ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΛΟΓΙΚΗΣ ΚΑΙ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΜΩΝ

Καταστάσεις Ισορροπίας σε Μοντέλα Παιγνίων Συμφόρησης: Ύπαρξη, Πολυπλοκότητα, και Απόδοση

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΒΑΣΙΛΕΙΟΥ Χ. ΣΥΡΓΚΑΝΗ
Φοιτητής Ηλεκτρολόγος Μηχανικός
και Μηχανικός Υπολογιστών Ε.Μ.Π.
(2004)

Επιβλέπων: Ευστάθιος Ζάχος
Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούλιος 2009

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΛΟΓΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΜΩΝ
CORELAB

Καταστάσεις Ισορροπίας σε Μοντέλα Παιγνίων Συμφόρησης: Ύπαρξη, Πολυπλοκότητα, και Απόδοση

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΒΑΣΙΛΕΙΟΥ Χ. ΣΥΡΓΚΑΝΗ

Επιβλέπων: Ευστάθιος Ζάχος
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 22^{α} Ιουλίου 2009.

....	....	....
Ε. Ζάχος	Α. Παγουρτζής	Δ. Φωτάκης
Καθηγητής Ε.Μ.Π.	Λέκτορας Ε.Μ.Π.	Λέκτορας Ε.Μ.Π.

Αθήνα, Ιούλιος 2009.

....

Βασίλειος Χ. Συργκάνης

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Περίληψη

Η παρούσα διπλωματική έγκειται στο ευρύτερο πεδίο της Θεωρίας Παιγνίων και αχολείται με τη μελέτη της αλληλεπίδρασης ιδιοτελών παικτών που δρουν αυτόνομα χωρίς κάποια ρυθμιστική αρχή. Η διπλωματική επικεντρώνεται σε καταστάσεις όπου οι παίκτες επιθυμούν να δεσμεύσουν πόρους επηρεάζοντας έτσι εμμέσως το όφελος των υπολοίπων παικτών. Τέτοιες καταστάσεις στη θεωρία παιγνίων συνήθως μοντελοποιούνται με τη χρήση Παιγνίων Συμφόρησης.

Αρχικά παρουσιάζουμε βασικές έννοιες σχετικές με την Θεωρία Παιγνίων και πιο κυριότερα με θέματα που άπτωνται και της Επιστήμης Υπολογιστών. Στη συνέχεια παραθέτουμε τα βασικότερα χαρακτηριστικά των Παιγνίων Συμφόρησης και αναφέρουμε σημαντικά παραπλήσια μοντέλα που έχουν προταθεί στη βιβλιογραφία.

Το πρώτο ερώτημα που μελετάμε είναι ποιες κλάσεις παιγνίων επιδέχονται Αμιγή Ισορροπία NASH. Αφού παραθέσουμε τα σημαντικότερα αποτελέσματα της βιβλιογραφίας παρουσιάζουμε μία νέα απόδειξη ύπαρξης αμιγούς ισορροπίας NASH σε παίγνια συμφόρησης όπου οι παίκτες έχουν βάρη, οι συναρτήσεις κόστους των πόρων είναι φθίνουσες και οι στρατηγικές των παικτών αποτελούν βάσεις κάποιου μητροειδούς.

Το δεύτερο ερώτημα με το οποίο ασχολούμαστε είναι η πολυπλοκότητα υπολογισμού μιας Αμιγούς Ισορροπίας NASH σε Παίγνια Συμφόρησης. Παραθέτουμε,τα σημαντικότερα αποτελέσματα των τελευταίων ετών στο πεδίο αυτό. Επιπλέον, εξάγουμε και κάποια νέα αποτελέσματα πολυπλοκότητας, που υποδεικνύουν τη δυσκολία υπολογισμού αμιγούς ισορροπίας NASH σε ένα μοντέλο παραπλήσιο με τα παίγνια συμφόρησης.

Τέλος, εξετάζουμε την απόδοση της ισορροπίας NASH υπό το πρίσμα κάποιας συνάρτησης κοινωνικού κόστους. Παρουσιάζουμε, αποτελέσματα για το τίμημα της αναρχίας σε ορισμένα ενδιαφέροντα μοντέλα παιγνίων συμφόρησης. Παράλληλα προτείνουμε ένα νέο μοντέλο για την περιγραφή συνυθηκών ιδιοτελούς δρομολόγησης και ανάθεσης μηκών κύματος σε πολυνηματικά οπτικά δίκτυα, για το οποίο υπολογίζουμε επακριβώς το τίμημα της αναρχίας για διαισθητικά χρήσιμες συναρτήσεις κοινωνικού κόστους.

Abstract

This diploma thesis lies in the general area of Game Theory and copes with the study of the interaction of independent selfish agents under the absence of some central authority. We mainly focus on situations where players want to allocate resources in order to perform some task and hence affect the other players implicitly. Such situations are typically modeled with Congestion Games.

Initially, we present some basic notions of Game Theory, mainly focusing on its aspects that are related to Computer Science. Subsequently we study the characteristics of Congestion Games and present some interesting game theoretic models related to them.

The first question that we focus on is the existence of a Pure Nash Equilibrium (PNE) in the most important models related to Congestion Games. We present significant results from bibliography and we give a novel proof on the existence of PNE in matroid weighted congestion games with non-increasing facility cost functions.

The second issue we investigate is the complexity of computing a PNE in Congestion Games. We give analytic proofs on recent results concerning this problem. Moreover, we present some new results on the complexity of computing a PNE in an alternative model of Congestion Games.

Moreover, we examine the deterioration caused by selfishness in congestion games and present some recent results on the Price of Anarchy for some interesting Congestion Game models. Finally, we introduce a new model that is related to selfish routing and wavelenght assignment in multifiber all-optical networks and we compute its Price of Anarchy for several intuitively interesting social cost functions.

Ευχαριστίες

Με την ολοκλήρωση της διπλωματικής μου καθώς και του προπτυχιακού μου κύκλου σπουδών αισθάνομαι την ανάγκη να ευχαριστήσω βαθύτατα μία σειρά από ανθρώπους που μου πρόσφεραν απλόχερα τη βοήθεια και τη συμπαράστασή τους καθ'ολη τη διάρκεια της πορείας μου.

Αρχικά θα ήθελα να ευχαριστήσω βαθύτατα τον καθηγητή μου Στάθη Ζάχο που μου ενέπνευσε τον έρωτα για τη θεωρητική πληροφορική από τα πρώτα κιόλας έτη της σχολής, με τον ενθουσιασμό και την μεταδοτικότητά του. Υπήρξε δίπλα μου καθόλη τη διάρκεια της διπλωματικής με αγάπη, κατανόηση και συμπαράσταση. Θέλω να τον ευχαριστήσω που με έκανε μέλος του εργαστηρίου του και μου πρόσφερε μία μοναδική εμπειρία στο τελευταίο έτος των προπτυχιακών μου σπουδών.

Επιπλέον, θα ήθελα να πω ένα μεγάλο ευχαριστώ στο λέκτορα κ. Δημήτρη Φωτάκη που με βοήθησε σε πολύ μεγάλο βαθμό να εκπονήσω τη συγκεκριμένη εργασία. Ήταν πάντα πρόθυμος να με βοηθήσει και να συζητήσει μαζί μου οποιαδήποτε απορία, ενώ με εμπιστεύτηκε πολλές φορές με τις ιδέες του, γεγονός που το εκτιμώ βαθύτατα. Θέλω, επίσης, να ευχαριστήσω τον λέκτορα κ. Άρη Παγουρτζή που στάθηκε δίπλα μου από τις πρώτες στιγμές που έγινα μέλος του CORELAB. Η συμβολή του στην πορεία μου ήταν καθοριστική, αφού μου πρόσφερε την πρώτη μου ευκαιρία να συμμετάσχω στην απίστευτη χαρά της έρευνας. Θέλω να τον ευχαριστήσω για την εμπιστοσύνη του και για την φιλικότητά του.

Θεωρώ τις εμπειρίες που συγκέντρωσα κατά το τελευταίο έτος των προπτυχιακών μου σπουδών ανεκτίμητες και ένα εφόδιο που σίγουρα θα με βοηθήσει στις μελλοντικές μου σπουδές. Για το λόγο αυτό θα ήθελα να πω ένα ευχαριστώ και σε όλα τα μέλη του CORELAB που με βοήθησαν κατά την εκπόνηση της διπλωματικής.

Επίσης, θα ήθελα να ευχαριστήσω όλους τους φίλους μου και τους συμφοιτητές μου, που με στήριξαν όλα τα χρόνια των σπουδών μου με την κατανόηση και τη συμπαράστασή τους. Ενώ, επίσης, θα ήθελα να ευχαριστήσω και τη Μαριέτα για την υποστήριξή της κατά τη διάρκεια των σπουδών μου.

Κύριως, όμως, θα ήθελα να ευχαριστήσω την οικογένειά μου που βρισκόταν πάντα δίπλα μου σε οποιαδήποτε επιλογή μου. Η υπέρμετρη αγάπη τους είναι το μεγαλύτερο στήριγμα στην πορεία μου και η σκέψη ότι θα με στηρίξουν σε οποιοδήποτε πρόβλημά μου εμφανιστεί είναι το πιο πολύτιμο αγαθό. Επομένως, θα ήθελα να αφιερώσω τη διπλωματική αυτή στους γονείς μου Χρήστο και Ευαγγελία και στα αδέρφια μου Δημοσθένη και Παναγιώτη.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Introduction to Game Theory

Game Theory is the field of studying in a formal way the interaction of independent, selfish agents. Ever since the initial work of von Neummann and Morgestern in 1944 it has played a key role in economics, politics, biology, law, sports and quite recently in Computer Science (hence the topic of this thesis).

In order to model game theoretic environments one should cope with two major issues: What does "selfish agent" mean, in other words what is the goal of a "rational" selfish agent?, How do we model the interaction of those agents or equivalently how do we formalize the "game" that the selfish agents are playing with each other and the "actions" available to them?

The first question is answered by the theory of Rationality and Rational Behaviour using concepts such as utility and preference. The second question has its answer in the several different game models proposed in bibliography, most of which are subclasses of or can be represented by what is called a Simultaneous Game. A Simultaneous Game is a game where all players make a decision without knowing the strategies chosen by other players. One of the most common representation of simultaneous games are Strategic (or Normal) form games.

The details of the above notions that will be of use in this work are presented in the following sections.

1.1.1 Rational Behaviour and Utility

When stating that agents act selfishly we must be cautious of what selfishness really means. It does not mean that agents try to hurt each other or that they ignore the others. It means that agents have preferences over the several possible outcomes of the game they play and act rationally in the sense that their actions have as goal to achieve their prefered outcome.

Generally, players have preference orderings over the outcomes of the general form: $o_i \succeq o_j$ to declare that they prefer outcome o_i over o_j. The most common attempt to model the agents preferences of the outcome of the game is using utility theory. Utility theory tries to quantify the preference of agents over outcomes by using a one dimensional utility function $u : O \rightarrow \Re$, where O is the set of outcomes o_i. Although such a representation of agents' preferences is intuitively reasonable there must be some justification as to why just a single dimensional function is sufficient to model preference over a multidimensional space of outcomes. In fact such a declaration cannot be made if agents' preferences dont satisfy specific axioms that hold in most real world situations.

These axioms where completely defined by von Neumman and Morgenstern and are namely the axioms of completeness, transitivity, substitutability, decomposability, monotonicity and continuity. Due to the fact that utility theory is not of main concern in this thesis we will not present the details of such axioms and the reader is redirected to pp.24-27 of [NM44] for their initial definition and to pp.49-54 of [Sho08] for a more elaborate explanation. The axioms are just refered here to portray that there exist many conditions for utility theory to be able to represent the preferences of a player in a game and it is not trivial to claim so. What we will need in subsequent analysis throughout this work is the following theorem that intuitively states that for the games studied here single dimensional utility functions are sufficient:

Theorem 1. *If a preference relation $\succeq$ satisfies the axioms of completeness, transitivity, substitutability, decomposability, monotonicity and continuity then there exists a function $u : O \mapsto [0, 1]$ with the properties that:*

1. $u(o_1) \geq u(o_2)$ *iff* $o_1 \succeq o_2$, *and*
2. $u([p_1 : o_1, \ldots, p_k : o_k]) = \sum_{i=1}^{k} p_i u(o_i)$

Where $[p_1 : o_1, \ldots, p_k : o_k]$ denotes a probabilistic outcome where outcome o_i will occur with probability p_i.

The second intuitively claims that the utility of a probabilistic outcome is the expected utility over the several deterministic outcomes.

With the above theorem in mind we will use utility functions in the rest of the thesis without making any explicit assumption on the players' preferences but always implying that the axioms presented here hold.

By using utilities we answer the question of how to efficiently and simply model the preference of the players over the possible set of game outcomes. However it is not trivial to define what rationality really means in a game. If the outcome of the game was completely decided by a player or her utility was unaffected by actions of others, then we would talk of a simple decision-making problem that would break down to an optimization problem of maximizing the player utility separately by each player. However, in a game the outcome and hence the utility of a player is affected by the actions of others and hence rationality is not trivial. The several concepts of rational behaviour are examined by the solution concepts discussed in the next section, with most dominant that of a Nash Equilibrium.

1.1.2 Modeling the Game. Extensive and Normal Form Games

The next issue that we will cope with is how to model the interaction of players and give a completely mathematical definition of a game. When we think of a game in practice what always comes to mind are games like chess, backgammon, tic-tac-toe etc., i.e. games where players take turns to play and every player chooses his moves based on the current state of the game. To illustrate the points we want to make in this section we will use the following simple game as an example.

Example 1. *(Matching Pennies). In the game of matching pennies we have two players. Player 1 chooses "heads" or "tails". Then player 2 without knowing what player 1 has played, chooses "heads" or "tails". If they choose the same then player 1 gives a penny to player 2. If not then player 2 gives a penny to player 1.*

The above way of defining the game of matching pennies is quite sequential and involves taking turns, i.e. similarly to the games mentioned in the first paragraph. These games led to the formalization of games in extensive form. The official definition of an extensive form game is as follows:

Definition 1. *An n-person game in extensive form consists of the following:*

- *a tree* Γ *with a distinguished vertex* A *called the starting point*
- *a function, called the payoff function, which assigns an n-vector to each terminal vertex of* Γ
- *a partition of the nonterminal vertices of* Γ *into n+1 sets* $S_0, S_1, \dots, S_n$ *called the player sets*
- *a probability distribution, defined at each vertex of* S_0*, among the immediate followers of this vertex*
- *for each* $i = 1, \dots, n$*, a subpartition of* S_i *into subsets* S_i^j*, called information sets, such that two vertices in the same information set have the same number of immediate followers and no vertex can follow another vertex in the same information set*
- *for each information set* S_i^j*, an index set* I_i^j*, together with a 1-1 mapping of te set* I_i^j *onto the set of immediate followers of each vertex of* S_i^j*.*

More informally a game in extensive form consists of a tree. Each vertex of the tree corresponds a state of the game. At each vertex either some player i makes the move and thus the vertex is a member of the set S_i or a moves is made by a chance factor not controlled by any player and thus it is a member of the set S_0. If it is a chance move then the game continues to the next step with some probability distribution over all possible next states. Moreover a player may not know the exact state of the game but a set of states that the game might be and these states constitute an information set.

Games in extensive form provide a very detailed and general framework that is very usefull when there is need to examine the dynamics of the game. But in our contexts we need to hide these dynamics and speak more about the strategic characteristics of a game. The model that is more suitable in such situations is that of a simultaneous move game in strategic form. In such games players choose their strategies withouth knowing those of the rest of the players. Players may have decided their strategies in different times but they all play once and together.

One practical game that might be most suitable to explain simultaneous move games is rock-paper-scissors. In such a game players decide their strategy which consists of one action: that of choosing between the three alternatives. They play without knowing the strategy that the other has chosen and their utility is computed according to the well known rules of the game. This is of course a special case of simultaneous move games where the notion of strategy and action coincide. It will be seen later on that this is not always the case and in fact when trying to simulate an extensive form game with a strategic one, a strategy will represent a whole plan of actions.

Definition 2. *A simultaneous-move, strategic form game consists of*

- *A set of N players, $\{1, 2, \ldots, N\}$.*
- *A set of possible strategies S_i for each player i*
- *For each player i a preference relation $\succsim_i$ on the set $S = \times_i S_i$ or most often a utility function $u_i : S \mapsto \Re$ such that $\forall s, s' \in S : s \succsim s' \Leftrightarrow u_i(s) \geq u_i(s')$.*

Thus a strategic form game can be succinctly denoted as a tuple $\langle N, (S_i), (\succsim_i) \rangle$ or $\langle N, (S_i), (u_i) \rangle$.

When a player plays, he chooses from a set of possible strategies S_i. We define with $s = (s_1, \ldots, s_n)$, where $s_i \in S_i$, the strategy vector. All possible strategy vectors define all possible outcomes of the game. Thus the space of possible outcomes O defined in the previous chapter, coincides in simultaneous move games with $S = \times_i S_i$. Consequently a possible outcome o_i coincides with a possible strategy vector s. Having in mind the discussion from the previous section players have a preference ordering over all possible strategy vectors and if that ordering satisfies the axioms refered there then we can model it with a utility function $u_i : S \mapsto \Re$.

Although simultaneous move games seem unconnected with extensive form games and dont seem to capture many well known games, there exists a very strong implicit connection between the two models. The notion of strategy in strategic form games is essentially a plan of all actions that a player will take in every possible state of the extensive form game. Although in real situations noone plans all his moves at every possible state, this is p ossible in theory. For example noone will make a plan of whatever she would play in a game of chess for every possible move of the other player. In theory we can assume that such game playing is possible, so as to examine the strategic and not the dynamic characteristics of a game. This full plan is what we formally defined as a strategy. Under this perspective it is easy to understand that all extensive form games can be transformed in a strategic form game hidding of course the dynamic characteristics that in some situations might seem usefull.

To illustrate the above we give the definition of the matching pennies in strategic form:

Example 2. *(Matching Pennies). In the games of matching pennies two players choose one side of the penny "heads" or "tails". They put forward their penny and if the sides coincide then player 1 gives one penny to player 2, else the opposite. More formally the matching pennies game consists of 2 players* $\{1, 2\}$. *The set of strategies for player* 1 *and* 2 *is* $S_1 = S_2 = \{H, T\}$. *The utility function for player* 1 *is:* $u_1((H, H)) = -1, u_1((H, T)) = 1, u_1((T, H)) = 1, u_1((T, T)) = -1$ *and for player* 2*:* $u_2(s) = -u_1(s)$.

1.1.3 Expressing Strategic Form Games

One next question that is of crucial importance in game theory and especially when we want to deal with its computational aspects, is how the utility functions are described and, generally, what is the typical encoding of a game. This issue will be very important when we will try to consider a game as input to an algorithm.

One option widely used is to define explicitly the value of the utility function for each possible strategy vector. This representation of normal form games is called *standard form* or *matrix form*. In 2 player games with few possible strategies it is the most convenient way to describe a game. In fact for two player games the *matrix form* is literally a matrix that contains the utility values of each player.

Example 3. *(Matching Pennies). The matching pennies game in matrix form is the following:*

	H	*T*
H	*-1,1*	*1,-1*
T	*1,-1*	*-1,1*

Although this type of representation is the most simple one, most of the times it is not suitable as it is obvious that it is exponential in the number of the strategies that players have. More succinct representations exist such as graphical games or congestion games and some of them will be described in later chapters of this thesis.

1.1.4 Solution Concept

Having clarified a bit what selfishness means in terms of game theory and how to mathematically define games that we will cope with, the next question to ask is:

Can we make a prediction of what will be the outcome of the game?

This prediction is what we formally call a solution concept. A solution concept is most often called an equilibrium of the game, which stems from the fact that we can consider a game as a dynamic system. The structure and the rationality of the players constitute its dynamics. The outcome of the game is where the corresponding dynamic system will stabilize, i.e. where it will converge to or if it ever arrives at that state it will never leave without external intervention.

Most widely known and studied solution concepts that will be of use in the thesis will be presented in the next section. But first we must examine some information considerations that will be useful in understanding the solution concepts.

1.1.5 Information Considerations

What we haven't coped with yet in the previous sections is what information is available to the players.

In most cases and in most solution concepts presented in the next section we assume that the form and the whole structure of the game is common knowledge to the players. This is not always the case. For example it is common practice for companies to hide from the clients the details of the mechanism they use to perform some task. One such example is that of sponsored search auctions. When participating in a sponsored search auction the bidders give their valuations for an ad to be presented to users when they search a specific keyword. Although auction companies like Google and Yahoo! publish the mechanism that they use to decide which advertiser will be given a slot they don't publish the code of the program that carries out the mechanism. Hence some details of the structure of the game that players (bidders) participate in are not common knowledge.

Moreover, in most solution concepts presented in the next section the rationality of other players is also common knowledge. This means that all players consider that the other players think rationally and have as goal to maximize their utility choosing the appropriate strategy. However, this also is not the case in real games, where there exist both mallicious and oblivious players. By malicious we refer to players that dont choose the strategy that maximizes their utility but the strategy that minimizes the utility of the rest of the players. By oblivious we refer to players that dont care to maximize their utility and play an arbitrary strategy. Most solution concepts presented in the next section are not robust to this kind of perturbations that are exhibited in real life situations.

Last but not least, another type of information that players may be uncertain is the utilities of the others. They might be totally ignorant about the utility function of others or they might have a probability distribution on possible utility values of other players for the outcomes of the game. This type of information consideration is tuckled by imperfect information games and by Bayesian games with their respective solution concepts such as Bayesian Equilibrium.

1.2 Equilibria in Strategic Form Games

1.2.1 Pure Nash Equilibrium

The Nash Equilibrium is the most widely accepted and used solution concept. It was first proposed by Nash in his revolutionary work "Equilibrium Points in n-Person Games" ([Nas50]) and in his PhD thesis "Non cooperative games ([Nas51]). Formally a Nash Equilibrium of a strategic form game is defined as follows:

Definition 3. *(Pure Nash Equilibrium). Let $G = \langle N, (S_i), (u_i) \rangle$ be a game in strategic form. A strategy vector s is said to be a Pure Nash Equilibrium (PNE) of the game if:*

$$u_i(s_i, s_{-i}) \geq u_i(s'_i, s_{-i}), \forall s' \in S_i \tag{1.1}$$

Where with $u_i(s_i, s_{-i})$ we denote the utility of the strategy vector where player i plays according to s_i and the rest of the players play their strategy in the strategy vector s

More informally a strategy vector constitutes a PNE if no player has incentive to change strategy unilaterally, i.e. will not gain more by changing his strategy if all other players remain in the same state.

The intuition behind the PNE and the reasons why it has a very broad acceptance when game theory is applied to areas such as economics and social sciences lies in two different perspectives: common knowledge and dynamics.

Under the perspective of common knowledge the PNE (as brillinatly stated in [OR94]) is the state where "each player holds the right expectation of other players' behaviour and acts rationally". As stated in the first chapter we take as granted that all players act rationally. Here we define what rationally means under the sense of PNE: a player plays rationally if he plays his best response to what the rest of the players are playing. So lets define the following function.

Definition 4. *(Best Response). We denote with $B_i(s_{-i}) : \times_{j \neq i} S_j \mapsto 2^{S_i}$ the function that maps a strategy vector of the players other than i to a set of strategies of player i that are best responses, in the sense that:*

$$s_i \in B_i(s_{-i}) \Leftrightarrow u_i(s_i, s_{-i}) \geq u_i(s'_i, s_{-i}), \forall s'_i \in S_i \tag{1.2}$$

Now it is clear to see that in a PNE all players know that they are playing their best response and more importantly that others are also playing theirs.

To make things clearer lets consider a two player game with PNE $s = (s_1, s_2)$. If player one thought that player two was playing strategy $s'_2 \neq s_2$ then he would play some strategy $s'_1 \in B_1(s'_2)$. But then since rationality is common knowledge player one would say: if I think that player two is playing s'_2 and then I would play s'_1 then player two would think rationally and play $s''_2 \in B_2(s'_1)$, so then I would have to play $s''_1 \in B_1(s''_2)$ and this goes on forever. However, if player one thought that player two was playing his Nash strategy s_2, then he would have to play his Nash strategy s_1 also. Then s_2 would also be a best response for player two and the, otherwise infinite, thinking would stop here.

In terms of dynamics the PNE can be thought of as the stable point of the following dynamic system. The initial point of the system is some arbitrary strategy vector s_0. At each time step every player plays his best response to the strategies of the rest of the players in the previous time step. That is

$$s_k \in B(s_{k-1}) = \times_i B_i(s_{k-1}) \tag{1.3}$$

We can restrict the dynamic system by saying that s_k is the first in some lexicographic order of the strategies in the set $\times_i B_i(s_{k-1})$. In the above discrete dynamic system the PNE is a stable point since it is a fixed point of the mapping B. This type of dynamic system can also be represented in graphical form. Consider a directed graph where each node n_i corresponds to a strategy vector $s(n_i)$ and there exists an edge from node n_i to n_j iff $s(n_j) \in B(s(n_i))$. This graph is called a Best Response graph and a path on that graph leading to a PNE is a Best Response Path.

The above dynamic system captures the case of a repeated game where every repetition is strategically independent of the others. You can think of strategic independence as a situation similar to what Phill Connors had to face playing the same game of life every day in the movie *Groundhod Day* [Mac].

The fact that the PNE is a fixed point of the mapping B also leads to some sufficient but not necessary conditions for a PNE to exist in a game. For instance if B satisfies Kakutani's theorem conditions [Kak41] then a PNE is guarranted to exist. For example in every finite game the above conditions do not hold and a PNE is not guarranted to exist.

1.2.2 Mixed Nash Equilibrium

A mixed strategy in a game is a distribution on the set of pure strategies. Having defined a strategic game $G = \langle N, (S_i), (u_i) \rangle$ we can define its mixed extension as $G = \langle N, (D(S_i)), (U_i) \rangle$ where:

- $D(S_i)$ is the set of probability distributions on the strategies in S_i
- $U_i : \times_i D(S_i) \mapsto \Re$ is the expected value of the utility of a player

Definition 5. *A Mixed Nash Equilibrium (NE) of a strategic form game is a Nash Equilibrium of its mixed extension.*

Another important notion related to mixed strategies is that of support.

