Εθνικο Μετσοβιο Πολυτεχνειο
Σχολη Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων

Τομεας Τεχνολογιας Πληροφορικης και Υπολογιστων

Εργαστηριο Λογικης και Επιστημης Υπολογισμων

Υπολογισμός σημείων ισορροπίας Nash σε παίγνια μηδενικού αθροίσματος με δύο ομάδες

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Φοίβος Καλογιάννης

Επιβλέπων: Δημήτριος Φωτάκης
Αναπληρωτής Καθηγητής Ε.Μ.Π.

Αθήνα, Ιουλίος του 2022

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Υπολογισμός σημείων ισορροπίας Nash σε παίγνια μηδενικού αθροίσματος με δύο ομάδες

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Φοίβος Καλογιάννης

Επιβλέπων: Δημήτριος Φωτάκης
Αναπληρωτής Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 11η Ιουλίου 2022.

.....	.....	.....
Δημήτριος Φωτάκης	Ιωάννης Παναγέας	Αριστείδης Παγουρτζής
Αναπληρωτής Καθηγητής Ε.Μ.Π.	Assistant Professor UC Irvine	Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούλιος του 2022

.....
Φοίβος Καλογιάννης
(Διπλωματούχος Ηλεκτρολόγος Μηχανικός & Μηχανικός Υπολογιστών Ε.Μ.Π.)

Περίληψη

Στην παρούσα εργασία εξετάζουμε τη (μη-)σύγκλιση μίας σειράς γνωστών αλγορίθμων βελτιστοποίησης για τον υπολογισμό σημείων ισορροπίας Nash σε παίγνια δύο ομάδων μηδενικού αθροίσματος. Τα παίγνια δύο ομάδων μηδενικού αθροίσματος μπορούν να μοντελοποιήσουν τη δυναμική της σύγκρουσης μεταξύ δύο αντιτιθέμενων μερών χωρίς να καταφεύγουν σε απλοϊκοποίηση του μοντέλου ως μία σύγκρουση μεταξύ δύο μετα-παικτών. Από άποψη υπολογιστικής πολυπλοκότητας, δείχνουμε ότι το πρόβλημα υπολογισμού σημείων ισορροπίας Nash είναι CLS-δύσκολο. Στη συνέχεια, αποδεικνύουμε ότι για μία οικογένεια παιγνίων δύο ομάδων, μία σειρά αλγορίθμων πρώτου βαθμού (GDA, OGDA, EG, OMWU) αποτυγχάνουν να συγκλίνουν. Στον αντίποδα, συνεισφέρουμε τον σχεδιασμό ενός νέου αλγορίθμου πρώτου βαθμού που κάτω από ικανές συνθήκες συγκλίνει σε σημείο ισορροπίας Nash τόσο στη συγκεκριμένη οικόγενεια παιγνίων όσο και σε οποιοδήποτε παίγνιο (πιθανά μη κυρτό-μη κοίλο). Τέλος, παρουσιάζουμε έναν αριθμό πειραμάτων σε αρχιτεκτονικές νευρωνικών δικτύων (GANs) όπου η μοντελοποίηση τους ως σύγκρουση δύο ομάδων έχει προνομιακό πεδίο εφαρμογής.

Λέξεις κλειδιά: Θεωρία παιγνίων, Βελτιστοποίηση, Μέθοδοι Πρώτου Βαθμού, Δυναμικά Συστήματα, Υπολογιστική Πολυπλοκότητα.

Abstract

In this dissertation we examine the (non-)convergence of an array of commonly used min-max optimization algorithms for the purpose of computing Nash equilibria in two-team zero-sum games. Two-team zero-sum games can model the dynamics of a conflict between two opposing parties without over-simplifying the model into a mere conflict between two meta-players. From a computational complexity perspective, we show that the problem of computing Nash equilibria in this class of games is CLS-hard. Consecutively, we prove that in a simple, yet non-trivial, family of two-team zero-sum games a list of first-order methods (GDA, OGDA, EG, OMWU) fail to converge. On a brighter note, we contribute the design of a novel first-order algorithm that provably converges under some sufficient conditions that we provide both in the aforementdioned family of games as well as any (possibly nonconvex-nonconcave) game. Finally, we present a number of experiments in "multi-agent" generative adversarial neural networks whose training process can favorably be modelled as a two-team zero-sum game.

Keywords: Game theory, Optimization, First-Order Methods, Dynamical Systems, Computational Complexity.

Ευχαριστίες

Αρχικά οφείλω να ευχαριστήσω τους άμεσους συνεργάτες μου τον καθηγητή Ιωάννη Παναγέα και τον δόκτορα πλεόν Μανώλη Βλατάκη. Είμαι ευγνώμων για την εμπιστοσύνη που μου προσφέρανε, τα ενδιαφέροντα προβλήματα που μου συστήσανε, τη σωρεία γνώσεων που μου μετέφεραν αλλά και για το γεγονός ότι μου αποκάλυψαν πόσα ακόμα πράγματα υπάρχουν που δεν γνωρίζω. Επιπλέον, οφείλω να ευχαριστήσω τον καθηγητή Δημήτρη Φωτάκη ο οποίος με μύησε στη θεωρία υπολογισμού και μαζί με τον καθηγητή Αριστείδη Παγουρτζή έχουν εξασφαλίσει μέσω του εργαστηρίου Corelab τις αναγκαίες συνθήκες ώστε εντός του Πολυτεχνείου να λαμβάνει χώρα έρευνα στη θεωρητική πλευρά της επιστήμης υπολογιστών η οποία αξιώνει να έρχεται σε διάλογο με το διεθνές γίγνεσθαι της επιστήμης.
Η διπλωματική εργασία στέκεται, όπως ο οδηγός σπουδών της σχολής μας, ως το επιστέγασμα της μαθητείας στην σχολή. Ως εκ τούτου δεν οφείλω ευχαρίστιες μονάχα στους άμεσους συνεργάτες μου στην εργασία αυτή αλλά και στους ανθρώπους που με στήριξανε συναισθηματικά και υλικά και μου προσφέρανε ανεκτίμητες εμπειρίες και γνώσεις όσο σπούδαζα. Θέλω να ευχαριστήσω τους γονείς μου για την άνευ όρων επίμονη στήριξή τους υλική και συναισθηματική, τη σύντροφό μου Εύα που ήταν πλάι μου στην προσπάθειά μου και τα αδέρφια μου Δέσποινα, Ιάσονα και Βασίλη που αν και νεαρότερα μου μαθαίνουν συνεχώς καινούργια πράγματα. Οφείλω να ευχαριστήσω τους φίλους μου Βαγγέλη, Έλλη και Θανάση και τους φίλους που απέκτησα στη σχολή και με έκαναν με την παρουσία τους να την αγαπήσω περισσότερο: την Αγγελική, τη Βασιλική, τον Γιώργο, τη Δήμητρα, τον Νίκο και τον Νίκο, τον Θωμά, τον Θοδωρή, τον Σωτήρη και τον Χρήστο.

Contents

List of Figures

Κεφάλαιο α

Εισαγωγικό σημείωμα

Μέσω της παρούσας διπλωματικής εργασίας, ελπίζουμε να πραγματοποιήσουμε μία σύντομη ανασκόπηση τσις πρόσφατες εξελίξεις στη Θεωρία Παιγνίων και τη Βελτιστοποίηση και τελικά να ρίξουμε φως στην κατανόηση μίας μάλλον παραγκωνισμένης κλάσης παιγνίων, των παιγνίων δύο ομάδων. Επικεντρωνόμαστε ιδιαίτερα στη min-max βελτιστοποίηση για τον υπολογισμό ισορροπίας Nash σε παίγνια μηδενικού-αθροίσματος δύο-ομάδων, ενώ επίσης παρέχουμε ένα αποτέλεσμα υπολογιστικής δυσκολίας για το πρόβλημα του υπολογισμού τους.

Πριν εμβαθύνουμε σε τεχνικά θέματα, υπάρχει μια πολύ φυσική ερώτηση που μπορεί να θέσει κάποιος: Γιατί πρέπει να μας ενδιαφέρει η Θεωρία Παιγνίων; Η απάντηση είναι ότι θα πρέπει να μας ενδιαφέρει η Θεωρία Παιγνίων όσο μας ενδιαφέρει το πώς λαμβάνονται οι ορθολογικές αποφάσεις σε περιβάλλοντα όπου συμφέροντα συγκρούονται. Σημειώνουμε πολύ συνοπτικά μία ιστορική αναδρομή της Θεωρίας Παιγνίων.

Σύντομο ιστορικό σημείωμα Η σύγχρονη Θεωρία Παιγνίων ασχολείται με την *μαθηματική* εκτίμηση του τρόπου στρατηγικής δράσης σε καταστάσεις όπου παίκτες αλληλεπιδρούν μεταξύ τους λελογισμένα και στρατηγικά. Όπως μπορεί κανείς να μαντέψει, οι άνθρωποι ανέκαθεν προσπαθούσαν να καταλήξουν σε θεωρίες βέλτιστων στρατηγικών αλληλεπιδράσεων. Πολύ πριν εμπλακούν τα μαθηματικά στις θεωρίες αυτές, η στρατηγική σκέψη στην οικονομία, την πολιτική και τις στρατιωτικές υποθέσεις είχε συζητηθεί ρητά από τον Ινδό πολυμαθή Chanakya [28] ήδη από τον 4ο αιώνα π.Χ. ενώ ακόμη νωρίτερα και ειδικά για τις στρατιωτικές υποθέσεις από τον Κινέζο φιλόσοφο Sun Tzu τον 6ο αιώνα π.Χ..

