EXTENSIONS TO THE CHICKEN GAME IN THE CONTEXT OF PUBLIC GOODS

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Bryan H. Chong

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

December 2020

Purdue University

West Lafayette, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF DISSERTATION APPROVAL

Dr. Mario Ventresca, Chair

School of Industrial Engineering

Dr. Simina Brânzei

Department of Computer Science

Dr. Seokcheon Lee

School of Industrial Engineering

Dr. Shreyas Sundaram

School of Electrical and Computer Engineering

Approved by:

Dr. Abhijit Deshmukh

Head of the School Graduate Program

DEDICATION & ACKNOWLEDGMENTS

I dedicate this dissertation to my parents for their unwavering love and support, and to my brother for being my brother. Of course, to Erica who is in my head and heart every moment of the day. And to Soda, my beloved cat of seven years and four one-bedroom apartments who by now certainly solves $\mathrm{argmax}_{x\in\mathrm{world}} \int_{\mathrm{life}} \frac{1}{d_t(\mathrm{me},x)} dt$.

I have received a tremendous amount of guidance and support throughout the course of my PhD, and have much to be thankful for. First and foremost my utmost gratitude goes out to my advisor, Prof. Mario Ventresca, for his patient and kind mentorship, for his encouragement to pursue any and all research opportunities, and for taking a chance on me in the first place. It has been a privilege to work with him both as a researcher and in the classroom, and I am proud to be his student and friend.

I also thank my dissertation committee members for their interest in and comments on the early versions of the Edge-Building Game. Your interest roused mine, and is what spurred me to develop it into this thesis.

My labmates in Ventresca lab have been an invaluable part of my time at Purdue. Thank you all for your insightful comments and suggestions over the years. In particular, to Viplove, Dali, and Dawei, who have been with me since day one, it has been a pleasure going on this academic journey together. Thank you also to Mintao for watching Soda all the time.

Finally, thank you to all my friends local and abroad who have brought me such happiness. Zhihao, Darren, and Maurice, for years of eclectic conversations. My dearest friend Li May and the gang back in Malaysia, for the 16 years of friendship that I could always count on. The men of Undue Ultimate, teammates and friends without whom I would have graduated a year sooner. Ben and Dwayne, for all the sports and good times back home, keeping me both fit and injured.

PREFACE

My research philosophy stems from an ardent belief in utility theory with the appropriate caveats on bounded rationality and discounting. Roughly speaking, every decision a person consciously makes is the one that maximizes his or her expected total utility over all future time, where future utility is discounted, and the expectation is over the person's beliefs about possible outcomes[1]. The argument is simple: if there had been a better-perceived decision, why was that one not selected instead? One might argue that the above definition is so narrow as to be meaninglessly tautological, and one might be right if the goal is to use this framework to make perfect decisions all the time. If, instead, we judiciously select the scope at which to use the lens of utility theory, I believe (and this is certainly not a novel idea) that we can take powerful steps toward better understanding and predicting human behavior.

Noting that utility theory broadly includes optimization over human behavior, almost all the research I have enjoyed working on has some aspect of utility theory to it. From maximizing likelihood of winning a multi-stage piano competition, to having nodes on a network make connection decisions based on its potential improvements in some local (bounded rationality!) subgraph, to having network attackers spend a budget to extract utility from other. The culmination of my utility-driven research is this dissertation, where I explore how two parties selfishly vie for outcomes with higher utility by risking getting nothing at all.

[1] These beliefs are almost necessarily wrong in some way, not only with regard to the likelihoods of outcomes, but also to the utility gained in each outcome

TABLE OF CONTENTS

Page

LIST OF TABLES

Table | Page

LIST OF FIGURES

Figure Page

SYMBOLS

u	Utility
c	Edge cost
f	A probability distribution function
F	A cumulative distribution function
g	A network or graph
$N_i(g)$	The set of of neighbors of node i on graph g
$t_{ij}(g)$	Length of the shortest path(s) between nodes i and j on graph g
$m(i, g)$	A metric on graph g from the perspective of node i

ABBREVIATIONS

EBG	Edge-building game
PNE	Pure Nash equlibrium
MNE	Mixed Nash equilibrium
CE	Correlated equilibrium
PDF	Probability distribution function
CDF	Cumulative distribution function
CEBG	Continuous edge building game
NEBG	Edge building game on networks
CC	Connected component
PAP	Probe-and-attack problem

ABSTRACT

Chong, Bryan H. Ph.D., Purdue University, December 2020. Extensions to the Chicken Game in the Context of Public Goods. Major Professor: Mario Ventresca.

We study the Edge-Building Game (EBG), a parameterized formulation of the Chicken Game in the context of a public goods game. In the EBG, each of two players can either propose or abstain from building a shared resource. At least one person must propose in order for the resource to be built, but neither wants to contribute the associated edge cost. They must then trade off the possible benefit of gaining the resource for free against the chance that neither get anything at all. Contexts include trade deals between countries and interest signalling in interpersonal relationships.

We first study two extensions to the game: the EBG on a continuous strategy set where players can propose from a continuum of strength (as opposed to just proposing or abstaining), and the EBG with multiple players on the same resource. Even with continuous strategy sets, the EBG is a discontinuous game, and we find discontinuous Nash equilibrium distributions for different game scenarios. These discontinuous equilibria take the following general form: Propose with a fixed probability, and with the remaining probability draw from some parameterized probability distribution. In the multiplayer game, we find that proposal probability drops rapidly with an increase in the number of players (at low edge-cost), and we find that the expected utility when playing a mixed-strategy Nash equilibrium is worse than the two-player case, but the reverse holding true for a particular correlated equilibrium.

Next, we play the game on a graph/network, under both simultaneous- and sequential-play rules. Here, we allow nodes to have different utility functions that depend on graph topology. We explore the general intractability of computing Nash and correlated equilibria of the simultaneous EBG on networks, and present some

special cases of the problem where Nash and correlated equilibria can be efficiently found. Correlated equilibria can even be optimized over when the simultaneous game on networks is played on a tree with bounded degree, and when node utilities only depend on a local subgraph with a particular radius.

We then run simulations of the sequential game, where pairs of players on the network take turns playing, and study how the parameterization of node utility functions affects output metrics. We also introduce a novel submodular network measure that we use in the sequential game simulations.

Relatedly, we introduce a combinatorial problem on attacking defended networks, the Probe-and-Attack Problem (PAP). In the PAP, nodes confer protection from attacks to all its neighbors. An attacker must decide how to allocate a budget between probing nodes to disable this protection, and attacking unprotected nodes to extract direct utility. We show it is NP-hard, produce upper- and lower-bounds that are found by solving a (much more tractable) knapsack problem, and present efficient heuristics or the problem, testing the heuristics on simulated networks.

1. INTRODUCTION

Consider broadly the following scenarios:

- **International Trade**: Suppose two countries wish to form a trade treaty with one another. If one country were to seem too eager, the other might take a tougher stance during negotiations, resulting in a worse deal for the eager country. The countries must simultaneously signal the amount of interest to each other to set the bar for negotiations.

- **Social Relations**: A similar situation can unfold at an interpersonal level in any relationship involving power dynamics, be it dating or friendships. Unfortunately, it is often the person who signals greater interest that has to contribute great effort toward the relationship.

- **Resource-Building**: In pledging to fund a common resource - e.g. neighbors sharing a fence or teammates working on a group project - most people would rather freeload off of others' capital. If you know the resource will be made with or without your contribution, the selfish but profitable move is to not contribute.

In each of these scenarios, each party wishes for some mutually beneficial relationship with the others, but wishes to pay no or minimal cost. We model this general setting as a public goods game, where we are interested in how players behave as they try and freeload off of each other, and the results of such behavior on societal welfare. We call our game the Edge-Building Game (EBG).

1.1 Background

1.1.1 Game Theory

We mainly study players' behavior under the lens of game-theoretic equilibria. Suppose a player has no incentive to deviate from his or her current behavior, given the assumption that no other player deviates. If this holds true for all players, then we are said to be in an equilibrium. We study three types of equilibria, with formal definitions appearing in later chapters as required:

1. Given a joint strategy (one strategy for each player), if no player can improve utility by switching to another strategy, the joint strategy is a pure-strategy Nash equilibrium (PNE).

2. Suppose that players are allowed to define a probability distribution over their strategies, such that they play each strategy with the corresponding probability. If, in expectation, no player can improve his or her *expected* utility (taken over the joint probability distributions of all other players), then set of probability distributions is a mixed-strategy Nash equilibrium (MNE).

3. Suppose a trusted third party draws from a distribution over joint strategies, where the exact distribution is known to all players. Then, when presented with each of their own strategies from the drawn joint strategy (without explicitly being told what joint strategy was drawn, i.e., without knowing what the other players were presented), if no player can improve upon their *conditional expected* utility, then the distribution on joint strategies is a correlated equilibrium (CE).

Why do we care about Nash equilibria? They are often the main focus of study (Kleinberg et al., 2011) for their mathematical properties, and for a vague notion that they are representative of real life decision-making. While it is sometimes true that players act according to the MNEs (Azar and Bar-Eli, 2011; Kümmerli et al., 2007), this not always the case for a variety of reasons including irrationality of players and inability of players to compute or communicate the equilibria (Carrillo and Palfrey, 2009; Daskalakis et al., 2010; Kleinberg et al., 2011). Camerer and Fehr (2004) in particular presents a compilation of differences in theoretical and experimental behavior for a number of canonical games. For instance, participants in the Prisoner's dilemma tend to cooperate about half the time, when it is optimal to defect all the time (Dawes, 1980). However, the limitations in laboratory-run experiments should be noted, particularly that theoretical behavior is more likely to occur if the the experimental rewards (and losses) are much larger than is generally found in a university-run research experiment (Wooldridge, 2012).

On the other hand, it is argued that game-theoretic equilibria can be reached through communication between players and evolution of play over multiple rounds, and that, if nothing else, the models lead to at least a high-level understanding that may be refined by deeper experimental study (Camerer, 1991; Samuelson, 2016). Furthermore, the seemingly unrealistic randomization for MNEs do sometimes play out in real life (Tambe, 2011). Thus, given the simplicity of our game, it might be reasonable to assume that the MNEs are in fact close to what a real person might play (Wooldridge, 2012), and so we claim fidelity to real-world behavior as a reason for study alongside mathematical interest. CEs, on the other hand, can arguably always be imposed upon players by a system designer/trusted third party, and thus can lead to realistic outcomes in real life.

We study continuous and multiplayer versions of the EBG (Chapter 3) as they have mathematically interesting equilibria while still being real-world applicable. We use a variety of analytical and numerical methods to compute these equilibria. For the continuous games, we and find MNEs with discontinuous CDFs, something rarely encountered in the literature. For the multiplayer game, we counterintuitively find that the expected utility under the MNE decreases as more players enter the game, as the loss from an increase in likelihood of no edge forming is greater than the gain from increased freeloading and shared costs.

1.1.2 Network Science

Network science is the study of complex networks, which is in turn, roughly speaking a catch-all term for representations of or simulated attempts at real-world networks. These are networks that do not have a tidy structure as opposed to say, Erdős–Rényi and Barabási–Albert random graphs (Erdős and Rényi, 1960; Barabási and Albert, 1999), upon which clean mathematical results can be derived. As such, when analyzing complex networks we generally have to rely on statistical measures of or metrics on the network such as betweenness and clustering coefficient.

We study the EBG on networks (Chapter 5) where nodes play games on the edges of a network. In some special cases of game setup (for instance, when played on trees, or with particularly graph-theoretic/math-friendly utility functions) we are able to characterize some Nash and correlated equilibria.

We also explore complex network generation, the problem of synthesizing networks comparable in some manner to those in real life. The problem is a complicated one, as it is difficult to even define a metric with which to judge generated networks. For instance, we must balance fidelity with the original real network against a desirable variation in the population of generated networks (see Arora et al. (2020) for a discussion on this particular spectrum). In this dissertation, we generate networks by simulating actual edge formation when nodes play the EBD with each other, using the MNEs as proposal/edge-formation probabilities. We take a mechanism design approach and vary certain continuous parameters of the nodes' utility functions, allow the networks to grow to a fixed size, then study the metrics on these returned networks. For instance, we find that betweenness of output networks is affected by most parameterizations we tried. That is, we study complex network generation from a bottom-up approach: instead of working toward a particular target real-world network, we study the variation in generated networks as a function of model parameters.

We also introduce a graph metric called the multipath connections utility that measures the utility a node gets from the network (Chapter 4). This measure takes into account all paths between a node and all other nodes, as we believe that having more paths (instead of just looking at the shortest path) to another node is a sign of closeness between nodes. We show submodularity properties of the metric, and incorporate the metric in the EBG network generation experiments.

1.1.3 Probe-and-Attack Problem

Largely from the work up to this point, but still tied to utility-driven games on networks and network formation, we introduce a combinatorial problem (the Probe-and-Attack Problem, PAP) involving attacking protected nodes on a graph. We show NP-hardness of the problem, model it as an integer program, and identify some network properties that may lead to an easier solve. We also show some upper and lower bounds to the problem found through solving the (relatively tractable) Knapsack Problem. We then arrive at some efficient heuristics and test them against the upper and lower bounds on some simulated instances of the game.

1.2 Dissertation Structure

In the short Chapter 2 we review the Chicken Game and the parameterization that results in the Edge-Building Game, along with its various equilibria. In Chapter 3 we cover continuous and multiplayer extensions to the EBG. In Chapter 4 we introduce the multipath connection model, a submodular network metric that takes into account all paths between two nodes. In Chapter 5 we study the EBG on networks, wherein we explore equilibria of the game on networks, and also the generation of networks based on equilibria. In Chapter 6 we cover the separate Probe-and-Attack Problem, a combinatorial problem about attacking defended nodes.

2. THE EDGE-BUILDING GAME

2.1 Introductory Remarks

In this short chapter, we first review the well-studied Chicken Game before we modify it to form the main building block of the dissertation, the two-player Edge-Building Game (EBG), a public goods game where players desire a resource but do not want to pay for it, each trying to get the other to 'chicken' out of freeloading.

2.2 Chicken Game

In the Chicken Game (Rapoport and Chammah, 1966), two drivers on the same street speed toward each other in their cars, each with the option to either swerve ('chicken') out of the way, or remain on their initial course. Remaining on the course with the other driver swerving yields positive utility, while the opposite scenario yields negative utility. Neither driver gets anything if both swerve safely out of the way, but staying their courses leads to a terrible crash where both pay a large cost. See Figure 2.1 for an example of a possible payoff matrix.

When studied under the lens of evolutionary game theory, the Chicken Game is often referred to as the Hawk-Dove Game (Killingback and Doebeli, 1996; Sigmund

		Driver 2	
		Swerve	Remain
Driver 1	Swerve	$(0, 0)$	$(-10, +10)$
	Remain	$(10, -10)$	$(-100, -100)$

Fig. 2.1.: Example payoff matrix for the Chicken Game, where (u_1, u_2) found at the intersection of row r and column c indicates that player i receives u_i utility when player 1 plays strategy r and player 2 plays strategy c.

and Nowak, 1999) or the snowdrift game (Sui et al., 2015). In this context, one is interested in the players' behavior over repeated games and the dynamics of each faction's shifting strategies (Smith and Price, 1973). In all scenarios in this thesis, we assume players are agnostic to all players' previous choices of strategies for ease of analysis and modeling, and so we do not touch further upon evolutionary game theory. The Chicken Game is also studied as brinkmanship (terrorism/nuclear weapons), generally focusing on credible threats/pre-committing to strategies (Melese, 2009; Nalebuff, 1986).

2.3 Edge-Building Game

The vanilla EBG is a discrete two-player game where each player can decide whether to **p**ropose (play strategy P) or **a**bstain from proposing (play strategy A) to build an edge with the other player. If at least one player plays strategy P, the edge is built, and each player derives one unit of utility (we generalize utilities in the next subsection). The cost to build the edge is $c \in (0, 1)$, such that a player always receives positive probability $(1 - c)$, and it is shared between all players playing strategy P. The payoff matrix is shown in Figure 2.2.

Two pure-strategy Nash equilibria (PNE) (Nash, 1951) can be observed, at (P, A) and (A, P): the proposer does not want to deviate as the edge will be lost (noting $c < 1$), and the abstainer does not want to pay any extra (similarly, $c > 0$) than the nothing she is currently contributing to the edge-building cost.

Moreover, the symmetric mixed-strategy Nash equilibria (MNE) can be identified, where each player proposes with probability

$$p^* = \frac{1 - c}{1 - \frac{c}{2}} \tag{2.1}$$

		Player 2	
		P	A
Player 1	P	$(1-\frac{c}{2}, 1-\frac{c}{2})$	$(1-c, 1)$
	A	$(1, 1-c)$	$(0, 0)$

Fig. 2.2.: Payoff matrix for the vanilla Edge-Building Game, with $0 < c < 1$

		Player 2	
		P	A
Player 1	P	$(u_1(\cdot) - c_1(\cdot, P), u_2(\cdot) - c_2(\cdot, P))$	$(u_1(\cdot) - c_1(\cdot, A), u_2(\cdot))$
	A	$(u_1(\cdot), u_2(\cdot) - c_2(\cdot, A))$	$(0, 0)$

Fig. 2.3.: Payoff matrix for the two-player Edge-Building Game with general utilities and costs.

2.3.1 Generalization

We can generalize utility functions to allow for edge-utility to be other than just 1, and cost-sharing schemes to be other than just an even split of the edge cost. In order to maintain contextual relevance, we will maintain that the total utility gained by a player must always be the difference between the utility from the edge (henceforth 'edge-utility', to distinguish the more general 'utility' associated with a joint strategy and player) and the cost the player ends up paying for said edge. The edge-utility does not depend on the specific joint strategy played (so long as the edge is built), as we assume a player only cares about the utility of the edge and the cost to build it and nothing extraneous such as, for instance, second-order utilities on fairness between players' utilities. Similarly, a cost is only incurred if the edge is built, and does depend on the other player's strategy. The general form is defined as we use different edge-utilities when playing the game on networks in Chapter 5.

2.4 Correlated Equilibria

Next, we look at correlated equilibria (CE) in the vanilla EBG. First, we show a quick result on undesirability (with respect to expected total utility) of strategy (A, A) in any correlated equilibrium.

Proposition 2.4.1 *Any correlated equilibrium that plays strategy* (A, A) *with positive probability is sub-optimal with respect to expected total utility.*

Proof Given any CE that plays (A, A) with positive probability, we construct a new CE from it with strictly higher expected total utility, and so no (A, A) is never in an optimal CE. Let $\pi_{s_1 s_2}$ be the probability of playing strategy (s_1, s_2). Then, the two constraints that define a distribution as a correlated equilibrium are:

$$\pi_{PP}(1 - \frac{c}{2}) + \pi_{PA}(1 - c) \geq \pi_{PP}(1) \tag{2.2}$$

$$\pi_{AP}(1) + \pi_{AA}(0) \geq \pi_{AP}(1 - \frac{c}{2}) + \pi_{AA}(1 - c) \tag{2.3}$$

Reducing the positive-valued π_{AA} to zero does not affect the first constraint, while making the second constraint easier to satisfy. We then scale the other probabilities by $\frac{1}{1-\pi_{AA}}$ to maintain a valid distribution, noting that this scaling does not affect satisfaction of either of the (linear) constraints.

Since all utilities other than those associated with π_{AA} are strictly positive (for $c < 1$), expected utility must increase. ■

Note that Proposition 2.4.1 is trivial when there exists any equilibrium that attains optimal total expected utility (over the set of all joint strategies), which is true in this case, but we include it for completeness. Next, assuming symmetry between (P, A) and (A, P), we have:

Proposition 2.4.2 *Playing strategies* $(P, P), (P, A), (A, P)$ *with probabilities* $1-2x, x, x$ *is a correlated equilibrium for* $x \geq \frac{c}{2}$.

Proof A player shown A has no reason to deviate since the other node must then have P. A player shown P will not deviate if there is no expected gain in utility:

$$\left(\frac{1-2x}{1-x}\right)\left(1-\frac{c}{2}\right)+\left(\frac{x}{1-x}\right)(1-c) \geq \frac{1-2x}{1-x} \tag{2.4}$$

which simplifies to $x \geq \frac{c}{2}$. ■

Moreover, the expected utility under such a correlated equilibrium is $(1-2x)(1-c/2)+x(1-c)+x = 1-c/2$, which is $\geq \frac{1-c}{1-\frac{c}{2}} = p^*$, the expected utility under the MNE.

Lastly, we note that any distribution where (P,P) and (A,A) are both played with probability zero (i.e., (P,A) and (A,P) with probabilities $x, 1-x$ for $0 \leq x \leq 1$) is a correlated equilibrium, corresponding to being shown either of the two PNEs.

3. CONTINUOUS AND MULTIPLAYER EXTENSIONS OF THE EDGE-BUILDING GAME

Next, we study a continuous extension of the Edge-Building Game (CEBG). Consider a good or resource that needs some minimum amount of effort or investment to be developed, for instance a building that is only built if sufficient money is pledged. Then, instead of having each player only be able to either propose or abstain, we allow them to propose with an effort $r_i \in [0, 1], i = 1, 2$. The good/edge is built if and only if the sum of the two players' efforts is greater or equal to 1 (without loss of generality), conferring edge utility and incurring cost appropriately.

3.1 Existence of Nash Equilibria

Through an extension to the Kakutani Fixed-Point Theorem (Kakutani et al., 1941), it is known that even games with uncountably infinite strategies have Nash equilibria if the utility functions are continuous and the strategy sets are compact (Glicksberg, 1952).

The CEBG is discontinuous only along the line $r_1 + r_2 = 1$, as utility jumps when the total effort goes from < 1 to 1. Games with discontinuous utility functions also admit Nash equilibria, so long as the discontinuities occur on a set of dimension strictly less than the dimension of the joint strategy space, and the utility functions obey certain semi-continuity properties (Dasgupta and Maskin, 1986). Unfortunately, while our utility functions fulfil the former condition, they do not the latter, even if we weaken said properties as is allowed for symmetric games (Yang, 1994): in particular, the utility of player 1, say $u_1(r_1, r_2)$, fails to be left lower semi-continuous in r_1 at $(1, 0)$ (see Definition 6 of Dasgupta and Maskin, 1986).