Definition 6. *(Support). The support of a mixed strategy is the set of pure strategies that receive non zero probability, and is denoted by $Supp(s)$.*

Moreover the following theorem can also be easily proved:

Theorem 2. *Let $G = \langle N, (S_i), (u_i) \rangle$ be a game in strategic form. A mixed strategy vector $a \in \times_i D(S_i)$ is a NE for the game if and only if for every player $i \in [N]$ every pure strategy $s \in Supp(a_i)$ is a best response to a_{-i}.*

In other words the above theorem states that all players are indifferent between the pure strategies in the support of their mixed strategy in the sense that they all yield the same utility.

What is really important about the NE is that it always exist in a finite game. This can be easily proved if we just see that in the mixed extension of a game the mapping B defined in the previous section satisfies Kakutani's conditions.

The problem with the NE is that it is not as intuitive as the PNE and it is not clear why would a player randomize over strategies in a real situation. However there exist a lot of work on how a mixed strategy can be interpreted in real situations and the reader is redirected to pp. 37-44 of [OR94] for a brief review.

1.2.3 Correlated Equilibrium

One of the intuitions behind the mixed strategies is that players receive external signals that are random variables with certain probability distributions. According to the signal they receive they execute some pure strategy in the support of the mixed strategy. The distribution of the random variable is such that the probability of playing a specific pure strategy is equal to the probability stated by the mixed strategy.

The random variables stated above are independent. A reasonable question to ask next is what would happen if there was a correlation between them. This question is answered by the solution concept of correlated equilibrium. The notion of correlated equilibrium was introduced by Aumann [Aum74] and some further intuition on it was given in his subsequent work [Aum87]. The formal definition is the following:

Definition 7. *A correlated equilibrium of a strategic form game $G = \langle N, (S_i), (u_i) \rangle$ consists of*

- *a finite probability space (Ω, π) (Ω is the set of states and π is a probability measure on Ω).*
- *for each player $i \in [N]$ a partition $\mathcal{P}_i$ of Ω*
- *for each player $i \in [N]$ a function $\sigma_i : \Omega \to S_i$ with $\sigma_i(\omega) = \sigma_i(\omega')$ whenever $\omega \in P_i$ and $\omega' \in P_i$ for some $P_i \in \mathcal{P}_i$*

such that for every $i \in [N]$ and every function $\tau_i : \omega \to S_i$ for which $\tau_i(\omega) = \tau_i(\omega')$ whenever $\omega \in P_i$ and $\omega' \in P_i$ for some $P_i \in \mathcal{P}_i$ we have:

$$\sum_{\omega \in \Omega} \pi(\omega) u_i(\sigma_{-i}, \sigma_i) \geq \sum_{\omega \in \Omega} \pi(\omega) u_i(\sigma_{-i}, \tau_i) \tag{1.4}$$

Thus we can state succinctly that a correlated equilibrium is a tuple $\langle (\Omega, \pi), (\mathcal{P}_i), (\sigma_i) \rangle$. The above definition is very general and not often used in practice due to the following theorem:

Theorem 3. *Let $G = \langle N, (S_i), (u_i) \rangle$ be a strategic form game. All probability distributions over outcomes that can be produced by a general correlated equilibrium can be produced by the set of restricted correlated equilibria where $\Omega = S$ and the information partition of player i consists of all the sets of the form $\{s \in S : a_i = b_i\}$ for some $b_i \in S_i$.*

Thus the representation of a correlated equilibrium as a tuple $\langle (\Omega, \pi), (\mathcal{P}_i), (\sigma_i) \rangle$ comes down to just a probability distribution on the set of possible strategy vectors s of the game. Thus the more informal definition that we will use hereafter is the following:

Definition 8. *Let $G = \langle N, (S_i), (u_i) \rangle$ be a strategic form game. A correlated equilibrium of the game is a probability distribution π over the space S such that:*

$$\forall s_i, s_i' \in S_i : \sum_{s_{-i}} p(s_i, s_{-i}) u_i(s_i, s_{-i}) \geq \sum_{s_{-i}} p(s_i, s_{-i}) u_i(s_i', s_{-i}) \tag{1.5}$$

Now lets look at the intuition behind the above definition. A correlated equilibrium implies that there exists an external factor other than the players that gives advice to the players on what to play. Lets call him the correlator. The correlator gives advice to the players according to a probability distribution on the outcomes. After performing some random procedure that complies with that distribution he picks an outome s. He then tells player i to play strategy s_i. If the probability of the correlator satisfies the equation given in definition 8 then the players have no incentive to choose a strategy other than that proposed to them by the correlator.

Thus we make the following mapping in the intuition: Ω=the set of outcomes, π=the probability distribution p, $\mathcal{P}_i$=the set of outcomes that make the correlator propose the same strategy to player i,σ_i=the strategy proposed by the correlator.

1.2.4 Bayesian Nash Equilibrium

In this section we cope with the third type of information consideration discussed in section 1.1.5, namely that where the players are uncertain of the utilities and preference of other players. The most widely used concept for coping with such uncertainty is Bayesian Games introduced by Harsanyi in his three part work "Games with Incomplete Information Played by "Bayesian" Players" (possibly his most important contribution for which he was awarded the Nobel in Economics).

A Bayesian Game as will be used in this thesis is defined as follows:

Definition 9. *A Bayesian Game consists of :*

- *A set of players $[N]$*
- *A set of actions for each player A_i*
- *A set of types for each player T_i*

- *A utility function* $u_i : \times_i (A_i \times T_i) \mapsto \Re$
- *For each player* i *a probability function* $p_i : T_i \mapsto D(T_{-i})$ *(the set of probability distributions over* T_{-i}*). This function specifies a probability distribution* $p_i(\bullet|t_i)$ *that represents player's* i *belief on other player types given that his type is* t_i*.*

Thus a Bayesian Game is succintly represented as a tuple $\langle N, (A_i), (T_i), (u_i), (p_i)\rangle$

A strategy in a Bayesian game is a bit different compared to that of a strategic form game in the sense that it is a function from the set of types of a player to the set of actions. Thus to define a Bayesian Nash Equilibrium we have to define the corresponding mixed strategy vector of a Bayesian Game.

Definition 10. *Let* $\langle N, (A_i), (T_i), (u_i), (p_i)\rangle$ *be a Bayesian Game. A strategy of a player in the game is a function* $s_i : T_i \mapsto A_i$*. A strategy vector* $(s_1, \ldots, s_N)$ *constitutes a Bayesian Nash Equilibrium if :*

$$\begin{aligned} \forall i \in [N], \forall t_i \in T_i, \forall s_i' \in S_i : \\ E_{p_i(t_i)}[u_i\left(t_i, t_{-i}, s_i(t_i), s_{-i}(t_{-i})\right)] \geq E_{p_i(t_i)}[u_i\left(t_i, t_{-i}, s_i', s_{-i}(t_{-i})\right)] \end{aligned} \tag{1.6}$$

In other words playing according to the function s_i *is a best response in expectation over the types of the players.*

We can easily define the mixed extension of a Bayesian Nash Equilibrium where a strategy is a function from the set of types to the set of probability distributions on players action. Then a strategy vector would be an Equilibrium if playing according to the mixed strategy, that is outputed by the function s_i, is a best response in expectation over the types of the players and in expectation over the possible outcomes defined by the mixed strategies.

1.2.5 ϵ-Approximate Equilibrium

The ϵ-approximate Nash Equilibrium is a solution concept recently proposed by the CS community. It was mostly brought to attention by the work of Lipton et al. [LMM03]. Before discussing the several arguments and motivations concerning this solution concept lets give its formal definition.

Definition 11. *Let* $G = \langle N, (S_i), (u_i)\rangle$ *be a strategic form game. A strategy vector* s *is called an* ϵ*-approximate Nash Equilibrium or* ϵ*-Nash Equilibrium (e-Nash) iff:*

$$\forall i \in [N], \forall s_i' \in S_i : u_i(s_i', s_{-i}) \leq u_i(s_i, s_{-i}) + \epsilon \tag{1.7}$$

In other words, no player will gain more than ϵ *by changing his strategy unilaterally.*

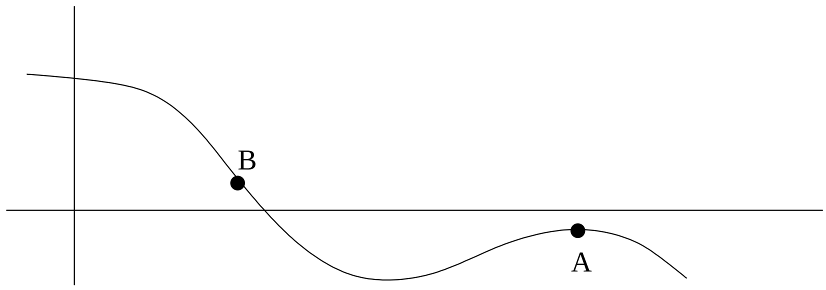

Figure 1.1: Parallelism of approximation notions between finding root of a function and equilibrium

First lets consider what approximate means in ϵ-approximate. From the above definition we get that it means that players are *approximately* rational, i.e. they play an approximate best response. This however does not at all imply that the ϵ-approximate nash equilibrium is an approximation of a nash equilbrium, i.e. that the probability distribution of the ϵ-Nash over the set of pure strategies is an approximmation of the probability distribution of some NE of the game.

To understand the different notions of approximation lets consider the problem of finding the root of a function. Lets say that the graph of the function is the one depicted in figure 1.1.

Now we have to consider the following: is A an approximation of the root or is it B. Well in analogy with $\epsilon - Nash$ equilibria we would have to say A. A great amount of work has been carried out in the other direction also under the more general issue of finding approximate fixed points of functions (see [EY07] for some latest results and a small review on past work).

What is left is to justify why choosing this type of approximation is interesting and why it constitutes a good solution concept. The answer is given by the work of Lipton et al. where they prove the following theorem (Theorem 2 [LMM03])

Theorem 4. *Let $s = (s_1, \ldots, s_N)$ be a NE of a game $G = \langle N, (S_i), (u_i) \rangle$. Then for any ϵ there exists, for every $k \geq \frac{3N^2 \ln(N^2 \max_i |S_i|)}{\epsilon^2}$, a set of k-uniform strategies $s'_1, s'_2, \ldots, s'_N$ such that:*

- $s' = (s'_1, s'_2, \ldots, s'_N)$ *is a* $\epsilon - Nash$
- $|u_i(s'_i) - u_i(s_i)| < \epsilon$ *for* $i = 1 \ldots N$

where a k-uniform strategy is a mixed strategy where all probabilities of the support are multiples of $1/k$.

The above theorem gives to major arguments of why to consider $\epsilon - Nash$ as a good solution concept. First, in every game there are guaranted to exist $\epsilon - Nash$ equilibria with very simple strategies. Simple means that all players have to just randomize over a very small support of pure strategies using a very simple probability distribution. This is very important in the contexts of bounded rationality. If the set of pure strategies was a set of resources in a game and thus was very large then randomizing over all pure strategies would be very difficult to implement by players. Moreover, randomizing with a very complex probability distribution is again difficult. On the other hand uniformly randomizing over a small set requires only polylogarithmically many bits to implement.

The second advantage of $\epsilon - Nash$ is that the utilities of the players in those $\epsilon - Nash$ that have small support don't deviate a lot from the utilities in some NE. Hence the disatisfaction of the players is limited and is counterweighted by the fact that they play a simple strategy.

One last argument in favor of $\epsilon - Nash$ equilibria that is implicitly captured by the above theorem is that there always exist such equilibria that involve computing only rational numbers. Although in 2-player games NE are always rational numbers when moving to more players, most of the times the probability distributions of NEs involve irrational numbers and hence computing the exact NE is inherently impossible. Thus we need some kind of approximation and the $\epsilon - Nash$ equilibria is a good solution in this direction.

1.2.6 In Search of New Notions

The pursuit of new solution concepts other than Nash dates back to the 70s. Mostly economists, started proposing refinements of the Nash Equilibrium that where either more suitable to specific situations or had better characteristics. Trembling-hand, perfect, sequential, subgame perfect, lexicographic are some of the most well known refinements of Nash Equilibrium (see [WG05] for a brief review). However most of those refinements where restrictions of Nash Equilibria to smaller subsets with favorable characteristics.

The increasing interest of the Computer Science community and the several complexity results regarding the computation of a Nash Equilibrium led to the search of completely new equilibrium notions. Before presenting what are favorable characteristics of a solution concept lets study what are the arguments against the Nash Equilibrium.

Arg 1. Nash is not always natural game playing

Arg 2. Nash Equilibrium is extremely fragile to oblivious or unexpected behaviour of players or to colluding players.

Arg 3. Nash Equilibrium doesn't take computational concerns into account.

Arg 4. Nash Equilibrium presumes that players are aware of the whole structure of the game and of all the players playing it. This is a truly unrealistic situation in games taking place in huge environments such as the internet.

The first argument is best portrayed by the following The Traveller's Dilemma game proposed by Basu [Bas07]:

Example 4. *An airline loses two suitcases belonging to two different travelers. Both suitcases happen to be identical and contain identical antiques. An airline manager tasked to settle the claims of both travelers explains that the airline is liable for a maximum of \$100 per suitcase, and in order to determine an honest appraised value of the antiques the manager separates both travelers so they can't confer, and asks them to write down the amount of their value at no less than \$2 and no larger than \$100. He also tells them that if both write down the same number, he will treat that number as the true dollar value of both suitcases and reimburse both travelers that amount. However, if one writes down a smaller number than the other, this smaller number will be taken as the true dollar value, and both travelers will receive that amount along with a bonus/malus: \$2 extra will be paid to the traveler who wrote down the lower value and a \$2 deduction will be taken from the person who wrote down the higher amount. The challenge is: what strategy should both travelers follow to decide the value they should write down?*

In the above game the only Nash Equilibrium is playing (2,2). The above outcome is the most unnatural one in real situations and in fact a lot of experimental results are available (even when the above game is played among game theorists) showing that a game playing of above 90 by both players is most probable.

For the second issue Halpern ([Hal08]) proposed the notion of (k, t)-robustness. An equilibrium concept is (k, t)-robust if it is unaffected by k players colluding or by t players behaving unexpectedly. Obviously the NE is $(1, 0)$ robust. A major disdvantage is that (k, t) robust equilibria are not guarranted to exist, except for the case where $(k, t) = (1, 0)$.

For the computational issue there has been a great amount of work mainly by the CS community to present solutions that take complexity characteristics into account. In most realistic situations players have bounded computational resources something that may make NE almost impossible to be the outcome of a game. Consider for example the following one player game: A person is given a number and is asked to state if it is a prime. If she guesses correct she is given 10 euros, if she guesses wrong she gives 10 euros and if she walks away no transaction is made. Obviously the NE of this game is guessing right. But guessing right requires a tremendous amount of computation that is most of the time not available to players. An even more robust statement would be the following slight modification of the above game: Instead of asking to state if a number is a prime you give the person a program and ask them to state if it halts. Claiming that this game reaches a NE for all games in the class (i.e. for all different programs given to the player) is equivalent to claiming that the player has the computational power to solve an undecidable problem!

The general way of trying to tuckle with this problem is by substituting players or strategies with some computational machine (e.g. automaton [PY94], polynomialy bounded turing machine [DHR00], or general turing machine [HP08a]). Moreover, efforts have been made by the economics community also, such as that of Rubinstein [Rub86] who tries to incorporate the computational cost in the player's utility function. Recent progress in this direction is made by Halpern and Pass

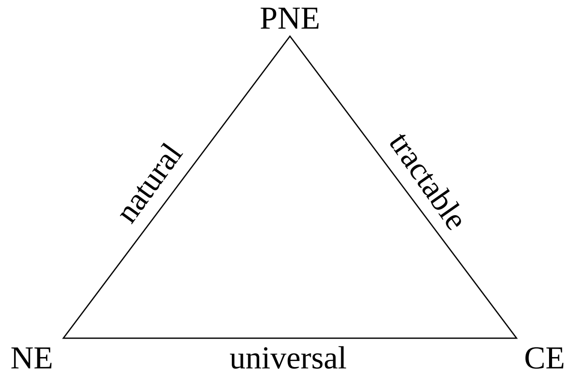

Figure 1.2: Succinct characterization of currently widely used solution concepts. None satisfies all three desired traits.

[HP08a] where they propose a general framework for taking into account computation in Game Theory.

All the above arguments against the NE solution are sufficient to trigger a whole new research area in search of new solution concepts. What is mostly desirable by a solution concept (as stated by Papadimitriou [Pap08]) is that it implies a natural game playing by the players (natural), that it always exists (universal) and that it is computationaly tractable (efficient).

Most of the existing solution concepts satisfy at most two of the above requirements. The PNE is natural and tractable in the standard form representation of the game. NE is universal and under some arguments natural but is inefficient even in the standard form. Correlated equilibrium is both universal and efficient but it is not natural in the sense that the existence of a correlating factor is not always present in games.

In the rest of this section we will discuss the following solution concepts proposed recently mainly by the CS community: (1) sink equilibria [GMV05], (2) program equilibria [Ten04], (3) unit-recall equilibria [FP08]. We will give a basic outline of all of them and present their major advantages and disadvantages.

Sink equilibria are closely related to the dynamic system intuition behind the PNE. In section 1.2.1 we presented a dynamic system and a Best Response Graph where all players played simultaneously their best response to the previous strategy vector. Goemans, Mirrokni and Vetta considered the case where only one player at a time could make a best response move. This type of dynamic system is represented in graphical form by the Nash Dynamics.

Definition 12. *The Nash Dynamics of a game* $G = \langle N, (S_i), (u_i) \rangle$ *is a directed graph* $\Gamma = (V, E)$ *where:*

$$V = \times_i S_i \tag{1.8}$$

$$E = \{(s_i, s_j) : s_i, s_j \in V \wedge (\exists k \in [N] : s_j \in (B_k(s_i), s_{-k}))\} \tag{1.9}$$

i.e. an edge exists between two nodes if the end node is a best response move to the start node for some player i.

On the Nash Dynamics graph we can define the following Random Walk (NDRW): At each node some player from the set of players that have a best response move is chosen uniformly at random. He performs an arbitrary move among his best responses. This random walk is generally not irreducible and has many transient states (see [MR96] for a brief introduction in random walks and markov chains). However a sink strong connected component of the Nash Dynamics graph forms a reducible Markov Chain with positive recurrent states and a stationary distribution can be calculated when we constrain the random walk in that component.

Definition 13. *A Sink Equilibrium of a game $G = \langle N, (S_i), (u_i) \rangle$ is a tuple (S, π) where S is a set of strategy vectors that constitute a sink stronly connected component of the Nash Dynamics of the game and π is the stationary distribution of the NDRW when constraint to S*

Thus a Sink Equilibrium is not a deterministic outcome of a game but a probability distribution on deterministic outcomes as defined in the previous definition. The major claim in favor of Sink Equilibria is that they may be more natural since they don't require randomization by the players, like NE does. On the other hand they always exist in a game. Thus, they are universal. However, as proved in [GMV05] they are intractable (PLS-complete) even in subclasses of strategic form games and as Fabrikant and Papadimitriou proved [FP08] they are PSPACE-complete for graphical games.

Moreover, if we construct the Nash Dynamics of the Traveller's Dilemma Game we could easily see that the only Sink Equilibrium of the game is (2,2). Thus, we argue that Sink Equilibria still don't imply natural game playing in all types of games. It is most probable that they are most suitable in games played over the internet such as congestion games, where it might be true that players don't update their strategies simultaneously as they repeatedly play the game but each person updates his strategy independent of others at an arbitrary time.

Program Equilibrium is a concept that was proposed by Tennenholtz [Ten04] mostly in order to capture the idea of game playing in a computer environment like the internet. Tennenholtz proposed that players don't choose strategies by themselves but instead choose a loop-free program to do that. The program of each player takes other players' programs as inputs and outputs a strategy for that player. Those strategies are then played simultaneously and so the one-shot nature of the game does not change.

Definition 14. *Let $G = \langle N, (S_i), (u_i) \rangle$ be a strategic game. Let $PROG_i(G)$ be the set of programs available to player i in game G. Let $PROG(G) = \times_i PROG_i(G)$ be the set of program profiles of game G. A program profile $(p_1, \ldots, p_n) \in PROG(G)$ is said to be a program equilibrium of game G iff:*

$$\forall i \in [N] : U_i(p_i, p_{-i}) \geq U_i(p_i', p_{-i}) \tag{1.10}$$

where $U_i(p) = u_i(s)$ where s is the mixed strategy vector that is the output of the programs in p.

One could say that program equilibrium is the unification of the two major contributions of von Neummann, the von Neummann architecture and Game Theory. One of the major characteristics of program equilibrium is that it allows for any individually rational strategy vector to be the outcome of a program equilibrium. A strategy vector is said to be individually rational if the payoff of each player in that vector is greater or equal to the amount $v_i = min_{s_{-i}} max_{s_i} u_i(s_i, s_{-i})$ (i.e. the utility of the minimum best response). Thus under this perspective we can claim that a program equilibrium is both universal and natural since an arbitrary individually rational strategy vector always exists and is efficiently computable. In addition it sometimes implies more natural game playing than that of the NE. For example in the Prisoner's Dilemma game it is possible for the (not confess,not confess) strategy to be the strategic outcome of a Program Equilibrium. However, program equilibrium implies the use of a medium like a computer where the programs of the players are gathered and run before playing the game. This mediator is equivalent to the correlator in a correlated equilibrium however here the mediator has no strategic meaning.

Unit Recall Equilibria were proposed by Fabrikant and Papadimitriou [FP08] as a form of equilibrium that is both universal and efficient. The origin of the notion of unit-recall comes from repeated games when bounded rationality is introduced. In repeated games players strategize having in mind the whole history of game playing. But if we try to incorporate bounded rationality then it is reasonable to claim that players remember (recall) only a limited number of past games.

Sink Equilibria have a major drawback, they are based on a very myopic game playing. The utility of each player is totally different among different sinks and thus it is reasonable to claim that in the first steps of the dynamics players are not myopic and are willing to sacrifice best response for some other strategy that will lead them to their favourite sink. Therefore, we have to capture somehow the strategic game playing of players. If we restrict to unit-recall players then each player would decide his strategy only based on the current state of the game. Thus the dynamics of the game would not be best response Nash Dynamics but would be defined by an automaton chosen by each player. The states of the automaton would be all possible strategies of the player and the input alphabet would be all possible strategy vectors of the rest of the players. In this new dynamics we allow all players to make simultaneous improvement moves from the current state to the next.

A profile $a = (a_1, \ldots, a_n)$ of player automata defines a function from the set S to itself. Since S is finite the repeated transitions from one strategy vector to another will eventually lead to a cycle. Lets denote such a cycle with $limitcycle(a)$ We define the utility of each player as his average utility over the strategy vectors in this cycle. With the above it is reasonable to define the following equilibrium.

Definition 15. *Let $G = \langle N, (S_i), (u_i) \rangle$ be a game. Each player chooses an automaton $a_i \in A_i$ where A_i is the space of automata with state space S_i and alphabet*

S_{-i}. An automaton strategy profile $a = (a_1, \dots, a_n)$ is a Unit Recall Equilirium (URE) iff:

$$\begin{gathered}\forall i \in [N], \forall a'_i \in A_i : \\ average_{limitcycle(a_i, a_{-i})}[u_i(a_i, a_{-i}] \geq average_{limitcycle(a'_i, a_{-i})}[u_i(a'_i, a_{-i}]\end{gathered} \tag{1.11}$$

Another way of viewing the intuition behind the unit recall equilibria has its origins at the best response mapping that was used in the dynamic intuition of PNEs and for the proof of existence of NEs. When Nash was trying to prove his existence theorem he thought of his equilibrium point as a fixed point of the best response mapping B. But what if we dont restrict ourselves to the best response behaviour? What if we examine the equilibrium (fixed point) of some other reasonable mapping? What if even better we give the players the freedom to choose among a set of mappings? And this is what the unit recall equilibrium really is about. When players are given the choice to choose among the set of finite automata A_i they are given the freedom to choose among the set of possible mappings defined by the automaton profile $(a_1, \dots, a_N)$. This freedom gives rise to new strategic outcomes in equilibrium that where not possible when restricted to best response dynamics.

URE unlike the PNE doesn't consider as equilibrium the fixed point of this new mapping over the set of mixed strategies, since a fixed point, just like the PNE, is not guarranteed to exist here neither. However, what is guarranteed to exist is a periodic cycle of strategy vectors where the dynamics are going to settle. This cycle is equivalent to a cycle in the best response dynamics whenever a game has no PNE. In the unit recall equilibrium we consider as utility of the player the average of his utility on the strategy vector of this periodic cycle.

Two major disadvantages of URE is that they don't always exist (even in the game of matching pennies) and that it is not yet proved if they are efficient. The reason why it seems promising is because of the great characteristics of a restriction of URE, called **Componentwise URE**. In CURE an equilibrium is reached when players cannot achieve a better average utility by changing only one transition of their automaton and not their whole automaton. CURE are both efficient and always exist, however insufficient results exist on why they would be considered natural even in specific game theoretic environments like the internet.

Another direction towards new solution concepts involves incorporating learning strategies by the players. Such solution concepts are regret-minimizing and learning protocols [BHLR07, KPT09, HP08b].

We conclude this section by pointing out that there is still a lot of research to be done towards identifying a universal, natural and efficient solution concept and it will certainly be a prolific research area in the years to come.

1.3 Game Theory and Computer Science

At the end of the last century computer science started coping with problems and environments where central coordination was impossible (e.g. the Internet).

Thus computer scientists had to turn to game theory leading to the formation of a new scientific field: Algorithmic Game Theory. Algorithmic Game Theory comes to answer two questions: how Game Theory can help Computer Science in contexts where introducing selfishness is inevitable; how Computer Science can help Game Theory by identifying complexity characteristics of game theoretic problems ([Das08]) or examining algorithmic aspects of mechanism design [NR99]. In the sections that follow we will try to describe briefly the interplay of the two fields (CS and Game Theory) in several contexts.

