cycles, the rest of the $(n - kn_k)$ nodes not included in a cycle are attached to the $n_k$ cycles in an arbitrary manner or else they would not be connected. An example of this enumeration for n=5 is given in Table 3.

We now state our main result relating to the properties of the minimum tipping set in games satisfying the SI condition.

**Theorem 1:** In an n-player game with SI, 1) The cardinality, |T| of the tipping set T is equal to the number of cycles in the influence graph induced by the non-degeneracy condition. Clearly, $1 \leq |T| \leq \frac{n}{2}$. 2) While any node in a given cycle would tip all the players associated with that cycle, the node with the longest out degree will bring about the tipping in the shortest time.

As an illustration, refer to Figure 5. If we pick P1 as the minimum tipping set, P1 will first force P2 to change who in turn will simultaneously influence P3 and P4. On the other hand, if we pick P2 as the minimum tipping set, then this will simultaneously force all the other to tip.

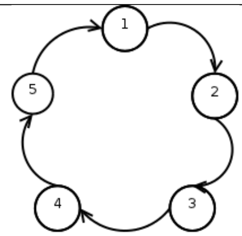
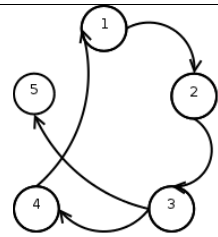
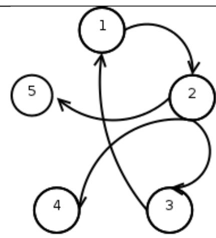
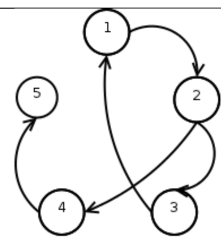
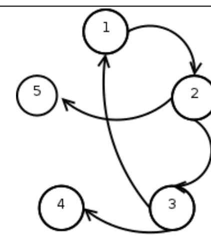
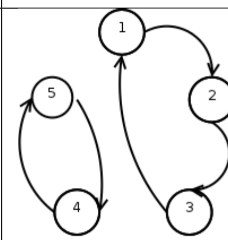
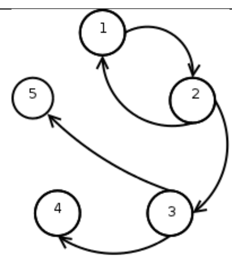
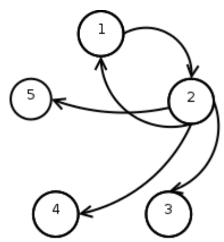
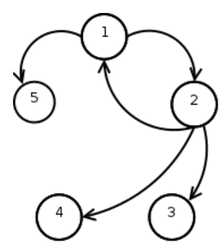
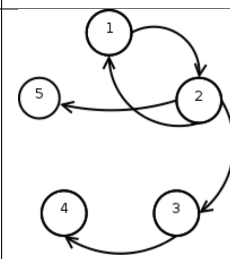
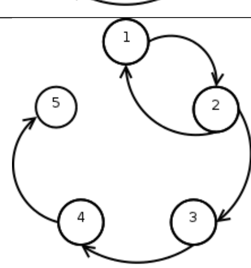
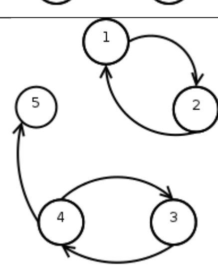
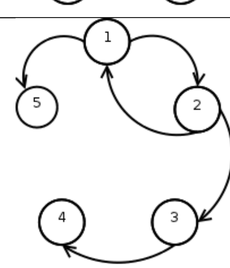
## 2.8 Complexity and Approximation of Tipping in a Super modular game

Analysis of tipping is concerned with the difference

$$u_i(s_{-i}1_i) - u_i(s_{-i}0_i) \tag{25}$$

where $(s_{-i}, 1_i, s_{-i}, 0_i)$ is an edge in the complete lattice considered as a hypercube. A sequence of differences along a path in the hypercube connecting the NE at $0^n$ to the one at $1^n$ is given by

Table 3: 5 Player Graphs

| k | | | | | |
|---|---|---|---|---|---|
| 5 | $n_5 = 1$ |  | | | |
| 4 | $n_4 = 1$ |  | | | |
| 3 | $n_3 = 4$ |  |  |  |  |
| 2 | $n_2 = 7$ |  |  |  |  |
| 2 | $n_2 = 7$ |  |  |  | |

$$u_i(0^{n-1}1_i) - u_i(0^{n-1}0_i) \geq u_i(0^{n-2}1_i1_j) - u_i(0^{n-2}1_i0_j) \geq ... \geq u_i(1^{n-1}1_i) - u_i(1^{n-1}0_i) \tag{26}$$

