Εθνικό Μετσόβιο Πολυτεχνείο
Σχολη Ηλεκτρολογων Μηχανικων
και Μηχανικων Υπολογιστων
Εργαστηριο Λογικης &
Επιστημης Υπολογισμων
CoReLab

Παίγνια
και
Ιδιοτελής Συμπεριφορά σε Δίκτυα

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΓΕΩΡΓΙΟΥ Ε. ΠΙΕΡΡΑΚΟΥ

Φοιτητής Ηλεκτρολόγος Μηχανικός &
Μηχανικός Υπολογιστών Ε.Μ.Π. (2002)

Επιβλέπων Καθηγητής: Ε. Ζάχος

Αθήνα, Ιούλιος 2007

Εθνικο Μετσοβιο Πολυτεχνειο
Σχολη Ηλεκτρολογων Μηχανικων &
και Μηχανικων Υπολογιστων
Εργαστηριο Λογικης &Υπολογισμων
CoReLab

Παίγνια
και
Ιδιοτελής Συμπεριφορά σε Δίκτυα

Διπλωματική Εργασία

του

ΓΕΩΡΓΙΟΥ Ε. ΠΙΕΡΡΑΚΟΥ

Φοιτητής Ηλεκτρολόγος Μηχανικός &
Μηχανικός Υπολογιστών Ε.Μ.Π. (2002)

Επιτροπή: Ευστάθιος Ζάχος
Ηλίας Κουτσουπιάς
Άρης Παγουρτζής

...	...	...
Ε. Ζάχος	Η. Κουτσουπιάς	Α. Παγουρτζής
Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Κ.Π.Α.	Λέκτορας Ε.Μ.Π.

Αθήνα, Ιούλιος 2007

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

Περίληψη

Το αντικείμενο της παρούσας διπλωματικής εργασίας είναι η μελέτη καταστάσεων στις οποίες πολλοί χρήστες αλληλεπιδρούν μεταξύ τους, υπό την απουσία κάποιας εξωτερικής ρυθμιστικής αρχής, με μόνο γνώμονα ο καθένας το προσωπικό του όφελος. Τέτοιες καταστάσεις είναι συνηθισμένες σε μεγάλα, κατανεμημένα συστήματα και δίκτυα, με χαρακτηριστικότερο παράδειγμα αυτό του Internet. Τα συστήματα αυτά, που χαρακτηρίζονται από ιδιοτελή συμπεριφορά χρηστών, αποτελούν παραδοσιακά αντικείμενο μελέτης της Θεωρίας Παιγνίων. Στη διπλωματική αυτή παρουσιάζουμε κάποιες βασικές έννοιες της Θεωρίας Παιγνίων και στη συνέχεια προχωράμε στη μελέτη τριών μοντέλων που έχουν προταθεί για την αναπαράσταση των συστημάτων αυτών. Τα μοντέλα αυτά είναι: πρώτον, το δίκτυο παράλληλων ακμών που πρωτομελετήθηκε στο [KP99] και ακολουθήθηκε από μία σειρά από άλλες δημοσιεύσεις που επέλυσαν διάφορα ανοιχτά προβλήματα. Δεύτερον, το μοντέλο των παιγνίων συμφόρησης, το οποίο έχει μελετηθεί ανεξάρτητα από το προηγούμενο μοντέλο (που αποτελεί υποπερίπτωση παιγνίου συμφόρησης) και το οποίο μπορεί να μοντελοποιήσει καταστάσεις δρομολόγησης κίνησης μέσα σε δίκτυα χρηστών ή καταστάσεις όπου οι χρήστες δεσμεύουν τους πόρους κάποιου συστήματος. Τέλος, το τρίτο μοντέλο είναι ένα μοντέλο απειροστής ροής, που έχει μελετηθεί κυρίως από τους Roughgarden και Tardos, ως η μη-ατομική επέκταση των παιγνίων συμφόρησης. Για κάθε μοντέλο που μελετάμε, εξετάζουμε δύο βασικά θέματα: αυτό της ύπαρξης και της υπολογισιμότητας των ισορροπιών Nash και αυτό των φραγμάτων για το τίμημα της αναρχίας, που ουσιαστικά ποσοτικοποιεί τις απώλειες που έχουμε λόγω της ιδιοτελούς συμπεριφοράς των χρηστών.

Λέξεις κλειδιά: παίγνια, ισορροπία Nash, τίμημα της αναρχίας, δίκτυα, ιδιοτελής δρομολόγηση, δέσμευση πόρων, ιδιοτελής συμπεριφορά, παίγνια συμφόρησης, παίγνια σε δίκτυα παράλληλων ακμών, μη-ατομικά παίγνια, παράδοξο του Braess

Abstract

This diploma thesis studies situations where many users interact, under the absence of some central regulatory authority, each one aiming at the maximization of his own personal profit. These situations are common in large, distributed networks and systems, with the Internet being an obvious example. These systems, which are characterized by selfish user behavior, are traditionally a field of study for Game Theory. In this thesis we present some basic concepts of Game Theory and we then move on to presenting three models that have been proposed for these systems. These models are: first, the network of parallel links originally studied in [KP99], which was followed by a series of papers resolving various open problems. Second, the model of congestion games, which has been studied independently of the previous model (which is in fact a subcase of congestion games) and which models situations where users want to route traffic through a network or want to allocate the resources of a system. Finally, our third model is the non-atomic extension of congestion games, mostly studied by Roughgarden and Tardos. For each model we discuss, we focus on 2 main questions: that of the existence and tractability of equilibria, and that of the effective bounding of the Price of Anarchy, which quantifies the inefficiency due to the selfish behavior of the users (i.e. due to the lack of coordination).

Key words: games, Nash equilibrium, Price of Anarchy, networks, selfish routing, resource allocation, selfish behavior, congestion games, games in parallel links networks, non-atomic games, Braess's paradox

Ευχαριστίες

Ολοκλρώνοντας την εκπόνηση της διπλωματικής μου εργασίας και μαζί με αυτήν και τον κύκλο σπουδών μου στο Ε.Μ.Π. θα ήθελα να ευχαριστήσω θερμά μία σειρά από ανθρώπους που με βοήθησαν καθόλη τη διάρκεια της πορείας μου αυτής.

Καταρχάς τον καθηγητή και επιβλέποντα της διπλωματικής κ. Στάθη Ζάχο, που ήταν και ο άνθρωπος που μου μετέδωσε την αγάπη του για τη Θεωρητική Πληροφορική, με τις διδακτικές του ικανότητες και τον ενθουσιασμό του. Τον λέκτορα κ. Άρη Παγουρτζή για τις πολλές και σημαντικές παρατηρήσεις του κατά την πορεία συγγραφής της διπλωματικής. Τον καθηγητή κ. Ηλία Κουτσουπιά, πάνω σε δημοσιεύσεις του οποίου βασίστηκε μεγάλο μέρος της διπλωματικής, ο οποίος ήταν πάντα πρόθυμος να με βοηθήσει με ό,τι απορίες είχα, ενώ παράλληλα το αντίστοιχο μάθημά του με βοήθησε να αποκτήσω μία άλλη οπτική στο θέμα της Θεωρίας Παιγνίων. Ακόμα, τον επίκουρο καθηγητή κ. Σπύρο Κοντογιάννη, μία ομιλία του οποίου, μου έδωσε το έναυσμα να ασχοληθώ με το θέμα αυτό.

Θα ήθελα να ευχαριστήσω όλους τους φίλους, συμφοιτητές και μη, που είναι δίπλα μου όλα αυτά τα χρόνια για να με στηρίξουν και να με συμβουλέψουν όποτε το έχω ανάγκη.

Τον μαθηματικό μου, κ. Χάρη, για την αγάπη που μου έδειξε στα λυκειακά μου χρόνια, για τον έρωτα που μου ενέπνευσε για τα μαθηματικά και για την εμπιστοσύνη που είχε πάντα σε μένα.

Και κυρίως τη μητέρα μου που είναι πάντα δίπλα μου, για να με στηρίζει και να με βοηθά, για την αγάπη με την οποία με περιβάλλει και για όλα όσα έχει κάνει για μένα.

Contents

1 Θεωρία Παιγνίων: Βασικές έννοιες **11**
- 1.1 Γενικά για τη Θεωρία Παιγνίων 11
- 1.2 Nash Equilibrium και Price of Anarchy 14
  - 1.2.1 Ορισμός Παιγνίου και Παραδείγματα 14
  - 1.2.2 Ισορροπίες σε Παίγνια 16
  - 1.2.3 Το Price of Anarchy 21
- 1.3 Μερικά παράδοξα της Θεωρίας Παιγνίων 23
- 1.4 Θεωρητική Πληροφορική και Θεωρία Παιγνίων 26

I The atomic case **29**

2 The Koutsoupias-Papadimitriou model **31**
- 2.1 The Model 32
- 2.2 Nash Equilibria 33
  - 2.2.1 Definitions 33
  - 2.2.2 The Fully Mixed Nash Equilibrium 35
  - 2.2.3 The Generalized Fully Mixed Nash Equilibrium 39
  - 2.2.4 The Pure Nash Equilibrium 42
- 2.3 Studying the Price of Anarchy 44
  - 2.3.1 Definitions 45
  - 2.3.2 The case of 2 links 48
  - 2.3.3 The case of m links 51

3 Congestion Games **59**
- 3.1 Definitions 59
- 3.2 Equilibria 61
  - 3.2.1 Existence of Equilibria 61
  - 3.2.2 Computing a PNE in various congestion games 63
  - 3.2.3 Some results on weighted congestion games 67
- 3.3 Studying the Price of Anarchy 69

3.3.1 The Pure Price of Anarchy 69
3.3.2 The Mixed Price of Anarchy 71

II The non-atomic case 73

4 Selfish routing 75
4.1 The model 75
4.2 Bounding the Price of Anarchy 81
4.3 Bounding Braess 's Paradox 85
4.3.1 Enlarging the Paradox 86
4.3.2 Detecting Braess's Paradox is hard 91
4.4 Reducing the Inefficiency of Equilibria 94
4.4.1 Capacity augmentation 94
4.4.2 Stackelberg routing 95
4.4.3 Pricing network edges 96

List of Figures

1.1 2 NE σε Selfish Task Allocation instance 23
1.2 Pigou's Example 24
1.3 A nonlinear variant of Pigou's Example 25
1.4 Braess's Paradox 26

3.1 Complexity Hierarchy of Search Problems 64
3.2 An example with unbounded PoA 71

4.1 The Braess Graphs for k=2 and k=3 87
4.2 The WE for G and H 88
4.3 The reduction from 2DDP to LINEAR NETWORK DESIGN 92

List of Tables

1.1 Bach or Stravinsky 15
1.2 Mathcing pennies 15
1.3 Prisoners' Dilemma 16
1.4 Players' payoffs: an example 19

3.1 Main results of [CK05] 69

4.1 Main differences of KP-model and selfish routing 77
4.2 The Pigou bound for some important sets of cost functions . . 84

Chapter 1

Θεωρία Παιγνίων: Βασικές έννοιες

Στο εισαγωγικό αυτό κεφάλαιο θα παρουσιάσουμε κάποιες βασικές έννοιες της αλγοριθμικής θεωρίας παιγνίων. Αρχικά θα κάνουμε μία ιστορική αναδρομή και θα αναφερθούμε συνοπτικά στις βασικές έννοιες που καλύπτονται μέσα από τη θεωρία αυτή, καθώς και στους διάφορους τύπους παιγνίων που έχουν κατά καιρούς προταθεί. Στη συνέχεια θα ορίσουμε επίσημα τι είναι ένα παίγνιο, τι είναι ισορροπία και πώς μελετάμε την ποιότητά της και θα αναφέρουμε κατόπιν κάποια παράδοξα, τα οποία θα ξανασυναντήσουμε στα επόμενα κεφάλαια, μιας και έχουν αποτελέσει τροφή για αρκετή από την ερευνητική δουλειά που περιλαμβάνεται στη διπλωματική αυτή. Θα κλείσουμε το κεφάλαιο αυτό αναφέροντας κάποια από τα θέματα που άπτονται της Θεωρίας Παιγνίων και αποτελούν αντικείμενα έρευνας για την κοινότητα της Θεωρητικής Πληροφορικής.

1.1 Γενικά για τη Θεωρία Παιγνίων

Η Θεωρία Παιγνίων αναφέρεται συχνά ως ένας κλάδος των εφαρμοσμένων μαθηματικών και των οικονομικών, που περιγράφει καταστάσεις στις οποίες πολλοί παίκτες παίρνουν αποφάσεις με μόνο σκοπό να μεγιστοποιήσουν το προσωπικό τους όφελος. Στην παρούσα διπλωματική θα ασχοληθούμε με τη λεγόμενη "μη συνεργατική θεωρία παιγνίων" (non-coalitional game theory), όπου ο κάθε παίκτης δρα αυτόνομα και εγωιστικά για τον εαυτό του, δηλαδή δεν υπάρχουν συνασπισμοί μεταξύ των παικτών.
Πιο συγκεκριμένα τα παίγνια προσπαθούν να μοντελοποιήσουν καταστάσεις όπου αλληλεπιδρούν πολλά άτομα (οι παίκτες του παιγνίου) υπό τις εξής προϋποθέσεις:

- Οι παίκτες είναι **έξυπνοι** υπό την έννοια ότι, με δεδομένες τις κινήσεις των άλλων παικτών ξέρουν πάντα τι τους συμφέρει, καθώς και τι συμφέρει τους άλλους παίκτες (το οποίο στην πράξη σημαίνει ότι μπορούν να *εκτιμήσουν* τις κινήσεις των -έξυπνων- συμπαικτών τους). Με βάση τις εκτιμήσεις αυτές παίζουν **ορθολογιστικά** (**rationally**) και αποφασίζουν τις επόμενες κινήσεις τους στρατηγικά ("reason strategically" [OR94])

- Οι παίκτες πράττουν πάντα το καλύτερο για τους ίδιους, αδιαφορώντας για τις συνέπειες της επιλογής τους στο κοινωνικό σύνολο (είναι δηλαδή **εγωιστές - selfish**)

- Δεν υπάρχει κάποια κεντρική αρχή που να μπορεί να κατευθύνει τις επιλογές των παικτών, δηλαδή οι παίκτες δρουν αυτόνομα με μόνο κριτήριο το συμφέρον τους.

- Οι επιλογές ενός παίκτη επηρεάζουν α) την ευημερία της κοινωνίας (social welfare) της οποίας είναι μέλος και β) τις επιλογές των υπολοίπων παικτών

Από την παραπάνω περιγραφή γίνεται σαφές ότι η Θεωρία Παιγνίων είναι σε θέση να περιγράψει πολλές καταστάσεις της καθημερινής ζωής και κυρίως να μοντελοποιήσει συστήματα στα οποία απουσιάζει η κεντρική διαχείριση. Ένα τέτοιο σύστημα είναι και το Διαδίκτυο: αποτελείται από εκατοντάδες χιλιάδες χρήστες (τερματικά ή εξυπηρετητές) όπου ο κάθε ένας δρα με μόνο κριτήριο την ελαχιστοποίηση της δικής του καθυστέρησης, χωρίς να υπάρχει κάποια κεντρική αρχή διαχείρισης η οποία να αποσκοπεί στην επίτευξη του "κοινωνικού βέλτιστου". Η εφαρμογή που βρίσκει η Θεωρία Παιγνίων στη μοντελοποίηση του Διαδικτύου αποτελεί από μόνη της ικανό λόγο για την τεράστια προσοχή που έχει προσελκύσει το αντικείμενο τα τελευταία χρόνια. Ενδεικτικές πληροφορίες μπορεί κανείς να βρει στο [Pap01]
Λαμβάνοντας όμως υπόψιν ότι το Διαδίκτυο είναι ένα τεχνολογικό κατασκεύασμα της τελευταίας εικοσαετίας, ενώ η πρώτη αναφορά στη Θεωρία Παιγνίων γίνεται το 1944 από τους John von Neumann και Oskar Morgenstern, αντιλαμβάνεται κανείς ότι οι εφαρμογές του κλάδου αυτού πρέπει να εκτείνονται και πέρα από αυτό. Όντως ήδη από την αρχή του '70 η Θεωρία Παιγνίων βρήκε εφαρμογή στην ανάλυση της συμπεριφοράς των ζώων, στην εξελικτική θεωρία, στην πολιτική επιστήμη και ηθική και κυρίως στα οικονομικά και στην κοινωνιολογία. Έχει χρησιμοποιηθεί μάλιστα ευρέως για τη μελέτη των ολιγοπωλιακών οικονομιών, του πολιτικού ανταγωνισμού και γενικά καταστάσεων όπου υπάρχει λήψη αποφάσεων, υπό την προϋπόθεση ότι οι αποφάσεις του ενός ατόμου εξαρτώνται από τις αποφάσεις των υπολοίπων.

Στη βιβλιογραφία έχουν προταθεί πολλά είδη παιγνίων. Στη συνέχεια θα αναφερθούμε επιγραμματικά στις βασικές διαφορές μεταξύ των πιο γνωστών κατηγοριών παιγνίων.

- **Strategic Games vs Extensive Games.** Ένα στρατηγικό παίγνιο είναι ένα μοντέλο μίας κατάστασης στην οποία κάθε παίκτης αποφασίζει τι θα κάνει (ποια ενέργεια θα διαλέξει) μία φορά και όλοι οι παίκτες αποφασίζουν ταυτόχρονα (δηλαδή όταν ο κάθε παίκτης λαμβάνει την απόφασή του δεν ξέρει τι έχουν αποφασίσει οι υπόλοιποι παίκτες). Μόλις διαλέξουν όλοι οι παίκτες κίνηση, αποτιμάται το αποτέλεσμα και ο κάθε παίκτης μπορεί να έχει ή όχι συμφέρον να αλλάξει την επιλογή του. Παρόλα αυτά, η ενδεχόμενη αλλαγή της επιλογής κάποιου παίκτη (με ό,τι αυτό συνεπάγεται για τις επιλογές των υπολοίπων παικτών: π.χ. διαδοχικά όλοι οι παίκτες μπορεί να θέλουν να αλλάξουν την επιλογή τους) δεν μας ενδιαφέρει στην περίπτωση των στρατηγικών παιγνίων. Αντίθετα το μοντέλο των extensive games μελετά ακριβώς αυτή την πιθανή αλληλουχία γεγονότων: κάθε παίκτης μπορεί να επιλέξει την κίνησή του όχι μόνο στο ξεκίνημα του παιγνίου, αλλά να κάνει επιλογές και καθ' όλη τη διάρκεια εξέλιξής του[1].

- **Noncooperative vs Cooperative Games.** Η μόνη διαφορά εδώ είναι ότι στα συνεργατικά παίγνια θεωρούμε ότι υπάρχουν συνασπισμοί μεταξύ των παικτών, δηλαδή οι παίκτες δρουν αφενός με σκοπό το προσωπικό όφελος και αφετέρου το όφελος της ομάδας στην οποία ανήκουν. Αξίζει να σημειωθεί ότι ένας συνασπισμός δε συμπεριφέρεται σαν ένας παίκτης.

- **Games with Perfect and Imperfect Information.** Οι παίκτες ενός παιγνίου μπορεί να ξέρουν τα πάντα ο ένας για τις κινήσεις/επιλογές του άλλου, ή και όχι. Σε πολλές περιπτώσεις μπορεί μάλιστα οι παίκτες να μην ξέρουν ούτε τα βασικά "χαρακτηριστικά" των συμπαικτών τους, όπως για παράδειγμα το πόσο "αξίζει" για κάποιον συμπαίκτη τους ένα αγαθό, ή ποιες είναι οι στρατηγικές του.

Εκτός από τις παραπάνω βασικές κατηγορίες, έχουμε και διάφορες άλλες υποκατηγορίες παιγνίων, όπως τα **συμμετρικά παίγνια**, στα οποία όλοι οι παίκτες έχουν τις ίδιες στρατηγικές και τα ίδια κέρδη ανά στρατηγική, τα **zero-sum παίγνια**, όπου ο κάθε παίκτης κερδίζει ότι χάνουν οι υπόλοιποι, έτσι ώστε οι συνολικές απολαβές να είναι σταθερές κ.α. Εξάλλου, ανάλογα με τις καταστάσεις που μοντελοποιεί ένα παίγνιο, μιλάμε για εξελικτικά παίγνια (evolutionary games), παιχνίδια σε δίκτυα (network games), παιχνίδια συμφόρησης (congestion games), selfish task allocation games κ.α.

[1] αντίθετα με τα στρατηγικά παίγνια που δεν έχουν διάρκεια

1.2 Nash Equilibrium και Price of Anarchy

1.2.1 Ορισμός Παιγνίου και Παραδείγματα

Ας προχωρήσουμε τώρα στον τυπικό ορισμό ενός παιγνίου.

Definition 1.2.1. Παίγνιο πολλών παικτών:
Ορίζουμε ως παίγνιο πολλών παικτών την τριάδα $(N, (A_i)_{i\in N}, (\succsim_i)_{i\in N})$ όπου

- N είναι ένα πεπερασμένο[2] σύνολο (το σύνολο των παικτών).
- Σε κάθε παίκτη $i \in N$ αντιστοιχούμε το μη κενό σύνολο A_i (σύνολο διαθέσιμων στρατηγικών για κάθε παίκτη).
- Σε κάθε παίκτη $i \in N$ αντιστοιχούμε μία σχέση προτίμησης $\succsim_i$ επί του συνόλου $A = \times_{j\in N} A_j$. Σχέση προτίμησης επί ενός συνόλου A είναι μία δυαδική, πλήρης (δηλαδή κάθε δύο στοιχεία του A σχετίζονται μεταξύ τους), ανακλαστική και μεταβατική σχέση επί του A.

Στην πράξη, οι παραπάνω σχέσεις προτίμησεις δίνονται υπό τη μορφή συναρτήσεων κέρδους (payoff functions) u_i ή συναρτήσεων κόστους c_i (cost functions) οι οποίες ορίζονται ως εξής: $c_i, u_i : A_1 \times \ldots \times A_n \to \mathbb{R}_+^*, i = 1 \ldots n$. Έτσι κάθε παίκτης αντιστοιχεί σε κάθε tuple στρατηγικών $(a_1, \ldots, a_n)$, το οποίο πλέον θα αποκαλούμε **προφίλ (γνήσιων) στρατηγικών**, μία τιμή η οποία είναι το κέρδος του (κόστος του) αν όλοι παίκτες παίξουν τις στρατηγικές που υπαγορεύονται από το προφίλ. Προφανώς ένας παίκτης προτιμά ένα προφίλ έναντι ενός άλλου, αν το κέρδος(κόστος) του για το προφίλ αυτό είναι μεγαλύτερο (μικρότερο).

Ας διευκρινίσουμε τώρα τον τρόπο με τον οποίο παίζουν οι παίκτες: το αναμενόμενο είναι κάθε παίκτης να επιλέγει μία στρατηγική από το σύνολο στρατηγικών του. Ο τρόπος αυτός παιξίματος, αν και διαισθητικά είναι ξεκάθαρος, έχει ένα σημαντικό πρόβλημα όπως θα δούμε στη συνέχεια. Ας δούμε όμως πιο πριν κάποια παραδείγματα παιγνίων 2 παικτών και πώς αυτά παριστάνονται.

Example 1.2.2. Bach or Stravinsky
Στο παράδειγμα αυτό υποθέτουμε ότι έχουμε ένα ζευγάρι, που θέλει να βγει έξω ένα βράδυ για να ακούσει ένα κονσέρτο κλασσικής μουσικής. Ο άνδρας προτιμά ένα κονσέρτο του Bach ενώ η γυναίκα προτιμά μία συμφωνία του Stravinsky. Και οι δύο όμως προτιμούν να μην πάνε στο θέαμα της επιλογής

[2] Προς το παρόν υποθέτουμε ότι εξετάζουμε πεπερασμένα παίγνια. Σε αντίθετη περίπτωση το N δε χρειάζεται να είναι πεπερασμένο.

τους, προκειμένου να είναι μαζί με το/τη σύντροφό τους.
Το παραπάνω σενάριο το οπτικοποιούμε συνήθως με τη χρήση του παρακάτω πίνακα: Οι δύο παίκτες (ο άνδρας και η γυναίκα εδώ) επιλέγουν ο μεν ένας

	B	S
B	2,1	0,0
S	0,0	1,2

Table 1.1: Bach or Stravinsky

τη γραμμή του πίνακα, ο δε άλλος τη στήλη. Ανάλογα με το συνδυασμό γραμμής - στήλης έχουμε το αποτέλεσμα. Εδώ τα δυνατά αποτελέσματα (προφίλ στρατηγικών) είναι 4 και τα κέρδη (ας το φανταστούμε σα μέτρο της ευχαρίστησης του κάθε παίκτη) που αντιστοιχούν στον καθένα φαίνονται στα κελιά. Πρώτο είναι το κέρδος του παίκτη γραμμή (άνδρας) και δεύτερο το κέρδος του παίκτη στήλη (γυναίκα). Παρατηρήστε ότι στο παίγνιο αυτό το σύνολο στρατηγικών είναι το ίδιο για τους 2 παίκτες.

Η παραπάνω οπτικοποίηση είναι πολύ συνηθισμένη για την περίπτωση των παιγνίων 2 παικτών και μπορεί να αναπαραστήσει και την περίπτωση όπου ο κάθε παίκτης έχει παραπάνω από 2 στρατηγικές.

Example 1.2.3. Στο δεύτερο παράδειγμα, γνωστό ως matching pennies, ο κάθε παίκτης έχει δύο στρατηγικές και διαλέγει μία. Αν και οι δύο παίκτες επιλέξουν την ίδια στρατηγική κερδίζει ο παίκτης γραμμή, αλλιώς κερδίζει ο παίκτης στήλη. Το παιχνίδι αυτό αντιστοιχεί στην κατάσταση όπου οι δύο παίκτες είναι ένας επιθετικός (παίκτης στήλη) και ένας τερματοφύλακας (παίκτης γραμμή) και ο επιθετικός ετοιμάζεται να εκτελέσει ένα πέναλτυ. Οι επιλογές που έχουν οι 2 παίκτες είναι κοινές: δεξιά ή αριστερά, που σημαίνει εκτέλεση του πέναλτυ στη δεξιά/αριστερή πλευρά και εκτίναξη δεξιά/αριστερά αντίστοιχα.
Ο αντίστοιχος πίνακας τώρα είναι:

	R	L
R	1,-1	-1,1
L	-1,1	1,-1

Table 1.2: Mathcing pennies

Το παίγνιο αυτό είναι zero-sum, δηλαδή ότι κερδίζει ο ένας παίκτης το χάνει ο άλλος.

Example 1.2.4. Το τελευταίο (και πιο διάσημο ίσως) παράδειγμα είναι το Δίλημμα των Φυλακισμένων (Prisoners' Dilemma) το οποίο λέει το εξής: έχουμε 2 ύποπτους για μία κλοπή και τους ανακρίνουμε στο αστυνομικό τμήμα, σε χωριστά κελιά. Οι αστυνομικοί τους δίνουν 2 επιλογές: είτε να μιλήσουν ή να σιωπήσουν. Επίσης λένε στον καθένα ότι αν αυτός μιλήσει και ο συνεταίρός του σιωπήσει θα τον αφήσουν ελεύθερο και τον άλλο θα τον βάλουν φυλακή για 4 χρόνια. Αν δε μιλήσει κανένας τότε λόγω έλλειψης στοιχείων θα τους βάλουν 1 χρόνο φυλακή τον καθένα ενώ αν μιλήσουν και οι 2 θα τους βάλουν 3 χρόνια φυλακή τον καθένα.
Ο αντίστοιχος πίνακας είναι:

	$Confess$	$Shh...$
$Confess$	3,3	0,4
$Shh...$	4,0	1,1

Table 1.3: Prisoners' Dilemma

Παρατηρήστε ότι στη συγκεκριμένη περίπτωση οι αριθμοί δε δείχνουν κέρδος αλλά κόστος.