1.3.1 Game Theory to the Rescue

Some of the main fields of Computer Science where Game Theory has become an invaluable tool are the analysis of the Internet, cryptography, study of competitive versions of classic algorithmic problems (facility location, k-median etc.).

The internet is arguably the most interesting and complex artificial system. Its characteristics have been the study of many subfields of computer science. What is unique about it is that it hasn't been designed by the beginning and it hasn't been developed by a single entity. Instead it is run by a very large and heterogeneous set of organizations each trying to maximize their own gain. The complexity of the Internet is so big that it is most suitable to be studied as a physical system with the use of the scientific method (Observations, Experiments, Falsifiable Theories, Specialized Applied Mathematics). Moreover, the intense connection among the Internet and the people that make use of it makes inevitable the study of its socioeconomic aspects. Thus, as stated by Papadimitriou the Internet has slightly turned Computer Science into a natural and social science over the past two decades.

When coming to the Specialized Applied Mathematics part of studying the web, the distributed and multiagent environment of the net makes Game Theory the ideal mathematical model to use ([Pap01]). As stated by Scott Schenker:

> The internet is an equilibrium, we just have to identify the game.

Thus an interesting part of Algorithmic Game Theory lies in the modeling of games in the Internet and of its characteristics. The most well studied model towards this direction is congestion games. With congestion games and more specifically with network congestion games we are trying to capture the selfish character of routing in the internet. Most of the results in this field will be presented in the second chapter of the thesis.

The interplay between Cryptography and Game Theory ([DR07]) has been mutual and mainly in the subfield of Cryptography called Multi-Party Computation (MPC). MPC studies the following case. We have $n \geq 2$ parties $P_1, \ldots, P_n$ where party P_i holds input t_i, and they wish to compute together a function $s = f(t_1, \ldots, t_n)$. The goal is that every party will learn the output s of the function but will not gain any knowledge on the inputs of other parties except what can be deduced from the pair (t_i, s). Game Theory comes into the scene by replacing parties that follow a cryptographic protocol with rational players that want to maximize some payoff function.

Ever since Game Theory has gained the attention of the CS community many competitive versions of classic algorithmic problem arised. The introduction of selfishness gave room for new questions and new research goals. For example the seminal work of Vetta [Vet02] introduced the competitive facility location and k-median problem.

1.3.2 Computer Science to the Rescue

What Computer Science has to offer to Game Theory mainly comprises of two parts: studying complexity characteristics of computing equilibria in games and studying complexity and algorithimic aspects of mechanism design. The two subsequesnt chapters cover in brief and certainly not completely the most recent results in these two areas.

Complexity Results

The major complexity result so far in game theory is that the problem of computing a Nash Equilibrium when given as input the standard form of game is complete for the complexity class PPAD. For the "game theory" reader there are two points that need explanation: How do we formally define what is the problem of computing a NE and how do we count its complexity?, What is PPAD?. We will try to answer these two questions in the following paragraphs.

First we have to state that Nash Equilibria for 3 players and more involve in general irrational numbers. Thus it is futile to try to compute NE exactly. So we have to use the notion of approximation. Moreover, from Theorem 2 we know that the pure strategies in the support of each players mixed strategy are a best response. In fact the difficulty of computing a NE lies in computing the support for each player. After finding the support what remains is just solving a linear system which is computationally easy.

By combining the above properties the combinatorial problem that we need to focus on is finding the support of an approximate Nash equilibrium. In the recent complexity results concerning the NE ([DGP06, CD06]) the combinatorial relaxation of finding a NE is the following:

Definition 16. *r-Nash is the search problem R with input x, a r-player game $G = \langle N, (S_i), (u_i) \rangle$ in standard form and a binary integer A and output a mixed strategy profile $a \in \times_i D(S_i)$ that is an ϵ-approximately well supported equilibrium, that is:*

$$\forall i \in [N], \forall s_i, s_i' \in S_i : U_i(s_i, a_{-i}) > U_i(s_i', a_{-i}) + \epsilon \Rightarrow s_i' \notin Supp(a_i) \tag{1.12}$$

The above means that we allow pure strategies that are a distance ϵ of the best response to be in the support of a players mixed strategy. This notion of approximate equilibrium is more strict than the ϵ-approximate equilibrium described in section 1.2.5 in the sense that the ϵ-approximate equilibrium is a subset. How ever, the problem of calculating an ϵ-approximate NE can be reduced in polynomial time to that of calculating an ϵ-approximately well supported NE. Thus any computational property of the second also applies to the first.

One advantage of the above approximate equilibrium against NE is that there always exists ϵ-approximately well supported equilibrium where the probabilities of the mixed strategies for each player are rational numbers. In fact by rounding a NE to the nearest mixed strategy profile where all probabilities are multiples of $\epsilon / \max_i \sum_s u_i(s)$ we achieve an ϵ-approximately well supported equilibrium.

Having defined the strict computational problem of computing a NE we will now state more strictly the recent results concerning its computational intractability.

Theorem 5. *r-Nash is computationally equivalent to 2-Nash and both are PPAD-complete*

The above theorem was a result of three papers. Initially Daskalakis et al. [DGP05] proved that 4-Nash is PPAD complete. This result was a breakthrough after a lot of years of efforts for some intractability result on the problem. After a really short time Daskalakis et al. [DP05] and Cheng et al. [CD05a] proved indepentently that 3-Nash is PPAD complete. Finally, in [CD05b] Cheng et al. achieved the final result that 2-Nash is PPAD complete. The last result was somehow unexpected since it was long thought that the two player case would be computationally easier since the techniques used in calculating equilibria in such games where similar to linear programming.

What remains still unanswered is what is the class PPAD and why theorem 5 consitutes an intractability result similar to the NP-complete results for other problems.

So why not NP-complete is a reasonable question. First of all the problem of finding a Nash Equilibrium is a search or function problem and not a decision so we would have to prove that it is FNP-complete. Second, the existence version of the r-Nash is trivial to answer since it is always yes. Thus it cannot be FNP-complete.

So how do we characterize and analyze problems where the decision version gives trivially a yes answer but the task of finding that answer is not trivial. There exists a bunch of such problems where existence is proved based on some non constructive property. Such problems are for example computing fixed points of brower functions, computing equilibria in exchange economies or finding a trichromatic triangle in the computational version of the sperner problem.

As correctly understood the common characteristic of the above problems is that they are search problems where a solution is guaranteed to exist. This fact led Megiddo and Papadimitriou to introduce in [MY89] the class TFNP. However, the problem about the class TFNP is that it is a semantic class, which means that the property that defines inclusion of a problem in the class is a function of all instances of the problem and not of a single computation like the one that defines P or NP. The difficulty with semantic classes is that they tend not to have complete problems. Thus in order to find complete problems Papadimitriou introduced in

[Pap94] the method of categorizing the problems in TFNP according to the non constructive property on which the proof of existence of a solution is based. Some of the classes of this sort are PLS (which will be revisited in the second chapter), PPP, PPA and PPAD.

Under this perspective PPAD is the class that is based on the fact:

> (parity argument for directed graphs) In every directed graph where all edges have indegree and outdegree at most one if there exists a source there must exist a sink.

A problem in PPAD is defined in terms of a polynomial algorithm A that takes as input nodes of a graph and outputs its predecessor and its successor if they exist. Node 0 is a source with node 1 as its successor. The problem asks to find a sink or a source in the graph other than 0.

The current status on the PPAD class is $FP \subseteq PPAD \subseteq TFNP \subseteq FNP$ but whether the above inclusions are tight still remains open. For example it is a really interesting problem whether $TFNP = FP$, in other words whether it is computationally easy to find a solution when we know that one exists.

In the initial work of Papadimitriou [Pap94] many interesting problems were proved to be complete in the PPAD class. However, the r-Nash problem remained open until the recent results.

Another important point on the results of PPAD-completeness is that adding to the r-Nash problem some other desired property about the NE (e.g. finding the best or worse NE under some social function, finding all the NEs) makes the problem most of the times NP-complete.

Now the last question that comes to mind concerning the above line of research is why is it important to find how hard it is to compute a Nash Equilibrium. The answer is in the moto: If your laptop cant find it then neither can the market. In other words for a game in practice to converge to a NE it has to perform certain tasks. Those tasks would essentially constitute an algorithm for finding a NE. Thus the convergence time for a game to reach a NE is greater or equal to its computation time.

Another interesting direction concerning the complexity of NE is finding a polynomial approximation scheme. Recent results such as [DP08] prove that there exists a PTAS with running time $poly(n)\,(1/\epsilon)^{O(\log^2(1/\epsilon))}$ for solving r-Nash in anonymous games (games where players utilities depend only on the ch osen strategy and the number of players playing each strategy). However, for this line of research to have a practical meaning there must be some proof that there exist natural game dynamics that lead to approximate equilibria. The reason for this is that the goal for proving that approximation is easy is to imply that it is also easy for the "market" to approximate. But then we also need to see if it is also natural to approximate under specific dynamics otherwise the above statement would not hold.

In conclusion, how do we explain the gist of the above results say when we try to explain it to an economist or an AI folk. Well, we could briefly state that

Nash Equilibrium for 2-player games has the same complexity as r-player Nash and finding fixed points and we have some evidence that no efficient solutions exist for such problems.

Algorithmic Mechanism Design

So far we have looked at games with a specific structure where we studied the outcomes of several solution concepts. A large part of the game theory literature is concerned with the opposite direction: Does there exist a game where if players play according to some solution concept then the outcome of the game implements some goal function; This part of game theory is called mechanism design or implementation theory.

More formally in mechanism design we have to define a modification of the strategic form games as follows. A strategic form game with consequences in C is a game $\langle N, (A_i), o\rangle$ where $o : A \to C$ is an outcome function. This game together with a preference relation for each player $\succeq_i$ on the set of outcomes is equivalent to a strategic form game $G = \langle N, (A_i), \succeq_i'$ where $a \succeq_i' b$ iff $o(a) \succeq_i o(b)$. Given a solution concept S and a goal function $f : U \to C$ from the set of utility profiles to the set of possible outcomes, we want to ask if there exists some function o such that $o(S(G, \succeq)) \in f(\succeq)$. I.e. the outcome of the game when players play under solution concept S is the same as that defined by function f. When we restrict the above concept to the case where the set of actions is the set of preference profiles, then we say that a function o truthfully implements f if $\succeq = S(G, \succeq)$ (the true preference profile of the players is a solution to the game), i.e. players dont gain by lying about their preference relation $\succeq$.

Mechanism design has been mostly applied to resource allocation situations. In such situations the outcome of the game is most of the times an allocation of the goods to the players plus a price for each player. Thus a mechanism constitutes of two functions. According to the goal function and to the solution concept used we have different results.

For example when the goal function is total welfare (i.e. the sum of the utilities of the players) and the solution concept is dominant strategies then the only mechanism proved to implement the above function is the VCG mechanism. The VCG mechanism takes as input the valuation of all players for each possible allocation and outputs an allocation and the payments for each player.

What Computer Science has to offer in the above field is the study of the complexity characteristics of the mechanisms. For example how efficient is it to implement a VCG mechanism in a specific environment? Moreover, is it efficient to approximate a goal function and is it efficient to approximate it truthfully?

Chapter 2

Congestion Games

2.1 Introduction

Congestion Games where introduced by Rosenthal as a class of games possesing Pure Nash Equilibria. They have gained a lot of attention recently both because of their interesting theoretic characteristics and because they have found application to several real world environments such as communication networks. In this part of the thesis we will present most aspects of the recent congestion games literature and give details of significant findings.

2.1.1 Model and Notation

Consider the following game theoretic situation:

Example 5. *n firms are engaged in production. Each firm has available different sets of production resources that it could use to carry out its work. All resources have a set up cost that is a function of the firms that will use the resource. The cost for each firm to produce the final product is the sum of the set up costs for each resource it uses.*

It is possible to represent the above game in normal form. However, it is easy to see that games of such nature constitute a subclass of normal form games, mainly because of the dependence of the facility cost only on the number of firms using it and not on the specific subset of firms. So we expect to be able to derive specific characteristics for such types of games and thus we need to introduce a new model.

Definition 17. *A congestion game G is a tuple $G = \langle N, F, (A_i)_{i\in[N]}, (d_f)_{f\in F}\rangle$ where $[N]$ denotes the set of players, F denotes the set of resources, $A_i \subseteq 2^F$ denotes the strategy space of player i and $d_f : \mathbb{N} \to \mathbb{Z}$ a cost function associated with resource f. $a = (a_1, \ldots, a_N)$ is an outcome of the game in which player i chooses strategy $a_i \in A_i$. For an outcome a we define the congestion $n_f(a)$ on resource f by $n_f(a) = |\{i \in [N] | f \in a_i\}|$. The cost for each player is the sum of the costs on the facilities of his strategy, $c_i(a) = \sum_{f\in a_i} d_f(n_f(a))$.*

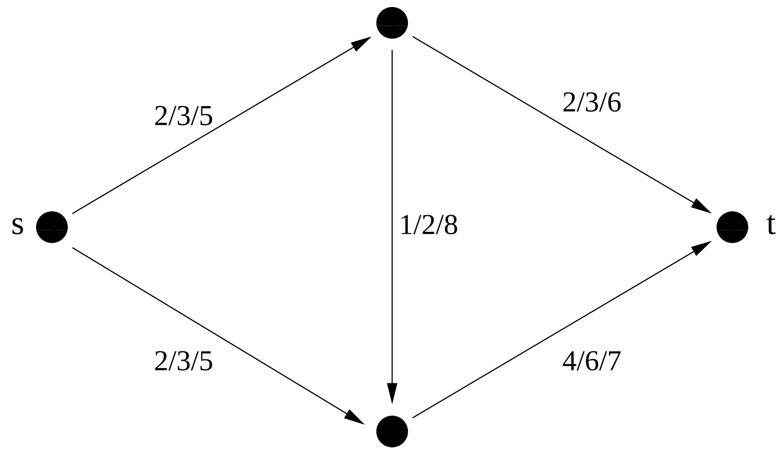

Figure 2.1: Example of a congestion game in a network. Three players want to route traffic from s to t. The delay of each edge for every possible congestion is given explicitly separated by a /.

A similar situation where the congestion game model could be directly applied is the following:

Example 6. *A network of roads is given. Each of n people wants to travel on this network from a source to a destination. The amount it takes for a person to traverse a road on the network is a increasing function of the people that use it. All players want to minimize the total time it takes to reach their destination. An example of such a network is depicted in figure 2.1.*

Thus, congestion games constitute a very large class of simultaneous move games and could be used to model many different real world environments.

Except for the applicability of the model, in order for it to be worth studying, it should have special characteristics not possesed by general normal form games. This is indeed true. The most important attribute of congestion games that was pointed out in the initial work of Rosenthal is the following:

Theorem 6. *All congestion games possess at least one pure Nash Equilibrium.*

Proof. We define the following potential function $\phi : \times_i A_i \to \mathbb{Z}$ ([Ros73]):

$$\phi(a) = \sum_{f \in F} \sum_{i=1}^{n_f(a)} d_f(i) \tag{2.1}$$

Let $a' = (a'_i, a_{-i})$.

$$
\begin{aligned}
&\phi(a) - \phi(a') = \\
&\sum_{f \in a_i / a'_i} \sum_{i=1}^{n_f(a)} d_f(i) + \sum_{f \in a_i / a'_i} \sum_{i=1}^{n_f(a)} d_f(i) - \sum_{f \in a_i / a'_i} \sum_{i=1}^{n_f(a')} d_f(i) - \sum_{f \in a'_i / a_i} \sum_{i=1}^{n_f(a')} d_f(i) \\
&= \sum_{f \in a_i / a'_i} \left(\sum_{i=1}^{n_f(a)} d_f(i) - \sum_{i=1}^{n_f(a)-1} d_f(i) \right) + \sum_{f \in a'_i / a_i} \left(\sum_{i=1}^{n_f(a)} d_f(i) - \sum_{i=1}^{n_f(a)+1} d_f(i) \right) \\
&= \sum_{f \in a_i / a'_i} d_f(n_f(a)) - \sum_{f \in a'_i / a_i} d_f(n_f(a) + 1) \\
&= c_i(a) - c_i(a')
\end{aligned}
\tag{2.2}
$$

Thus every improvement move of a player decreases the potential by at least one. Since the potential is upper and lower bounded by some finite amount, every sequence of improvement steps must be finite and end up at a pure Nash Equilibrium.

□

The intuition behind the potential function of Rosenthal can be more clearly seen by another representation of the same potential, proposed by Vocking [Vöc06]:

Let $n_f^{(i)}(a)$ denote the number of players whose index is $\leq i$ and use facility f and let $c'_i(a) = \sum_{f \in a_i} d_f(n_f^{(i)}(a))$ i.e. $c'_i(a)$ is the cost of player i if players $i+1, \ldots, N$ don't exist. It can be observed after exchanging the order of summation in Rosenthal's potential that we can write:

$$
\phi(a) = \sum_{i=1}^{N} \sum_{f \in a_i} d_f(n_f^i(a)) = \sum_{i=1}^{N} c'_i(a) \tag{2.3}
$$

This leads as to the following interpretation of Rosenthal's potential: Users are added one after the other into the game, and the potential acounts for the cost of the users at their insertion time. Such a calculation of the potential of an outcome of the game of figure 2.1 is depicted in figure 2.2.

From this interpretation it can be easily seen that the potential accounts for the real cost of the last user inserted. Thus if the last user makes an improvement move then the potential drops by the same factor as the cost of the user. However, since the order of the insertion in the game is irrelevant for the overall value of the potential (remember the initial formulation), the last claim can be made for any player.

Rosenthal's proof, gave rise to the use of potential functions in games that was later widely studied in bibliography. In brief, Rosenthal defined a real valued function on the strategy profiles of the game that exaclty tracked the difference of the utility of a single player when he changed strategy unilateraly. Due to the finiteness

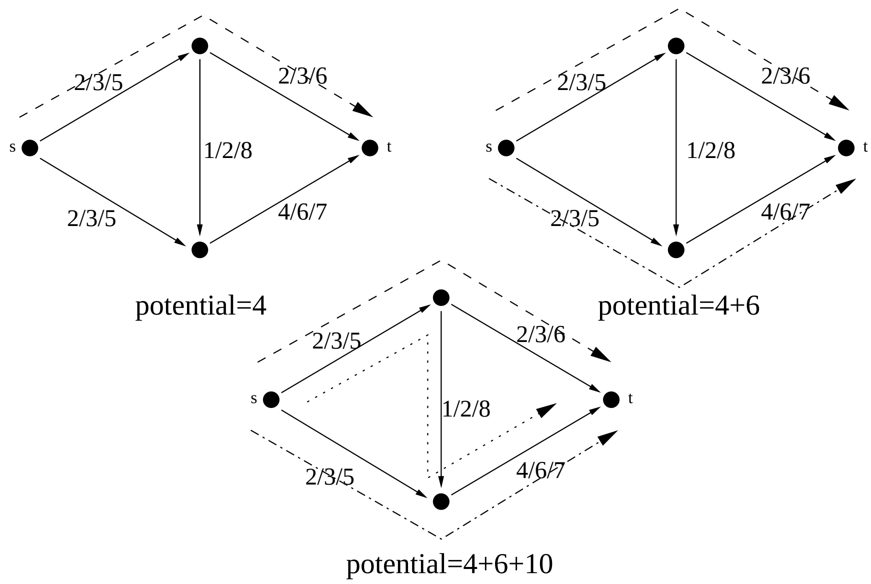

Figure 2.2: Example of calculation of the potential of an arbitrary outcome of a network congestion game. The potential accounts for the true cost of the last player inserted

of the strategy profile space, such a function must have a maximum/minimum. All strategy profiles in the *argmin/argmax* of this potential inevitably constitute a pure Nash Equilibrium, since any improvement move of a player from a strategy profile in that set would lead to a contradiction.

In a way potential functions have the role of a Lyapunov function if you think of the game as a dynamic system described by the Nash Dynamics.

2.2 Superclasses, Subclasses and Alternatives

Despite the interesting attributes and the generality of the basic model of congestion games, it doesn't capture well all types of game theoretic situations where players choose among shared resources.

In some situations the generality of the model makes it difficult to study and better attributes emerge if we restrict ourselves to smaller subclasses of games. This is the reason why many subclasses of congestion games have been largely studied in bibliography. The two main subclasses that we will study in this section are Network Congestion Games and Load Balancing Games.

Moreover, despite the generality of the model it still fails to capture some intuition behind what kind of games it consists of and what is the essense of the class of congestion games. In order to study congestion games under another perspective Monderer and Shapley introduced Potential Games [MS96] which is a class of games isomorphic to congestion games, but defined in a totally different way.

Furthermore, the basic model doesn't capture situations where players are not homogeneous, i.e. some of them are more "important" than others (weighted congestion games), some of them are affected by facility congestion in different ways than others (congestion games with player specific paoyoffs) or some of them may cooperate (Coalitional Congestion Games). This three generalizations will be studied in brief in the next chapters.

Last but not least, several alternative models that deal with situations similar to those of congestion games have been proposed and cannot be cast as generalizations or subclasses of them. In particular nonatomic congestion games is a model almost identical to congestion games where the players are infinite and the affect of an individual player on the cost of a facility is negligible. Another, alternative model is Bottleneck Games. The only difference between Bottleneck and Congestion Games is the function that aggregates the costs of the facilities that players use to give the total cost of a player. While in congestion games this function is the *sum*, in Bottleneck Games it is the max. Both classes of games can be viewed as a subclass of Generalised Congestion Games, which is the class of games with arbitrary aggregation function and are introduced in [Kuk04].

Last but not least, a model that was first proposed during the process of the thesis is Colored Resource Allocation Games. This class of games are Generalised Congestion Games where players have their strategies in several copies. Colored Congestion Games and Colored Bottleneck Games are subclasses of Colored Resource Allocation Games where the aggregation function is the max and *sum* accordingly.

The definition and most important characteristics of all the aforementioned models will be discussed in the next chapters and most of them will be revisited in the next sections under the perspective of some specific attribute.

2.2.1 Potential Games

Potential Games where introduced by Monderer and Shapley [MS96]. They introduced several notions of potentials in strategic form games. The most general potential model is the following:

Definition 18. *(Ordinal Potential Games) Let $G = \langle N, (S_i), (u_i) \rangle$ be a game in strategic form. A function $P : \times_i S_i \to \mathbb{R}$ is called an ordinal potential for G, if for every $i \in [N]$ and for every $y_{-i} \in S_{-i}$:*

$$u_i(x, y_{-i}) - u_i(z, y_{-i}) > 0 \; iff \; P(x, y_{-i}) - P(z, y_{-i}) > 0 \tag{2.4}$$

*for every $x, z \in S_i$. G is called an **Ordinal Potential Game** if it admits an ordinal potential.*

Intuitively an ordinal potential is a function that follows the sign of change of any players utility when he changes unilateraly. An even more relaxed type of potential is the generalized ordinal potential:

Definition 19. *(Generalized Ordinal Potential) Let $G = \langle N, (S_i), (u_i) \rangle$ be a game in strategic form. A function $P : \times_i S_i \rightarrow \mathbb{R}$ is called a generalized ordinal potential for G, if for every $i \in [N]$ and for every $y_{-i} \in S_{-i}$:*

$$u_i(x, y_{-i}) - u_i(z, y_{-i}) > 0 \;\Rightarrow\; P(x, y_{-i}) - P(z, y_{-i}) > 0 \tag{2.5}$$

for every $x, z \in S_i$.

The admitance of the above type of potential is the necessary and sufficient condition for a game to have the Finite Improvement Property, i.e. every best response sequence is finite.

A more strong kind of potential is the one encountered so far in congestion games where a function tracks not only the sign but also the amount of the change.Inspired possibly by Rosenthal's potential Monderer and Shapley introduced a more strong potential game model.

Definition 20. *Let $G = \langle N, (S_i), (u_i) \rangle$ be a game in strategic form. A function $P : \times_i S_i \rightarrow \mathbb{R}$ is called an exact potential (or just potential) for G, if for every $i \in [N]$ and for every $y_{-i} \in S_{-i}$:*

$$u_i(x, y_{-i}) - u_i(z, y_{-i}) = P(x, y_{-i}) - P(z, y_{-i}) \tag{2.6}$$

for every $x, z \in S_i$. G is called a ***Potential Game*** *if it admits a potential.*

With arguments similar to those used in the proof of theorem 2.1.1 we can prove that any improvement path in a potential game is finite (Finite Improvement Property) and consequently all potential games pocess at least one PNE.

An exact potential game may have many potentials but all differ with each other by some constant (as is usual with the use of potentials in physics). The set of local minima of the potential is exactly the set of PNE of the initial game. Moreover, the total minima of the potential could be used as an equilibrium refinement tool.

An interesting question is whether there exists an easy way to check if a game in normal form is a potential game. This is answered by the following theorem:

Theorem 7. *G is a potential game iff for every $i, j \in [N]$, for every $a \in S_{-\{i,j\}}$ and for every $x_i, y_i \in S_i$ and $x_j, y_j \in S_j$:*

$$\begin{gathered} u_i(y_i, x_j, a) - u_i(x_i, x_j, a) + u_j(y_i, y_j, a) - u_j(y_i, x_j, a) + \\ u_i(x_i, y_j, a) - u_i(y_i, y_j, a) + u_j(x_i, x_j, a) - u_j(x_i, y_j, a) = 0 \end{gathered} \tag{2.7}$$

The intuition of the above condition is that the total improvement incured to the players during a closed path on strategy profiles (as shown in figure 2.3) is zero.