Since $0^n$ and $1^n$ are NE, clearly $u_i(0^{n-1}1_i) - u_i(0^{n-1}0_i) > 0$ and $u_i(1^{n-1}1_i) - u_i(1^{n-1}0_i) < 0$. That is, the above sequence of decreasing real numbers starts from a positive value and ends up at a negative value. Thus there is a zero crossing point that corresponds to a subset $\{k_1, ..., k_i\}$ players choosing pure strategy 1. But there are $(n)!$ distinct sequences corresponding to $(n)!$ disjoint paths in the underlying hypercube. Hence, we get $(n)!$ subsets, one for each path. The minimum tipping set is then to be derived from those sets. Hence, the problem of finding the minimum tipping set is generally NP-complete.[12]

HK proved that for the special case satisfying two conditions (player and state independence assumptions) there exists a minimum tipping set. Above we gave a simpler algorithm to find such a tipping set for this special case. Here our goal is to find an approximation to the minimum tipping set for the general case.

### 2.8.1 An Approximation Algorithm - Hypercube step

Recall that a given decreasing sequence of inequalities of the differences in the losses uniquely induces a mapping of the sequences in (26) onto edges in the hypercube of dimension n as follows:

$$(0^{n-1}1_i, 0^{n-1}0_i) \xrightarrow{1} (0^{n-2}1_i1_1, 0^{n-2}0_i1_1) \xrightarrow{2} (0^{n-3}1_i1_11_2, 0^{n-3}, 0_i1_11_2) ... \xrightarrow{(n-1)} (1^{n-1}1_i, 1^{n-1}0_i) \tag{27}$$

where clearly $(0^{n-1}1_i, 0^{n-1}0_i)$is an edge in the hypercube [19]. Thus, one end (say the right end) of the starting edge in the above sequence of edges traces a path living entirely in one sub-cube of dimension$(n-1)$ whose $i^{th}$ bit is fixed at 0, namely

$$0^{n-1}0_i \xrightarrow{1} 0^{n-2}0_i1_1 \xrightarrow{2} 0^{n-3}0_i1_11_2... \xrightarrow{(n-1)} 1^{n-1}0_i \tag{28}$$

This path is from the NE $0^{n-1}0_i$ to a neighbor $1^{n-1}0_i$ of the NE $1^n$. Similarly, the left end traces a corresponding path in the complementary sub-cube of dimension $n-1$ defined by $i^{th}$ bit equal to 1, namely

$$0^{n-1}1_i \xrightarrow{1} 0^{n-2}1_i1_1 \xrightarrow{2} ... \xrightarrow{(n-1)} 1^{n-1}1_i \tag{29}$$

Here we have a path from the neighbor $0^{n-1}1_i$ of the NE at $0^n$ to the NE at $1^n$. Our approximation algorithm relies on the well-known topological property of node disjoint path distribution in a hypercube with nodes representing states of the game.

**Theorem 2** [14]: Let x and y be nodes in a hypercube of dimension k where the Hamming distance between x and y, $H(x,y) = r$ for some $1 \leq r \leq k$. Then (a) there are exactly r node disjoint paths each of length r between x and y (b) there are exactly $(k-r)$ node disjoint paths each of length $r+2$ and (c) The set of all paths in groups a and b are node disjoint.

We apply this theorem to the two nodes $x = 0^{n-1}1_i$and $y = 1^{n-1}1_i$ with Hamming distance $n-1$ in the sub-cube of dimension $n-1$ whose $i^{th}$ bit is fixed at 1. Therefore, there are r=n-1 node disjoint paths each of length $n-1$ in that sub-cube.

Thus, if we denote the path in (28) succinctly as

$$0^{n-1}0_i\{1,2,...,n-1\}1^{n-1}0_i \tag{30}$$

where the {} term is the order of dimensions along which you move to the next node in the path from $0^{n-1}0_i$ to $1^{n-1}0_i$. It can be shown [14] that by left circular shift the elements in the ordered set we can generate the n-1 node disjoint paths in the n-1 dimensional sub-cube. These are given by

$$0^{n-1}0_i \left\{ \begin{array}{cccccc} 1 & 2 & 3 & ... & n-2 & n-1 \\ 2 & 3 & 4 & ... & n-1 & 1 \\ 3 & 4 & 5 & ... & 1 & 2 \\ . & . & . & ... & . & . \\ . & . & . & ... & . & . \\ . & . & . & ... & . & . \\ n-1 & 1 & 2 & ... & n-3 & n-2 \end{array} \right\} 1^{n-1}0_i \tag{31}$$

Likewise there is a corresponding sequence of paths in the other sub-cube given by

$$0^{n}1_i \left\{ \begin{array}{cccccc} 1 & 2 & 3 & ... & n-2 & n-1 \\ 2 & 3 & 4 & ... & n-1 & 1 \\ 3 & 4 & 5 & ... & 1 & 2 \\ . & . & . & ... & . & . \\ . & . & . & ... & . & . \\ . & . & . & ... & . & . \\ n-1 & 1 & 2 & ... & n-3 & n-2 \end{array} \right\} 1^{n}1_i \tag{32}$$

Thus, for each player i, there are $n-1$ distinct increasing sequences of inequalities that start from a negative number and end in a positive number.