Fortunately, the assumptions we made in Section 2.3.1, about our utility functions always being an edge-utility less an edge-cost, leads us to a straightforward condition that ensures existence of Nash equilibria. We simply require that (1) for any joint strategy, the utility gained from an edge is at least as great as the maximum possible cost paid to build the edge, $\min_{r_i} u_i(r_i) \geq \max_{r_i,r_j} c_i(r_i, r_j)$, and (2) that the cost a player pays is increasing in his or her effort given the other player's effort, $c_i(r_i, r_j)$ is increasing in r_i. Thus, any joint strategy with $r_1 + r_2 = 1$ is a PNE. That is, no player wants to contribute less effort as doing so would disconnect the edge, yielding less or equal utility by (1), and by (2) neither wants to contribute more as doing so would incur greater cost by while maintaining the same edge utility.

3.2 Nash Equilibria Under Particular Cost Functions

In a continuous game, players can choose from uncountably infinite strategies, and so we must modify the definition of a mixed strategy from being a distribution over discrete values (a probability mass function) to a distribution over a range of values (a probability distribution function (PDF)). The notion of MNEs remains the same, where we require the other player to derive the same expected value from any of their strategies. When the game is simple, as is in some of our cases, we are able to derive the MNEs using algebra or simple differential equations. Strategies for more complicated continuous games involve the use of discretizing the continuous space (Kroer and Sandholm, 2015), or even machine learning (Kamra et al., 2019; Perkins et al., 2015).

Formally, let $u_i(r_i, r_j)$ be the utility that player i receives under joint strategy (r_i, r_j). Next, let $v_i(r_i, F_j(\cdot)) = \mathbb{E}_{R_j \sim f_j}[u_i(r_i, R_j)]$, where f_j is the PDF corresponding to the CDF F_j. That is, v_i is the expected utility of player i when she plays r_i and player j plays a mixed strategy with CDF F_j. Then, to find mixed strategies corresponding to an MNE, we need constants k_i such that:

$$v_1(r_1, F_2(\cdot)) = k_1, \forall r_1 \in [0,1] \tag{3.1}$$

$$v_2(r_2, F_1(\cdot)) = k_2, \forall r_2 \in [0,1] \tag{3.2}$$

As in the discrete case, player 1's MNE distribution $F_1(r_1)$ depends only on the utility function of player 1, and not on his own MNE distribution nor on the utility function of player 2. We thus drop the subscripts for ease of notation:

$$v(r, F(\cdot)) = k, \forall r \in [0,1] \tag{3.3}$$

Next, we explore Nash equilibria of the CEBG under three different cost scenarios.

3.2.1 Committed Effort Scenario

We first study the cost scenario without the Section 2.3.1 assumptions about incurring cost only when the edge is built. In this scenario, the player pays the cost regardless of whether the edge is built or not. Contextually, one can think of this as an *actual* expended effort that is wasted if the other party does not contribute the remaining cost.

Let player i's cost be an increasing function $g_i(r_i)$ of her effort, multiplied by some constant $c \in (0,1)$, and let her edge utility be $\tilde{u}_i$ Then, her utility is

$$u_i(r_i, r_j) = \tilde{u}_i \mathbb{1}_{\{r_i + r_j \geq 1\}} - c g_i(r_i) \tag{3.4}$$

where $\mathbb{1}_{\{\cdot\}}$ is the indicator function, evaluating to 1 if the argument is true and 0 otherwise. We call this the Committed Effort CEBG and drop subscripts to show:

Theorem 3.2.1 *Let strategy π be to propose with maximum effort ($r = 1$) with probability $1 - \frac{c}{\tilde{u}}$, and to propose using CDF $F(r) = 1 - g(1 - r)$ with probability $\frac{c}{\tilde{u}}$. Then, π is an MNE to the symmetric Committed Effort CEBG if and only if g is an increasing function on $[0, 1]$, with $g(0) = 0$ and $g(1) = 1$.*

Proof We show that the asserted equilibrium satisfies the MNE conditions of equal expected utility. Let $\hat{F}$ be the distribution corresponding to strategy π (including the positive probability of proposing with full effort), noting the distinction from distribution F that does not include the positive maximum-effort probability. Recall that for $\hat{F}$ to be an MNE, every r must be a best response to $\hat{F}$, resulting in Equation 3.3: $v(r, \hat{F}(\cdot)) = k, \forall r \in [0, 1]$.

$$
\begin{aligned}
v(r, \hat{F}(\cdot)) &= \mathbb{E}_{\hat{R}\sim\hat{f}}[\tilde{u}\mathbb{1}_{\{r+\hat{R}\geq 1\}} - cg(r)] && (3.5)\\
&= \tilde{u}P(r + \hat{R} \geq 1) - cg(r) && (3.6)\\
&= \tilde{u}(P(\hat{R} = 1) + P(r + \hat{R} \geq 1)) - cg(r) && (3.7)\\
&= \tilde{u}\left((1 - \frac{c}{\tilde{u}}) + \frac{c}{\tilde{u}}(1 - F(1 - r))\right) - cg(r) && (3.8)\\
&= \tilde{u}\left(1 - \frac{c}{\tilde{u}}(1 - g(r))\right) - cg(r) && (3.9)\\
&= \tilde{u} - c = k && (3.10)
\end{aligned}
$$

3.6 and 3.7 are by the definitions of v and expectation of an indicator function respectively. For 3.8, the first two terms are from substituting in the offered solution: the first term is the $1 - \frac{c}{\tilde{u}}$ chance of proposing with maximal effort (resulting in the edge being formed regardless of r), and the second term is the $\frac{c}{\tilde{u}}$ chance of proposing with a CDF, applying the definition of a CDF on the probability term in the previous line. 3.10 is from explicitly substituting in the asserted CDF of the solution, $F(r) = 1 - g(1 - r)$.

The corresponding PDF is then $f(r) = g'(1-r)$, and so $F(r)$ is a valid CDF if and only if g is increasing on $[0,1]$, $g(0) = 0$, and $g(1) = 1$[1] as required.

■

Furthermore, the stated MNE is, roughly speaking, the unique MNE where each strategy is played with positive probability (that is, where the PDF takes positive value on the whole of $[0,1]$)

It is interesting to note that the conditions on g that give us the required MNE ensure that g is also a valid CDF. Furthermore, the transform $F(r) = 1 - g(1-r)$ is equivalent to rotating the plot of g 180 degrees around the 'center of the box' at $(0.5, 0.5)$.

Next, suppose we restrict the upper limit to a player's proposal effort, say to some value $1-\epsilon$ for some $0 < \epsilon < 0.5$. Then, we have the following MNE:

Corollary 3.2.1 *When the Committed Effort CEBG is restricted such that the upper limit to a player's proposal effort is* $1-\epsilon$ *for some* $\epsilon \in [0, .5]$*, an MNE is to propose with effort* $1-\epsilon$ *with probability* $1-\frac{c}{\tilde{u}}(g(1-\epsilon)-g(\epsilon))$*, and with probability* $\frac{c}{\tilde{u}}(g(1-\epsilon)-g(\epsilon))$ *propose with effort drawn from the CDF* $F(r) = \frac{g(1-\epsilon)-g(1-r)}{g(1-\epsilon)-g(\epsilon)}, r \in [\epsilon, 1-\epsilon]$.

Proof Any proposal value in $[0, \epsilon)$ will never be played, as such a strategy will never yield positive utility, and the proposal efforts are then automatically restricted restricted to $[\epsilon, 1-\epsilon]$. The remainder of the corollary can be verified as in Theorem 3.2.1.

■

The value of the game is $\tilde{u} - cg(1-\epsilon)$, an improvement over the full-range game for any $\epsilon > 0$. At $\epsilon = 0$, we have exactly the scenario and functions in Theorem 3.2.1. Similarly, at $\epsilon = 0.5$, the value of the game is $\tilde{u} - cg(0.5)$. In the corresponding 'game'

[1]Actually, omitting the $g(0) = 0$ and $g(1) = 1$ conditions maintain the validity of F if an appropriate constant term is also added to F, but we then potentially end up with a game with negative value. Forcing these conditions is exactly analogous to scaling the game to have at least 0 (and at most 1) payoff.

(with no actual choices) each player must propose with effort 0.5 (yielding $-cg(0.5)$ utility) and the edge is always built (yielding $+\tilde{u}$ utility). Finally, we note that when we vary ϵ, the CDF $F\hat{(}r)$ is simply scaled and translated (see Figure 3.1).

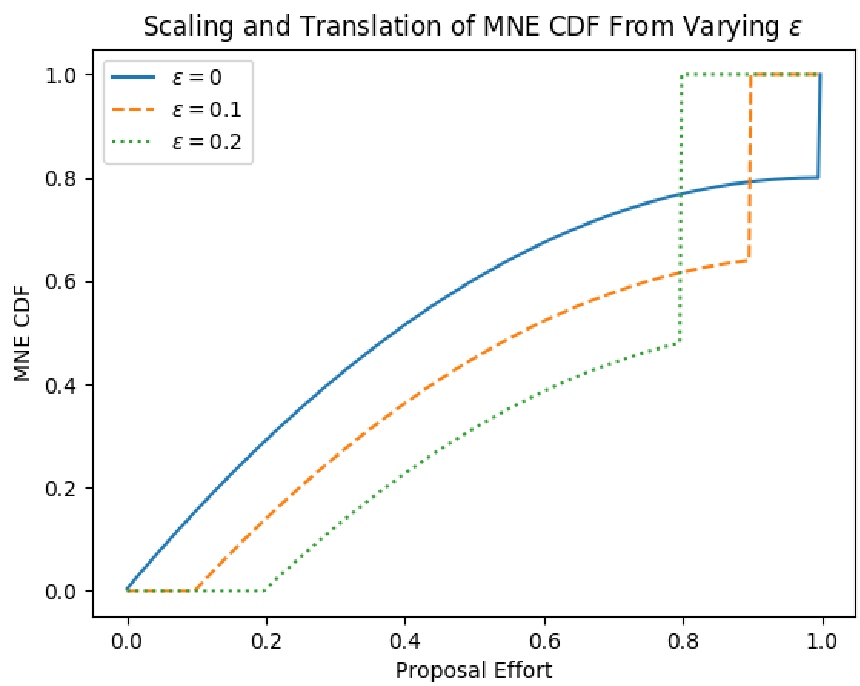


Fig. 3.1.: Varying the upper (or lower) proposal effort limit only translates and scales the mixed Nash equilibrium CDF for the Committed Effort CEBG. The probability of a maximal-effort $(1-\epsilon)$ proposal increases along with the value of the game. $g(r) = r^2$ in this example.

3.2.2 Uncommitted Effort Scenario

In this cost scenario, we restore the assumption that the cost is only paid if the edge is built, but otherwise maintain the structure of the cost and edge utility. Player i's utility is then

$$u_i(r_i, r_j) = (\tilde{u}_i - cg_i(r_i))\, \mathbb{1}_{\{r_i+r_j\geq 1\}} \tag{3.11}$$

We call this the Uncommitted Effort CEBG and show a similar MNE result:

Theorem 3.2.2 *Let strategy π be to propose with maximum effort with probability $1 - \frac{c}{\tilde{u}}$, and to propose using CDF $F(r) = \frac{1-g(1-r)}{1-\frac{c}{\tilde{u}}g(1-r)}$ with probability $\frac{c}{\tilde{u}}$. Then, π is an MNE to the symmetric Uncommitted Effort CEBG if and only if g is an increasing function on $[0,1]$, with $g(0) = 0$ and $g(1) = 1$.*

Proof We similarly require a distribution $\hat{F}$ that fulfils $v(r, \hat{F}(\cdot)) = k, \forall r \in [0,1]$. Instead of substituting the asserted $\hat{F}$ and showing that the equation is satisfied as with the proof of Theorem 3.2.1, we provide a more constructive method for the sake of instruction.

First, on the $1 - \frac{c}{\tilde{u}}$ and $\frac{c}{\tilde{u}}$ terms. Player 1 proposing with full effort and with zero effort should yield the same expected utility, by definition of a MNE. Player 2's strategy has no effect in the former case, and the utility is $\tilde{u} - cg(1) = \tilde{u} - c > 0$. In the latter case, the edge is only built if and only if Player 2 proposes with full effort, and utility is positive if and only if the edge is built. Thus, Player 2 must propose with full effort with positive probability, which is not possible if $\hat{F}$ is continuous (as a continuous distribution on the set $[0,1]$ must take probability zero on any elementary event). From this result, it is easy to find this full-effort probability by equating expected outcome with the previous case: $P(R = 1)\tilde{u} - cg(0) = \tilde{u} - c \implies P(R = 1) = 1 - \frac{c}{\tilde{u}}$.

$$\begin{aligned}
v(r, \hat{F}(\cdot)) &= \mathbb{E}_{\hat{R}\sim\hat{f}}[(\tilde{u} - cg(r))\mathbb{1}_{\{r+R\geq 1\}}] && (3.12)\\
&= P(r + \hat{R} \geq 1)(\tilde{u} - cg(r)) && (3.13)\\
&= \left((1 - \frac{c}{\tilde{u}}) + \frac{c}{\tilde{u}}(1 - F(1-r))\right)(\tilde{u} - cg(r)) && (3.14)\\
&= \left(1 - \frac{c}{\tilde{u}}F(1-r)\right)(\tilde{u} - cg(r)) := k && (3.15)
\end{aligned}$$

We substitute r for $1 - r$ and solve for F to get

$$F(r) = \frac{u}{c}\left(1 - \frac{k}{\tilde{u} - cg(1-r)}\right) \tag{3.16}$$
$$= \frac{1 - g(1-r)}{1 - \frac{c}{u}g(1-r)} \tag{3.17}$$

where boundary conditions for F ($F(0) = 0, F(1) = 1$) allow us to substitute k in the second equality. The PDF is then

$$f(r) = \frac{g'(1-r)(1 - \frac{c}{u})}{(1 - \frac{c}{u}g(1-r))^2} \tag{3.18}$$

which is ≥ 0 if and only if $g'(1-r) \geq 0, \forall r \in [0,1]$, and so $F(r)$ is a valid CDF if and only if g is increasing on $[0,1]$, $g(0) = 0$, and $g(1) = 1$, as required.

■

As in the the Committed Effort CEBG, the value of the game is $\tilde{u} - c$. We have a similar corollary when restricting the upper or lower proposal values:

Corollary 3.2.2 *When the Uncommitted Effort CEBG is restricted such that the upper limit to a player's proposal effort is* $1 - \epsilon$ *for some* $\epsilon \in [0, .5]$*, an MNE is to propose with effort* $1-\epsilon$ *with probability* $1-\alpha$*, and with probability* α *propose with effort drawn from the CDF* $F(r) = \frac{c}{u\alpha}\frac{g(1-\epsilon)-g(1-r)}{1-\frac{c}{\tilde{u}}g(1-r)}, r \in [\epsilon, 1-\epsilon]$*, where* $\alpha = \frac{c(g(1-\epsilon)-g(\epsilon))}{\tilde{u}-cg(\epsilon)}$.

There also exist 'trivial' MNEs for all three cost scenarios, such as proposing only with effort ϵ or $1 - \epsilon$ for $\epsilon \in [0, .5]$, but the results are uninteresting and against the spirit of an MNE: a distribution over strategies.

3.2.3 Uncommitted Shared Effort Scenario

This final scenario is the same as the Uncommitted Effort Scenario in Subsection 3.2.2, except for an important modification on the cost function: we will split the cost of the edge between both players such that the cost each player pays is some function of the ratio of proposed efforts. Player i's utility is then

$$u_i(r_i, r_j) = \left(\tilde{u}_i - cg_i\left(\frac{r_i}{r_i + r_j}\right)\right)\mathbb{1}_{\{r_i + r_j \geq 1\}} \tag{3.19}$$

We call this the Uncommitted Shared Effort CEBG. Having the cost be a function of the other player's proposed effort complicates the math significantly and we were not able to find an explicit functional form. For starters, as in the proof of Theorem 3.2.2, the fact that proposing with full effort and with 0 effort should yield the same utility tells us that the MNE CDF is discontinuous at $r = 1$. That is, we are again looking for a solution of the form "propose with effort 1 with probability $1 - \alpha$, and propose with effort from some distribution $F(r)$ with probability α". Unfortunately, unlike in Theorem 3.2.2, the full effort utility now depends on the other player's CDF due to the ratio-of-efforts cost function and we get:

$$P(R = 1)\left(\tilde{u} - cg\left(\frac{1}{2}\right)\right) + (1 - P(R = 1))\int_0^1 \tilde{u} - cg\left(\frac{1}{r+1}\right)dF(r) = k \tag{3.20}$$

$$P(R = 1)\tilde{u} = k \tag{3.21}$$

which simplifies to

$$P(R = 1) = \frac{\tilde{u} - c\int_0^1 g\left(\frac{1}{r+1}\right)dF(r)}{cg(\frac{1}{1+1}) + \tilde{u} - c\int_0^1 g\left(\frac{1}{r+1}\right)dF(r)} \tag{3.22}$$

We then must find CDF F that also results in the same utility $k = P(R = 1)\tilde{u}$ for any other $r \in (0, 1)$:

$$P(R = 1)\left(\tilde{u} - cg\left(\frac{r}{r+1}\right)\right) + (1 - P(R = 1))\int_{1-r}^1 \tilde{u} - cg\left(\frac{r}{r+t}\right)dF(t) = P(R = 1)\tilde{u} \tag{3.23}$$

which, after cancelling the hefty denominators in $P(R=1)$ and some judicious rearrangement, results in the much cleaner:

$$\frac{\int_{1-r}^{1} \tilde{u} - cg(\frac{r}{r+t})dF(t)}{\int_{0}^{1} \tilde{u} - cg(\frac{1}{1+t})dF(t)} = \frac{g(\frac{r}{r+1})}{g(\frac{1}{1+1})} \tag{3.24}$$

Unfortunately, we were not able to progress further than Equation 3.24. We were, however, able to solve for F and $P(R=1)$ numerically using a simple discretization scheme along with linear programming (LP), as shown in the next section.

3.2.4 Experimental Approximation of Mixed Nash Equilibria on Continuous Games

The discretization relies on the following conjecture:

Conjecture 3.2.1 *Let $\hat{F}_m$ be the MNE CDF of the Uncommitted Shared Effort CEBG played on the discrete strategy space $A_m = \{0, \frac{1}{m}, \frac{2}{m}, \ldots, \frac{m-1}{m}, 1\}$, and let $\hat{F}$ be the MNE CDF of the game with continuous strategy space $A = [0,1]$. Then,*

$$\lim_{m \to \infty} \hat{F}_m = \hat{F} \tag{3.25}$$

That is, we discretize the strategy space uniformly in $[0,1]$, and assert that the solution to the discrete game approaches that of the continuous game as the number of discrete strategies tends to infinity.

Conjecture 3.2.1 can likely can be proven through results from Kroer and Sandholm (2015). The rough idea would be to show that MNEs for the discretized game can be transformed into an ϵ-Nash equilibrium for the continuous game, with $\epsilon \to 0$ as $m \to \infty$.

Note that the discrete game with $m=1$, $A_m = \{0,1\}$ corresponds exactly to the original EBG in Chapter 2.3 where each player either proposes or does not propose.

We use an LP solver to find a feasible assignment of discrete MNE probabilities. The required constraints are $v(r, \hat{F}_m(\cdot)) = k$ for each $r \in A_m$:

$$\sum_{j=m-i}^{m} p_j \left(\tilde{u} - cg\left(\frac{i}{i+j}\right)\right) = k, \forall i \in \{0, 1, \ldots, m\} \tag{3.26}$$

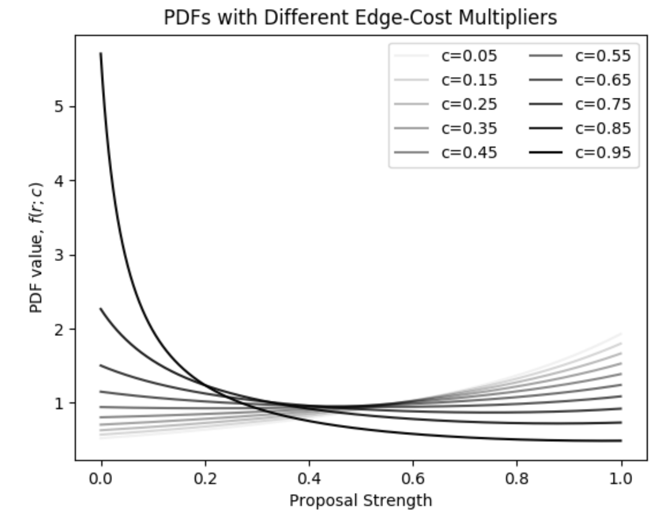
$$\sum_{i=0}^{m} p_i = 1 \tag{3.27}$$

$$p_i \geq 0, \forall i \in \{0, 1, \ldots, m\} \tag{3.28}$$

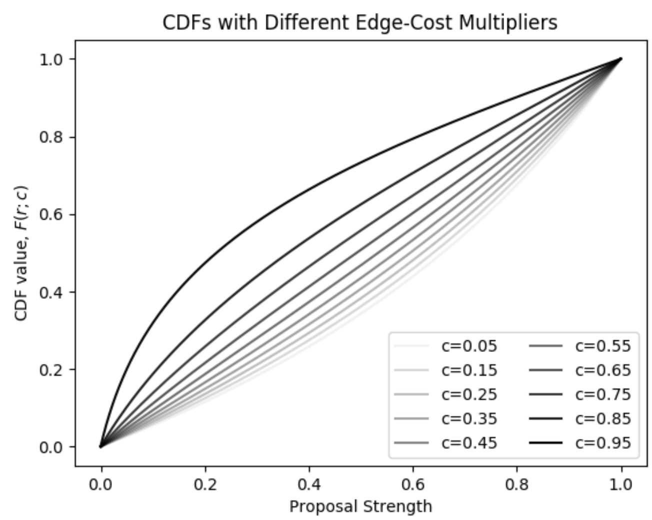
where p_i is the probability of proposing with effort $\frac{i}{m}$. The resulting LP (more accurately a constraint program) has $m + 2$ variables and $m + 2$ constraints, and is fairly tractable even for large m. We run the discretization solver with $\tilde{u} = 1$, $g(x) = x$, and $m = 3000$, where the size was selected such that the returned value of $p_m = P(R = 1)$ is stable to at least 4 decimal places. As shown in Figure 3.2a, at high c values, the likelihood to propose with low effort rises sharply, while the likelihood to propose with full effort, $P(R = 1)$ decreases sharply (Figure 3.4a). Also, the PDF is (numerically verified to be) convex for all (200 tested) values of c. We include a 3D plot to show how the PDF varies smoothly with c (Figure 3.3).