Example 7. *Consider the Prisoner's Dilemma game: Two prisoners are held separately in custody. They have no means of communication and have the options*

$$
\gamma = \begin{tikzcd}
(x_i, x_j, a) \arrow[d] & (x_i, y_j, a) \arrow[l] \\
(y_i, x_j, a) \arrow[r] & (y_i, y_j, a) \arrow[u]
\end{tikzcd}
$$

Figure 2.3: The simple closed path of player responses. If the total gain of the players on any such path is zero then the game admits a potential

of either to Confess (C) or Defect (D). The police tries to convince them to confess. They tell them that if they confess and the other defects then they will be set free for cooperating and the other person will go to prison for 4 years. If they both confess they will both get a 3 year imprisonement for cooperating. If they both defect then they will be be imprisoned for 1 year for minor offenses. The Prisoner's Dilemma game is an exact potential game. The game in matrix form and its potential are depicted in figure 2.1.

c_1, c_2:

P_1 \ P_2	C_2	D_2
C_1	$3, 3$	$0, 4$
D_1	$4, 0$	$1, 1$

P:

P_1 \ P_2	C_2	D_2
C_1	3	4
D_1	4	5

Table 2.1: Prisoner's Dilemma game and one of its exact potential

Now that we have seen some of the basic facts and examples of potential games the major issue is what is their connection with congestion games. From Rosental's potential we know that all congestion games are potential games, but is there a stronger connection.

Let $G_1 = \langle N, (S_i^1), (u_i^1) \rangle$, $G_2 = \langle N, (S_i^2), (u_i^2)$ be strategic form games. We say that G_1 is isomorphic to G_2 if there exists bijections $g_i : S_i^1 \to S_i^2, i \in [N]$ such that:

$$\forall (s_1, \ldots, s_N) \in S^1 : u_i^1(s_1, \ldots, s_N) = u_i^2(g_1(s_1), \ldots, g_N(s_N)) \tag{2.8}$$

Under the perspective of this type of isomorphism Monderer and Shapley proved the following amazing connection between the two class:

Theorem 8. *Every Potential Game is isomorphic to a Congestion Game*

Thus we see that the two classes talk about the same type of games but in a totally different way. The proof of theorem 8 provided in [MS96] is quite technical. We will provide here a more intuitive proof given by Voorneveld et al. [VBM99].

Definition 21. *A game $G = \langle N, (S_i), (u_i)\rangle$ is a:*

- *coordination game if there exists a function $u : S \to \mathbb{R}$ such that $u_i = u$ for all $i \in [N]$.*
- *dummy game if for all $i \in [N]$ and for all $x_i, y_i \in S_i : u_i(x_i, x_{-i}) = u_i(y_i, y_{-i})$*

The following theorem states that a strategic game is a potential game iff the utility of each player i can be broken into a part that is identical for all players and a part that is dependent only on the strategies of the players except i.

Theorem 9. *Let $G = \langle N, (S_i), (u_i)\rangle$ be a strategic game. G is a potential game iff there exist functions c_i and d_i such that:*

- $\forall i \in [N] : u_i = c_i + d_i$
- $\langle N, (S_i), (c_i)\rangle$ *is a coordination game*
- $\langle N, (S_i), (d_i)\rangle$ *is a dummy game*

Proof. The opposite direction is obvious since c_i is a potential for the game. To prove the forward direction we just write the utilities in the form: $u_i(s) = P(s) + (u_i(s) - P(s))$ and observe that $\forall s'_i \in S_i : u_i(s_i, s_{-i}) - P(s_i, s_{-i}) = u_i(s'_i, s_{-i}) - P(s'_i, s_{-i})$. Thus, if we set $c_i = P$ and $d_i = u_i - P$ we get the desired result. □

To prove the final isomorphism result we first prove two intermediate results.

Theorem 10. *Every coordination game $\langle N, (S_i), (c_i = P)\rangle$ is isomorphic to a Congestion Game.*

Proof. For every strategy profile s in the potential game introduce a facility $f(s)$ in the congestion game. The cost function of the facility is the following:

$$d_{f(s)}(i) = \begin{cases} P(s) & i = N \\ 0 & \text{otherwise} \end{cases} \tag{2.9}$$

The strategy space of player i in the congestion game is defined by the bijection $h_i(s_i) : S_i \mapsto A_i$:

$$h_i(s_i) = \{f(s_i, s_{-i}) | s_{-i} \in S_{-i}\} \tag{2.10}$$

It can be observed that for every strategy profile $(s_1, \ldots, s_N)$ of the potential game the only facility that is used by N players in the strategy profile $(h_1(s_1), \ldots, h_N(s_N))$ of the congestion game is facility $f(s)$. Thus the cost of all players in the congestion game is $d_{f(s)}(N) = P(s) = c_i(s)$ since all other facilities incur zero cost. □

Theorem 11. *Every dummy game $\langle N, (S_i), (u_i)\rangle$ is isomorphic to a Congestion Game.*

Proof. For every player $i \in [N]$ in the potential game we define a facility $f(s_{-i})$ for every $s_{-i} \in S_{-i}$ in the congestion game. The cost of each such facility is defined as:

$$d_{f(s_{-i})}(i) = \begin{cases} u_i(s_{-i}) & i = 1 \\ 0 & \text{otherwise} \end{cases} \tag{2.11}$$

The strategy of player i in the congestion game is defined by the bijection $g_i(s_i) : S_i \mapsto A_i$:

$$\begin{aligned} g_i(s_i) =& \{f(s_{-i}) | s_{-i} \in S_{-i}\} \cup \\ & \{f(x_{-j}) | j \in N/i, x_{-j} \in S_{-j}, x_i \neq s_i\} \end{aligned} \tag{2.12}$$

Let $s = (s_1, \ldots, s_N)$ be a strategy profile of the potential game and $g(s) = (g_1(s_1), \ldots, g_N(s_N))$ the corresponding strategy profile in the congestion game. The only player that uses facility $f(s_{-i})$ in $g(s)$ is player i while all other facilities are used by more than one player. Thus the cost of each player in the congestion game is $d_{f(s_{-i})}(1) = u_i(s_{-i})$. □

From the above two results it is trivial to prove theorem 8 by using as bijection the function $h_i(s_i) \cup g_i(s_i)$.

To make the above proof clearer we will now provide an example of how to make the Prisoners Dilemma game a congestion game, going through all the steps described in the proofs.

Example 8. *We first break the Prisoners Dilemma game in a coordination and a dummy game as follows:*

c_1, c_2:

	C_2	D_2
C_1	$3, 3$	$0, 4$
D_1	$4, 0$	$1, 1$

c_1^{coord}, c_2^{coord}:

	C_2	D_2
C_1	$3, 3$	$4, 4$
D_1	$4, 4$	$5, 5$

c_1^{dum}, c_2^{dum}:

	C_2	D_2
C_1	$0, 0$	$-4, 0$
D_1	$0, -4$	$-4, -4$

Table 2.2: Break up of the Prisoner's Dilemma game into a coordination and a dummy game

Then we present the bijection from the strategy space of the coordination/dummy game to that of the congestion game:

facilities:

	C_2	D_2
C_1	$X = f(C_1, C_2)$	$Y = f(C_1, D_2)$
D_1	$Z = f(D_1, C_2)$	$W = f(D_1, D_2)$

c_i-congestion game:

	$\{X, Z\}$	$\{Y, W\}$
$\{X, Y\}$	$3, 3$	$4, 4$
$\{Z, W\}$	$4, 4$	$5, 5$

Table 2.3: Bijection h_i from coordination game to congestion game

	C_2	D_2
C_1	$a = f(C_2), x$	$b, x = f(C_1)$
D_1	$a, y = f(D_1)$	$b = f(D_2), y$

facilities

	$\{b, x, y\}$	$\{a, x, y\}$
$\{a, b, y\}$	$0, 0$	$-4, 0$
$\{a, b, x\}$	$0, -4$	$-4, -4$

c_i-congestion game

Table 2.4: Bijection h_i from dummy game to congestion game

And at last the final congestion game that is isomorphic to the prisoner's dilemma game is constructed by the union of the two bijections:

	C_2	D_2
C_1	$3, 3$	$0, 4$
D_1	$4, 0$	$1, 1$

c_1, c_2

	$\{X, Z\} \cup \{b, x, y\}$	$\{Y, W\} \cup \{a, x, y\}$
$\{X, Y\} \cup \{a, b, y\}$	$3, 3$	$0, 4$
$\{Z, W\} \cup \{a, b, x\}$	$4, 0$	$1, 1$

c_i-congestion game

Table 2.5: Prisoner's Dilemma Game and its isomorphic congestion game

2.2.2 Subclasses

From the definition of congestion games we identify two ways to constraint them to a subclass. First by adding some limitation on the strategy space of the players (e.g. network congestion games, load balancing games) and second by adding some limitation on the type of facility cost function d_f (e.g. polynomials, linear, M/M/1).

In the next paragraphs we will present three very well studied subclasses of congestion games. All of these subclasses are derived by restricting the strategy space of the players to some intuitively interesting subset.

Definition 22. *A **Symmetric Congestion Game** is a Congestion Game* $\langle N, F, (A_i)_{i\in[N]}, (d_f)_{f\in F}\rangle$ *where* $A_i = M \subset 2^F$*, i.e. where all players have the same strategy space.*

Definition 23. *A **Network Congestion Game*** $\langle \Gamma, ((s_i, t_i))_{i\in[N]}, (d_e)_{e\in E}\rangle$ *is a Congestion Game* $\langle N, F, (A_i)_{i\in[N]}, (d_f)_{f\in F}\rangle$ *where* $\Gamma = (V, E)$ *is a graph,* $F = E$*,* $A_i = \{s_i - t_i \text{ paths}\}$*. I.e. a Network Congestion Game is a congestion game where facilities are edges in a graph G, players are pairs of nodes in the graph and the strategy space of each player is the set of possible paths between the pair of nodes of the player.*

Definition 24. *A **Load Balancing Game** is a Congestion Game* $\langle N, F, (A_i)_{i\in[N]}, (d_f)_{f\in F}\rangle$ *where the strategy space of each player is exactly the set* F*.*

Load Balancing Games can also be described as Symmetric Network Congestion Games on a graph of parallel links, or as scheduling games, where tasks want to be assigned at machines.

2.2.3 Generalizations

In the basic model of congestion games studied so far, all players account for a congestion of 1 when they use a facility. The case not addressed so far is what happens when each player controlls more than a unit quantity of congestion or generally players don't affect congestion homogeneously.

Definition 25. *A* ***Weighted Congestion Game*** *is a tuple* $\langle N, F, (A_i)_{i\in[N]}, (d_f)_{f\in F}, (w_i)_{i\in[N]}\rangle$*.* $[N], F, A_i, d_f$ *are defined exactly as in congestion games and* $w_i \in \mathbb{N}$ *is a weight for player* i*. Given a strategy profile* $a = (a_1, \dots, a_N) \in A$ *the congestion of facility is* $n_f(a) = \sum_{i\in[N], f\in a_i} w_i$*. The utility of each player is* $u_i(a) = \sum_{f\in a_i} d_f(n_f(a))$*.*

Weighted congestion games are certainly a generalization and many important attributes of congestion games such as the existence of PNE dont continue to hold.

Another, generalization proposed initialy by Miltaich ([Mil96]) and studied recently under the perspective of the equilibrium existence problem is the following:

Definition 26. *A* ***Congestion Game with Player Specific Payoffs*** *is a tuple* $\langle N, F,\ (A_i)_{i\in[N]}, (d^i_f)_{f\in F, i\in[N]}\rangle$*.* $[N], F, A_i$ *are defined exactly as in congestion games and* $d^i_f : \mathbb{N} \mapsto Z$ *is the facility cost that player* i *suffers from facility* f*. The utility of each player in a strategy profile* a *is accordingly defined as* $u_i(a) = \sum_{f\in a_i} d^i_f(n_f(a))$*.*

One important fact concerning Player-Specific Congestion Games that was observed and proved by Monderer [Mon07] is that they are as strong as general strategic form games. In a sense they have the same equivalence that Congestion Games have with Potential Games. To reach this result Monderer introduces the notion of q-potential games. A q-potential game is a game where we can group the players in q groups and define a potential for each group of players, that exactly follows their utility. These types of games are isomorphic to Player-Specific q-Congestion Games which are again games where we can define q groups of delay function vectors of size $|F|$, and players can be grouped into q groups where each players player specific delays on the facilities are exactly the delay functions of some vector. The isomorphism is proved with a slight modification of the Potential-Congestion Games isomorphism presented previously.

In the extreme case where $q = N$ Player-Specific q-Congestion Games coincide with general Player-Specific Congestion Games and q-Potential Games coincide with general strategic form games since the utility of each player is the potential for each of the N groups.

The last generalization that will be studied in this thesis captures the situations where users dont act on their own but in coalitions. They aim not at minimizing their own cost but some aggregate cost of the coalition they belong to. They were first introduced by Harapetyan et al [HTW06] and are defined as follows:

Definition 27. *A* ***Coalitional Congestion Game*** *is a tuple* $\langle N, F, (A_i)_{i\in[N]}, (d_f)_{f\in F}, (P_j)_{j\in[k]}\rangle$*.* $[N], F, A_i, d_f$ *are defined exactly as in congestion games and*

$P = (P_1, \dots, P_k)$ is a partition of the player set $[N]$ in k subsets (coalitions). Each coalition P_i constitutes a coalitional player in the Coalitional Congestion Game that chooses a strategy for each of the players in the coalition. The utility of each coalitional player P_j in a strategy profile $a \in A$ is defined as $u_{P_j}(a) = \sum_{i \in P_j} \sum_{f \in a_i} d_f(n_f(a))$.

An alternative model for coalitional congestion games inspired by the KP-model was introduced by Fotakis et al. [FKS08].

Definition 28. *A **Coalitional Max-Congestion Game** is a tuple $\langle N, F, (A_i)_{i \in [N]}, (d_f)_{f \in F}, (P_j)_{j \in [k]} \rangle$. $[N], F, A_i, d_f, P_j$ are defined exactly as in the Coalitional Congestion Game model. Each coalition P_i constitutes a coalitional player in the Coalitional Max-Congestion Game that chooses a strategy for each of the players in the coalition. The utility of each coalitional player P_j in a strategy profile $a \in A$ is defined as $u_{P_j}(a) = \max_{i \in P_j} \sum_{f \in a_i} d_f(n_f(a))$.*

Fotakis et al. [FKS08] are mainly interested in the case of parallel links for the above model and their results that will be discussed in later sections concern only this case that we will refer to as the Coalitional KP-model.

2.2.4 Alternatives

Nonatomic Network Games aka Selfish Routing

In the models studied so far we assumed that there exist a small number of players in the game. There exist an alternative model based on the assumption that players are infinite and control a the affect of a single player on the congestion of a facility is negligible. This model has been extensively studied in the case of Network Congestion Games and thus we are going to present the model only in that case since it becomes more intuitive.

In Network Congestion Games players want to sent traffic from a source to a destination node on a graph. If we assume that the number of users in the Network is very large and each player controlls a negligible amount of traffic then we are led to the following quite different formulation

Definition 29. *A Nonatomic Network Congestion Game or Nonatomic Selfish Routing Game is a tuple $\langle \Gamma, r, c \rangle$. Γ is a directed graph (V, E). r is a set of requests $r_i : i \in [k]$ for traffic between pairs of nodes (s_i, t_i) called commodities. c is a set of cost functions of the edges of the network. We assume that the players of the game are hidden in the requests and each game controlls an infitesimal amount of each request r_i. Let $\mathcal{P}_i$ denote the set of paths between nodes (s_i, t_i) and $\mathcal{P} = \cup_{i \in [k]} \mathcal{P}_i$. A flow f is feasible for the game if all traffic requests are routed: i.e. $\forall i \in [k], \sum_{P \in \mathcal{P}_i} f_P = r_i$, where f_P is the amount of traffic of commodity i that chooses path P to travel from s_i to t_i. The possible outcomes of the game are the all feasible flows. The congestion of an edge in the network is the amount of traffic routed through that edge $f_e = \sum_{P \in \mathcal{P}, e \in P} f_P$. The cost of an edge incurred on the commodities that use it is $c_e(f_e)$ and the total cost suffered by an amount of traffic* *routed through a path P is $c_P(f) = \sum_{e \in P} c_e(f_e)$. Implicitly this is the cost of the infinite users that participate in creating traffic f_P.*

Although, an implicit reference to the players of a Nonatomic Network Congestion Game was made in the previous definition, most of the time such games are studied just in terms of flows. This is reasonable given that the identity of players is not important in such games.

Given the above special characteristic of the nonatomic model it is not trivial to state what is a Nash Equilibrium of the game. Such equilibria where first proposed in transporation bibliography and specifically where introduced by Wardrop [War52], who defined the conditions of equilibrium.

Definition 30. *A feasible flow f of a Nonatomic Network Congestion Game $\langle \Gamma, r, c \rangle$ is at Nash Equilibrium or equivalently is a nash flow iff for all commodities i, $s_i - t_i$ paths $P_1, P_2 \in \mathcal{P}_i$ with $f_{P_1} > 0$:*

$$c_{P_1}(f) \leq c_{P_2}(f) \tag{2.13}$$

Equivalently, in a Nash Equilibrium all flows travel through minimum-cost paths.

The basic characteristic of existence of PNE in the atomic case carries over to the nonatomic model in the following slightly modified theorem:

Theorem 12. *Every Nonatomic Network Congestion Game admits a Nash flow.*

General Aggregation Function and Bottleneck Games

To present the bottleneck games we will first examine the structure of the utility of players in the basic congestion game model. Recall that in a congestion game the cost of player i in a strategy profile a is $c_i(a) = \sum_{f \in a_i} d_f(n_f(a))$. This cost consists of calculating the local cost of a player at a facility $(d_f(n_f(a)))$ and then aggregating the local costs into a total cost through some aggregation function (sum).

The basic model of Congestion Games is quite general as far as the local cost part is concerned, and allows the freedom to choose among all types of local cost functions d_f. However, the aggregation function is specific and not general. Moreover, most of the characteristics of Congestion Games presented so far are based on the fact that the aggregation function is the sum. Thus by using some other function we are led in a totally different class of games that need to be examined separately. The most well studied case of such a model based on situations observed in communication networks is Bottleneck Games.

Definition 31. *A* ***Bottleneck Game*** *is a tuple $\langle N, F, (A_i)_{i \in [N]}, (d_f)_{f \in F} \rangle$ where $[N], F, A_i$ and d_f are defined as in Congestion Games. The cost of a player in a strategy profile a is $c_i(a) = \max_{f \in a_i} d_f(n_f(a))$.*

Through the use of potential functions it can be proved that Bottleneck Games always admit a PNE and the proof will be given in section 2.3.

Inspired by the proof of the isomorphism of Potential and Congestion Games we will present a class of games that can be similarly proved to be isomorphic to bottleneck games. The overall goal of this approach is to prove that Bottleneck Games are also isomorphic to this class of games and hence prove an equivalence between the two classes.

Definition 32. *A game $G = \langle N, (S_i), (u_i) \rangle$ in normal form is a Max-Potential Game if and only if there exist functions c_i and d_i such that:*

- $\forall i \in [N] : u_i = \max(c_i, d_i)$
- $\langle N, (S_i), (c_i) \rangle$ *is a coordination game*
- $\langle N, (S_i), (d_i) \rangle$ *is a dummy game*

By using the same bijection of the strategy space of players between the two types of games as the one used for the Potential-Congestion Game isomorphism we can conclude the following theorem:

Theorem 13. *Every Max-Potential Game is isomorphic to a Bottleneck Game.*

Proving the opposite direction will lead to equivalence of the two classes and this will give rise to new ways of examining the characteristics of Bottleneck Games.

Colored Resource Allocation Games

Colored Resourced Allocation Games is an enriched model of Congestion Games with general aggregate functions, proposed during the process of the thesis [BPPS09]. The motivation of Colored Resource Allocation Games is to model non-cooperative versions of routing and wavelength assignment problems in multifiber all-optical networks. They can be viewed as an extension of congestion games where each player has his strategies in multiple copies (colors). When restricted to (optical) network games, facilities correspond to edges of the network and colors to wavelengths. The number of players using an edge in the same color represents a lower bound on the number of fibers needed to implement the corresponding physical link.

Definition 33. *A* ***Colored Resource Allocation Game*** *is a tuple $\langle N, F, (A_i), (d_f), W \rangle$ where $[N], F, A_i$ and d_f is defined as in congestion games and $[W]$ is a set of colors. The strategy of each player is a pair $s_i = (a_i, c_i)$ where $a_i \in A_i$ and $c_i \in [W]$. Let s be a strategy profile. The colored congestion of a facility is defined as $n_{f,c}(s) = |\{i \in [N] : f \in a_i, c_i = c\}|$. According to the aggregation function used we have the following two subclasses:*

- *Colored Congestion Games (CCG), with player cost* $c_i(s) = \sum_{f \in A_i} d_f(n_{f,c_i}(s))$
- *Colored Bottleneck Games (CBG), with player cost* $c_i(s) = \max_{f \in A_i} d_f(n_{f,c_i}(s))$

2.3 Existence of Nash Equilibrium

As is proved in the previous section Congestion Games always admit a PNE and this also trivially holds for all their subclasses. However, it is not trivial to deduce the above result to some assertion on the existence of PNE in the superclasses and alternatives of Congestion Games. We will present the latest results on existence of PNE in the classes of Weighted Congestion Games, Congestion Games with Player-specific Payoffs, Coalitional Congestion Games and Bottleneck Games.

2.3.1 Existence in Weighted and Player Specific

When Milchtaich [Mil96] introduced the classes of weighted congestion games and games with player-specific payoffs he showed that in the general case for both classes, there exist instances that don't admit a PNE. Therefore the only hope for existence of PNE is constraining these two classes in smaller subsets and examining the PNE-existence problem there.

There exist two types of constraining the general case in smaller subsets: (1) Examining restricted player strategy spaces, (2) Examining restricted subclasses of delay functions.

In the first direction the most general result was proved by Ackermann et al. [ARV06b]. In order to define the maximal property of the strategy space of the players that consitutes a class of games that always admit a PNE, we first give a brief definition of matroids.

Definition 34. *A tuple $\mathcal{M} = (F, l)$is a matroid if F is a finite set of resources and l is a non-empty family of subsets of F such that, if $I \in l$ and $J \subseteq I$ then $J \in l$, and if $I, J \in l$ and $|J| < |I|$ then there exists an $i \in I \setminus J$ with $J \cup \{i\} \in l$.*

Some of the most useful notations and properties of matroids are the following:

- All sets in l are called independent sets

- We call the independent sets of maximal cardinality the bases of the matroid

- From the second property of matroids all the bases have equal cardinality which we call the rank of the matroid

- If we assign weights to each resource in F then the weight of an independent set is the sum of the weights of the resources it includes

- The base of the matroid with the minimum weight can be found by the greedy algorithm and only problems that can be defined as a minimization/maximization problem on a matroid can be solved by a greedy algorithm

Some intuitive examples of matroids are the following:

Example 9. *(Graphic Matroid) Given a graph $G = (V, E)$, the tuple $M = (E, l)$ where l is the set of forests of the graph is a matroid. If the graph is connected then the bases of the matroid are the spanning trees of the graph. In addition, if we assign a weight to each edge of the graph then the problem of finding the minimum weight base of the matroid is exactly the problem of finding a minimum spanning tree, which is known to be solved by the greedy algorithms of Prim and Kruskal.*

Example 10. *(Linear Matroid) Given a $R \times C$ matrix A the tuple $M = ([R], l)$ where $x \in 2^{[R]}$ belongs to l if and only if the rows in x are linearly independent, is a matroid.*

Based on the definition of matroids we define the following subclass of Congestion Games:

Definition 35. *A Congestion Game $G = \langle N, F, (A_i)_{i\in[N]}, (d_f)_{f\in F}\rangle$ is called a Matroid Congestion Game if and only if for every player $i \in [N]$, A_i is the set of bases of a matroid $\mathcal{M}_i = (F, l_i)$. Additionally we denote by $rk(G) = max_{i\in[N]} rk(\mathcal{M}_i)$ the rank of the game G. The classes of Matroid Weighted Congestion Games and Matroid Player-specific Congestion Games are defined accordingly.*

From the examples of matroids given so far, an intuitively usefull case of Matroid Congestion Games is for example when the strategies of a players are spanning trees on a subgraph of a given graph G (figure 2.4). Another, well studied case is where the strategies of the players are singleton facilities.

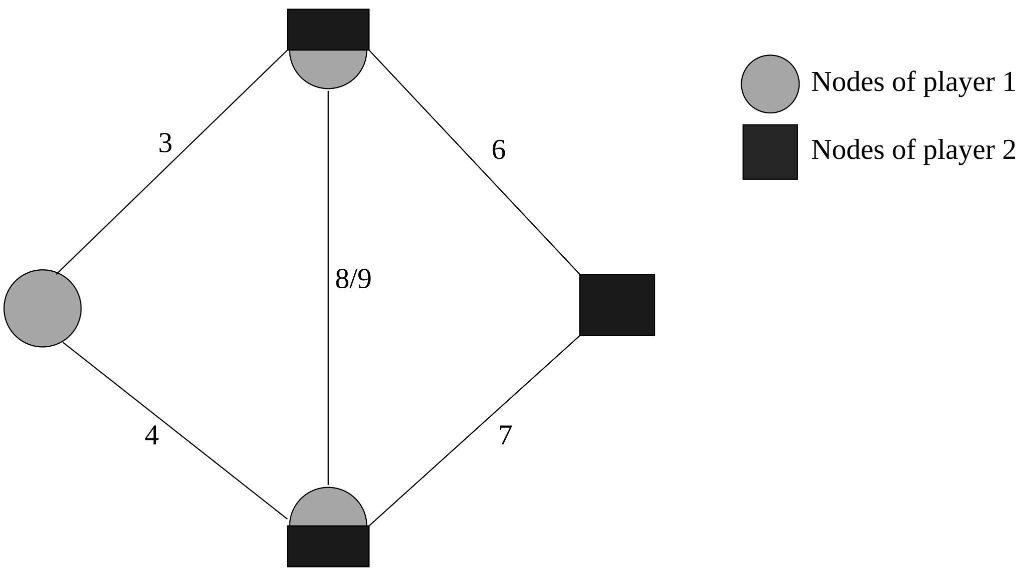

Figure 2.4: Example of a matroid congestion game. Players want to buy a spanning tree on their subgraph.