1.2.2 Ισορροπίες σε Παίγνια

Ας προχωρήσουμε τώρα να ορίσουμε μία αποδεκτή έννοια ισορροπίας για τα παίγνια. Διαισθητικά περιμένουμε ότι ένα προφίλ στρατηγικών, δηλαδή ένα σύνολο από επιλογές, μία για κάθε παίκτη, θα είναι ισορροπία αν όλοι είναι ευχαριστημένοι. Ας υποθέσουμε ότι πράγματι όλοι είναι ευχαριστημένοι με την επιλογή τους και ότι δε θα την άλλαζαν, ό,τι και αν έκαναν οι υπόλοιποι συμπαίκτες τους. Αυτή είναι η έννοια του Dominant Equilibrium που ορίζεται ως εξής.

Definition 1.2.5. Ορίζουμε ως Dominant Equilibrium (DE) ένα προφίλ στρατηγικών $(a_1, \ldots, a_n)$ τέτοιο ώστε για κάθε παίκτη i:

$$c_i(a_1, \ldots, a_n) \leq c_i(a'_1, \ldots, a'_n)$$

για κάθε άλλο προφίλ στρατηγικών $(a'_1, \ldots, a'_n)$.

Προφανώς το DE είναι μία πολύ ισχυρή (και επιθυμητή) έννοια ισορροπίας και αντιστοιχεί σε ένα ολικό ελάχιστο των συναρτήσεων κόστους. Το δυσάρεστο είναι ότι τα περισσότερα παίγνια δε διαθέτουν DE [3]. Έτσι αναγκαζόμαστε να συμβιβαστούμε με μία πιο ασθενή έννοια, αυτή του Nash Equilibrium, που ορίστηκε από τον John Nash το 1951 στο [Nas51].

[3] είναι εύκολο να δει κανείς ότι κανένα από τα προαναφερθέντα παίγνια δεν έχει DE

Definition 1.2.6. Ορίζουμε ως γνήσιο (pure) Nash Equilbrium (PNE) ένα προφίλ στρατηγικών $(a_1, \ldots, a_n)$ τέτοιο ώστε για κάθε παίκτη i:

$$c_i(a_1, \ldots, a_n) \leq c_i(a_1, \ldots, a_i', \ldots a_n)$$

για κάθε άλλη στρατηγική a_i'.

Με άλλα λόγια, το NE είναι μία κατάσταση από την οποία κανείς παίκτης δεν έχει συμφέρον να φύγει, με δεδομένο ότι και οι υπόλοιποι παίκτες θα διατηρήσουν τις υπάρχουσες στρατηγικές τους. Είναι εύκολο να καταλάβει κανείς ότι το NE αντιστοιχεί σε τοπικό ελάχιστο των συναρτήσεων κόστους και αποτελεί μία πολύ λογική έννοια ισορροπίας. Αλλά το πιο σημαντικό για μία ισορροπία είναι να εξασφαλίσουμε την ύπαρξή της σε κάθε παίγνιο (κάτι το οποίο δεν καταφέραμε στην περίπτωση του DE). Αν προσέξουμε θα δούμε ότι στο Παράδειγμα 1.2.2 τα προφίλ (B,S) και (S,B) είναι και τα 2 PNE, ενώ αντίστοιχα στο Παράδειγμα 1.2.4 το προφίλ (Confess,Confess) είναι PNE. Αντίθετα το Παράδειγμα 1.2.3 δε διαθέτει PNE. Αυτό είναι απογοητευτικό. Δυστυχώς όμως είναι γεγονός ότι υπάρχουν παίγνια που δε διαθέτουν PNE. Προκειμένου να εξασφαλίσουμε καθολική ύπαρξη ισορροπίας, πρέπει να κάνουμε ένα ακόμα βήμα: να επιτρέψουμε στους παίκτες να παίζουν με πιθανότητες. Αυτό πρακτικά σημαίνει ότι ο κάθε παίκτης δεν επιλέγει πλέον μία στρατηγική από το σύνολο των στρατηγικών του, αλλά αποδίδει σε όλες μία πιθανότητα στο $[0, 1]$.
Έτσι αρχικά ορίζουμε το σύνολο

$$\Delta(A_i) \equiv \{z \in [0, 1]^{|A_i|} : \sum_i z_i = 1\}$$

. Το z είναι μία κατανομή πιθανότητας πάνω στα στοιχεία κάθε συνόλου A_i και το $\Delta(A_i)$ είναι το σύνολο όλων των δυνατών τέτοιων κατανομών. Ορίζουμε στη συνέχεια το **μεικτό προφίλ στρατηγικών** $\mathbf{p} = (p^1, \ldots, p^n)$, όπου $\forall i \in N, p_i \in \Delta(A_i)$. Θεωρώντας ότι έχουμε ορίσει τις συναρτήσεις κέρδους για κάθε παίκτη ορίζουμε ως $c^i_\alpha(\mathbf{p})$ το αναμενόμενο (expected) κόστος για τον παίκτη i, αν τελικά επιλέξει τη στρατηγική α και με δεδομένο ότι οι υπόλοιποι παίκτες παίζουν με βάση το μεικτό προφίλ $\mathbf{p}$. Έτσι παρατηρούμε ότι πλέον οι παίκτες δεν αποφασίζουνε με βάση το προκύπτον κόστος από τις επιλογές των υπολοίπων παικτών, αλλά με βάση το εκτιμώμενο κόστος.

Definition 1.2.7. Ως (μεικτό) Nash equilibrium ορίζουμε ένα μεικτό προφίλ στρατηγικών $\mathbf{p}$ τέτοιο ώστε:

$$\forall i \in N, \forall \alpha, \beta \in A_i,\ p^i_\alpha > 0 \Rightarrow c^i_\alpha(\mathbf{p}) \leq c^i_\beta(\mathbf{p})$$

όπου ο εκθέτης δηλώνει τον παίκτη και ο δείκτης τη στρατηγική.

Με άλλα λόγια, στο (μεικτό) NE ο κάθε παίκτης παίζει με μη μηδενική πιθανότητα, μόνο τις στρατηγικές αυτές που ελαχιστοποιούν το κόστος του. Με τη γενίκευση αυτή, ο Nash κατάφερε να αποδείξει το εξής:

Theorem 1.2.8. Θεώρημα Nash: *Σε κάθε πεπερασμένο παίγνιο πολλών παικτών υπάρχει ένα μεικτό προφίλ στρατηγικών το οποίο είναι Nash equilibrium.*

Παρατηρήστε τώρα, ότι, με βάση τον παραπάνω ορισμό, το NE του Παραδείγματος 1.2.3 είναι το μεικτό προφίλ $((1/2, 1/2), (1/2, 1/2))$. Το παραπάνω θεώρημα είναι πολύ σημαντικό γιατί είναι το πρώτο θεώρημα που εξασφαλίζει την ύπαρξη κάποιου είδους ισορροπίας σε κάθε παίγνιο. Παρόλα αυτά είναι ένα μη κατασκευαστικό θεώρημα. Αυτό σημαίνει ότι η απόδειξή του δε μας υπαγορεύει κάποιον τρόπο για να κατασκευάσουμε ένα NE. Συγκεκριμένα ο Nash χρησιμοποίησε στην απόδειξή του τεχνικές από fix-point theorems. Ένα τέτοιο θεώρημα είναι το εξής:

Theorem 1.2.9. Brouwer's Fixpoint Theorem: *Έστω $f : S \to S$ μία συνεχής συνάρτηση από ένα μη κενό, συμπαγές*[4] *και κυρτό σύνολο $S \subseteq \mathbb{R}^n$ στον εαυτό του. Τότε η f έχει σταθερό σημείο, δηλαδή υπάρχει $\widetilde{x} \in S$ τέτοιο ώστε $\widetilde{x} = f(\widetilde{x})$.*

Η ιδέα της απόδειξης του Θεωρήματος 1.2.8 είναι, δοθέντος ενός παιγνίου, να κατασκευάσει κανείς μία συνάρτηση η οποία να ικανοποιεί τις προϋποθέσεις του Θεωρήματος 1.2.9[5] και να επιδέχεται fix-point. Η συνάρτηση πρέπει να είναι έτσι κατασκευασμένη, ώστε το fix-point αυτής να αντιστοιχεί σε NE του παιγνίου. Η κατασκευή της συνάρτησης μάλιστα δεν είναι ιδαίτερα δύσκολη. Έτσι εξασφαλίζουμε την ύπαρξη του NE, χωρίς όμως να έχουμε πληροφορία για την κατασκευή του.
Προφανώς ένα από τα πρώτα θέματα που απασχόλησε την κοινότητα της Θεωρητικής Πληροφορικής είναι αυτό του υπολογισμού ενός NE. Μάλιστα ήδη από το 1964 υπάρχει ο αλγόριθμος Lemke-Howson (ένας simplex-like αλγόριθμος) για εύρεση ενός NE σε bimatrix games. Για πολλά χρόνια εικαζόταν ότι στη χειρότερη περίπτωση ο αλγόριθμος αυτός απαιτεί εκθετικό αριθμό βημάτων, κάτι όμως που μόλις πρόσφατα αποδείχθηκε ότι ισχύει, ανεξάρτητα από το σημείο εκκίνησης, ακόμα και για παιχνίδια νίκης-ήττας [SvS04]. Χωρίς να μπούμε σε πολλές λεπτομέρειες σχετικά με τον υπολογισμό του NE σε παίγνια πολλών παικτών, θα παρουσιάσουμε ένα παράδειγμα που διασαφηνίζει το σκεπτικό πίσω από τον υπολογισμό:

Example 1.2.10. Θεωρούμε παίγνιο στο οποίο αντιστοιχεί ο παρακάτω πίνακας. Καταρχάς εξετάζουμε κατά πόσον υπάρχει PNE. Παρατηρούμε ότι

[4] δηλαδή κλειστό και φραγμένο
[5] αρχικά ο Nash χρησιμοποίησε το fix-point theorem του Kakutani

	b_1	b_2
a_1	3,0	1,1
a_2	1,3	2,0

Table 1.4: Players' payoffs: an example

υπάρχουν 4 δυνατοί συνδυασμοί από pure στρατηγικές, δηλαδή 4 υποψήφια PNE, τα εξής: $(a_1, b_1), (a_1, b_2), (a_2, b_1), (a_2, b_2)$. Επίσης παρατηρούμε τις εξής σχέσεις μεταξύ των payoffs των παραπάνω γνήσιων προφίλ στρατηγικών:

$$u_B(a_1, b_1) < u_B(a_1, b_2)$$

$$u_A(a_1, b_2) < u_A(a_2, b_2)$$
$$u_B(a_2, b_2) < u_B(a_2, b_1)$$
$$u_A(a_2, b_1) < u_A(a_1, b_1)$$

όπου οι συναρτήσεις u_A, u_B ορίζονται στους παρακάτω πίνακες.

Παρατηρούμε δηλαδή ότι για κάθε δυνατό προφίλ στρατηγικών τουλάχιστον ένας παίκτης έχει συμφέρον να αλλάξει τη στρατηγική του. Το αποτέλεσμα είναι ένας κύκλος στο γράφο με κόμβους τα δυνατά προφίλ στρατηγικών και ακμές τις μεταβάσεις μεταξύ προφίλ, όπου μόνο ένας παίκτης αλλάζει στρατηγική. Ο κύκλος αυτός είναι ενδεικτικός της μη ύπαρξης PNE.

Προχωράμε τώρα στον υπολογισμό ενός (μεικτού) NE. Έστω p η πιθανότητα με την οποία ο παίκτης A επιλέγει τη στρατηγική a_1 (πρώτη γραμμή) και q η πιθανότητα με την οποία ο παίκτης B επιλέγει τη στρατηγική b_1 (πρώτη στήλη). Οι πιθανότητες για τα a_2 και b_2 είναι $1-p$ και $1-q$ αντίστοιχα. Τότε έχουμε:
Αν ο A παίξει a_1 το εκτιμώμενο κέρδος του είναι: $3 \cdot q + 1 \cdot (1-q) = 2 \cdot q + 1$.
Αν ο A παίξει a_2 το εκτιμώμενο κέρδος του είναι: $1 \cdot q + 2 \cdot (1-q) = -q + 2$.
Αν ο B παίξει b_1 το εκτιμώμενο κέρδος του είναι: $0 \cdot p + 3 \cdot (1-p) = -3 \cdot p + 3$.
Αν ο B παίξει b_2 το εκτιμώμενο κέρδος του είναι: $1 \cdot p + 0 \cdot (1-p) = p$.

Κάθε παίκτης αποδίδει τις πιθανότητες p και q με βάση τα εκτιμώμενα κέρδη. Έτσι έχουμε τις παρακάτω τιμές για τα p και q ανά περίπτωση και ανά παίκτη:

Για τον A, προκειμένου να μην επιθυμεί να αλλάξει στρατηγική πρέπει:
Αν $2 \cdot q + 1 > -q + 2 \Leftrightarrow q > 1/3$ τότε $p = 1$. (1)
Αν $2 \cdot q + 1 < -q + 2 \Leftrightarrow q < 1/3$ τότε $p = 0$. (2)
Αν $2 \cdot q + 1 = -q + 2 \Leftrightarrow q = 1/3$ τότε απλά $0 < p < 1$. (3)

Αντίστοιχα για τον B:
Αν $-3 \cdot p + 3 > p \Leftrightarrow p < 3/4$ τότε $q = 1$. (4)
Αν $-3 \cdot p + 3 < p \Leftrightarrow p > 3/4$ τότε $q = 0$. (5)
Αν $-3 \cdot p + 3 = p \Leftrightarrow p = 3/4$ τότε απλά $0 < q < 1$. (6)

Παρατηρούμε ότι αν επιλέξουμε κάποια pure στρατηγική για κάποιον από τους δύο παίκτες οδηγούμαστε σε άτοπο, μέσω των σχέσεων (1), (2), (4) και (5). Για παράδειγμα έστω ότι ο A παίζει a_1 με $p = 1$. Τότε λόγω της (5) ο B πρέπει να παίξει b_2, δηλαδή έχουμε $q = 0$. Όμως τότε λόγω της (2), ο A πρέπει να έχει $p = 0$, άτοπο. Με παρόμοια συλλογιστική αποκλείονται όλες οι γνήσιες στρατηγικές.
Έτσι καταλήγουμε στις σχέσεις (3) και (6). Πράγματι για $q = 1/3$ και $p = 3/4$, οι (3) και (6) ικανοποιούνται αμφότερες και το αντίστοιχο μεικτό προφίλ στρατηγικών είναι NE.

Η παραπάνω μέθοδος μπορεί να συστηματοποιηθεί, ώστε να εξάγουμε μία σειρά από εξισώσεις υπό μορφή complementarities όπως λέγονται, τις οποίες μετά θα προσπαθήσουμε να επιλύσουμε. Για την επίλυση των εξισώσεων μπορούμε είτε να δουλέψουμε εξαντλητικά, πάνω στο σύνολο όλων των δυνατών supports του παιγνίου[6], ή να χρησιμοποιήσουμε τον αλγόριθμο Lemke-Howson. Σε κάθε περίπτωση όμως μπορεί να χρειαστούμε εκθετικό αριθμό βημάτων, ως προς τον αριθμό των παικτών και των στρατηγικών.

Εφόσον καθώς φαίνεται, για το πρόβλημα της κατασκευής/εύρεσης NE δεν υπάρχει αποδοτικός (πολυωνυμικός) αλγόριθμος, θα θέλαμε να δείξουμε ότι το πρόβλημα είναι όντως δύσκολο (ιδανικά $\mathcal{NP}$-complete). Το πρόβλημα όμως με την εύρεση NE, είναι ότι ξέρουμε ότι υπάρχει -άρα δεν μπορεί να αποδειχθεί ότι είναι $\mathcal{NP}$-complete, μιας και το πρόβλημα απόφασης δίνει απάντηση "υπάρχει" τετριμένα, χωρίς όμως η απόδειξη να παρέχει έναν αλγόριθμο κατασκευής του. Συνδυάζοντας τα παραπάνω μπορούμε να πούμε ότι το πρόβλημα εύρεσης ενός NE έχει "inefficient proof of existence", κάτι που το τοποθετεί στην κλάση $\mathcal{PPAD}$ [Pap01],[Pap94]. Παρόλα αυτά μέχρι πρόσφατα δεν ήταν γνωστό κατά πόσον το πρόβλημα αυτό ήταν πλήρες για την κλάση αυτή. Τελικά στις δημοσιεύσεις [DGP05] και [DP05] αποδείχθηκε ότι το πρόβλημα είναι $\mathcal{PPAD}$-complete ακόμα και για παίγνια 4 και 3 παικτών αντίστοιχα, και μόλις πέρσι αποδείχθηκε στο [CD06] ότι το ίδιο ισχύει και για την περίπτωση των 2 παικτών. Το γεγονός αυτό, επιβεβαιώνει τη διαίσθηση που είχαμε, ότι το πρόβλημα εύρεσης NE είναι μάλλον[7] δύσκολο.

[6] περισσότερες λεπτομέρειες για τα supports στο 2ο Κεφάλαιο της διπλωματικής
[7] η ακριβής σχέσης της $\mathcal{PPAD}$ με την $\mathcal{NP}$ δεν είναι γνωστή

Στο σημείο αυτό αξίζει να σημειωθεί ότι η κλάση $\mathcal{PPAD}$ είναι υποσύνολο της κλάσης $\mathcal{TFNP}$, η οποία περιλαμβάνει όλα τα προβλήματα εύρεσης ("$\mathcal{NP}$-search problems") για τα οποία όμως ξέρουμε ότι υπάρχει λύση. Μία άλλη υποκλάση της $\mathcal{TFNP}$ είναι η $\mathcal{PLS}$, την οποία θα ξαναδούμε στο Κεφάλαιο 3.

Ένα άλλο σημαντικό ερώτημα που προκύπτει άμεσα από την παραπάνω συζήτηση και έχει επίσης αποτελέσει σημαντικό πεδίο ερευνών, είναι αυτό της ύπαρξης προσεγγιστικών αλγορίθμων για την εύρεση ενός NE. Γενικά το σημαντικό είναι ότι ως τώρα δεν υπάρχει κανένα πολυωνυμικού χρόνου σχήμα προσέγγισης για ε-ApproxNE[8] για κάθε (σταθερό) ε. Παρόλα αυτά για κάθε ε, αποδεικνύεται στο [LMM03] ότι υπάρχουν ε-ApproxNE με $O(\log(m+n)/\varepsilon^2)$ πιθανές στρατηγικές ανά παίκτη, κάτι το οποίο υπονοεί υποεκθετικό χρόνο υπολογισμού. Επίσης στο [CDT06] αποδεικνύεται ότι εκτός αν $\mathcal{PPAD} \subseteq \mathcal{P}$, δεν υπάρχει αλγόριθμος εύρεσης ε-ApproxNE με πολυπλοκότητα $poly(n, 1/\varepsilon)$ για κάθε $\varepsilon = n^{-\Theta(1)}$, το οποίο σημαίνει ότι μάλλον δεν υπάρχει FPTAS. Πολύ πρόσφατα αποτελέσματα πάνω στο θέμα των προσεγγιστικών NE υπάρχουν στα [KPS06] και [DMP06].

Τέλος να τονίσουμε ότι, παρόλο που το πρόβλημα εύρεσης ενός NE δεν είναι $\mathcal{NP}$-complete, το πρόβλημα απόφασης της ύπαρξης περισσοτέρων του ενός NE σε ένα παίγνιο είναι: [CS03]. Το ίδιο ισχύει για το πρόβλημα εύρεσης όλων των NE ενός παιγνίου (τα οποία μπορεί να είναι εκθετικά πολλά), καθώς και για το πρόβλημα εύρεσης του NE με το μέγιστο συνολικό payoff.

Κλείνοντας το κεφάλαιο για τα equilibria να παρατηρήσουμε ότι κάθε DE είναι NE, ισχύει δηλαδή μία σχέση εγκλεισμού ανάμεσα στα 2 είδη ισορροπίας. Μία πιο γενική έννοια ισορροπίας (που περιέχει τα NE) είναι αυτή του Correlated Equilibrium.

1.2.3 To Price of Anarchy

Ας θυμηθούμε τώρα το Παράδειγμα 1.2.4. Είχαμε καταλήξει ότι το μόνο NE για το παράδειγμα αυτό είναι το (Confess,Confess). Το άσχημο εδώ είναι ότι οι παίκτες μένουν ευχαριστημένοι (ισορροπούν) για ένα προφίλ στρατηγικών που τους αναγκάζει να πάνε 3 χρόνια φυλακή, ενώ θα μπορούσαν να γλιτώσουν με 1 μόνο χρόνο. Το παράδειγμα αυτό καταδεικνύει μία από τις μεγάλες αδυναμίες της αυτόνομης και εγωιστικής συμπεριφοράς σε τέτοια συστήματα: ότι δηλαδή, μπορεί το τελικό αποτέλεσμα να απέχει από το βέλτιστο. Το

[8] δηλαδή ενός "equilibrium" όπου κανένας παίκτης δεν μπορεί να κερδίσει παραπάνω από ε αλλάζοντας στρατηγική

τίμημα της αναρχίας (price of anarchy - PoA) μας δίνει ακριβώς αυτό το λόγο μεταξύ του χειρότερου δυνατού αποτελέσματος (NE) του παιχνιδιού και του βέλτιστου, το οποίο όμως δεν αντιστοιχεί κατ' ανάγκη σε NE. Το ερώτημα που καλούμαστε να απαντήσουμε είναι αν το PoA μπορεί να φραχθεί για τα συγκεκριμένα παίγνια που εξετάζουμε.

Ας προχωρήσουμε όμως στον τυπικό ορισμό. Αρχικά θεωρούμε μία αντικειμενική συνάρτηση $C : A_1 \times \ldots \times A_n \to \mathbb{R}_+^*$ η οποία αντιστοιχεί σε κάθε προφίλ στρατηγικών, δηλαδή σε κάθε υποψήφιο αποτέλεσμα του παιγνίου έναν αριθμό, που αντιστοιχεί στο κόστος του αποτελέσματος για την κοινωνία. Έτσι θα λέμε ότι η συνάρτηση αυτή αναπαριστά το κοινωνικό κόστος (social cost - SC) του παιγνίου. Συνηθισμένες επιλογές για τη C είναι οι συναρτήσεις $\max_i c_i$ και $\sum_i c_i$. Φυσικά έχουν κατά καιρούς προταθεί και άλλες συναρτήσεις.
Αυτό που θέλουμε εμείς τώρα είναι να συγκρίνουμε το κόστος του χειρότερου δυνατού NE, δηλαδή του NE με το μεγαλύτερο δυνατό κόστος, με το κόστος της βέλτιστης λύσης (έστω OPT), η οποία, να ξανατονίσουμε, δεν είναι ανάγκη να αντιστοιχεί σε NE[9]. Σκοπός τώρα είναι να βρούμε το λόγο των δύο παραπάνω μεγεθών, οπότε ορίζουμε:

Definition 1.2.11. Ως Price of Anarchy ενός παιγνίου ορίζουμε το λόγο:

$$PoA = \max_{P \in NE} \frac{SC(P)}{OPT}$$

Στο σημείο αυτό πρέπει να κάνουμε μία ακόμα επεξήγηση. Παρατηρούμε ότι στο PoA υπολογίζουμε το SC (που είναι κάποια αντικειμενική συνάρτηση) για κάποια προφίλ στρατηγικών που είναι NE. Τι γίνεται όμως όταν οι παίκτες παίζουν με μεικτές στρατηγικές (όπως συμβαίνει συνήθως); Η προφανής απάντηση είναι ότι γενικεύουμε τη συνάρτηση κοινωνικού κόστους, ώστε πλέον να μας επιστρέφει το εκτιμώμενο (estimated) κοινωνικό κόστος, με βάση τις πιθανότητες των διαφόρων παικτών για κάθε στρατηγική.
Με το PoA θα ασχοληθούμε εκτεταμένα στα επόμενα κεφάλαια. Για ένα πρώτο παράδειγμα δείτε το 1.3.1.
Το μόνο που αξίζει να σημειωθεί είναι ότι, πέρα από τον παραπάνω ορισμό, έχουν προταθεί και άλλοι τέτοιοι λόγοι, όπως το PPoA (pure price of anarchy) που υπολογίζεται πάνω στο σύνολο των PNE (βλ. Κεφάλαιο 3), καθώς και το price of stability, ή optimistic price of anarchy, το οποίο θεωρεί το λόγο του καλύτερου NE προς τη βέλτιστη λύση και βοηθά το σχεδιαστή του συστήματος, μιας και περιγράφει τις ελάχιστες δυνατές απώλειες σε σχέση με το βέλτιστο όταν οι παίκτες είναι σε NE (με άλλα λόγια σε ποιο NE θέλουμε να "βάλουμε"

[9] αυτό πρακτικά σημαίνει ότι αν αφήσουμε τους παίκτες να παίξουν μόνοι τους, χωρίς κάποια κεντρική αρχή, ποτέ δεν πρόκειται να καταλήξουν στη βέλτιστη λύση

τους παίκτες).
Το PoA ορίστηκε στο [KP99], το οποίο μελετάμε στο 2ο Κεφάλαιο, κατ' αναλογία του approximation ratio (στους προσεγγιστικούς αλγορίθμους) και του competitive ratio (στους online αλγορίθμους).

1.3 Μερικά παράδοξα της Θεωρίας Παιγνίων

Στο μέρος αυτό θα παρουσιάσουμε 3 παραδείγματα που σε πρώτη ανάγνωση μπορεί να μας ξαφνιάσουν και αποτελούν καλό εφαλτήριο για την ερευνητική δουλειά που παρουσιάζεται στα επόμενα κεφάλαια της διπλωματικής.

Example 1.3.1. Selfish Task Allocation
Ας θεωρήσουμε ότι έχουμε 4 εργαζόμενους/παίκτες που ο κάθε ένας θέλει να τρέξει ένα πρόγραμμα προσομοίωσης σε έναν υπολογιστή. Οι διάρκειες των προγραμμάτων προσομοίωσης είναι 1, 2, 3 και 4 λεπτά αντίστοιχα και το γραφείο διαθέτει 2 υπολογιστές, οι οποίοι εφαρμόζουν round-robin χρονοδρομολόγηση στις διεργασίες που τρέχουν. Αυτό πρακτικά σημαίνει ότι παραχωρούν εκ περιτροπής λίγο χρόνο σε όλες τις διεργασίες που είναι φορτωμένες σε αυτούς, έτσι ώστε, τελικά, όλες οι διεργασίες να τελειώνουν ταυτόχρονα. Οι εργαζόμενοι αποφασίζουν μόνοι τους σε ποιον υπολογιστή τους θα τρέξουν το πρόγραμμά τους, με μόνο κριτήριο να τελειώσουν όσο το δυνατό νωρίτερα.