Multiple computational attempts were made toward determining the analytical form of the PDF:

1. Polynomial and rational curve fitting (in Python using scipy.optimize.curve_fit). We tested degree (or combinations of degree, for rational functions) up to 5. Polynomial fits had high error, while rational fitting yields good results starting as early as degree 2 on both numerator and denominator (see Figure 3.4). There may be intuition behind the rational curve fitting, as will be discussed at the end of the section.

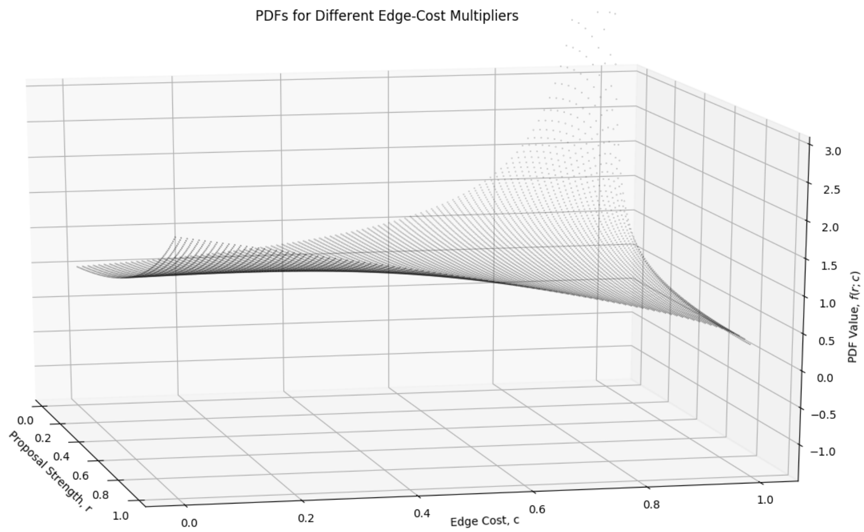


(a)

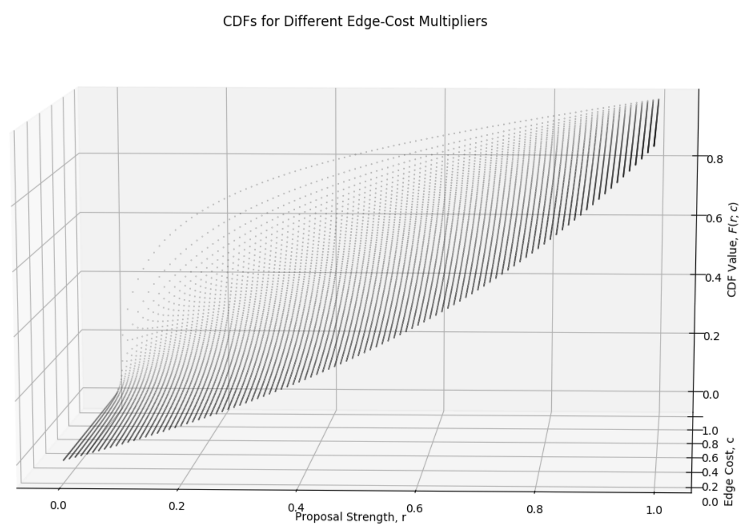


(b)

Fig. 3.2.: PDFs (a) and CDFs (b) of MNE for the Uncommitted Shared Effort Scenario, for various values of c. At high c values, the likelihood to propose with low effort rises sharply.

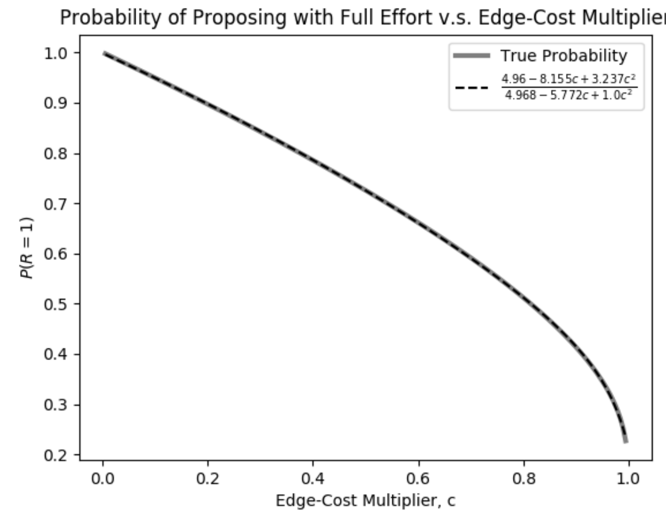


(a)

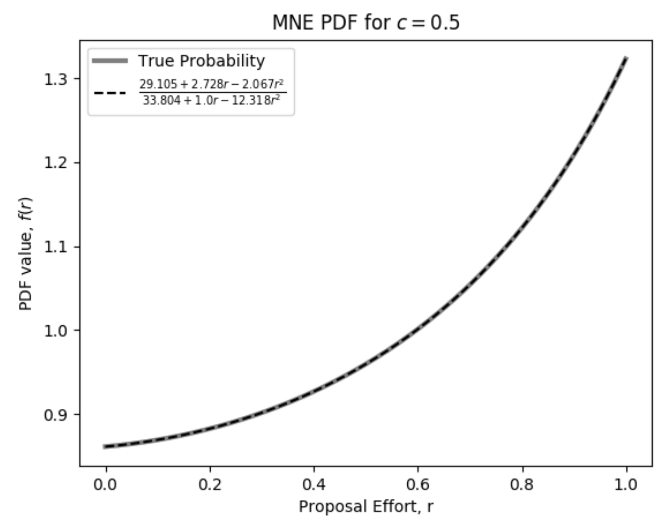


(b)

Fig. 3.3.: 3D plots of PDFs (a) and CDFs (b) of MNE for the Uncommitted Shared Effort Scenario, for various values of c. At high c values, the likelihood to propose with low effort rises sharply.



(a)



(b)

Fig. 3.4.: (a): Probability of full effort proposal as a function of edge-cost multiplier and a fitted rational function. (b): Mixed Nash equilibrium distribution for $c = 0.5$ with a fitted rational function.

2. Symbolic regression using a genetic algorithm (through the Grammatical Evolution R package). This approach yielded ill-fitted results even after ≥ 10 hours of runtime.

3. Symbolic solving (using Sympy in Python). This method involved computationally performing the algebra required to solve for MNE distributions in highly discretized games $m = 1, 2, 3, 4$ (i.e., solving for MNE by hand is easy for $m = 1, 2$, but quickly becomes intractable for humans on paper for much higher than that). For instance, the first four values of $P(R = 1)$ are:

Table 3.1.: $P(R = 1)$ as found through symbolic solving using Sympy for the first few values of m as an attempt to extrapolate patterns.

m	$P(R=1)$
1	$\frac{2(1-c)}{2-c} = \frac{1-c}{1-\frac{c}{2}}$
2	$\frac{18(1-c)(2-c)}{(6-c)(6-5c)}$
3	$\frac{160(3-c)(3-2c)(1-c)}{(2-c)(61-c^2-720c+720)}$
4	$\frac{78750(4-c)(4-3c)(2-c)(1-c)}{34233c^4+728500c^3+3246500c^2-5040000c+2520000}$

The following observations were made:

(a) The numerator takes the form $M_m \prod_{i=1}^{m}(m - ic)$, with no discernible pattern for M_m.

(b) There is no discernible pattern for the denominator.

(c) Both numerator and denominator are polynomials of degree m. This is a fact and not just an observation, as it is verifiable by extrapolation when computing MNEs by hand for the first few values of m.

Symbolic solving without the use of explicit numbers was also attempted, in case some are being factored out between the numerator and denominator, and to see the explicit factors that make up M_m. Unfortunately, this too led nowhere[2] with no discernible pattern in the sequences of integers.

It is also worth noting that in analogous discretizations of the Committed Effort and Uncommitted Effort CEBGs, the value of $P(R = 1)$ does not change with m. In this game, the change is because the players' cost functions depend on each other's actions.

The third observation on symbolic solving explains the why the rational function fitting worked well, specifically that $P(R = 1)$ is the limit of a sequence of rational functions.

Proposition 3.2.1 *Assuming Conjecture 3.2.1, the probability of proposing with full effort in the Uncommitted Shared Effort CEBG can be expressed as*

$$\lim_{m\to\infty} \frac{P_m(c)}{Q_m(c)} \tag{3.29}$$

for some $(P_m(c))_m^\infty$ *and* $(Q_m(c))_m^\infty$*, each a sequence of polynomial functions of degree* m*.*

The proof of Proposition 3.2.1 follows directly from Conjecture 3.2.1, and the fact that $P(R = 1)$ returns a ratio of degree-m polynomials for the discretized game on A_m. The latter fact can be directly observed when solving the simultaneous equations for any particular discretized game.

Sequences of polynomials and rationals (Saff, 1971) often converge to something with an exponential form, and so we posit that the distribution may be similar to a gamma or modified Gaussian distribution.

Note that the expected utilities under these complicated distributions are the exact utilities gained when proposing at full effort (by construction, in fact). Why

[2]Including various attempts at searching the Online Encyclopedia of Integer Sequences

then, aside from theoretical interest, do we care about these MNEs when we could simply propose with full effort for roughly the same result? There are two lenses under which we could answer this, though both are under the larger lens of having a higher-level utility function. Firstly, one might be interested in an increased variance in utility, or perhaps only derive 'true' value above some minimal in-game utility greater than the exact full-effort utility. Secondly, if the other player knows you will be proposing at full effort, she will propose with zero effort. While this constitutes a PNE and does not affect your direct utility, one might imagine some higher-level value gained from not being taken advantage of.

3.3 Correlated Equilibria of the CEBG

Finally, we note that correlated equilibria in continuous games amount to joint distributions over strategy spaces, and are significantly more complicated to analyze (Stein et al., 2011). We do not pursue correlated equilibria beyond the 'trivial' ones corresponding to PNEs and MNEs, but we justify this omission to some extent with the following proposition, where social welfare is the sum of all players' utilities:

Proposition 3.3.1 *In the CEBG with convex edge cost function g, a social welfare maximizing correlated equilibria results when each player proposes with effort $\frac{1}{2}$. If the edge cost function is concave, a social welfare maximizing correlated equilibria results if one player proposes with full effort, and the other with zero effort.*

Proof We show that a PNE that achieves optimal social welfare exists, and so we simply present that as the CE.

The set of PNE are simply all strategies $(x, 1-x), x \in [0,1]$. The corresponding social welfare is $\tilde{u} - g(x) + \tilde{u} - g(1-x)$, and we can seek to minimize $g(x) + g(1-x)$. Suppose that g is convex (concave is shown similarly), then:

$$\begin{aligned} g(x) + g(1-x) &= 2(\frac{1}{2}g(x) + \frac{1}{2}g(1-x)) \\ &\geq 2g(\frac{1}{2}x + \frac{1}{2}(1-x)) \\ &= 2g\left(\frac{1}{2}\right) \end{aligned}$$

by Jensen's inequality. That is, among all PNE, cost is minimized when each player proposes with half effort.

Given any correlated equilibrium, the expected social welfare is found by integrating (with respect to the distribution corresponding to the particular CE) over all possible joint strategies. By the above, no single joint strategy will return a higher social welfare than $2g(\frac{1}{2})$, and so no CE can achieve a higher social welfare. Note that this is not the case in general, as the joint strategy with optimal social welfare may not be a PNE, and so the CE might not attain the optimal value. For instance, see 5.3.2. ∎

Note that for a 'fairer' correlated equilibrium for the concave cost function case, we can randomize equally between the strategies $(1, 0)$ and $(0, 1)$, though strictly speaking this is not required without some explicit measure of fairness between players.

3.4 Multiplayer Edge-Building Game

Next, we stray from the context of edge-building and return to the realm of public goods, if only because building an edge does not make sense for multiple players. Suppose there are $N + 1$ players each contributing funds to build a good, and each player can either propose or abstain from proposing, and all proposers split the cost c of the good evenly. Then, aside from the trivial PNE where exactly one player (any player) is proposing, we can once again find symmetric MNE by equating utilities:

$$1 - c\sum_{r=0}^{N} \frac{1}{1+r}\binom{N}{r} p^r (1-p)^{N-r} = 1 - (1-p)^N \tag{3.30}$$

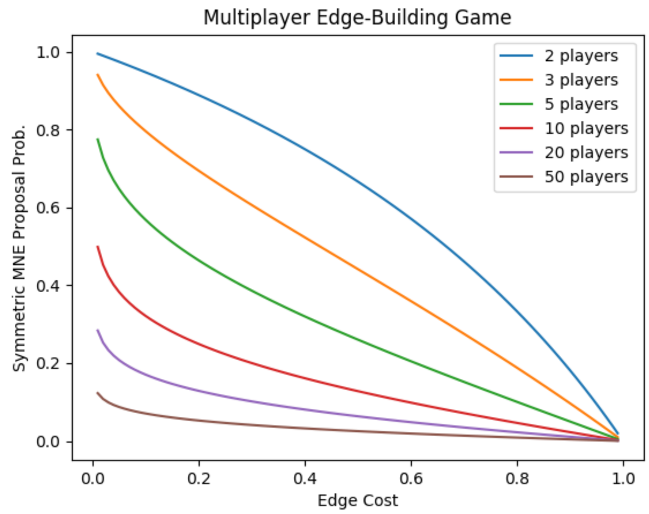
where p is the probability of a particular player proposing an edge in the symmetric MNE. On the LHS is the expected utility from proposing: we have 1 utility from the guaranteed edge/good existence, and the summed term is the expected share of cost when all other players propose with probability p. On the RHS, the expected utility of abstaining is the probability that least one of the other players proposes. This is then a polynomial of degree N, and we will not attempt to find a general closed-form solution, instead showing some experimental results using polynomial root solvers in Python.

We first study the symmetric proposal probability by varying edge-cost and number of players (Figure 3.5):

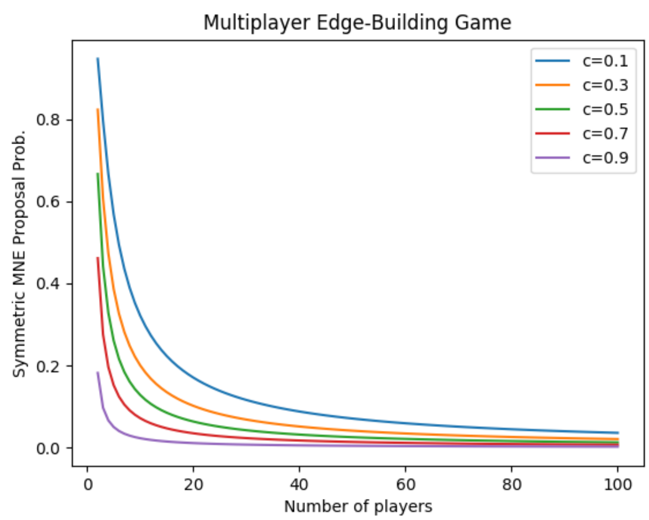
- We see that directionality matches intuition: probability of proposal goes down as either number of players or edge-cost increases.
- With just a few players, we rapidly lose incentive to propose at even extremely low edge costs: At $c = 0.01$, p has already dropped to 49.9% when there are 10 players (Figure 3.5a).
- The main drop in probability comes from just the first few players (Figure 3.5b).

We also somewhat counterintuitively find that expected utility in the symmetric MNE case decreases with increasing players (Figure 3.6). Proposal probability goes down as number of players goes up, as the chance to freeload increases, but any gains from freeloading are more than offset by the decrease in probability of the edge/good actually being made.

Let the number of players now be N, then a simple correlated equilibrium can be found as in Proposition 2.4.2:



(a)



(b)

Fig. 3.5.: (a). Proposal probability in the Multiplayer Edge-Building Game is low even for tiny values of c as long as there are just a few players. (b). Loss in proposal probability is steep for the first few players, and tapers off as player count increases.

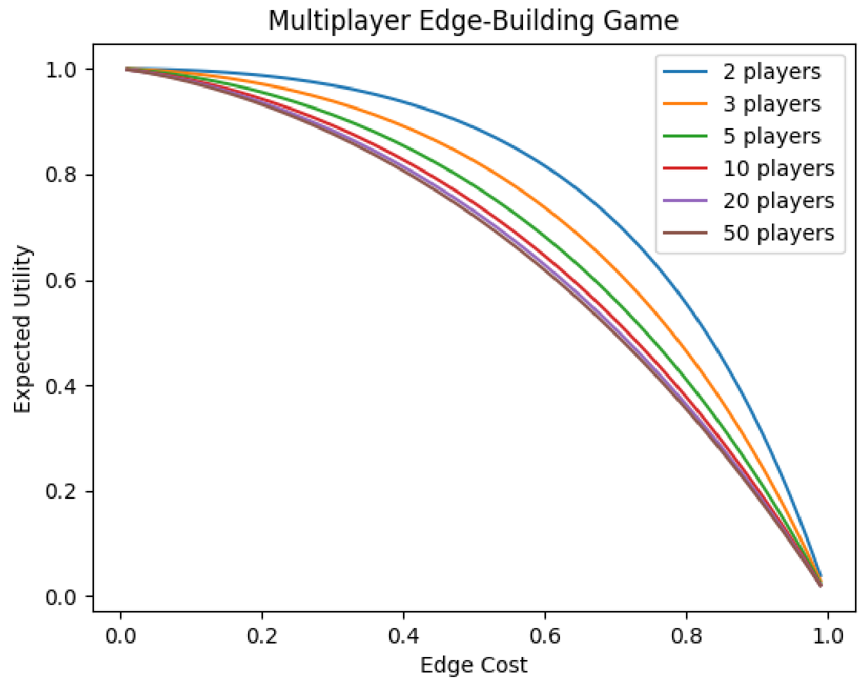


Fig. 3.6.: Expected utility decreases with increasing players. The ability to freeload does not catch up with the loss in overall edge-formation probability.

Proposition 3.4.1 *Playing strategies* $(P, P, \ldots, P), (P, A, \ldots, A), (A, P, \ldots, A),$ $\ldots, (A, A, \ldots, P)$ *with probabilities* $1 - Nx, x, x, \ldots, x$ *is a correlated equilibrium for* $x \geq \frac{c}{N} \leq x \leq \frac{1}{N}$.

The proof of Propsition 3.4.1 is almost identical to 2.4.2 and hence we omit it. More importantly, the expected utility from this CE is $(1 - Nx)(1 - \frac{c}{N}) + x(1 - c) + (N - 1)x = 1 - \frac{c}{N}$. While the MNE yields a worse expected utility with increasing players, the CE yields a better one.

Finally, we note that the polynomial root solver found, for the most part[3], just one root in $[0, 1]$. It might have been conceivable that there are more than one probabilities at which everyone could propose to attain a MNE, but it seems this is not the case.

[3]Other roots did occasionally appear, but we attribute this to rounding errors, as they are sparse, and appear and disappear with no discernible pattern.

It is possible that the uniqueness of root in $[0, 1]$ is provable using Sturm's Theorem, though the polynomial in Equation 3.30 does not unravel well enough for an attempt.

4. THE MULTIPATH CONNECTIONS MODEL

Before we continue on to discuss the Edge-Building Game on networks, we first discuss a new network metric that will we use in our network games, noting that this metric was *not* thought up specifically for the EBG, but as a project on its own. We first briefly review some definitions:

Definition 4.0.1 (Monotone set function) *The set function $f : 2^V \to \mathbb{R}$ is monotone if, $\forall S \subseteq T,\ f(S) \leq f(T)$.*

Definition 4.0.2 (Submodular set function) *The set function $f : 2^V \to \mathbb{R}$ is submodular if, $\forall S \subseteq T, v \notin T$,*

$$f(S \cup \{v\}) - f(S) \geq f(T \cup \{v\}) - f(T)$$

Recall that greedy addition of elements yields a tight $(1 - \frac{1}{e})$-approximation to cardinality-constrained submodular function maximization (Nemhauser et al., 1978).

4.1 The Connections Model

Consider the famous Connections Model by Jackson and Wolinsky (1996). In this model, nodes get utility from each other through a metric on the shortest path distance between them. Formally, the utility of player i on graph g is

$$u_i(g) = w_{ii} + \sum_{j \neq i} \delta^{t_{ij}(g)} w_{ij} - \sum_{j: ij \in g} c_{ij} \tag{4.1}$$

where w_{ij} is the intrinsic value node i derives from node j, c_{ij} is the cost of maintaining edge ij, δ is some discounting factor in $(0, 1)$, and $t_{ij}(g)$ is the length of the shortest path from i to j on graph g with $t_{ij}(g) = \infty$ if i and j are not connected on g.

The Connections model is well studied, with many foundational results appearing in the original paper that loosely speaking defines the field of strategic network formation (Jackson, 2005). For instance, parameter combinations that result in pairwise stability of particular outcome networks. In this work, we do not touch on strategic network formation, but instead use the metric computationally on a network game. As such, we drop the cost terms, instead opting to impose a one-time edge-building cost as in the previous chapters. We further assume nodes do not confer 'intrinsic value' to themselves, and that the value of all other nodes is set to one. We then have:

$$u_i(g) = \sum_{j \neq i} \delta^{t_{ij}(g)} \tag{4.2}$$

We will show that a particular decision model results in submodularity in the Connections Model. First, we cover some notation: Let $N_i(g)$ be the set of neighbors of node i in graph g. For any set of edges S and graph $g = (V, E)$, let $g + S$ be the graph g with all edges in E added to it, $(V, E \cup \{ij : ij \in S\})$.