Ένα ενδιαφέρον στιγμιότυπο πρώιμης μαθηματικής παιγνιοθεωρητικής σκέψης έχει κάνει πολύ πιθανά την εμφάνισή του στο Βαβυλωνιακό Ταλμούδ (0-500 μ.Χ.). Έχει υποστηριχθεί ότι μια φαινομενικά παράδοξη λύση σε ένα πρόβλημα χρεοκοπίας που προσφέρει το Ταλμούδ συμπίπτει με τη σύγχρονη λύση της θεωρίας παιγνίων του *nucleolus* στο *θεωρία παιγνίων συνεργασίας*[4].

Φυσικά, μπορούμε να επισημάνουμε την έκδοση του κλασικού πλέον κειμένου των Von Neumann και Morgenstern [66] ως το ορόσημο για την έναρξη του προγράμματος για μια μαθηματική θεωρία των παιγνίων και της στρατηγικής. Όμως, υπήρξαν περιπτώσεις σε μαθηματικά και οικονομικά κείμενα όπου κεντρικές έννοιες της σύγχρονης Θεωρίας Παιγνίων έκαναν μία πρωτόλεια εμφάνιση. Ξεκινώντας από τον 18ο αιώνα, σημειώνουμε ότι μια επιστολή με ημερομηνία το 1713 η οποία αποδίδεται στον Charles Waldegrave [5] αναφέρει μια minimax λύση μικτής στρατηγικής σε ένα παιχνδί με χαρτιά (το παιχνίδι "le Her"). Ο Augustin Cournot, σε ένα πρόβλημα ολιγοπωλίου με δύο επιχειρήσεις [12] προσφέρει μια λύση που συμπίπτει με την ισορροπία Nash για παίγνια δύο παικτών.

Επιπλέον, ένα έργο υψίστης σημασίας για το πεδίο είναι η απόδειξη του Nash για την ύπαρξη ισορροπιών Nash σε κάθε πεπερασμένο παίγνιο [43].

Κεφάλαιο β

Εκτεταμένη ελληνόγλωσση περίληψη

Σε αυτό το κεφάλαιο προσφέρουμε μία περιληπτική παρουσιάση των περιεχομένων της εργασίας στα ελληνικά. Εδώ περιλαμβάνονται ορισμοί βασικών εννοιών και ένα σκαρίφημα της έρευνάς μας. Για λεπτομέρειες παραπέμπουμε στο appendix και το αντίστοιχο αγγλόγλωσσο τμήμα του παρόντος κειμένου.

β.1 Κυρτή ανάλυση και βελτιστοποίηση

Η έννοια της *κυρτότητας* ανακύπτει συνεχώς στη θεωρία βελτιστοποίησης. Για τον λόγο αυτόν θεωρούμε σκόπιμο να αναφέρουμε κάποιους βασικούς ορισμούς από την κυρτή ανάλυση.
Αρχικά ορίζουμε το *κυρτό σύνολο*:

Ορισμός 1 (Κυρτό σύνολο)**.** *Ένα σύνολο $S \subseteq \mathbb{R}^d$ καλείται κυρτό όταν για κάθε δύο του στοιχεία $x, y \in S$ και κάθε $\lambda \in [0,1]$ το στοιχείο $z = \lambda x + (1-\lambda)y$ ανήκει στο σύνολο S.*

Ακολουθεί η έννοια της *κυρτής συναρτήσεως*:

Ορισμός 2 (Κυρτή συνάρτηση)**.** *Μία συνάρτηση f με πεδίο ορισμού ένα κυρτό σύνολο S με $f : S \to \mathbb{R}$ καλείται κυρτή όταν για κάθε δύο στοιχεία του πεδίου ορισμού της $x, y \in S$ και κάθε $\lambda \in [0,1]$ ισχύει η ακόλουθη ανισότητα:*

$$f\big(\lambda x + (1-\lambda)y\big) \leq \lambda f(x) + (1-\lambda) f(y).$$

Λήμμα 1 (Ανισότητα Jensen)**.** *Έστω μία κυρτή συνάρτηση $f : S \to \mathbb{R}$ και στοιχεία $x_1, \ldots, x_n \in S$. Για κάθε σύνολο συντελεστών $a_1, \ldots, a_n$ με $a_i \geq 0, \forall i \in [n]$ και $\sum_i a_i = 1$ ισχύει η ακόλουθη ανισότητα:*

$$f\left(\sum_i^n a_i x_i\right) \leq \sum_i^n a_i f(x_i).$$

Η απόδειξη του τελευταίου λήμματος έπεται από τον ορισμό της κυρτής συνάρτησης.

β.2 Βασικές έννοιες θεωρίας παιγνίων

β.2.1 Παίγνια σε κανονική μορφή

Ορισμός 3 (Παίγνιο σε κανονική μορφή)**.** *Ένα παίγνιο σε κανονική μορφή ορίζεται ως η πλειάδα* $\Gamma = \Gamma(\mathcal{N}, \mathcal{A}, \boldsymbol{u})$ *αποτελούμενη από:*

- *ένα πεπερασμένο σύνολο* $\mathcal{N}$ *από* N *παίκτριες,* $\mathcal{N} = \{1, \cdots, N\}$
- *μία συλλογή από πεπερασμένα σύνολα* $\mathcal{A} = \{A_1, \cdots, A_N\}$ *όπου* A_i *είναι το σύνολο των διαθέσιμων αμιγών στρατηγικών/κινήσεων στην παίκτρια* i
- *μία διανυσματική συνάρτηση* $\boldsymbol{u} = (u_1, \cdots, u_N)$ *της οποίας το* i*-οστό στοιχείο αντιπροσωπεύει τη συνάρτηση ωφέλειας/κέρδους της παίκτριας* i*,* $u_i : \Delta(\mathcal{A}) \to \mathbb{R}$ *που απεικονίζει κατανομές πιθανότητας επί των στρατηγικών σε ένα βαθμωτό μέγεθος, την ωφέλεια/το κέρδος που αποκομίζει η παίκτρια αυτή.*

Με $\Delta(A_i)$ συμβολίζουμε το μονόπλεγμα πιθανοτήτων επί του συνόλου A_i, δηλαδή το σύνολο όλων των διανυσμάτων $\boldsymbol{x}$ με διάσταση $d = |A_i|$ και $x_a \geq 0$, $\sum_{a=0}^{d} x_a = 1$.
Ένα διάνυσμα στρατηγικής για την παίκτρια i ορίζεται ως ένα διάνυσμα $\boldsymbol{x}$ πιθανοτήτων επί των στρατηγικών $\boldsymbol{x} \in \Delta(A_i)$. Όταν μονάχα ένα στοιχείο του $\boldsymbol{x}$ είναι μη-μηδενικό, τότε το διάνυσμα αυτό ονομάζεται *αμιγής στρατηγική* ενώ όταν το διάνυσμα έχει παραπάνω από ένα μη-μηδενικό στοιχείο ονομάζεται *μικτή στρατηγική*.

Ορισμός 4 (Παίγνιο δύο-παικτών μηδενικού-αθροίσματος)**.** *Ένα παίγνιο δύο-παικτών μηδενικού-αθροίσματος σε κανονική μορφή ορίζεται ως η πλειάδα* $\Gamma = \Gamma(\mathcal{N}, \mathcal{A}, \boldsymbol{u})$ *όπου:*

- *το πλήθος των παικτών είναι* $|\mathcal{N}| = 2$
- *ο χώρος στρατηγικών* $\mathcal{A}$ *περιέχει δύο πεπερασμένα σύνολα αμιγών στρατηγικών για κάθε παίκτρια* $\mathcal{A} = \{A_1, A_2\}$
- *για τη συνάρτηση* $\boldsymbol{u} = (u_1, u_2)^\top$ *ισχύει* $-u_1 = u_2 = u$ *και* $u = \boldsymbol{x}^\top A \boldsymbol{y}$ *όπου* $A \in \mathbb{R}^{n \times m}$ *με* $n = |A_1|$*,* $m = |A_2|$*.*

Η παίκτρια με τη στρατηγική $\boldsymbol{y}$ ελέγχει τη στήλη του πίνακα A και προσπαθεί να μεγιστοποιήσει την ποσότητα u ενώ η παίκτρια στρατηγική $\boldsymbol{x}$ ελέγχει τη γραμμή του πίνακα προσπαθώντας να ελαχιστοποιήσει την u.
Το παιχνίδι αυτό μπορεί να κωδικοποιηθεί στην παρακάτω παράσταση:

$$\min_{\boldsymbol{x} \in \Delta(A_1)} \max_{\boldsymbol{y} \in \Delta(A_2)} \boldsymbol{x}^\top A \boldsymbol{y}.$$

Με βάση το Minimax θεώρημα του Von Neumann γνωρίζουμε ότι:

$$\min_{\boldsymbol{x} \in \Delta(A_1)} \max_{\boldsymbol{y} \in \Delta(A_2)} \boldsymbol{x}^\top A \boldsymbol{y} = \max_{\boldsymbol{y} \in \Delta(A_2)} \min_{\boldsymbol{x} \in \Delta(A_1)} \boldsymbol{x}^\top A \boldsymbol{y}.$$

Δηλαδή με απλούς όρους, δεν έχει σημασία ποια από τις δύο παίκτριες θα παίξει πρώτη. Είτε παίξει πρώτα η παίκτρια-γραμμή είτε η παίκτρια-στήλη, οι βέλτιστες στρατηγικές τους παραμένουν οι ίδιες όπως και οι πληρωμές τους.
Ένας πολύ απλός και αποδοτικός τρόπος να "λύσουμε" τα παιχνίδια δύο-παικτών μηδενικού-αθροίσματος είναι μέσω του *γραμμικού προγραμματισμού*. Μάλιστα, η ισχυρή και ασθενής *δυϊκότητα* του γραμμικού προγραμματισμού προσφέρουν μία εναλλακτική απόδειξη της του θεωρήματος του Von Neumann.