Hence, for all $n$ players, there are a total of $n(n-1) = O(n^2)$ sequences and hence $O(n^2)$ subsets of players who are potential candidates for the tipping set. Our goal is to find a minimum tipping set from these $O(n^2)$ candidate subsets.

### 2.8.2 An Example

Consider an example of a $n = 4$ player airline game with the following values of the parameters: $L_i = 1000$, $c_i = 99$, $p_i = 0.1$ for all $1 \leq i \leq 4$ and the matrix of $q_{ij}$'s given by

$$q = \begin{bmatrix} 0 & .99 & .02 & .02 \\ 1 & 0 & 0 & 0 \\ 0 & .99 & 0 & .02 \\ 0 & .99 & .02 & 0 \end{bmatrix}$$

Then using the expression in (14) we can readily compute the $16 \times 4$ payoff (or loss in our case) table for each of the $2^4 = 16$ plays for each of the 4 players. For lack of space, we will not explicitly show this table. It can be verified that for this choice of parameter values, $0^4$ and $1^4$ are two NE with $1^4$ being better than $0^4$. $1^4$ is optimal in this case. Applying the approximation algorithm developed previously, we get three disjoint sequences of inequalities induced by the three node disjoint paths for Player 1 as follows. Referring to the hypercube in Figure 2 the first disjoint sequence is given by

$$u_1(1000) - u_1(0000) \geq u_1(1100) - u_1(0100) \geq u_1(1110) - u_1(0110) \geq u_1(1111) - u_1(0111)$$

which for the above example becomes $99 \geq -1 \geq -1 \geq -1$.

It follows that $\{2\}$ is a candidate tipping set for player 1. That is, if player

Table 4: First Candidates

Candidate tipping sets

| candidate tipping sets | Path 1 | Path 2 | Path 3 |
|---|---|---|---|
| $T1$ | $\{2\}$ | $\{2,4\}$ | $\{2,3,4\}$ |
| $T2$ | $\{1,3,4\}$ | $\{1,3,4\}$ | $\{1,3,4\}$ |
| $T3$ | $\{1,2,4\}$ | $\{1,4\}$ | $\{1,2,4\}$ |
| $T4$ | $\{1,2,3\}$ | $\{1,3\}$ | $\{1,2,3\}$ |

2 changes his strategy from 0 to 1, some other players can reduce their losses by switching from 0 to 1. Thus, player 2 has influence over those other players.

Similarly, the second sequence

$$u_1(1000)-u_1(0000) \geq u_1(1001)-u_1(0001) \geq u_1(1101)-u_1(0101) \geq u_1(1111)-u_1(0111)$$

which for the above example becomes $99 \geq 99 \geq -1 \geq -1$

from which we get $\{2,4\}$ as a candidate tipping set. From the third sequence

$$u_1(1000)-u_1(0000) \geq u_1(1010)-u_1(0010) \geq u_1(1011)-u_1(0011) \geq u_1(1111)-u_1(0111)$$

which leads to $99 \geq 99 \geq 99 \geq -1$.

from which we get {2,3,4} as a candidate tipping set.

By repeating the above procedure for Players 2, 3, and 4, we can obtain three candidate tipping sets for each of the players, which is summarized in Table 4

Our goal is to find a minimum tipping set from these $n(n-1) = 12$ candidate tipping sets.

In the following Section, we describe a simple algebraic method for extracting a minimum tipping set.

### 2.8.3 An Approximation Algorithm - Algebraic Step

Above, we transformed similar facts about state independent games into a graph form to examine the influence patterns. We constructed these influence graphs by representing each player by a node $i$ with an edge from player 2 to player 1 if player 2's choice to invest impacts or influences player 1 directly.

Generating such a graph for this example gives us the graph in Figure 5. General super modular games lack some of the structure we exploited in state independent games. Unlike state independent games, one node's influence is not enough to tip given node. In this example, all of the influencing nodes are needed to tip a node, thus adding complexity. To address this added challenge, we have developed an algebraic method for finding a minimum tipping set.

Let $\Sigma = \{0, 1\}$, let $+$ and $*$ denote the usual logical AND and OR operators respectively. For simplicity, $a * b$ is denoted by $ab$. Then, $(\Sigma, +, *)$ defines a commutative semi-ring. For all $a, b \in \Sigma$, we have $a + a = a$, $aa = a$, $a + ab = a(1 + b) = a$.

Let $a, b, c, d$ be the four binary variables corresponding to players 1 though 4 respectively. We now encode a subset $\{1, 2, 4\}$ of players by $abd$. Accordingly, the information in Table 4 can be encoded as follows:

$$\begin{aligned} T1 &= b + bd + bcd = b \\ T2 &= acd + acd + acd = acd \\ T3 &= abd + ad + abd = ad \\ T4 &= abc + ac + abc = ac \end{aligned} \tag{33}$$

These expressions represent the conditions for each player to choose to invest. T1 gives the conditions for player $a$, T2, for $b$, and so on.