Στο Σχήμα 1.1 φαίνονται δύο δυνατές κατανομές των εργασιών σε υπολογιστές. Παρατηρούμε ότι και στις δύο περιπτώσεις όλοι οι εργαζόμενοι είναι ευχαριστημένοι και κανείς δεν πρόκειται να αλλάξει υπολογιστή, δηλαδή οι επιλογές τους αντιστοιχούν σε NE. Στην αριστερή κατανομή κάθε παίκτης πρέπει να περιμένει 5 λεπτά.
Ας εστιάσουμε τώρα στη δεξιά κατανομή των εργασιών σε μηχανές. Παρατηρούμε ότι και σε αυτή την περίπτωση κανένας παίκτης δεν έχει κάποιο συμφέρον να αλλάξει υπολογιστή (ακόμα και ο παίκτης 2, αφού σε κάθε

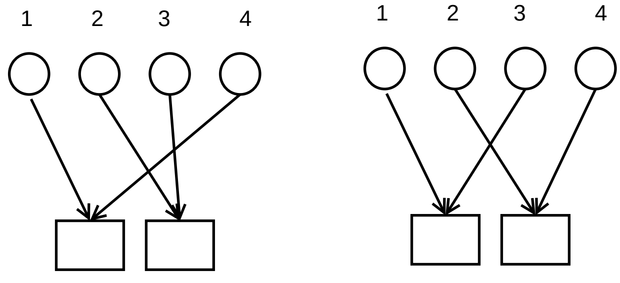


Figure 1.1: 2 NE σε Selfish Task Allocation instance

περίπτωση θα περιμένει 6 λεπτά). Έτσι και αυτή η κατάσταση αντιστοιχεί σε NE, όπου όμως κάποιοι παίκτες περιμένουν 6 λεπτά (θεωρώ δηλαδή σαν κοινωνικό κόστος το max).
Με άλλα λόγια, στο παράδειγμα αυτό βλέπουμε ότι ένα παίγνιο μπορεί να επιδέχεται 2 NE διαφορετικής ποιότητας το καθένα (διαφορετικό PoA). Συγκεκριμένα, παρατηρώντας ότι η βέλτιστη ανάθεση εδώ αντιστοιχεί στο πρώτο NE, μπορούμε να επαληθεύσουμε ότι τα 2 NE έχουν PoA 1 και 6/5 αντίστοιχα.

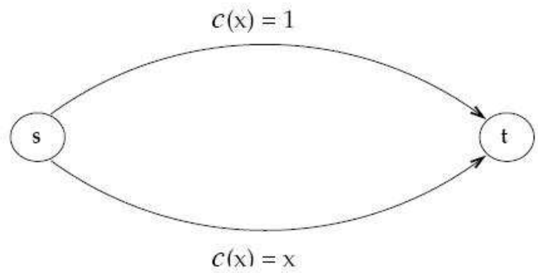


Figure 1.2: Pigou's Example

Example 1.3.2. Pigou's Network
Ας θεωρήσουμε τώρα ένα δίκτυο όπως αυτό του Σχήματος 1.2. Δύο ακμές συνδέουν τους κόμβους s και t. Οι κόμβοι αυτοί μπορεί να αντιστοιχούν στους τερματικούς κόμβους κάποιου δικτύου, στο οποίο θέλουμε να μεταφέρουμε κάποια ποσότητα πληροφορίας. Κάθε μία από τις ακμές έχει μία συνάρτηση κόστους $c(\cdot)$, η οποία περιγράφει την καθυστέρηση για τους χρήστες που χρησιμοποιούν την ακμή αυτή για να μεταφέρουν τα δεδομένα τους. Η πάνω ακμή έχει σταθερή καθυστέρηση $c(x) = 1$, δηλαδή δεν επηρεάζεται από το φορτίο της, ενώ η κάτω ακμή έχει καθυστέρηση $c(x) = x$, δηλαδή η καθυστέρησή της αυξάνεται με το φορτίο της (όπως θα περιμέναμε). Παρατηρήστε ότι η κάτω ακμή είναι πιο γρήγορη από την πάνω αν και μόνο αν τη διαρρέει λιγότερο από μία μονάδα δεδομένων.

Ας υποθέσουμε λοιπόν ότι έχουμε 1 μονάδα δεδομένων προς μεταφορά και ότι η μονάδα αυτή είναι συνεχής, δηλαδή μπορούμε να τη σπάσουμε όπως επιθυμούμε. Η ιδέα είναι ότι έχουμε ένα μεγάλο αριθμό χρηστών που όλοι μαζί διαχειρίζονται τη συγκεκριμένη μονάδα δεδομένων, ελέγχοντας ο καθένας ένα μικρό (αμελητέο) κομμάτι αυτής. Αν ο κάθε χρήστης διαλέγει ανεξάρτητα ένα μονοπάτι για να δρομολογήσει τη ροή του, περιμένουμε ότι όλοι οι χρήστες (οπότε και όλη η κίνηση) θα διαλέξουν την κάτω ακμή, που δεν είναι ποτέ χειρότερη από την πάνω για ροή μέχρι 1. Αντίθετα, μπορεί να είναι και καλύτερη, αν κάποιοι χρήστες είναι αρκετά αφελείς ώστε να επιλέξουν την

πάνω ακμή. Έτσι, θεωρώντας ότι οι παίκτες παίζουν εγωιστικά, περιμένουμε όλη η ροή να δρομολογηθεί με μία μονάδα καθυστέρησης.
Ας υποθέσουμε τώρα ότι υπάρχει μία κεντρική αρχή που μπορεί να ρυθμίσει την κίνηση. Αν η αρχή αυτή αναγκάσει τους μισούς παίκτες να χρησιμοποιήσουν την πάνω ακμή, τότε η μεν ροή που δρομολογείται από πάνω, έχει πάλι καθυστέρηση 1 (δηλαδή το ίδιο με πριν) ενώ η ροή που δρμολογείται από κάτω, έχει τώρα καθυστέρηση 1/2. Έτσι το μέσο κόστος τώρα γίνεται 3/4 από 1 που ήταν πριν. Παρατηρήστε βέβαια, ότι οι παίκτες της πάνω ακμής δεν είναι ευχαριστημένοι και θέλουν να κατέβουν στην κάτω ακμή, δηλαδή η κατάσταση αυτή δεν αντιστοιχεί σε NE. Το παράδειγμα αυτό επιβεβαιώνει ξανά ότι η εγωιστική συμπεριφορά οδηγεί σε αποτελέσματα που υπολείπονται του βέλτιστου. Εν προκειμένω το PoA αποδεικνύεται ότι είναι 4/3 (μιας και το παραπάνω NE είναι και το μοναδικό για το παίγνιο αυτό).

Το παραπάνω φαινόμενο μπορεί να οξυνθεί, αν επιτρέψουμε μη γραμμικές

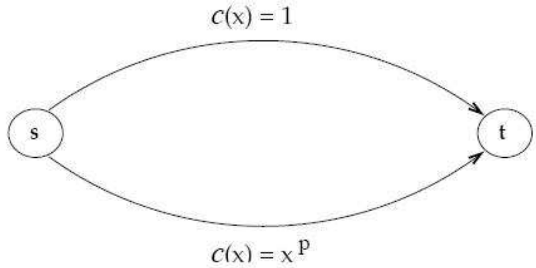
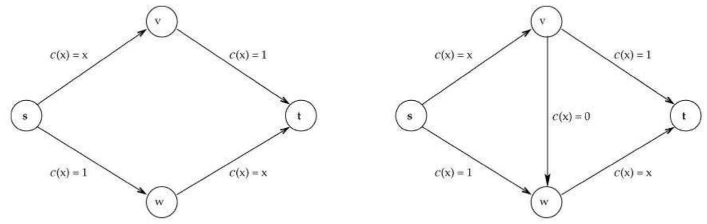


Figure 1.3: A nonlinear variant of Pigou's Example

συναρτήσεις καθυστέρησης. Συγκεκριμένα υποθέστε ότι έχουμε το Σχήμα 1.3 όπου η κάτω ακμή έχει συνάρτηση καθυστέρησης $c(x) = x^p$ για p μεγάλο. Κατ' αναλογία με την προηγούμενη περίπτωση, στο NE όλοι οι παίκτες επιλέγουν την κάτω ακμή με συνολικό κόστος 1. Αντίθετα, στο βέλτιστο, δρομολογούμε ένα μικρό μέρος $\epsilon > 0$ της ροής από την πάνω ακμή, με αποτέλεσμα το μέσο κόστος να πέσει στο $\epsilon + (1-\epsilon)^p + 1$ το οποίο τείνει στο 0 καθώς $\epsilon \to 0$ και $p \to \infty$. Έτσι τώρα το PoA τείνει και αυτό στο άπειρο, και μάλιστα όπως το $p/\ln p$
Στο 3ο Κεφάλαιο θα δούμε ότι τα δίκτυα του παραδείγματος, αποτελούν κατά κάποιον τρόπο καθολικά "κακά" παραδείγματα για το παίγνιο αυτό, υπό την έννοια ότι για κάθε επιτρεπώμενο σύνολο συναρτήσεων κόστους, το χειρότερο PoA προκύπτει από ένα δίκτυο αυτού του τύπου. Θα δούμε δηλαδή, πώς μπορούμε να χρησιμοποιήσουμε τα Pigou's Networks για να φράξουμε αποτελεσματικά το PoA.

Example 1.3.3. Braess's Paradox
Συνεχίζουμε στο ίδιο πνεύμα με το προηγούμενο παράδειγμα, δηλαδή θεωρούμε



(a) Initial network (b) Augmented network

Figure 1.4: Braess's Paradox

συνεχή ροή 1 μονάδας. Θεωρείστε τώρα το δίκτυο 4 κόμβων του Σχήματος 1.4(a). Υπάρχουν 2 ξένα μονοπάτια από το s στο t, το κάθε ένα με κόστος $1 + x$, όπου x είναι το φορτίο της ακμής. Επειδή τα δύο μονοπάτια είναι πανομοιότυπα, η κίνηση θα μοιραστεί στη μέση και κάθε μονοπάτι θα μεταφέρει 1/2 ροή. Έτσι στην περίπτωση αυτή, η μέση καθυστέρηση θα είναι 3/2.
Υποθέστε τώρα, ότι, σε μία προσπάθεια να βελτιώσουμε την απόδοση του δικτύου, προσθέτουμε μία πολύ γρήγορη ακμή (με $c(x) = 0$) μεταξύ των κόμβων v και w (Σχήμα 1.4(b)). Πώς θα αντιδράσουν οι ιδιοτελείς χρήστες;
Παρατηρούμε ότι η προηγούμενη δρομολόγηση παύει να αποτελεί πλέον NE. Όντως το μονοπάτι $s \to v \to w \to t$ δεν είναι ποτέ χειρότερο από τα 2 αρχικά μονοπάτια, για μέχρι 1 μονάδα ροής και είναι καλύτερο αν κάποιος χρήστης επιλέξει (χαζά) κάποιο από τα αρχικά μονοπάτια. Με το σκεπτικό αυτό, περιμένουμε όλους τους ιδιοτελείς χρήστες να επιλέξουν το νέο μονοπάτι, με αποτέλεσμα όλη η ροή τώρα να έχει μέση καθυστέρηση 2. Έτσι το κόστος / καθυστέρηση της ροής αυξήθηκε κατά ένα παράγοντα 4/3, στην προσπάθειά μας να βελτιώσουμε την ποιότητα του δικτύου!
Στο 3ο Κεφάλαιο θα δούμε πώς μπορούμε να γενικεύσουμε το παράδοξο αυτό στην περίπτωση μεγάλων δικτύων, με προσθήκη πολλών ακμών και θα αποδείξουμε ένα φράγμα, εξαρτώμενο από τον αριθμό των κόμβων, καθώς και τη στενή σχέση του παραδόξου αυτού με το PoA.

1.4 Θεωρητική Πληροφορική και Θεωρία Παιγνίων

Στην Αλγοριθμική Θεωρία Παιγνίων, όπως λέγεται ο κλάδος της Θεωρητικής Πληροφορικής που ασχολείται με τη Θεωρία Παιγνίων, το μεγαλύτερο κομμάτι της έρευνας αφορά τα μη συνεργατικά, στρατηγικά παίγνια, πλήρους πληροφορίας, χωρίς αυτό να σημαίνει ότι δεν υπάρχει έρευνα και για άλλες κατηγορίες.

Σε ό,τι αφορά το είδος της έρευνας που πραγματοποιείται στον τομέα αυτό, μερικά από τα βασικά ερωτήματα που έχει κληθεί να απαντήσει η Θεωρητική Πληροφορική είναι τα εξής:

- Επιδέχεται ένα συγκεκριμένο παίγνιο PNE, ή κάποιο άλλο είδος ισορροπίας του οποίου η ύπαρξη δεν είναι τετριμμένη (όπως πχ του μεικτού NE)

- Αν υπάρχει ισορροπία (οποιουδήποτε είδους), πόσος χρόνος απαιτείται για να βρεθεί; Αντίστοιχα πόσος χρόνος απαιτείται για την εύρεση της καλύτερης ή της χειρότερης ισορροπίας; Πόσος χρόνος απαιτείται για την εύρεση κάποιας ισορροπίας που πληροί κάποιες προδιαγραφές;

- Αν η εύρεση μίας ισορροπίας είναι δύσκολη, πόσο δύσκολο είναι να βρεθεί μία προσεγγιστική ισορροπία, δηλαδή μία κατάσταση από την οποία κανείς δεν κερδίζει "πολύ" αν φύγει;

- Πόσο κακή μπορεί να είναι μία ισορροπία; Μπορεί να φραχθεί ικανοποιητικά το PoA ενός παιγνίου; Αντίστοιχα, πόσο καλή μπορεί να είναι μία ισορροπία (PoS), δηλαδή αν υπήρχε μία κεντρική αρχή που θα υπαγόρευε την ισορροπία αυτή, πόσο θα χάναμε σε σχέση με το βέλτιστο;

- Αν ένα παίγνιο δεν επιδέχεται καλές ισορροπίες, τι μπορούμε να κάνουμε για να τις βελτιώσουμε (coordination mechanisms);

- Να σχεδιάσουμε μηχανισμούς για το διαμοιρασμό αγαθών και για δημοπρασίες (auction and mechanism design).

Κάποια από τα παραπάνω ερωτήματα έχουν μελετηθεί γενικά για τα (στρατηγικά) παίγνια (πχ πολυπλοκότητα εύρεσης NE), ενώ άλλα έχουν νόημα (και έχουν μελετηθεί) μόνο για συγκεκριμένες κατηγορίες παιγνίων, όπως π.χ. τα network games, ή τα congestion games (χαρακτηριστικά το PoA ή το network design).
Παρατηρούμε ότι υπάρχουν ερωτήματα δομικά, που σχετίζονται με την ύπαρξη ή μη ισορροπιών, ερωτήματα που αφορούν την κατασκευή αποδοτικών αλγορίθμων για τον υπολογισμό ισορροπιών (ή την απόδειξη ότι δεν υπάρχουν) και ερωτήματα που αφορούν τη σχεδίαση "καλών" παιγνίων.
Όταν έχουμε ένα νέο πρόβλημα που μοντελοποιείται με τη Θεωρία Παιγνίων, η τάση είναι να προσπαθούμε πρώτα να απαντήσουμε στα δομικά - αλγοριθμικά ζητήματα που αφορούν το πρόβλημα, μετά να μελετάμε το PoA και τέλος να προσπαθούμε να αναπτύξουμε μηχανισμούς που το βελτιώνουν. Για τα προβλήματα που θα αναπτυχθούν στα επόμενα κεφάλαια της διπλωματικής, έχει γίνει πολλή έρευνα σχετική με όλα τα παραπάνω ερωτήματα. Στην παρούσα διπλωματική ασχολούμαστε ως επί το πλείστον με την ανάλυση των παιγνίων

αυτών (equilibria, PoA) και όχι τόσο με τη σχεδίασή τους (network design, coordination mechanisms).

Part I

The atomic case

Chapter 2

The Koutsoupias-Papadimitriou model

We begin our analysis of the atomic case of selfish routing with the model introduced by Koutsoupias and Papadimitriou in their paper [KP99], which we shall henceforth call the *KP-model*. The reasons for picking this model at the beginning are numerous. First of all they are historical: the paper in which this model was introduced was one of the first to consider the game-theoretic aspect of routing traffic through a congested network and had a great impact on the research community. It was followed by a lot of papers, some of which (such as [MS01], [KMS03], [CV02], [FKK$^+$02]) resolved open problems related to the KP-model (existence and uniqueness of a FMNE, existence of a PNE, tight PoA bounds), while some other ([FKS04], [FKS05]) include highly non-trivial generalizations of this model which we shall encounter in the next chapters of this thesis. Apart from giving some food for thought to a lot of researchers, the [KP99] paper had another major contribution in algorithmic game theory: it quantified for the first time the cost of the lack of coordination among the players, by introducing the notion of the *coordination ratio*, which is the now known under the name "Price of Anarchy". In this chapter we discuss the model and present the proofs as they appear in [KP99], but we also enhance them with some of [MS01]'s results, since this paper resolves a lot of open problems and proves some conjectures posed in [KP99]. We then cite [KMS03] and [CV02] for some results on the PoA.

2.1 The Model

One of the characteristics of KP-model that make it so appealing is its simplicity: it consists of a network of m parallel links between a source and a destination node. These links can be seen as parallel unrelated machines (eg servers) that route traffic independently from the source to the destination. Apart from the machines there are n players (or agents) that want to route a specific amount of traffic through the network. A good analogy is to think of the players as tasks that need to be scheduled on those machines[1]. Each such task has a different execution time (the traffic of player i) that we shall denote by w_i. Of course the longer a task lasts, the bigger the latency for all other tasks scheduled on the same machine. Equivalently if the traffic on one link of our network is heavy, it normally produces a bigger delay. We say then that traffic determines delay and that the delay suffered by each agent on a link equals the total amount of traffic routed through this link (this of course is a simplification - more complex cost functions have also been studied). We have implicitly made the assumption that every link has the same speed (or as we use to say the same *capacity*). However we are also going to study the problem for links with different capacities s^j.

Now to sum things up, the KP-model consists of:

- m (unrelated) parallel links: $[m] = 1, \ldots, m$.
- n (selfish) players: $[n] = 1, \ldots, n$.
- n amounts of traffic $w_i, i = 1, \ldots n$, one for each player. We assume that $w_1 \geq \ldots \geq w_n$.
- m capacities-speeds, $s^j, j = 1, \ldots m$, one for each link. We assume that $s^1 \leq \ldots \leq s^m$.

Now let us see things from an agent's point of view. Since the players are selfish their aim is to minimize their delay, while routing their whole amount of traffic from the destination to the source. In order to route their traffic they can either pick one of the m links, or they can assign to each link j a non-negative number that indicates the probability of picking it. We denote these probabilities for player i by p_i^j -in general we shall use subscripts for agents and superscripts for links. The two cases above obviously evoke the notions of PNE and NE respectively. In the present thesis we shall mostly consider the mixed strategies case, for which there are very interesting results. Some information on the existence of a PNE can be found in the next chapter.

[1] the problem is also known under the name "selfish task allocation"

So the set of *pure strategies* for agent i is $\{1, \ldots, m\}$ and a *mixed strategy* is a distribution on this set. Let $(j_1, \ldots, j_n) \in \{1, \ldots, m\}^n$ be a combination of pure strategies, one pure strategy for each agent, called the *pure strategies profile*; then its cost for agent i, denoted by $C^i(j_1, \ldots, j_n)$ is:

$$L^{j_i} + \sum_{j_k = j_i} w_k$$

if we take all link capacities to be equal to unit. This cost gives us the finish time of the link j_i chosen by i, if its initial load is a task of length L^{j_i}. That means that the link will be available after L^{j_i} time units. Then starts a round-robin way of task processing, in which each task receives a very small amount of processing time and then it gives its place to the next one, until they all finish (practically at the same time). This is the so called *standard model*. There exists also another model, the *batch model*, which we will not consider in this thesis. For more information see [KP99].

2.2 Nash Equilibria

2.2.1 Definitions

We shall now attempt to characterize the Nash Equilibria in the standard model of the game, where all link capacities are unit. We define *the expected traffic* or *expected load* M^j on link j to be:

$$M^j = L^j + \sum_i p_i^j w_i \tag{2.1}$$

It is also obvious that the cost as defined above is not useful for the case of a mixed strategies profile. What we need here is the *estimated finish time* or *estimated cost* for each player when he assigns his traffic to link j. That is, each player knows the mixed strategy profile of every other player and tries to choose one link that will minimize his latency. The problem is that he can only make estimations about other players' behavior based on their probability distribution. The resulting finish time c_i^j then is:

$$c_i^j = w_i + L^j + \sum_{i \neq t} p_t^j w_t = M^j + (1 - p_i^j) w_i \tag{2.2}$$

The mixed strategies profile of each player, that is the probabilities p_i^j define a NE if no player has an incentive to deviate and pick another link.

Thus, agent i will assign nonzero probabilities only to links j that minimize c_i^j. We will denote this minimum by

$$c_i = \min_j c_i^j$$

and we call the set of links $S_i = \{j : p_i^j > 0\}$ the *support* of agent i. We also define the indicator variable S_i^j that takes value 1 when $p_i^j > 0$.
In order to become more formal we shall henceforth represent mixed strategies profiles by an $n \times m$ probability matrix $\mathbf{P}$. Now it is easy to see that we can fully characterize a mixed strategies profile $\mathbf{P}$ that is a NE, based solely on the supports S_i^j. More precisely we have the following proposition:

Proposition 2.2.1. *Take a Nash Equilibrium* $\mathbf{P}$*. Then for every link* $j \in [m]$ *and every agent* $i \in [n]$*,*

$$p_i^j = (M^j + w_i - c_i)/w_i \tag{2.3}$$

subject to the following constraints:
(1) for all j: $M^j = L^j + \sum_i S_i^j (M^j + w_i - c_i)$
(2) for all i: $\sum_i S_i^j (M^j + w_i - c_i) = w_i$

To see that the above proposition holds notice that (2.3) is equivalent to (2.2), where we have replaced c_i^j by c_i, so that (2.3) holds only for the links of minimum latency, for which $p_i^j > 0$ The constraints (1) and (2) are equivalent to (1.1) and to the fact that the probabilities of agent i must sum up to 1 respectively. Here we once again take into account only the links for which $p_i^j > 0 (\Leftrightarrow S_i^j > 0)$. One should also notice that if we fix the nm supports S_i^j then we have $n + m$ constraints and $n + m$ unknowns and this set of constraints should then have a unique solution. If the resulting probabilities are in the interval (0,1], then the above equations define a unique equilibrium with supports S_i^j.

As we clearly stated back in chapter 1, there are now a couple of very natural questions that one may ask. They have to do with the general structure of the resulting equilibria: since the **existence** of a NE is trivial thanks to [Nas51], we focus our interest on the PNE. We also care about the **uniqueness** of equilibria and about the **computational complexity** of finding one. We are therefore interested in finding closed form expressions for the above probabilities p_i^j. In fact notice that a NE is fully defined only by the supports S_i^j (since in a NE $S_i^j = 1 \Leftrightarrow c_i^j = c_i$). One way for finding a NE would be then to try all possible values for all S_i^j and solve the resulting system in polynomial time. The problem here is that we need to examine 2^{nm} different supports and hence this method is in general exponential in n and m. But can we do better?

2.2.2 The Fully Mixed Nash Equilibrium

The first paper to follow [KP99] was [MS01] in which the first existence and uniqueness result for equilibria was presented. We shall give here a brief sketch of the proof method. Though not really difficult, the proof has a lot of technical details which we shall omit here in favor of space.
The model considered in [MS01] differs from the one we considered up to now in some points. We summarize them right below:

- It considers the initial load L^j of each link j to be equal to zero. That is no restriction at all for the standard model though, since initial loads can be considered as jobs of m additional agents. In fact from now on we implicitly consider all L^j equal to zero.

- It considers only a special case of mixed Nash equilibria, called fully mixed Nash equilibria. In the FMNE each user assigns his traffic on every link with positive probability ($S_i^j = 1, \forall j \in [m], i \in [n]$) and his support is $[m]$. Although a more restricted type of NE, the FMNE deserves our attention because it allows us to solve the equations of Proposition 2.2.1, hence providing us with a closed and remarkably insightful type for the probabilities p_i^j of all agents. One final remark on those types yields the required existence (which now is not trivial) and uniqueness result for a FMNE. Moreover it is easy to intuitively understand that a FMNE favors collisions of users across the links, thus increasing the maximum latency, the (not yet formally defined) Social Cost and the Coordination Ratio, in which we are particularly interested. This characteristic alone makes FMNE a NE worth studying.

- It considers links of arbitrary capacities. Since we are also going to consider this kind of links when we try to bound the PoA, it is a good idea to give the analogous of Proposition 2.2.1 for links of arbitrary capacities:

Proposition 2.2.2. *Take a Nash Equilibrium* $\mathbf{P}$*. Then for every link* $j \in [m]$ *and every agent* $i \in [n]$*,*

$$p_i^j = (M^j + w_i - s^j c_i)/w_i \tag{2.4}$$

subject to the following constraints:
(1) for all j: $M^j = L^j + \sum_i S_i^j (M^j + w_i - s^j c_i)$
(2) for all i: $\sum_i S_i^j (M^j + w_i - s^j c_i) = w_i$

and

$$c_i = \min_j c_i^j = \min_j (M^j + (1 - p_i^j) w_i)/s^j \tag{2.5}$$

Finally let us define a *solo link* as a link $j \in [m]$ such that $\sum_i S_i^j = 1$. Clearly that means that a solo link is traversed by only one user. A link that is not solo is a *non-solo link*. By a simple rearrangement of terms in constraint (1) of Proposition 2.2.2 we can prove the following:

Lemma 2.2.3. *Take any Nash Equilibrium* $\mathbf{P}$. *Then, for any non-solo link* $j \in [m]$,

$$M^j = \frac{-\sum_i S_i^j w_i + s^j \sum_i S_i^j c_i}{\sum_i S_i^j - 1}$$

It is easy to see that j being a non-solo link is necessary, in order for the denominator to be unequal zero. The key point now is that in a FMNE by definition there are no solo links at all, and hence this lemma applies to all $j \in [m]$. We are now going to use this fact in order to provide a series of other propositions that will finally yield the desired result. The first proposition in this series follows right away:

Proposition 2.2.4. *Take any Nash Equilibrium* $\mathbf{P}$. *Let* $S \subseteq [m]$ *denote the set of solo links. Then, for any user* $i \in [n]$,

$$c_i \Big(\sum_j S_i^j s^j - \sum_{j \notin S} \frac{S_i^j}{\sum_k S_k^j - 1} s^j \Big) - \sum_{k \neq i} c_k \Big(\sum_{j \notin S} \frac{S_i^j S_k^j}{\sum_k S_k^j - 1} s^j \Big)$$

$$= w_i \Big(\sum_j S_i^j - 1 - \sum_{j \notin S} \frac{S_i^j}{\sum_k S_k^j - 1} \Big) - \sum_{k \neq i} w_k \Big(\sum_{j \notin S} \frac{S_i^j S_k^j}{\sum_k S_k^j - 1} \Big) + \sum_{j \in S} S_i^j M^j$$

Although it looks scary, the above type is actually not that hard to prove: the basic trick here is to derive two separate expressions for $\sum_{j \in S} S_i^j M^j$. The first one follows directly from constraint (2) of Proposition 2.2.2 and the second one follows using the expressions for the estimated traffic M^j derived in

Lemma 2.2.3. Equating those two expressions yields the above equation.