β.3 Σημεία ισορροπίας Nash

Η παραπάνω "λύση" του παιχνιδιού είναι μία πολύ ειδική περίπτωση αυτού που στη θεωρία παιγνίων ονομάζουμε *σημείο ισορροπίας Nash*, δηλαδή μίας κατατάστασης από την οποία κανείς/καμία παίκτης/παίκτρια δεν μπορεί να παρεκκλίνει μονομερώς και να αποκομίσει καλύτερη πληρωμή.

Ορισμός 5 (Σημείο Ισορροπίας Nash)**.** *Ένα γινόμενο κατανομών πιθανοτήνων $x = (x_1, \ldots, x_n)$ αποτελούμενο από κατανομές x επί του A_i σε ένα παίγνιο μεγιστοποίησης κέρδους ονομάζεται* σημείο ισορροπίας Nash *εάν για κάθε παίκτρια $i \in [n]$ και κάθε αμιγή στρατηγική $a'_i \in A_i$ ισχύει η ακόλουθη ανισότητα*

$$\mathbb{E}_{a\sim x}\left[u_i(a)\right] \geq \mathbb{E}_{a\sim x}\left[u_i(a'_i; a_{i-1})\right] \tag{NE}$$

Τα σημεία ισορροπίας Nash είναι πλέον αναφαίρετο κομμάτι της θεωρίας παιγνίων και δύσκολα θα μπορούσε να φανταστεί την τελευταία χωρίς αυτήν. Στην πραγματικότητα η απόδειξη της ύπαρξής τους σε κάθε πεπερασμένο παίγνιο (δλδ. παίγνιο με πεπερασμένο πλήθος παικτών και στρατηγικών) χρονικά έπεται της έκδοσης του πρώτου κειμένου θεωρίας παιγνίων.

Θεώρημα 1 (Nash)**.** *Κάθε πεπερασμένο παίγνιο N παικτών έχει τουλάχιστον ένα σημείο ισορροπίας Nash.*

Η απόδειξη του θεωρήματος βασίζεται σε ένα θεώρημα σταθερού σημείου. Συγκεκριμένα, ο John F. Nash απέδειξε το θεώρημα το 1950 με το θεώρημα σταθερού σημείου του Kakutani ενώ το 1951 με χρήση του θεωρήματος σταθερού σημείου του Brouwer.
Σημειώνουμε ότι η απόδειξη μέσω της χρήσης του θεωρήματος σταθερού σημείου του Brouwer σε συνάρτηση με την απόδειξή του τελευταίου μέσω του λήμματος του Sperner έχει αποδειχθεί καίριας σημασίας για την εξέλιξη της θεωρίας πολυπλοκότητας υπολογισμού σημείων ισορροπίας Nash.

β.4 Μέθοδοι βελτιστοποίησης πρώτου βαθμού

Η επίλυση του παιχνιδού μέσω γραμμικού προγραμματισμού αφορά σε ένα "συγκεντρωτικό" τρόπο επίλυσης. Δηλαδή, οι παίκτες πρέπει να εμπιστευτούν έναν αλγόριθμο ο οποίος με είσοδο το παιχνίδι θα αναθέσει σε κάθε παίκτη μία στρατηγική.
Αν αντίθετα θέλουμε κάθε παίκτρια να μπορεί *να μάθει ενώ παίζει*, αποκεντρωμένα, θα πρέπει να ακολουθήσουμε ένα διαφορετικό υπόδειγμα επίλυσης παιγνίων.
Επιγραμματικά θα αναφέρουμε τον μετα-αλγόριθμο της οκνηρής online κατοπτρικής κατάβασης (lazy online mirror descent).
Το πλαίσιο στο οποίο μαθαίνει μία παίκτρια είναι αυτό της online κυρτής βελτιστοποίησης (ή του επαναλαμβανόμενου παιχνιδιού). Σε κάθε χρονική στιγμή tη παίκτρια επιλέγει μία κίνηση a^t από ένα σύνολο δυνατών κινήσεων ενώ μία αντίπαλος επιλέγει μία κυρτή συνάρτηση κόστους f_t. Σκοπός της παίκτριας είναι να μετανιώσει εκ των υστέρων τις επιλογές της το λιγότερο δυνατό.

Η έννοια του regret και της εξομάλυνσης Προτού μιλήσουμε για τον μετα-αλγόριθμο οκνηρής κατοπτρικής κατάβασης πρέπει να ορίσουμε δύο ακρογωνιαίες έννοιες, 1. την έννοια του regret ("μετάνοιας") καθώς και 2. την έννοια της εξομάλυνσης (regularization).
Ο μετα-αλγόριθμος κατοπτρικής κατάβασης γενικεύει τον αλγόριθμο κατάβασης κλίσης (gradient descent). Αντί να πραγματοποιήσει το βήμα κατάβασης ευθέως στον χώρο τον μεταβλητών, πραγματοποιεί κατάβαση στον χώρο των "πληρωμών" και στη συνέχεια προβάλει τη νέα τιμή στο χώρο των μεταλητών. Για το βήμα κατάβασης χρησιμοποιείται και μία συνάρτηση εξομάλυνσης R. Η παράλληλη ελαχιστοποίηση του κόστους με την ελαχιστοποίηση της "απόστασης" (που επάγει η

συνάρτηση εξομάλυνσης R) προηγούμενου και επόμενου σημείου υλοποιεί τον συμβιβασμό ανάμεσα στην ανάγκη κατάβασης και την ανάγκη μη-απόκλισης από την τελευταία τιμή της μεταβλητής.

Ο μετα-αλγόριθμος online mirror descent Ο μετα-αλγόριθμος οκνηρής κατοπτρικής κατάβασης περιγράφεται από την εξής σχέση, έστω $\boldsymbol{y}_1 : R(\boldsymbol{y}_1) = 0$ και $\boldsymbol{x}_1 = B_R(\boldsymbol{x}, \boldsymbol{y}_1)$.

$$\begin{aligned}\boldsymbol{y}_{t+1} &= (\nabla R)^{-1} \left(\nabla R(\boldsymbol{y}_t) - \nabla f_t(\boldsymbol{x}_t)\right)\\ \boldsymbol{x}_{t+1} &= \arg\min_{\boldsymbol{x}\in\mathcal{X}} B_R(\boldsymbol{x}, \boldsymbol{y}_{t+1}),\end{aligned}$$

με B_R την "απόσταση" Bregman (Bregman divergence) η οποία ορίζεται για τη συνάρτηση εξομάλυνσης R.
Η έννοια του regret είναι ένα μέτρο να αξιολογήσει εκ των υστέρων η παίκτρια τις επιλογές κινήσεων που έκανε κατά την διάρκεια του παιχνιδιού συγκρίνοντας τις με την καλύτερη δυνατή επιλογή κίνησης, δηλαδή:

$$\mathrm{Reg}^{(T)} = \sum_{t=0}^{T} f_t(a^t) - \min_{a^*} \sum_{t=0}^{T} f_t(a^*)$$

Για την εξομάλυνση επιλέγουμε ισχυρά-κυρτές συναρτήσεις όπως π.χ. την ℓ_2-νόρμα και την αρνητική εντροπία. Η χρήση διαφορετικών συναρτήσεων εξομάλυνσης οδηγεί σε διαφορετικούς αλγορίθμους.
Για την περίπτωση χρήσης της ℓ_2-νόρμας ανακτούμε τον αλγόριθμο κατάβασης κλίσης (gradient descent) για τον ένα παίχτη (απλή ελαχιστοποίηση) και τον αλγόριθμο κατάβασης/ανάβασης κλίσης(gradient descent-ascent) για τους δύο παίχτες. Ενώ με την χρήση της αρνητικής εντροπίας τον αλγόριθμο multiplicative weights update.

β.5 Δυναμικά συστήματα και ευστάθεια

Όπως σε πολλούς τομείς των μαθηματικών και της μηχανικής, έτσι και στη θεωρία βελτιστοποίησης, η θεωρία των δυναμικών συστημάτων αποδεικνύεται εξόχως χρήσιμη και σημαντική. Αδρά, ένα δυναμικό σύστημα απαρτίζεται από 1. τον *χώρο φάσεων* δηλαδή κάθε δυνατή κατάσταση στην οποία μπορεί να βρεθεί ένα σύστημα 2. τον *χρόνο* ο οποίος δύναται να είναι διακριτός ή συνεχής 3. έναν *κανόνα χρονικής εξέλιξης* που υπαγορεύει πώς το σύστημα μεταβαίνει από μία κατάσταση σε κάποια άλλη.
Συγκεκριμένα, υπάρχουν κυρίως δύο είδη δυναμικών συστημάτων: *διαφορικές εξισώσεις* και *εξισώσεις διαφορών* που περιγράφουν την εξέλιξη συστημάτων σε συνεχή και διακριτό χρόνο αντίστοιχα.