First, we can algorithmically trim the number of players to examine. While this is not needed in order to get an answer, it improves both the accuracy of the approximation and the typical run time.

1) Any case in which a player's action is determined entirely by a single other player, we can collapse those players. In the context of the game, this is because the influences caused by the impacting player contains the influences of the other. In this example, $b$ completely determines the action of $a$. This means that the impact of $b$ contains the entire impact of $a$, and that when picking minimum tipping sets, there is no reason to pick $a$. Collapsing $a$ into $b$ requires moving $a$'s influences to $b$. This is to make sure that $b$ now influences all players that $a$ previously did. This should be done one at a time in order to stop cycles when a node is influenced only by itself.

If no such players exist, we simply move on to the next step. Then from 33 we get:

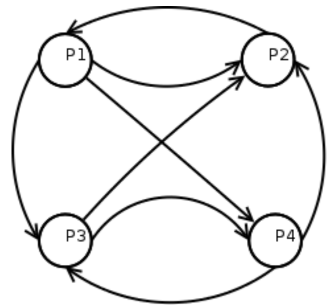
$$T2 = bcd; T3 = bd; T4 = bc$$

2) We also do not want to double count a player. If a player is externally motivated to invest, we do not care about the conditions to tip that player. To reflect that, we add to each expression the set consisting only of that player. This yields:

$$T2 = b + bcd = b;\ T3 = c + bd;\ T4 = d + bc$$

This canceling and trimming procedure can be thought of as a variant of Unit Propagation, which is a technique used in solving satisfiability problems.

Figure 5: Multi-Graph




Now we are ready for the core procedure. We want the value of each of these expressions to be 1, which is the same as their product being 1. Thus we want to find the minimum subset variables to set to 1 such that

$T2 * T3 * T4 = 1$

Substituting from above we get:

$(b)(c + bd)(d + bc) = 1$

$(bc + bd)(d + bc) = (bcd + bc + bd + bcd) = 1$

Thus we have candidate tipping sets of $bcd$, $bc$, $bd\,bcd$. Any of these will give a tipping set, but since we are interested in minimum tipping sets, we take one of the smaller sets $bc$ or $bd$. to choose for our minimum set.

## 2.9 Approximation Complexity

The algebraic step of the approximation presented is far from trivial. In fact, we are approximating what is known as the Minimum Monotone Satisfiability Assignment (MMSA)[6]

An instance of the MMSA problem is a monotone formula $\tau$ over a basis of logical-and and logical-or (or disjunction and conjunction). A solution takes the form of an assignment of the independent variables of $\tau$ such that $\tau$ is true. The objective function that is minimized is the number of variables in the assignment which are equal to true.[1]

The transformation to this problem is trivial. The combined equation $T2 * T3 * T4 = 1$ is exactly the required tautology, with 1 representing true and the Boolean semi-ring over * and + is exactly the logical operators for the MMSA problem. The length of the propositions is bound by a function of the number of players. Thus we have an easy transformation to $MMSA_{n>3}$.

This problem is not only a complex problem to solve, but it is complex to approximate within a factor of $2^{log(1-o(1)}n$ [1] In graph terms, this is equivalent to solving (or approximating) the label-cover of a bipartite graph.[6] This is still an improvement, just not to the worst case. The run time of these approximations are much faster than the original NP problem we are addressing. This is due, in large part, the algebraic collapsing method above and is similar to the use of Unit Propagation in SAT problems.

Chapter 3

# 3 Applications

## 3.1 Airline Security - The Uniform Case

One special form of the Airline Security game is the uniform case. In this case, the agents behave identically with

$$p_i = p_j = p, c_i = c_j = c, L_i = L_j = L, q_{ij} = q_{kl} = q \tag{34}$$

By examining the behavior of the approximations in this case along with the case's properties, we gain some insight regarding the performance of the method. The most relevant property of the uniform game is the following:

**Lemma 2:** A uniform game can be tipped by any set S of players with $|S| > n - 1 - \frac{log(c/pL)}{log(1-q)}$ .

**Proof**: Generally, the decision of player $i$ to invest is determined by:

$$p_i L_i + (1 - p_i) Q_i(S) L_i - (c + Q_i(S) L_i) > 0 \tag{35}$$

where $Q_i(S) = 1 - \prod_{i \neq j, j \notin S} (1 - q_{ji})$

If 35 is true, the player $i$ will invest.

35reduces through simplification as follows:

$p_i L_I - c + Q_i(S) L_i - p_i Q_i(S) L_i - Q_i(S) L_i > 0$

$p_i L_i - c - p_i Q_i(S) L_i > 0$

$p_i L_i (1 - Q_i(S)) > c$

$1 - Q_i(S) > \frac{c_i}{p_i L_i}$

In a uniform game, this becomes:

$1 - Q(S) > \frac{c}{pL}$ with $Q(S) = 1 - (1-q)^{n-1-|S|}$

Note that this quantity is independent of identity, and only depends on p,c,L, q, and the number of investing players in S.