We now focus on the fully mixed case. Since there are no solo links in the fully mixed case, $S = \emptyset$ and the term $\sum_{j \in S} S_i^j M^j$ is eliminated. We then set $S_i^j = 1, \forall j \in [m], i \in [n]$ in the above equation, rearrange the terms and solve the resulting linear system to obtain that $(c_1, \ldots, c_n)$ is a linear transformation of $(w_1, \ldots, w_n)$:

Lemma 2.2.5. *Consider any fully mixed Nash Equilibrium* $\mathbf{P}$. *Then:*

$$\begin{pmatrix} c_1 \\ \vdots \\ c_n \end{pmatrix} = \frac{1}{\sum_j s^j} \begin{pmatrix} m & 1 & \ldots & 1 \\ 1 & m & \ldots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \ldots & m \end{pmatrix} \begin{pmatrix} w_1 \\ \vdots \\ w_n \end{pmatrix}$$

We can now substitute the above expressions into the expressions for the expected traffics (Lemma 2.2.3) to obtain that $(M^1, \ldots, M^m)$ is also a linear transformation of $(w_1, \ldots, w_n)$:

$$(M^1, \ldots, M^m) = T(w_1, \ldots, w_n) \tag{2.6}$$

(The exact type is omitted here, see [MS01])

A closer look at type (2.4), Lemma 2.2.5 and type (2.6) points out to the fact that we can finally write down a formula for the probabilities p_i^j that depends only on the known quantities w_i and s^j. Indeed by replacing the type of Lemma 2.2.5 and type (2.6) in (2.4) we get:

Lemma 2.2.6. *Consider any Nash Equilibrium* $\mathbf{P}$, *in the fully mixed case. Then for all users* $i \in [n]$ *and links* $j \in [m]$,

$$p_i^j = \Big(1 - \frac{ms^j}{\sum_l s^l}\Big)\Big(1 - \frac{\sum_k w_k}{(n-1)w_i}\Big) + \frac{s^j}{\sum_l s^l} \tag{2.7}$$

But do the quantities p_i^j in (2.7) indeed represent probabilities? For them to do so, it must be that (1), for each user $i \in [n]$, $\sum_j p_i^j = 1$ and that, (2) for each link $l \in [m]$, $0 \leq p_i^j \leq 1$. Since the quantities were specifically derived for the case of fully mixed strategies, condition (2) should more accurately be stated as (2′): for each link $l \in [m]$, $0 < p_i^j < 1$. A straightforward calculation verifies that conditions (1) and (2′) may or may not hold, depending on the particular values of w_i and s^j. Hence, we obtain an inexistence result for FMNE:

Corollary 2.2.7. *Assume that there exist a user $i \in [n]$ and a link $j \in [m]$ such that*

$$\Big(1 - \frac{ms^j}{\sum_l s^l}\Big)\Big(1 - \frac{\sum_k w_k}{(n-1)w_i}\Big) + \frac{s^j}{\sum_l s^l} \notin (0,1)$$

Then there exists no FMNE.

We continue to show that the necessary condition determined in Corollary 2.2.7 is also sufficient, in the case of fully mixed strategies:

Theorem 2.2.8. *Assume that for all users $i \in [n]$ and links $j \in [m]$,*

$$\Big(1 - \frac{ms^j}{\sum_l s^l}\Big)\Big(1 - \frac{\sum_k w_k}{(n-1)w_i}\Big) + \frac{s^j}{\sum_l s^l} \in (0,1)$$

Then in the fully mixed case the above expression defines the probabilities p_i^j, which all together form a FMNE $\mathbf{P}$.

Proof. The assumption implies that for all users $i \in [n]$ and links $j \in [m]$, $0 < p_i^j < 1$. Thus, by definition of FMNE, we need to show that for any user i and link j, $c_i = c_i^j$. So fix any user i and link j. Then,

$$
\begin{aligned}
c_i^j &= \tfrac{w_i + \sum_{k\neq i} p_k^j w_k}{s^j} && \text{(by definition)}\\
&= \tfrac{w_i}{s^j} + \tfrac{1}{s^j}\textstyle\sum_{k\neq i} p_k^j w_k\\
&= \tfrac{w_i}{s^j} + \tfrac{1}{s^j}\textstyle\sum_{k\neq i}\Big(\Big(1 - \tfrac{ms^j}{\sum_l s^l}\Big)\Big(1 - \tfrac{\sum_{k'} w_{k'}}{(n-1)w_k}\Big) + \tfrac{s^j}{\sum_l s^l}\Big)w_k && \text{(by replacing } p_k^j)\\
&= \tfrac{w_i}{s^j} + \tfrac{1}{s^j}\Big(1 - \tfrac{ms^j}{\sum_l s^l}\Big)\textstyle\sum_{k\neq i}\Big(1 - \tfrac{\sum_{k'} w_{k'}}{(n-1)w_k}\Big)w_k + \tfrac{1}{s^j}\tfrac{s^j}{\sum_l s^l}\textstyle\sum_{k\neq i} w_k\\
&= \tfrac{w_i}{s^j} + \tfrac{1}{s^j}\Big(1 - \tfrac{ms^j}{\sum_l s^l}\Big)\textstyle\sum_{k\neq i} w_k\\
&\quad - \tfrac{1}{s^j}\Big(1 - \tfrac{ms^j}{\sum_l s^l}\Big)\tfrac{1}{n-1}\textstyle\sum_{k'} w_{k'}\sum_{k\neq i}\tfrac{w_k}{w_k} + \tfrac{1}{\sum_l s^l}\textstyle\sum_{k\neq i} w_k\\
&= \tfrac{w_i}{s^j} + \tfrac{1}{s^j}\Big(1 - \tfrac{ms^j}{\sum_l s^l}\Big)\textstyle\sum_{k\neq i} w_k\\
&\quad - \tfrac{1}{s^j}\Big(1 - \tfrac{ms^j}{\sum_l s^l}\Big)\tfrac{1}{n-1}\textstyle\sum_{k'} w_{k'}(n-1) + \tfrac{1}{\sum_l s^l}\textstyle\sum_{k\neq i} w_k\\
&= \tfrac{w_i}{s^j} + \tfrac{1}{s^j}\Big(1 - \tfrac{ms^j}{\sum_l s^l}\Big)\textstyle\sum_{k\neq i} w_k - \tfrac{1}{s^j}\Big(1 - \tfrac{ms^j}{\sum_l s^l}\Big)\textstyle\sum_{k'} w_{k'} + \tfrac{1}{\sum_l s^l}\textstyle\sum_{k\neq i} w_k\\
&= \tfrac{w_i}{s^j} - \tfrac{1}{s^j}\Big(1 - \tfrac{ms^j}{\sum_l s^l}\Big)w_i + \tfrac{1}{\sum_l s^l}\textstyle\sum_{k\neq i} w_k\\
&= \tfrac{1}{s^j}\tfrac{ms^j}{\sum_l s^l}w_i + \tfrac{1}{\sum_l s^l}\textstyle\sum_{k\neq i} w_k\\
&= \tfrac{m}{\sum_l s^l}w_i + \tfrac{1}{\sum_l s^l}\textstyle\sum_{k\neq i} w_k\\
&= \tfrac{1}{\sum_l s^l}\Big(mw_i + \textstyle\sum_{k\neq i} w_k\Big)\\
&= c_i && \text{(using Lemma 2.2.5)}
\end{aligned}
$$

□

Corollary 2.2.7 and Theorem 2.2.8 together establish the following:

Theorem 2.2.9. (Existence and Uniqueness of FMNE) *Consider the fully mixed case. Then, for all users $i \in [n]$ and links $j \in [m]$,*

$$\left(1 - \frac{ms^j}{\sum_l s^l}\right)\left(1 - \frac{\sum_k w_k}{(n-1)w_i}\right) + \frac{s^j}{\sum_l s^l} \in (0,1)$$

if and only if there exists a Nash Equilibrium, which must be unique and has an associated probability matrix $\mathbf{P} = [p_i^j]$, where

$$p_i^j = \left(1 - \frac{ms^j}{\sum_l s^l}\right)\left(1 - \frac{\sum_k w_k}{(n-1)w_i}\right) + \frac{s^j}{\sum_l s^l} \in (0,1)$$

, for each user $i \in [n]$ and link $j \in [m]$.

The above expressions for the probabilities p_i^j enjoy, as functions of the link capacities s^j and the player traffics w_i, a particularly insightful form. Their first term is the product of two factors: the first one $1 - \frac{ms^j}{\sum_l s^l}$ depends solely on link capacities, while the second one $1 - \frac{\sum_k w_k}{(n-1)w_i}$ depends solely on user traffics. Their second term $\frac{s^j}{\sum_l s^l}$ also depends solely on link capacities. The first factor in the first term vanishes if we take all link capacities to be equal to each other; thus, we conclude that, in the case of *uniform capacities*, the FMNE $\mathbf{P}$ is independent of the user traffics w_i. A corresponding elimination is lacking for the case where the user traffics are all equal to each other (*identical traffics*), since the second factor of the first term does not vanish. That means that FMNE do depend on link capacities, even in the case of identical traffics. This subtle difference manifests an inherent asymmetry between link capacities and user traffics, as parameters that define a FMNE.

One interesting final remark is that Theorem 2.2.8 implies that for the fully mixed case, NE can be checked for existence and evaluated (if it exists) in time $\Theta(nm)$, which is polynomial and not exponential as in the general case.

2.2.3 The Generalized Fully Mixed Nash Equilibrium

Having completed the study of the FMNE a very natural question arises. Why do we care in a so restricted form of equilibria? In fact one could say that the FMNE is really the exact opposite of a PNE, in the sense that in the former we want each player to spread his probability distribution over all links, whereas in the latter we want him to choose only one of them. Since the PNE is what we (and economics) really care about, why bother with the FMNE? The answer here is obvious: the FMNE provided us with the first closed form expression for p_i^j, with a uniqueness result, with some PoA bounds (as we shall see in the next section) and above all it resulted in the definition of another (more general) form of equilibria, which has even more interesting properties. This is the Generalized Fully Mixed Nash Equilibria (GFMNE) which was defined in [FKK+02] as follows:

Definition 2.2.10. A mixed strategy profile $\mathbf{P}$ is *generalized fully mixed* if there exists a subset $\mathbf{Links} \subseteq [m]$, such that for each pair of a player $i \in [n]$ and link $j \in [m]$, $S_i^j = 1 \Leftrightarrow j \in \mathbf{Links}$. If $\mathbf{P}$ is also a NE, we call it the GFMNE.

From the above definition it is obvious that the FMNE is the special case of GFMNE if $\mathbf{Links} = [m]$.

The reason we choose a more general form of equilibria is that the GFMNE always exists and in fact there is a nice polynomial time algorithm that computes this equilibrium, thus placing the problem of finding a NE in $\mathcal{P}$.

Before presenting the algorithm we need the following Lemma that follows the discussion in the last section (see Theorem 2.2.8):

Lemma 2.2.11. *Consider the case of FMNE under the model of arbitrary capacities and assume that all the traffics are identical. Define the normalized capacity $\widetilde{s^j}$ of link j to be $\widetilde{s^j} = s^j / \sum_l s^l$. Then for all links $j \in [m]$, $\widetilde{s^j} \in (\frac{1}{m+n-1}, \frac{n}{m+n-1})$ if and only if there exists a FMNE, which must be unique.*

Although Lemma 2.2.11 determines a collection of $2m$ necessary and sufficient conditions for a FMNE, the fact that all normalized capacities sum up to 1 implies that each pair reduces to one condition (say the one establishing the lower bound for $\widetilde{s^j}$). Furthermore all m conditions hold, if the one for $\min_j s^j$ holds. Thus the above lemma provides us with a way to determine the existence of a FMNE (and thus of a NE) in $\Theta(m)$ time, by finding the minimum s^j and checking whether $\widetilde{s^j} > \frac{1}{m+n-1}$.

With the aid from the above lemma we get:

Theorem 2.2.12. *Assume that all traffics are identical. Then the problem of finding a NE (GFMNE) by computing its supports is in $\mathcal{P}$.*

Proof. We present an algorithm A_{gfmne} that computes a GFMNE:

0. Sort the capacities of the links in non-increasing order: $s^1 \geq \ldots \geq s^m$ and compute all normalized capacities. Take the set **Links** $= [m]$.

1. Take the minimum capacity link and check the condition stated in Lemma 2.2.11. If the condition is true then we have found a GFMNE for **Links** and we stop. Else we go to step 2.

2. We drop the slowest link m' (link with minimum capacity) and we repeat step 1 with **Links** $\leftarrow$ **Links**$\backslash\{m'\}$.

The above algorithm studies all generalized fully mixed strategy profiles where the set **Links** consists each time of the $m', 1 \leq m' \leq m$ fastest links. Hence to establish correctness for A_{gfmne} we need to show that at least one of those generalized fully mixed strategy profiles is a GFMNE. We argue inductively on m:

- if $m = 1$ then it is trivial to see that the only NE is for all players to assign their traffic on the unique link

- let our assumption hold for $m = k$. Then for $k + 1$ we have: if for the slowest link the condition of Lemma 2.2.11 holds then we are done. If not, then the algorithm removes this link and we are left with k links. But then, by inductive hypothesis, there must be a GFMNE for the set of those k fastest links. A close look now at the definition of the GFMNE and the fact that every subset of $[k]$ is also a subset of $[k+1]$ as well, yields the result

Finally let us check the running time of the algorithm. In the preprocessing step we make a sorting which takes $\Theta(m \log m)$ time and the computation of the normalized capacities which takes $\Theta(m)$ time. Then steps 1 and 2 require constant time since they only involve two comparisons in order to check the validity of the conditions for a NE. Since these steps are executed at most m times the total running time of the algorithm is $\Theta(m \log m)$. $\square$

The above theorem gives a partial (for the case of identical traffics) answer to the complexity problem of computing a NE. In order to do so we had to define two new more restricted forms of equilibria, the GFMNE -which always exists- and the FMNE -which does not always exist. Apart from providing us with a powerful tool to study the complexity of finding a NE, the FMNE gave rise to a very interesting conjecture. As we have stated in the beginning of section 2.2, the structure of FMNE is such that it favors collisions of users across the links, hence increasing the social cost. So Fotakis et. al. conjectured in [FKK$^+$02] the following:

Conjecture 2.2.13. *Consider the case of identical traffics and arbitrary link capacities. Then for any instance such that a FMNE* $\mathbf{F}$ *exists and for any associated NE* $\mathbf{P}$*,* $\mathbf{SC}(\mathbf{P}) \leq \mathbf{SC}(\mathbf{F})$*, where by* $\mathbf{SC}$ *we denote the social cost.*

Although the **SC** will be formally defined in the next section, the above conjecture is easy to understand: it says that among all possible NE for an instance of the problem, the worst one -in terms of social benefit- is the FMNE.

The above has been nicely handled in [GLM$^+$03], whose results provide substantial evidence for Conjecture 2.2.13 and a complete proof for the case of PNE. A special case was also handled in [FKK$^+$02] where the following result appears:

Proposition 2.2.14. *Consider the model of uniform capacities and assume* $n = 2$*. Then the worst NE is the FMNE.*

A careful case analysis on the supports of the two players and of the structure of NE can indeed yield the above result. Furthermore [FKK$^+$02] showed that the social cost of a FMNE is within a constant factor from the worst case social cost.

2.2.4 The Pure Nash Equilibrium

We now study the case where each player has a pure strategy, he picks namely one link to route his traffic. This case is very important because of the applications of the notion of PNE.
We shall present the work of [FKK$^+$02] on the subject. We start with an existence result:

Theorem 2.2.15. *In the KP-model there exists at least one PNE.*

Proof. Consider the universe of pure strategy profiles. Each such profile induces a *sorted* expected latency vector $\mathbf{M} = (M^1, \ldots, M^m)$, such that $(M^1 \geq \ldots \geq M^m)$ (rearrangement of links is necessary). Of all the possible vectors $\mathbf{M}$ consider the lexicographically minimum, say $\mathbf{M_0}$. We claim that this vector corresponds to a pure strategy profile $\mathbf{P_0}$ that is a PNE. Assume that it does not: then there exists one player i, who has picked link j and who has incentive to deviate to link k, namely

$$c_i^j > c_i^k \Leftrightarrow^{(2.2)} M^j + (1 - p_i^j)w_i > M^k + (1 - p_i^k)w_i \tag{2.8}$$

where we considered unit capacities wlog. Since we have pure strategy profiles, $p_i^j = 1$ and $p_i^k = 0$. Hence equation (2.8) results in:

$$M^j > M^k + w_i > M^k \tag{2.9}$$

Now let us construct from $\mathbf{P_0}$ the new pure strategy profile $\widehat{\mathbf{P_0}}$ which differs from $\mathbf{P_0}$ only in links j and k, i.e. player i has moved from j to k. Then it is obvious that the expected latency of j decreases and the one of k increases. Let $\widehat{\mathbf{M_0}}$ denote the induced sorted expected latency vector (note that the indices of the sorted elements in this vector are not necessarily ordered 1,...,m, like in $\mathbf{M_0}$). We have that $\widehat{M^j} = M^j - w_i < M^j$ and $\widehat{M^k} = M^k + w_i < M^j$ (from (2.9)). Now since $\mathbf{M_0}$ is sorted in non-increasing order and $M^j > M^k$, it follows that M^j precedes M^k in $\mathbf{M_0}$. But $\widehat{\mathbf{M_0}}$ differs from $\mathbf{M_0}$ only in the positions of $\widehat{M^j}$ and $\widehat{M^k}$, which are both smaller than M^j and they will follow it in $\widehat{\mathbf{M_0}}$. So the first j elements of $\widehat{\mathbf{M_0}}$ stay the same as in $\mathbf{M_0}$. The j-th entry may be either $\widehat{M^j}$ or $\widehat{M^k}$ or some other element of $\mathbf{M_0}$, following M^j (i.e. smaller than M^j). In any case the resulting sorted expected latency vector $\widehat{\mathbf{M_0}}$, has its first $j-1$ entries equal to the ones of $\mathbf{M_0}$ and the j-th entry smaller. Then $\widehat{\mathbf{M_0}}$ is lexicographically smaller than $\mathbf{M_0}$, which is a contradiction.
Of course this proof can be easily generalized in the case of arbitrary speeds. □

Although the proof of Theorem 2.2.15 is based on an algorithmic procedure, the algorithm implied is inefficient in the following sense: since each player can pick among m strategies the total number of pure strategy profiles is m^n, i.e. exponential. However we can do better as the following theorem states.

Theorem 2.2.16. *In the KP-model, the problem of finding a PNE by computing its supports is in $\mathcal{P}$.*

Proof. The algorithm A_{pure} we present is a typical greedy algorithm: it considers all the player weights to be sorted $w_1 \geq \ldots \geq w_n$ and it allows player 1 (the heavier player) to pick his link first, in such a way that his own latency is minimized. Then goes player 2 and so on. The key idea is to settle the heavier players first. Player i then picks his link according to the choices of the previous (heavier) players. It is obvious that the running time of the algorithm is $\Theta(n \log n)$ (due to the sorting) and that in the end of the algorithm each player will have assigned his traffic to one link with unit probability. It suffices to show that the final pure strategy profile is indeed a PNE. We argue inductively on the number of iterations i of the main loop of A_{pure}. We show that the system is in PNE after each such iteration.

1. If $i = 1$ then trivially we have a PNE.

2. Let the inductive hypothesis hold for $i = k$. In iteration $k+1$ user $k+1$ picks a link j such that for all links l:

$$c_i^j \leq c_i^l \Leftrightarrow \frac{L^j + w_{k+1}}{s^j} \leq \frac{L^l + w_{k+1}}{s^l} \tag{2.10}$$

where by L^j we denote the load of link j until this iteration (which is not estimated but well known!). Now let us prove that there is no user with an incentive to deviate. User $k+1$ has no such incentive since he picked his link last. For the sake of contradiction we assume that there exists a player $p \leq k$ which has an incentive to deviate after the move of player $k+1$. In order for this to happen, player $k+1$ must route his traffic on the same link as player p, or else the load of player's p link will remain the same and the load of some other link will increase, causing p no incentive to deviate. Now since p has routed his traffic on link j and he wants to deviate there must exist a link y such that

$$\frac{L^j + w_{k+1}}{s^j} > \frac{L^y + w_p}{s^y} \tag{2.11}$$

But we have $w_p \geq w_{k+1}$ which yields

$$\frac{L^y + w_p}{s^y} \geq \frac{L^y + w_{k+1}}{s^y} \tag{2.12}$$

Combining inequalities (2.11) and (2.12) we get

$$\frac{L^j + w_{k+1}}{s^j} > \frac{L^y + w_{k+1}}{s^y} \tag{2.13}$$

which contradicts (2.10).

□

Finally the following theorem can be proved via reduction from BIN PACKING (see e.g. [Pap]).

Theorem 2.2.17. *Finding the supports for the best and the worst PNE are both $\mathcal{NP}$-hard problems in the KP-model.*

2.3 Studying the Price of Anarchy

After having sufficiently studied the structure of equilibria we move on to studying their quality. In order to do so we will first define some measures of the efficiency of an equilibrium, such as the *Social Cost* **SC** and we will then examine separately the cases of 2 and m links. The reasons for doing so, are mostly historical. We will follow the structure of [KP99] for both cases and we will then present the answers of [KMS03] and [CV02] to some questions about the case of m links, originally posed in [KP99].

2.3.1 Definitions

In our effort to quantify the inefficiency of equilibria, we need a quantity, or more precisely an objective function, that is indicative of the quality of a selfish outcome in a network of parallel links. One such quantity can be defined as the maximum latency that is observed on a network link. Another (commonly used) idea would be to consider as an objective function the sum of all delays on the network links. In our case we choose for the max criterion, so the objective function $C(j_1, \ldots, j_n)$,with $(j_1, \ldots, j_n) \in [m]^n$ can be formalized as follows:

$$C(j_1, \ldots, j_n) = \max_{j \in [m]} \frac{\sum_{k:j_k=j} w_k}{s^j} \tag{2.14}$$

This way we have a well-defined optimization problem, in which we wish to minimize $C(j_1, \ldots, j_n)$. That is we wish to find the *social optimum* (henceforth denoted by **OPT**), which is the minimum value of the maximum latency over all links $j \in [m]$, among all pure strategies profiles. That is:

Definition 2.3.1 (The Optimum).

$$\mathbf{OPT} = \min_{(j_1,\ldots,j_n) \in [m]^n} C(j_1, \ldots, j_n) = \min_{(j_1,\ldots,j_n) \in [m]^n} \max_{j \in [m]} \frac{\sum_{k:j_k=j} w_k}{s^j} \tag{2.15}$$

Equation (2.15) gives us the best possible outcome of a network with m parallel links, which we could achieve, were there a central authority that could force all agents to make the right choice for the social welfare. Unfortunately our users are selfish (which means that they are only interested in maximizing their own profit) and we can afford no central authority. The reason for not even considering the case of an authority is the exact same reason that favors distributed computations against centralized ones: every form of central control in a network is bound to have a negative impact on the network's speed.

Taking into account then that **OPT** is hard to reach with selfish agents present, what is the *estimated social cost* (henceforth denoted by **SC**) that is associated to a distribution over $[m]$ for each player (namely to a probability matrix **P**)?

The answer is:

Definition 2.3.2 (The Social Cost).

$$\mathbf{SC}(\mathbf{P}) = E[C(j_1, \ldots, j_n)] = \sum_{(j_1,\ldots,j_n)\in[m]^n} \left(\prod_i p_i^{j_i} \max_{j\in[m]} \frac{\sum_{k:j_k=j} w_k}{s^j} \right) \quad (2.16)$$

Remark 2.3.3. A more usual notation for the social cost and the optimum include the weight vector $\mathbf{w} = (w_1, \ldots, w_n)$: $\mathbf{SC}(\mathbf{w}, \mathbf{P})$ and $\mathbf{OPT}(\mathbf{w})$, since both quantities depend on it. Here we use a simpler notation and we skip $\mathbf{w}$.

Remark 2.3.4. We notice that the above definition considers the social cost to be the expected maximum cost and not the maximum expected cost $(\max_j M^j)$. But are those two quantities essentially different? The answer is affirmative. In fact $\max_j M^j \leq \mathbf{SC}$ or even $\max_j M^j << \mathbf{SC}$. Indeed let us consider a network with m links and just one player with weight w. Then the **SC** (expectation of the maximum load) is w, whereas the $M^j = w/m$ for all links j. If the number of links gets very big M^j tends to zero and **SC** remains w. On the other hand if we allow only pure strategies, i.e. force each player to pick one link, it is easy to verify then that $\max_j M^j = \mathbf{SC}$. However in the general case the inequality holds and therefore it is preferable to define the **SC** as the expected maximum cost.

We are now ready to define the *coordination ratio* or as it is commonly known, the *Price of Anarchy* (denoted by PoA):

Definition 2.3.5 (The Price of Anarchy or Coordination Ratio).

$$PoA = \sup_{\mathbf{P}\, is\, NE} \frac{\mathbf{SC}(\mathbf{P})}{\mathbf{OPT}} \quad (2.17)$$

Namely the PoA is a size (greater ore equal than one) that tells us how bad a selfish outcome can actually be. In order to do so, it computes the **SC** for every NE of the game, finds the cost of the worst NE (see the sup in 2.17) and compares it to the **OPT** of the game. The bigger the PoA, the larger a possible deviation from the optimum solution.
By now it must be clear that it is of crucial importance to ensure an upper bound on the PoA. This would mean that the selfish outcome of a game cannot be too far away from the optimum solution, and thus it cannot be too bad.