Δυναμικά συστήματα συνεχούς χρόνου Υποθέτουμε τον *χώρος φάσεων* $\mathcal{S} \subset \mathbb{R}^n$ να είναι ένα ανοικτό σύνολο και την f μία συνεχώς διαφορίσιμη απεικόνιση με $f : \mathcal{S} \to \mathcal{S}$. Η ακόλουθη μορφή ορίζει ένα *αυτόνομο, συνεχο χρόνου* δυναμικό σύστημα:

$$\frac{d\boldsymbol{x}}{dt} = f(\boldsymbol{x}).$$

Δυναμικά συστήματα διακριτού χρόνου Πάλι, θεωρούμε τον χώρο καταστάσεων $\mathcal{S} \subset \mathbb{R}^n$ ως ένα ανοιχτό σύνολο και f μία απεικόνιση $f : \mathcal{S} \to \mathcal{S}$. Τότε ένα *αυτόνομο, διακριτού χρόνου* δυναμικό σύστημα μπορεί να περιγραφεί ώς:

$$\boldsymbol{x}^{(k+1)} = f(\boldsymbol{x}^{(k)}).$$

Έπεται ότι οι επαναληπτικοί αλγόριθμοι μπορούν να περιγραφούν με σχεδόν προφανή τρόπο από δυναμικά συστήματα διακριτού χρόνου. Η παρατήρηση αποδεικνύεται ιδιαιτέρως βοηθητική στην ανάλυση και απόδειξη σύγκλισης και μη-σύγκλισης επαναληπτικών αλγορίθμων.

Η έννοια της ευστάθειας κατά Lyapunov Εδώ εισάγουμε την έννοια του σημείου ισορροπίας κατά Lyapunov:

Ορισμός 6 (Σημείο ισορροπίας/σταθερό σημείο)**.** *Ένα σημείο $\hat{\boldsymbol{x}} \in \mathcal{S}$ καλείται σημείο ισορροπίας του δυναμικού συστήματος $f : \mathcal{S} \to \mathcal{S}$ αν:*

$$\mathbf{0} = f(\hat{\boldsymbol{x}})$$

Καλούμε ένα σημείο ισορροπίας $\hat{\boldsymbol{x}}$ μεμονωμένο εάν δεν υπάρχει γειτονιά U του $\boldsymbol{x}$ τέτοια ώστε το σημείο $\hat{\boldsymbol{x}}$ να είναι το μοναδικό σημείο ισορροπίας στην U.

Ορισμός 7. *Θεωρήστε το αυτόνομο δυναμικό σύστημα f:*

$$\boldsymbol{x}^{(k+1)} = f(\boldsymbol{x}^{(k)})$$

Το σημείο ισορροπίας $\hat{\boldsymbol{x}} = \mathbf{0}$ καλείται:

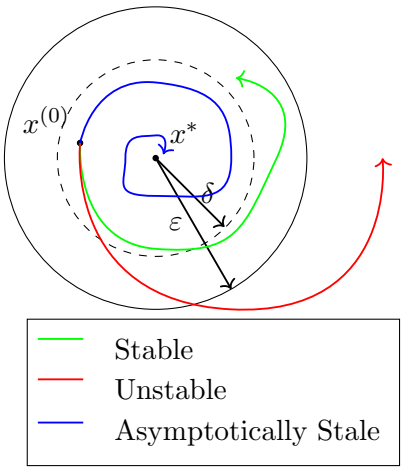
- ***ευσταθές*** *εάν, για κάθε $\varepsilon > 0$, $\exists \delta = \delta(\varepsilon) > 0$ τέτοια ώστε:*

$$\|\boldsymbol{x}^{(0)}\| < \delta \;\Rightarrow \|\boldsymbol{x}^{(k)}\| < \varepsilon, \;\forall k > 0$$

- ***ασταθές*** *εάν δεν είναι ασταθής*
- ***ασυμπτωτικά ευσταθές*** *εάν είναι ευσταθές και επιπλέον το δ μπορεί να επιλεχθεί τέτοιο ώστε:*

$$\|\boldsymbol{x}^{(0)}\| < \delta \;\Rightarrow \lim_{k\to\infty} \boldsymbol{x}^{(k)} = \mathbf{0}$$

Θα χρησιμοποιούμε τους όρους *σταθεροποίηση* και *σύγκλιση* μίας επαναληπτικής διαδικασίας καθώς η ασυμπτωτική ευστάθεια ενός σταθερού σημείου αντιστοιχεί σχεδόν ακριβώς στη σύγκλιση σε αυτό.



β.6 Παίγνια δύο-ομάδων μηδενικού αθροίσματος

Είμαστε πλέον σε θέση να κάνουμε λόγο για το κύριο μέρος της δουλειάς μας το οποίο εστιάζει σε παίγνια δύο-ομάδων μηδενικού-αθροίσματος. Στα παίγνια αυτά ένα πεπερασμένο πλήθος παικτών είναι χωρισμένο σε δύο ομάδες. Αυτό που εννοείται εδώ με τον όρο "ομάδα" περιορίζεται στο γεγονός ότι όλοι/όλες οι παίκτες/παίκτριες μοιράζονται το ίδιο ακριβώς κέρδος το οποίο διαμορφώνεται από τις κινήσεις κάθε παίκτη/παίκτριας.
Αυστηρά μιλώντας, ο ορισμός δίνεται ως εξής:

Ορισμός 8. *Ένα παίγνιο δύο-ομάδων σε κανονική μορφή ορίζεται από μία πλειάδα* $\Gamma = \Gamma(\mathcal{N}, \mathcal{A}, u)$ *που αποτελείται από*

- *ένα πεπερασμένο πλήθος παικτριών* $\mathcal{N}$*, χωρισμένων σε δύο ομάδες* A, B *με* k_A *και* k_B *το πλήθος παίκτριες αντίστοιχα ούτως ώστε:* $\mathcal{N} = \mathcal{N}_A \cup \mathcal{N}_B = \{A_1, \cdots, A_{k_A}, B_1, \cdots, B_{k_B}\}$
- *ένα πεπερασμένο πλήθος κινήσεων (ή αμιγών στρατηγικών)* $\mathcal{A}_i = \{\alpha_1, \ldots, \alpha_{n_i}\}$ *ανά παίκτρια* $i \in \mathcal{N}$
- *τη συνάρτηση ωφέλειας/κέρδους κάθε ομάδας* $u_A, u_B : \mathcal{A} \to \mathbb{R}$*, όπου* $\mathcal{A} := \prod_i \mathcal{A}_i$ *συμβολίζει την αλληλουχία από όλες τις πιθανές διαμορώσεις στρατηγικών του συνόλου των παικτριών* $\alpha = (\alpha_{A_1}, \ldots, \alpha_{A_{k_A}}, \alpha_{B_1}, \ldots, \alpha_{B_{k_B}})$ *ενώ η ατομική ωφέλεια μίας παίκτριας είναι ταυτόσημη με αυτή των συμπαιχτριών της, δλδ,* $u_i = u_A$ & $u_j = u_B$ $\forall (i,j) \in \mathcal{N}_A \times \mathcal{N}_B$.

Σε αυτό το γενικό συγκείμενο, οι παίκτριες μπορούν να ενεργήσουν με βάση *μικτές στρατηγικές*, δλδ, κατανομές πιθανοτήτων $s_k \in \Delta(\mathcal{A}_k)$ επί των κινήσεων $\alpha_k \in \mathcal{A}_k$. Αντίστοιχα, ορίζουμε το γινόμενο κατανομών $\boldsymbol{x} = (s_{A_1}, \ldots, s_{A_{k_A}})$, $\boldsymbol{y} = (s_{B_1}, \ldots, s_{B_{k_B}})$ ως την στρατηγική της κάθε ομάδας. Εν τέλει, θα γράφουμε $\mathcal{X} := \prod_{i \in \mathcal{N}_A} \mathcal{X}_i = \prod_{i \in \mathcal{N}_A} \Delta(\mathcal{A}_i), \mathcal{Y} := \prod_{i \in \mathcal{N}_A} \mathcal{Y}_i = \prod_{i \in \mathcal{N}_B} \Delta(\mathcal{A}_i)$ σημειώνοντας τον χώρο όλων των πιθανών στρατηγικών κάθε ομάδας A, B.
Όμοια με τα δι-γραμμικά παίγνια δύο παικτών, οι συναρτήσεις ωφέλειας των δύο ομάδων μπορούν να εκφραστούν με τη χρήση ενός *τανυστή-κέρδους* $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{\tau}$ με $\tau = \prod_{i \in \mathcal{N}} |\mathcal{A}_i|$ δίνοντας τη μορφή:

$$u_A = \mathbf{A}_{\boldsymbol{x}}^{\boldsymbol{y}} \;\&\; u_B = \mathbf{B}_{\boldsymbol{x}}^{\boldsymbol{y}}$$

Όταν το παίγνιο δύο ομάδων είναι μηδενικού αθροίσματος έχουμε ότι:

$$u = -u_A = u_B.$$

β.6.1 Για την υπολογιστική πολυπλοκότητα υπολογισμού Nash ισορροπίας

Το πρώτο φυσικό ερώτημα που μπορεί να ανακύψει μετά τον ορισμό μίας κλάσης παιγνίων είναι το πόσο δύσκολη υπολογιστικά μπορεί να αποδειχθεί η απόπειρα υπολογισμού σημείων ισορροπίας Nash. Για να το απαντήσουμε αυτό δείχνουμε ότι μία κλάση παιγνίων γνωστής δυσκολίας μπορεί να αναχθεί σε παίγνια δύο-ομάδων μηδενικού-αθροίσματος. Αυτό βέβαια προδίδει το "κάτω φράγμα" της δυσκολίας τους και όχι το "άνω". Δηλαδή, αποδεικνύουμε ότι το πρόβλημα υπολογισμού Nash ισορροπίας στα εν λόγω παίγνια είναι *δύσκολο* για μία κλάση υπολογιστικής πολυπλοκότητας αλλά όχι *πλήρες*.
Η κλάση πολυπλοκότητας για την οποία ο υπολογισμός ϵ-προσεγγιστικής Nash ισορροπίας είναι δύσκολος είναι η κλάση CLS. Σε αυτή την κλάση τα προβλήματα επιδέχονται μίας συνάρτησης δυναμικού η οποία ελαττώνεται επί των μονοπατιών που βελτιώνουν την προσεγγιστική λύση. Η απόδειξη βασίζεται στην αναγωγή ενός παιγνίου συμφόρησης σε ένα παίγνιο δύο-ομάδων μηδενικού-αθροίσματος.