Suppose nodes are deciding which new neighbors to connect to. That is, if node i is making decisions in graph g, $f_i(S) = u_i(g + S)$, where f_i is defined on all subsets of $X_1 = \{ij : j \notin \{N_i(g) \cup \{i\}\}\}$, the set of edges between i and all of its non-neighbors.

Proposition 4.1.1 *The Connections Model utility function $f_i(S) = u_i(g + S)$ is submodular in new neighbors of a node, $S \subseteq X_1$.*

Proof First, define $f_{ij}(S) = \delta^{t_{ij}(g+S)}, \forall j \neq i$, such that $f_i(S) = \sum_{j \neq i} f_{ij}(S)$. Then, it suffices to show submodularity of $f_{ij}(S)$ for all $j \neq i$.

Note that f_{ij} is monotone in S: if $g_1 \subseteq g_2$, then the shortest paths in g_2 are at least as short as those in g_1.

Let $S \subseteq T \subseteq V, v \notin T$. For any $j \neq i$, suppose that *all* shortest paths in $g + \{T \cup \{v\}\}$ from i to j include node v. We may assume that the first node (other than i itself) on any such path p is v, or else we could shorten the path by shortcutting to v and continuing along the sub-path of p that follows v, a contradiction against p being a shortest path. Since iv is always the first edge, Path p must also be available in $g+\{S\cup\{v\}\}$, so we have $f_{ij}(S\cup\{v\}) = f_{ij}(T\cup\{v\})$, and submodularity is achieved since $f_{ij}(S) \leq f_{ij}(T)$ by monotonicity.

Suppose instead that there is at least one shortest path that does not include node v. Then, the RHS of the submodularity inequality is 0, and by monotonicity we have $f_{ij}(S \cup \{v\}) - f_{ij}(S) \geq 0$. ■

We note the usefulness of submodularity here by showing that greedily adding edges does not always lead to the optimal solution of a cardinality-constrained instance of maximizing the Connections Model utility function. As shown in Figure 4.1, consider a network of six nodes, where five are in a line, and the node making connection decisions is isolated (Figure 4.1). Given a cardinality constraint of two neighbors, an optimal solution is to connect to the second and fourth node in the line, yielding a utility of $2\delta + 3\delta^2$. A greedy approach would first connect to the middle node in the line, leading to a lower final utility of $2\delta + 2\delta^2 + \delta^3$. That is, greedy is not always optimal, but submodularity shows it is a good approximation.

If instead of only being able to add edges to new neighbors, a node has the ability to add edges anywhere in the network, we have the case where $f_i(S)$ is defined on the set of non-existent edges in g, $S \subseteq X_2 = \{ij : ij \notin g\}$.

Fig. 4.1.: Optimal (left) and greedy (right) solutions when node i is constrained to two new neighbors.

Theorem 4.1.1 *The Connections Model utility function* $f_i(S) = u_i(g + S)$ *is not submodular in new edges on a graph,* $S \subseteq X_2 = \{ij : ij \notin g\}$.

Proof We show a counterexample in Figure 4.2, where $S = \phi$ and $T = \{ik\}$ is a single edge. $f_i(S \cup \{v\}) - f(S) = 0$, as the addition of v does not decrease any shortest paths from i, but $f_i(T \cup \{v\}) - f(T) = \delta^2 - \delta^3 > 0$, as the shortest path between i and j is decreased, and so submodularity is violated. ■

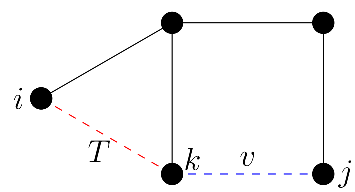


Fig. 4.2.: Let the solid lines be the current edges in the network. Then, adding edge $v = jk$ only improves utility if edge ik already exists. The Connections Model utility function is thus not submodular in edges.

4.2 Multipath Connections Model

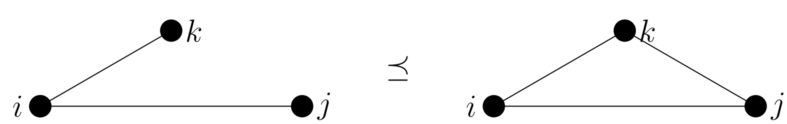


Fig. 4.3.: The graph on the right may be preferred by node i due to a ‘stronger’ tie to node j, but $u_i(g)$ is the same for both in the Connections Model.

The Connections Model does not capture the benefit of additional paths to a node. For instance, in Figure 4.3, both the utility of node i in both graphs is 2δ, but it is reasonable to assume that node i gets more utility in the triangle graph due to an additional path to node j through node k.

We formulate the Multipath Connections Model (MCM) to address this gap. Let $\mathcal{P}_{ij}(g)$ be the set of all simple paths from i to j in g, let $|p|$ denote the number of edges in path p, and let h_1, h_2 be concave, non-decreasing, real-valued functions.

$$u_i^m(g) = h_1\left(\sum_{j \neq i} u_{ij}^m(g)\right) \tag{4.3}$$

$$u_{ij}^m(g) = h_2\left(\sum_{p \in \mathcal{P}_{ij}(g)} \delta^{|p|}\right) \tag{4.4}$$

The concave functions exist to impose diminishing marginal returns in extra paths to any particular node. We similarly define a set function $f_i^m(S) = u_i^m(g + S)$ on subsets of $X_1 = \{ij : j \notin \{N_i(g) \cup \{i\}\}\}$.

Theorem 4.2.1 *The Multipath Connections Model utility function $f_i^m(S) = u_i^m(g + S)$ is submodular in new neighbors of a node, $S \subseteq X_1 = \{ij : j \notin \{N_i(g) \cup \{i\}\}\}$.*

Proof We note that positive combinations of submodular functions are submodular, and that concave, increasing functions of submodular functions are submodular. It thus suffices to show that $f_{ij}^m(S) = \sum_{p \in \mathcal{P}_{ij}(g+S)} \delta^{|p|}$ is submodular.

We will show that $f_{ij}^m(S) = \sum_{v \in S} c_v$ for some c_v. That is, $f_{ij}(S)$ is linearly additive in the value contributed by the elements of S. To show this, we note that a new neighbor v adds to $\sum_{p \in \mathcal{P}_{ij}(g)} \delta^{|p|}$ exactly the value corresponding to all paths from v to j in the initial graph g (ignoring the initial node i). Formally, $c_v = \sum_{p \in \mathcal{P}_{vj}(g-i)} \delta^{|p|+1}$.

Such linearly additive functions (known as modular functions) are also submodular, and so through the compositions of positive combinations and concave functions, $f_i^m(S)$ is submodular.

■

The greedy counterexample shown for the Connections Model in Figure 4.1 also fails for the Multipath Connections Model for at least some concave functions h_1, h_2 and values of δ (for instance, $h_1(x) = x, h_2(x) = \ln x, \delta = 0.3$).

Finally we note that when the Multipath Connections utility function has $h_1(x) = h_2(x) = x$, it is related to Bonacich centrality (Bonacich, 1987):

$$m_b(i,g) = \left([\mathbf{I} - \delta \mathbf{A}_g]^{-1} \cdot \mathbf{1}\right)_i \tag{4.5}$$

where $m_b(i,g)$ is the Bonacich centrality of node i in graph g, $\mathbf{I}$ is the identity matrix, $\mathbf{A}_g$ is the adjacency matrix of g, and $0 < \delta < 1$ as in the Multipath Connections Model.

The Taylor expansion of the first term is then $\sum_{k=0}^{\infty} \delta^k \mathbf{A}_g^k$. We see it is the sum of all walks (not just paths!) of length k, discounted by δ^k. Since all paths are walks, the Bonacich centrality is an upper bound to the Multipath Connections utility function.

In summary, we have defined a new graph metric that we believe patches a gap in the Connections Model. Our model lends less to theoretical analysis, but in showing submodularity we perhaps have a computational foothold for potential future experiments. We use a truncated version of the Multipath Connections model in the next chapter, where we play the EBG on networks, noting that we do not actually utilize the submodularity in this dissertation.

5. THE EDGE-BUILDING GAME ON NETWORKS

5.1 Introductory Remarks

Next, we study the EBG on undirected and unweighted networks (NEBG), where nodes are players and (potential) edges represent a game between the two nodes. Contextually, one could consider trade agreements between countries or friendships on a social network, where the resulting network represents paths of influence between countries or people respectively.

We take two approaches in studying this problem. In the first half of this chapter, we cover a few special cases of network and utility structures where simultaneous equilibria can be explicitly found or efficiently computed. In the latter half, we take a network science view and simulate larger-scale networks where nodes form edges by taking turns to play local games.

In our models, nodes do not have the ability to remove edges once they are formed. That is, we do not take the network stability/strategic network formation approach (Jackson and Wolinsky, 1996; Watts, 2001; Pagan and Dörfler, 2019). In such work, nodes can be seen as having to 'continuously' pay a maintenance cost on the edge, and thus must consider dropping edges that are no longer useful. In contrast, our models only consider the cost of the edge at the moment of edge creation, and allow the edge to provide a continued benefit in future connection games. A similar piece of work is the Network Creation Game of Fabrikant et al. (2003), a one-shot edge-creation game where nodes only consider the cost of building the edge, and attempt to minimize their cost which involves a per-edge cost term, and a penalty for further shortest distances to all other nodes. Our model differs in that nodes can share the cost of an edge, also in the generality of our utility functions (instead of just having a penalty on shortest distances).

In our simulation/experiment portion, we study the properties of the resulting graphs as we vary parameters in the nodes' utility functions. While this can be seen as a kind of mechanism design on networks, we differ from existing literature (Galeotti et al., 2019; Brown and Patange, 2020; Wang et al., 2006) wherein nodes' utilities are tweaked on a static network.

5.2 General Model Setup

There are numerous modeling setups to decide between, with the two most significant being as follows:

1. Allowable edges: Which pairs of nodes can play the edge-building game? Does it make sense to allow *any* two nodes to play with each other? If not, how should these parings be limited?

2. Game simultaneity: Do all nodes on the network play their games at the same time, or do they take turns?

We address the first question by using a input "skeleton" g_0 of the network that defines allowable edges: a game can only be played between two nodes if the potential resulting edge exists on the skeleton network. Using such a skeleton network serves as a general framework for various choices of allowable-edge scenarios. For instance, allowing any pair of nodes to play the game is equivalent to having the complete network as the skeleton. Without loss of generality we assume that the skeleton is connected, as disconnected components can be played as independent games.

As for simultaneity, we demarcate three separate cases. First, the fully sequential case, where pairs of nodes take turns to play, and only one edge is under consideration at any given moment. Second is the node-level simultaneous scenario, where nodes take turns making decisions, but each node makes all its connection decisions at the same time, playing one game with all its neighbors as opposed to one game with each of its neighbors. Finally, the fully simultaneous case is where all nodes make all connection decisions at the same time.

Which version of simultaneity makes the most sense? Outside of our our context of edge-creation on networks, decision-making in general has been shown to be more effective when simultaneous choices are presented and made (Basu and Savani, 2017; Bohnet et al., 2016), with sequential decisions leading to lower post-selection satisfaction (Mogilner et al., 2013). In contrast, a somewhat separate issue is whether or not choices are actually made simultaneously versus separately - a study of 83 decisions by the board of a mid-sized company found that 40% were yes-or-no type decisions, and 55% where decisions between just two options (Gemünden and Hauschildt, 1985). Decision points tend to arrive in a manner such that only sequential choices are feasible. Furthermore, the aptness of each model also depends on the amount of time between decisions and updates to the system - if decisions arrive one-at-a-time, but the system only updates every so often, the decisions must effectively be made simultaneously. We leave the discussion of model suitability at the door for now, noting that each model has its benefits and drawbacks across different contexts.

We study a few special cases across both simultaneous and sequential games, and various skeleton network setups. However, when it comes to larger-scale experiments, computational tractability limits us to (1) fairly sparse skeleton networks and/or (2) playing a sequential game where pairs of nodes take turns playing.

At a more detailed level, we also need to decide on utility functions - both edge-utility and edge-cost functions as seen in Figure 2.3 - and the objective function (or social welfare) for whomever is studying or designing the network.

Contextually, a node cares about a new edge because it confers an improvement to the node in the network. Thus, for edge-utilities, we use the difference between a node's centrality measure in the proposed network (with the potential new edge) and its centrality measure in the existing graph. Note that in doing so, we implicitly impose the condition that adding edges will only ever benefit a node. In particular we use the following **m**easures, where $m_x(i, g)$ is the centrality measure x for node i in graph g:

- Size of connected component (CC). We subtract one from the node's CC size so that the utility gained in the two player case is 1 and the scenario is equivalent to the vanilla EBG.

$$m_{cc}(i, g) = |j : t_{ij}(g) < \infty| - 1 \tag{5.1}$$

  where $t_{ij}(g)$ is the shortest distance between nodes i and j in graph g.

- Efficiency, the average reciprocal of shortest distances from the node in questions to all other nodes.

$$m_{eff}(i, g) = \frac{1}{|N_i(g)|} \sum_{j \in N_i(g)} \frac{1}{t_{ij}(g)} \tag{5.2}$$

- Multipath Connections Model utility (Chapter 4), the scaled and weighted utility obtained from considering all paths to other nodes:

$$m_{mp}(i,g) = h_i\left(\sum_{j\neq i} h_2(\sum_{p\in\mathcal{P}_{ij}(g)} \delta^{|p|})\right) \tag{5.3}$$

where h_1 and h_2 are concave, non-decreasing functions, $0 < \delta < 1$, and $\mathcal{P}_{ij}(g)$ is the set of all simple paths from i to j in g.

We mostly apply these centrality measures only over a local subgraph centered on the node being measured, capturing the notion that nodes do not have full information about the network (alternatively, that nodes display bounded rationality and cannot compute the utility gained on the whole network) while also reducing the computational burden. Define the set of all nodes within distance r of node i on graph g to be $V_i^r(g)$, and let g_i^r be the subgraph of g induced by $V_i^r(g)$. Then, the radius-r version of measure p is $m_p(i, g_i^r)$.

We first study analytical Nash equilibria on the fully simultaneous NEBG.

5.3 Nash Equilibria in the Simultaneous NEBG for Connected Component Size Edge-Utility

Nash equilibria (pure and mixed) are (PPAD-)hard to compute in general (Daskalakis et al., 2009). Something as seemingly simple as determining existence of pure Nash equilibria turns out to be NP-hard even under highly restrictive conditions (Gottlob et al., 2005). Correlated equilibria seem promising, as they can be found and optimized over in time polynomial in the input length of the problem through linear programming. Unfortunately, in a game on networks, the input length is exponential in the number of players as the joint strategy space is the Cartesian product of each player's strategies. CEs for certain classes of multiplayer games can still be found efficiently (Papadimitriou and Roughgarden, 2008).

Fortunately, equilibria for the NEBG are easy to find in some cases. Suppose all nodes have CC size as their edge-utility, with the standard shared cost.

Proposition 5.3.1 *When all nodes in a simultaneous NEBG have connected component size as their edge-utilities, a PNE can be found in polynomial time.*

Proof We characterize all PNEs, at which point it is trivial to find them. The set of PNEs is exactly the set of joint strategies where (1) at most one node is proposing on each (allowed) edge and (2) the resulting network is a spanning tree.

In such a spanning tree, no node has incentive to propose an edge not already on the tree. CC utility is maximum at $N-1$ (where N is the size of the network) so there is no extra utility to be gained from a new edge, but edge-cost would still be incurred. No node that is proposing an edge has reason to abstain, as doing so would disconnect the tree due to condition (1), and necessarily reduce the node's utility. As for the converse, any PNE must have at most one node proposing on an edge, or the two nodes would be able to unilaterally deviate for reduced cost and no reduction in edge-utility. Any joint strategy that results in an unconnected network cannot be a PNE as at least one edge can propose for an increase in utility (under the assumption of a connected skeleton network). Similarly, a strategy corresponding to a network larger than a tree must have a cycle, and an edge along that cycle can be dropped.

It is clearly easy to find a such a joint strategy by first efficiently finding any spanning tree, then arbitrarily assigning each of the $N-1$ edges to one of its two nodes. ■

Corollary 5.3.1 *PNEs to the simultaneous NEBG with connected component size as edge-utilities can be uniformly sampled from in polynomial time.*

Proof Each PNE corresponds to exactly one spanning tree, and each spanning tree has the same number of PNEs: 2^{N-1}, one for each assignment of each $N-1$ edge to either of its two nodes. Thus, we can sample uniformly from the set of all PNEs by first sampling uniformly from the set of all spanning trees (say, using Wilson's Algorithm by Wilson (1996)), then by sampling uniformly on the returned spanning tree by randomly assigning each edge to one of its nodes. ■

As in Section 3.3, appropriate conditions on the edge costs allow us to optimize for social welfare. Recall that in the continuous EBG, when the edge cost function is convex, the optimal solution is for each edge to propose with effort $\frac{1}{2}$. When the edge cost function is concave, the solution is for one edge to propose at full strength, and the other at zero strength. We can define a cost structure in the NEBG in a way analogous to convexity/concavity in the continuous case, and derive similar results.

Let c_s be the cost paid when **s**haring the edge-cost, and c_a be the cost paid when paying for the edge **a**lone. We assume that nothing is ever paid when abstaining. Then, the contextual requirement is $0 < c_s \leq c_a < 1$. If $c_a > 2c_s$, we call the costs cooperative, and it is analogous to having convex cost functions in the continuous case, where having both players split the cost sums to less than the cost for a sole-proposer. Similarly, if $c_a < 2c_s$, we call the costs uncooperative, and it is analogous to the concave case.

Corollary 5.3.2 *In the simultaneous NEBG with cooperative costs and connected component utility function, the optimal social welfare is $(N-1)(N-2c_s)$. There is no correlated equilibria that can attain this value, even in expectation.*

Proof As in Proposition 5.3.1, it can be easily shown that a joint strategy that results in a spanning tree maximizes social welfare. From there, the edge allocation that minimizes the sum of all costs is the one where each edge is paid for by both nodes, as is clear from the $c_a > 2c_s$ requirement. Thus, each node gets $N-1$ edge-utility, the total cost is $N-1$ edges at c_s per edge, and the social welfare is $N*(N-1)-(N-1)*2c_s$ as required.

To determine that no correlated equilibrium can attain this social welfare, notice that (again, using similar logic as in Proposition 5.3.1) the only joint strategies that result in the optimal social welfare are spanning trees in which all edge costs are shared equally. The expected social welfare of a correlated equilibrium that achieves the optimal social welfare must then only have positive probability on such joint strategies. A node presented with nothing but such joint strategies is then free to deviate from proposing to abstaining, and the correlated equilibrium condition is violated. ■

It is yet unknown what the optimal social welfare correlated equilibrium is in the cooperative costs scenario.

Corollary 5.3.3 *In the simultaneous NEBG with uncooperative costs and connected component utility function, the optimal social welfare is* $(N-1)(N-c_a)$. *A corresponding PNE can be efficiently found.*

Proof From Proposition 5.3.1 and the condition $c_a < 2c_s$, it is clear that the optimal social welfare is attained only when each edge in a spanning tree is proposed to by exactly one node, and that such joint strategies can be efficiently found. ■

For this uncooperative case, we can further consider a notion of fairness by achieving the optimal social welfare while requiring that each node has the same expected utility.

Corollary 5.3.4 *In the simultaneous NEBG with uncooperative costs and connected component utility function, a correlated equilibrium with optimal social welfare that has the same expected utility for each node can be efficiently found.*

Proof We construct a CE by randomly selecting a node to freeload off the rest. Given any social welfare maximizing joint strategy - i.e., one that results in a spanning tree where each edge is paid for by exactly one node - there is always exactly one freeloading node that does not pay any edge cost, while reaping the same benefit as all the others. For any particular spanning tree, we are free to select the freeloading node by rooting the tree at that node and having all other nodes propose to their parents. This way, each other than the freeloading root pays for exactly one edge (see Figure 5.1).

To construct a fair CE, we select one joint strategy for each node, wherein the relevant node is the freeloader. Note that in such a CE, nodes do not have incentive to deviate: deviating from abstaining (being the freeloader) will grant no extra edge-utility, and deviating from proposing will either disconnect the network (by abstaining), or grant no additional utility (by swapping the proposed edge or proposing to new edges). The CE then has each joint strategy selected with probability $\frac{1}{N}$.

■

5.4 Correlated Equilibria for the Simultaneous NEBG for the Edge-Building Game on Networks

We next study correlated equilibria on various setups of network games. First, we review the mathematical definition of a correlated equilibrium, and define some terminology for games on networks.