Let $n - 1 - |S| = k$ Then the condition becomes: $1 - (1 - (1-q)^k) = (1-q)^k > \frac{c}{pL}$

Since all are constant except for k, we can see that we can determine the number of players needed to tip the system as:

$k * log(1-q) > log(\frac{c}{pL})$

$k < \frac{log(c/pL)}{log(1-q)}$

$n - 1 - |S| < \frac{log(c/pL)}{log(1-q)}$

$|S| > n - 1 - \frac{log(c/pL)}{log(1-q)}$

This is the same for all players. Thus if one player is tipped by $|S|$ investing players, all the remaining ones will be as well thus there can be no cascades.

Note that all players payoff is the same, so the choice of which players are chosen is irrelevant. Any $|S|$ players will tip the system.

□

**Corollary 1:** In the uniform game, there can be no cascades.

**Proof**: By above lemma, all players are tipped by any |S| other players. Less than |S| players will not tip any single player by the construction of the condition from a single activation function. If it were to tip a single player then all the other players would make the same decision and tip, thus giving a tipping set less than |S|.

□

**Lemma 3:** The hypercube left shift sampling will find a set of n-1 groups of |S| consecutive players to tip each individual player.

**Proof:** Since any s=|S| players will tip a given player, the first set of length s that is tested will tip the player, thus the first length s consecutive player set is chosen. Now, as per the algorithm, we left shift our starting point, then repeat. This gives us another length s set that is shifted one player to the left. Thus we end up with n-1 length s tipping sets for each player.

□

**Theorem** 3 Given the above a uniform game and n-1 groups of |S| players tipping each single player, the algebraic simplification will find the optimal solution.

**Proof**: Given an order there are only n different length s contiguous sets of players. Each player has the n-1 that do not contain themselves. Thus given any length s candidate set S, it will be contained in the possible candidates list for all the players not contained in S.

This along with simple Boolean algebra, gives the correct result. That is, the set S will tip all players except the ones in S, which would be externally motivated by having selected S as the tipping set.

□

**Example**

Now we look at how the approximation system behaves in this context. Since any k players will tip any player. For players a,b,c, and d, set the following parameters:

$$p = 0.1, c = 99, q = .009, L = 1000 \tag{36}$$

For clarity, examine what happens when |S|=2. This means that any 2 players will tip any player, thus each player's set of candidate tipping sets will be the first two corrected players and the corresponding first two from every left shift. So for players a,b,c,d, we would have:

$$a : bc, cd, bd$$

$$b : cd, ad, ac$$

$$c : ab, bd, ad$$

$$d : ab, bc, ac$$

Now, when we apply the algebraic method on this set, we get:

$$\begin{aligned} a &= a + bc + cd + bd \\ b &= b + cd + ad + ac \\ c &= c + ab + bd + ad \\ d &= d + ab + bc + ac \end{aligned}$$

This system of equations simplify as:

$$a * b * c * d = 1$$

$$(a + bc + cd + bd)(b + cd + ad + ac)(c + ab + bd + ad)(d + ab + bc + ac) = 1$$

$$(ab + ad + ac + bc + cd + bd)(cd + bc + ac + ab + bd + ad) = 1$$

$$ab + ad + ac + bc + cd + bd = 1$$

This finds several size 2 tipping sets, which is the optimal. [3]

## 3.2 Empirical Results

In order to test our approximation method, we chose a couple of models. First, we chose the well studied and analyzed airline security investment model from Heal and Kunreuther.[9]

### 3.2.1 Airline Security - Methodology

In this model, detailed in the sections above, we start with the number of players n. There are n players labeled 1 through n, each endowed with two pure strategies denoted by 1 (invest) and 0 ( not invest). A play $s$ is defined by the n-tuple $s = (s_{1,}, s_2, ..., s_n)$ where $s_i \in \{0, 1\}$ denotes the choice of pure strategy by player $1 \leq i \leq n$. For each player, we have the loss of an event, $L_i$, the probability of such an event occurring due to player i's inaction, $p_i$, and the cost of investment $c_i$. These together determine the dependent component of the loss. We then also have the matrix of interdependency, where each entry $q_{ji}$ is the probability of player i suffering a loss due to the inaction of player j. Thus we have a loss function for each non-investing player given by:

$$p_i L_i + (1 - p_i) L_i (1 - \prod_{i \neq j, j \notin S} (1 - q_{ji}))$$

and each investing player by:

$$c_i + L_i (1 - \prod_{i \neq j, j \notin S} (1 - q_{ji}))$$

As previously discussed, the interesting structure stems from the matrix of $q_{ji}$ while the $p_i, L_i, c_i$ are simple constraints for the system. Thus, for our purposes, we set the $p, L, c$ to all be invariant among players, and with bounds giving room to choose a wide range of q.