Θεώρημα 2. *Το πρόβλημα υπολογισμού ϵ-προσεγγιστικού σημείου Nash ισορροπίας είναι* CLS-*δύσκολο.*

β.6.2 Για τη θεωρία βελτιστοποίησης στον υπολογισμό Nash ισορροπίας σε παίγνια δύο-ομάδων

Το δεύτερο μέρος των αποτελεσμάτων μας για τη θεωρία των παιγνίων δύο-ομάδων μηδενικού-αθροίσματος εστιάζει στη χρήση επαναληπτικών αλγορίθμων πρώτου βαθμού για τον υπολογισμό Nash σημείων ισορροπίας.
Προτού εκθέσουμε το πρώτο αρνητικό αποτέλεσμα πρέπει να προσθέσουμε ένα ορισμό για τον χαρακτηρισμό των σημείων Nash ισορροπίας.

Ορισμός 9 (Ασθενώς ευσταθή σημεία ισορροπίας Nash)**.** *Έστω ένα σημείο ισορροπίας Nash $x = (x_1, \ldots, x_n)$. Έστω ότι ένας παίκτης j αλλάξει τη μικτή στρατηγική του x_j σε μία αμιγή από όσες παίζει με μήδενική πιθανότητα ήδη. Αν όλοι οι άλλοι παίκτες $-j$ παραμείνουν αδιάφοροι σε αυτήν την αλλαγή ως προς τις αμιγείς στρατηγικές που παίζουν με θετική πιθανότητα ήδη, τότε το σημείο ισορροπίας x καλείται ασθενώς ευσταθές.*

Τώρα παρουσιάζουμε το πρώτο και πιο γενικό αποτέλεσμά μας:

Θεώρημα 3. *Τα μη ασθενώς ευσταθή σημεία ισορροπίας είναι ασταθή για τον αλγόριθμο gradient descent-ascent. Δηλαδή, ο αλγόριθμος gradient descent-ascent συγκλίνει σε αυτά με πιθανότητα* 0.

Προχωράμε στον ορισμό ενός παιγνίου δύο-ομάδων μηδενικού-αθροίσματος με δύο παίκτριες ανά ομάδα. Το παίγνιο κωδικοποιείται με τον εξής κανόνα: "συντονίσου με τις συμπαίκτριές και παίξε ένα παιχνίδι matching pennies, μη συγχρονιστείς και χάσε κατά ω", όπου ω είναι μία θετική σταθερή παράμετρος του παιχνιδιού.
Το παιχνίδι αυτό το καλούμε generalized matching pennies (GMP) και ορίζεται από τον παρακάτω πίνακα:

	ΚΚ	ΚΓ/ΓΚ	TT
ΚΚ	$1, -1$	$\omega, -\omega$	$-1, 1$
ΚΓ/ΓΚ	$-\omega, \omega$	$0, 0$	$-\omega, \omega$
ΓΓ	$-1, 1$	$\omega, -\omega$	$1, -1$

Λήμμα 2 (Μοναδικό σημείο ισορροπίας Nash)**.** *Το παίγνιο generalized matching pennies εμφανίζει ένα και μοναδικό σημείο ισορροπίας Nash equilibrium το οποίο είναι το*

$$(\boldsymbol{x}^*, \boldsymbol{y}^*) = ((\frac{1}{2}, \frac{1}{2}), (\frac{1}{2}, \frac{1}{2})).$$

Η οικογένεια αυτή παιγνίων ενώ είναι απλή, παραμένει μία οικογένεια παιγνίων μη κυρτών-μη κοίλων παρόλο που η μη-κυρτότητα οφείλεται απλώς στην πολυγραμμικότητα της συνάρτησης ωφέλειας. Επιπλέον παρατηρούμε ότι για τα παίγνια αυτά $\min_{\boldsymbol{x}\in\mathcal{X}} \max_{\boldsymbol{y}\in\mathcal{Y}} u(\boldsymbol{x}, \boldsymbol{y}) \neq \max_{\boldsymbol{y}\in\mathcal{Y}} \min_{\boldsymbol{x}\in\mathcal{X}} u(\boldsymbol{x}, \boldsymbol{y})$.
Παρόλα αυτά τα παίγνια αυτά είναι μία απλή και φυσική γενίκευση παιγνίων δύο παικτών.

Θεώρημα 4 (Οι αλγόριθμοι GDA, OGDA, EG και OMWU αποτυγχάνουν)**.** *Θεωρήστε το παίγνιο GMP με* $\omega \in (0,1)$. *Υποθέστε* $\eta_{GDA} < \frac{1}{4}$, $\eta_{OGDA} < \min(\omega, \frac{1}{8})$, $\eta_{EG} < \frac{\omega}{2}$, *και* $\eta_{OMWU} < \min\left(\frac{1}{4}, \frac{\omega}{2}\right)$ *(φράγματα από το μέγεθος βήματος για τους* GDA, OGDA, EG, *και* OMWU *αντίστοιχα). Ισχύει ότι οι* GDA, OGDA, EG *και* OMWU *συγκλίνουν με πιθανότητα* 0 *όταν επιλέγουμε τυχαία τις αρχικές τιμές των στρατηγικών.*

Τέλος, σημειώνουμε τη μέθοδο που συνεισφέραμε για τον υπολογισμό σημείων ισορροπίας Nash τόσο σε παίγνια δύο-ομάδων μηδενικού αθροίσματος όσο και σε κάθε πιθανώς μη κυρτό-μη κοίλο το οποίο ικανοποιεί την ικανή συνθήκη που αναφέρουμε.
Η μέθοδος είναι η εξής, αν $\boldsymbol{z}^\top = (\boldsymbol{x}^\top, \boldsymbol{y}^\top)$ και $\boldsymbol{\theta}^\top = (\boldsymbol{\theta}_{\boldsymbol{x}}^\top, \boldsymbol{\theta}_{\boldsymbol{y}}^\top)$ όπου $\boldsymbol{\theta}_{\boldsymbol{x}}, \boldsymbol{\theta}_{\boldsymbol{y}}$ είναι ίδιας διάστασεις με τα $\boldsymbol{x}, \boldsymbol{y}$ αντίστοιχα:

$$\begin{cases} \boldsymbol{z}^{(k+1)} = \Pi_{\mathcal{Z}}\left\{\boldsymbol{z}^{(k)} + \eta\binom{-\nabla_{\boldsymbol{x}} f(\boldsymbol{z}^{(k)})}{\nabla_{\boldsymbol{y}} f(\boldsymbol{z}^{(k)})} + \eta K(\boldsymbol{z}^{(k)} - \boldsymbol{\theta}^{(k)})\right\} \\ \boldsymbol{\theta}^{(k+1)} = \Pi_{\mathcal{Z}}\left\{\boldsymbol{\theta}^{(k)} + \eta P(\boldsymbol{z}^{(k)} - \boldsymbol{\theta}^{(k)})\right\} \end{cases}$$

Επιπλέον K, P είναι πίνακες κατάλληλου μεγέθους και υπερ-παράμετροι του αλγορίθμου.
Μπορούμε να επιλέξουμε $K = k \cdot \mathbf{I}$, $P = p \cdot \mathbf{I}$ ώστε να εξασφαλίσουμε ανερξάρτητη εκμάθηση για κάθε παίκτρια. Έτσι προχωράμε στο επόμενο θεώρημά μας:

Theorem β.6.1. *Υποθέστε ένα παίγνιο δύο-ομάδων μηδενικού-αθροίσματος ούτως ώστε η ωφέλεια της ομάδας* B *να είναι* $U(\boldsymbol{x}, \boldsymbol{y})$ *και αντίστοιχα της ομάδας* A *να είναι* $-U(\boldsymbol{x}, \boldsymbol{y})$. *Θεωρήστε ένα σημείο ισορροπίας Nash* $(\boldsymbol{x}^*, \boldsymbol{y}^*)$ *για το παίγνιο. Επιπλέον έστω:*

$$H := \begin{pmatrix} -\nabla^2_{\boldsymbol{xx}} U(\boldsymbol{x}^*, \boldsymbol{y}^*) & -\nabla^2_{\boldsymbol{xy}} U(\boldsymbol{x}^*, \boldsymbol{y}^*) \\ \nabla^2_{\boldsymbol{yx}} U(\boldsymbol{x}^*, \boldsymbol{y}^*) & \nabla^2_{\boldsymbol{yy}} U(\boldsymbol{x}^*, \boldsymbol{y}^*) \end{pmatrix}.$$

και E *το σύνολο των ιδιοτιμών* ρ *του* H *με θετικό πραγματικό μέρος, δλδ.* $E = \{H's\ eigenvalues\ \rho : Re(\rho) > 0\}$. *Υποθέτουμε ότι ο πίνακας* H *είναι αντιστρέψιμος και επιπλέον:*

$$\beta = \min_{\rho \in E} \frac{Re(\rho)^2 + Im(\rho)^2}{Re(\rho)} > \max_{\rho \in E} Re(\rho) = \alpha.$$

Θέτοντας $K = k \cdot \mathbf{I}$, $P = p \cdot \mathbf{I}$. *Τότε, υπάρχει αρκόυντως μικρό μέγεθος βήματος* $\eta > 0$ *και θετικό βαθμωτό* $p > 0$ *για κάθε* $k \in (-\beta, -\alpha)$ *τέτοια ώστε η προτεινόμενη μέθοδος με τους πίνακες* K, P *όπως τους επιλέξαμε να συγκλίνει τοπικά στο σημείο* $(\boldsymbol{x}^*, \boldsymbol{y}^*)$.