More on the Social Cost

Having formally defined the **SC** the following question arises: how effective can the computation of the **SC** be in an instance of the game? The answer follows and it was provided by [FKK+02]:

Theorem 2.3.6. *Given an instance of the problem and a NE* $\mathbf{P}$ *we want to compute its social cost. This problem is* $\sharp\mathcal{P}$*-complete when restricted to mixed equilibria.*

Proof. We will prove the theorem through a reduction from the following problem: given a set of integer weights $J = \{w_1 \ldots, w_n\}$ and an integer $C \geq \frac{\sum_i w_i}{2}$ count the number of subsets of J with total weight at most C. This problem corresponds to counting the number of solutions of a KNAPSACK instance, which is a $\sharp\mathcal{P}$-complete problem (see e.g. [Pap]). The way to do the reduction is to define n Bernoulli random variables Y_i taking the values w_i and 0 with probability 1/2 each: these probabilities denote whether w_i is considered to be a member of a subset of J or no. Consider now the sum $Y = \sum_i Y_i$ of those variables. Estimating the probability $Pr[Y \leq C]$ is equivalent to finding the fraction (and hence the number) of subsets of J that have a total weight at most C. Thus the computation of $Pr[Y \leq C]$ is $\sharp\mathcal{P}$-complete. We next show that there is a way to compute $Pr[Y \leq C]$, if we know the **SC** of a given (mixed) NE, for an instance of our game.
So let us now consider an instance of the problem with $n+1$ agents and 3 links, denoted by 0, 1 and 2. Let agent 0 have a weight $C \geq \frac{\sum_i w_i}{2}$ and each of the rest n players have weights w_i. It easy to confirm that if player 0 picks link 0 with $p_0^0 = 1$ and each other player picks the remaining two links with equal probability ($p_i^1 = p_i^2 = 1/2$) then this mixed strategy profile corresponds to a NE. Let us now consider the random variables Y_i to indicate the weight each player i assigns to one of the two remaining links 1 and 2, say wlog 1. Indeed with probability 1/2 player i assigns link 1 a weight w_i and with probability 1/2 he assigns no weight at all ($Y_i = 0$). But then the total weight assigned to link 1 is Y. Now notice that it is not possible for the loads of both links 1 and 2 to be more than C (or else their sum would exceed $\sum_i w_i$). Hence the only possibilities are the following: either both loads on 1 and 2 are at most C, or only one of them exceeds C (caution- we have to separate subcases here: either 1 or 2 exceeds C). Finally, it is obvious that the maximum load on a link will be at least C. The above discussion gives the intuition behind $\mathbf{SC}_1 = C + 2\sum_{B=C+1}^{\infty} Pr[Y \geq B]$. Now let us consider the instance where player 0 has a traffic of $C+1$. Then $\mathbf{SC}_2 = C + 1 + 2\sum_{B=C+2}^{\infty} Pr[Y \geq B]$. Some algebra and we result in $2Pr[Y \geq C+1] = 1 + \mathbf{SC}_1 - \mathbf{SC}_2$. Hence $Pr[Y \leq C] = 1 - Pr[Y \geq C+1] = \frac{1+\mathbf{SC}_1-\mathbf{SC}_2}{2}$. □

Finally it is quite easy to prove the following:

Theorem 2.3.7. *For the model of uniform capacities, there exists a fully polynomial, randomized approximation scheme to compute the* **SC**.

Proof. The idea is to pick a random variable that can be easily sampled (i.e. in polynomial time) and which gives a good approximation of the **SC**. This random variable is taken to be the maximum latency over all links. We repeat N times (where N must be shown to be polynomial) the following experiment: we assign each user to a link of his support according to the given probability matrix $\mathbf{P}$. For each experiment i we find the maximum latency (say L_i). The output of the algorithm is the mean $\frac{\sum_i L_i}{N}$. Since **SC** is the estimation of the maximum latency over all links, by the Strong Law of Large Numbers it follows that $\left|\frac{\sum_i L_i}{N} - \mathbf{SC}(\mathbf{P})\right| \leq \epsilon\mathbf{SC}(\mathbf{P})$, for any constant $\epsilon > 0$ provided that $N \geq \frac{\mathbf{SC}(\mathbf{P})}{\epsilon}$. In the next section we shall prove that $\mathbf{SC}(\mathbf{P}) = O(\frac{\log m}{\log\log m})\mathbf{OPT}$. Since $\mathbf{OPT} \leq \sum_i w_i$ we have that $\mathbf{SC}(\mathbf{P}) = O(\frac{\log m}{\log\log m})\sum_i w_i$ and hence it suffices to take $N = \frac{1}{\epsilon}O(\frac{\log m}{\log\log m})\sum_i w_i$. So for a polynomial number of samplings we get a fully polynomial approximation algorithm with constant ratio. □

2.3.2 The case of 2 links

Before moving on to giving the actual bounds for the PoA, we first give a few bounds on **OPT** which will later come handy. First of all we note that computing **OPT** is an NP-complete problem, as can be easily proved through a reduction from the partition problem. However for the purposes of upper bounding PoA here, it suffices to use two simple approximations of it:

$$\mathbf{OPT} \geq \max\{w_1, \sum_i w_i/m\} = \max\{w_i, \sum_j M^j/m\} \tag{2.18}$$

We remind the reader that we assume $w_1 \geq \ldots \geq w_n$. For the time being we consider links of unit capacities (as we can see from (2.18)). It is then easy to intuitively understand the above inequality: $\mathbf{OPT} \geq w_i$ since the traffic of each player i, must be somehow routed through the network, thus causing a delay of at least w_i (depending on whether it will be routed on a solo-link or not). On the other hand it is easy to see that for every pure strategies profile $(j_1, \ldots, j_n)$ we have that $\sum_i w_i$ is exactly the sum of the loads of all links $j \in [m]$, which is less than m times the maximum load $C(j_1, \ldots, j_n)$. Thus $\sum_i w_i \leq mC(j_1, \ldots, j_n)$. As this holds for all pure strategies profiles it follows that $\mathbf{OPT} \geq \sum_i w_i/m$. The equality in (2.18) follows from type (2.1).

We now move on and immediately give a result that lower bounds the PoA:

Theorem 2.3.8. *The coordination ratio for 2 links is at least 3/2.*

Proof. The proof of the theorem is trivial. Consider a game with only two agents, each with unit traffic. We have $w_1 = w_2 = 1$. It is easy to check that

$$\mathbf{P} = \begin{pmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{pmatrix}$$

is a NE. The expected maximum traffic then is $\mathbf{SC}(\mathbf{P}) = 3/2$, whereas the optimum is $\mathbf{OPT} = 1$ and can be achieved by allocating each job to its own link with unit probability. The above discussion provides us with an instance of the problem and with a NE $\mathbf{P}$, for which $\frac{\mathbf{SC(P)}}{\mathbf{OPT}} = \frac{3}{2}$. This implies that the PoA (as a supremum) must be at least this big. Consequently $PoA \geq 3/2$ $\square$

Remark 2.3.9. The above proving method works in general when trying to lower bound the PoA. That is, since the PoA is defined as a supremum over all equilibria, it suffices to give an instance of the problem and an equilibrium $\mathbf{P}$, with $\frac{\mathbf{SC(P)}}{\mathbf{OPT}} = x$ to show that $PoA \geq x$.

The proof for a matching upper bound is much more technical. Although the corresponding proof of [KMS03] for the case of m links (that we are going to present in the next section) also covers the problem for the case of 2 links, we present here the proof of [KP99], since it helps us gain great insight to the problem.

In order to move on with the proof we must define two new types of probabilities. First we define the *contribution probability*: the contribution probability q_i is equal to the probability that player i routes his traffic on the link of maximum load (if there are more than one maximum load links, we consider the lexicographically first, say). Clearly then we have $\mathbf{SC} = \sum_i q_i w_i$, since in the we have defined as cost the maximum of all link loads and the above expression gives the estimation of this size. We also define the *collision probability* t_{ik} as the probability of agent i and k routing their traffic on the same link.

The observation that both agents i and k can contribute to the social cost only if they collide leads to inequality 2.19:

$$\begin{aligned} 1 &\geq Pr[X_i = 1 \vee X_k = 1] \\ &= Pr[X_i = 1] + Pr[X_k = 1] - Pr[X_i = 1 \wedge X_k = 1] \\ &\geq Pr[X_i = 1] + Pr[X_k = 1] - t_{ik} \\ &\Rightarrow q_i + q_k \leq 1 + t_{ik} \end{aligned} \tag{2.19}$$

where X_i is a random variable indicating the link choice of player i.

The reason for defining the collision probability is that it has a very useful property, stated below. This property also holds for any number of links.

Lemma 2.3.10. *The collision probabilities of a NE of n agents and m links satisfy*

$$\sum_{k\neq i} t_{ik} w_k = c_i - w_i$$

Proof. Observe first that $t_{ik} = \sum_j p_i^j p_k^j$, as a union of m independent possibilities. Therefore

$$\sum_{k\neq i} t_{ik} w_k = \sum_{k\neq i}\Big(\sum_j p_i^j p_k^j\Big) w_k = \sum_j p_i^j \sum_{k\neq i} p_k^j w_k = \sum_j p_i^j (M^j - p_i^j w_i)$$

where the second equality follows from a summation rearrangement and the third one follows from type (2.1) (considering $L^j = 0$). From type (2.3) we can use $p_i^j w_i = M^j + w_i - c_i$. Although this result only holds if link j belongs to the support of player i ($p_i^j > 0$), when $p_i^j = 0$ there is no problem in substituting $p_i^j w_i$ with any term, since the product will always be zero. We then get

$$\sum_{k\neq i} t_{ik} w_k = \sum_j p_i^j (M^j - (M^j + w_i - c_i)) = \sum_j p_i^j (c_i - w_i) = c_i - w_i$$

□

Before proving the theorem we provide one more bound (which also holds for any number of agents and links). We have

$$\begin{aligned}
c_i &= \min_j c_i^j \\
&\leq \tfrac{1}{m}\textstyle\sum_j c_i^j && \text{because: } m \min_j c_i^j \leq \textstyle\sum_j c_i^j \\
&= \tfrac{1}{m}\textstyle\sum_j (M^j + (1 - p_i^j) w_i) && \text{see (2.2)} \\
&= \tfrac{\sum_j M^j}{m} + \tfrac{m - \sum_j p_i^j}{m} w_i \\
&= \tfrac{\sum_j M^j}{m} + \tfrac{m-1}{m} w_i && \text{probabilities sum up to 1} \\
&= \tfrac{\sum_i w_i}{m} + \tfrac{m-1}{m} w_i && \text{see (2.1)}
\end{aligned}$$

$$\Rightarrow c_i \leq \frac{\sum_i w_i}{m} + \frac{m-1}{m} w_i \tag{2.20}$$

Theorem 2.3.11. *The coordination ratio for any number of players and m=2 links is at most 3/2.*

Proof. Inequality (2.19) $q_i + q_k \leq 1 + t_{ik}$ implies:

$$
\begin{aligned}
\textstyle\sum_{k\neq i}(q_i+q_k)w_k &\leq \textstyle\sum_{k\neq i}(1+t_{ik})w_k \\
&= \textstyle\sum_{k\neq i} w_k + \sum_{k\neq i} w_k t_{ik} \\
&= \textstyle\sum_k w_k - w_i + c_i - w_i && \text{see Lemma 2.3.10} \\
&\leq \textstyle\sum_k w_k - w_i + \frac{\sum_k w_k}{2} + \frac{w_i}{2} - w_i && \text{see (2.20)} \\
&= \textstyle\frac{3}{2}\sum_k w_k - \frac{3}{2}w_i \\
&= \textstyle\frac{3}{2}\sum_{k\neq i} w_k
\end{aligned}
$$

Thus we have $\sum_{k\neq i}(q_i+q_k)w_k \leq \frac{3}{2}\sum_{k\neq i} w_k$ which can be written:
$\sum_k q_i w_k - q_i w_i + \sum_k q_k w_k - q_i w_i \leq \frac{3}{2}\sum_{k\neq i} w_k \Rightarrow$
$\sum_k q_k w_k \leq 2q_i w_i - q_i \sum_k w_k + \frac{3}{2}\sum_k w_k - \frac{3}{2}w_i \Rightarrow$
$\sum_k q_k w_k \leq (\frac{3}{2} - q_i)\sum_k w_k + (2q_i - \frac{3}{2})w_i$
But we have already noticed that $\mathbf{SC} = \sum_i q_i w_i$, so:

$$\mathbf{SC} \leq (\frac{3}{2} - q_i)\sum_k w_k + (2q_i - \frac{3}{2})w_i$$

Recall now type (2.18):

$$\mathbf{OPT} \geq \max\{w_i, \sum_i w_i/2\}$$

Assume that for some agent i, $q_i \geq \frac{3}{4}$, then $(2q_i - \frac{3}{2}) \geq 0$, implying $(2q_i - \frac{3}{2})w_i \leq (2q_i - \frac{3}{2})\mathbf{OPT}$. Of course $\frac{3}{2} - q_i \geq 0$, since q_i is a probability.
Then: $\mathbf{SC} \leq (\frac{3}{2} - q_i)2\mathbf{OPT} + (2q_i - \frac{3}{2})\mathbf{OPT} = \frac{3}{2}\mathbf{OPT}$.
Otherwise if $\forall i\, q_i \leq \frac{3}{4}$, then $\mathbf{SC} = \sum_i q_i w_i \leq \frac{3}{4}\sum_i w_i \leq \frac{3}{4}2\mathbf{OPT} = \frac{3}{2}\mathbf{OPT}$.
So in every case we have $\mathbf{SC} \leq \frac{3}{2}\mathbf{OPT}$ which implies that for every equilibrium $\mathbf{P}$, $\frac{\mathbf{SC(P)}}{\mathbf{OPT}} \leq \frac{3}{2} \Rightarrow PoA \leq 3/2$, □

We also have to consider the case of links of arbitrary capacities. This case was studied in [KP99] where a lower bound of $\phi = 1.618$ (the golden ratio) was derived. The authors conjectured that Theorem 2.3.11 can be appropriately generalized to the case of links of different speeds. Indeed in 2002 Czumaj and Vöcking proved that there is a tight upper bound for the general case of m links with arbitrary capacities. We are going to present these results in the forthcoming section.

2.3.3 The case of m links

In this section we consider the case of m links. This case was partially studied in [KP99], which provides us with a lower bound of $\Omega(\log m/\log\log m)$

and the conjecture that this lower bound is tight. This was indeed the case as it was proved independently by [KMS03] and [CV02]. The latter paper also contains the proof for the general case of m links with arbitrary capacities. Here we will present the work of [KP99] for the lower bound of the PoA, as well as some upper bounds provided in this paper. Although these bounds are not tight, they consist a proving method which is certainly worth reviewing because of the interesting mathematics it employs (mainly Probability Theory and especially the Azuma-Hoeffding bound, which has been also used by a lot of recent papers on this subject). After providing these results we sketch the proof of the tight lower bound conjecture for the case of uniform capacities and give the result for the case of arbitrary capacities without a proof. In this point we mention that before the final answer to this conjecture, a partial answer that involved FMNE was provided by [MS01] for both the cases of uniform and arbitrary capacities. In fact [MS01] confirmed the conjecture for the case of FMNE.

The tight lower bound and some first upper bounds

Theorem 2.3.12. *The coordination ratio for m identical links is*

$$\Omega(\log m/\log\log m).$$

Proof. Consider the case where there are m agents, each with an amount of traffic equal to unit: $w_i = 1$. Again $\mathbf{OPT} = 1$ and it can be achieved by allocating each job to its own link with unit probability. Also the uniform $m \times m$ probability matrix

$$\mathbf{P} = \begin{pmatrix} 1/m & \dots & 1/m \\ \vdots & \ddots & \vdots \\ 1/m & \dots & 1/m \end{pmatrix}$$

is a NE. But then we have a problem that is identical to the problem of throwing m balls into m bins and asking for the expected maximum number of balls in a bin. This problem is well studied and the answer is known to be $\Theta(\log m/\log\log m)$. Since we have provided an instance of the problem with $\frac{\mathbf{SC(P)}}{\mathbf{OPT}} = \Theta(\log m/\log\log m)$ it must be that $PoA = \Omega(\log m/\log\log m)$. $\square$

Conjecture 2.3.13. *The above lower bound is tight.*

Theorem 2.3.11 shows that the conjecture holds for $m = 2$. In the next part we will give the proof of the conjecture provided by Koutsoupias et. al. in [KMS03] for the case of uniform speed links. We will also sketch the (more general) proof provided by [CV02] for the case of general speed links. For now we present a weaker upper bound of the PoA provided by [KP99]. But first we need the following result:

Theorem 2.3.14. *For m uniform speed links, the expected load M^j of any link is at most* $(2 - 1/m)\mathbf{OPT}$.

Proof. The proof is trivial. Observe that

$$\begin{aligned}
M^j &\leq c_i && \text{see (2.2)}\\
&\leq \frac{\sum_i w_i}{m} + \frac{m-1}{m} w_i && \text{see (2.20)}\\
&\leq 2\mathbf{OPT} + \frac{m-1}{m}\mathbf{OPT} = && \text{see (2.18)}\\
&= (2 - 1/m)\mathbf{OPT}
\end{aligned}$$

□

We now prove an upper bound for the case of m identical links.

Theorem 2.3.15. *The coordination ratio of any number of agents and m identical links is at most $T = 3 + \sqrt{4m \ln m}$.*

Proof. Our main tool for this proof is the Azuma-Hoeffding inequality[2] which gives a concentration result for the values of martingales[3] that have bounded differences. By using this inequality we will show that the probability that the maximum load of a given link j exceeds $(T-1)\mathbf{OPT}$ is at most $1/m^2$. Then, using the union bound we argue that the probability that there exists one link $j \in [m]$, whose maximum load exceeds $(T-1)\mathbf{OPT}$, is at most $m \times 1/m^2 = 1/m$. So the probability that the maximum load on all links does not exceed $(T-1)\mathbf{OPT}$ is at least $1 - 1/m$.
The above discussion implies that the expected maximum load on any network link (i.e. the **SC**) is bounded by $(T-1)\mathbf{OPT}$ with probability $1-1/m$ and by $m\mathbf{OPT}$[4] with probability $1 - (1 - 1/m) = 1/m$. Hence

$$\begin{aligned}
\mathbf{SC} &\leq (1 - 1/m)(T-1)\mathbf{OPT} + 1/m(m\mathbf{OPT})\\
&= (T-1)\mathbf{OPT} - 1/m(T-1)\mathbf{OPT} + \mathbf{OPT}\\
&= T\mathbf{OPT} - 1/m(T-1)\mathbf{OPT}
\end{aligned}$$

[2] Suppose $X_i, i = 1, 2, \ldots$ is a martingale and $|X_{t+1} - X_t| < c_t$. Then for all positive integers n and all positive reals x, $Pr[X_n - X_0 \geq x] \leq \exp\{-\frac{x^2}{2\sum_i c_i^2}\}$

[3] In probability theory, a martingale is a stochastic process (i.e., a sequence of random variables) such that the expected value of an observation at time $t+1$, given all the observations up to time t, is equal to the observation at time t,namely $E[X_{t+1}|X_t] = X_t$.

[4] Notice that the maximum possible load is $\sum_i w_i$, which is less than m **OPT** (2.18)

$\leq T\mathbf{OPT}$
$\Rightarrow PoA \leq T$

So it suffices to show that indeed the probability that the load of a given link j exceeds $(T-1)\mathbf{OPT}$ is less than $1/m^2$. Let X_i be a random variable denoting the contribution of agent i to the load of link j. Clearly $Pr[X_i = w_i] = p_i^j$ and $Pr[X_i = 0] = 1 - p_i^j$. The random variables $X_1, \ldots, X_n$ are independent. We are interested in upper bounding the probability $Pr[\sum_i X_i > (T-1)\mathbf{OPT}]$. In order to do so we intend to use the Azuma-Hoeffding inequality (unfortunately the good concentration bounds of sums of binomial variables are of no use here). However before applying the Azuma-Hoeffding inequality we must "fix" our random variables, so that they form a martingale satisfying the inequalities necessary conditions ($|X_{t+1} - X_t| < c_t$). Therefore we define the new random variables $Y_t = X_1 + \ldots + X_t + \mu_{t+1} + \ldots + \mu_n$, where by μ_i we denote $E[X_i]$. It is easy to verify that $E[Y_{t+1}|Y_t] = Y_t$: note that $Y_{t+1} = Y_t + X_{t+1} - \mu_{t+1}$; since $\mu_i = E[X_i]$ the result follows. Observe now that $|Y_{t+1} - Y_t| = |X_{t+1} - \mu_{t+1}| = |X_{t+1} - p_{t+1}^j w_{t+1}| \leq w_{t+1}$, since $X_{t+1} \in \{w_{t+1}, 0\}$. We then apply the Azuma-Hoeffding inequality which yields:

$$Pr[Y_n - Y_0 \geq x] \leq \exp\{-\frac{x^2}{2\sum_i w_i^2}\}$$

Note that

$$Y_n = \sum_i X_i$$

and that

$$Y_0 = \sum_i \mu_i = \sum_i E[X_i] = E[\sum_i X_i] = M^j \leq (2 - 1/m)\mathbf{OPT} \leq 2\mathbf{OPT}$$

where we used Theorem 2.3.14. The probability we want to estimate is: $\sum_i X_i \geq (T-1)\mathbf{OPT} \Leftrightarrow Y_n \geq (T-1)\mathbf{OPT} = (T-3)\mathbf{OPT} + 2\mathbf{OPT} \Rightarrow Y_n \geq (T-3)\mathbf{OPT} + Y_0 \Leftrightarrow Y_n - Y_0 \geq (T-3)\mathbf{OPT}$. Let $x = (T-3)\mathbf{OPT}$. Then:

$$Pr[\sum_i X_i > (T-1)\mathbf{OPT}] \leq Pr[Y_n - Y_0 \geq x] \leq \exp\{-\frac{x^2}{2\sum_i w_i^2}\}$$

Finally it is not hard to establish that
(1): $\sum_i w_i^2 \leq m w_1^2$, assuming $w_1 \geq \ldots w_n$ and
(2): $\sum_i w_i^2 \geq 1/m(\sum_i w_i)^2 = m(\sum_i w_i/m)^2$, from Cauchy-Schwarz

Hence $(1),(2) \Rightarrow m(\sum_i w_i/m)^2 \leq \sum_i w_i^2 \leq m w_1^2$
$\Rightarrow \sum_i w_i^2 \leq \max\{m w_1^2, m(\sum_i w_i/m)^2\} \leq m\mathbf{OPT}^2$
using (2.18).

Thus, the probability that the load of link j exceeds $(T-1)\mathbf{OPT}$ is at most

$$\exp\{-\frac{(T-3)^2\mathbf{OPT}^2}{2\sum_i w_i^2}\} \leq \exp\{-\frac{(T-3)^2\mathbf{OPT}^2}{2m\mathbf{OPT}^2}\} = \exp\{-\frac{(T-3)^2}{2m}\}$$

For $T = 3 + \sqrt{4m \ln m}$, this probability becomes $1/m^2$ and the proof is complete. □

The corresponding upper bound and the case of arbitrary speeds

In this section we are going to sketch the proof for the corresponding upper bound in the case of uniform speeds, as it appears in [KMS03]. This paper introduces a powerful technique called "ball fusion", which is essentially an extension of the classical "balls in bins" problem, when we consider balls of arbitrary weights and arbitrary probabilities of a ball choosing a bin. This technique also applies to the case of arbitrary link speeds, or even for general latency functions. However it cannot yield a result for the game we are studying, due to a series of other implications. The final answer for the case of arbitrary link capacities was given by [CV02], which attempts a complete different, quite more technical, approach to the problem. This attempt results in the same upper bound for the coordination ratio of the uniform case.

Koutstoupias, Mavronicolas and Spirakis in [KMS03] focus their attention on the class of mixed strategies, in which the expected latency M^j through each link is at most a constant multiple of $\mathbf{OPT}$. These mixed strategies profiles are called *approximate equilibria* and it is easy to see that for the game we are studying, all NE belong to this class, since $M^j \leq 2\mathbf{OPT}$ (see Theorem 2.3.14). Of course the inverse does not hold, that is, there exist approximate equilibria that are not NE. The reason for considering this more general class of equilibria is that the following bounds hold in an obvious way:

$$\left.\begin{array}{ll} \text{from type (2.18):} & \max_i w_i \leq \mathbf{OPT} \\ \text{from definition of approx. equal:} & \max_j M^j \leq 2\mathbf{OPT} \end{array}\right\} \Rightarrow$$

$$\max\{\max_i w_i, \max_j M^j\} \leq 2\mathbf{OPT} \tag{2.21}$$

So it suffices to show the following:

Theorem 2.3.16.

$$\forall \mathbf{P} : \mathbf{SC}(\mathbf{P}) = O(\frac{\log m}{\log \log m}) \max\{\max_i w_i, \max_j M^j\}$$

because then: $\mathbf{SC}(\mathbf{P}) = O(\frac{\log m}{\log \log m})\mathbf{OPT} \Rightarrow PoA = O(\frac{\log m}{\log \log m})$

The analysis to prove that $\mathbf{SC}(\mathbf{P}) \leq O(\frac{\log m}{\log \log m}) \max\{\max_i w_i, \max_j M^j\}$ consists of two major steps. In the first step (the ball fusion) we reduce the case of *arbitrary weights* to the case of *almost equal weights* (that is, where all weights are within a factor of 2 from each other). The method to achieve that is to "fuse" the two (currently) smallest balls[5] together to form a new, larger ball with weight equal to the sum of weights of the other 2, only if the resulting weight does not exceed the $2\max_i w_i$ in the original game. When we cannot fuse any more balls we stop. For every pair of balls that we fuse, we assign to the new ball a probability in a way that the M^j is preserved (notice that M^j is really important for this proof, since it will eventually upper bound the social cost). We then show that the social cost of the resulting game is no less than the social cost of the original game.

The next step now is to upper bound the social cost for the case of *identical weights*. The social cost for this case is even worse (i.e. bigger) than the one in the case of *almost equal weights* (but at most twice this worse). The social cost for the case of *identical weights* is upper bounded by use of probabilistic arguments: they use techniques for estimating tails and Chernoff bounds [Che52] to show that the social cost of the identical weights case is at most $O(\frac{\log m}{\log \log m})$ of the maximum expected latency $\max_j M^j$. That together with (2.21) establishes the result.

A few more words about ball fusion: In each step of the ball fusion we replace two balls with their sum and assign to it a probability such that all M^j remain the same. Although all the expected traffics remain the same (and so does the maximum expected traffic $\max_j M^j$) the same does not hold for the expected maximum traffic, namely the **SC**. Indeed, since we now deal with bigger weights, we expect the **SC** to either increase or remain the same. This is indicative of the subtle difference between the terms "expected maximum traffic" and "maximum expected traffic" (the second one is always smaller then first one - see also Remark 2.3.4). In fact this difference is an important tool in the proof of [CV02].

[5] We often use the terms "balls" instead of "users" and "bins" instead of "links".

Now, in order to prove the claim that the social cost is indeed bigger in the case of *almost equal weights* Koutsoupias et. al. use an inductive argument on the number of balls and first show that the **SC** grows up after one fusion, if we limit ourselves in the pure strategies profiles. If we repeat this until no more balls can be fused (without exceeding the initial $2\max_i w_i$) we create a new set of weights which are all in the interval $[h/2, h]$, where $h = 2\max_i w_i$. Indeed, if there was a weight less than $h/2$ then it could be fused with the original ball of maximum weight $h/2$. This new set's **SC** is proved to be no less than the **SC** of the original set. If we now keep the possibilities for the new set of balls the same and increase all weights to h (so that they have *identical weights*) then the **SC** will be even bigger (but at most twice this big, since we are at most doubling the weights).

The next step is to prove Theorem 2.3.16. The intuition behind this theorem is that the **SC** is maximized if all balls fall into one bin (namely if we have a big $\max_j M^j$) or if there exists a particularly heavy ball (namely if we have a big $\max_i w_i$). So we can upper bound **SC** using those two terms. The results of ball fusion allow us to focus only on the case of *identical weights* (since it upper bounds the **SC** of the *arbitrary weights* case) and even consider all the weights to be 1. In fact we can prove the following Lemma, using probabilistic arguments that are commonly used in the theory of random allocations.

Lemma 2.3.17.

$$\forall \mathbf{P} : \mathbf{SC}(\mathbf{P}) \leq (\frac{2e \log m}{\log \log m} + 1) \max\{1, \max_j M^j\}$$

Obviously if the identical weights are not set to unit but to $\max_i w_i$ the above lemma implies Theorem 2.3.16.