Table 5: Set Variables

| | |
|---|---|
| p | .1 |
| L | 10000 |
| c | 900 |

We used three different case sizes, 10 players, 15 players, and 20 players. For each of these scenarios, a set of games is randomly generated. We set the values for p's, L's, and c's as denoted in Table 5, then randomly generate q's. The q's completely determine the interdependence dynamics, thus the choice of p, L, and c is only important to give the possibility of extreme equilibria as discussed previously. The q's are chosen from a uniform distribution between 0 and p. This makes sure the q's are less than or equal to the p's, which matches typical scenarios in game examples. A game is a valid, or viable, if it has NE at both all 0 and all 1. The set variables are chosen to allow some range of viable games, so we can have a good amount of valid games from the randomization. After the q's are generated, the games were evaluated and thrown out if extreme NE were not found. A smaller sample size for 20 players was used due to computational limitations. Results are summarized in Tables 6 and 7.

### 3.2.2 Observations

As expected, the approximation runs were not optimal, but did give reasonable approximations for the tipping sets. The sets given were always tipping sets, but smaller sets often existed. One interesting note is that the difference in approximate and exact solutions, in terms of percent error, decreased as the number of players increased. This may be due to the completely random distribution of the q's. This would be roughly equivalent to the system not having many long cascades, which is consistent with random generation.

Table 6: Results Summary (Mean)

| Players | Exact | Approx | Diff | % Diff | Sample Size | Std Dev |
|---|---|---|---|---|---|---|
| 10 | 4.68 | 7.20 | 2.53 | 54% | 826 | 0.189 |
| 15 | 8.89 | 12.08 | 3.19 | 35% | 999 | 0.073 |
| 20 | 13.37 | 17.04 | 3.67 | 27% | 99 | 0.052 |

Table 7: Results Summary (Median)

| Players | Exact | Approx | Diff | % Diff | Sample Size | Std Dev |
|---|---|---|---|---|---|---|
| 10 | 5 | 7 | 2 | 40% | 826 | 0.189 |
| 15 | 9 | 12 | 3 | 33% | 999 | 0.073 |
| 20 | 13 | 17 | 4 | 31% | 99 | 0.052 |

## 3.3 Dynamic Game Implications

There is another class of attempted approximations to finding minimum tipping sets. These involve solving the problem in a continuous space, using dynamic approaches. For some cases this works well, mirroring partial investments and partial results. In other cases, mostly all or nothing cases, or ones with certain key players influencing others, these systems may not work as well. Here the learning method employed by Kearns and Ortiz for approximating tipping sets is used and examined[11].

### 3.3.1 Method Summary

One of the cases examined in [11] was a non-uniform transfer case. While this did not meet the requirements to use the primary method in their work, a second learning / dynamics method as also used for non-uniform transfer examples.

To generate a tipping set, the system is simulated with subsets of players clamped to constant investment. The other players investment is then considered in the continuous space between 0 and 1. Each player is allowed to adjust their amount of investment according to what is best for them. Formally, each player adjusts their level of investment, $x_t$ at time $t$ by:

$x_{t+1} = x_i + (0.05 * (u(0_i s_{-i}) - u(1_i, s_{-i})))$

This has been translated to the $u$ notation for consistency, but was derived from base variables in [11].

This is then iterated on $t$ until a solution is found (or 500 rounds). The players clamped to 1 were chosen in a greedy method. For many of the previously examined cases, a NE is not guaranteed, but we have made sure the cases examined here do have NE when all players are investing.

### 3.3.2 Constraints

The initial parameters $p, L, c$ are set to match the settings in [11]. One of the key indicating factors used was $R_i = \frac{c_i}{L_i}$. In these simulations, given that $c$ and $L$ are constant across players, we can use $R = \frac{c}{L}$. These are:

$p = 0.01$

$L = 10000$

$c = 90$

These allow for the payoff ratio of $R_i = .009$ used in [11] to be preserved. This implies that a low payoff occurrence, such as being impacted by the lack of security investment, is roughly 110 times worse than investing. The $q_{ij}$ are then generated randomly from a normal distribution to fill out the game.

To maintain consistency, the cases examined are still required to have a NE at all players investing, which was not required in [11]. We remove the restraint that there is an NE at all players not investing in order to help accommodate the other restriction set forth in [11]. The players initial state is still non-investment. In the cases used, 50 iterations were sufficient to reach equilibrium.

For the dynamic method, the game was run starting with no players clamped to invest, then adding players in order until the whole system tipped. Typical results are shown in Figure 6. All players not plotted in the graphs in Figure 6 are locked into investing.

The same method was then run clamping the results of the algebraic method to invest, obtaining the results in Figure 7. Again, all players not plotted in the figures are locked into investing.