The basic result of [ARV06b] is the following:

Theorem 14. *Matroid Weighted Congestion Games and Matroid Player-specific Congestion Games admit a PNE when the delay functions are non-decreasing. Moreover, for the player-specific case the best responce dynamics of a game G converge to a PNE in at most $2N^2|F|rk^2(G)$.*

The matroid characteristic of the strategy space of the players is somehow maximal since any game with non-matroid strategy spaces doesn't necessarily admit a PNE.

In the second direction of defining subclasses that admit a PNE, the result for the weighted case that we present here is a combination of the results of three works [FKS05a, PS05, HKM09]:

Theorem 15. *For each game G in the class of Weighted Congestion Games where the delay functions are linear ($\forall f \in F : d_f(x) = a_f x + b_f, a_e, b_e \geq 0$) the following function:*

$$\Phi(s)\frac{1}{2}\left(\sum_{f\in F} d_f(n_f(s))n_f(s) + \sum_{i\in[N]}\sum_{f\in s_i} d_f(w_i)w_i\right) \tag{2.14}$$

is a w-potential for G and thus G admits a PNE.

For each game G in the class of Weighted Congestion Games where the delay functions are exponential ($d_f(x) = \exp(x)$), the following function:

$$\Phi(s) = \sum_{f\in F} \exp(n_f(s)) \tag{2.15}$$

is a w-potential for G and thus G admits a PNE.

Moreover, Weighted Congestion Games admit a w-potential if and only if the delay functions are either only linear or only exponential.

For Player-Specific Congestion Games Gairing et al. [GMT06] prove the following theorem:

Theorem 16. *Player-specific Congestion Games on parallel links where the delay functions are linear without a constant term ($d_{i,f}(x) = a_{i,f}x$), admit the following exact potential:*

$$\Phi(s) = \prod_{i\in[N]} a_{i,s_i} \prod_{f\in F} n_f(s) \tag{2.16}$$

and thus always pocess a PNE.

The most interesting part about the above potential is that it includes only products and no summation and it is the first such potential function. This characteristic is interesting since the above method could give rise to potential functions of similar form for cases where summation style functions have not proven useful.

To our knowledge the above results are the best known so far for the two generalisations of congestion games examined. In this thesis we provide a novel proof for the existence of PNE for matroid weighted congestion games when the delays are non-increasing.

Theorem 17. *Any Matroid Weighted Congestion Game with non-increasing delay functions admits a PNE.*

Proof. Given a strategy profile s, we call a best response s_i* of player i lazy if it can be decomposed into a sequence of strategies $s_i = s_i^0, s_i^1, \ldots, s_i^k = s_i*$ with $|s_i^{j+1} \setminus s_i^j| = 1$ and $c_i(s_i^{j+1}, s_{-i}) < c_i(s_i^j, s_{-i})$ for $0 \leq j < k$.

From the properties of matroids we know that given a matroid $M = (F, l)$ with weights $F \mapsto \mathbb{N}$, a basis $B \in l$ is a minimum weight basis of M if and only if there exists no basis $B^* \in l$ with $|B \setminus B^*| = 1$ and $w(B^*) < w(B)$.

This property leads to the fact at every strategy profile which is not a PNE there exists at least one player i who can descrease her cost by playing a lazy best response s_i^*. Since s_i^* is a lazy best response there exists a sequence of strategies $s_i = s_i^0, s_i^1, \ldots, s_i^k = s_i^*$ such that for every $0 \leq j < k$, $|s_i^{j+1} \setminus s_i^j| = 1$ and

$$c_i(s) = c_i(s_i^0, s_{-i}) > c_i(s_i^1, s_{-i}) > \ldots > c_i(s_i^k, s_{-i}) = c_i(s_i^k, s_{-i}) \tag{2.17}$$

Since from the properties of matroids we know that all the basis have equal cardinality. Therefore, s_i^{j+1} is deduced from s_i^j by exchanging some resource. Let r_j the unique resource that is contained in s_i^j but not in s_i^{j+1} and let r_j^* the unique resource that is contained in s_i^{j+1} but not in s_i^j.

Lets denote with $n_f(s)$ the congestion of a facility which is the sum of the weights of players using facility f in strategy profile s. Moreover, we denote with $z_f(s) = (d_f(n_f(s)), n_f(s))$. We define an ordering on z_f as follows: $z_f(s) \leq z_f(s')$ if and only if $d_f(n_f(s)) < d_f(n_f(s'))$ or $d_f(n_f(s)) = d_f(n_f(s'))$ and $n_f(s) > n_f(s')$. Moreover, $z_f(s) < z_f(s')$ if and only if $z_f(s) \leq z_f(s')$ and $z_f(s) \neq z_f(s')$. Consider the vector of size $|F|$ that contains the pairs $z_f(s)$ of each resource sorted in increasing order.

Since player i is switching facility r_j for r_j^* we know that:

$$n_{r_j^*}(s_i^{j+1}, s_{-i}) = n_{r_j^*}(s_i^j, s_{-i}) + w_i > n_{r_j^*}(s_i^j, s_{-i}) \tag{2.18}$$

Moreover, from the monotonicity of the delay functions we know that:

$$d_{r_j^*}(n_{r_j^*}(s_i^{j+1}, s_{-i})) \leq d_{r_j^*}(n_{r_j^*}(s_i^{j+1}, s_{-i})) \tag{2.19}$$

In addition since each step of the sequence is an improving move:

$$d_{r_j^*}(n_{r_j^*}(s_i^{j+1}, s_{-i})) < d_{r_j}(n_{r_j}(s_i^j, s_{-i})) \tag{2.20}$$

The above inequalities yield:

$$min(z_{r_j^*}(s_i^{j+1}, s_{-i}), z_{r_j}(s_i^{j+1}, s_{-i})) < min(z_{r_j}(s_i^j, s_{-i}), z_{r_j^*}(s_i^j, s_{-i})) \tag{2.21}$$

The last inequality states that whenever an improvement move is made during the sequence $s_i = s_i^0, s_i^1, \ldots, s_i^k = s_i^*$ the vector of $z_f(s)$ strictly decreases lexicographically, hence a PNE is guarranteed to exist and a lazy best response sequence of moves is guarranteed to end at a PNE. □

2.3.2 Existence in Coalitional Congestion Games

We now move on to the problem of PNE existence in the class of Coalitional Congestion games, as defined by Hayrapetyan et al. [HTW06].

In [HTW06] it is proved that for the case of parallel links we can conclude the following theorem:

Theorem 18. *Coalitional Congestion Games in a parallel links network with convex latencies always admit a PNE.*

The above theorem is proved in a way very similar to the proof of existence of PNE in player specific congestion games. We will give here a sketch of the proof:

Proof. (Proof Sketch). We prove existence through induction on the number of players in the game. When a new player is added to the game the best strategy for the coalition, which the new player belongs to, is to put him on a new link without changing the rest of its strategy. The above holds only in the case of convex delay functions. After, a coalition places a player on a link, say s, then only a coalition from link s might have an improving move. Let j be such a coalition. After j removes a player from s then the resulting strategy profile is an equilibrium. So, now coalition j has to place the player it removed to some other link t and only some coalition on that link might have an improving move. This best response can be proved to be finite. The basic fact is that whenever a coalition places a player to a link s during the above best response process then it never removes this player from that link. □

A stronger result is proved in [FKS08] for the case of linear edge delays. When the players in the underlying congestion game are weighted then a coalitions cost is a weighted sum of the costs of its players. Fot this case Fotakis et al. [FKS08] give the following result:

Theorem 19. *Every weighted congestion game with coalitions and linear facility delays is an exact potential game with the following potential:*

$$\Phi(s) = \frac{1}{2}\left(\sum_{f\in F} n_f(s)d_f(n_f(s)) + \sum_{f\in F}\sum_{j=1}^{k} n_f(s_j)d_f(n_f(s_j))\right) \tag{2.22}$$

where $n_f(s)$ is the total weight on facility f and $n_f(s_j)$ is the weight on facility f caused only by coalition j.

For the case of Max-Coalitional Congestion Games presented in [FKS08], it is easy to prove that there always exist a PNE in the case of parallel links (even with dynamic coalitions) and identical delays with the following lexicographic argument: Whenever a coalition C wants to make an improvement move and take its players from resource $r(t)$ to resource $r(t+1)$ then it must hold:

$$L \equiv \max_{j\in C}\{n_{r_j(t)}(s(t))\} > \max_{j\in C}\{n_{r_j(t+1)}(s(t+1))\} \tag{2.23}$$

Therefore if we construct a vector of size equal to the sum of the weights and at each position i we put the number of players that suffer a delay of i then this vector strictly decreases at each step.

2.3.3 Existence in Bottleneck Games

The PNE existence problem in bottleneck games is tuckled with a similar lexicographic argument to that in Max-Coalitional Congesiton Games. The following theorem can be found in [BO06], but we provide the proof here under our notation since it is succinct and quite usefull.

Theorem 20. *Bottleneck Games always admit a PNE.*

Proof. Consider the vector $B(s)$ that contains all the elements of $\{c_i(s)|i \in [N]\}$ sorted in non-increasing order. Let $B_i(s)$ denote the ith element of $B(s)$. Define with $\mathbb{B} \subseteq \mathbb{R}^N$ the set of all vectors $B(s)$ that correspond to a strategy profile: $\mathbb{B} = \{B(s)|s \text{ is a strategy profile}\}$.

We define the lexicographic ordering on that finite set of vectors: $B(x) < B(y)$ iff there exists an $i \in [N]$ such that $B_j(x) = B_j(y)$ for all $j \in [1, i-1]$ and $B_i(x) < B_i(y)$. Moreover, we say that $B(x) = \min \mathbb{B}$ if there is no $B(y) \in \mathbb{B}$ such that $B(y) < B(x)$. Since $\mathbb{B}$ is finite we know that such a minimum exists.

Let $B(s) = \min \mathbb{B}$. We argue that s is a PNE for the bottleneck game. By definition of $B(s)$ we know that for each $i \in [N]$ their cost is minimized with respect to all vectors $B(x) \in \mathbb{B}$ that satisfy $B_j(x) = B_j(s)$ for $j \in [1, i-1]$. Thus, a player can make an improvement move only if he modifies a player with larger cost. Suppose that player i increases the cost of some player with already larger cost. For this to happen player's i cost must also increase to that amount, so it cannot be an improvement move. So the only possibility is that player i decreases the cost of players with larger cost. But since we are talking about an improvement move for player i, such a modification will lead to a vector $B(x) < B(s)$ which contradicts the initial assumption that $B(s) = min\mathbb{B}$. Therefore, no player can make an improvement move. □

The vector defined in the last proof can also be viewed as a generalized ordinal potential of bottleneck games. Consider an arbitrary initial strategy profile s and the corresponding vector $B(s)$. Let player i make an improvement move. The only elements of the vector that could potentially become affected are the elements with cost equal to the cost of player i (if i had a common resource with them) and the ones with smaller bottleneck (player i moved on the resources of a player with smaller bottleneck or left from resources of a player with smaller cost). The last ones strictly decrease. So, the smaller ones that become even smaller can only decrease lexicographically the vector. The only ones that might increase the vector are those whose cost increases. But those costs cannot become larger than the new cost of player i which is strictly smaller than the old one. Thus, the new vector is definitely lexicographically smaller than the old one.

2.3.4 Discussion on Methods of PNE Existence

Based on the proofs studied in previous paragraphs we can identify three major methods that can be used to prove PNE existence in congestion-like games.

First we have the **potential method**. In this technique we try to prove that a function $P : \times_i S_i \mapsto \mathbb{R}$ is either an exact or a weighted potential for the game. This automatically leads to PNE existence and gives immediately a lower bound on the convergence time of best response paths to a PNE. However, the class of games that admit such potentials is quite small and we cannot hope to achieve many results with this technique. On the other hand it might be our first attempt whenever we tuckle with a PNE existence problem since proving such a characteristic for our class of games tells us a lot about the nature of our game.

Second we have the **lexicographic method**. In this technique we try to define an ordered finite space and a mapping from the strategy space to that space. Then we try to prove that whenever a player makes an improvement move the element in our ordered space that corresponds to the new strategy profile is strictly smaller than that in our initial strategy profile.

Third we have the **inductive method**. In this technique we try to prove the equilibrium existence by using induction on some parameter of the game. For example it could be players of a coalition used in coalitional congestion games or tokens of a strategy used in matroid player specific congestion games. In order to prove the induction step, most of the times, we use some of the previous techinques so as to show that when a new element is added there exist some best response sequence that leads the game again to a PNE.

Whenever we want to solve an equilibrium existence problem in some new model it is interesting to see what could be derived using the above techniques and then move on to some ad hoc argument. Of course the small number of results so far leads us to think that there might exist some other more general method that would enable new results on the problem.

2.4 Complexity of Computing Pure Nash Equilibria

In this section we investigate the complexity of computing a PNE in a congestion game-like representation of a game. It is trivial to notice that in a normal form representation of a game, finding a PNE requires just to check every entry of the payoff matrices, which is polynomial for that input. But when a strategic game is given in the succinct form of a congestion game or its alternatives then the complexity of computing a PNE arises as a major issue.

2.4.1 The Class PLS

The fundamendal results in the complexity of the basic model of Congestion Games where given by Fabrikant et al. in [FPT04], where they prove among others that finding a PNE in the class of Congestion Games is PLS-complete. So we will first give the basics of the complexity class PLS (Polynomial Local Search).

PLS is a class consisting of search problems, similar to FNP, FP, PPAD and other classes described in the introduction chapter. A problem L in PLS can be either a maximization or a minimization problem and is specified as follows: L has a set D_L of instances, which are a polynomial-time recognizable subset of $\{0,1\}*$. For each instance x we have a finite set $F_L(x)$ of solutions, which can also be considered as $\{0,1\}*$ strings of polynomially bounded length $p(|x|)$. For each solution $s \in F_L(x)$ we have a non-negative integer cost $c_L(s,x)$ and a subset $N(s,x) \subseteq F_L(x)$ called the neighborhood of s. What remains for L to be in PLS is the existence of three polynomial-time algorithms A_L, B_L and C_L. Algorithm A_L, given $x \in D_L$ produces a particular standard solution $A_L(x) \in F_L(x)$. Algorithm B_L, given an instance x and a string s, determines whether $s \in F_L(x)$ and if so computes $c_L(s,x)$. Algorithm C_L, given an instance x and a solution $s \in F_L(x)$, has two possible types of outputs: If there is any solutions $s' \in N(s,x)$ with better cost than that for s, C_L outputs such a solution. Otherwise it reports that no such solutions exist and hence s is locally optimal.

Example 11. *A problem that will be proved usefull in proving PLS completeness of finding a PNE is MAX CUT: Given a weighted graph, find a partition of its vertices, into two possibly unequal parts, so that the weight of the cut cannot be increased by moving a vertex from one side to the other. To illustrate better the definition of a PLS problem lets make the correspondence with MAX CUT: The set of instances x are the set of binary representations of the adjacency matrix of a graph. The set of solutions is the set of binary representations of partitions of nodes of the graph. A neighborhood of solution is all the solutions that can be derived by moving a node from one partition to the other. The cost of a solution is the sum of the weights of the cut of the partition. Algorithm A_L outputs an initial partition of the graph. Algorithm B_L given a graph and a partition, determines if the partition is valid and if so returns the cost of the cut. Algorithm C_L given a graph and a partition checks if a partition produced by moving a node from one partition to the other, has a better cost and if so retuns such a partition. Otherwise returns false.*

The definition of a PLS problem gives immediately a trivial algorithm for finding a solution that we call the standard algorithm:

1. Given x, use A_L to produce a starting solution s
2. Until current solution is not locally optimal: Apply C_L to current solution s. If C_L gives a better cost neighbour s' set $s = s'$.

Since the solutions of the instance are finite the above algorithm must halt at a local optimum. However, for several problems the above trivial algorithm takes exponential time to halt.

Generally, the class PLS is a subclass of TFNP. Any problem in PLS is a search problem where a solution is guarranteed to exist. Moreover, the existence of a solution is guarranteed because of the fact that a local optimum always exist in a finite totally ordered space. Equivalently, the existence of a solution is based on the following non-constructive theorem:

Every DAG has a sink.

The graph constructed by the moves made by the standard algorithm form a DAG and therefore we know that an endpoint in that sequence of moves will sometime halt.

As a subclass of TFNP based on a non-constructive proof it is an open and important problem of how difficult are the problems it includes. In other words how difficult is local search. It certainly holds that $FP \subseteq PLS \subseteq FNP$, but two major questions are $FP = ?PLS = ?FNP$. We know that PLS$\neq$FNP unless $NP = co-NP$. On the other side proving that $FP = PLS$ would mean that there exists an algorithm that doesn't need to follow the whole path produced by the standard algorithm but finds a way to shortcut in order to reach a local optimum. A similar success story was that of linear programming. In linear programming the simplex method is very similar to the standard algorithm following an improvement path on nodes of a polytope until it finds an optimum, which can be exponential. However, the ellipsoid method makes a shortcut visiting only the optimum node of the polytope in polynomial time. Intrigued by linear programming one could try to find a shortcut algorithm to solve the standard algorithm problem: Find the local optimum that the standard algorithm produces. But the standard algorithm problem is NP-hard for several PLS problems and this is the reason why in the definition of PLS we seek an arbitrary local optimum.

Generally, when trying to identify how difficult are the problems of a class we want to find complete problems. Therefore, we now move on to defining what is a PLS reduction.

We say that a problem L in PLS is PLS-reducible to another, K, if there are polynomial-time computable functions f and g such that: (a) f maps instances x of L to instances $f(x)$ of K, (b) g maps (solution of $f(x)$,x) pairs to solutions of x and (c) for all instances x of L, if s is a local optimum for instance $f(x)$ of K, then $g(s, x)$ is a local optimum for x.

The above definition of PLS reduction has the two desirable properties of reductions:

- If K is PLS-reducible to L and L is PLS-reducible to J then K is PLS-reducible to J.
- if K is PLS-reducible to L and we can find locally optimal solutions to L in polynomial time, then we can also find locally optimal solutions for K in polynomial time.

Having defined the notion of PLS-reduction, PLS-completeness can be defined accordingly.

The above PLS-reduction although keeps the two desirable properties of common reductions it doesn't preserve any properties of the efficiency of the standard algorithm. An important concept related to the standard algorithm is the transition graph. The transition graph is the subgraph of the neighborhood graph such that every edge goes from a node with worse to a node with better cost. This graph is a DAG and the costs of the nodes are a topological order of the graph. Moreover, the sinks of the transition graph correspond to local optima of the initial problem. The standard algorithm chooses an initial vertex in the transition graph and follows a path to a sink. The shortest path from a vertex v to a sink is called the height of v and corresponds to a lower bound of the standard algorithm. Moreover, the largest height of the transition graph (called the height of the graph) corresponds to the worst case running time of the standard algorithm. Thus if the height of the graph is exponential then so is the complexity of the standard algorithm.

Under the previous perspective it would be helpful to have a PLS-reduction from a problem I to problem J such that the height of the transition graph of J would be at least as large as that of I and if the standard algorithm problem is NP-hard for I then it is also for J. This type of PLS-reduction is called tight and was defined by Schaffer et al. [SY91].

Definition 36. *Let P, Q be PLS problems and let (f,g) be a PLS-reduction from P to Q. We say that the reduction is tight if for any instance I of P we can choose a subset $\mathcal{R}$ of feasible solutions for the image instance $J = f(I)$ of Q such that the following properties are satisfied:*

- *$\mathcal{R}$ contains all local optima of J*
- *For every feasible solution p of I, we can construct in polynomial time a solution $q \in \mathcal{R}$ of J such that $g(q, I) = p$*
- *Suppose that the transition graph of J, contains a directed path $q \to q'$, such that $q, q' \in \mathcal{R}$, but all internal path vertices are outside of $\mathcal{R}$ and let $p = g(q, I)$ and $p' = g(q', I)$ be the corresponding feasible solutions of I. Then either $p = p'$ or the transition graph of I contains an edge from p to p'.*

In the initial works in PLS [JPY88, SY91, PSY90] many interesting problems where proved to be PLS-complete and most of them through tight PLS-reductions. The MAX CUT problem defined previously is PLS-complete. Some other PLS-complete problems are the following:

- FLIP: Given a boolean circuit with n input and n output bits, find an input such that now greater/smaller output can be produced by flipping one bit of the input.
- WEIGHTED SAT: The input is a boolean formula in CNF and weights for each clause. Given an assignment of the variables the cost of the assignment is the sum of the weights of the satisfied clauses. Find an assingment to the variables such that no greater assignment can be produced by flipping the value of a variable

- WEIGHTED kSAT (even WEIGHTED 2SAT): Similar to the WEIGHTED SAT but where each clause contains at most k literals
- WEIGHTED NAE kSAT: The cost and the neighborhood of an assignment are defined as in WEIGHTED kSAT. A clause consists of at most k literals and is satisfied if its literals dont all have the same value.
- POS NAE kSAT (even POS NAE 3SAT): Similar to the WEIGHTED NAE kSAT but with the restriction that all literals in the clauses are positive
- SWAP GRAPH PARTITIONING: You are given a weighted undirected graph with even number of vertices. A feasible solution is a partition of its nodes into two sets V_1, V_2 of equal size and the cost of the solution is the weight of the cut of the partition. Find a partition such that no greater/smaller solution can be achieved by swapping one node from partition V_1 with a node from partition V_2.

From the abundane of problems in PLS it is well claimed that it is a quite extensive class with very interesting problems.

2.4.2 Complexity of basic model

From the definition of PLS it could be observed that the problem of finding a PNE in any class of games that admit a polynomial-time computable generalized potential function is in PLS.

If a class of games admits a generalized potential function then the Nash Dynamics graph where the cost of each node is the generalized potential of the corresponding strategy profile, is the transition graph of the PLS problem of finding a PNE. We can also accordingly define the standard algorithm of the PNE problem which is just following a Best Response path in the Nash Dynamics graph. In fact as proved by Fabrikant et al. [FPT04] all PLS problems can be cast as games with a polynomially computable generalized potential and thus computing a PNE in the broad class of games that admit a polynomially computable generalized potential is PLS-complete.

We will now give a PLS-completeness result for two major classes of congestion games. The following theorem is part of the results presented by Fabrikant et al. [FPT04]:

Theorem 21. *It is PLS-complete to find a PNE in the following classes of games:*

i. Congestion Games

ii. Symmetric Congestion Games

Proof. For (i) we will present a PLS-reduction from MAX CUT to finding a PNE in a Congestion Game. Given an instance of MAX CUT we construct a congestion game as follows. For each edge e of weight w, we have two facilities $f_e^{(A)}$ and $f_e^{(B)}$, with cost 0 if used but only one player and cost w if used by more players.

The players correspond to the nodes of the Graph. Player v has two strategies: one strategy contains all $f_e^{(A)}$ for edges e incident to v, and another contains all $f_e^{(B)}$'s for the same edges. The first strategy corresponds to assigning v to the set A and the latter strategy corresponds to assigning v to B. This one-to-one correspondence between the assignments of the nodes in the MAX CUT instance and the strategies of the players in the congestion game has the property that the local optima of the MAX CUT instance coincide with the PNE of the congestion game. Hence, our construction is a PLS-reduction from MAX CUT to finding a PNE in congestion games. In fact it is trivial to observe that the reduction is tight and hence the standard algorithm problem for finding a PNE in congestion games is NP-hard (find the PNE that the standard algorithm computes when it starts from a specific initial strategy profile).

For (ii) we will present a PLS-reduction from the general to the symmetric case. Suppose we are given a general congestion game with strategy spaces $S_1, \ldots, S_N \subseteq 2^F$. We extend the facility set F with additional facilities $f_1, \ldots, f_N$ with cost 0 if used by one player and cost M, otherwise (where M is a large number). For $i \in [N]$, let $S_i' = \{s \cup \{f_i\} | s \in S_i\}$. The symmetric game has the common strategy space $S = S_1' \cup \ldots \cup S_N'$. If M is chosen sufficiently large then any equilibrium of this game has one player using a strategy from S_i'. This property yields an obvious correspondence between the PNEs of the symmetric and the assymetric game, and, hence, gives a PLS-reduction. □

We will now move on to proving PLS-completeness for the class of Asymmetric Network Congestion Games. The methodology that we will use was proposed by Ackermann et al. [ARV06a] as an alternative of the proof first presented in [FPT04].

First we will prove PLS-completeness of the following subclass of Congestion Games called Threshold Congestion Games:

Definition 37. *In a Threshold Congestion Game facilities F are divided into two disjoint subsets F_{in} and F_{out} with $|F_{out}| = N$. Each player has two strategies, namely $s_i^{out} = \{f_i\}$ for a unique facility $f_i \in F_{out}$ and a strategy $s_i^{in} \subseteq F_{in}$. Each $f_i \in F_{out}$ is used only by player i and no other player. From the above definition in a given strategy profile s of the game, strategy s_i^{in} is a best response for player i if $c_i(s_i^{in}, s_{-i}) \leq d_{f_i}(1)$. Thus, the delay $d_{f_i}(1)$ is a threshold indicating whether i plays strategy s_i^{in} or not, in other words whether it interferes with other players or not. We denote by $T_i = d_{f_i}(1)$ the threshold of player i.*

We can also define the class of k-threshold congestion games as threshold congestion games where for each facility $f \in F_{in}$ there are at most k players i with $f \in s_i^{in}$.