Let there be N players, and for all $i \in [N]$ let player i's (finite) set of strategies be S_i. Then, $\mathbf{S} = \prod_{i=1}^{N} S_i$ is the set of all joint strategies, and $S_{-i} = \prod_{j \neq i} S_j$ is set of joint strategies other than S_i. Let $u_i(\mathbf{s}) = u_i(s_i, s_{-i}), \mathbf{s} \in \mathbf{S}, s_i \in S_i, s_{-i} \in S_{-i}$ be the utility player i receives when joint strategy $\mathbf{s}$ is played. Then, a correlated equilibrium is any probability distribution $x(\mathbf{s})$ over the $\mathbf{S}$ that satisfies the following constraints:

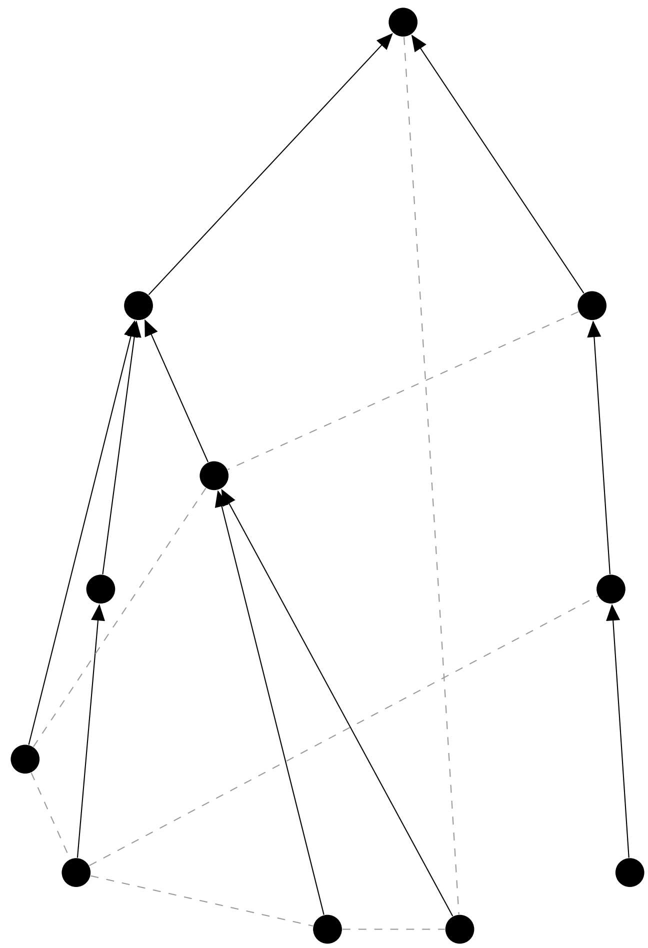


Fig. 5.1.: An example of a rooted spanning tree where the top node freeloads off the rest of the network, where dashed edges are edges of the skeleton network that are not built. To construct a social-welfare-optimal correlated equilibrium with equal expected utilities: (1) Select a node to freeload. (2) Create a spanning tree. (3) Root the freeloading node. (4) Have all other nodes propose to their parents. (5) Repeat (1) through (4) for each node, and play each of the N trees (joint strategies) with equal probability.

$$\sum_{s_{-i}\in S_{-i}} u_i(s_i, s_{-i})x(s_i, s_{-i}) \geq \sum_{s_{-i}\in S_{-i}} u_i(s_i', s_{-i})x(s_i, s_{-i}), \forall i \in [N], s_i, s_i' \in S_i \tag{5.4}$$

$$\sum_{\mathbf{s}\in\mathbf{S}} x(\mathbf{s}) = 1 \tag{5.5}$$

$$x(\mathbf{s}) \geq 0, \forall \mathbf{s} \in \mathbf{S} \tag{5.6}$$

A more intuitive version (but less programmable) of constraint 5.5 is where we divide by s_i on both sides to replace $x(s_i, s_{-i})$ with $x(s_{-i}|s_i)$: a correlated equilibrium is when, conditioned on being shown a particular strategy for themselves, nobody gains any expected utility when deviating.

Correlated equilibria always exist, trivially so because Nash equilibria are a special case of CEs. At first glance, CEs seem easy to find, since all the constraints are linear and have length polynomial in the size of the game (that is, in the number of joint strategies and players). One might then attempt to optimize some welfare function by having the objective function of the LP be a linear combination of joint strategy probabilities. Unfortunately, the number of joint strategies is exponential in the number of players: suppose each player has at most k neighbors. Then, each player has up to 2^k strategies even in the case of binary actions on each edge. With N players, this is 2^{nk} joint strategies, and so this linear program cannot directly be solved efficiently. In general, optimizing over correlated equilibria is NP-hard in multiplayer games (Papadimitriou and Roughgarden, 2008) outside of specific classes of games. Among those classes are graphical games (Kearns et al., 2013), games that can be represented as a graph where each player's utility only depends on the strategies of neighboring nodes.

Formally, a graphical game on N nodes/players is a tuple $(g, \mathcal{M})$ where g is a graph and $\mathcal{M}$ is a set of N matrices $M_i, i \in [N]$ such that M_i is a $(|N_i(g)| + 1)$-

dimensional matrix defining the utilities of node i based on the strategies of i and i's neighbors on g. The size of the game is then $O(2^d)$, where $d << N$ is the size of the largest local neighborhood.

Then, the NEBG on graph $g = (V, E)$ where edge-utilities are radius-r centrality measures (i.e., $m_p(i, g_i^r)$) can be seen as a graphical game $(g', \mathcal{M})$, where the graphical game network g' is induced by g and r: $g' = (V, E'), E' = \{(i, j) : t_{ij}(g) \leq r\}$ (see Figure 5.2), and the game matrices $M_i, \forall i$ are the joint strategy/utility tables over nodes $\{\{i\} \cup N_i(g)\}$.

When our skeleton network g is a tree and the radius r is 1, then the induced graph g' is equivalent to g itself, and we can directly apply Theorem 8 (Efficient Tree Algorithm) in Kakade et al. (2003): In tree graphical games, optimal correlated equilibria (with regard to a linear objective function) can be found in time polynomial in the size of the graphical game. Granted, having $r = 1$ on top of g being a tree results in our edge-utilities/measures (and thus the whole game) being much less meaningful. For instance, the CC measure with radius 1 just reduces to number of neighbors.

If we relax the $r = 1$ condition but impose a maximum degree on the tree, we can still find optimal correlated equilibria somewhat efficiently. To show this, we briefly review the definition of tree decomposition and tree-width.

Definition 5.4.1 (Tree Decomposition) *Given a graph $g = (V, E)$, a tree decomposition on g is a tuple $(T, \mathbf{X})$ such that T is a tree with m nodes, and $\mathbf{X} = \{X_1, \ldots, X_m\}$ is a set of subsets of V such that each node i of T corresponds exactly to subset X_i. Additionally, the following three conditions must hold:*

1. *$\cup_m X_m = V$, each node in g appears in at least one node of T.*

2. *$\forall (i, j) \in E, \exists k$ such that $i \in X_k$ and $j \in X_k$. Each edge in g appears in at least one node of T.*

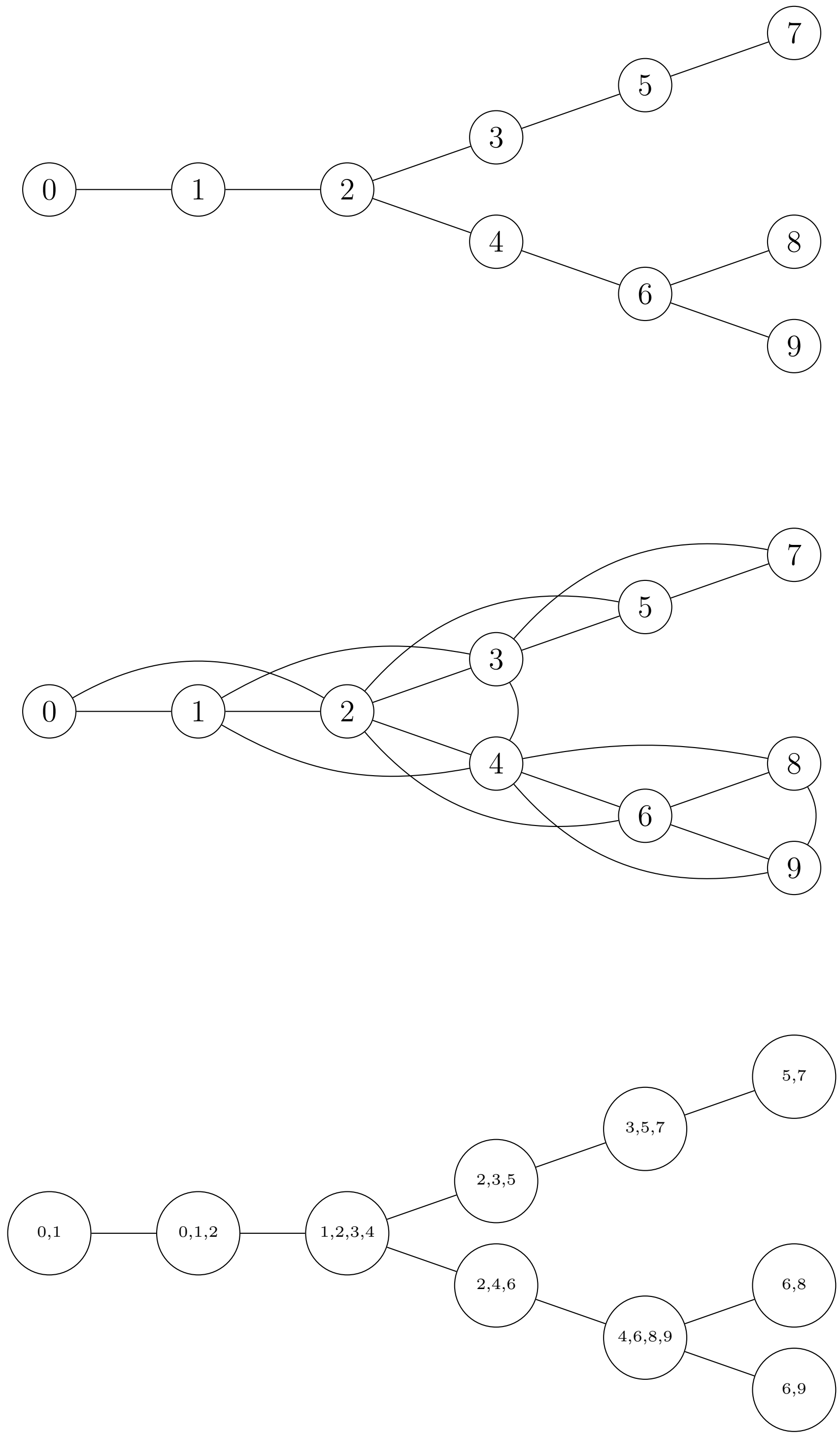


Fig. 5.2.: Top: NEBG network g. Middle: Graphical game graph g' induced by g with radius $r = 2$. Each node's utilities only depends on the actions of its neighbors in g'. Bottom: Tree decomposition T on g' with tree-width $k\frac{(k-1)^{\lceil r/2 \rceil}-1}{r-2}$, where k is maximum degree of g.

3. *$\forall X_a, X_b$, let $P_{ab} \subseteq \mathbf{X}$ correspond to the unique path between X_a and X_b. Then, $\forall i \in X_a \cap X_b$ and $X_c \in P_{ab}$, we must have that $i \in X_c$. That is, for all $i \in V$, the sets in $\mathbf{X}$ containing i must correspond to a set of connected nodes in T.*

The tree-width of a tree decomposition is then $\max_i |X_i| - 1$, one less than the size of the largest node of the decomposition.

Lemma 5.4.1 *Let g' be the graph of the graphical game induced by a NEBG on a tree g with maximum degree k, where edge-utilities are radius-r centrality measures. The tree-width of g' is upper-bounded by $k\frac{(k-1)^{\lceil r/2 \rceil}-1}{k-2} \leq 2k^{\frac{r}{2}}$.*

Proof Given g' induced by g as in Figure 5.2 (middle), the tree decomposition $(T, \mathbf{X})$ on g' is simply to set $T = g$ and $X_i = \{j : t_{ij}(g) \leq \lceil \frac{r}{2} \rceil\}, \forall i$ (See Figure 5.2 (bottom)). T is clearly a tree since g is a tree, and so we proceed to check the three requirements in Definition 5.4.1:

1. For each node i in g', i appears in X_i.

2. Each edge (i, j) in g' corresponds to a pair of nodes i, j with distance $\leq r$ in g, by construction of g'. Thus, there exists some node k in the unique path on g between i and j such that $t_{ik}(g) \leq \lceil \frac{r}{2} \rceil$ and $t_{jk}(g) \leq \lceil \frac{r}{2} \rceil$, and so X_k contains both i and j by definition of X_k.

3. For each node i in g', any two sets X_j and X_k that both contain i are at most r distance away on $T = g$ by construction, and so the nodes on T that contain i are connected.

The tree-width is then the number of neighbors in radius $\lceil \frac{r}{2} \rceil$ of the original graph g, easily found to be $k + k(k-1) + \ldots + k(k-1)^{\lceil \frac{r}{2} \rceil - 1} = k\frac{(k-1)^{\lceil \frac{r}{2} \rceil}-1}{k-2} < \frac{k}{k-2}k^{\frac{r}{2}} < 2k^{\frac{r}{2}}$, noting that the $k = 2$ case is when g is a line in which case the tree-width of g' is at most $2\lceil \frac{r}{2} \rceil \leq r + 1$.

■

Then, we apply a result from Papadimitriou and Roughgarden (2008): (Theorem 5.9) Optimal (again with respect to a linear combination of joint strategy probabilities) correlated equilibria of graphical games with bounded tree-width can be found in time polynomial in the size of the game, which is exponential in k, the maximum degree of the graph.

Corollary 5.4.1 *In the NEBG on a tree with maximum degree k and radius-r centrality measures as edge-utilities, optimal correlated equilibria can be found in time exponential in $k^{\frac{r}{2}}$.*

Proof The proof is a modification of the proof of Theorem 5.9 from Papadimitriou and Roughgarden (2008), where there are $O(2^{2k^{\frac{r}{2}}})$ subproblems due to the $2k^{\lceil r/2 \rceil}$ tree-width upper bound of Lemma 5.4.1. ■

5.5 Sequential Edge-Building Game on Networks

Next we study the sequential version of the NEBG, where nodes take turns (in pairs) playing the EBG. Contextually, this model is suitable when there is little-to-no time lag between connection decisions and edge formation (and thus system-updating), much like in a social network.

Rather than the Nash equilibria, we are interested in the network formation process, where we use the EBG to generate complex networks and study metrics on the output network. We adopt a mechanism design approach and set up the model to be parameterized in some way, so that we may vary the model in order to study variation in the output network.

At a high level, the model runs as follows:

1. Iterate over the edges (pairs of nodes)

2. Determine the MNE of the two nodes with respect to the current state of the system (and of course, their utility functions)

3. Have nodes propose to each other with the MNE probabilities. If at least one node actually ends up proposing, the edge is made and the graph is updated.

4. Repeat until some maximum number of edges is reached.

In practice, we make some modifications such as normalizing utility functions or bounding proposal probabilities (see Algorithm 1, Appendix).

Algorithm 1 High-level framework for sequential NEBG. Examples of helper functions: $\tilde{m}$, a normalization of $m(i, g + ij) - m(i, g)$. b, a linear interpolation between explicit proposal probability bounds.

procedure SEQUENTIAL-NEBG($g_0 = (V, E), \tilde{m}(i, g, j), p_{\min}, p_{\max}, M, c$)
    $g \leftarrow (V, E' = \{\phi\})$
    while $|E'| < M$ **do**
        Select edge $(i, j) \in E$ ▷ Only look on skeleton network g_0
        $u_i \leftarrow \tilde{m}(i, g, j)$ ▷ Determine raw utility i gets from j
        $u_j \leftarrow \tilde{m}(j, g, i)$
        $\tilde{u_i} \leftarrow b(u_i, p_{\min}, p_{\max})$ ▷ Scale raw utility so that MNE probabilities
        $\tilde{u_j} \leftarrow b(u_j, p_{\min}, p_{\max})$ ▷ are well-defined
        $p_i \leftarrow \frac{u_j - c}{u_j - \frac{c}{2}}$
        $p_j \leftarrow \frac{u_i - c}{u_i - \frac{c}{2}}$
        if $\text{rand}_1() < p_1$ **or** $\text{rand}_2() < p_2$ **then**
            $E' \leftarrow E + (i, j)$
    return g

Throughout the chapter, we make comparisons with the $G(n, M)$ random graph model, where a graph is sampled uniformly at random from the set of all graphs with n nodes and M edges. We find that the metrics on our output graphs are close in value to those of the random graphs, though we still observe a statistically significance difference between the two for particular graph metrics. That they are close is no great surprise: our model selects edges at random, then forms the edge with some probability according to the utility functions of the two nodes, with the utilities dependent on the current state of the network. Then, we can see the $G(n, M)$ network as a specific case of our model, where the utilities are infinite and the edge proposal probabilities are always 1. We note the omission of other random graphs like the Barabási-Albert (BA) preferential attachment model (Albert and Barabási, 2002) or the Watts-Strogatz (WS) small-world model (Watts and Strogatz, 1998). We focus on the $G(n, M)$ model due to its relationship with ours. We expect high values of all three metrics for the BA level, due to pathways through the hubs, and high values for local CC Size and average efficiency for the WS model, due to the model's emphasis on short connections. We do not expect the WS model to perform as well on the average multipath value, with fewer paths to other nodes due to the ring structure (and the rewired paths).

Note that if the edge proposal probabilities are not 1 but are fixed throughout the simulation, the model is still equivalent to the $G(n, M)$ network. In the parameter setups we have chosen, the average edge-formation rate (number of edges divided by the number of attempts at edge formation) is somewhere between 70 and 90 percent. That is to say, the 10 to 30 percent of the time nodes decide not to form an edge contributes entirely to the statistically significant differences between the two sets of output networks.

5.5.1 Effect of Utility Function on the Same Utility in the Output Network

Suppose we simulate games on networks where nodes use a particular function $m(i, g)$. Then, what does the final network look like with respect to $m(i, g)$? We use the a real collaboration network as a baseline, ca-netscience from Rossi and Ahmed (2015) with $N = 379$ nodes and $m = 978$ edges, chosen for contextual relevance and appropriateness of size for computation. We use the three measures mentioned in Section 5.2: Size of local connected component in radius 3 of the node; Efficiency, the

Table 5.1.: Effect of a particular utility function on the utility function on the final graph, with best values in each column bolded. Utility functions/models used: LCC: Local Connected Component with radius 3. EFF: Efficiency over entire network. MPC: Multipath Connections Model. GNM: $G(n, M)$ Erdos-Renyi graph. CA: real network, ca-netscience, with $N = 379$ nodes and $M = 914$ edges.

Model	Avg. Local CC Size	Avg. Efficiency	Avg. Multipath Value
LCC	$\mathbf{114.61 \pm 0.72}$	0.13674 ± 0.00045	316.72 ± 0.94
EFF	112.67 ± 0.58	$\mathbf{0.13708 \pm 0.00035}$	$313.81, \pm 0.80$
MPC	112.69 ± 0.72	0.13693 ± 0.00035	314.95 ± 1.01
GNM	111.81 ± 0.66	0.13681 ± 0.00026	313.71 ± 1.06
CA	52.253	0.10162	**389.38**

average reciprocal of shortest distances from a node to all other nodes, taken over the entire network; The Multipath Connections Model, with $\delta = 0.1$ and search radius 3. For each metric, we generate 30 graphs with the same number of nodes and edges as ca-netscience, and compare the same metrics on the output graphs, averaged over all nodes. We also include $G(n, M)$ Erdos-Renyi edge-based random graphs as another baseline. The results are summarized in Table 5.1 with 95% confidence intervals.

We see that in general the output networks are close to the $G(n, M)$ random graph, as was mentioned at the end of the previous section. However, we notice that the LCC model returns networks with the best average local CC size, by a statistically significant margin from the random network (p-value $\approx 10^{-7}$). The efficiency model returns networks with the best efficiency, but this effect is not significant (p-value $= 0.239$). The multipath connections model, however, fails to demonstrate any benefit over the competitors, and the original network actually claims the highest metric value. We conclude that there is mild evidence to suggest that it may be possible to generate output networks with strength in desired metrics by having node utilities be related to that metric.

5.5.2 Effect of Varying a Single Parameter in a Utility Function

Next focus on varying parameters for one edge-utility, opting for the local multipath connections utility from Chapter 4[1], which is parameterized by the local radius within which to consider paths, and the discounting factor for increasing path length (δ). We show the effect of varying δ on the output graphs, for the local connected component measure and efficiency measure as in Table 5.1, along with the following three network measures:

- Transitivity, the fraction of all possible triangles:

$$m_t(g) = 3 * \frac{\text{Number of triangles in } g}{\text{Number of triads in } g} \tag{5.7}$$

- Average clustering, the fraction of edges that a node's neighbors have between themselves.

$$m_c(g) = \frac{1}{N} \sum_{i=1}^{N} \frac{2|\{r : p, q \in N_i(g), (p, q) \in E\}|}{|N_i(g)|(|N_i(g)| - 1)} \tag{5.8}$$

- Average betweenness, where the betweenness of node i the sum of fractions of shortest paths that i is on.

$$m_{be}(g) = \frac{1}{N} \sum_{i=1}^{N} \sum_{j \neq i \neq k} \frac{\sigma_{jk(i)}}{\sigma_{jk}} \tag{5.9}$$

[1] The multipath connections utility requires two real-valued concave increasing functions, h_1 and h_2 which impose decreasing marginal utility with respect to additional paths. Here, we use $h_1(x) = h_2(x) = x$ for simplicity

where σ_{jk} is the number of shortest paths between nodes j and k, and $\sigma_{jk}(i)$ is the number of those paths that include node i.

As shown in Figures 5.3, 5.4, and 5.5, we again see that the model outputs are statistically significantly different from random graphs, even if the difference is small. Varying δ has an effect on some metrics (efficiency, average betweenness, and local CC size), but not on other (average clustering and transitivity). The metric values are just as far off from that of the real network of the same size, ca-netscience, as in Table 5.1, and we omit the real network values from the plot to emphasize the difference between our model and the random network.