Chapter 1

Introduction

Through the present thesis, we hope to review recent advances in Game Theory and optimization and ultimately shed some light on understanding a rather overlooked setting of games, two-team games. We particularly focus on min-max optimization for Nash equilibrium computation in two-team zero-sum games while we also provide a computational hardness result for the problem of computing them.
Before delving into technical matters, there is a very natural question for someone to ask: Why should we care about Game Theory? The answer is that we should care about Game Theory as much as we care about how rational decisions are made. We very briefly look at different dimensions of Game Theory and how it has interacted with other sciences as well as society. Next, we talk about this field's fascinating meeting with Computer Science at which point we encounter several matters that organically give rise to the scientific concerns this thesis aspires to address.

1.1 Game Theory from different perspectives

Short historical note Modern Game Theory is concerned with *mathematically* assessing how to act in situations where strategical agents interact with each other. As one might guess, people have always tried to come up with theories of optimal strategical interactions. Long before mathematics was involved, strategic thought in the economy, politics, and military affairs has explicitly been discussed by Indian polymath Chanakya [28] as early as the 4th century BCE and even earlier particularly for military affairs by Chinese philosopher and strategist Sun Tzu in the 6th century BCE.
An interesting instance of early mathematical game-theoretic thought has quite likely made its appearance in the Babylonian Talmud (0-500 CE). It has been argued that an ostensibly strange solution to a bankruptcy problem given by coincides with the modern game-theoretic solution of the *nucleolus* in the *theory of coalitional games*[4].
Of course, we can pinpoint the release of Von Neumann and Morgenstern's now classic text [66] as the milestone for the start of the program for a mathematical theory of games and strategy. But, there have been instances in mathematical and economic texts where central concepts of modern Game Theory have made an early appearance. Starting in the 18th century, we note that a letter dated 1713 and attributed to Charles Waldegrave [5] contributes a mixed-strategy minimax solution to a game of cards called le Her. Augustin Cournot, in a problem of duopoly [12] theorizes a solution that coincides with the Nash equilibrium for two-player

games. Moreover, a work of utmost importance to the field is Nash's proof of the existence of Nash equilibria in every finite game [43].

Philosophical foundations of Game Theory A *game* in our context is nothing but a metonymy for all situations entailing multiple agents that act strategically in order to maximize their happiness/utility and whose actions affect each other. Of course, the notion of *utility* and *rationality* can only be defined axiomatically. In Game Theory, we suppose that every agent's desire is to maximize a measurable quantity that is referred to as *utility* and a *rational* decision is the one that maximizes it. This model of an agent's desires can be traced back to the utilitarian philosophy of Jeremy Bentham[6]:

> Nature has placed mankind under the governance of two sovereign masters, pain and pleasure. It is for them alone to point out what we ought to do, as well as to determine what we shall do. On the one hand, the standard of right and wrong, on the other the chain of causes and effects, is fastened to their throne. They govern us in all we do, in all we say, in all we think: every effort we can make to throw off our subjection, will serve but to demonstrate and confirm it. In words a man may pretend to abjure their empire: but in reality, he will remain. subject to it all the while. The principle of utility recognizes this subjection and assumes it for the foundation of that system, the object of which is to rear the fabric of felicity by the hands of reason and of law.

We will not argue philosophically whether this exclusionary binary of pleasure and pain is a good model for the drives of a human person and sequentially the meaning of rationality (see for example the intricate relationship between pleasure, pain, and a person's drives drawn in the work of Sigmund Freud [19]). Nevertheless, we note that even in the textbook that initiated the project of Game Theory, «Theory of games and economic behavior», the authors recognize that the assumption of a numerical value that measures the utility is rather questionable.
Of course, a numerical value that can model the goal of a given agent and the assumption that an agent's sole motive is the maximization of this quantity makes for a convenient set of assumptions in the analysis of strategic interactions.
There is an interesting further discussion on different aspects of Game Theory, briefly narrated in [34].

Game Theory and Society Undeniably, the formal theory of games began as a part of Economics. As such, it is bound to have implications that affect societal reality in one way or the other.
Apart from its impact on Economics, Game Theory has been applied to military applications [16, 25] and [32] provides a very interesting historical retrospective of Game Theory's relations to other fields of society.

Game Theory and Biology Price laid the foundation of *Evolutionary Game Theory* or the application of mathematical Game Theory to the genetic Darwinian competition of evolving populations. Further, Smith and Price initiated game-theoretic research of animal behavior. Ever since Evolutionary Game Theory has an established collection of works [69, 65, 60]. Many concepts that take into account the dimension of time originate from the encounter in these fields. Further, the application of contemporary Game Theory results has yielded mathematically informed conjectures about evolutionary phenomena [38].

1.2 Game Theory & Computer Science

The formal encounter between Game Theory and the Theory of Computation has proven immensely fruitful for both disciplines. It has yielded a number of directions and concepts that would be extremely unlikely to have sprouted if the fields had kept on developing individually. Of course, one of the forefathers of Game Theory, John Von Neumann, has been proven to be one of the most important computer scientists to date and certain ideas have been shared between the fields before the more recent formal encounter such as the Linear Programming Duality theory and the Minimax theorem of Von Neumann.
Today, we can speak about such concepts as the *Price of Anarchy* and the *Price of Stability*, the *algorithmic Mechanism Design*, the *Complexity of computing equilibria*, and a lot more thanks to this fortunate meeting.
The internet itself has catalyzed the development of the field of Algorithmic Game Theory. Mechanism design theory for example has a broad use in ad markets and auctions taking place online in intensive volumes. Of course, prior to that, broadband sharing and competition boosted the need for a theory of *congestion games*. Notably, algorithmic Game Theory landed Koutsoupias and Papadimitriou, Roughgarden and Tardos, Nisan and Ronen the Gödel prize for the effects of selfish internet use.

Optimization in games After the ground rules of a game have been laid, what is effectively left for an agent to do is to *optimize* her strategy or *solve the game*. We can draw a rough distinction between algorithms that compute a solution to the game in a *centralized* manner or a *decentralized* one. What is meant by centralized is that the computation of a solution is performed by a central "authority" of some sort that has access to information regarding every player while in a decentralized algorithm every agent performs necessary computation on their own with the information that is only available to them.
As far as centralized algorithms go, Linear Programming offered a way to compute Nash equilibria in two-person zero-sum games and the Lemke-Howson algorithm did so in two-person general-sum games. The analysis of decentralized algorithms for equilibrium computation was done by Julia Robinson [49] who proved the conjectured convergence to Nash equilibrium posed in [9] for an algorithm (*fictitious-play*) under which every player acts upon self-interest in a *decentralized* manner.

Applications in actual games So far we have not actually talked about Game Theory applied in games. Actually, through the vehicle of optimization and Machine Learning techniques, several games that were intricate and hard to mathematically reason about have been "solved" in the sense that some computer software is able to beat the best human players in each game.
Chess was probably the first game that received attention from the emerging computer science community. Claude Shannon himself laid his hands on the problem of automating the strategy-making in chess and contributed an algorithm [55]. Automated chess-playing has improved in leaps and bounds giving rise to a super-human performance that uses Machine Learning (reinforcement learning in particular) and optimization theory [56].
The game of *backgammon* received one of the first applications of optimization in games [64] that achieved near-human performance through the use of reinforcement learning and a training scheme that lets the learning agent play against itself.
Checkers is yet another game that has received interest early by the computer science community. Arthur Samuel provided the *Checkers Playing Program* that additionally used Machine

Learning in its design [53]. Finally, we mention the crown jewel of automated (board) game playing that achieves super-human game performance.
The game of *go* is a board game invented in China more than 2500 years ago and has been played continually since. It consists of a simple board, two sets of marbles, and two simple rules. The number of legal moves has been estimated to be around 2.1×10170. The game takes years to master and players receive formal training throughout Asia much like players of Chess do in other parts of the world. Yet, Deepmind's AlphaGo [57] managed to beat Lee Sedol, probably the top player of the game in the world at the time, in four out of five games in 2016.

Applications in Machine Learning This topic is of special interest for this thesis. The connection between Machine Learning and Game Theory has as previously mentioned contributed to several game-playing software applications able to beat the best humans. But there are many advances in which game-theoretic reasoning made possible, including *robust* machine learning models against so-called *adversarial examples* [36], long-standing state-of-the-art *generative models*, namely Generative Adversarial Networks [21, 1].

1.3 Our Motivation

As we briefly mentioned, the joint encounter of Game Theory, Optimization, and Machine Learning has borne fruit. We can now elaborate on some more technical issues that highlight the necessity to investigate some settings that very naturally give rise to the question we concern ourselves with in this thesis:

"Can we (efficiently) compute Nash equilibria in two-team zero-sum games?"