Theorem 22. *The problem of finding a PNE of a 2-threshold congestion game Γ is PLS-complete.*

Proof. We will present a reduction from the MAX CUT problem on a weighted graph $G = (V, E, w)$. For each vertex v of the graph we will denote with $w_v =$

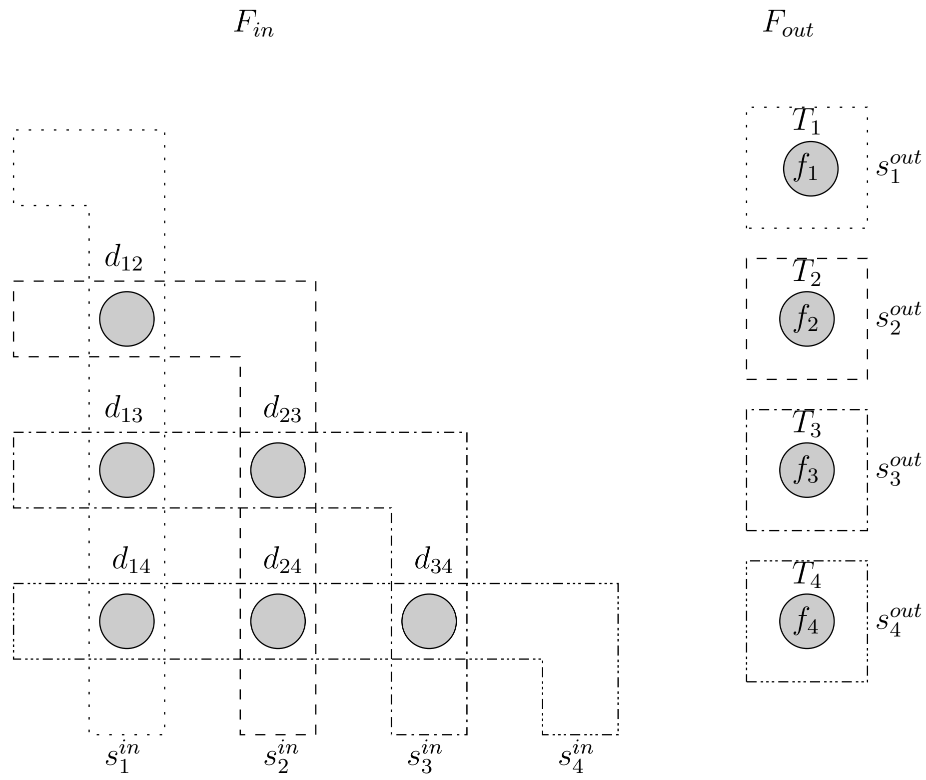

Figure 2.5: Example of a 2-threshold game with 4 players.

$\sum_{e\ inc\ v} w_e$. From G we construct the following 2-threshold congestion game. For every edge $e \in E$, there is a facility $f_e \in F_{in}$ with delay $d_{f_e}(1) = 0$ and $d_{f_e}(2) = w_e$. For every vertex $v \in V$ there is a facility $f_v \in F_{out}$ with delay function $d_{f_v}(1) = \frac{w_v}{2}$. For each vertex v we have a player p_v with strategies $s_{p_v}^{out} = \{f_v\}$ and $s_{p_v}^{in} = \{f_e | e\ inc\ v\}$.

Given a PNE s of the 2-threshold congestion game we can construct a local optimum of the initial MAX CUT problem. For each vertex v if player p_v chose strategy $s_{p_v}^{out}$ then assign v to partition V_1 else to partition V_2. It is easy to observe that the sum of the weights of the edges incident to a vertex v with the other endpoint in V_2 ($w(v, V_2)$) is exactly $c_{p_v}(s_{p_v}^{in}, s_{-i})$. Moreover, since s is a PNE if a player chose strategy $s_{p_v}^{out}$ then it must hold $c_{p_v}(s_{p_v}^{out}, s_{-i}) \leq c_{p_v}(s_{p_v}^{in}, s_{-i})$ or equivalently $w_v/2 \leq w(v, V_2)$. Therefore if vertex v of V_1 was flipped to V_2 it would contribute less in the cut. In addition if a player chose strategy $s_{p_v}^{in}$ it must hold $c_{p_v}(s_{p_v}^{in}, s_{-i}) \leq c_{p_v}(s_{p_v}^{out}, s_{-i})$ or equivalently $w(v, V_2) \leq w_v/2$. Therefore if a vertex v of V_2 was flipped to V_1 it would again contribute less to the cut, which ends the proof that the constructed partition is a local optimum of the initial MAX CUT problem. □

We will now use the PLS-completeness of 2-threshold congestion games to prove that finding a PNE in Asymmetric Network Congestion Games is PLS complete even if we restrict to non-decreasing, linear delay functions.

Theorem 23. *Computing a PNE for an Asymmetric Network Congestion Game with non-decreasing, linear delay functions is PLS-complete.*

Proof. We will present a reduction from 2-threshold congestion games to asymmetric network congestion games. In fact w.l.o.g we will assume that in the 2-threshold game there exists a unique resource $f_{i,j} \in F_{in}$ contained in s_i^{in} and s_j^{in}.

The directed graph of the asymmetric network congestion game is an NxN grid in which edges are directed downwards and from left to right. The source nodes of the players are the nodes in the first column, $s_1, \dots, s_N$ from top to bottom. The end nodes are the nodes in the last row $t_1, \dots, t_N$ from left to right. For every player $i \in [N]$ we add an additional edge from s_i to t_i. Because of the direction of the edges of the grid this additional edge can be used only by player i.

We now want to assign delays to the edges such that there exist only two undominated strategies for player i: the shortcut edge (s_i, t_i) or the row-column path from s_i to t_i (the path from s_i along edges in row i until column i and then along the edges of column i to t_i. All other paths will be assigned such high delays that they will be certainly dominated by these two strategies.

We achieve the above goal by assigning delay 0 to all edges pointing downwards and delay $i \cdot D$ to all edges of row i, where D is a large integer. Furthermore, for now the shortcut edge has a delay of $D \cdot i \cdot (i-1)$. The delays of the only two undominated strategies of each player are so far identical.

Now we assign additional delay functions to the nodes of the grid. This can be done by replacing nodes with gadgets such that whenever a path passes throught that gadget is is incured an additional cost equal to the delay we will assign to the nodes. For $1 \leq i < j \leq N$ the node in column i and row j is identified with facility $f_{i,j} \in F_{in}$ from the 2-threshold game and we assume that the node has the same delay function as that of $f_{i,j}$. This way, the row-column path of player i corresponds to s_i^{in} of the threshold game. Furthermore, we increase the delay of the shortcut edge (s_i, t_i) from $D \cdot i \cdot (i-1)$ to $D \cdot i \cdot (i-1) + T_i$ and therefore the shortcut edge corresponds to strategy s_i^{out} of the 2-threshold game.

From the above construction, by choosing D to be sufficiently large we can ignore all other (s_i, t_i) paths except the shortcut and row-column paths. Moreover, those two strategies and their delays are isomorphic to the strategies and cost functions in the 2-threshold game in the sense that the PNEs of the two games coincide. Therefore, the above function is a PLS-reduction. Moreover, since all edges in the graph are used by at most two players their delays can be described in terms of a linear function. $\square$

An sketch of the network congestion game that the 2-threshold game of figure 2.5 reduces to is depicted in figure 2.6.

With a slight modification of the above proof we can also prove that finding a PNE in Asymmetric Network Congestion Games on undirected graphs is also PLS-complete and the reader is redirected to [ARV06a] for the proof.

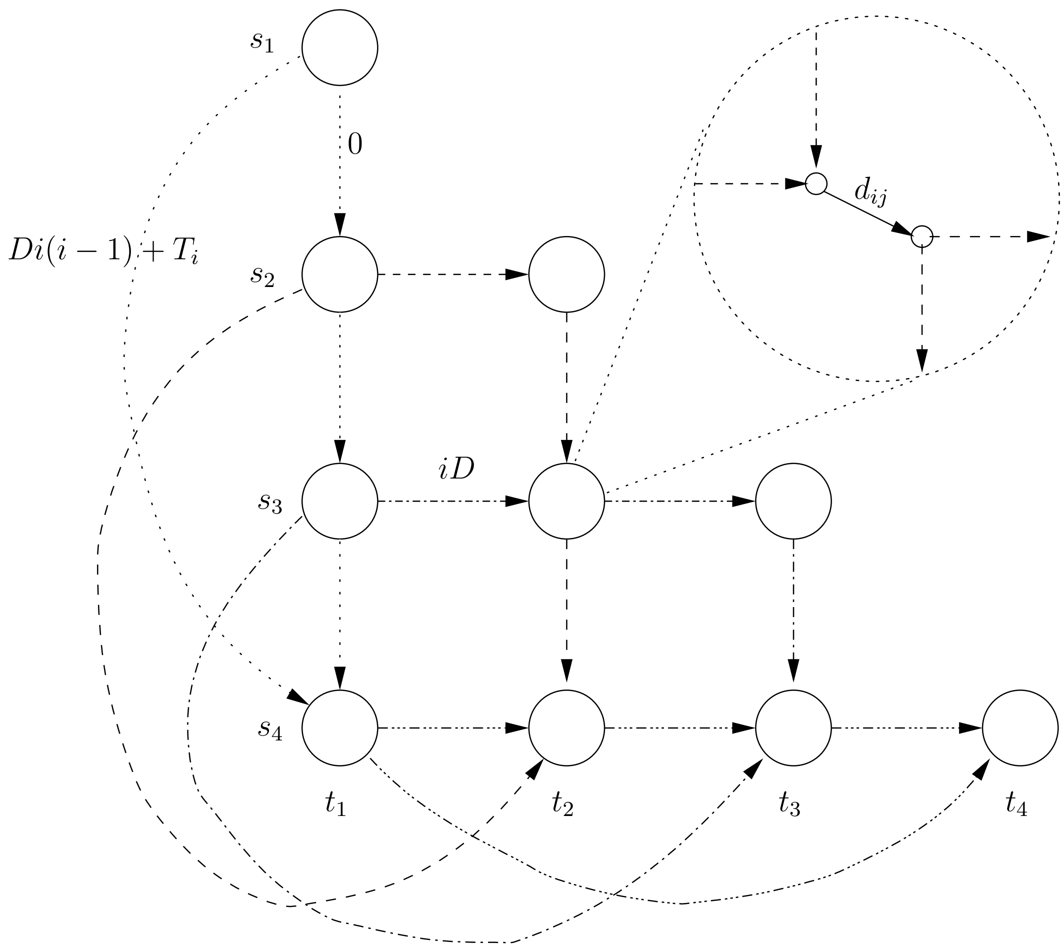

Figure 2.6: Example of a reduction from a 2-threshold game to a Network Assymetric Congestion Game

The above PLS-completeness results state that as far as we don't find a general polynomial time algorithm for solving local search optimization problems, we don't have a chance of computing a PNE in polynomial time. Moreover, since both reductions presented above are tight PLS-reductions the standard algorithm problem for PNE is NP-hard and the worst case running time of the standard algorithm is exponential.

Thus finding a PNE for the above general classes of congestion games is under some perspective intractable and a best response sequence might take exponential time to reach a PNE. However, we might hope that when restricting to smaller subclasses of congestion games we could derive better complexity results, and this will be the subject of the next section.

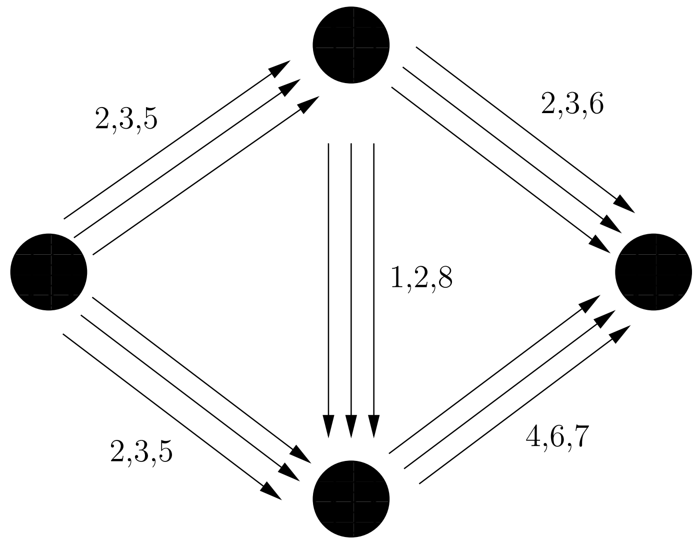

Figure 2.7: Example of a reduction from finding a PNE in a Symmetric Network Congestion Game (fig. 2.1) to a min-cost flow problem. The comma separated values represent the costs of each edge. All edges have unit capacity.

2.4.3 Tractable Subclasses

Theorem 24. *There is a polynomial time algorithm for finding a PNE in Symmetric Network Congestion Games with non-decreasing delay functions.*

Proof. We will reduce the problem to a min-cost flow problem as follows. Each edge e is replaced by n parallel edges $e_1, \ldots, e_N$ between the same nodes. Edge e_i is assigned cost $d_e(i)$, for $1 \leq i \leq N$. All edges have capacity 1. Observe that if a min-cost flow solution uses some of the edges $e_1, \ldots, e_N$ then it sends an integral amount of flow along theses edges. If it sends k units of flow along these edges, then it uses the k cheapest edges. W.l.o.g. these are edges $e_1, \ldots, e_k$ as the delay functions are non-decreasing. Thus the cost for sending the flow along these edges is $d_e(1) + \ldots + d_e(k)$, which corresponds to Rosenthal's potential for edge e if k players are using it. Consequently, we can translate the optimal solution of the min-cost flow problem into a state of the congestion game whose potential corresponds to the cost of the flow. Hence, the min-cost flow solution corresponds to a PNE that globally minimizes Rosenthal's potential function. □

Despite the fact that computing a PNE in Symmetric Network Congestion Games is in P there exist instances of games in that class that have strategy profiles with an exponential distance to any PNE in the transition graph. Therefore the standard algorithm for the class of Symmetric Network Congestion Games still takes exponential time in the worst-case.

Another way of identifying subclasses where finding a PNE is easy, is by examining the combinatorial structure of the strategy spaces of players as we did in the PNE existence problem for weighted and player-specific congestion games. In this direction a positive result came from Ackerman et al. [ARV06a].

Theorem 25. *Let G be a matroid congestion game. Then players reach a PNE after at most $N^2 mrk(G) \leq N^2 m^2$ best response improvement steps, where $m = |F|$.*

Proof. The proof is based on a property of matroids that was also used in theorem 17. Let L be a list of all values $d_f(i)$ for $i \in [N]$ and $f \in F$ sorted in a non-decreasing order. For each facility f we define an alternative cost $\widetilde{d_f}(i)$ which equals the rank of $d_f(i)$ in the sorted list L. Equal cost values receive the same rank. We will now prove that whenever a player makes a best response move then his cost with respect to the alternative cost functions $\widetilde{d_f}$ also decreases. From the properties of matroids we know that any best response move of a player can be decomposed into a sequence of steps where at each step the player exchanges a facility f in his strategy with another facility f', such that the cost of the player doesn't increase. Lets call s_k the strategy profile at step k during this sequence of moves. We know that at each step $d_f(n_f(s_k)) \geq d_{f'}(n_{f'}(s_{k+1}))$. If $d_f(n_f(s_k)) > d_{f'}(n_{f'}(s_{k+1}))$ then it also holds that $\widetilde{d_f}(n_f(s_k)) > \widetilde{d_{f'}}(n_{f'}(s_{k+1}))$ since $d_f(n_f(s_k))$ occurs after $d_{f'}(n_{f'}(s_{k+1}))$ in the sorted list L. Moreover, if $d_f(n_f(s_k)) = d_{f'}(n_{f'}(s_{k+1}))$ then the same holds for the alternative costs too. Furthermore, since the cost of a player strictly decreases after the whole sequence of steps there must exist a step where the alternative cost strictly decreases. Hence whenever a player makes a best response move his cost with respect to the alternative costs strictly decreases. Now if we consider Rosenthal's potential with respect to the alternative cost, we know from the previous argument that whenever a player makes a best response the potential decreases. Since there are at most Nm different cost values, $\widetilde{d_f}(i) \leq Nm$ for all facilities $f \in F$ and values $i \in [N]$. Consequently:

$$\widetilde{\Phi}(s) = \sum_{f \in F} \sum_{i=1}^{n_f(s)} \widetilde{d_f}(i) \leq \sum_{f \in F} \sum_{i=1}^{n_f(s)} Nm \leq N^2 mrk(G) \tag{2.24}$$

where the last inequality holds since each player occupies at most $rk(G)$ resources. Since the above potential decreases with every best response step and it cannot drop below zero, the theorem is proved. □

The matroid characteristic of the strategy space of the players is again a maximal condition for efficient convergence of the best response dynamics. It is maximal under the perspective that if we are only examining the strategy spaces of the players and we don't impose another restriction on the game then if the strategy spaces are allowed to be non-matroid the resulting class of games contains instances of games where the best response dynamics can have an exponentially long best response sequence.

The essence of matroid strategy spaces that keep the best response sequences polynomially long is the fact that any best response can be decomposed in a sequence of steps where a facility of the current strategy is exchanged by another facility. Ackerman et al. [ARV06a] prove that if a strategy space is non-matroid then there exist three resources a, b, c used by a player such that keeping constant the delays of the rest of the resources for any choice of delays of a, b and c either a will be in the best response or b and c together. They call this property the $(1 - 2)$-exchange property in the sense that a user is obliged sometime during a best response sequence to exchange one facility of his current strategy by two facilities or vice versa. Based on this property they devise, congestion games with an exponentially long best response sequence.

Some, more ad hoc, polynomial-time algorithms for computing PNEs have been proposed by Fotakis et al. [FKS05b] and Gairing et al. [GLMM04] for even more restricted subclasses of Congestion Games.

In particular, Fotakis et al. prove that a very trivial algorithm called Greedy Best Response (GBR) computes a PNE in time $O(N|F| \log |F|)$ in the case of symmetric series-parallel graphs. GBR inserts users one after the other into the game placing them at their best response strategy at the time of insertion. This algorithm also works for weighted instances (where it inserts players in non-increasing weight) under the extra assumption that given an initial flow on the network all players have the same set of Best Response strategies.

2.4.4 Approximate Solutions

Since the problem of finding a PNE in the general class of Congestion Games seems hard to tuckle one could turn to some approximate solution.

We will first deal with the problem of finding an a-approximate PNE in the sense of the ϵ-Nash presented in the introduction. An a approximate PNE is a strategy profile where no player can decrease his cost by a factor of a when changing his strategy unilateraly. The major result in settling the complexity of computing an a-appproximate PNE was given by Skopalik and Vocking in [SV08] and mainly comprises in the following theorem:

Theorem 26. *Finding an a-approximate equilibrium in a congestion game with positive and increasing delay functions is PLS-complete, for every polynomial-time comptable $a > 1$.*

The proof of theorem 26 is quite novel and possibly the first gap introducing PLS-reduction in bibliography. The high level idea of the proof is to make a PLS-reduction of a minimization instance of the FLIP problem to a congestion game where the delays of the different strategies of each player at any strategy profile differ by at least a factor of a. From the PLS-reduction we know that all PNEs of the congestion game are locally optimal inputs of the initial FLIP problem and since in the congestion game all PNEs are a-approximate PNEs, the same holds for a-approximate PNEs.

The basic block of the reduction from a FLIP problem to a congestion game is constructing a game that somehow simulates the semantics of a cicuit S of 2-input NAND gates that implements boolean function $f_S(u)$. We will refer to the congestion game as $CG(S)$. The construction is quite simple but ingenious. We refer to the two inputs of the gates as a, b. Let k denote the number of gates in S. Let $g_1, \ldots, g_k$ denote the gates of the circuit in reverse topological order (a gate is connected only with gates of smaller index). Gate g_i is associated with a player G_i that has a zero- and a one-strategy. For $1 \leq i \leq k$, the zero-strategy of player G_i contains the resources $Bit0a_i$ and $Bit0b_i$. Both of these resources have delay 0 when allocated by one player and delay a^{2i} if two or more players are on that resource. The one-strategy of player G_i contains the resource $Bit1_i$ with delay 0 when allocated by at most two players and delay a^{2i} otherwise. Let $j \in \{1, \ldots, i-1\}$ denote an index of a gate g_j with gate g_i as input. Then the one-strategy of g_i additionally contains $Bit1_j$ and the zero-strategy of g_i additionally contains the resource $Bit0a_j$ if g_i corresponds to input a and the resource $Bit0b_j$ if g_i corresponds to input b. The inputs of the circuit also correspond to players with a zero- and a one-strategy. The strategies of these players contain the bit resources of the gates to which they are connected in the same way as the gate players.

If we assume that the input players prefer some specific strategy over the other for some input then we can state the following statement: Fix any input vector u for S. The delay differences between the zero- and the one-strategy of any gate player in any state of $CG(S)$ is more than a. $CG(S)$ has a unique equilibrium in which the output player uses strategy $f_S(u)$.

An example of a construction of a congestion game that simulates a boolean circuit can be seen in figure 2.8.

Thus we have seen how to contruct a congestion game that in equilibrium simulates a boolean circuit computation. The major problem of the reduction arises when we try to implement a feedback circuit. The above construction cannot implement a feedback circuit because the delay difference of the strategies of the output players is significantly smaller than that of the input player. Hence it is impossible for the output player to affect the input players by sharing some resources.

A first attempt to make the reduction would be to construct a circuit S' that when the input x is a local optimum in S then x is a fixed point in S'. Thus we would have to construct a congestion game that in equilibrium, not only simulate the circuit semantics but also the inputs correspond to fixed point of S'. With this approach we are stuck at the feedback problem since there is no way for the output to affect the input.

The reduction proposed by Skopalik and Voecking creates a congestion game that simulates a whole processor system (circuits, controller, clock). Let n be the size of the input and m the size of the output. The clock is represented by m players Y_i and the value of the clock $val(y)$ is the binary number represented by the strategies of the player. We have $2n \cdot m$ circuits. Circuit (i, j, b) outputs 1 whenever: (a) $f_S(b, x_{-i}) < f_S(1-b, x_{-i})$ and (b) the highest bit at which $f_S(b, x_{-i})$ is smaller

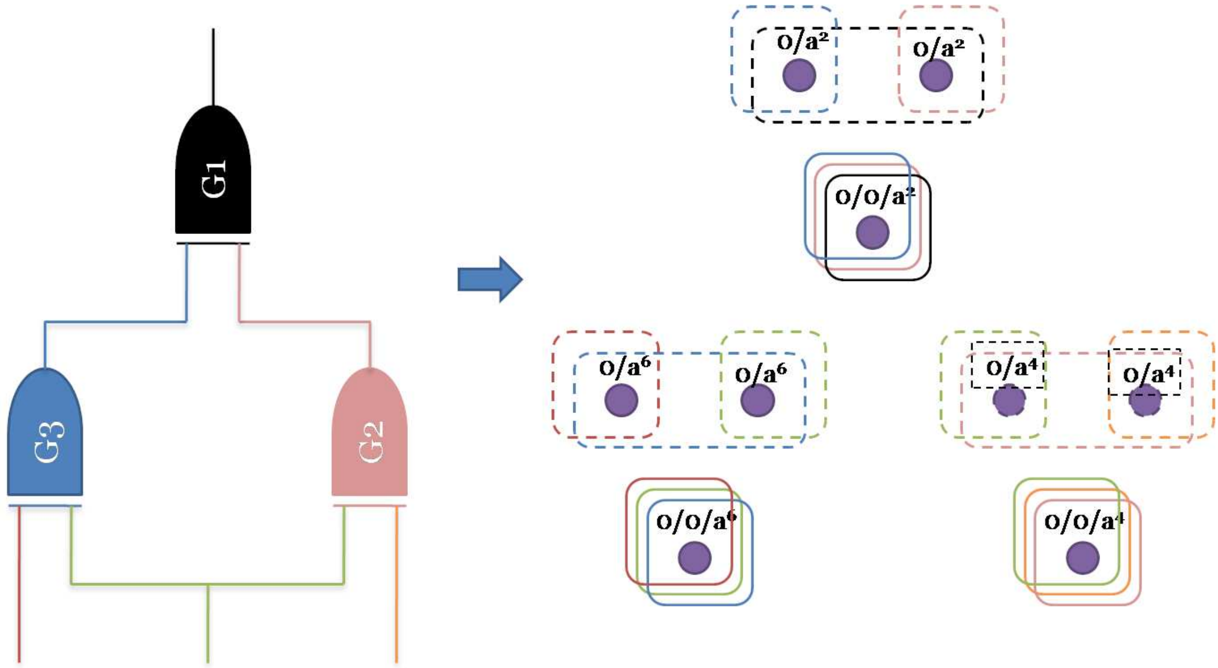

Figure 2.8: An example of a congestion game that simulates a 2-input NAND circuit. Sets with the same color and dashing consitute a common strategy. The colors imply the correspondence between players and signals in the circuit.

than $val(y)$ is greater than j or more formally $\sum_{i=j+1}^{m} y_i 2^{i-1} + 2^j > f_S(b, x_{-i})$. The controller is associated with a player C.

Moreover, we slightly modify the congestion games that simulate NAND circuits by adding Lock resources that exist in any strategy of all players of the circuit. The lock facilities are available only to the controller player. Moreover, we deprive the output players from their zero strategy so that the output players always play their one-strategy. The delays of the lock resources are such that the controller has incentive to lock them only when they don't violate the NAND semantics (i.e. only when the input of the circuits gives output 1). Moreover, whenever the controller locks a circuit then the players of the circuit don't have incentive to change or otherwise they would incure a tremendous cost.

The logic of the congestion game is the following:

- With a circuit S_0 we always have the guarrantee that $val(y) \geq f_S(x)$. If this condition is violated then player C has a best response of choosing a reset strategy at which the clock is reset to the value $2^m - 1$ where the condition stated definitely holds.
- Whenever the input players don't correspond to a local minimum then there exists a circuit $S_{(i,j,b)}$ whose output is 1 and such that Y_j plays his one

strategy and thus controller can lock it (if any of the above conditions doesn't hold then the controller would incur a very high cost for locking circuit $S_{(i,j,b)}$.

- By locking that circuit player Y_j has incentive to change to a strategy called $change_{(i,j,b)}$ at which he triggers player X_i to move to his b strategy and all players $Y_{j'}$ with $j' < j$ to move to their one-strategy. If player X_i was not at his b strategy then the above sequence triggers in some sense an improvement flip.

- After the above triggers are finished player Y_j has incentive to change to a strategy we call $check_{(i,j,b)}$. This strategy has the semantics of a zero boolean value. After the above changes the condition $val(y) \geq f_S(x)$ still holds since $val(y)$ has become $\sum_{i=j+1}^{m} y_i 2^{i-1} + 2^j$ after the Y_j triggering and thus we know that the above value is greater than or equal $f_S(b, x_{-i})$ which is the boolean output of the current input vector. Thus the output of circuit S_0 is 1 and the controller can lock that circuit.