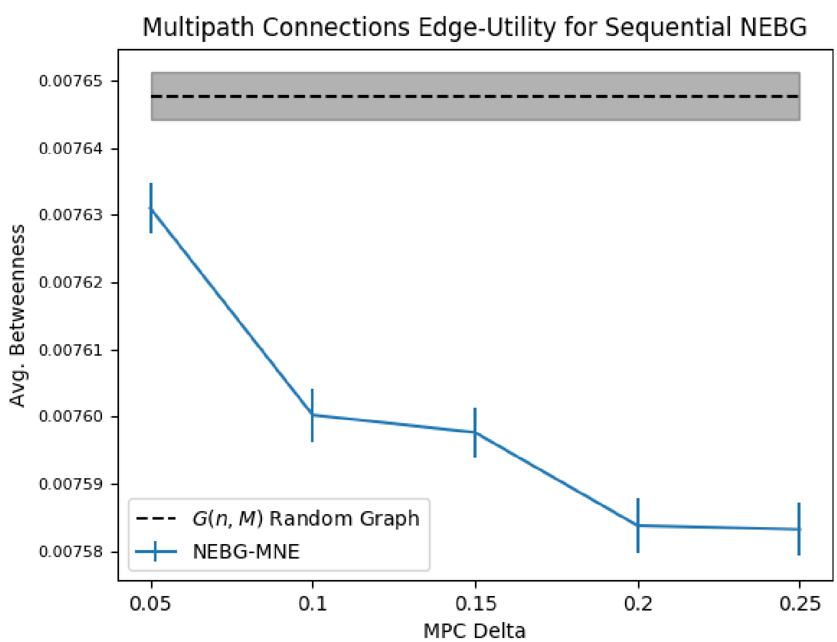
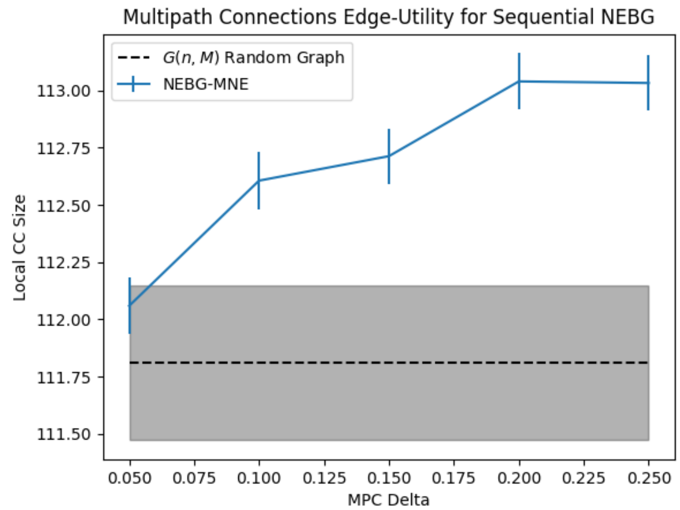
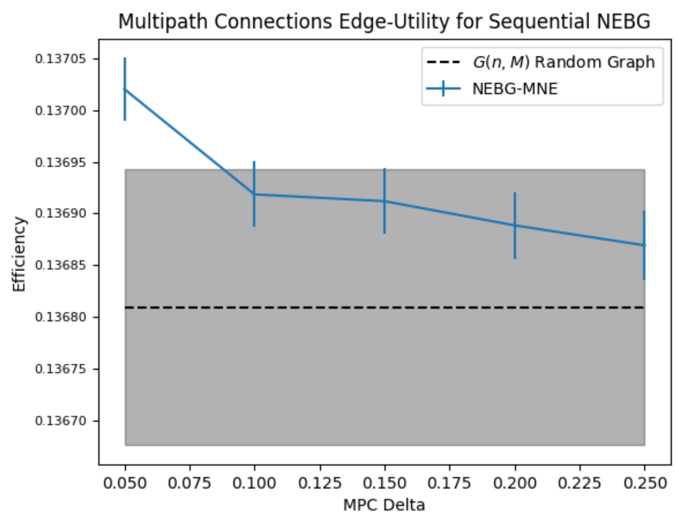


Fig. 5.3.: The sequential NEBG model with multipath connections utility function, with a $G(n, M)$ random graph of the same size $(n = 379, M = 914)$ for comparison. We see slight but statistically significant differences between the two sets of networks for average betweenness, and monotonicity with the multipath connections model discount factor δ.

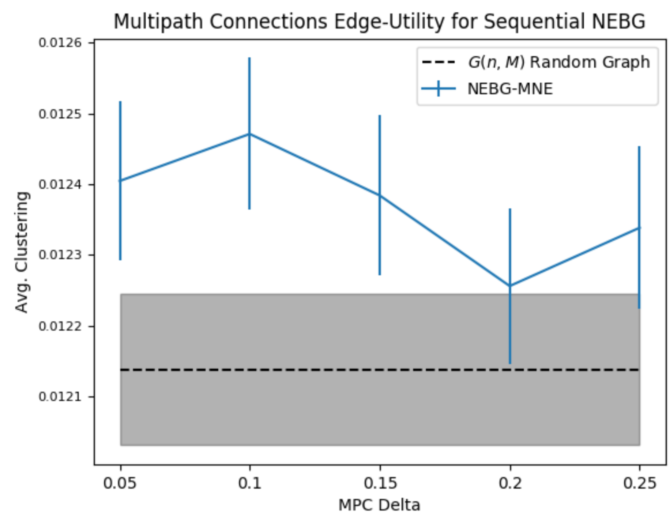


(a)

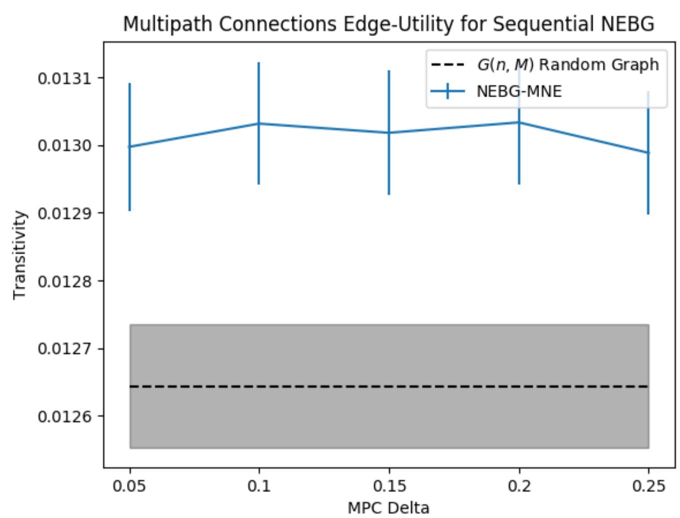


(b)

Fig. 5.4.: The sequential NEBG model with multipath connections utility function, with a $G(n, M)$ random graph of the same size ($n = 379, M = 914$) for comparison. We see slight but statistically significant difference local CC size (a) but not for graph efficiency (b), and monotonicity with the multipath connections model discount factor δ for both.



(a)



(b)

Fig. 5.5.: The sequential NEBG model with multipath connections utility function, with a $G(n, M)$ random graph of the same size $(n = 379, M = 914)$ for comparison. We see slight but statistically significant differences between the two sets of networks for average node betweenness (a) and local CC size (b), but no noticeable link between multipath connections model δ and the metric.

5.5.3 Convex Combination of Two Utility Functions

Next, we study the effect of varying a node's importance between two utility functions, by taking a convex combination between local CC utility and efficiency utility, such that when the canonical convexity hyperparameter λ is 1, the node's utility is entirely the local CC utility, and when $\lambda = 0$, it is the efficiency utility. Here, we generate networks with just 100 nodes and 400 edges, but take 5000 replications for the sake of tighter standard error bars.

In Figures 5.7 and 5.6, we see yet again a difference between the null/random model and ours, and a continuous and monotonic change of the graph metric with the convexity hyperparameter for all metrics shown.

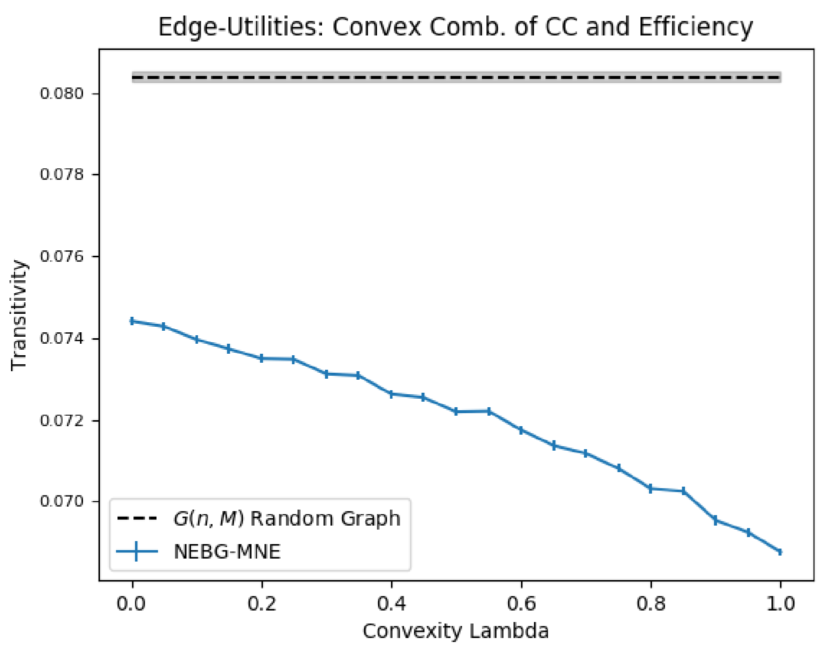
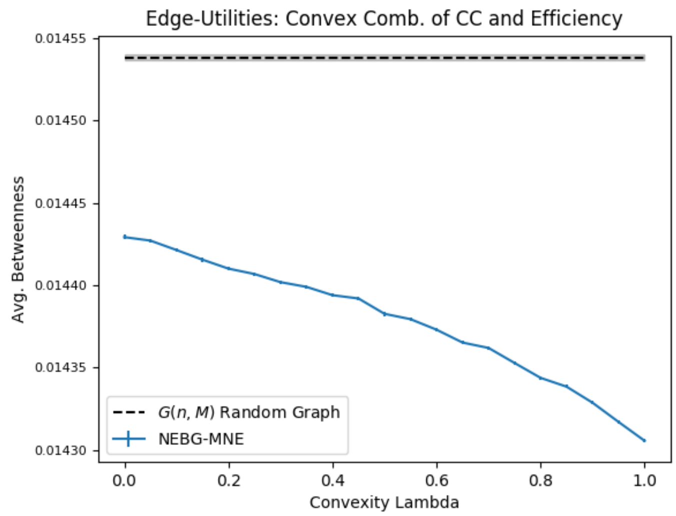
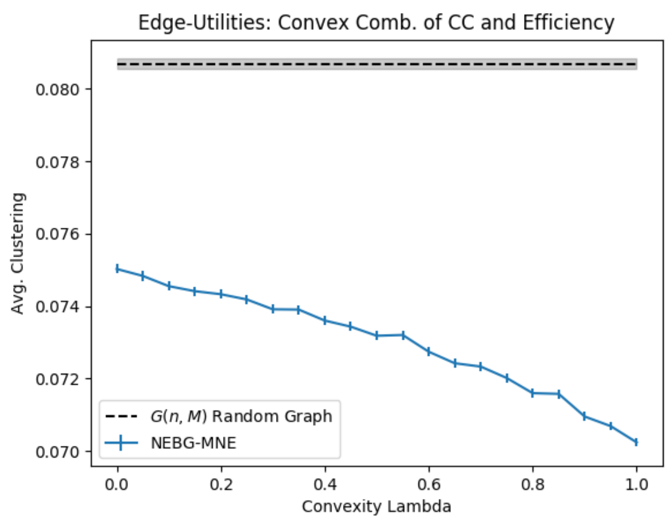


Fig. 5.6.: The sequential NEBG with a convex combination of local CC utility and efficiency, with a $G(n, M)$ random graph of the same size ($n = 100, M = 400$) for comparison. We see slight but statistically significant differences between the two sets of networks for transitivity, and monotonicity with the multipath connections model discount factor δ.



(a)



(b)

Fig. 5.7.: The sequential NEBG with a convex combination of local CC utility and efficiency, with a $G(n, M)$ random graph of the same size ($n = 100, M = 400$) for comparison. We see slight but statistically significant differences between the two sets of networks for average node betweenness (a) and local CC size (b), and monotonicity with the multipath connections model discount factor δ.

In conclusion, the sequential NEBG model is a computationally cheap simulation of playing an edge-building game on networks[2]. Almost by construction, the model is similar to a random graph, and thus we observe graph metrics of close (but statistically significantly different) values between the two. The difference between the two models is attributed to the 10 to 30 percent of the time[3] when an edge is selected for a game in the sequential model, but the edge does not form. We find some small level of control on the graph metrics by tweaking parameters of the utility functions. However, we do not find the model suitable for comparisons with real-world networks, noting large differences in metrics with the collaboration network in Table 5.1. We note that this study is a more bottom-up approach, where we study the direct variation of output networks due to variation in model parameter, without a target real-world network in mind (as opposed to models which try to match generated networks to given real-world networks).

There is vast potential for future work for the NEBG. Firstly, we hope that using heterogeneous utility functions (across nodes) will yield a richer set of generated networks. In such an experiments, we would have to decide the distribution of utility functions across nodes, and also be more judicious in how non-edges are selected for a game. For instance, perhaps nodes of a certain utility type tend to initiate edge-building more frequently than others, and so a uniform selection of edge types would be inappropriate. Furthermore, a general decrease in proposal probability (by suitably altering model parameters) might also yield more interesting results, as it would be stepping further away from the random graph model, though doing so would increase the computation load of graph generation. We also hope to derive some theoretical results on the model, noting the trade-off between complexity of

[2]5000 replications of 100 nodes and 400 edges takes about 24 hours when not run in parallel. We did not parallelize as we were running 20 such experiments at a time, and so there was no room left for parallization

[3]corresponding to single-node proposal probabilities between 45 and 70 percent

network generation and ease of theoretical analysis. For instance, it is trivial to see that when a global connected component utility function is used for all nodes (and only edges that expand a connected component are chosen as game edges), the model terminates in $\Theta(N)$ time: at least $N-1$ steps are required, and each step is at least as fast as the initial step in expectation. However, this result is lost the moment any facet of the model becomes more complicated (e.g., by using heterogeneous node utilities, or a more complicated edge-selection method).

Next, we might attempt a mechanism design approach, where the designer optimizing for social welfare could contextually be the United Nations or some other regulatory body. For instance, the designer might have a fixed budget to spend on exogenously increasing utility contributions of particular edges in order to incentivize creation of those edges (Correa et al., 2018; Khodabakhsh et al., 2019).

Finally, further future work would involve modeling the context that inspired the EBG, namely trade between countries. Serrano and Boguná (2003) was the first to model general international trade as a complex network, and found it to be a small world network that exhibits scale-free-ness[4]. Our goal would be to find the model (choice of heterogeneous utility and cost functions) and parameterizations that best match the state and/or evolution of the trade network. There is potential for a machine learning approach, where we use exogenous features of the nodes/countries such as Gross Domestic Product or population to determine appropriate node-level utility functions and parameters. Given an model that explains evolution of the trade network, we could take a (economically-speaking) positive approach to explain the general policies of each node/country, or we might instead take a normative approach to (1) state what policies each country *should* take to achieve a certain network

[4]Some notable work on this trade network includes link-prediction to model potential evolution (Vidmer et al., 2015), studying the link between its topology and countries' Gross Domestic Product (Garlaschelli and Loffredo, 2005), and between topology and position in financial crises (Kali and Reyes, 2010).

structure that is beneficial to the world as a whole, by comparing actual against socially optimal behavior, and (2) to identify where and when good or bad policies were made, by tracing network evolution backwards to key decision points.

6. THE PROBE-AND-ATTACK PROBLEM

[1]In the previous chapter we studied the EBG on networks with simultaneous and sequential games. In both cases, network structures arise due to the respective Nash equilibria, which in turn are driven by node utilities, similar to the game-/utility-driven topology changes seen in the field of strategic network formation (Jackson and Wolinsky, 1996; Fabrikant et al., 2003). In all these cases, the networks are formed as a result of actions that benefit individual nodes. Suppose instead that we are interested in directly optimizing for the well-being of the entire network as an aggregate entity; For instance, what is the 'best' possible network, given a constraint on total number of edges? Here, there problem lies in two parts: the definition of 'best' or selecting a measure of network fitness, and the method of obtaining an optimal network with regard to that definition. In this chapter, we study a problem that can be seen as one particular such evaluation of network fitness: robustness against certain type of attack. The second half of this problem, finding a network optimal in this fitness measure, is left as future work. We clarify that outside of the above stated context, the problem is otherwise entirely separate from the EBG.

We study a utility-based problem on networks, the Probe-and-Attack Problem (PAP) (Chong and Ventresca, 2017), a combinatorial problem where an attacker attempts to extract utility from a protected network. In general, attackers do not have complete information about the structure and function of a target network, and thus have to distribute resources between actions that directly achieve an objective

[1]A version of this chapter has been previously published in the proceedings of the International Conference on Complex Networks and their Applications. Springer, Cham, 2017. https://doi.org/10.1007/978-3-319-72150-7_56

to return some utility ('Attack'), and actions that expose new knowledge about the network ('Probe') allowing new actions to be taken. We show NP-hardness of PAP and provide methods to compute upper and lower bounds on the optimal solution. We then compare the bounds to the performance of two greedy algorithms on specialized instances of the problem. We note a shift in mindset: as opposed to the rest of this dissertation, in this context, having more edges is detrimental as it allows attackers more avenues of attack.

6.1 Problem Background

In many real-world networks, nodes grant protection to, or have influence over other nearby nodes. A network attacker may not be able to immediately assault every node in the network, but must first expend resources to exert influence on reachable, surface-level nodes to gain access to deeper, more protected nodes.

For instance, computer security systems have levels of firewalls protecting each layer of the system, and a hacker must first disable outer layers of security before being able to retrieve desired data from components farther in. In counter-terrorism, to gain access to the well-hidden members of terrorist cells, resources must first be spent befriending accessible members. In management and the military, is generally a strict hierarchical chain of command, where nodes have authority over their descendants.

In section 6.3, we formulate the PAP, a maximization problem on networks in which nodes grant all neighbouring nodes protection from attacks. By probing a node, an attacker disables this protection, thus allowing other nodes to be attacked. Integer Programs (IPs) are formulated for PAP and some variants. In section 6.4, we provide theoretical results, including an NP-completeness proof of the decision version of PAP, a method to compute upper and lower bounds to the optimal value, and an exact recursion for PAP on tree networks. Section 6.5 contains two heuristic algorithms for PAP, along with simulated results of the algorithms, compared against the exact solution and the upper and lower bounds to the optimal value.

6.2 Related Work

Network attacks are often studied from the perspective of disrupting the actual structure of the network. For instance, the critical node detection problem involves removing nodes to minimize pairwise connectivity in the residual graph (Addis et al., 2013; Ventresca and Aleman, 2014). Robustness of networks against such attacks (vertex removal) is also well-studied (Iyer et al., 2013; Peixoto and Bornholdt, 2012). In PAP, we maintain the network structure, with the attacker extracting utility from performing actions on individual nodes.

Defensive measures against network attack include methods to deceive the attacker (Albanese et al., 2014; Rowe and Goh, 2007), and preemptive strategies during network construction (Brown et al., 2005, 2006). Attack and defence problems are often considered together as a sequential game, where the goal is to find sub-game perfect equilibria of multi-stage network attack-and-defence games, in which the defender spends resources on network infrastructure given the attacker's strategy, and both of their payoff functions (Alderson et al., 2011; Brown et al., 2005, 2006; Dziubiński and Goyal, 2013; Scaparra and Church, 2008).

Similar in vein to the probing aspect of PAP, in the network discovery problem (Beerliova et al., 2006), querying a vertex on a network reveals a set of relevant edges (and non-edges), and the goal is to reveal the entire network using the minimum number of queries.

In the open-pit mining problem, an ore mine is modelled as a 3D grid of blocks, and an efficiently-solvable IP is formulated to determine the optimal sequence of block removal (Hochbaum and Chen, 2000; Lerchs and Grossman, 1964). The IP is almost identical in formulation to our PAP IP for tree networks, but a budget constraint in PAP results in NP-hardness.

6.3 Problem Formulation

In the PAP, an attacker wishes to assault a network, and has two potential actions to take on each node in the network - probing and attacking. Nodes are either visible or invisible to the attacker, and actions can only be taken on visible nodes. Attacking a node incurs an attack cost and returns some utility. Probing a node incurs a probe cost, yields no direct utility, but reveals all its neighbouring invisible nodes, allowing them to be probed and attacked. There is a single, shared pool of resources to spend on both types of costs, and only one node (designated the 'source' node) is initially visible.

Formally, the input to the PAP is an undirected graph $g = (V, E)$, a budget $B > 0$, and non-negative parameters $p_v, c_v, u_v, \forall v \in V$, respectively representing probe cost, attacking cost, and utility from attack for each node. We formulate two IPs, each of which might be preferable to the other depending on how tree-like a network is.

In the first IP, PAP-IP1, the number of constraints and variables are both polynomial in the size of the network:

$$\textbf{maximize} \quad \sum_{n \in V} u_n x_n \tag{6.1.1}$$

$$\textbf{subject to} \quad x_i \le \sum_{e \in \delta(S)} z_e \qquad \forall i \in S, \forall S \subseteq V \setminus 1 \tag{6.1.2}$$

$$y_i \le \sum_{e \in \delta(S)} z_e, \qquad \forall i \in S, \forall S \subseteq V \setminus 1 \tag{6.1.3}$$

$$z_e \le x_{e_1} + x_{e_2} \qquad \forall e \in E \tag{6.1.4}$$

$$\sum_{n \in V} c_n x_n + p_n y_n \le B \tag{6.1.5}$$

$$y_1 = 1 \tag{6.1.6}$$

$$x_n, y_n \in \{0, 1\} \qquad \forall n \in V \tag{6.1.7}$$

$$z_e \in \{0, 1\} \qquad \forall e \in E \tag{6.1.8}$$

Without loss of generality let the initially visible ‘source’ node be node 1. We view the actions as taken over discrete time steps, with at most one action taken per step. There can be at most $|V|$ probe actions and $|V|$ attack actions, so no more than $2|V|$ time steps are required in the IP formulation. The decision variables are as follows: $x_{t,v}$ ($y_{t,v}$) is set to 1 if node v is attacked (probed) at time step t and 0 if it is not - there is one variable per node per time step for each of the two actions. Let $N(v)$ be the set of all neighbours of node v.

Constraints (6.1.2) (and (6.1.3)) ensure that each node is attacked (probed) at most once. Constraints (6.1.4) ensure that at most one action is taken at each time step. Constraints (6.1.5) ensure that if an action is taken in time step t, there must have been an action taken in time step $t-1$. Without constraints (6.1.5), solutions to the IP would still correspond to optimal sequences of actions, but there could be many symmetric solutions due to gaps/time steps in which no actions occur. Constraints (6.1.6) (and (6.1.7)) ensure that an action is valid: if node v is attacked (probed), then $x_{t,v}$ ($y_{t,v}$) is set to 1, and so at least one variable in the set $\{y_{s,w} : 1 \leq s \leq t-1, w \in N(v)\}$ is set to one, implying that at least one of the neighbours of node v has been probed during a previous time step. Constraint (6.1.8) is simply the budget constraint.