Finally we give the following theorem for the case of arbitrary link capacities without a proof:

Theorem 2.3.18. *[CV02] The PoA for a parallel link network with arbitrary link capacities is* $\Theta(\frac{\log m}{\log \log \log m})$.

Chapter 3

Congestion Games

In this Chapter we focus on congestion games, which are the natural generalization of the network of parallel links of Chapter 2. In fact congestion games have been studied long before the actual KP-model was considered in 1999, mostly as a form of a class of games, called “potential games”. However, the interest on these games grew up at the late 90’s along with the upcoming trend of algorithmic game theory.
In this thesis we discuss the main types of congestion games, their relationship with potential games, we highlight the fact that they always admit a PNE, we present some interesting results concerning its tractability and ofcourse we study their Price of Anarchy.

3.1 Definitions

We start off by giving the intuition behind congestion games. We can imagine a congestion game either as a situation, where we want to route traffic through a network -which in contrast with KP-model, consists of arbitrary many links and nodes, in arbitrary topology-, or as a situation, where we have some resources and some players, who want to use these resources. In either case, the resources (network edges resp.) have cost functions (i.e. latency functions), which are non-decreasing. Hence, the more players pick this resource (equiv. the more traffic is routed through this link) the bigger the delay for all the players who choose it.
Formally a congestion game consists of:

- a set of resources E (possibly network edges)
- a set of players N
- the resource delays d_e

and for each player i

- an action set $A_i \subseteq 2^E$
- a weight (traffic demand) w_i

Using all of the above we can define a *weighted* congestion game, i.e. where the players have different weights. Henceforth however, when we refer to congestion games, we shall implicitly assume that all players have unit weights. The reason for that is that the majority of results that we are going to present here, refer to unweighted congestion games.
Some more useful notation concerns the way players act and the outcomes of their choices:

- let $A = (a_1, \ldots, a_n), a_i \in A_i$ be a profile of pure strategies
- let $P_e(A)$ be the set of players picking resource e in profile A
- let $n_e(A)$ be the number of players picking resource e in profile A. Obviously $n_e(A) = |P_e(A)|$
- define the load of resource e with respect to profile A to be $L_e(A) = \sum_{i \in P_e(A)} w_i$. For unweighted games this is simply $L_e(A) = n_e(A)$
- finally define the cost of each player to be $c_i(A) = \sum_{e \in A_i} L(e)$,for the strategy profile $A = (a_1, \ldots, a_i, \ldots, a_n)$

It is easy to extend the notion of mixed strategies in the case of congestion games. However we are not going to do that. The reason for that is that, as we shall soon see, the congestion games always possess PNE, so we won't need to concentrate on the mixed case. The truth is that there has been recent research on the case of mixed equilibria as well, which is mainly due to the fact that -for most cases of congestion games- the PNE cannot be easily computed. In this thesis we shall briefly present some of the results concerning the mixed PoA.

Congestion games are usually separated into many categories based on various criteria. The main types of congestion games are the following:

- **Symmetric** vs **Asymmetric** congestion games: in the symmetric congestion games, all players have the same action set and the same weights, i.e. they are indistinguishable.

- **Network** vs **General** congestion games: in network congestion games, we consider the resources to be the edges of a directed graph, each player has a sink and a source vertex (the commodities: (s_i, t_i)) and the action sets for each player are $s_i - t_i$ paths.

- **Single** vs **multi**-commodity (network) games. In single-commodity network games there is only one commodity $s - t$ for all players.

3.2 Equilibria

In this section we shall discuss the main characteristics of congestion games equilibria. We point out a theorem that guarantees their existence and then we investigate their tractability. We briefly present some results on weighted congestion games too.

3.2.1 Existence of Equilibria

We shall prove the existence of equilibria for a more general type of games, called **potential games** and we shall then show that every (unweighted) congestion game is in fact a potential game, or, as we say, admits an exact potential.
Let us first define the exact potential:

Definition 3.2.1. A function $\Phi : E \to \mathbb{R}$ is an exact potential for game G iff

$$\forall i \in N, \forall a'_i \in A_i, \forall A : c_i(A) - c_i(A^{-i}, a'_i) = \Phi(A) - \Phi(A^{-i}, a'_i),$$

where (A^{-i}, a'_i) is the standard notation in Game Theory for the strategy profile that results from A by replacing its i-th entry with a'_i.

The crucial observation is the following:

Proposition 3.2.2. *Every game that admits an exact potential possesses a PNE.*

Proof. We can start from an arbitrary pure strategy profile A, and at each step one player reduces its cost. That means, that at each step, Φ is reduced identically. Since Φ can accept a finite amount of values, it will eventually reach a local minimum. At this point, no player can achieve any improvement, and we reach a PNE. $\square$

Hence we have a well defined class of games, the potential games, which all have the important property of possessing a PNE. Already from 1973 Rosenthal had proved the following theorem, which states that every congestion game is an exact potential game.

Theorem 3.2.3. *[Ros73]*
Every unweighted congestion game admits an exact potential.

Proof. Let $\Phi(A) = \sum_{e\in E}\sum_{k=1}^{n_e(A)} d_e(k)$, $a_i' \in A\backslash a_i$ and $A' = (A^{-i}, a_i')$. Then we have:

$$
\begin{aligned}
\Phi(A) - \Phi(A') &= \sum_{e\in E}\sum_{k=1}^{n_e(A)} d_e(k) - \sum_{e\in E}\sum_{k=1}^{n_e(A')} d_e(k) \\
&= \sum_{e\in\cup_i a_i'\backslash a_i}\left[\sum_{k=1}^{n_e(A)+1} d_e(k) - \sum_{k=1}^{n_e(A)} d_e(k)\right] \\
&\quad + \sum_{e\in\cup_i a_i\backslash a_i'}\left[\sum_{k=1}^{n_e(A)-1} d_e(k) - \sum_{k=1}^{n_e(A)} d_e(k)\right] \\
&= \sum_{e\in a_i'\backslash a_i} d_e(n_e(A)+1) - \sum_{e\in a_i\backslash a_i'} d_e(n_e(A)) \\
&= \sum_{e\in a_i'} d_e(n_e(A')) - \sum_{e\in a_i} d_e(n_e(A)) \\
&= c_i(A') - c_i(A)
\end{aligned}
$$

where we exploit the fact that $\forall e \in E\backslash(a_i \cup a_i')$ and $\forall e \in a_i \cap a_i'$ the load of those resources remains the same in A and A'. Additionally $\forall e \in a_i'\backslash a_i$, $n_e(A') = n_e(A) + 1$ and $\forall e \in a_i\backslash a_i'$, $n_e(A') = n_e(A) - 1$. □

So a natural question now is, what other games can be proven to have PNE by use of potential functions? Monderer and Shapley [MS96] have provided an early and devastating answer: only for (inconsequential generalizations of) congestion games can we have an exact potential functions, or as they stated:

Proposition 3.2.4. *[MS96] Every finite exact potential game is isomorphic to an unweighted congestion game.*

However there exist games (e.g. the party affiliation game defined in [FPT04]), where the Nash dynamics converges, i.e. there exists a PNE, and the game is no congestion game. Indeed in this case we cannot find an exact potential, but we can find an ordinal potential (or general potential function), for which we do not require that the differences are identical, but just that they have the same sign. The ordinal potential remains a sufficient condition for the existence of PNE. Moreover, in [FPT04] they show that, under the relaxed definition of potential, the class of potential games is much richer, essentially encompassing all of the class $\mathcal{PLS}$[1]: every problem in $\mathcal{PLS}$ corresponds to a game that admits an ordinal potential (and therefore possesses a PNE).

3.2.2 Computing a PNE in various congestion games

We move on now to discuss the complexity of finding a PNE. Our source for this section is [FPT04]. We shall give 1 positive result and 3 negative results:

- for the case of symmetric[2] network congestion games the problem of finding a PNE is in $\mathcal{P}$
- for the case of
  - asymmetric network congestion games
  - general symmetric congestion games
  - general asymmetric congestion games

  the problem of finding a PNE is $\mathcal{PLS}$-complete

The positive result is easy to understand and prove as follows:

Theorem 3.2.5. *There is a polynomial time algorithm for computing PNE in symmetric network congestion games.*

Proof. The algorithm computes the minimum of the potential function Φ. In order to do that we reduce our problem to the problem of finding the min-cost flow in a network, which can be solved in polynomial time. The reduction is simple and can be performed in polynomial time: for each edge in the original network construct n edges ($n =$ the number of players in the original network) all with capacity 1 and with costs $d_e(1), d_e(2), \ldots, d_e(n)$. It is easy to see that the cost of every flow corresponds to the value of Φ for the corresponding strategy profile, since it is integral (easy to prove) and since the parallel edges get filled from lower cost to higher cost (due to the min-cost). Hence a min-cost flow corresponds to a minimum of Φ. $\square$

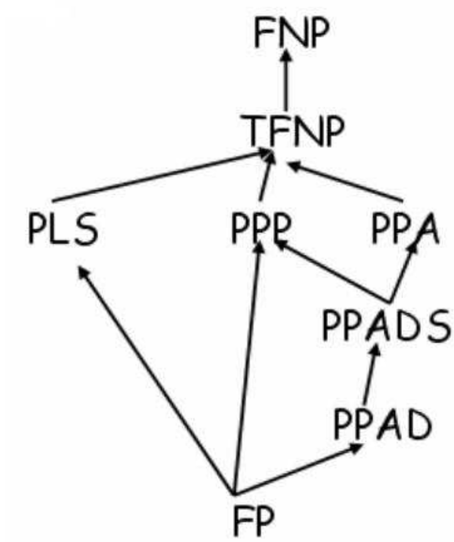


Figure 3.1: Complexity Hierarchy of Search Problems

To understand why the latter 3 results are negative, we need to discuss some things about complexity classes and the so-called inefficient proofs of existence. This discussion can be found in [Pap94], where the hierarchy of Figure 3.1 is presented. This hierarchy is used to describe search problems (and not decision problems), i.e. problems where we want to find a solution (and not just whether it exists or not). Actually Papadimitriou focused on the semantic class $\mathcal{TFNP}$ which (informally) is the class of all search problems which are guaranteed to have a solution, but where the solution seems hard to be found. In order for this to happen, we need some sort of non-constructive proof, that guarantees the solution, but does not provide us with an efficient algorithm for tracking it. This kind of proofs have been categorized based on the sort of argument one uses and the classes depicted in Figure 3.1 correspond exactly to those arguments.
Namely

- the class $\mathcal{PLS}$ (Polynomial Local Search) is based on "every finite directed acyclic graph has a sink". Other known complete problems for this class are POSNAE3FLIP and CIRCUITFLIP

- the class $\mathcal{PPP}$ (Polynomial Pigeonhole Principle) is based on "pigeonhole principle"

[1] which we shall see in a while
[2] symmetric here also implies a single-commodity network

- the class $\mathcal{PPA}$ (Polynomial Parity Argument) is based on "all graphs of max degree 2 have an even number of leaves". A known problem for this class is "given one Hamilton Path in a graph with odd degrees, find a second one" (existence is guaranteed)
- the class $\mathcal{PPAD}$ (Polynomial Parity Argument Directed) is the same as $\mathcal{PPA}$, only now the graph is directed and we are searching for a source or a sink. Known complete problems for this class are 2NE, Brower's and Kakutani's fix-point theorems, 3D-SPERNER
- the class $\mathcal{PPADS}$ (Polynomial Parity Argument Directed Sink) is the same as $\mathcal{PPAD}$, only now we are searching for a sink

We note that the problem of finding a NE in general 2-or-more player games (2NE) is complete for $\mathcal{PPAD}$, as is the problem of finding a PNE for the above mentioned types of congestion games complete for $\mathcal{PLS}$. This happens because in both cases, the existence of the equilibrium is guaranteed by an "inefficient proof".

Let us discuss now $\mathcal{PLS}$ a bit more since this is the class that we are going to use. $\mathcal{PLS}$ was independently defined in [JPY88] in order to describe problems where we need to "find some local optimum in a reasonable search space". Its ingredients are:

- a problem with a search space, i.e. a set of feasible solutions which has a neighborhood structure
- a poly-time algorithm s(x) which, given an instance x, computes an initial (arbitrary) solution
- a poly-time cost function c(x,s) that given an instance x and a solution s, it computes its cost
- a poly-time neighbor function g(x,s) that given an instance x and a solution s, either returns an other one in its neighborhood with lower cost, or "none" if none exist

It must be obvious that, using the above features, we can always find a local minimum, starting at some arbitrary solution and moving towards a better one at each step. Ofcourse the local minimum most probably will not be a global minimum, but in the case of PNE, a local minimum is all we are looking for (cf. the local minimum interpretation of NE as opposed to the global minimum interpretation of DE). So, why is $\mathcal{PLS}$-completeness a bad thing? The answer lies in thy number of steps of such a naive "gradient descent" method. It can be proven that

- finding a local optimum reachable from a specific state is $\mathcal{PSPACE}$-complete
- there are instances with states exponentially far from any local optimum

We are now ready to discuss the reductions for the three latter cases of congestion games. In order to prove those cases $\mathcal{PLS}$-complete we need a new kind of reduction, the $\mathcal{PLS}$-reduction, whose aim is to map the neighboring structures and the local minima of one instance to those of another, and an initial $\mathcal{PLS}$-complete problem: the POSNAE3FLIP.

Definition 3.2.6. POSNAE3FLIP[SY91]
Given a boolean formula in CNF, with all its clauses containing 3 positive literals, find a truth assignment s.t. by flipping the value of just one variable, we cannot reduce the total weight of "bad" clauses: clauses that have all variables equal to 1 or 0.
Example: for $(x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee x_2 \vee x_4) \wedge (x_1 \vee x_2 \vee x_5) \wedge (x_3 \vee x_4 \vee x_5)$, with clause weights 10, 10, 10, 2 respectively, one solution is $(x_1, x_2, x_3, x_4, x_5) = (1, 1, 0, 0, 0)$ of cost 2 and another solution is $(x_1, x_2, x_3, x_4, x_5) = (1, 0, 1, 0, 0)$ of cost 0. However getting from one solution to another would require two bitflips, which is why the solutions are not neighbors, and are hence both of them local minima.

Theorem 3.2.7. *For the case of*

1. *general asymmetric congestion games*
2. *general symmetric congestion games*
3. *asymmetric network congestion games*

the problem of finding a PNE is $\mathcal{PLS}$-complete

Proof. 1. Reduction from POSNAE3FLIP: Each variable x of the formula corresponds to a player of the game. For each clause create two resources e_c e'_c. The strategy set of player i contains two actions/subsets of resources: $A_i = \{\{e_c | x \text{ appears in clause } c\}, \{\{e'_c | x \text{ appears in clause } c\}\}$. Depending on the value of the corresponding variable, each player picks either the first or the second set of resources. From the above definition of strategy sets it is obvious that no resource can be played by more than 3 players. The resource delays are defined $d_e(0) = d_e(1) = d_e(2) = 0$ and $d_e(3) = w$, where w is the weight of the corresponding clause. It is now easy to check that every PNE of the game corresponds to a solution of POSNAE3FLIP: player i does not want to change his strategy (i.e. flip his value) because he cannot have a smaller cost; but the cost is determined only by resources with 3 players on them (all 0 or all 1), which correspond to "bad" clauses. So since this holds for all players, we have that the sum of costs of "bad" clauses cannot be improved by a deviation, i.e. a bitflip.

2. Reduction from the asymmetric case: We augment the network of the first case by adding n additional resources $e_1, \dots, e_n$, which have zero delay if they are picked by just one player and infinite delay in any other case: $d_e(1) = 0$ and $d_e(k) = M$ otherwise, where M a sufficiently large number. The new, **common** action set is $A = \bigcup_i \{s \cup \{e_i\} | s \in A_i\}$. The idea is that, since the new resources have very large costs if picked by two or more players, at PNE we expect each player to pick a different e_i. Indeed, if for example 2 players are crowded in an e_i, then there exists (at least) one e_j which is free and hence the players have a profitable deviation. So at NE every resource is picked by exactly one player. But then, thanks to the definition of A, we can identify the "anonymous" players of the symmetric case according to the strategy set A_i they use and match them with the corresponding players of the asymmetric case. Hence a PNE of the symmetric case is mapped to a PNE of the asymmetric case.

3. Reduction from WITNESSED XPNAE3FLIP: for the case of asymmetric network congestion games, the reduction is very complex. A first idea would be to follow the construction of the general asymmetric case, but take care to add some extra edges, so each variable-player traverses either all e_c edges, or all e'_c edges. The difficulty is to prevent a player from taking a path that doesn't correspond to a consistent assignment. In fact for a dense instance of POS-NAE-3SAT, this appears unavoidable. Then, in [FPT04] they notice that the original reduction for POSNAE3FLIP in [SY91] produces a very structured, sparse instance of POSNAE3FLIP. So, what they do is tweak the formulae produced by the [SY91] reduction and then carefully arrange the network so "non-canonical" paths are never a good choice. The resulting reduction has 39 variable types and 124 clause types and is omitted.

$\square$

3.2.3 Some results on weighted congestion games

This section contains some results concerning equilibria in weighted congestion games. The sources for this section are [FKS04], [FKS05] and [Mil96].

For the model we discussed thus far we first present two negative results:

Theorem 3.2.8. *[FKS04] There exist weighted single-commodity (network) congestion games with resource delays that are linear or 2-wise linear*[3] *functions of the loads, for which there is no PNE.*

Theorem 3.2.9. *[FKS04] There exist weighted single-commodity (network) congestion games which are not exact potential games, even with resource delays identical to their loads.*

and then two positive results:

Theorem 3.2.10. *[FKS04] For any weighted l-layered*[4] *network congestion game with resource delays identical to their loads, at least one PNE exists and can be computed in pseudo-polynomial time.*

Theorem 3.2.11. *[FKS05] For any weighted multi-commodity network congestion game with linear resource delays, at least one PNE exists and can be computed in pseudo-polynomial time.*

Finally we shall briefly discuss an alternative model which was proposed by Milchtaich back in 1996. In his model, Milchtaich considered resource delay functions which are not universal, but player-specific, i.e. they are of the form d_e^i, for all resources e and players i. In order to simplify the model, he made two crucial assumptions:

1. each player may choose only one resource from a pool E of resources (shared to all the players) for his service (this is exactly the KP-model of parallel links: [KP99]).

2. the incurred delay is monotonically non-decreasing with the number of players selecting it. Although they do not always admit a potential, these games always possess a PNE.

Two very important results for this model are the following:

Theorem 3.2.12. *[Mil96] Every unweighted congestion game on parallel links with player-specific resource delays possesses a PNE.*

Proposition 3.2.13. *[Mil96] For 3-players, 3-actions* **weighted** *congestion games with player-specific resource delays, there exist instances with no PNE.*

[3] i.e. the maximum of two linear functions

[4] a network where every (simple) directed s-t path has length exactly l and each node lies on a directed s-t path

3.3 Studying the Price of Anarchy

In this section we present some results about the PoA of congestion games. Driven by the fact that PNE always exist, Christodoulou and Koutsoupias proved in [CK05] a series of tight bounds for the pure PoA of (many types of) congestion games. Earlier Fotakis et. al.[FKS04] had proved some results on the mixed PoA which extended the results of the KP-model (using essentially the same techniques). We shall discuss some of these results here.

3.3.1 The Pure Price of Anarchy

In [CK05] the writers present some tight bounds for various types of congestion games. In order to reach these results they:

- handle general congestion games (both symmetric and asymmetric). They do not explicitly discuss network congestion games, although some of their results apply to this case as well.
- concentrate on the PPoA only. However some results extend to the case of mixed PoA as well.
- consider latency functions of the form: $d_e(x) = x$. This case extends immediately to the more general case of linear latency functions $d_e(x) = ax+b$ and the results can be generalized for bounded degree polynomials as well.
- consider both MAX and SUM as objective functions for the SC.

The results of [CK05] are summarized in the following Table:

	SUM	MAX
Symmetric	5/2	5/2
Asymmetric	5/2	$\Theta(\sqrt{n})$

Table 3.1: Main results of [CK05]

In this thesis we shall only present the proof for the asymmetric case, when we consider the average cost (i.e. SUM). The proof methodology for the other cases is similar.

Theorem 3.3.1. Lower bound
There are linear congestion games with 3 or more players with pure price of anarchy (PPoA) for the average social cost (SUM) equal to 5/2.

Proof. We will construct a congestion game for $N \geq 3$ players and $|E| = 2N$ facilities with PoA $= 5/2$.
We divide the set E into two subsets $E_1 = \{h_1, \ldots, h_n\}$ and $E_2 = \{g_1, \ldots, g_n\}$ each of N facilities. We define the strategy set of player i to be $A_i = \{\{h_i, g_i\}, \{g_{i+1}, h_{i-1}, h_{i+1}\}\}$. The optimal allocation is for each player to select the first strategy while the worst-case PNE is for each player to select the second strategy. It is not hard to verify that this is indeed a PNE and that each player has cost 5: resource g_{i+1} is picked by only one player, whereas resources h_{i-1}, h_{i+1} are picked by exactly two players. Since at the optimal allocation the cost of each player is 2 (each resource is picked by exactly one player), we have PPoA $= 5/2$. □

Before proving the upper bound we need the following Lemma which can be proved by induction.

Lemma 3.3.2. *For every pair of nonnegative integers α, β, it holds*

$$\alpha(\beta + 1) \leq \frac{1}{3}\beta^2 + \frac{5}{3}\alpha^2$$

Theorem 3.3.3. Upper bound
For linear congestion games, the pure price of anarchy (PPoA) of the average social cost (SUM) is at most $5/2$.

Proof. Let A be a profile that is a PNE and P an arbitrary (possibly optimal) profile. The cost of player i at PNE is $c_i(A) = \sum_{e \in A_i} n_e(A)$.
We want to bound the social cost, which we take to be the sum of the cost of the players:

$$SUM(A) = \sum_i c_i(A) = \sum_{e \in E} n_e^2(A)$$

(as follows by a simple reversal of the sums), with respect to the optimal cost

$$SUM(P) = \sum_i c_i(P) = \sum_{e \in E} n_e^2(P).$$

At PNE, the cost of player i should not decrease when the player switches to strategy P_i:

$$c_i(A) = \sum_{e \in A_i} n_e(A) \leq \sum_{e \in P_i} n_e(A^{-i}, P_i) \leq \sum_{e \in P_i} (n_e(A) + 1)$$

If we sum over all players i, we can bound the social cost as

$$SUM(A) = \sum_i c_i(A) \leq \sum_i \sum_{e \in P_i} (n_e(A) + 1) = \sum_{e \in E} n_e(P)(n_e(A) + 1)$$

With the help of Lemma 3.3.2, by setting $\alpha = n_e(P)$, $\beta = n_e(A)$ and summing for all i, the last expression is at most $\frac{1}{3}n_e(A)^2 + \frac{5}{3}n_e(P)^2 = \frac{1}{3}SUM(A) + \frac{5}{3}SUM(P)$, from which the result follows. $\square$

3.3.2 The Mixed Price of Anarchy

We end this chapter with some results on the mixed PoA. Although we have not formally defined the mixed case of congestion game, the generalization should be obvious.
Some interesting work on the mixed PoA can be found in [FKS04], which we shall briefly discuss here. In this paper, Fotakis et. al. focus their interest on weighted l-layered network congestion games where the **resource delays are identical to their loads**. This case consists a highly non-trivial generalization of the well-known KP-model. The main reason why they focus on this specific category of resource delays is that there exist instances of (even unweighted) congestion games on layered networks that have unbounded price of anarchy even if we only allow linear resource delays. In fact the writers modified an example given in [RT02] where the price of anarchy is indeed unbounded (see Figure 3.2). This instance can be easily converted into an l-layered network. The resource delay functions used are either constant, or M/M/1-like delay functions. However, we can have equally bad results even with linear resource delay functions. Hence in [FKS04] they focus on resource delays equal to their loads and prove the following interesting theorem.

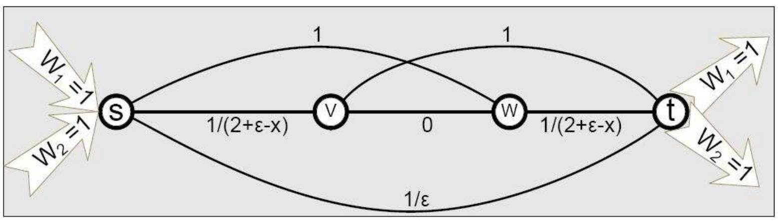


Figure 3.2: An example with unbounded PoA

Theorem 3.3.4. *[FKS04] The PoA of any weighted, l-layered network congestion game with resource delays equal to their loads is* $\Theta(\log m/\log\log m)$

In another paper they also proved:

Theorem 3.3.5. *[FKS05] The PoA of any unweighted, single-commodity network congestion game with resource delays* $d_e(x) = a_e \cdot x, a_e \geq 0$, *is* $O(\log m/\log\log m)$

Remark 3.3.6. One could say that, since we are basically interested in PNE, which, in the case of congestion games, always exist and since we have a very good, tight bound for the PPoA, why should we even consider the mixed PoA? One answer here would be that it does not suffice to know that a PNE exists, one must also be capable of finding it in reasonable time. Since this is not possible for some types of congestion games, we should look at the big picture as well. Another answer should be that the [CK05] paper succeeded the [FKS04] paper.

Remark 3.3.7. By the discussion about congestion games so far, the significance of allowing distinguishable players (i.e. players with different action sets, or with different traffic demands, or both) must be now obvious. We saw, how allowing the players to have distinct weights may lead to games with no PNE or with large PoA. Finally, the last two theorems and the familiar result of $\Theta(\log m/\log\log m)$ seem to point out to some kind of "equivalence" between games with unit-demand players on arbitrary networks with delays equal to their loads and games with players of varying demands on layered networks.

Part II

The non-atomic case

Chapter 4

Selfish routing

Having extensively considered the atomic case of selfish routing[1], we move on to consider the case where we have an infinite number of players (the non-atomic case). The basic idea is the same as in multicommodity networks of Chapter 3: we have a network with some source-sink pairs and an arbitrarily large number of players that wish to route their traffic through the network, in a way that minimizes their personal delay. Although, at a first glance, the study of arbitrarily large populations seems demanding, it allows us to use methods from continuous mathematics, thus increasing our analytical tractability. In this chapter we are mostly going to focus on the inefficiency of the equilibria induced on such games. In order to quantify this inefficiency we use once again the PoA and Braess ratio, a new measure introduced for this model; we shall also see the importance of the two motivating examples (paradoxes) of Chapter 1. Most of the results presented in this thesis come from the work of Tim Roughgarden and Éva Tardos. However the model considered has a long history in the transportation science literature and has also been widely studied by the computer networking community.

4.1 The model

Back in Chapter 1 we defined a game as a triple consisting of the set of players, the set of strategies for each player and the set of payoff (or cost) functions for each player. In the atomic case that we have studied so far the set of players was a finite set. From now on we are going to consider the case of infinitely many players, in order to model arbitrarily large populations. We could follow the definitions in Chapter 1 and try to define a payoff per

[1]The term "selfish routing" was actually used to denote the non-atomic case. The atomic-case games are more often called "congestion games" or "resource allocation games"

strategy profile. However this would make most definitions pretty hard to understand and difficult to handle. Hence, we trust our intuition and make the following remark: we only care what fraction of population picks each strategy. So we will try to model only this. The question now is, what are the ingredients that we need to describe the game, when we only focus on the former fraction? We list them right below:

1. a finite number of player types
2. the population sizes, one for each player type
3. some finite strategy sets, one fore each player type
4. for each fraction of population using each strategy a cost per strategy

Now let us explain what we mean with the above (how they are interpreted in a network game) and why they suffice to describe our game. To understand the following the reader must keep in mind that we now approach the game from the network's point of view, as opposed to the player's point of view that we have used so far.