- By locking circuit S_0 the controller unblocks the zero-strategy of player Y_j who now moves to each zero strategy.

Whenever the game is not in some of the above described states we can impose a huge delay on the controller so that he has incentive to move to each reset strategy and subsequently to locking circuit S_0. Thus, we need to check only the above states. From the sequance of steps one can see that the only states that we allow the sequence to stop is at those where the controller locks S_0, which we call base states. At any other state some player has incentive to change regardless of whether the X players correspond to a local minimum or not. Moreover we know that for the sequence to stop at a base state then no circuit $S_{(i,j,b)}$ outputs 1. If x was not a local mimimum of the FLIP problem then it holds that for some i and b $f_S(b, x_{-i}) < f_S(x)$. Moreover, $val(y) \geq f_S(x) > f_S(b, x_{-i})$. The above two conditions lead to the fact that some circuit $S_{(i,j,b)}$ must output 1 which leads to a contradiction and thus x is definately a local minimum.

So on a high level we solve the feedback problem in the following way: we put a clock that counts downwards. Whenever, the clock is higher than the value of the circuit that corresponds to the current input or whenever there exists an improvement flip in the current input the controller causes some or both of the following actions to happen: the clock to drop down to a value closer to the current $f_S(x)$, the input to make an improvement flip. This sequence will eventually lead to a state where no improvement flips can be made and where the clock represents the circuit output of the current input.

We now continue to some positive results in calculating approximate solutions in Congestion Games.

One of the most important positive results corcerns the subclass of Symmetric Congestion Games. Chien and Sinclair [CS07] prove that for symmetric congestion games computing an a-approximate PNE is easy and the a-Nash dynamics (Nash Dynamics where an edge exists iff some player can improve by a) converge to an equilibrium in polynomially many steps when the following bounded jump condition holds: the delay function of each facility satisfies the inequality $d_f(t+1) \leq ad_f(t)$. Both symmetry and bounded jump conditions are violated by the construction of Skopalik and Voecking and this is why a positive result was possible for this subclass of congestion games.

Another positive result is towards another direction of approximation, that of approximating the social cost function.of the sum of player costs. Awerbuch et al. [AAE$^+$08] show that the convergence time of a-Nash Dynamics to a solution that is arbitrarily close to the Price of Anarchy requires polynomially many steps when the delay functions satisfy the bounded jump condition.

Another interesting result that we first present in this thesis is the observation that the construction of Skopalik and Voecking works also for Bottleneck Games since the facility that causes the dynamics of the congestion game constructed to follow the semantics of a processor is always that of the maximum delay. Hence we conclude to the following theorem:

Theorem 27. *Computing an a-approximate PNE in the class of Bottleneck Games is PLS-complete.*

2.5 Quantifying the Inefficiency of Congestion Games

In 1968 Garret Hardin wrote an infulential article that portrayed the deterioration caused to a resource when it is used freely by selfish users. This deterioration was called the Tragedy of the Commons.

Consider for example the following game:

Example 12. *(Pollution Game). Consider the game where n countries want to decide whether to pass a law on controlling polution or not. Tha cost of pollution controll is 3 for each country but a country that pollutes adds a cost of 1 to all countries (health costs, etc.). Consider the case when k countries pollute. Then those countries suffer k and the rest $n-k$ countries suffer $k+3$. Each country that controlls polution has incentive to change and suffer $k+1$. The only Nash Equilibrium in this game is when all countries pollute and suffer n.*

In this section we will present some basic definitions and results concerning the deterioration caused by selfishness in the restricted case of Congestion Games, under the perspective of some intuitive social cost functions.

Since computing a PNE of a Congestion Game is hard (PLS-complete) and we cannot possibly do it in large games then the question is: does there exist a way of mathematically bounding the deterioration caused by lack of coordination without actually having to compute the PNEs of a game?

The answer is yes for several classes of games and this line of research has been very fruitful the past decade and especially in the class of congestion games.

2.5.1 Price of Anarchy

When trying to examine the inefficiency of the outcome of a game we have to define what is the social cost that we want to minimize. So formally we need to quantify the quality of any strategy profile using some social cost function $SC(s) : \times_i S_i \to \mathbb{R}$.

Having defined a social cost function we need some measure that will tell us how bad the outcome of a game can be when compared to the optimal strategy profile (the one that minimizes social cost). To come up with some measure we first want to say which outcome we want to examine. We have seen in the introduction several notions of equilibria each claiming to be a possible outcome of a game. The most well studied case is that of Nash Equilibria (mixed and pure). However, even if we restrict to Nash Equilibria the outcome of a game is not unique.

The Price of Anarchy (PoA) [KP99], the most popular measure of the inefficiency of equilibria resolves the issue of multiple equilibria by adopting a worst-case approach. Formally the price of anarchy is defined as the ratio of the worst social cost of an equilibrium of a game to the social cost of an optimal outcome:

$$\text{PoA} = \max_{NE} \frac{SC(NE)}{SC(OPT)} \tag{2.25}$$

Chapter 3

Social Contexts

In this section we examine two newly proposed strategic game models closely related to congestion games. Their similarity as well as their great expressiveness as opposed to the already known models is what lead us to descriminate them and to devote a separate chapter on describing them and on declaring their properties.

The two models that we will examine are Graphical Congestion Games and Congestion Games with Social Contexts. Their similarity is that both these games introduce a player graph in an underlying congestion game. In the first model the player graph tries to capture the fact that in a congestion game players may not be aware of all the players but only in their neighbours in a player graph. In the second model the player graph models social contexts that might exist among the players in a congestion game. The social graph tries to model relations and the players cost is not the cost in the congestion game but some function of his and his neighbours costs.

3.1 Graphical Congestion Games

The Graphical Congestion Game model is based on the general graphical game model that has been well studied in game theory. It is generally motivated by some information considerations discussed in the introduction and generally tries to model situations where players are not aware of everyone who is participating in the game.

Lets first give the definition of a graphical game:

Definition 38. *A graphical game is defined in terms of:*

- *A directed graph $G = (V, E)$ where the nodes represent the players of the game and the edges represent dependencies between the players*
- *For each node $v \in V$:*
  - *A set of available strategies S_v*

– *A utility function* $u_v : S_v \times (\times_{u \in N(v)} S_u) \mapsto \Re$ *where* $N(v)$ *is the neighbors of player* v *in graph* G.

The major difference between graphical games and general strategic games is the fact that a players utility depends on the strategy profile of only a subset of the players in the game, whilst in general strategic games it depended on every player. Moreover, graphical games are a very succinct representation of situations where many players participate in the game with few dependencies (e.g. in the internet).

Similar to the graphical game model we can define a more restricted model where the underlying game is not a general strategic game but a congestion game:

Definition 39. *A graphical congestion game is defined as a tuple* $H = \langle G = ([N], M), F, (S_i)_{i \in [N]}, (d_f)_{f \in F}$ *where* $G = ([N], M)$ *is a directed graph, called the social knowledge graph,* $[N]$ *is a set of players,* F *is a set of facilities,* $S_i \subseteq 2^F$ *is a set of pure strategies for player* i*, each consisting of a set of facilities, and* d_f *is the latency function for the facility* f *depending on the number of players using* f.

Given a strategy profile s *we note with* $G_f(s) = (N_f(s), M_f(s))$ *the subgraph of* G *induced by the set of players using facility* f *in* s*, i.e.* $N_f(s) = \{i \in N : f \in s_i\}$ *and* $M_f(s) = \{(i, j) \in M : i, j \in N_f(s)\}$. *Let* $n_f(s) = |N_f(s)|$ *and* $m_f(s) = |M_f(s)|$ *be the number of nodes and edges in* $G_f(s)$ *respectively, and* $\delta^i_f(s)$ *the degree of node* i *in* $G_f(s)$. *The cost of player* i *in the strategy profile* s *is* $c_i(s) = \sum_{f \in s_i} d_f(n^i_f(s))$, *where* $n^i_f(s)$ *is the number of nodes adjecent to* i *in* $G_f(s)$ *together with* i*, that is* $n^i_f(s) = \delta^i_f(s) + 1$

3.1.1 Existence of PNE

Since Graphical Congestion Games are a generalization of Congestion Games (when the social graph is a complete graph then the two models coincide) the first question we need to ask is whether they always admit a PNE.

The question is negative for the general class of congestion games. We provide here an instance of a graphical congestion game with linear latency functions where no PNE exists:

Example 13. *The social graph is defined as* $G = (\{1, 2, 3\}, \{(1, 2), (2, 3), (3, 1)\})$. *All players have the same set of strategies* $\{\{f_1\}, \{f_2\}\}$ *and the latency functions are* $d_{f_1}(x) = d_{f_2}(x) = x$. *It is easy to observe that any strategy profile of the game is not a PNE. In any possible state of the game, at least two players use the same facility . Thus one of them has cost* 1*, while the other has cost* 2 *and can decrease it to* 1 *by changing her strategy.*

Thus we need to restrict to subclasses of graphical congestion game in order to prove existence of PNE. The positive existence results we provide here where first proved by Bilo et al. [BFFM08] and concern only the case of linear delay functions.

Theorem 28. *Every graphical congestion game with linear delay funcions ($d_f(x) = a_f x + b_f$) defined over an undirected social graph is an exact potential game with potential function:*

$$\Phi(s) = \sum_{f \in F} \left(a_f(m_f(s) + n_f(s)) + b_f n_f(s) \right) \tag{3.1}$$

For directed social graphs we have a positive existence result when the graph is a DAG and it is also easy to prove that there always exists a sequence of best response dynamics that can be computed in polynomial time that end in a PNE.

3.1.2 Price of Anarchy Bounds

Another interesting question is what is the PoA of the model for the subclasses that always admit a PNE. For the case of linear latency function Bilo et al. give upper bounds for two intuitive social cost functions: the total presumed social cost $SC_{PR}^{sum}(s) = \sum_{i \in [N]} c_i(s) = \sum_{f \in F} \sum_{i: f \in s_i} d_f(n_f^i(s))$ and the total perceived social cost $SC_{PE}^{sum}(s) = \sum_{f \in F} n_f(s) d_f(n_f(s))$ (the toal cost due to the actual congestion on facilities).

For the presumed social cost the current known upper bound on the PoA is $\Delta + 1$ where Δ is the maximum degree of a node in the social graph. This bound holds both for undirected and directed acyclic graphs. The lower bound for both cases is again quite tight ($\frac{2\Delta+1}{3}$ and $\frac{\Delta+1}{2}$ respectively).

For the perceived social cost the corresponding upper bound is $N(\Delta+1)$ again both for undirected and directed acyclic graphs. The lower bounds are $\frac{N^2}{4}$ and $N(\Delta + 1)$ respectively.

Despite the tighness of the results it can be claimed that better bounds are still to be found due to the following fact: If the social graph is complete then the model coincides with congestion games. For complete graphs $\Delta = N$ and thus it leads to a price of anarchy of $N + 1$ while from [CK05] we know that in congestion games with linear cost functions the PoA for the sum social cost is $\frac{5}{2}$. This leads to a contradiction showing that the maximum degree of the graph is most probably not the right parameter with which to express the PoA in graphical congestion games.

3.2 Social Contexts in Congestion Games

Another very interesting extention to the basic congestion games model was proposed very recently by Ashlagi et al. [AKT08]. In the models of the previous chapter we almost all the times neglected the social connection between the players. In most real life situations players in a congestion game (e.g. in a network routing game) are not completely independent of each other and most of the times are playing under some social context (e.g. friendship, collaboration, competition etc.). This social context is not captured by the congestion games model were we consider that a player's cost is the delay of his strategy.

A first attempt to capture such relations was coalitional congestion games but the expressiveness there was quite restricted only to the case where players form coalitions and take decisions together in collaboration. Moreover, it was implicit that people where divided into groups and were related only with the players in their group and with noone else. This is not so descriptive of real life situations where the social dependencies between people is best described by a social network.

Under this very intuitive perspective we formally introduce the notion of social context game that could be applied to any strategic game and then we study only the case when it is applied to congestion games.

Definition 40. *Given an underlying game H, a social context game is generated by considering a neighborhood graph over the players and aggregation functions that determine how a game is affected by the graph. A social context is a tuple $F = (G, (f_i)_{i\in[N]})$, where $G = (N, E)$ is an undirected graph and for every $i, f_i : G \times \mathbb{R}^N \to \mathbb{R}$ is an aggregation function. The aggregation function maps a payoff profile of the underlying game into a utility profile, as a function of the graph. The aggregation function tries to capture a players social behaviour.*

Given an underlying game $H = \langle N, (S_i)_{i\in[N]}, (c_i)_{i\in[N]}\rangle$, and a social context $F = (G, (f_i)_{i\in[N]})$, a social context game $S = \langle N, (S_i)_{i\in[N]}, (t_i)_{i\in[N]}\rangle$ is a strategic game where N is the set of players S_i is the set of strategies for player i and $t_i : \times_i S_i \to \mathbb{R}$ satisfies that $t_i(s) = f_i(G, c_1(s), \ldots, c_N(s))$. For convenience we will write $f_i(G, s)$ to refer to $f_i(G, c_1(s), \ldots, c_N(s))$.

In the above definition it is improtant to notice that there exist to types of costs. The costs of the underlying game H which are denoted as the immediate costs and the final costs of the players in the social context game S.

The above model is quite general in that it allows for arbitrary social attitude aggregation functions f_i. Things become more clear when we constraint to special types of aggregation functions that capture different social attitudes. A very important notion in defining intuitive aggregation functions is that of the group of a player i which is defined as $g(i) = i \cup N(i)$, where $N(i)$ are the neighbors of i in the social graph.

Some types of the aggregation functions introduced in [AKT08] are the following:

- Best-Member Collaboration: A palyer's cost is the minimal immediate cost in her group. $f_i(G, s) = \min_{j\in g(i)} c_j(s)$.
- MinMax Collaboration: A player's cost is the maximal immediate cost in her group. $f_i(G, s) = \max_{j\in g(i)} c_j(s)$.
- Surplus Collaboration: a player's cost is the average of the immediate costs of her group. $f_i(G, s) = \frac{1}{|g(i)|}\sum_{j\in g(i)} c_j(s)$.

In the next section we will cope solely with surplus collaboration games when the underlying game is a congestion game. The rest of the above definition where given in order to demonstrate the expressiveness of the model.

If we restrict to social graphs that consist of cliques of players then surplus collaboration games is almost equivalent to coalitional congestion games with the major difference that in this model the players remain the same and only unilateral deviations are allowed at an equilibrium.

3.2.1 Existence of Pure Nash Equilibria

The current results in the existence of PNE in surplus collaboration games with congestion games as the underlying game are very few.

The inexistence of PNE can be proved by a counter example even in the case of parallel links.

Example 14. *Consider a social context game with 4 players and 2 parallel links. Each link has cost function* $(1, 5, 6, 6)$*. The graph* G *has a* 3*-clique on the vertices* $1, 2, 3$ *and vertex* 4 *is isolated. Any strategy profile of the above game is not a PNE: We may assume that player* 4 *is assigned to resource* 2*. It is easy to verify that for each partition of the rest of the players to the links the resulting strategy profile is not an equilibrium.*

An objection on the above example is that the social graph is disconnected. However, the following example shows that even when the graph is a tree there exists a congestion game that does not admit a PNE.

Example 15. *Consider the following social context game:* G *is an undirected tree with one root and* 6 *children, and let* H *have* 2 *identical links with cost function* $(1, 1, 2.9, 5, 5, 5, 5)$*. Any strategy profile in the above game doesn't constitute a PNE.*

The above negative results show that Congestion Games with Social Contexts is a much more general class of games and certainly doesn't preserve some of the most significant properties of congestion games.

The only positive result on the existence problem is the following:

Theorem 29. *Let* H *be a congestion game on* m *identical parallel links and* G *be a tree with maximal degree* $m - 2$*. Then, there exists a PNE in the corresponding surplus collaboration social context game.*

Chapter 4

Colored Resource Allocation Games

In this chapter we will present Colored Resource Allocation Games, a new model of congestion games proposed during the process of the thesis. It is a joint work of Evangelos Bampas, George Pierrakos, Aris Pagourtzis and the author. The motivation for proposing this new model lies in the area of optical networks.

In optical networking it is highly desirable that all communication be carried out *transparently*, that is, each signal should remain on the same wavelength from source to destination. The need for efficient access to the optical bandwidth has given rise to the study of several optimization problems in the past years. The most well-studied among them is the problem of assigning a path and a color (wavelength) to each communication request in such a way that paths of the same color are edge-disjoint and the number of colors used is minimized. Nonetheless, it has become clear that the number of wavelengths in commercially available fibers is rather limited—and will probably remain such in the foreseeable future. Therefore, the use of multiple fibers has become inevitable in large scale networks. In the context of multifiber optical networks several optimization problems have been defined and studied, the objective usually being to minimize either the maximum fiber multiplicity per edge or the sum of these maximum multiplicities over all edges of the graph.

There has been growing interest recently for studying the behaviour of optical networks under lack of centralized control, that is, under the assumption that users can choose the desired route and wavelength for their requests *selfishly* [BM04, BFM05, GKS05, FFM$^+$06, MPP07, BPPP08, FMM$^+$08].

Colored Resource Allocation Games, is a class of games that can model non-cooperative versions of routing and wavelength assignment problems in multifiber all-optical networks. They can be viewed as an extension of congestion games where each player has his strategies in multiple copies (colors). When restricted to (optical) network games, facilities correspond to edges of the network and colors to wavelengths. The number of players using an edge in the same color represents a lower bound on the number of fibers needed to implement the corresponding physical link. Having this motivation in mind, we consider both egalitarian (max) and utilitarian (sum) player costs. For our purposes it suffices to restrict our study to identity latency functions.

We use the price of anarchy (PoA) introduced in [KP99] as a measure of the deterioration caused by lack of coordination. We estimate the PoA of our games under three different social cost functions. Two of them are standard in the literature (see e.g. [CK05]): the first (SC_1) is equal to the maximum player cost and the second (SC_2) is equal to the sum of player costs (equivalently, the average player cost). The third one is specially designed for the setting of multifiber all-optical networks; it is equal to the sum over all facilities of the maximum color congestion on each facility. Note that in the optical network setting this function represents the total fiber cost needed to accommodate all players; hence, it captures the objective of a well-studied optimization problem. Let us also note that the SC_1 function under the egalitarian player cost captures the objective of another well known problem, namely minimizing the maximum fiber multiplicity over all edges of the network.

	Colored Congestion Games	Congestion Games
$SC_1(A) = \max_{i \in [N]} C_i(A)$	$\Theta\left(\sqrt{\frac{N}{W}}\right)$	$\Theta\left(\sqrt{N}\right)$ [CK05]
$SC_2(A) = \sum_{i \in [N]} C_i(A)$	$\frac{5}{2}$	$\frac{5}{2}$ [CK05]
$SC_3(A) = \sum_{f \in F} \max_{a \in [W]} n_{f,a}(A)$	$\Theta\left(\sqrt{W\|F\|}\right)$	—

Table 4.1: The pure price of anarchy of Colored Congestion Games. Results for classical congestion games are shown in the right column.

	Colored Bottleneck Games	Bottleneck Games
$SC_1(A) = \max_{i \in [N]} C_i(A)$	$\Theta\left(\frac{N}{W}\right)$	$\Theta(N)$ [BMI06]
$SC_2(A) = \sum_{i \in [N]} C_i(A)$	$\Theta\left(\frac{N}{W}\right)$	$\Theta(N)$ [BMI06]
$SC_3(A) = \sum_{f \in F} \max_{a \in [W]} n_{f,a}(A)$	F	—

Table 4.2: The pure price of anarchy of Colored Bottleneck Games. Results for classical bottleneck games are shown in the right column.

Our main contribution is the derivation of tight bounds on the price of anarchy for Colored Resource Allocation Games. These bounds are summarized in Tables 4.1 and 4.2. It can be shown that the bounds for Colored Congestion Games remain tight even for network games.

Observe that known bounds for classical congestion and bottleneck games can be obtained from our results by simply setting $W = 1$. On the other hand one might notice that our games can be casted as classical congestion or bottleneck games with $W|F|$ facilities. However we are able to derive better upper bounds for most cases by exploiting the special structure of the players' strategies.

4.1 Model Definition

Definition 41 (Colored Resource Allocation Games)**.** *A Colored Resource Allocation Game is defined as a tuple $\langle F, N, W, \{\mathcal{E}_i\}_{i\in[N]}\rangle$ such that:*

1. *F is a set of facilities f_i*
2. *$[W]$ is a set of colors*
3. *$[N]$ is a set of players*
4. *$\mathcal{E}_i$ is a set of possible facility combinations for player i such that:*
   a. *$\forall\ i \in [N] : \mathcal{E}_i \subseteq 2^F$*
   b. *$S_i = \mathcal{E}_i \times [W]$ is the set of possible strategies of player i*
   c. *$A_i = (E_i, a_i) \in S_i$ is the notation of a strategy for player i with $E_i \in \mathcal{E}_i$ denoting the set of facilities used and $a_i \in [W]$ the corresponding color*
5. *$A = (A_1, \ldots, A_N)$ is a strategy profile for the game*
6. *For a strategy profile A, $\forall f \in F$, $\forall c \in [W]$, $n_{f,c}(A)$ is the number of players that use facility f in color c in strategy profile A*

Depending on the player cost function we define two subclasses of Colored Resource Allocation Games:

- *Colored Congestion Games (CCG), where the player cost is $C_i(A) = \sum_{e\in E_i} n_{e,c_i}(A)$*
- *Colored Bottleneck Games (CBG), where the player cost is $C_i(A) = \max_{e\in E_i} n_{e,c_i}(A)$*

For each of the above variations we will consider three different social cost functions:

- $SC_1(A) = \max_{i\in[N]} C_i(A)$
- $SC_2(A) = \sum_{i\in[N]} C_i(A) = \sum_{f\in F} \sum_{a\in[W]} n_{f,a}^2(A)$

- $SC_3(A) = \sum_{f \in F} \max_{a \in [W]} n_{f,a}(A)$

From the definition of pure Nash Equilibrium we can derive the following two facts that hold in Colored Congestion and Bottleneck Games respectively:

Fact 1. For a PNE strategy profile A of a CCG it holds:

$$\forall E_i' \in \mathcal{E}_i, \forall c' \in [W] : C_i(A) \leq \sum_{e \in E_i'} (n_{e,c'}(A) + 1) \tag{4.1}$$

Fact 2. For a PNE strategy profile A of CBG it holds:

$$\forall E_i' \in \mathcal{E}_i, \forall c' \in [W] : C_i(A) \leq \max_{e \in E_i'} (n_{e,c'}(A) + 1) \tag{4.2}$$

Equivalently:

$$\forall E_i \in \mathcal{E}_i, \forall c \in [W], \exists e \in E_i : C_i(A) \leq n_{e,c}(A) + 1 \tag{4.3}$$

4.2 Colored Congestion Games

In this section we compute the pure price of anarchy of colored congestion games for three different social cost functions.

4.2.1 Pure PoA for Social Cost SC_1

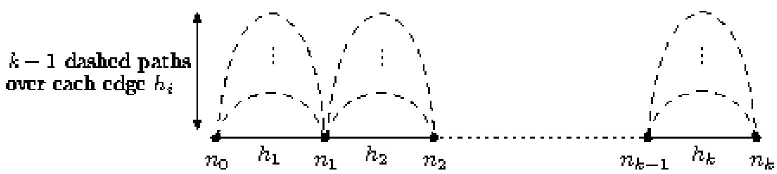

Figure 4.1: A worst-case instance that proves the tightness of the upper bound, depicted as network game. A dashed line represents a path of length k connecting its two endpoints.

Theorem 30. *The price of anarchy of any Colored Congestion Game $\langle F, N, W, \{\mathcal{E}_i\}_{i \in [N]} \rangle$ with social cost SC_1 is $O\left(\sqrt{\frac{N}{W}}\right)$.*

Proof. Let A be a Nash Equilibrium and let OPT be an optimal strategy profile. Without loss of generality we consider the first player to have the maximum cost, $SC_1(A) = C_1(A)$. Thus we need to bound $C_1(A)$ with respect to the optimum social cost $SC_1(\mathrm{OPT}) = \max_{j\in[N]} C_j(\mathrm{OPT})$.

Since A is a Nash Equilibrium every player has no benefit of changing either her color or her choice of facilities. We denote with $\mathrm{OPT}_1 = (E_1^\star, a_1^\star)$ the strategy of player P_1 in OPT. Since A is a N.E. it must hold:

$$\forall a \in [W]: \; C_1(A) \le \sum_{e\in E_1^\star} (n_{e,a}(A)+1) \le \sum_{e\in E_1^\star} n_{e,a}(A) + C_1(\mathrm{OPT}) \tag{4.4}$$

The second inequality holds since any strategy profile cannot lead to a cost for a player that is less than the size of her facility combination.