There are exactly $4|V|^2$ variables and $4|V|^2 + 4|V|$ constraints, and so the linear relaxation to PAP-IP1 can be found in polynomial time using methods such as the ellipsoid method (Khachiyan, 1980), opening up avenues for relaxation-based approximation algorithms.

A second formulation uses the fact that a node v is only visible if every node on at least one path between the source and node v has been probed. For each v, let there there be k_v unique, simple paths between the source and node v, and let the set of nodes on each of these paths be $P_{v,h}, h = 1, \ldots, k_v$. For ease of formulation, we exclude node v itself from all path-sets $P_{v,h}$. The second IP, PAP-IP2 is then:

$$\textbf{maximize} \quad \sum_{v \in V} u_v x_v \tag{6.2.1}$$

$$\textbf{subject to} \quad x_v \leq \sum_{h=1,\ldots,k_v} z_{v,h}, \qquad \forall v \in V, v \neq 1 \tag{6.2.2}$$

$$y_v \leq \sum_{h=1,\ldots,k_v} z_{v,h}, \qquad \forall v \in V, v \neq 1 \tag{6.2.3}$$

$$|P_{v,h}| z_{v,h} \leq \sum_{w \in P_{v,h}} y_w, \forall v \in V, h = 1, \ldots, k_v \tag{6.2.4}$$

$$\sum_{v \in V} c_v x_v + p_v y_v \leq B \tag{6.2.5}$$

$$x_v, y_v, z_{v,h} \in \{0, 1\} \quad \forall v \in V, h = 1, \ldots, k_v \tag{6.2.6}$$

The decision variables are as follows: x_v (y_v) is set to 1 if node v is attacked (probed) and 0 if it is not, and $z_{v,h}$ is set to 1 if all nodes along the path defined by set $P_{v,h}$ are probed, and 0 if at least one node in set $P_{v,h}$ is not probed.

Constraints (6.2.2) ensure that an attack on a node is valid: if node v is attacked, x_v is set to 1, then at least one of variable in $\{z_{v,h}, h = 1, \ldots, k_v\}$ must be set to 1, and so at least one path from to the source consists entirely of probed nodes. Similarly, the constraints (6.2.3) ensure that nodes can only be probed if there is a fully probed path to the source. Constraints (6.2.4) enforces the validity of the $z_{v,h}$ variables. The summation contains the $|P_{v,h}|$ probe-variables corresponding to path $P_{v,h}$, and so $z_{v,h}$ can be set to 1 if and only if all probe-variables in the summation are set to 1.

The number of variables and constraints in PAP-IP2 is linear in the size of the network and the number of paths between nodes. Unfortunately, there is generally an exponential number of paths between any two nodes for arbitrary networks, and so even populating the IP can be expensive. The fewer cycles there are, the fewer paths there are between two nodes, and so PAP-IP2 can be preferable to PAP-IP1 for more tree-like networks.

6.3.1 PAP on trees

A case of interest is when the PAP network G is a tree, as systems in the real world are often hierarchical in nature, with sequential levels of security and/or authority between components with zero or a low number of cycles (e.g., military chain-of-command, hierarchical organizations in management).

In this scenario, there exists only one path between any two nodes, and so the number of variables and constraints is linear in the size of the network. Without loss of generality, label the source node as node s, and let the tree be rooted at the source node. If we designate the parent of node v as node $r(v)$, The IP then simplifies to:

$$\begin{aligned}
\textbf{maximize} \quad & \sum_{v \in V} u_v x_v \\
\textbf{subject to} \quad & x_v \le y_{r(v)}, && \forall v \in V, v \neq s \\
& y_v \le y_{r(v)}, && \forall v \in V, v \neq s \\
& \sum_{v \in V} c_v x_v + p_v y_v \le B \\
& x_v, y_v \in \{0, 1\} && \forall v \in V
\end{aligned} \tag{6.3}$$

This smaller IP has exactly $2|V|$ variables and $2|V| + 1$ constraints, and is thus much more tractable than the IPs for PAP on general networks.

6.3.2 Generalizing the Probe-Attack Problem: The Sequential Attack Problem

The network in an instance of PAP can be seen as being implicitly defined by the correspondences between probing nodes, and the sets of nodes subsequently made available for attack. If there were alternative probing actions that influence differing sets of nodes, we would then require a multilayer network. A practical example would be the spreading of influence over political systems that utilize checks and balances between branches of governance. By design, it is difficult to directly influence one branch from another, but individual members of a branch may have close ties with members of other branches, with different actions on an individual influencing different sets of other individuals. We generalize PAP to an arbitrary number of actions and formulate the optimization problem.

Let the Sequential Attack Problem (SAP) be described as follows: An attacker wants to maximize utility by assaulting a network, and has a number of potential actions to take on each node. Each node-action pairing incurs a cost, returns some utility, and, once made, allows specific other node-action pairs to be taken. Whether or not an action can be taken on a node is thus determined by all node-action pairs taken up to that point.

Formally, let the set of nodes and actions be V and D respectively. Let the total budget be $B > 0$, with the set of all possible node-action pairs designated as $S \subseteq V \times D$. For ease of notation, let the node-action pairs of S be enumerated as $m = 1, \ldots, |S|$. Let the costs and utilities for taking each node-action pair be $c_m \geq 0$ and $u_m \geq 0$ respectively, $m \in S$. Let the set of node-action pairs allowed by node-pair m be $R_m \subseteq S$, with an initial allowed set of node-actions pairs defined as R_{s_0}. The optimization problem is then:

$$\begin{aligned}
\underset{\mathbf{s} \subseteq S}{\textbf{maximize}} \quad & \sum_{s_i \in \mathbf{s}} u_{s_i} \\
\textbf{subject to} \quad & s_i \in \bigcup_{j=0}^{i-1} R_{s_j} \; \forall i = 1, \ldots, |\mathbf{s}| \\
& \sum_{s_i \in \mathbf{s}} c_{s_i} \leq B
\end{aligned} \tag{6.4}$$

where $\mathbf{s} \subseteq S$ is the chosen set of node-action pairs, and s_i represents the i^{th} (chronologically) chosen node-action pair in $\mathbf{s}$. An IP similar to PAP-IP1 can be set up for SAP.

6.4 Theoretical Results

First, we show NP-completeness of PAP (and thus SAP). The decision problem version of PAP (SAP), D-PAP (D-SAP) can be stated as: Given an instance of PAP (SAP), can we achieve utility of at least Y given budget B?

Let the 0/1-knapsack problem (KNAP) be defined as follows: Given N items of weights and values w_n and v_n respectively $\forall n = 1, \ldots, N$, find a bundle of maximal value with total weight not exceeding W. The decision version (D-KNAP) is then: Is there a bundle of items of total weight not more than W such that the total value of items selected is greater than U?

Theorem 6.4.1 *D-PAP and D-SAP are NP-complete.*

Proof D-PAP and D-SAP are in NP, as it is trivial to check the validity and cost of any solution efficiently. Next, we reduce D-PAP from D-KNAP. Given an instance $K \in$ D-KNAP we construct a D-PAP instance as follows: Let $V = \{1, \ldots, N+1\}, E =$

$\{(N+1,n), n=1,\ldots,N\}, B=W+1, Y=U, u_{N+1}=1, c_{N+1}=B+1, p_{N+1}=1$, and let $u_n=v_n, c_n=w_n, \forall n=1,\ldots,N$. Call this instance $P(K)$ of D-PAP.

That is, construct a star network with one leaf node for each item from K, with a dummy source node designated as node $N+1$ as the centre of the star. To solve $P(K)$, $p_{N+1}=1$ unit of resource must be spent on probing the source, which allows all other nodes to be attacked with a remaining budget of $B-1=W$. Clearly, $P(K)$ returns YES if and only if K returns YES and thus KNAP$\leq_P$D-PAP. ■

Next we provide a method to compute upper and lower bounds on the optimal solution to PAP. Given an instance P of PAP with N nodes and source node 1, we define an instance $K_1(P)$ of KNAP as follows: Let $W=B-p_1, v_n=u_n, w_n=c_n, \forall n=1,\ldots,N$. We also define instance $K_2(P)$ of KNAP: Let $W=B-p_1$. For $n=1,\ldots,N$, let $v_n=u_n, w_n=c_n+l_n$, where l_n is the minimum total probing cost over all paths from node n to the root, excluding the probe cost of node n itself.

Theorem 6.4.2 *If the optimal value of instance I of an optimization problem is $OPT(I)$, then $OPT(K_2(P)) \leq OPT(P) \leq OPT(K_1(P))$.*

Proof Suppose $OPT(P) > OPT(K_1(P))$. The cost to attack all nodes in the optimal solution of instance P is at most $B-p_1$ (p_1 to probe the source). The corresponding bundle in $K_1(P)$ has total weight at most $B-p_1=W$, but has total value greater than $OPT(K_1(P))$, a contradiction. The other inequality is shown through a similar argument. ■

Since KNAP admits numerous pseudo-polynomial time algorithms, along with a fully polynomial-time approximation scheme (Kellerer and Pferschy, 1999; Magazine and Oguz, 1981), the bounds in Theorem 6.4.2 can thus be computed with relative ease.

For PAP on trees, we present a recursion that returns an optimal solution. Let k_v be the number of children of node v, with the children enumerated as $d_{v,1}, \dots, d_{v,k_v}$, and let $k = \max_{v \in V} k_v$. Let $m(T, b)$ be the maximum value attainable on tree T with budget b, and let T_v be the subtree corresponding to node v. The recursion is then:

$$m(T_v, b) = \max\Big(m_1(T_v, b), m_2(T_v, b), m_3(T_v, b)\Big) \tag{6.5}$$

$$m_1(T_v, b) = \begin{cases} u_v, & \text{if } b \geq c_v \\ 0, & \text{otherwise} \end{cases} \tag{6.6}$$

$$m_2(T_v, b) = \begin{cases} \max\limits_{\substack{\sum_{j=1}^{k_v} b_j = b - p_v \\ b_j \in \mathbb{Z}_{\geq 0}}} \left(\sum_{j=1}^{k_v} m(T_{d_{v,j}}, b_j)\right), & \text{if } b > p_v \\ 0, & \text{otherwise} \end{cases} \tag{6.7}$$

$$m_3(T_v, b) = \begin{cases} u_v + \max\limits_{\substack{\sum_{j=1}^{k_v} b_j = b - p_v - c_v \\ b_j \in \mathbb{Z}_{\geq 0}}} \left(\sum_{j=1}^{k_v} m(T_{d_{v,j}}, b_j)\right), & \text{if } b > p_v + c_v \\ 0, & \text{otherwise} \end{cases} \tag{6.8}$$

$m_1(T_v, b)$ corresponds to not probing node v, and hence utility comes from attacking node v itself, if the budget allows it. $m_2(T_v, b)$ is the maximum value attained when probing, but not attacking, node v, in which case we recurse over the sum of maximum attainable values on the subtrees corresponding to children of node v. Note that this involves partitioning $b - p_v \approx b$ into k_v sub-problems. There are $\binom{b+k_v-1}{k_v-1} \approx O(b^{k_v-1})$ ways to do so, for $n \gg k$. Similarly, $m_3(T_v, b)$ is the maximum value attained when both probing and attacking node v. Note that if node v is a leaf node, then $m_2(T_v, b) = m_3(T_v, b) = 0$, and $m(T_v, b) = m_1(T_v, b)$, as expected. The optimal value of a PAP instance is then $m(T_1, B)$, and the optimal solution can be found using memoization.

The sums in the max functions in $m_2(\cdot, \cdot)$ and $m_3(\cdot, \cdot)$ have at most k terms, the max is taken over $O(B^{k-1})$ partitions of B, and there are $|V|B$ total $m(\cdot, \cdot)$ terms to populate. The whole recursion thus has runtime $O(k|V|B^k)$, and so the recursion runs in pseudo-polynomial time for fixed k.

6.5 Heuristic algorithms for PAP

In reality, node costs and utilities are likely to depend on network properties such as node degree and connectivity. In this dissertation we only study the simpler case in which all costs and utilities are independent of each other, and of network structure. We explore two separate scenarios. In the "limited knowledge" scenario, the attacker knows neither the number of nodes nor the edge set, and can only see costs and utilities of nodes as they become visible (i.e, as neighbours of nodes are probed). In the "full knowledge" scenario, the attacker knows everything about the network at the beginning of the problem - the number of nodes, the edge set, and all costs and utilities - but can only attack or probe visible nodes.

We present greedy algorithms for each scenario, and compare them against the optimal solution for PAP on trees, and against the bounds from Theorem 6.4.2 for PAP on general networks.

6.5.1 Limited Knowledge PAP

In this scenario, for any algorithm, one can design pathological instances such that the algorithm will perform arbitrarily badly, since there is always the possibility that there is a high-valued node that the algorithm never encounters.

Suppose an attacker has decided to attack a visible node v. Instead of immediately attacking, the attacker could reserve the appropriate amount of budget (in this case c_v) and use the remaining budget to probe other nodes, in the hopes of discovering a better node than v to attack. The following algorithm "LK-Greedy" operates under this principle, by probing visible, unprobed nodes with the lowest probe cost until an additional probe would render the remaining budget insufficient to attack all visible nodes, and then attacking all visible nodes in decreasing order of utility-per-cost. LK-Greedy runs in $O(|V|^2)$ time.

Algorithm 2 Greedy algorithm for limited knowledge PAP

procedure LK-Greedy$(g = (V, E), \mathbf{u}, \mathbf{c}, \mathbf{p}, B)$
    $S_P \leftarrow \{\phi\}, \quad S_A \leftarrow \{\phi\}, \quad S_E \leftarrow \{1\}$ ▷ **P**robed/**A**ttacked/Visibl**e** nodes
    $R_B \leftarrow B$ ▷ Remaining budget
    while $R_B > \sum_{v \in S_E} c_v$ **do** ▷ Probing stage: Probe until attack costs unaffordable
        $v_{\text{pr}} \leftarrow \text{argmin}_{v \in S_E \cap S_P^c} p_v$
        if $R_B \geq p_{v_{\text{pr}}} + \sum_{v \in S_E} c_v$ **then**
            $S_P \leftarrow S_P \cup \{v_{\text{pr}}\}$
            $S_E \leftarrow S_E \cup N(v_{\text{pr}})$ ▷ Neighbours of probed node become visible
            $R_B \leftarrow R_B - p_{v_{\text{pr}}}$
        else
            break
    while $R_B \geq \min_{v \in S_E \cap S_A^c} c_v$ and $|S_E \cap S_A^c| > 0$ **do**
        $v_{\text{at}} \leftarrow \text{argmax}_{\{v : c_v \leq R_B, v \in S_E \cap S_A^c\}} \frac{u_v}{c_v}$ ▷ Attack visible nodes in order of decreasing u/c
        $S_A \leftarrow S_A \cup \{v_{\text{at}}\}$
        $R_B \leftarrow R_B - c_{v_{\text{at}}}$
    return S_P, S_A

Instead of attacking in decreasing order of utility-per-cost, one could solve a knapsack of all visible nodes and budget R_B at the end of the probing stage. Runtime would no longer be polynomial, but at the beginning of the attacking stage, S_V is likely to be much smaller than V (as with R_B and B), resulting in a relatively small knapsack.

It should be noted that dependence between parameters and network structure is likely to occur in reality. For instance, nodes with higher utility might cost more to attack, and nodes allowing access to many other nodes would likely be better protected, so probing cost might correlate with node degree. Leveraging this information would aid an attacker in this scenario.

6.5.2 Full Knowledge PAP

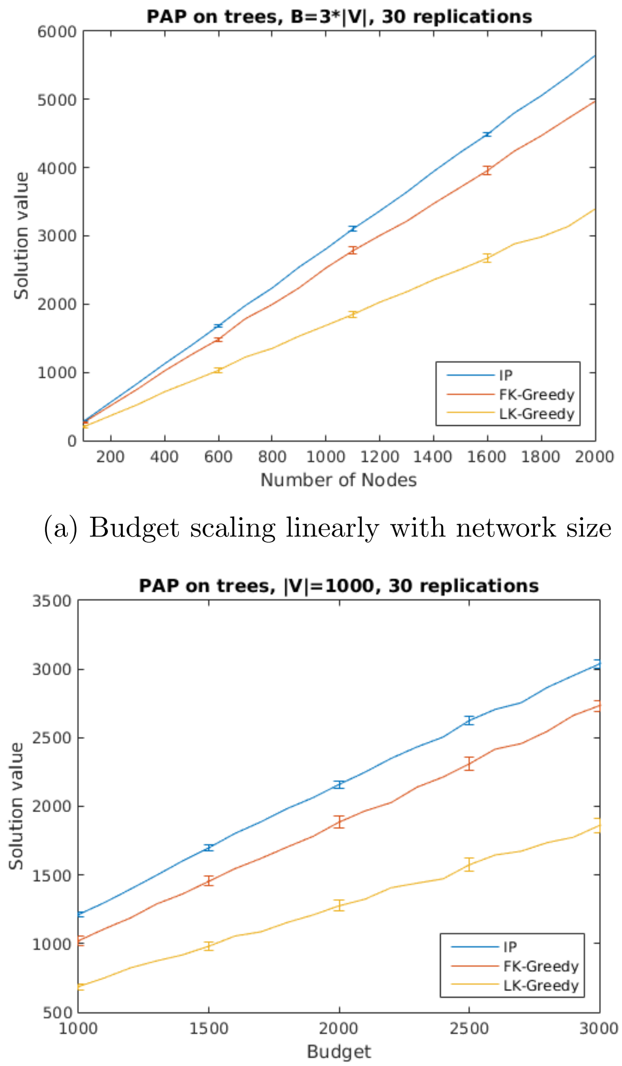
Next, we describe a greedy algorithm "FK-Greedy" in the scenario where the attacker knows all costs and utilities, along with the network structure. Similar in vein to the canonical greedy algorithm on KNAP, we sort nodes in descending order of utility over "cost". In this case, we define the "cost" of attacking node v to be the sum of the direct attack cost c_v, and the cost of probing all unprobed nodes along the cheapest (in terms of probe cost) path from the source to node v, which can be found in polynomial time using standard shortest path algorithms on a weighted directed network $g' = (V, E')$ as specified in Algorithm 3. Roughly, g' has a directed edge in each direction between two nodes if there is an edge between those nodes in g, with edge weights equal to the probe cost of the node an edge is pointing to. After a node is probed, edge-weights of all edges into that node are set to zero. This ensures that the correct cheapest paths are returned, though an explanation is omitted for brevity. FK-Greedy using Dijkstra's algorithm runs in $O(|V|^2(|E| + |V| \ln |V|))$ time.

6.5.3 Simulation Results

We ran LK-Greedy and FK-Greedy on PAP instances of various sizes and budgets, each on the order of 10^3, with all costs and utilities drawn from uniformly from the integers 1 to 10. First, we tested the algorithms on trees with average branching factor $k = 2$. The PAP IP for trees is tractable for networks of the chosen sizes, and so we are able to compute the exact optimal values. Next, we generated random networks drawn uniformly from the set of all connected networks (Propp and Wilson, 1998) (for fixed numbers of nodes and edges). The IP quickly becomes intractable, and so we compare the greedy algorithms to the knapsack bounds from Theorem 6.4.2.

Algorithm 3 Greedy algorithm for full knowledge PAP

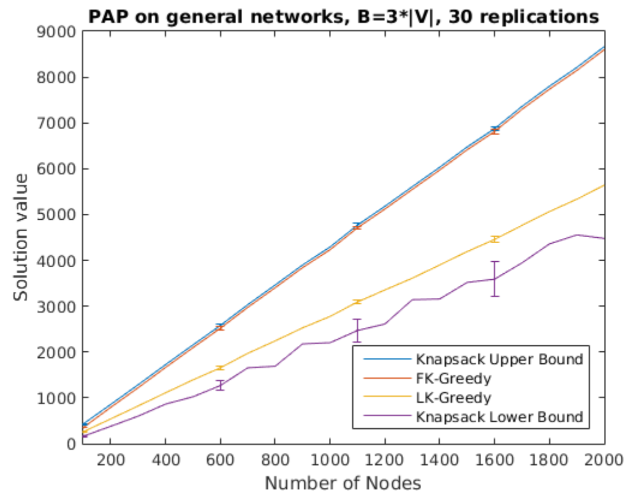
procedure FK-Greedy$(g = (V, E), \mathbf{u}, \mathbf{c}, \mathbf{p}, B)$
  $S_P \leftarrow \{1\}, \quad S_A \leftarrow \{\phi\}$ ▷ **P**robed/**A**ttacked nodes
  $R_B \leftarrow B - p_1$ ▷ Remaining budget (source node is probed)
  $E' \leftarrow \{\phi\}$ ▷ initialize edge weights to determine probe costs on paths
  for $(i, j) \in E$ **do** ▷ Directed edges in both directions for each undirected edge in E
    $E' \leftarrow E' \cup \{(i, j) \cup (j, i)\}$
    $w_{E'}(i, j) = p_j * 1\{j \in S_P^c\}$
    $w_{E'}(j, i) = p_i * 1\{i \in S_P^c\}$
  while $R_B > 0$ **do**
    $S_C \leftarrow \{\phi\}$ ▷ Set of affordable nodes
    for $v \in S_A^c$ **do**
      $E'_v \leftarrow E', \mathbf{w}_{E'_v} \leftarrow \mathbf{w}_{E'}$
      for $(i, v) \in E'_v$ **do** $w_{E'_v}(i, v) \leftarrow 0$
      $P_v \leftarrow$ **cheapestPath**$(g = (V, E'_v), \mathbf{w}_{E'_v}, v)$ ▷ run Dijkstra's, etc.
      $C_v \leftarrow c_v + \sum_{w \in P_v \cup S_P^c} p_w$
      if $R_B > C_v$ **then** $S_C \leftarrow S_C \cup \{v\}$
    if $|S_C| == 0$ **then Break**
    $v_{\text{at}} \leftarrow \text{argmax}_{v \in S_C} \frac{u_v}{C_v}$
    $S_A \leftarrow S_A \cup \{v_{\text{at}}\}, \quad S_P \leftarrow S_P \cup P_{v_{\text{at}}}, \quad R_B \leftarrow R_B - C_{v_{\text{at}}}$
    for $(i, v) \in E' : v \in P_{v_{\text{at}}}$ **do** $w_{E'}(i, v) \leftarrow 0$
  return S_P, S_A



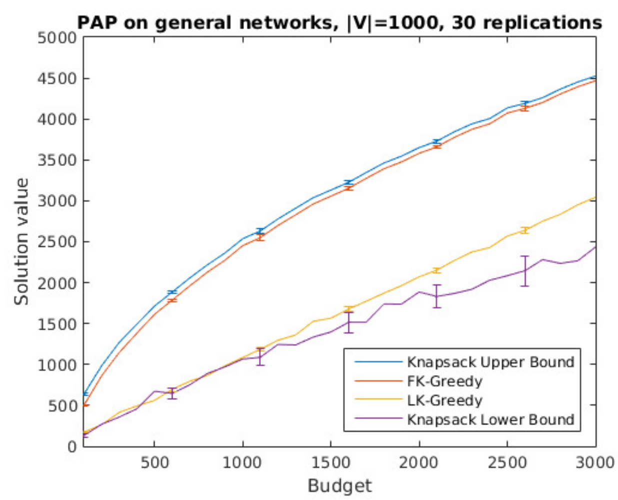
(a) Budget scaling linearly with network size

(b) Fixed network size ($|V| = 1000$)

Fig. 6.1.: LK-Greedy, FK-Greedy, and the exact IP solution for PAP, with results averaged over 30 randomly generated trees of mean branching factor $k = 2$.