Ingredient 1: Remember the multicommodity games introduced in Chapter 3, where each player had an action set consisting of paths between a unique origin-destination pair of nodes (s_i, t_i), called commodities. In selfish routing each player wants to route some traffic from some origin nodes to some destination nodes, so the problem is essentially the same. The difference is that now we have infinitely many players. The idea is to separate the infinitely many players in a finite number of player types, based on the commodity of each player's action set. Namely if our network consists of n commodities $(s_1, t_1), \dots (s_k, t_k)$ then we have k player types, with type i wishing to route some traffic from s_i to t_i.

Ingredient 2: In weighted multicommodity games, we allowed each player to have a weight, which represented the amount of traffic he wanted to route on the network from s_i to t_i. Here, instead of assigning each player a separate weight, we focus only on the total amount of traffic routed through the network for each player type (i.e. for each commodity).

Ingredients 3 & 4: Finally the finite strategy sets that correspond to each player type can be easily interpreted in the various (s_i, t_i) paths. We then define the cost of each edge and use it to derive a proper cost function for the fraction of population picking each strategy ((s_i, t_i) path).

Before moving on to the formal definitions let us summarize some more differences between this model and the KP-model introduced in Chapter 2. Most of those differences will become apparent as we move through this Chapter.

KP-model	selfish routing model
atomic	non-atomic
we consider both NE and PNE	we only consider PNE
social cost defined as max cost	social cost defined as average cost
only parallel link networks	general multicommodity networks
linear cost functions	nonlinear cost functions as well

Table 4.1: Main differences of KP-model and selfish routing

Definitions

So a selfish routing game formally consists of a multicommodity network flow described by a directed graph $G = (V, E)$, with vertex set V and edge set E and a set $(s_1, t_1), \ldots (s_k, t_k)$ of source-sink vertex pairs, the commodities. Parallel edges are allowed and a vertex can participate in more than one commodities.

We use $\mathcal{P}_i \neq 0$ to denote the set of simple (s_i, t_i)-paths and $\mathcal{P}$ to denote their union: $\mathcal{P} = \bigcup_i \mathcal{P}_i$. Let r be a nonnegative vector indexed by the commodities, that denotes the *traffic rates*, i.e. the total amount of traffic to be routed between one source-sink pair. A *flow* f in G is a nonnegative vector indexed by $\mathcal{P}$. Then f_P denotes the amount of traffic (fraction of total traffic between s_i and t_i) that chooses path P to navigate from s_i to t_i. Obviously for any feasible flow f the following must hold:

$$\sum_{P \in \mathcal{P}_i} f_P = r_i \tag{4.1}$$

A flow f induces a *flow on edges* $\{f_e\}_{e \in E}$, where $f_e = \sum_{P \in \mathcal{P}: e \in P} f_P$ and it denotes the total amount of flow that uses edge e.

To model the negative consequences of congestion we give each edge e of G a nonnegative, continuous, nondecreasing *cost function* $c_e(f_e)$, which denotes the travel time (cost) incurred by all traffic traversing edge e, given the flow f_e. As we shall see, the above properties of the cost function are very important to prove existence and uniqueness of equilibrium. Finally we define the overall cost of a path P to be $c_P(f) = \sum_{e \in P} c_e(f_e)$.

Now we can formally define a selfish routing game as a triple (G, r, c), where G is a multicommodity network, r is a vector of traffic rates and c is a set of cost functions. We call (G, r, c) an *instance*.

Equilibria

It is now obvious that in the non-atomic model we replace the pure (or mixed) strategy profile, which sketches the choices of each player, with the flow f. Hence f now denotes the selfish outcome of such a network and we will try to define the notion of an equilibrium using f. Keep in mind that an equilibrium is a collection of choices (one for each player), where no player has an incentive to change his choice: in our model this implies that for every type of player (i.e. for all commodities (s_i, t_i)) all corresponding players pick a strategy (i.e. an (s_i, t_i)-path P) that minimizes the incurred cost c_P. This leads to the following definition of Wardrop equilibrium (first formulated by Wardrop for road traffic), a notion of equilibrium that is equivalent to the one of *Nash flows*.

Definition 4.1.1. Let f be a feasible flow for the instance (G, r, c). The flow f is a *Wardrop equilibrium* if, for every commodity $i \in \{1, \ldots, k\}$ and every pair of paths $P, \widetilde{P} \in \mathcal{P}_i$ of (s_i, t_i) paths with $f_P > 0$,

$$c_P(f) \leq c_{\widetilde{P}}(f)$$

In other words a flow f that is a Wardrop equilibrium (WE) is a flow that routes all traffic on the paths of minimum cost, among all other (s_i, t_i)-paths. It is straightforward that all paths of a given commodity used by a Wardrop equilibrium must have equal costs, in order to avoid defections. Hence the corollary:

Corollary 4.1.2. *All paths $P \in \mathcal{P}_i$ of a given commodity i used by a flow f that is a WE, must have equal costs. We shall denote this by $c_{\mathcal{P}_i}(f)$.*

Remark 4.1.3. The above definition implies that each player deterministically picks one path to route his traffic. The case of mixed strategy profile for this model is not considered, since we can prove all kinds of interesting results for a pure profile, which is what we are after all really interested in.

Remark 4.1.4. In Definition 4.1.1 we are implicitly assuming that each player controls a negligible portion of the overall traffic and thus his choice has no effect on the network congestion. That is also the meaning of defining flows: a flow consists of an arbitrarily large number of negligible player traffics.

The two fundamental questions of existence and uniqueness of equilibrium have been resolved from the 50's by Beckmann, McGuire and Winsten, who have formulated the following proposition:

Proposition 4.1.5. *Let (G, r, c) be an instance of a selfish routing game:*

1. *The instance (G, r, c) admits at least one WE.*

2. *If f and $\widetilde{f}$ are WE for (G, r, c), then $c_e(f_e) = c_e(\widetilde{f}_e)$ for every edge e.*

The proof of the above proposition is remarkably simple. The key idea is to show that WE of the above instance, are exactly the flows that minimize the following potential function:

$$\Phi(f) = \sum_{e \in E} \int_0^{f_e} c_e(x)dx \tag{4.2}$$

over all feasible flows. Since cost functions are continuous and the space of all flows is compact, Weierstrass's Theorem implies the existence of a minimum and thus of a WE. Now since c_e are taken to be nondecreasing and since c_e is essentially the derivative of Φ, Φ must be convex. But for a continuous, convex function every local minimum is also global, which implies that the values of the cost functions c_e at all minima (i.e. at all WE) are the same.[2]

Before moving on to defining the PoA, let us see a very useful Lemma, that we shall use later to upper bound the PoA.

Lemma 4.1.6. *A flow f feasible for (G, r, c) is a WE iff*

$$\sum_{e \in E} c_e(f_e) f_e \leq \sum_{e \in E} c_e(f_e) f_e^* \tag{4.3}$$

for all feasible flows f^*.

Proof. We shall first prove that the following inequality:

$$\sum_{P \in \mathcal{P}} c_P(f) f_P \leq \sum_{P \in \mathcal{P}} c_P(f) f_P^*$$

where f^* is a feasible flow for (G, r, c) holds iff f is a WE. The proof is simple:

[2] This does not imply that all flows that are WE are identical, but only that they induce identical edge costs. Nonetheless this suffices, as we shall see.

$$
\begin{aligned}
\sum_{P\in\mathcal{P}} c_P(f) f_P &= \sum_{P\in\mathcal{P}:f_P>0} c_P(f) f_P \\
&= \sum_{i=1}^{k} \sum_{P\in\mathcal{P}_i:f_P>0} c_P(f) f_P && \text{(split paths into commodities)} \\
&= \sum_{i=1}^{k} c_{\mathcal{P}_i}(f) \sum_{P\in\mathcal{P}_i:f_P>0} f_P && \text{(see Corollary 4.1.2)} \\
&= \sum_{i=1}^{k} c_{\mathcal{P}_i}(f) r_i && \text{(see (4.1))} \\
&= \sum_{i=1}^{k} c_{\mathcal{P}_i}(f) \sum_{P\in\mathcal{P}_i} f_P^* && \text{(see (4.1))} \\
&= \sum_{i=1}^{k} \sum_{P\in\mathcal{P}_i} c_{\mathcal{P}_i}(f) f_P^* \\
&\le \sum_{i=1}^{k} \sum_{P\in\mathcal{P}_i} c_P(f) f_P^* && \text{(from Definition 4.1.1)} \\
&= \sum_{P\in\mathcal{P}} c_P(f) f_P^*
\end{aligned}
$$

Now notice that:

$$
\begin{aligned}
\sum_{P\in\mathcal{P}} c_P(f) f_P &= \sum_{P\in\mathcal{P}} \sum_{e\in P} c_e(f_e) f_P && \text{(definition of } c_P(f)\text{)} \\
&= \sum_{e\in P} \sum_{P\in\mathcal{P}:e\in P} c_e(f_e) f_P && \text{(rearrangement of summations)} \\
&= \sum_{e\in P} c_e(f_e) \sum_{P\in\mathcal{P}:e\in P} f_P && \text{(definition of } f_e\text{)} \\
&= \sum_{e\in P} c_e(f_e) f_e
\end{aligned}
$$

Likewise we can prove that $\sum_{P\in\mathcal{P}} c_P(f) f_P^* = \sum_{e\in P} c_e(f_e) f_e^*$. Combining these with the above inequality yields the result. □

The Price of Anarchy

We conclude the discussion about the model with the definition of the PoA. To define this we need an objective function that represents the efficiency loss of the system. Unlike KP-model, where we considered a cost function that corresponded to the maximum latency among all links, here we shall focus on the average cost induced by a flow f, henceforth denoted by $C(f)$:

$$C(f) = \sum_{P \in \mathcal{P}} c_P(f) f_P = \sum_{e \in E} c_e(f_e) f_e \tag{4.4}$$

The first equality in 4.4 is a definition and the second one follows from the same reversal of sums as in Lemma 4.1.2. The fact that we do not consider mixed strategy profiles, simplifies the expression for the social cost, in the sense that we do not need to compute the estimation of the cost function: **SC** here is merely $C(f)$. Also we define the optimal flow f^* for (G, r, c) to be the flow that minimizes $C(f)$ among all feasible flows f. The corresponding value $C(f^*)$ is **OPT**. Once again Weierstrass's theorem implies the existence of an optimal flow f^*.

Remark 4.1.7. Notice here that for an optimal flow f^* the following must hold for all feasible flows f: $C(f^*) \leq C(f) \Leftrightarrow \sum_{e \in E} c_e(f_e^*) f_e^* \leq \sum_{e \in E} c_e(f_e) f_e$. Comparing this relationship with the one of Lemma 4.1.2 for WE, we see that they look quite the same. This similarity implies that a WE cannot be much worse than the optimal flow, or equivalently that the PoA can be well bounded. We could end up at the same result by noticing that a WE is a minimizer of a potential function and remembering the discussion in Chapter 3 about potential functions.

Definition 4.1.8. The Price of Anarchy $\rho(G, r, c)$ of an instance (G, r, c) is:

$$\rho(G, r, c) = \frac{C(f)}{C(f^*)}$$

where f is a WE and f^* is an optimal flow for (G, r, c). The PoA $\rho(\mathcal{I})$ of a non-empty set $\mathcal{I}$ of instances is $\sup_{(G,r,c) \in \mathcal{I}} \rho(G, r, c)$.

We immediately notice a difference with Definition 2.3.5: there is no sup in front of the fraction. The reason for that is Proposition 4.1.5, which implies that all WE have equal cost. Thus the PoA is the same, no matter which WE we consider. In the special case of a zero cost flow, where all WE have zero cost, we define the PoA to be unit (in order to have $\rho(G, r, c) \geq 1$ for all instances (G, r, c)).

4.2 Bounding the Price of Anarchy

In this section we shall provide a lower bound for the PoA of selfish routing and a corresponding upper bound, for a variety of cost functions. In the following analysis Pigou's example plays a crucial role, so we repeat the basic results right below and we present a nonlinear variant as well.

We have a single-commodity network where the source vertex s and the sink vertex t are connected through two disjoint edges, one with cost function $c(x) = 1$ and the other with $c(x) = x$. Say we want to route one unit of traffic. As we discussed in Chapter 1, there is only one reasonable choice for rational players and that is to route all traffic on the lower edge. We can easily verify that this flow is a WE. We also discussed that if somehow we convinced half the players to route their traffic on the upper edge, then all the players would be better off, in the sense that the delay for half the players would be $1/2$ instead of 1. In terms of C, we can say that in the first case we have $C(f) = 1 \cdot 0 + 1 \cdot 1 = 1$ and in the second case $C(f^*) = 1 \cdot 1/2 + 1/2 \cdot 1/2 = 3/4$. It is trivial to show that the flow f^* is an optimal flow for this network[3]. and that, as a result, the PoA is $4/3$.

We can show that the PoA of the Pigou example can be arbitrarily large if we allow nonlinear cost functions. Indeed set the cost of the lower link to $c(x) = x^p$, which is highly nonlinear if p is sufficiently large. Once again, for a unit traffic, all users choose the lower link, inducing a WE of total cost 1. If on the other hand, the optimal flow f^* is to route a small ϵ fraction of the total traffic on the upper link; then the cost is $C(f^*) = \epsilon + (1-\epsilon)^{p+1}$, which (for $\epsilon \to 0$) approaches 0 as p tends to infinity. This means that the PoA for this instance tends to infinity with p.

The above observations generate a series of other questions: can the PoA be arbitrarily large if the cost functions are "not too nonlinear"? Is the PoA in general larger in bigger, more complicated networks or in networks with more commodities? The answer to all these questions is negative. In fact Pigou's example provides a universal "bad case" for selfish routing, in the sense that it bounds the PoA for a corresponding set of cost functions. The proof of this claim is the objective of this section.

The Pigou Bound

The above discussion implies that the PoA of a selfish routing instance depends, at the very least, on the set of allowable cost functions $\mathcal{C}$. We therefore aim for a (lower) bound that is parametrized by $\mathcal{C}$. Common sets of cost functions are the constant cost functions, linear functions, polynomials and

[3] set x the flow of the upper link, $1-x$ the flow for the lower link, write down $C(x)$ and compute x such that $C'(x) = 0$

queueing delay functions.

The idea here is, that for every set $\mathcal{C}$, Pigou-like examples should provide a natural lower bound for the PoA of every set $\mathcal{I}$ of instances (G, r, c), where G is of a special form and $c \in \mathcal{C}$. So we are going to define a bound $\alpha(\mathcal{C})$, which we shall call the Pigou bound, in order to show that $\rho(\mathcal{I}) \geq \alpha(\mathcal{C})$ for this appropriate set $\mathcal{I}$.

The only assumption we need to make about $\mathcal{C}$ for now is that it contains all constant functions. So choose a cost function $c_2 \in \mathcal{C}$ at random and a traffic rate r. Then assume that c_1 is a function everywhere equal to $c_2(r)$. Because of the former assumption we have $c_1 \in \mathcal{C}$. Now consider the usual single-commodity, two-node, two links network of the Pigou example, where we assign the upper and lower edge cost functions c_1 and c_2 respectively and where the traffic rate is r. As usual the lower edge is never worse off than the upper edge and thus routing all traffic on this edge yields a WE of cost $c_2(r)r$. The optimum cost can be formalized as follows: $\min\limits_{0 \leq x \leq r} (xc_2(x) + (r-x)c_2(r))$ and the PoA is then:

$$\max_{0 \leq x \leq r} \frac{rc_2(r)}{xc_2(x) + (r-x)c_2(r)}$$

With a closer look at the denominator we can see that the fraction reaches its max value for $x \leq r$. So the PoA can be written:

$$\max_{x, r \geq 0} \frac{rc_2(r)}{xc_2(x) + (r-x)c_2(r)}$$

Consider now the set $\mathcal{I}$ to be the set of single-commodity instances with a two-node, two link network and cost functions in $\mathcal{C}$. Obviously the above Pigou-like network belongs to $\mathcal{I}$. By the definition of $\rho(\mathcal{I})$ as a supremum over all instances in $\mathcal{I}$, it follows that the above PoA is also a lower bound for $\rho(\mathcal{I})$. To get a better lower bound, we choose the cost functions in the worst possible way, i.e. in a way that maximizes the above PoA, and we get the Pigou bound:

Definition 4.2.1. Let $\mathcal{C}$ be a nonempty set of cost functions. The Pigou bound $\alpha(\mathcal{C})$ is

$$\alpha(\mathcal{C}) = \sup_{c \in \mathcal{C}} \sup_{x, r \geq 0} \frac{rc_2(r)}{xc_2(x) + (r-x)c_2(r)} \tag{4.5}$$

We assume that 0/0=1.

From the discussion above and the definition of Pigou bound the next Proposition follows immediately.

Proposition 4.2.2. *Let $\mathcal{C}$ be a set of cost functions that includes all constant functions, and let $\mathcal{I}$ denote the set of single-commodity instances with a two-node, two link network and cost functions in $\mathcal{C}$. Then*

$$\rho(\mathcal{I}) \geq \alpha(\mathcal{C})$$

It must be by now quite clear that finding an arbitrary lower bound for the PoA of a set of instances $\mathcal{I}$ is not really hard: actually the PoA of any instance $(G, r, c) \in \mathcal{I}$ lower bounds $\rho(\mathcal{I})$. The question is whether the lower bound is tight enough. To prove that for the Pigou bound we must provide a corresponding upper bound. Before that we summarize some interesting results for some very useful cases of cost functions.

$\mathcal{C}$	$\alpha(\mathcal{C})$	References
$\{ax + b : a, b \geq 0\}^4$	4/3	[RT02], [Rou02]
concave cost functions	4/3	[CSM04]
polynomials[5] with nonnegative coefficients and degree at most p	$[1 - p \cdot (p + 1^{-(p+1)/p})]^{-1}$	[Rou02]
(nondecreasing) polynomials with arbitrary coefficients and degree at most p	?	-
set of M/M/1 delay functions with queue service rate $u \geq u_{min}$ and traffic rate $r \leq R_{max} < u_{min}$	$\frac{1}{2}\left(1 + \sqrt{\frac{u_{min}}{u_{min} - R_{max}}}\right)$	[Rou02]

Table 4.2: The Pigou bound for some important sets of cost functions

Remark 4.2.3. Although Proposition 4.1.2 assumes that $\mathcal{C}$ contains all constant functions it can be proved for more general set of constant functions as well.

We shall now prove an upper bound on the PoA. We first state the following Proposition which follows immediately from Definition 4.2.1.

Lemma 4.2.4. *Let $\mathcal{C}$ be a set of cost functions and $\alpha(\mathcal{C})$ the corresponding Pigou bound. For $c \in \mathcal{C}$ and $x, r \geq 0$,*

$$x \cdot c(x) \geq \frac{r \cdot c(r)}{\alpha(\mathcal{C})} + (x - r)c(r)$$

We now use the above Lemma and inequality (4.3) to prove the optimality of the Pigou bound. We should note here that the following theorem has undergone several iterations and modifications over the years. It was first proved for the special case of linear cost functions by Roughgarden and Tardos [RT02] and it was extended step by step ([Rou02]). Some of the proofs in the bibliography are fairly complex. The key idea is to use inequality (4.3) in order to simplify the proof. Here we follow the proof given by Correa, Schulz and Stier Moses in [CSM04].

Theorem 4.2.5. *Let $\mathcal{C}$ be a set of cost functions and $\alpha(\mathcal{C})$ the corresponding Pigou bound. If (G, r, c) is an instance with $c \in \mathcal{C}$, then*

$$\rho(G, r, c) \leq \alpha(\mathcal{C})$$

Proof. Let f^* and f be an optimal flow and a WE respectively, for an instance (G, r, c) with $c \in \mathcal{C}$, then

$$\begin{aligned} C(f^*) &= \sum_{e \in E} c_e(f_e^*) f_e^* \\ &\geq \tfrac{1}{\alpha(\mathcal{C})} \sum_{e \in E} c_e(f_e) f_e + \sum_{e \in E} (f_e^* - f_e) c_e(f_e) && \text{from Lemma 4.2.4} \\ &\geq \tfrac{C(f)}{\alpha(\mathcal{C})} && \text{(4.3) implies } \sum_{e \in E} (f_e^* - f_e) c_e(f_e) \geq 0 \end{aligned}$$

□

The above theorem implies that the lower bounds of Table 4.2 are the best possible, namely the PoA of each case is exactly $\alpha(\mathcal{C})$. Another interesting remark is that the worst-possible PoA for a set of instances occurs in the very simple Pigou-like networks. Hence we assume that the complexity of the allowable network topologies has nothing to do with the inefficiency of the resulting equilibria. In fact the PoA is independent of the number of commodities as well and depends only on the set of allowable cost functions.

4.3 Bounding Braess 's Paradox

In this section we focus on Braess's paradox (see Example 1.3.3). We have already discussed how startling and unintuitive this result is. In this section we will try to quantify Braess's paradox, by defining the Braess ratio, and we shall study the problem in more general networks. Before studying the problem at large, it is useful to verify that for the original instance, the flows proposed in Example 1.3.3, are indeed WE and that their corresponding costs

$C(f)$ are indeed 3/2 and 2.

Example 1.3.3 shows that adding to a network a new edge can increase the incurred cost, no matter how "fast" this edge is. Equivalently removing an edge from an existing network with linear cost functions can decrease its cost by a factor of 4/3. The question is, can the cost be decreased by a larger factor

- in larger networks,
- or with multicommodity topologies,
- or with arbitrary cost functions,
- or when multiple edge removal is allowed?

4.3.1 Enlarging the Paradox

In this section we shall show that the severity of Braess's Paradox depends on all of the above factors. More precisely we shall show that it can be arbitrarily severe in large single-commodity network, only if nonlinear cost functions and multiple edge removal are allowed.
We measure the severity of Braess's Paradox with the Braess ratio defined below. The Braess ratio indicates the maximum factor by which the cost of a WE can decrease from a network to one of its subnetworks.

Definition 4.3.1. The Braess ratio $\beta(G, r, c)$ of a single-commodity instance (G, r, c) is

$$\beta(G, r, c) = \max_{H \subseteq G} \frac{C(f)}{C(f^H)} \tag{4.6}$$

where H ranges over subnetworks of G that contain an $s-t$ path, and f and f^H denote WE for (G, r, c) and (H, r, c) respectively.

Remark 4.3.2. For now we limit our discussion on single-commodity networks. Later we shall discuss multiple ways to extend the above definition to multicommodity networks.

Remark 4.3.3. Notice that Definition 4.3.1 allows multiple edge removal, since we examine the cost for all subnetworks of G. So we could say that multiple edge removal is always possible. The question is whether it increases the Braess ratio or not: we shall soon see it does.

It is also easy to verify that the Braess ratio in Example 1.3.3 is 4/3. We claim that no larger Braess ratio is possible in single-commodity networks with linear cost functions. This fact is a consequence of the following (much stronger) connection between the PoA and the Braess ratio.

Proposition 4.3.4. *If (G, r, c) is a single-commodity instance, then*

$$\beta(G, r, c) \leq \rho(G, r, c)$$

Proof. For every subgraph H of G, a WE f^H is a feasible flow for (G, r, c) as well. Hence $C(f^H) \geq C(f^*)$, where by f^* we denote the optimum flow for (G, r, c). Also we have $\rho(G, r, c) \geq \frac{C(f)}{C(f^*)}$ for every WE f of (G, r, c). Combining these inequalities we have $C(f^H) \geq \frac{C(f)}{\rho(G,r,c)}$, which yields the desired result. □

Now Proposition 4.3.4 implies that any single-commodity network with linear cost functions, has Braess ratio that is at most 4/3. Since we have a corresponding lower bound, 4/3 is tight. We shall give a construction which indicates that this upper bound is also tight (up to constant factors) for other cost functions as well.

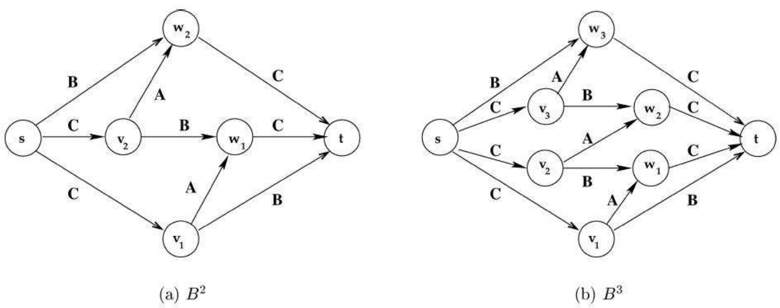
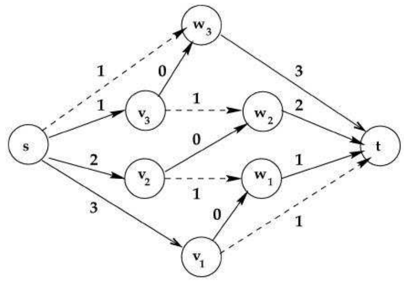


Figure 4.1: The Braess Graphs for k=2 and k=3

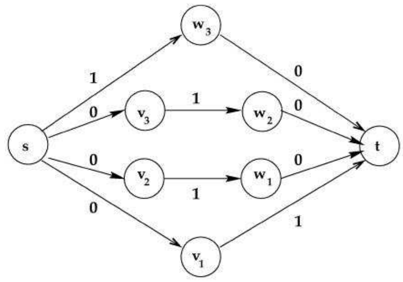
Theorem 4.3.5. *For every $n \geq 2$, there is a single-commodity instance (G, r, c) with n vertices and*

$$\beta(G, r, c) \geq \left\lfloor \frac{n}{2} \right\rfloor$$

Proof. We shall prove the theorem for n even. The case of odd n reduces to this case, by simply adding an isolated node in the network. Wlog we can also assume that n is at least 4. So, write $n = 2k + 2$ for $k \geq 1$.
We define the kth *Braess graph* B^k as follows:



(a) Wardrop equilibrium in $(B^3, 3, c^3)$



(b) Wardrop equilibrium in the optimal subgraph

Figure 4.2: The WE for G and H

- we have $2k+2$ nodes, $V^k = \{s, v_1, \ldots v_k, w_1, \ldots w_k, t\}$
- and the edge set E^k is the union of the sets
  - $\{(s, v_i), (v_i, w_i), (w_i, t) : 1 \leq i \leq k\}$
  - $\{(v_i, w_{i-1}) : 2 \leq i \leq k\}$ and
  - $\{(v_1, t) \cup (s, w_k)\}$

The Braess graphs for $k = 2, 3$ are depicted in Figure 4.1 Note that B^1 is the graph in the original Braess's Paradox (Example 1.3.3). We now separate the edges in types and define the costs for each such type:

- edges of the form (v_i, w_i) are type A edges and have cost $c_e^k(x) = 0$
- edges of the form $(v_i, w_{i-1}), (s, w_k), (v_1, t))$ are type B edges and have cost $c_e^k(x) = 1$
- for each $i \in \{1, \ldots, k\}$ edges of the form $(s, v_{k-i+1}), (w_i, t)$ are type C edges and have a continuous nondecreasing cost function $c_e^k(x)$ with $c_e^k(k/(k+1)) = 0$ and $c_e^k(1) = i$.