Let $I \subset [N]$ the set of players that, in A, use some facility $e \in E_1^\star$. The sum of their costs is:

$$\begin{aligned} \sum_{i\in I} C_i(A) \ge \sum_{e\in E_1^\star}\sum_{a\in[W]} n_{e,a}^2(A) \ge \frac{(\sum_{e\in E_1^\star}\sum_{a\in[W]} n_{e,a}(A))^2}{|E_1^\star|W} \ge \\ \frac{(W\min_{a\in[W]}\sum_{e\in E_1^\star} n_{e,a}(A))^2}{|E_1^\star|W} \ge \frac{W(\min_{a\in[W]}\sum_{e\in E_1^\star} n_{e,a}(A))^2}{|E_1^\star|} \end{aligned} \tag{4.5}$$

The first inequality holds since a player in I might use facilities (e,a) not in $E_1^\star$ and the second inequality holds from the Cauchy-Schwarz inequality. Let $a_{\min} = \arg\min_{a\in[W]} \sum_{e\in E_i^\star} n_{e,a}(A)$. Thus we have:

$$\left(\sum_{e\in E_1^\star} n_{e,a_{\min}}(A)\right)^2 \le \frac{|E_1^\star|}{W}\sum_{i\in I} C_i(A) \tag{4.6}$$

From [CK05] we have:

$$\sum_{i\in[N]} C_i(A) \le \frac{5}{2}\sum_{i\in[N]} C_i(\mathrm{OPT}) \tag{4.7}$$

Combining the above two inequalities we have:

$$\left(\sum_{e\in E_1^\star} n_{e,a_{\min}}(A)\right)^2 \le \frac{|E_1^\star|}{W}\sum_{i\in I} C_i(A) \le \frac{|E_1^\star|}{W}\sum_{i\in[N]} C_i(A) \le \frac{5}{2}\frac{|E_1^\star|}{W}\sum_{i\in[N]} C_i(\mathrm{OPT}) \tag{4.8}$$

Combining with (4.4) for $a_{\min}$, we get

$$C_1(A) \le C_1(\mathrm{OPT}) + \sqrt{\frac{5}{2}\frac{|E_1^\star|}{W}\sum_{i\in[N]} C_i(\mathrm{OPT})} \tag{4.9}$$

Since $|E_1^\star| \leq C_1(\text{OPT})$ and $C_i(\text{OPT}) \leq SC_1(\text{OPT})$, we get

$$C_1(A) \leq \left(1 + \sqrt{\frac{5}{2}\frac{N}{W}}\right) SC_1(\text{OPT}) \tag{4.10}$$

$\square$ $\square$

Theorem 31. *There exists an infinite set of Colored Congestion Games $\langle F, N, W, \{\mathcal{E}_i\}_{i\in[N]}\rangle$ with social cost SC_1, that have pure price of anarchy $\Omega\left(\sqrt{\frac{N}{W}}\right)$.*

Proof. We will describe the lower bound instance as a network game. The underlying network is illustrated in Figure 4.1.

In that network W major players want to send traffic from n_0 to n_k. For every i, $0 \leq i \leq k-1$, there are $(k-1)W$ minor players that want to send traffic from node n_i to node n_{i+1}. In the worst-case equilibrium A all players choose the short central edge, leading to social cost $SC_1(A) = k^2$. In the optimum the minor players are equally divided on the dashed-line paths and the major players choose the central edge. This leads to $SC_1(\text{OPT}) = k$, and the price of anarchy is therefore:

$$\text{PoA} = k = \Theta\left(\sqrt{\frac{N}{W}}\right) \tag{4.11}$$

$\square$ $\square$

4.2.2 Pure PoA for Social Cost SC_2

The price of anarchy for Social Cost SC_2 is upper-bounded by 5/2, as proved in [CK05]. For the lower bound, we use a slight modification of the instance described in [CK05]. We have NW players and $2N$ facilities. The facilities are separated into two groups: $\{h_1, \ldots, h_N\}$ and $\{g_1, \ldots, g_N\}$. Players are divided into N groups of W players. Each group i has strategies $\{h_i, g_i\}$ and $\{g_{i+1}, h_{i-1}, h_{i+1}\}$. The optimal allocation is for all players in the i-th group to select their first strategy and be equally divided in the W colors, leading to $SC_2(\text{OPT}) = 2NW$. In the worst-case NE players choose their second strategy and are equally divided in the W colors, leading to $SC_2(A) = 5NW$. Thus, the PoA of this instance is 5/2 and the upper bound remains tight in our model too.

4.2.3 Pure PoA for Social Cost SC_3

Theorem 32. *The price of anarchy of colored congestion games with social cost SC_3 is $O\left(\sqrt{W\,|F|}\right)$.*

Proof. We denote by $\overline{n_e(S)}$ the vector $[n_{e,a_1}(S), \ldots, n_{e,a_W}(S)]$. In terms of the above vector we can write:

$$SC_3(S) = \sum_{e\in F} \max_{a\in[W]} n_{e,a}(S) = \sum_{e\in F} \|\overline{n_e(S)}\|_\infty \tag{4.12}$$

From norm inequalities we have:

$$\frac{\|\overline{n_e(S)}\|_2}{\sqrt{W}} \leq \|\overline{n_e(S)}\|_\infty \leq \|\overline{n_e(S)}\|_2 \tag{4.13}$$

hence:

$$SC_3(S) = \sum_{e\in F} \|\overline{n_e(S)}\|_\infty \leq \sum_{e\in F} \sqrt{\sum_a n_{e,a}^2(S)} \leq \sqrt{|F|}\sqrt{\sum_{e\in F}\sum_a n_{e,a}^2(S)}\ , \tag{4.14}$$

where the last inequality is a manifestation of the norm inequality $\|\vec{x}\|_1 \leq \sqrt{n}\|\vec{x}\|_2$, where $\vec{x}$ is a vector of dimension n. Now, from the first inequality of (4.13) we have:

$$SC_3(S) \geq \frac{1}{\sqrt{W}} \sum_{e\in F} \sqrt{\sum_a n_{e,a}^2(S)} \geq \frac{1}{\sqrt{W}} \sqrt{\sum_{e\in F}\sum_a n_{e,a}^2(S)} \tag{4.15}$$

Combining (4.15) and (4.14) gives:

$$\frac{1}{\sqrt{W}}\sqrt{SC_2(S)} \leq SC_3(S) \leq \sqrt{|F|}\sqrt{SC_2(S)} \tag{4.16}$$

From [CK05] we know that the price of anarchy with social cost $SC_2(S)$ is 5/2. Let A be a worst-case Nash Equilibrium in the case of SC_3 and let OPT be an optimal strategy profile. From (4.16) we know that $SC_3(A) \leq \sqrt{|F|}\sqrt{SC_2(A)}$ and $SC_3(\text{OPT}) \geq \frac{1}{\sqrt{W}}\sqrt{SC_2(\text{OPT})}$. Thus:

$$\text{PoA} = \frac{SC_3(A)}{SC_3(\text{OPT})} \leq \sqrt{W\,|F|}\sqrt{\frac{SC_2(A)}{SC_2(\text{OPT})}} \leq \sqrt{W\,|F|}\sqrt{\frac{5}{2}} \tag{4.17}$$

□ □

Theorem 33. *There exists an infinite set of Colored Congestion Games with social cost SC_3 that have* PoA $= \sqrt{W\,|F|}$.

Proof. Consider a colored congestion game with N players, $|F| = N$ facilities and $W = N$ colors. Each player has strategies the singleton sets consisting of one facility. In other words $\mathcal{E}_i = \{\{f_1\}, \{f_2\}, \ldots, \{f_N\}\}$.

The above instance has a worst-case equilibrium with social cost N when all players choose a different facility in an arbitrary color. On the other hand in the optimum strategy profile players fill all colors of the necessary facilities. This needs $\frac{N}{W}$ facilities with maximum capacity over their colors 1. Thus the optimum social cost is $\frac{N}{W}$ leading to a PoA $= \sqrt{W\,|F|}$. □ □

4.3 Colored Bottleneck Games

4.3.1 Convergence to Equilibrium

Definition 42. (Player Congestion Vector). *A player congestion vector for a strategy profile A of a CBG is a vector $[b_N, \ldots, b_1]$ where*

$$b_i = |\{P_k \in [N] : c_k(A) = i\}| \tag{4.18}$$

Theorem 34. *For a CBG $\langle F, N, W, \{\mathcal{E}_i\}_{i\in[N]}\rangle$ any Nash dynamics converges to a Nash Equilibrium in finitely many steps.*

Proof. Consider an arbitrary initial strategy profile A_0 and its corresponding Player Congestion Vector $\mathcal{CV}_0$. At every step m of a Nash dynamics one player k must make an improving move. Let $C_k(A_m) = j$. Then, b_j of $\mathcal{CV}_m$ must decrease at least by 1, since no other player's cost can be increased to b_j and no player with higher cost is affected. Thus, the quantity $\sum_{i\in N}(b_i(A_m)N^i)$ decreases at every step and must converge to a PNE in a finite number of steps. □ □

Corollary 1. *For any CBG $\langle F, N, W, \{\mathcal{E}_i\}_{i\in[N]}\rangle$ the price of stability is* 1.

4.3.2 Pure PoA for Social Cost SC_1

Theorem 35. *The price of anarchy of any CBG game with social cost $SC_1(A)$ is at most $\frac{N}{W}$.*

Proof. It is obvious that $SC_1(\mathrm{OPT}) \geq 1$. Let $SC_1(A) \geq \frac{N}{W} + 1$. From Fact 2 at least $\frac{N}{W}$ players must play each of the other colors. This needs at least N+1 players. □ □

Theorem 36. *There exist instances of a CBG game with pure* PoA $= \frac{N}{W}$.

Proof. Consider the following class of CBG games. We have N players and N facilities. Each player P_i has two possible strategies: $\mathcal{E}_i = \{\{f_i\}, \{f_1, \ldots, f_N\}\}$. In a worst-case NE all players choose the second strategy and they are equally divided in the W colors. This leads to player cost $\frac{N}{W}$ for each player and thus to a social cost $\frac{N}{W}$. In the optimal strategy all players would choose their first strategy leading to player and social cost 1. Thus the PoA for this instance is $\frac{N}{W}$. □ □

4.3.3 Pure PoA for Social Cost SC_2

Theorem 37. *The price of anarchy of any CBG game with social cost $SC_2(A)$ is at most $\frac{N}{W}$.*

Proof. By thm 4.3.2 we know that $C_i(A) \leq \frac{N}{W}$. Moreover it is obvious that $SC_2(\mathrm{OPT}) \geq N$. Thus PoA $= \frac{N \cdot C_i(A)}{SC_2(\mathrm{OPT})} \leq \frac{N}{W}$. □ □

The instance used in the previous section can also be used here to prove that the above inequality is tight for a class of CBG games.

4.3.4 Pure PoA for Social Cost SC_3

To state the following thm we have to define a set.

Definition 43. *We define E_S to be the set of facilities used by at least one player in the strategy profile $S = (A_1, \ldots, A_N)$, i.e.*

$$E_S = E_1 \cup \ldots \cup E_N \tag{4.19}$$

Theorem 38. *The price of anarchy of any CBG game with social cost $SC_3(A)$ is at most F.*

Proof. We exclude from the sum over the facilities, those facilities that are not used by any player since they do not contribute to the social cost. Thus we focus on facilities with $\max_a n_{e,a} > 0$. Let $a_{\max}(e)$ denote the color with the maximum multiplicity at facility e. Let P_i be a player that uses the facility copy $(e, a_{\max}(e))$. Since $C_i = \max_{e \in E_i} n_{e,a_i}(A)$ it must hold that $n_{e,a_{\max}(e)}(A) \leq C_i(A)$. In fact we can state the following general property:

$$\forall e \in F,\ \exists i \in [N] : n_{e,a_{\max}(e)} \leq C_i(A) \tag{4.20}$$

From the above sections we know that $C_i(A) \leq \frac{N}{W}$. Moreover

$$SC_3(\text{OPT}) \geq \sum_{e \in E} \frac{\sum_{a \in [W]} n_{e,a}(\text{OPT})}{W} = \sum_{i \in [N]} \frac{|\text{OPT}_i|}{W} \geq \frac{N}{W} \tag{4.21}$$

From the above we can conclude:

$$\frac{SC_3(A)}{SC_3(\text{OPT})} \leq |E_A| \leq F \tag{4.22}$$

□ □

Theorem 39. *There exists a class of CBGs with* PoA $= F - 1$*.*

Proof. The lower bound is a Network Colored Bottleneck Game on a ring network. We have a ring of size $|F|$. We have W players that want to route traffic between two subsequent nodes of the ring. In the optimal allocation all players route through the direct edge, which leads to a Social Cost 1. On the other hand it is a PNE for players to route their traffic from the other side of the ring using all $F - 1$ edges and divided in the W colors. In that case the Social Cost is $F - 1$ leading to the desired PoA. □ □

Bibliography

[AAE$^+$08] Baruch Awerbuch, Yossi Azar, Amir Epstein, Vahab S. Mirrokni, and Alexander Skopalik, *Fast convergence to nearly optimal solutions in potential games*, ACM Conference on Electronic Commerce, 2008, pp. 264–273.

[AKT08] Itai Ashlagi, Piotr Krysta, and Moshe Tennenholtz, *Social context games.*, WINE (Christos H. Papadimitriou and Shuzhong Zhang, eds.), Lecture Notes in Computer Science, vol. 5385, Springer, 2008, pp. 675–683.

[ARV06a] Heiner Ackermann, Heiko Röglin, and Berthold Vöcking, *On the impact of combinatorial structure on congestion games*, FOCS, 2006, pp. 613–622.

[ARV06b] ______, *Pure nash equilibria in player-specific and weighted congestion games*, WINE, 2006, pp. 50–61.

[Aum74] Robert J. Aumann, *Subjectivity and correlation in randomized strategies*, Journal of Mathematical Economics **1** (1974), no. 1, 67–96.

[Aum87] ______, *Correlated equilibrium as an expression of bayesian rationality*, Econometrica **55** (1987), no. 1, 1–18.

[Bas07] K Basu, *The traveler's dilemma*, Sci Am (2007).

[BFFM08] Vittorio Bilò, Angelo Fanelli, Michele Flammini, and Luca Moscardelli, *Graphical congestion games*, WINE, 2008, pp. 70–81.

[BFM05] Vittorio Bilò, Michele Flammini, and Luca Moscardelli, *On nash equilibria in non-cooperative all-optical networks.*, STACS, 2005, pp. 448–459.

[BHLR07] Avrim Blum, Mohammadtaghi Hajiaghayi, Katrina Ligett, and Aaron Roth, *Regret minimization and the price of total anarchy*, Tech. report, In STOC b••08: Proceedings of the fortieth annual ACM symposium on Theory of computing, 2007.

[BM04] Vittorio Bilò and Luca Moscardelli, *The price of anarchy in all-optical networks.*, SIROCCO, 2004, pp. 13–22.

[BMI06] Costas Busch and Malik Magdon-Ismail, *Atomic routing games on maximum congestion*, AAIM, 2006, pp. 79–91.

[BO06] Ron Banner and Ariel Orda, *Bottleneck routing games in communication networks*, INFOCOM, 2006.

[BPPP08] Evangelos Bampas, Aris Pagourtzis, George Pierrakos, and Katerina Potika, *On a non-cooperative model for wavelength assignment in multifiber optical networks*, ISAAC, 2008, pp. 159–170.

[BPPS09] Evangelos Bampas, Aris Pagourtzis, George Pierrakos, and Vasileios Syrgkanis, *Colored resource allocation games*, Cologne-Twente Workshop on Graphs and Combinatorial Optimization, 2009.

[CD05a] Xi Chen and Xiaotie Deng, *3-nash is ppad-complete*, Electronic Colloquium on Computational Complexity (ECCC) (2005), no. 134.

[CD05b] ———, *Settling the complexity of 2-player nash-equilibrium*, Electronic Colloquium on Computational Complexity (ECCC) (2005), no. 140.

[CD06] ———, *Settling the complexity of two-player nash equilibrium*, Foundations of Computer Science, Annual IEEE Symposium on **0** (2006), 261–272.

[CK05] George Christodoulou and Elias Koutsoupias, *The price of anarchy of finite congestion games*, In Proceedings of the 37th Annual ACM Symposium on Theory of Computing (STOC, 2005, pp. 67–73.

[CS07] Steve Chien and Alistair Sinclair, *Convergence to approximate nash equilibria in congestion games*, SODA, 2007, pp. 169–178.

[Das08] Constantinos Daskalakis, *The complexity of nash equilibria*, Ph.D. thesis, EECS Department, University of California, Berkeley, Aug 2008.

[DGP05] Konstantinos Daskalakis, Paul W. Goldberg, and Christos H. Papadimitriou, *The complexity of computing a nash equilibrium*, Electronic Colloquium on Computational Complexity (ECCC) (2005), no. 115.

[DGP06] Constantinos Daskalakis, Paul W. Goldberg, and Christos H. Papadimitriou, *The complexity of computing a nash equilibrium*, STOC '06: Proceedings of the thirty-eighth annual ACM symposium on Theory of computing (New York, NY, USA), ACM, 2006, pp. 71–78.

[DHR00] Yevgeniy Dodis, Shai Halevi, and Tal Rabin, *A cryptographic solution to a game theoretic problem*, CRYPTO '00: Proceedings of the 20th Annual International Cryptology Conference on Advances in Cryptology (London, UK), Springer-Verlag, 2000, pp. 112–130.

[DP05] Konstantinos Daskalakis and Christos H. Papadimitriou, *Three-player games are hard*, Electronic Colloquium on Computational Complexity (ECCC) (2005), no. 139.

[DP08] Constantinos Daskalakis and Christos H. Papadimitriou, *Discretized multinomial distributions and nash equilibria in anonymous games*, FOCS, 2008, pp. 25–34.

[DR07] Yevgeniy Dodis and Tal Rabin, *Chapter 8 - cryptography and game theory*, Algorithmic Game Theory (Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay Vazirani, eds.), Cambridge University Press, 2007, pp. 209–242.

[EY07] Kousha Etessami and Mihalis Yannakakis, *On the complexity of nash equilibria and other fixed points*, Proceedings of 48th IEEE Symp. on Foundations of Computer Science (FOCS), 2007., IEEE, Oct 2007.

[FFM+06] Angelo Fanelli, Michele Flammini, Giovanna Melideo, Luca Moscardelli, and Alfredo Navarra, *Game theoretical issues in optical networks*, In Proceedings of the 8th International Conference on Transparent Optical Networks (ICTON 2006), Nottingham, UK, vol. 3, 2006, pp. 243–247.

[FKS05a] Dimitris Fotakis, Spyros C. Kontogiannis, and Paul G. Spirakis, *Selfish unsplittable flows*, Theor. Comput. Sci. **348** (2005), no. 2-3, 226–239.

[FKS05b] ______, *Symmetry in network congestion games: Pure equilibria and anarchy cost*, WAOA, 2005, pp. 161–175.

[FKS08] Dimitris Fotakis, Alexis C. Kaporis, and Paul G. Spirakis, *Atomic congestion games: Fast, myopic and concurrent*, SAGT, 2008, pp. 121–132.

[FMM+08] Michele Flammini, Gianpiero Monaco, Luca Moscardelli, Mordechai Shalom, and Shmuel Zaks, *Selfishness, collusion and power of local search for the adms minimization problem*, Computer Networks **52** (2008), no. 9, 1721–1731.

[FP08] Alex Fabrikant and Christos H. Papadimitriou, *The complexity of game dynamics: Bgp oscillations, sink equilibria, and beyond*, SODA '08: Proceedings of the nineteenth annual ACM-SIAM symposium on Discrete algorithms (Philadelphia, PA, USA), Society for Industrial and Applied Mathematics, 2008, pp. 844–853.

[FPT04] Alex Fabrikant, Christos Papadimitriou, and Kunal Talwar, *The complexity of pure nash equilibria*, STOC '04: Proceedings of the thirty-sixth annual ACM symposium on Theory of computing (New York, NY, USA), ACM Press, 2004, pp. 604–612.

[GKS05] George F. Georgakopoulos, Dimitris J. Kavvadias, and Leonidas G. Sioutis, *Nash equilibria in all-optical networks.*, WINE, 2005, pp. 1033–1045.

[GLMM04] Martin Gairing, Thomas Lücking, Marios Mavronicolas, and Burkhard Monien, *Computing nash equilibria for scheduling on restricted parallel links*, STOC, 2004, pp. 613–622.

[GMT06] Martin Gairing, Burkhard Monien, and Karsten Tiemann, *Routing (un-) splittable flow in games with player-specific linear latency functions*, ICALP (1), 2006, pp. 501–512.

[GMV05] M. Goemans, Vahab Mirrokni, and A. Vetta, *Sink equilibria and convergence*, Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science (FOCS), 2005, pp. 142–151.

[Hal08] Joseph Y. Halpern, *Beyond nash equilibrium: Solution concepts for the 21st century.*

[HKM09] T. Harks, M. Klimm, and R. Mohring, *Characterizing the existence of potential functions in weighted congestion games.*

[HP08a] Joseph Y. Halpern and Rafael Pass, *Game theory with costly computation*, 2008.

[HP08b] ______, *Iterated regret minimization: A more realistic solution concept*, 2008.

[HTW06] Ara Hayrapetyan, Éva Tardos, and Tom Wexler, *The effect of collusion in congestion games*, STOC, 2006, pp. 89–98.

[JPY88] David S. Johnson, Christos H. Papadimtriou, and Mihalis Yannakakis, *How easy is local search?*, J. Comput. Syst. Sci. **37** (1988), no. 1, 79–100.

[Kak41] Shizuo Kakutani, *A generalization of brouwer's fixed point theorem*, Duke Math. J. **8** (1941), 457–459.

[KP99] Elias Koutsoupias and Christos H. Papadimitriou, *Worst-case equilibria.*, STACS, 1999, pp. 404–413.

[KPT09] Bobby Kleinberg, George Piliouras, and Eva Tardos, *Multiplicative updates outperform generic no-regret learning in congestion games*, STOC, 2009.

[Kuk04] Nikolai S. Kukushkin, *Congestion games revisited*, Game Theory and Information 0412010, EconWPA, December 2004.

[LMM03] Richard J. Lipton, Evangelos Markakis, and Aranyak Mehta, *Playing large games using simple strategies*, ACM Conference on Electronic Commerce, 2003, pp. 36–41.

[Mac] D.W. MacKenzie, *The economics of groundhog day.*

[Mil96] Igal Milchtaich, *Congestion games with player-specific payoff functions*, Games and Economic Behavior **13** (1996), no. 1, 111 – 124.

[Mon07] Dov Monderer, *Multipotential games*, IJCAI, 2007, pp. 1422–1427.

[MPP07] Ioannis Milis, Aris Pagourtzis, and Katerina Potika, *Selfish routing and path coloring in all-optical networks*, CAAN, Lecture Notes in Computer Science, vol. 4852, 2007, pp. 71–84.

[MR96] Rajeev Motwani and Prabhakar Raghavan, *Randomized algorithms*, ACM Comput. Surv. **28** (1996), no. 1, 33–37.

[MS96] D. Monderer and L. S. Shapley, *Potential games*, Games and Economic Behavior **14** (1996), 124–143.

[MY89] Nimrod Megiddo and Christos H. Papadimitriou Y, *Anoteontotal functions, existence theorems, and computational complexity*, 1989.

[Nas50] John F. Nash, *Equilibrium points in n-person games*, Proceedings of the National Academy of Sciences of the United States of America **36** (1950), no. 1, 48–49.

[Nas51] John Nash, *Non-cooperative games*, The Annals of Mathematics **54** (1951), no. 2, 286–295.

[NM44] John Von Neumann and Oskar Morgenstern, *Theory of games and economic behavior*, Princeton University Press, 1944.

[NR99] Noam Nisan and Amir Ronen, *Algorithmic mechanism design (extended abstract)*, STOC '99: Proceedings of the thirty-first annual ACM symposium on Theory of computing (New York, NY, USA), ACM Press, 1999, pp. 129–140.

[OR94] M.J. Osborne and A. Rubinstein, *A course in game theory*, The MIT Press, 1994.

[Pap94] Christos H. Papadimitriou, *On the complexity of the parity argument and other inefficient proofs of existence*, J. Comput. Syst. Sci. **48** (1994), no. 3, 498–532.

[Pap01] Christos H. Papadimitriou, *Algorithms, games and the internet*, In STOC, ACM Press, 2001, pp. 749–753.

[Pap08] Christos H. Papadimitriou, *The search for equilibrium concepts*, SAGT '08: Proceedings of the 1st International Symposium on Algorithmic Game Theory (Berlin, Heidelberg), Springer-Verlag, 2008, pp. 1–3.

[PS05] Panagiota N. Panagopoulou and Paul G. Spirakis, *Efficient convergence to pure nash equilibria in weighted network congestion games*, WEA, 2005, pp. 203–215.

[PSY90] C. H. Papadimitriou, A. A. Schäffer, and M. Yannakakis, *On the complexity of local search*, STOC '90: Proceedings of the twenty-second annual ACM symposium on Theory of computing (New York, NY, USA), ACM, 1990, pp. 438–445.

[PY94] Christos H. Papadimitriou and Mihalis Yannakakis, *On complexity as bounded rationality (extended abstract)*, STOC '94: Proceedings of the twenty-sixth annual ACM symposium on Theory of computing (New York, NY, USA), ACM, 1994, pp. 726–733.

[Ros73] R. W. Rosenthal, *A class of games possessing pure-strategy nash equilibria*, Int. J. Game Theory **2** (1973), 65–67.

[Rub86] Ariel Rubinstein, *Finite automata play the repeated prisoner's dilemma*, Journal of Economic Theory **39** (1986), no. 1, 83–96.

[Sho08] Yoav Shoham, *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*, Cambridge University Press, December 2008.

[SV08] Alexander Skopalik and Berthold Vöcking, *Inapproximability of pure nash equilibria*, STOC '08: Proceedings of the 40th annual ACM symposium on Theory of computing (New York, NY, USA), ACM, 2008, pp. 355–364.

[SY91] Alejandro A. Schäffer and Mihalis Yannakakis, *Simple local search problems that are hard to solve*, SIAM J. Comput. **20** (1991), no. 1, 56–87.

[Ten04] Moshe Tennenholtz, *Program equilibrium*, Games and Economic Behavior **49** (2004), no. 2, 363–373.

[VBM99] M. Voorneveld, P. Borm, and F. van Megen, *Congestion games and potentials reconsidered*, Discussion Paper 98, Tilburg University, Center for Economic Research, 1999.

[Vet02] Adrian Vetta, *Nash equilibria in competitive societies, with applications to facility location, traffic routing and auctions*, Foundations of Computer Science, Annual IEEE Symposium on **0** (2002), 416.

[Vöc06] Berthold Vöcking, *Congestion games: Optimization in competition*, ACiD, 2006, pp. 9–20.

[War52] J. Wardrop, *Some theoretical aspects of road traffic research*, Proceedings of the Institution of Civil Engineers, Part II **1** (1952), no. 36, 352–362.

[WG05] Robert B. Wilson and Srihari Govindan, *Refinements of nash equilibria*, 2005.