Generative Adversarial Networks Generative Adversarial Networks [21] are a mathematical device engineered to capture the parameters of unknown distributions in the form of Nash equilibria of a (nonconvex-nonconcave) zero-sum game. The game is played between two opposing neural networks, a *generator* G and a *discriminator* D. A generator strives to learn to mimic a given dataset p_{real} (e.g. pictures) with the goal of making the discriminator unable to tell between a *real* sample and a *generated/false* one. Formally, the optimization objective is the following:

$$\min_D \max_G \mathbb{E}_{z\sim p_{\text{real}}} [\log D(z)] + \mathbb{E}_{z\sim \text{noise}} [\log (1 - D(G(z)))] \qquad \text{(GAN Opt. Func.)}$$

Since the beginning of GAN research, the instability of its training process (vanilla GDA) made its presence felt, i.e. the model could enter cycles without ever approximating the distribution (see [13] for a more nuanced analysis of this behavior). Apart from that, a phenomenon known as mode collapse was observed – the model outputs samples whose features are centered around a small subset (*mode*) of the real distribution. The Wasserstein-GAN [1] forces Lipschitzness on the function of the discriminator network and uses a Wasserstein distance objective function in an attempt to remedy cycling behavior and mode collapse. Plus, the training process leveraged the (at the time novel) Adam optimization algorithm [30].
The theory-oriented part of the community attempted to give solutions to the drawbacks of the training process. In fact, vanilla GDA was already known to converge only *in average* to Nash equilibria in normal-form games. Averaging the parameters of nonconvex function approximators is irrelevant, hence the interest in methods with *last-iterate* convergence guarantees.

In fact, [14] devised an Optimistic version of the Adam algorithm beating the vanilla-Adam trained WGAN in terms of *Inception Score*.

Multi-generator/discriminator GANs A different approach to surpassing the obstacles of GANs has been followed both by the application-oriented part of the community [44, 18, 26] as well as the theory-oriented one [2]. This approach focuses on the phenomenon of mode collapse and proposes the addition of extra generators and/or discriminators to remedy it. Arora et al. underline and prove the expressive superiority of multi-generator, multi-discriminator GANs.

Questions raised by multi-agent applications Although the results are promising, the *objective function landscape* of games between two opposing teams and the *training dynamics* are not very well understood. We follow the thread of these questions and hope to shed a fraction of light on the equilibrium learning dynamics of such settings.

1.4 Our Contribution

We contribute the proof of the CLS-hardness of Nash equilibrium computation in normal-form two-team zero-sum games, a *class-membership* proof that remained elusive and *remained an open question*. Next, we show that many first-order algorithms fail to converge in a very simple (but non-trivial) family of two-team zero-sum games we designed. This family of games can serve as a benchmark for theoretical and applied future advances. Lastly, we contribute the design of a novel first-order method that leverages ideas from Control Theory and we manage to describe a sufficient condition of the last-iterate local-convergence of the method to any game that satisfies them.

Chapter 2

Preliminaries

2.1 Convex analysis

Although in the current work we are concerned with nonconvex(-nonconcave) functions and possibly nonconvex feasible sets, a preliminary note on convex analysis is deemed necessary. Convex analysis is the field of mathematics occupied with the concept of *convexity*. It is inherently related to the concept of mathematical optimization and its ideas and tools are omnipresent in fields like machine learning, leading to a dramatic rise of interest in the field in recent years. Convexity is –intuitively speaking– about the property of elements of a set being bound to end up in the same set after certain kinds of "interactions" amongst them. The reader should probably have by now acquainted themselves with some kind of convexity throughout their studies. Usually this incorporates thinking about convexity in terms of *convex functions* and the property of things ending up in the initial set might seem somewhat confusing. We are going to try and bridge the layman's grasp of convexity as tangent lines of a curve remaining below it at any point and the initial informal description about the property of sets whose elements will under certain "operations" result in elements belonging to the initial set.

Notation

Notation will not stray from conventions most people hold when writing mathematics. The d-dimensional Euclidean space will be signified by $\mathbb{R}^d$; scalars will be denoted with symbols like x, y, α, β, etc while random variables will favor capital letters like X and Y. Vectors (be they random or deterministic) will be written in the usual bold font: $\boldsymbol{x}$, $\boldsymbol{y}$, $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, etc and ℓ_p vector norms will be represented by $\|\cdot\|_p$.

2.1.1 Convex Sets

Let's consider the set of $\mathbb{R}$ as well as the sets $\mathbb{R}^d$ where $d \in \mathbb{N}$ known. Vectors $\boldsymbol{x} \in \mathbb{R}^d$ are d-tuples $(x_1, x_2, \ldots, x_d)$. We will now move on to discuss certain concepts in the broad concept of convexity limited to such subsets of $\mathbb{R}^d$ and functions defined on them. Let's start with the elementary definition of a convex set. (We will use the words *vector* and *point* more or less interchangeably for the time being).

Definition 2.1.1 (Convex Set)**.** A subset C of $\mathbb{R}^d$ is said to be *convex* when for every pair

$\boldsymbol{x}, \boldsymbol{y} \in C \subseteq \mathbb{R}^d$ and every $\lambda \in \mathbb{R}$ for which $0 \leq \lambda \leq 1$ the following holds:

$$\boldsymbol{z} = (1 - \lambda)\boldsymbol{x} + \lambda \boldsymbol{y} \in C$$

Geometric Interpretation. A set C is called convex when for every pair of points $\boldsymbol{x}, \boldsymbol{y}$ every point $\boldsymbol{z}$ on the straight line segment defined by the pair lies within C.

Theorem 2.1.1. *The intersection of convex sets is a convex set.*

Proof. Proof is trivial, easily derived from the definition. ■

Great. We defined what a convex set is using two vectors. What's in for us if we are to consider more than two vectors? Best we can do for the time being is a *convex combination* of vectors. It's really nothing fancy:

Definition 2.1.2 (Convex combination of vectors)**.** Let there be n vectors $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n \in \mathbb{R}^d$ and n non-negative coefficients λ_i such that $\lambda_1 + \lambda_2 + \ldots \lambda_n = 1$, any vector $\boldsymbol{z}$ for which the following holds is called a *convex combination* of the former vectors:

$$\lambda_1 \boldsymbol{x}_1 + \lambda_2 \boldsymbol{x}_2 + \ldots + \lambda_n \boldsymbol{x}_n$$

Geometric Interpretation. The set of the convex combinations of n vectors $\boldsymbol{x}_i$ is the *convex hull* of the set of points defined by $\boldsymbol{x}_i$.

Theorem 2.1.2. *A set is convex iff it contains all the convex combinations of its elements.*

2.1.2 Geometric concepts in $\mathbb{R}^d$

We will now move to define some extensions to certain concepts one finds quite familiar and useful in 2-D or 3-D space in order to not only establish a common ground with regards to vocabulary, but also help motivate intuitive geometric thinking as it can prove to be quite valuable and fruitful in the context of convex analysis.

Definition 2.1.3. (Half-space) Given the space $\mathbb{R}^n$, a vector $\boldsymbol{a}$ and a scalar b with $\boldsymbol{a} \in \mathbb{R}^n$, $b \in \mathbb{R}$ then the set that is defined by the inequality:

$$\boldsymbol{a}^\top \boldsymbol{x} \leq b, \boldsymbol{x} \in \mathbb{R}^n$$

is called a (closed) half-space of $\mathbb{R}^n$ If the inequality holds strictly (i.e. $\boldsymbol{a}^\top \boldsymbol{x} \leq \boldsymbol{b}$) the set is called an open half-space of $\mathbb{R}^n$.

Definition 2.1.4. (Hyper-plane) Let vector $\boldsymbol{a} \in \mathbb{R}^n$ and scalar $b \in \mathbb{R}$ be constant and $\boldsymbol{x} \in \mathbb{R}^n$ a variable. The set of points that lie on the set H defined by the equation:

$$\boldsymbol{a}^\top \boldsymbol{x} = b$$

is called a hyper-plane.

Observe that a hyper-plane in $\mathbb{R}^n$ has dimension $n-1$. The concept of the hyper-plane extends what one naturally would call a line in $\mathbb{R}^2$ (i.e. a sub-space of $\mathbb{R}^2$ with a basis of dimension 1) and a plane in $\mathbb{R}^3$ (i.e. the sub-space of $\mathbb{R}^3$ that has a basis of dimension 2). One more observation is that the vector $\boldsymbol{a}$ is perpendicular to the hyper-plane. Abusing the analogy we draw between liners $\mathbb{R}^2$ and hyper-planes in $\mathbb{R}^d$, we could also refer to $\boldsymbol{a}$ as the *slope* of the hyper-plane.

Definition 2.1.5. (Supporting half-space) Let C be a convex set $C \in \mathbb{R}^n$. A supporting half-space is a half-space that:

- contains C
- has a point of C on its boundary.

It follows naturally that one would define a *supporting hyper-plane* in the following way:

Definition 2.1.6. (Supporting hyper-plane) The boundary of a supporting half-space of a convex set C is called a supporting half-space.

2.1.3 Convex functions

Since we have already defined some fundamental ideas that will function as building blocks for more complex concepts, we will move on to discuss the matters that will mainly concern us, namely convexity with respect to a function.

2.1.3.1 Definitions

Let f be a function mapping values from $\mathbb{R}^d$ to $\mathbb{R}$. We can imagine f as defining a hyper-surface in the joint space of its input space and its output space, $\mathbb{R}^d \times \mathbb{R}$. The points above that surface whose perpendicular projections on $\mathbb{R}^d$ remain in $\operatorname{dom} f$ form the epigraph of the given function. More formally:

Definition 2.1.7. An *epigraph* of a fuction $f : \mathbb{R}^n \to \mathbb{R}$ is said to be the set of points $(\boldsymbol{x}, \mu)$ such that $\mu \geq f(\boldsymbol{x})$ and it is noted as:

$$\operatorname{epi} f = \{(\boldsymbol{x}, \mu) \,|\, \mu \geq f(\boldsymbol{x})\}$$

In 2.1 the epigraphs of two different function can be seen.