Furthermore let us denote by

- P_i the path $s \to v_i \to w_i \to t$ for $i \in \{1, \ldots, k\}$
- Q_1 the path $s \to v_1 \to t$
- Q_i the path $s \to v_i \to w_{i-1} \to t$ for $i \in \{1, \ldots, k\}$
- Q_{k+1} the path $s \to w_k \to t$

Now consider the instance (B^k, k, c^k) where the cost functions are defined as above. Note that routing one unit of flow on each of $P_1, \dots, P_k$ yields a WE in which all traffic incurs cost $k+1$ (Figure 4.2(a)).
On the other hand, if H is the subgraph obtained from B^k by deleting the k type A edges, then routing $k/(k+1)$ units of flow on each of $Q_1, \dots, Q_{k+1}$ yields a WE f^H for (H, k, c^k), in which all traffic incurs only one unit of cost (Figure 4.2(b)).Thus

$$\beta(G, r, c) \geq C(f)/C(f^H) = k + 1 = n/2$$

□

Remark 4.3.6. Although it is not so obvious we can use similar arguments as in Theorem 4.3.5 to adapt to scenarios where arbitrary cost functions are not allowed. Then we could show that this lower bound for the Braess ratio matches the upper bound that follows from Proposition 4.3.4 and Theorem 4.2.5. For more information see [Rou01].

Proposition 4.3.4 and Theorem 4.3.5 also imply that, in order to exhibit a family of instances with arbitrarily large Braess ratio, we need to have cost functions drawn from a sufficiently rich set (e.g. polynomials with unbounded degree). However this alone is not enough. In order to achieve a large Braess ratio, we also need larger, more complicated networks[6]. The following theorem implies exactly that.

Theorem 4.3.7. *If (G, r, c) is a single-commodity instance with n vertices, then*

$$\beta(G, r, c) \leq \left\lfloor \frac{n}{2} \right\rfloor$$

This Theorem shows that, among single-commodity networks, the Braess ratio is maximized by the networks in the proof of Theorem 4.3.5. In order to prove this Theorem, we need another, stronger result, which we present here, without its (quite technical) proof. For more information see [LRT04].

Theorem 4.3.8. *Let (G, r, c) be a single-commodity instance, H a subgraph of G, and f, $\tilde{f}$ WE for (G, r, c) and (H, r, c) respectively. Let S denote the edges in G but not H (namely, the edges we remove). If every undirected matching of $S \backslash \{s, t\}$, where $\{s, t\}$ are the source, and destination nodes, contains at most k edges, then*

$$C(f) \leq (k + 1)C(\tilde{f})$$

[6] notice the difference with the PoA, where we only cared about the cost functions

Theorem 4.3.8 implies Theorem 4.3.7 as well as an upper bound on the severity of Braess's Paradox, parametrized by the number of edges removed. Let us first prove Theorem 4.3.7:

Proof. Since there are only $n-2$ vertices of G, apart from s and t, every matching of G has at most $\lfloor (n-2)/2 \rfloor = \lfloor n/2 \rfloor - 1$ edges. So does every matching of $G \backslash H$. The result follows from Theorem 4.3.8 for $k = \lfloor n/2 \rfloor - 1$. □

Corollary 4.3.9. *Removing k edges from a single-commodity network decreases the cost of a WE by at most a factor of $k+1$*

Corollary 4.3.9 implies that the only way to achieve arbitrarily large Braess ratios is to allow an unlimited number of edge removals. In fact the bound of the Corollary is matched by the construction of Theorem 4.3.5, so it is tight.

Multicommodity networks

We now extend the notion of Braess ratio for multicommodity networks. We could use here Definition 4.3.1 as well, but although Proposition 4.3.4 still holds and we have a tight bound for the case of linear latency functions, no corresponding bounds are possible for networks with arbitrary cost functions. In fact, even in two-commodity, three-node networks, removing a single edge can decrease the cost of a WE by an arbitrarily large factor.
Hence we define the Braess ratio for multicommodity networks as follows.

Definition 4.3.10. The Braess ratio $\beta(G, r, c)$ of a multicommodity instance (G, r, c) is

$$\beta(G, r, c) = \max_{H \subseteq G} \min_{i=1}^{k} \frac{d_i(G, r, c)}{d_i(H, r, c)} \tag{4.7}$$

where $d_i(G, r, c)$ denotes the common cost incurred by all traffic of commodity i in a WE for (G, r, c) and H ranges over subnetworks of G that contain an $s_i - t_i$ path.

Thus the Braess ratio of a multicommodity network instance is large only if removing some set of edges decreases the cost incurred by the traffic of **every** commodity.
It has been shown in [LRTW05] that the upper bound of Theorem 4.3.7 does not carry on to multicommodity networks. In fact it can grow exponentially with the networks size, even in two-commodity networks:

Theorem 4.3.11. *There is a family of two-commodity networks $\{(G^n, r^n, c^n)\}_{n=1}^{\infty}$ s. t. G_n has $O(n)$ vertices and edges and $\beta(G^n, r^n, c^n) = 2^{\Omega(n)}$ as $n \to \infty$.*

On the other hand we have that the Braess ratio is at most exponential in the networks size, due to the following theorem

Theorem 4.3.12. *There is a constant $c > 0$ such that for every $k, n \geq 1$ and every instance (G, r, c) with k commodities and n vertices, $\beta(G, r, c) \leq 2^{ckn}$.*

Theorems 4.3.11 and 4.3.12 together do not establish a tight bound, because the upper bound seems to depend on the number of commodities k. Whether it really depends or not, is still an open question.

4.3.2 Detecting Braess's Paradox is hard

Braess's paradox suggests a natural algorithmic question: given a network, is it suffering from the paradox? If so, which edges should be removed to recover the best-possible WE?
This question turns out to be extremely difficult ($\mathcal{NP}$-hard) to answer, even for single-commodity networks with linear cost functions. In order to prove that, let us formulate the problem as an optimization problem:

Definition 4.3.13. LINEAR NETWORK DESIGN: Given a single-commodity instance with linear cost functions, find a subnetwork H that minimizes the cost of a WE of the instance (H, r, c), for H ranging over all subnetworks of G ($H \subseteq G$).

A trivial algorithm to solve LINEAR NETWORK DESIGN, would be to enumerate all subgraphs of G, compute the WE of each one and pick the best solution. Although computing a WE is easy, the subgraphs of G can be exponentially many and the running time of the algorithm would be prohibitive.
So, instead of looking for an algorithm that finds the exact solution, we shall seek for a γ-approximation algorithm, i.e. an algorithm that returns a solution worst than the optimal, but less than γ times as costly as the optimal: $OPT \leq x \leq \gamma \cdot OPT$. We want γ as close to 1 as possible.
Note that even the *trivial* algorithm that returns the entire networks as a solution can be viewed as an approximation algorithm. In fact, since the Braess ratio in single-commodity networks with linear cost functions is at most 4/3 (Proposition 4.3.4) the trivial algorithm is a 4/3-approximation algorithm for LINEAR NETWORK DESIGN.
Our goal is to design more a clever algorithm, with a better approximation ratio. However, none exist, unless $\mathcal{P} = \mathcal{NP}$.

Theorem 4.3.14. *For every $\epsilon > 0$, there is no $(4/3 - \epsilon)$-approximation algorithm for LINEAR NETWORK DESIGN (assuming $\mathcal{P} \neq \mathcal{NP}$).*

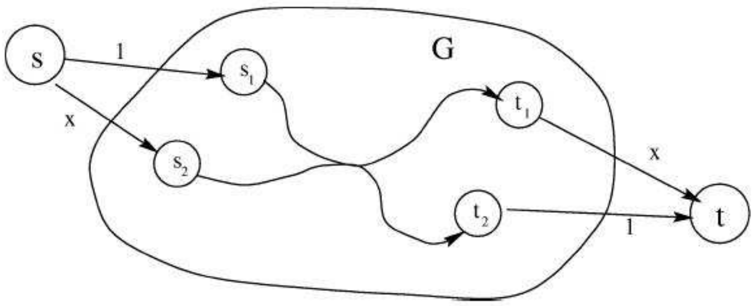


Figure 4.3: The reduction from 2DDP to LINEAR NETWORK DESIGN

Proof. We present a polynomial-time "gap" reduction from the $\mathcal{NP}$-complete problem 2 DIRECTED DISJOINT PATHS (2DDP): given a directed graph $G = (V, E)$ and distinct vertices $s_1, s_2, t_1, t_2 \in V$, are there $s_i - t_i$ paths P_i for $i = 1, 2$, such that P_1 and P_2 are vertex disjoint? We shall show how a $(4/3 - \epsilon)$-approximation algorithm can be used to distinguish between "yes" and "no" instances of 2DDP.
First of all, given an instance $\mathcal{I}$ of 2DDP with $G = (V, E)$ we shall construct an instance of selfish routing in polynomial time. In order to do that we augment the vertex set V by an additional source s and sink t and we include in the edge set E the directed edges $(s, s_1), (s, s_2), (t_1, t), (t_2, t)$ (see Figure 4.3). We define the cost functions on the edges of E to be $c(x) = 0$, on the edges $(s, s_1), (t_2, t)$ to be $c(x) = 1$ and on the edges $(s, s_2), (t_1, t)$ to be $c(x) = x$. The new graph is G' and we have thus constructed the instance $(G', 1, c)$ of selfish routing in polynomial time.
We want to show that the following statements are true:

- if $\mathcal{I}$ is a "yes" instance of 2DDP, then G' admits a subnetwork H such that the WE for $(H, 1, c)$ has cost $3/2$: this means that the optimum solution has cost at most $3/2$. Hence the solution returned by the approximation algorithm, which is at most $(4/3-\epsilon)$ times the optimum solution, is at most $(4/3 - \epsilon) \cdot 3/2 < 2$.

- if $\mathcal{I}$ is a "no" instance of 2DDP, then for every subnetwork H of G' the WE for $(H, 1, c)$ has cost at least 2[7]. Since this holds for every subnetwork H, it must hold for the optimum one as well, so the optimum solution is at least 2, hence the solution returned by the approximation algorithm has cost at least 2 as well.

If we can prove the above properties of our construction, we can use the approximation algorithm for LINEAR NETWORK DESIGN to solve 2DDP

[7]notice the resemblance to the costs of the original Braess's paradox

in polynomial time as follows: we run our approximation algorithm and if the returned solution is < 2 then the corresponding instance of 2DDP is a "yes" instance, otherwise it is a "no" instance.
So we are left with the task of proving that the above statements are true.
Consider the case of a "yes" 2DDP instance: there exist vertex-disjoint $s_1 - t_1$ and $s_2 - t_2$ paths P_1 and P_2 respectively. Obtain H by deleting all edges in G not contained in some P_i. This operation does not ruin P_1, P_2 and assures that they are the **only** two $s_i - t_i$ paths in H. It is then easy to verify that H admits a WE of cost $3/2$, by routing half the traffic on each path P_i.
Now consider the case of a "no" 2DDP instance and take an arbitrary subgraph H of G', possessing a $s - t$ path, wlog. We have to study two cases:

- if the corresponding 2DDP instance has $s_i - t_i$ paths for both $i = 1$ and 2, then it must be the case that these paths share a common vertex, which implies that if H has two $s - t$ paths containing $s_1 - t_1$ and $s_2 - t_2$ paths, then it also has an $s - t$ path containing a $s_2 - t_1$ path. Since we have only one unit of flow, this path is always cheaper, (remember the original Braess Paradox) and the WE is for all the flow to be routed on this path, incurring a cost of 2. If H does not have two $s - t$ paths, but just one, then the WE is to route all traffic on this path and has a cost of 2 as well.

- if the corresponding 2DDP problem has a $s_i - t_i$ path for precisely one $i \in \{1, 2\}$, then the $s - t$ path of H may contain a $s_i - t_i$ path for just one i and/or a $s_1 - t_2$ path. In either case the WE is to route the whole traffic on one path, incurring a cost of 2.

□

The above result implies that there is no way to distinguish between a selfish routing instance that does not suffer from the Braess paradox and one that has a Braess ratio of 4/3, i.e. the worst possible Braess ratio for single-commodity instances with linear cost functions. Equivalently we can say that **detecting Braess's paradox is $\mathcal{NP}$-hard**.
The above result can be extended to the case of more general networks. For example, let GENERAL NETWORK DESIGN be the analogous optimization problem for single-commodity networks with arbitrary cost functions. Theorem 4.3.7 implies that the trivial algorithm here yields a $\lfloor n/2 \rfloor$-approximation result. As is the case with linear cost functions, we can prove the following inapproximability result, using the concept of Braess graphs. The full proof is omitted here. For more details on the hardness of network design, see [Rou01].

Theorem 4.3.15. *Assuming $\mathcal{P} \neq \mathcal{NP}$, for every $\epsilon > 0$, there is no $(\lfloor n/2 \rfloor - \epsilon)$-approximation algorithm for GENERAL NETWORK DESIGN.*

4.4 Reducing the Inefficiency of Equilibria

Back in section 4.2 we studied the PoA of selfish routing and we derived some tight bounds for it, depending on the type of cost functions in the network. In this section we present some widely used techniques for reducing the inefficiency of equilibria, i.e. for reducing the PoA. Apart from forcing optimal routing, which usually does not correspond to a flow at WE, there are three other popular techniques: increasing the capacity of the network, routing a small amount of traffic centrally (known as Stackelberg routing) and influencing traffic with edge taxes.

4.4.1 Capacity augmentation

We shall demonstrate this technique using an example:

Example 4.4.1. Consider the nonlinear variant of Pigou's example (see Example 1.3.2). Remember that with one unit of traffic, the WE routes all traffic on the lower edge, incurring 1 unit of cost, while the optimal flow routes ϵ units of flow on the upper edge and the rest $1 - \epsilon$ units on the lower edge, incurring 0 cost. Now consider the case, where we want to route 2 units of traffic through the network. Now the optimal flow is to route $1 + \epsilon$ units of flow on the upper edge and the rest $1 - \epsilon$ units on the lower edge. The cost of the optimal routing is now 2, as $p \to \infty$.

Example 4.4.1 implies a more general result: for every amount of traffic r, the optimal flow feasible for twice the original traffic (i.e. for $2r$) has cost at least equal to the cost of the WE for the original traffic (i.e. for r). In fact this result holds for all feasible flows of traffic $2r$, as the following theorem states.

Theorem 4.4.2. *If f is a WE for (G, r, c) and f^* is feasible for $(G, 2r, c)$, then*

$$C(f) \leq C(f^*)$$

For details on the proof see [RT02].

Example 4.4.1 is a tight example which shows that the above bound is the best possible, i.e. there exist feasible flows f^* s. t. $C(f) = C(f^*)$. In fact the equality holds for the optimal feasible flow of $(G, 2r, c)$.

In order for the title "capacity augmentation" to make sense, we need to rewrite Theorem 4.4.2 in the following equivalent form.

Corollary 4.4.3. *Let (G, r, c) be an instance and define the modified cost function $\tilde{c}_e$ by $\tilde{c}_e(x) = c_e(x/2)/2$ for each edge e. Let $\tilde{f}$ be a WE for $(G, r, \tilde{c})$ with cost $\tilde{C}(\tilde{f})$, and f^* a feasible flow for (G, r, c) with cost $C(f^*)$. Then $\tilde{C}(\tilde{f}) \leq C(f^*)$.*

Now notice that Corollary 4.4.3 takes a particularly nice form in the case of cost functions which represent M/M/1 delay functions. Then we have $c_e(x) = 1/(u_e - x)$ and $\tilde{c}_e(x) = 1/2(u_e - x/2) = 1/(2u_e - x)$. So in this case, we have the following advice: in order to outperform optimal routing, just double the capacity of every edge. In fact Theorem 4.4.2 says that by improving our links we can have even better results, than trying to route all traffic centrally (i.e. by telling all players what to do).

4.4.2 Stackelberg routing

The next technique is named after Stackelberg games, where there exist one leader and a lot of followers and the leader determines with his actions, the course of actions of the other players. Here the leader is some central authority routing a fraction $\gamma \in [0, 1]$ of the total amount of traffic, as he pleases, and the followers are the rest players of the network, which make their choices selfishly as usual. The main difference here between the central authority and the players is that the central authority controls a non-negligible portion of the total traffic and cannot be therefore considered as one more player in the network.[8]. Let us now describe the Stackelberg routing technique via two examples.

Example 4.4.4. Consider the nonlinear variant of Pigou's example in Figure 1.3 Suppose we are granted to route a $\gamma \in [0, 1]$ portion of the traffic as we wish, knowing that the rest $1 - \gamma$ fraction is routed selfishly by the players. We call a routing of the centrally controlled traffic a Stackelberg strategy. It is easy to see, that for every Stackelberg strategy, the rest $1 - \gamma$ portion of the traffic is routed through the lower edge, which is never worse than the upper edge. However, if we choose as a Stackelberg strategy, to route some of our own traffic γ on the upper edge, we reduce the total cost. The reason for that, is that we mimic the routing of the traffic in the optimal flow: we choose to route our traffic on the link that no rational player prefers, i.e. on the slower link. For γ sufficiently small ($\gamma \to 0$ as $p \to \infty$) we can induce a flow of 0 cost, i.e. exactly the optimal flow.

The next example shows that Stackelberg routing has nonetheless its limitations.

[8]this problem occurs because of the non-atomic nature of selfish routing

Example 4.4.5. Suppose we modify Example 4.4.4 by replacing the cost function of the lower edge by the cost function $c(x) = x^p/(1-\gamma)^p$. Now, supposing we route γ of the total traffic on the upper edge, the average cost is $\gamma \cdot 1 + \frac{(1-\gamma)^p}{(1-\gamma)^p} \cdot (1-\gamma) = \gamma + 1 - \gamma = 1$. Hence, no matter how we route the centrally controlled traffic, the lower edge will be fully congested and the average cost will be 1. On the other hand, the optimal flow routes $\gamma+\epsilon$ units of flow on the upper edge and the rest on the lower edge and its average cost is $(\gamma+\epsilon) \cdot 1 + \frac{(1-\gamma-\epsilon)^p}{(1-\gamma)^p} \cdot (1-\gamma) \to \gamma$ as $\epsilon \to 0$ and $p \to \infty$.

We now present a theorem that bounds the worst case ratio between the cost of the best flow possible with Stackelberg strategy and that of the optimal flow. Example 4.4.5 shows that this ratio can grow to get arbitrarily close to $1/\gamma$ even in two-node, two link networks. The corresponding upper bound follows from the following Theorem (the proof is in [Rou04]).

Theorem 4.4.6. *For every instance (G, r, c) with a* **network of parallel links***, and every $\gamma \in (0, 1]$, there is a Stackelberg strategy that routes γr units of traffic and yields a flow with cost at most $1/\gamma$ times the cost of an optimal flow.*

Theorem 4.4.6 provides a smooth trade-off between optimal flows and WE, as a function of the fraction of the centrally controlled traffic. When $\gamma = 0$, no traffic is centrally controlled and we are dealing with WE, which (in the case of arbitrary cost functions) can cost arbitrarily more than the optimal flow (unbounded PoA). On the other hand, if $\gamma = 1$ then we can route all traffic centrally and achieve the optimal flow.

Proposition 4.4.7. *The optimization problem of computing an optimal Stackelberg strategy is $\mathcal{NP}$-hard([Rou04]), though it can be approximated in polynomial time([KM02]).*

Theorem 4.4.6 applies only to networks of parallel links. The question of whether or not such a result holds for general single-commodity networks is open. Partial results have been derived however for a wide class of networks, including series-parallel networks and the Braess graphs.

4.4.3 Pricing network edges

One final, very natural approach to reduce congestion and thus the PoA, is to impose taxes on the edges. This subject has been extensively studied in the literature. Pigou originally suggested the *marginal cost taxes*. The basic idea is to charge each network user on each edge for the additional cost its

presence causes for the other users of the edge[9]. So let us denote by τ_e the tax of edge e. Then we have the selfish routing instance $(G, r, c + t)$, where at WE all traffic tries to minimize the incurred sum of edge latencies and taxes.
More formally, the principle of the marginal cost taxes asserts that for a flow f feasible for an instance (G, r, c), the tax τ_e is $\tau_e = f_e \cdot c'_e(f_e)$, where c'_e is the derivative of c_e. It is easy to see that c'_e is the increase of the edge's cost, due to one user (who controls a negligible portion of the traffic) and f_e is the total amount of traffic that suffers from this increase. So all traffic f_e (i.e. all users that consist this traffic) are charged an additional tax τ_e.
What is really important is that these taxes can in fact eliminate all of the inefficiency of the equilibria, as the following theorem states.

Theorem 4.4.8. *Let (G, r, c) be an instance with differentiable cost functions, admitting an optimal flow f^*. Let $\tau_e = f^*_e \cdot c'_e(f^*_e)$ denote the marginal cost tax for edge e with respect f^*. Then f^* is a WE for $(G, r, c + \tau)$.*

In other words, by imposing taxes on a selfish routing instance, we make an optimal flow be a WE.
The basic drawbacks of this method, are the universal handling of taxes and costs, as if they were the same thing, and the fact that the taxes can in some cases become too big (when the derivative is large), that they no longer consist a reasonable control measure for congestion.

[9]notice the resemblance with the idea of the VCG mechanisms

Bibliography

[CD06] Xi Chen and Xiaotie Deng. Settling the complexity of two-player nash equilibrium. In *FOCS*, pages 261–272, 2006.

[CDT06] Xi Chen, Xiaotie Deng, and Shang-Hua Teng. Computing nash equilibria: Approximation and smoothed complexity. In *FOCS*, pages 603–612, 2006.

[Che52] H. Chernoff. A measure of asymptotic efficiency for tests of a hypothesis based on a sum of observations. *Ann. Math. Statist.*, 23:493–507, 1952.

[CK05] George Christodoulou and Elias Koutsoupias. The price of anarchy of finite congestion games. In *STOC*, pages 67–73, 2005.

[CS03] Vincent Conitzer and Tuomas Sandholm. Complexity results about nash equilibria. In *IJCAI*, pages 765–771, 2003.

[CSM04] José R. Correa, Andreas S. Schulz, and Nicolás E. Stier Moses. Selfish routing in capacitated networks. *Math. Oper. Res.*, 29(4):961–976, 2004.

[CV02] Artur Czumaj and Berthold Vöcking. Tight bounds for worst-case equilibria. In *SODA*, pages 413–420, 2002.

[DGP05] Konstantinos Daskalakis, Paul W. Goldberg, and Christos H. Papadimitriou. The complexity of computing a nash equilibrium. *Electronic Colloquium on Computational Complexity (ECCC)*, (115), 2005.

[DMP06] Constantinos Daskalakis, Aranyak Mehta, and Christos H. Papadimitriou. A note on approximate nash equilibria. In *WINE*, pages 297–306, 2006.

[DP05] Konstantinos Daskalakis and Christos H. Papadimitriou. Three-player games are hard. *Electronic Colloquium on Computational Complexity (ECCC)*, (139), 2005.

[FKK+02] Dimitris Fotakis, Spyros C. Kontogiannis, Elias Koutsoupias, Marios Mavronicolas, and Paul G. Spirakis. The structure and complexity of nash equilibria for a selfish routing game. In *ICALP*, pages 123–134, 2002.

[FKS04] Dimitris Fotakis, Spyros C. Kontogiannis, and Paul G. Spirakis. Selfish unsplittable flows. In *ICALP*, pages 593–605, 2004.

[FKS05] Dimitris Fotakis, Spyros C. Kontogiannis, and Paul G. Spirakis. Symmetry in network congestion games: Pure equilibria and anarchy cost. In *WAOA*, pages 161–175, 2005.

[FPT04] Alex Fabrikant, Christos H. Papadimitriou, and Kunal Talwar. The complexity of pure nash equilibria. In *STOC*, pages 604–612, 2004.

[GLM+03] Martin Gairing, Thomas Lücking, Marios Mavronicolas, Burkhard Monien, and Paul G. Spirakis. Extreme nash equilibria. In *ICTCS*, pages 1–20, 2003.

[JPY88] David S. Johnson, Christos H. Papadimitriou, and Mihalis Yannakakis. On generating all maximal independent sets. *Inf. Process. Lett.*, 27(3):119–123, 1988.

[KM02] V. S. Anil Kumar and Madhav V. Marathe. Improved results for stackelberg scheduling strategies. In *ICALP*, pages 776–787, 2002.

[KMS03] Elias Koutsoupias, Marios Mavronicolas, and Paul G. Spirakis. Approximate equilibria and ball fusion. *Theory Comput. Syst.*, 36(6):683–693, 2003.

[KP99] Elias Koutsoupias and Christos Papadimitriou. Worst-case equilibria. *Lecture Notes in Computer Science*, 1563:404–413, 1999.

[KPS06] Spyros C. Kontogiannis, Panagiota N. Panagopoulou, and Paul G. Spirakis. Polynomial algorithms for approximating nash equilibria of bimatrix games. In *WINE*, pages 286–296, 2006.

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

[LRT04] Henry Lin, Tim Roughgarden, and Éva Tardos. A stronger bound on braess's paradox. In *SODA*, pages 340–341, 2004.

[LRTW05] Henry Lin, Tim Roughgarden, Éva Tardos, and Asher Walkover. Braess's paradox, fibonacci numbers, and exponential inapproximability. In *ICALP*, pages 497–512, 2005.

[Mil96] Igal Milchtaich. Congestion games with player-specific payoff functions. *Games and Economic Behavior*, 13(1):111–124, March 1996.

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

[MS01] Marios Mavronicolas and Paul G. Spirakis. The price of selfish routing. In *STOC*, pages 510–519, 2001.

[Nas51] J.F. Nash. Non cooperative games. *Annals of Mathematics*, 54:286–295, 1951.

[OR94] M.J. Osborne and A. Rubinstein. *A Course in Game Theory.* MIT Press, 1994.

[Pap] C. H. Papadimitriou. *Computational Complexity.*

[Pap94] Christos H. Papadimitriou. On the complexity of the parity argument and other inefficient proofs of existence. *J. Comput. Syst. Sci.*, 48(3):498–532, 1994.

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

[Ros73] R.W. Rosenthal. A class of games possessing pure-strategy nash equilibria. *International Journal of Game Theory*, 2:65–67, 1973.

[Rou01] Tim Roughgarden. Designing networks for selfish users is hard. In *FOCS*, pages 472–481, 2001.

[Rou02] Tim Roughgarden. The price of anarchy is independent of the network topology. In *STOC*, pages 428–437, 2002.

[Rou04] Tim Roughgarden. Stackelberg scheduling strategies. *SIAM J. Comput.*, 33(2):332–350, 2004.

[RT02] Tim Roughgarden and Éva Tardos. How bad is selfish routing? *J. ACM*, 49(2):236–259, 2002.

[SvS04] Rahul Savani and Bernhard von Stengel. Exponentially many steps for finding a nash equilibrium in a bimatrix game. In *FOCS*, pages 258–267, 2004.

[SY91] Alejandro A. Schäffer and Mihalis Yannakakis. Simple local search problems that are hard to solve. *SIAM J. Comput.*, 20(1):56–87, 1991.