(a) Budget scaling linearly with network size



(b) Fixed network size ($|V| = 1000$)

Fig. 6.2.: LK-Greedy, FK-Greedy, and optimal value bounds from Theorem 6.4.2, with results averaged over 30 randomly generated networks uniformly drawn from the space of all connected networks (for a particular value of $|V|$) of edge density 0.05.

We see that FK-Greedy performs well on random networks with the chosen parameters, returning solution values close to the Theorem 6.4.2 upper bounds (and thus the algorithm solution and the upper bound are both close to the optimal solution), and that LK-Greedy performs at least better than the Theorem 6.4.2 lower bounds. Note that in all plots, only a subset of error bars are displayed, for visual clarity.

6.6 Defense Problem

Knowing that an attacker is to attack a network based on PAP, how should a defender best construct a network, given the probe costs, attack costs, and utilities of each node? We say that a network is more PAP robust when an optimal attacker gains less utility from it.

Lemma 6.6.1 *There exists an optimally PAP robust network that is a line.*

Proof Observe that any network can be made more PAP robust by moving nodes farther (in terms of probe cost) from the source node. The proof immediately follows. ■

We perform simulated annealing on the permutations of nodes in a line to find PAP robust networks, to some success. We acknowledge that a line network is not a practical network for almost anything, but we can use the robustness of an optimal line as an upper bound to that of an arbitrary graph. We hope to properly develop the defense problem in future work on the PAP.

7. SUMMARY AND FUTURE WORK

In this thesis, we explored extensions to the Edge-Building Game, a public goods game where, by way of studying Nash and correlated equilibria, we are interested in how players try to avoid paying the cost of the good. We study continuous, multiplayer, and network extensions, and also generate networks based on edges actually being formed through the game.

In the continuous version of the game, we proposed three different models of edge-cost payment and found interesting functional forms for MNEs that play all strategies with positive probability[1]. In terms of expected utility, these MNEs are trivially equivalent to simply proposing with full effort every time. However, they are still MNEs, in that both players can play the MNE with no incentive to deviate, whereas proposing with full effort leads to deviating to proposing with zero effort We found that expected utility is maximized (while maintaining fairness) when a correlated equilibrium is imposed. In the multiplayer game with N players, we found the symmetric MNE proposal probability to be the root of a degree-$(N-1)$ polynomial, and observed that the benefit from potentially freeloading off other players is outweighed by the loss from an increased chance in the edge not being formed.

Further work in these areas include properly finding a closed-form solution for the third edge-cost payment model in the continuous case, and more ambitiously, characterizing exactly when a game can lead to discontinuous MNE probabilities over the whole strategy space. In the multiplayer game, even though there are no closed-form solutions for general higher-degree polynomials, we hope that ours is a

[1]more accurately, the PDFs are positive for *almost all* values of proposal effort

special case for which we might find analytic roots, in order to derive the asymptotic (with respect to number of players) expected utility as a function of edge cost.

We also studied the Edge-Building Game on networks. In simultaneous games, we showed special cases where we can efficiently find equilibria. We hope to expand the scope of cases wherein this is possible, for instance, with other utility functions and with more relaxed assumptions on the underlying skeleton network. In the sequential network game, we ran the EBG edge-by-edge, forming the edge if at least one of the two players proposed, where proposal probabilities were given by the MNEs of the game. We observed that this iteration of the sequential model is not a good generator for real-world networks. Network metrics of the sequential model were found to be close in value to that of the $G(n, M)$ random graph, but still different with statistical significance.

This closeness is because the sequential model can be seen as a generalization of the random graph, and for future work we will run more experiments with much lower proposal probabilities. There is a large space to explore here, as there are many different modeling choices and hyperparameters to tune that may lead to qualitatively different results. In particular, we are interested in modeling heterogeneous node utilities (as seen in Arora and Ventresca (2017)). We are also interested in incentivizing edges from a mechanism design perspective, and also modeling real-world trade networks in the hopes of making both positive and normative economic statements about countries' current and potential policies.

We also introduced the multipath connections metric, an extension to the Connections Model in Jackson and Wolinsky (1996) such that all paths between nodes matter, not just the shortest ones. We showed submodularity of the metric under certain decision setups. The multipath model is necessarily more complicated, but we hope to derive strategic network results as was done for the Connections Model.

Finally, within the context of optimizing a network toward a particular measure of fitness, we introduced as a fitness measure the combinatorial Probe-and-Attack Problem, where an attacker attempts to extract utility from guarded nodes. We showed NP-hardness of the problem, along with somewhat computable upper- and lower-bounds, along with an efficient heuristic. Future work here includes testing the problem on real-life networks, wherein the difficulty involves assigning contextually appropriate costs and utility parameters to the nodes. Additionally, we are interested in the defensive problem of constructing a network given some budget and edge-building cost to be optimally robust against a PAP attacker.

BIBLIOGRAPHY

Addis, B., Di Summa, M., and Grosso, A. (2013). Identifying critical nodes in undirected graphs: Complexity results and polynomial algorithms for the case of bounded treewidth. *Discrete Applied Mathematics*, 161(16):2349–2360.

Albanese, M., Battista, E., Jajodia, S., and Casola, V. (2014). Manipulating the attacker's view of a system's attack surface. In *Communications and Network Security (CNS), 2014 IEEE Conference on*, pages 472–480. IEEE.

Albert, R. and Barabási, A.-L. (2002). Statistical mechanics of complex networks. *Reviews of modern physics*, 74(1):47.

Alderson, D. L., Brown, G. G., Carlyle, W. M., and Wood, R. K. (2011). Solving defender-attacker-defender models for infrastructure defense. Technical report, Naval Postgraduate School Monterey CA Dept. of Operations Research.

Arora, V., Guo, D., Dunbar, K. D., and Ventresca, M. (2020). Examining the variability in network populations and its role in generative models. *Network Science*, 8(S1):S43–S64.

Arora, V. and Ventresca, M. (2017). Action-based modeling of complex networks. *Scientific reports*, 7(1):1–10.

Azar, O. H. and Bar-Eli, M. (2011). Do soccer players play the mixed-strategy nash equilibrium? *Applied Economics*, 43(25):3591–3601.

Barabási, A.-L. and Albert, R. (1999). Emergence of scaling in random networks. *science*, 286(5439):509–512.

Basu, S. and Savani, K. (2017). Choosing one at a time? presenting options simultaneously helps people make more optimal decisions than presenting options sequentially. *Organizational Behavior and Human Decision Processes*, 139:76–91.

Beerliova, Z., Eberhard, F., Erlebach, T., Hall, A., Hoffmann, M., Mihal'ak, M., and Ram, L. S. (2006). Network discovery and verification. *IEEE Journal on selected areas in communications*, 24(12):2168–2181.

Bohnet, I., Van Geen, A., and Bazerman, M. (2016). When performance trumps gender bias: Joint vs. separate evaluation. *Management Science*, 62(5):1225–1234.

Bonacich, P. (1987). Power and centrality: A family of measures. *American journal of sociology*, 92(5):1170–1182.

Brown, G., Carlyle, M., Salmerón, J., and Wood, K. (2006). Defending critical infrastructure. *Interfaces*, 36(6):530–544.

Brown, G. G., Carlyle, W. M., Salmeron, J., and Wood, K. (2005). Analyzing the vulnerability of critical infrastructure to attack and planning defenses. In *Emerging Theory, Methods, and Applications*, pages 102–123. INFORMS.

Brown, W. and Patange, U. (2020). Targeted intervention in random graphs. In *International Symposium on Algorithmic Game Theory*, pages 211–225. Springer.

Camerer, C. F. (1991). Does strategy research need game theory? *Strategic Management Journal*, 12(S2):137–152.

Camerer, C. F. and Fehr, E. (2004). Measuring social norms and preferences using experimental games: A guide for social scientists. *Foundations of human sociality: Economic experiments and ethnographic evidence from fifteen small-scale societies*, 97:55–95.

Carrillo, J. D. and Palfrey, T. R. (2009). The compromise game: two-sided adverse selection in the laboratory. *American Economic Journal: Microeconomics*, 1(1):151–81.

Chong, B. H. and Ventresca, M. (2017). Attacking unexplored networks-the probe-and-attack problem. In *International Conference on Complex Networks and their Applications*, pages 692–703. Springer.

Correa, J., Guzmán, C., Lianeas, T., Nikolova, E., and Schröder, M. (2018). Network pricing: How to induce optimal flows under strategic link operators. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, pages 375–392.

Dasgupta, P. and Maskin, E. (1986). The existence of equilibrium in discontinuous economic games, i: Theory. *The Review of economic studies*, 53(1):1–26.

Daskalakis, C., Frongillo, R., Papadimitriou, C. H., Pierrakos, G., and Valiant, G. (2010). On learning algorithms for nash equilibria. In *International Symposium on Algorithmic Game Theory*, pages 114–125. Springer.

Daskalakis, C., Goldberg, P. W., and Papadimitriou, C. H. (2009). The complexity of computing a nash equilibrium. *SIAM Journal on Computing*, 39(1):195–259.

Dawes, R. M. (1980). Social dilemmas. *Annual review of psychology*, 31(1):169–193.

Dziubiński, M. and Goyal, S. (2013). Network design and defence. *Games and Economic Behavior*, 79:30–43.

Erdős, P. and Rényi, A. (1960). On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci*, 5(1):17–60.

Fabrikant, A., Luthra, A., Maneva, E., Papadimitriou, C. H., and Shenker, S. (2003). On a network creation game. In *Proceedings of the twenty-second annual symposium on Principles of distributed computing*, pages 347–351.

Galeotti, A., Golub, B., and Goyal, S. (2019). Targeting interventions in networks. *Available at SSRN 3054353*.

Garlaschelli, D. and Loffredo, M. I. (2005). Structure and evolution of the world trade network. *Physica A: Statistical Mechanics and its Applications*, 355(1):138–144.

Gemünden, H. G. and Hauschildt, J. (1985). Number of alternatives and efficiency in different types of top-management decisions. *European Journal of Operational Research*, 22(2):178–190.

Glicksberg, I. L. (1952). A further generalization of the kakutani fixed point theorem, with application to nash equilibrium points. *Proceedings of the American Mathematical Society*, 3(1):170–174.

Gottlob, G., Greco, G., and Scarcello, F. (2005). Pure nash equilibria: Hard and easy games. *Journal of Artificial Intelligence Research*, 24:357–406.

Hochbaum, D. S. and Chen, A. (2000). Performance analysis and best implementations of old and new algorithms for the open-pit mining problem. *Operations Research*, 48(6):894–914.

Iyer, S., Killingback, T., Sundaram, B., and Wang, Z. (2013). Attack robustness and centrality of complex networks. *PloS one*, 8(4):e59613.

Jackson, M. O. (2005). A survey of network formation models: stability and efficiency. *Group formation in economics: Networks, clubs, and coalitions*, 664:11–49.

Jackson, M. O. and Wolinsky, A. (1996). A strategic model of social and economic networks. *Journal of economic theory*, 71(1):44–74.

Kakade, S., Kearns, M., Langford, J., and Ortiz, L. (2003). Correlated equilibria in graphical games. In *Proceedings of the 4th ACM Conference on Electronic Commerce*, pages 42–47.

Kakutani, S. et al. (1941). A generalization of brouwer's fixed point theorem. *Duke mathematical journal*, 8(3):457–459.

Kali, R. and Reyes, J. (2010). Financial contagion on the international trade network. *Economic Inquiry*, 48(4):1072–1101.

Kamra, N., Gupta, U., Wang, K., Fang, F., Liu, Y., and Tambe, M. (2019). Deepfp for finding nash equilibrium in continuous action spaces. In *International Conference on Decision and Game Theory for Security*, pages 238–258. Springer.

Kearns, M., Littman, M. L., and Singh, S. (2013). Graphical models for game theory. *arXiv preprint arXiv:1301.2281*.

Kellerer, H. and Pferschy, U. (1999). A new fully polynomial time approximation scheme for the knapsack problem. *Journal of Combinatorial Optimization*, 3(1):59–71.

Khachiyan, L. G. (1980). Polynomial algorithms in linear programming. *USSR Computational Mathematics and Mathematical Physics*, 20(1):53–72.

Khodabakhsh, A., Horn, J., Nikolova, E., and Pountourakis, E. (2019). Prosumer pricing, incentives and fairness. In *Proceedings of the Tenth ACM International Conference on Future Energy Systems*, pages 116–120.

Killingback, T. and Doebeli, M. (1996). Spatial evolutionary game theory: Hawks and doves revisited. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 263(1374):1135–1144.

Kleinberg, R. D., Ligett, K., Piliouras, G., and Tardos, É. (2011). Beyond the nash equilibrium barrier. In *ICS*, pages 125–140.

Kroer, C. and Sandholm, T. (2015). Discretization of continuous action spaces in extensive-form games. In *AAMAS*, pages 47–56.

Kümmerli, R., Colliard, C., Fiechter, N., Petitpierre, B., Russier, F., and Keller, L. (2007). Human cooperation in social dilemmas: comparing the snowdrift game with the prisoner's dilemma. *Proceedings of the Royal Society B: Biological Sciences*, 274(1628):2965–2970.

Lerchs, H. and Grossman, I. (1964). Optimum design of open-pit mines. In *Operations Research*, volume 12, page B59. Inst. Operations Research Management Sciences 901 Elkridge Landing rd, STE 400, Linthicum Hts, MD 21090-2909.

Magazine, M. and Oguz, O. (1981). A fully polynomial approximation algorithm for the 0–1 knapsack problem. *European Journal of Operational Research*, 8(3):270–273.

Melese, F. (2009). A brinkmanship game theory model of terrorism. In *Mathematical Methods in Counterterrorism*, pages 319–332. Springer.

Mogilner, C., Shiv, B., and Iyengar, S. S. (2013). Eternal quest for the best: Sequential (vs. simultaneous) option presentation undermines choice commitment. *Journal of Consumer Research*, 39(6):1300–1312.

Nalebuff, B. (1986). Brinkmanship and nuclear deterrence: the neutrality of escalation. *Conflict Management and Peace Science*, 9(2):19–30.

Nash, J. (1951). Non-cooperative games. *Annals of mathematics*, pages 286–295.

Nemhauser, G. L., Wolsey, L. A., and Fisher, M. L. (1978). An analysis of approximations for maximizing submodular set functions—i. *Mathematical programming*, 14(1):265–294.

Pagan, N. and Dörfler, F. (2019). Game theoretical inference of human behavior in social networks. *Nature communications*, 10(1):1–12.

Papadimitriou, C. H. and Roughgarden, T. (2008). Computing correlated equilibria in multi-player games. *Journal of the ACM (JACM)*, 55(3):1–29.

Peixoto, T. P. and Bornholdt, S. (2012). Evolution of robust network topologies: Emergence of central backbones. *Physical review letters*, 109(11):118703.

Perkins, S., Mertikopoulos, P., and Leslie, D. S. (2015). Mixed-strategy learning with continuous action sets. *IEEE Transactions on Automatic Control*, 62(1):379–384.

Propp, J. G. and Wilson, D. B. (1998). How to get a perfectly random sample from a generic markov chain and generate a random spanning tree of a directed graph. *Journal of Algorithms*, 27(2):170–217.

Rapoport, A. and Chammah, A. M. (1966). The game of chicken. *American Behavioral Scientist*, 10(3):10–28.

Rossi, R. A. and Ahmed, N. K. (2015). The network data repository with interactive graph analytics and visualization. In *AAAI*.

Rowe, N. C. and Goh, H. C. (2007). Thwarting cyber-attack reconnaissance with inconsistency and deception. In *Information Assurance and Security Workshop, 2007. IAW'07. IEEE SMC*, pages 151–158. IEEE.

Saff, E. (1971). The convergence of rational functions of best approximation to the exponential function. *Transactions of the American Mathematical Society*, 153:483–493.

Samuelson, L. (2016). Game theory in economics and beyond. *Journal of Economic Perspectives*, 30(4):107–30.

Scaparra, M. P. and Church, R. L. (2008). A bilevel mixed-integer program for critical infrastructure protection planning. *Computers & Operations Research*, 35(6):1905–1923.

Serrano, M. A. and Boguná, M. (2003). Topology of the world trade web. *Physical Review E*, 68(1):015101.

Sigmund, K. and Nowak, M. A. (1999). Evolutionary game theory. *Current Biology*, 9(14):R503–R505.

Smith, J. M. and Price, G. R. (1973). The logic of animal conflict. *Nature*, 246(5427):15–18.

Stein, N. D., Parrilo, P. A., and Ozdaglar, A. (2011). Correlated equilibria in continuous games: Characterization and computation. *Games and Economic Behavior*, 71(2):436–455.

Sui, X., Cong, R., Li, K., and Wang, L. (2015). Evolutionary dynamics of n-person snowdrift game. *Physics Letters A*, 379(45-46):2922–2934.

Tambe, M. (2011). *Security and game theory: algorithms, deployed systems, lessons learned*. Cambridge university press.

Ventresca, M. and Aleman, D. (2014). A fast greedy algorithm for the critical node detection problem. In *International Conference on Combinatorial Optimization and Applications*, pages 603–612. Springer.

Vidmer, A., Zeng, A., Medo, M., and Zhang, Y.-C. (2015). Prediction in complex systems: The case of the international trade network. *Physica A: Statistical Mechanics and its Applications*, 436:188–199.

Wang, W.-X., Ren, J., Chen, G., and Wang, B.-H. (2006). Memory-based snowdrift game on networks. *Physical Review E*, 74(5):056113.

Watts, A. (2001). A dynamic model of network formation. *Games and Economic Behavior*, 34(2):331–341.

Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of 'small-world'networks. *nature*, 393(6684):440–442.

Wilson, D. B. (1996). Generating random spanning trees more quickly than the cover time. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 296–303.

Wooldridge, M. (2012). Does game theory work? *IEEE Intelligent Systems*, 27(6):76–80.

Yang, C.-L. (1994). A simple extension of the dasgupta-maskin existence theorem for discontinuous games with an application to the theory of rent-seeking. *Economics Letters*, 45(2):181–183.

APPENDIX

APPENDIX I: DETAILS ON ALGORITHM FOR SEQUENTIAL EDGE-BUILDING GAME ON NETWORKS

In the Sequential-NEBG model (Algorithm 1), an edge is selected at random and the two corresponding nodes play an EBG given some utility function. We must normalize these functions in some manner, such that the games still make sense. For instance, consider the connected component size utility function. The node's utility is then the increase in size of its connected component, and this increase may be anywhere between 1 and $N-1$, though the latter case will happen only rarely. How should the probabilities of edge proposal scale with this increase? If we are not careful, nodes may propose with too high (or two low) a probability for 'reasonable' values of the graph metric.

We handle this in three steps (not shown, but taking place in lines 5-8 in Algorithm 1): (1) Define a maximum increase in utility (2) Scale the raw increase down by the maximum, to get a scaled utility $\in [0, 1]$. (3) Transform the scaled utility to a probability, potentially in some desired bounds. More concretely:

1. Define a maximum and minimum proposal probability, $p_{\min}, p_{\max}$. Define a maximum allowed increase in utility, $u_{\max}$.

2. Given a metric $m(i, g)$, find the increase in utility i receives from connecting to j: $\tilde{m}(i, g, j) = m(i, g + ij) - m(i, g)$.

3. Scale this raw increase down by the maximum increase: $u_i = \min(\frac{\tilde{m}(i,g,j)}{u_{\max}}, 1)$.

4. Take the scaled increase u_i and pass it through some interpolation given the defined probability bounds $p_{\min}, p_{\max}$. For instance, scale linearly such that a node proposes with probability $p_{\min}$ when the scaled increase is 0, and $p_{\max}$ when the scaled increase is 1.

Thus, all proposal probabilities lie within some inputted range. The maximum allowed increase $u_{\max}$ is then a hyperparameter, and we tune for it manually by observing ranges of raw utility increases. Of course, since the observable ranges depends on the choice of hyperparameter in the first place, we have to iterate a few times to find a suitable value. Our default choice of $p_{\min}, p_{\max}$ was 0.05 and 0.95 respectively.

Alternatively, one could use something like a transformed sigmoid function. If the standard sigmoid is $S(x) = \frac{e^x}{e^x+1}$, then we amplify it up by a factor of two, and translate it down so that the positive reals map to $[0, 1]$, and add a scaling parameter t: $S'(x; t) = \frac{2e^{tx}}{e^{tx}+1} - 1 = \frac{e^{tx}-1}{e^{tx}+1}$. This would not absolve us of the tuning stage, as we would still need to find a suitable value of t. However, the non-linearity might result in more interesting games.