### 3.3.3 Results and Commentary

The plotting / learning route used in the dynamic method was run first against all players, then against candidate tipping sets from the algebraic method. In the second case, we used only players from the candidate tipping sets as players who can be locked into investment. This validated that the algebraic method did find feasible solutions, and solutions that converge quickly. Both methods performed well, although some better for different types of cases.

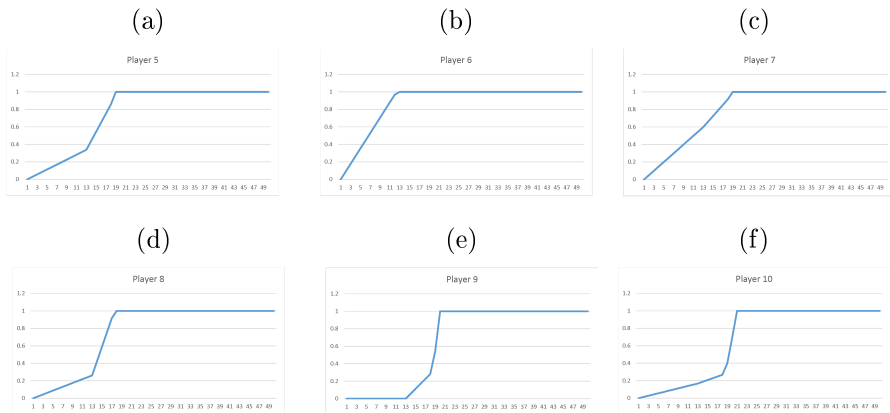
Generally the algebraic method does well at identifying key players to tip the system, but fails to take advantage of cascade effects, and thus overestimates the size of the tipping sets. The dynamic method fairs better with cascades, but does little to identify key players, using a simple greedy approach to adding players to the tipping set. This observation prompts investigation into using a combination of the methods.

The results of the two methods were often similar. This appears to be because of the random nature of our interdependence. As observed above, differently constructed games can have significant impacts on performance. Games with key players can be missed by the dynamic method, and games with long cascades in their optimal solutions often are missed by the algebraic method.

### 3.3.4 Combination of Methods

Given the characteristics above, a combination of these two methods was attempted. To do this, the algebraic method was used to identify key players,

Figure 6: Dynamic Results




then the dynamic method was used. to trim away players not needed to tip the system. This means that fewer combinations of players locked to investment need be considered. It also takes advantage of the algebraic method's ability to find key players as well as the dynamic method's ability to cope with cascades.

A case was chosen that had such cascades, a known weakness of the algebraic method. The tipping set was still required to tip the entire system, but could be chosen only from the output of the algebraic method. The results are in Figure 8. In this case, the algebraic method found a candidate set of eight players, which was reduced to four by the dynamic method.

As previously mentioned, all players without an investment graph are clamped to investing.

Figure 7: Algebraic Results

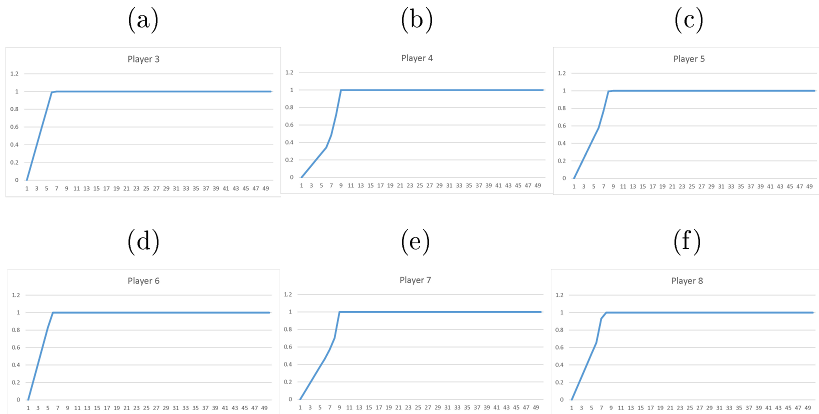
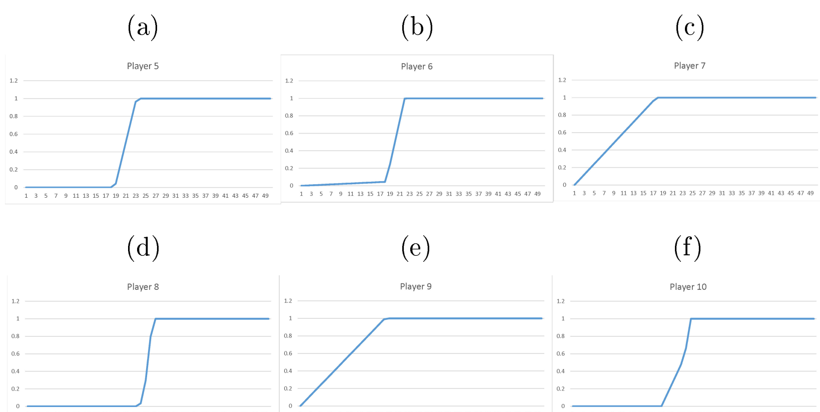



Figure 8: Mix Results



Chapter 4

# 4 Work Summary

We have analyzed the methods for finding minimum tipping sets in SI games, and transformed the problem into a graph theory problem. We then presented a graph-based algorithm to find a minimum tipping set. We then examined the complexity of the general case of Airline Security games. This motivates an approximation method, backed by hypercube sampling and an algebraic method to approximate the minimum tipping set for general Airline Security games. This method is then placed in context with dynamic methods for identifying tipping sets in airline security games.

Chapter 5

# 5 Open Questions

There are several possible improvements on the performance of the above methods. Methods of trimming the input space could show great improvements, as restricting the number of players has large benefits. It is of interest if the players can be subdivided before running the algorithm, to tip the system in groups. There are also tunable parameters that could be investigated. Allowing short cascades, for example, could improve the accuracy of the approximated sets. There is also the fact that the algebra used in the approximation can use compacting methods to collapse some of the combinations of candidates.

This could also impact more loosely defined games. There are many types of games that lack some of the structure of the airline security games, but similar methods may be able to be employed. It appears these methods would work well in pure dynamic games, without perfect foresight. There is also promise from incorporating incremental incentives, instead of all at once. The work examining methods used in [11] show promise in examining how this complex approximation could be incorporated into existing methods of approximation. It may also become easier to solve the problem with a relaxed concept of tipping, for example, if there is interest in hitting most of a group, not the whole group.

# References

[1] Alekhnovich, M., Buss, S., Moran, S., Pitassi, T., (2001) "Minimum Propositional Proof Length is NP-Hard to Linearly Approximate." **Journal of Symbolic Logic 66,** pp 171-191

[2] Bellhouse, D (2007) "The Problem of Waldegrave", **Journal Electronique d'Histore des Probabilites et de la Statistiques** pp 1-12

[3] Cremeans, B., Lakshmivarahan, S., Dhall, S. (2012) "An approximation algorithm for computing a tipping set in super modular games for interdependent security, **Procedia Computer Science**, Vol 12 pp 404-412

[4] Daskalakis, C (2004) "The Complexity of Nash Equilibria",Diploma (National Technical University of Athens) 2008

[5] Dixit, A. K. (2002) "Clubs with Entrapment", **American Economic Review**, Vol 93, No. 5, pp 18124-1829

[6] Dinur, I., Safra, S., (2004) "On the hardness of approximating label-cover", **Journal Information Processing Letters** Vol 89, Issue 5, pp 247-254

[7] Dhall, S. K. and S. Lakshmivarahan, and P Verma (2009), "On the Number and the Distribution of the Nash Equilibria in SuperModular Games and their Impact on the Tipping Set", **Proceedings of the International Conference on Game Theory and Networks**, Game Nets, 2009, Istanbul, pp 691-696

[8] Gladwell, M (2000) **Tipping Point**, Little Brown and Co. 301 pages

[9] Heal, G.M. and Kunreuther, H. (2005) "IDS Models of Airline Security", **Journal of Conflict Resolution**, Vol 41, pp 201-217

[10] Heal, G.M., and H. Kunreuther (2006), "Supermodularity and Tipping", **National Bureau of Economic Research**, 1050 Mass Ave, Cambridge MA.

[11] Kearns, M., and Ortiz, L.E. (2004), “Algorithms for Interdependent Security Games”, **In Advances in Neural Information Processing Systems,**

[12] D. Kempe, J. Kleinberg, and E. Tardos, “Maximizing the spread of influence through a social network,” in KDD ’03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining. New York, NY, USA: ACM, 2003, pp. 137–146.

[13] Kunreuther, H. and Heal, G.M. (2003) “Interdependent Security”, Journal of Risk and Uncertainty, **Special Issue on Terrorist Risks**, Vol 26, No 2/3, pp 231-249.

[14] Lakshmivarahan, S., and S. K. Dhall (1990) **Analysis and design of Parallel Algorithms**, McGraw Hill, New York Chapter 2.

[15] Nash, J. “Noncooperative games”, (1951), **Ann. Math**. 54, 289-295.

[16] Owen, G (1995) **Game Theory**, Academic Press (Third Edition), 447 pages.

[17] Schelling, T. (1978) **Micromotives and Macrobehavior,** Norton, New York. 272 pages

[18] Straffin, P. (1993) Game Theory and Strategy, The Mathematical Association of America, Washington, DC 247 pages

[19] Topkis, D. (1978) "Minimizing a Submodular Modular Function on a Lattice", **Operations Research,** Vol 26, pp 305-321.

[20] Topkis, D. (1979) "Equilibrium Points in Nonzero-Sum n-Person Submodular Games", **SIAM Journal of Control and Optimization**, Vol 17, pp 773-787.

[21] Topkis, D (1998) **Supermodularity and Complementarity**, Princeton University Press, Princeton, NJ. 272 pages