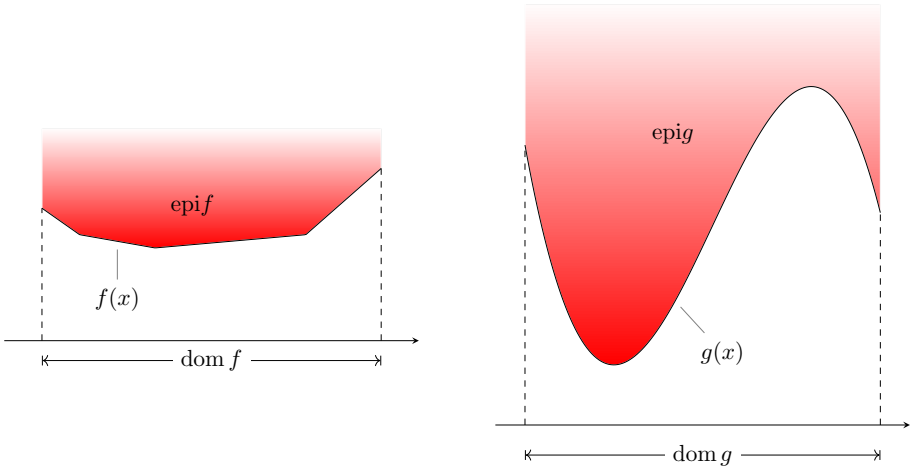


Figure 2.1: Two different functions with their respective epigraphs (the fading red areas)

Definition 2.1.8 (Convex function)**.** A function $f : \mathbb{R}^d \to \mathbb{R}$ is said to be convex when $\operatorname{dom} f$ is convex and for any $\boldsymbol{x}, \boldsymbol{y} \in \operatorname{dom} f$ for any $t \in [0, 1]$ the following inequality holds:

$$f(t\boldsymbol{x} + (1-t)\boldsymbol{y}) \leq tf(\boldsymbol{x}) + (1-t)f(\boldsymbol{y})$$

When the latter inequality is strict we say that f is strictly convex.

This definition can extend to what we will call Jensen's inequality, generalizing the inequality from the convex combination of two points to a convex combination of n points is something that can at times be proven quite useful.

Theorem 2.1.3 (Jensen's inequality)**.** *Let f be a convex function, $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m \in \operatorname{dom} f$ and $\lambda_1, \lambda_2, \ldots, \lambda_m \in [0, 1]$ such that $\lambda_1 + \lambda_2 + \ldots + \lambda_m = 1$. Then, it always holds that:*

$$f(\lambda_1\boldsymbol{x}_1 + \lambda_2\boldsymbol{x}_2 + \ldots + \lambda_m\boldsymbol{x}_m) \leq \lambda_1 f(\boldsymbol{x}_1) + \lambda_2 f(\boldsymbol{x}_2) + \ldots + \lambda_m f(\boldsymbol{x}_m)$$

Another definition of a convex function is the one that uses a function's epigraph.

Definition 2.1.9 (Convex function – alternative definition)**.** A function is convex when epi$f$ is a convex set.

We can also demonstrate that the two latter definitions are equivalent.
For the sake of completeness, we will need to give the definitions of some more concepts, namely that of the effective domain, which we have already used without giving a proper definition, and that of a proper function.

Definition 2.1.10 (Effective Domain of a Convex Function)**.** The effective domain of a convex function $\operatorname{dom} f$ is the set of $\boldsymbol{x}$ s.t:

$$\operatorname{dom} f = \{\boldsymbol{x} \mid f(\boldsymbol{x}) < -\infty\}$$

Definition 2.1.11 (Proper function)**.** A function f is called proper if its epigraph is non-empty and contains no vertical lines.

Proposition 2.1.1. *Let f be a convex function, f is proper iff there exists at least one point $\boldsymbol{x}$ such that $f(\boldsymbol{x}) < +\infty$ and $f(\boldsymbol{x}) > -\infty$ anywhere else. Or equivalently, its effective domain $\operatorname{dom} f$ is non-empty and f takes at least one finite value.*

One more theorem we are going to state about all convex functions, but not yet prove, is the following one that can make us think of convex functions in an intriguing way.

Theorem 2.1.4. *Every closed convex function f is the pointwise supremum of the colleciton of all affine functions h such that $h \leq f$.*

Geometric Interpretation. The latter tells us that if we consider a function $f : \mathbb{R} \to \mathbb{R}$, the curve defined by f can be described at any one of its points as the maximum value of all the lines h for which $h \leq f$.

2.1.3.2 First and Second Order Conditions

Of course, however easy it might be to grasp the definition, it could fall short in usefulness or practicality with respect to trying to characterize a function as being convex or not. Luckily the next two theorems –restricted on differentiable and twice differentiable functions– can offer a way that can prove quite helpful in characterizing functions as convex or not.

Theorem 2.1.5 (First Order Condition)**.** *Let a function $f : \mathbb{R}^d \to \mathbb{R}$ be differentiable. Then, f is convex iff* dom f *is a convex set and for any $\boldsymbol{x}, \boldsymbol{y} \in$ dom f the next inequality holds:*

$$f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^\top (\boldsymbol{y} - \boldsymbol{x})$$

Geometric Interpretation. What the latter means is quite simple to comprehend. Given two points $x, y \in$ dom f, regardless of their relative position (i.e. it could very well be $x < y$ or $x > y$), if we were to start following the tangent line to the curve that passes trough $\big(x, f(x)\big)$ with horizontal direction that would lead to y, we will consistently find ourselves *below* $f(y)$. We tried to illustrate this in figure 2.2.

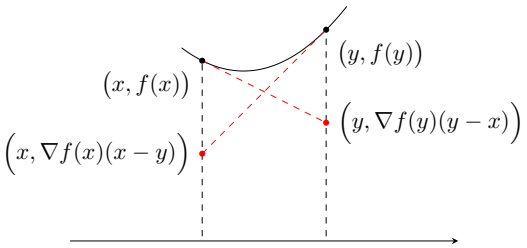


Figure 2.2: Illustration of the first order condition

Theorem 2.1.6 (Second Order Condition)**.** *Let a function f be twice differentiable and* dom f *convex. Function f is convex iff for every $\boldsymbol{x} \in$ dom f:*

$$\nabla^2 f(\boldsymbol{x}) \succeq 0$$

(i.e. the Hessian matrix[1] of f is positive semi-definite or $\boldsymbol{z}^\top \nabla^2 f(\boldsymbol{x}) \boldsymbol{z} \geq 0, \forall \boldsymbol{z}$)

2.1.4 Conjugate Transform & Fenchel's inequality

In this section we are going to discuss the conjugate transform of functions. It is a transform that maps the parameters of hyper-planes tangent to the curve of a function to a certain value.

[1]Reminder: $\nabla^2 = \begin{pmatrix} \frac{\partial^2}{\partial x_1^2} & \frac{\partial^2}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2}{\partial x_1 \partial x_n} \\ \frac{\partial^2}{\partial x_2 \partial x_1} & \frac{\partial^2}{\partial x_2^2} & \cdots & \frac{\partial^2}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2}{\partial x_n \partial x_1} & \frac{\partial^2}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2}{\partial x_n^2} \end{pmatrix}$

It may not be the first time one sees such a transform, one that shifts our attention to a parameter space.
Our subject revolves around tangent lines (or hyper-planes for function domains with dimension greater than 1) on a convex function. We will demonstrate a way that has been devised in order to represent elegantly the whole set of these tangent lines.

Definition 2.1.12 (Convex conjugate)**.** Let a function f be convex. We define its conjugate transform as the function f^* such that:

$$f^*(\boldsymbol{p}) = \sup_{\boldsymbol{x} \in \text{dom } f} \{\boldsymbol{p}^\top \boldsymbol{x} - f(\boldsymbol{x})\}$$

Frankly the definition seems a bit awkward. Considering its geometric interpretation could maybe shed some light as to what this is supposed to mean.

Geometric Interpretation. The conjugate transform of a function f is merely a function f^* that maps slopes α to the maximum available offset β such that the given line $\alpha x + \beta$ will be tangent to the curve defined by f.

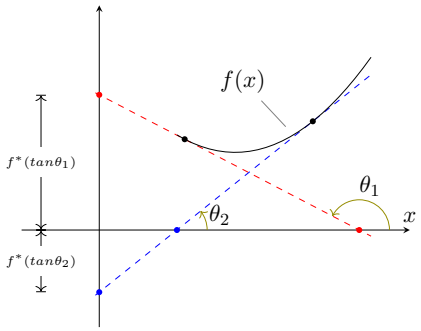


Figure 2.3: Geometric meaning of the conjugate transform

The conjugate transform f^* of a function, f if certain conditions hold for the latter, can give us all the information we need about f. Keeping in mind the theorem about f being described as the point-wise supremum of all affine functions h such that $h \leq f$, it seems rather intuitive. We will state this formally:

Theorem 2.1.7 (Fenchel-Moreau Theorem)**.** *Let f be lower semi-continuous*[2] *and convex, then:*

$$f^{**} = f \tag{2.1}$$

2.1.5 Subgradients, subdifferentials

2.1.5.1 Definitions

This section is concerned with function that are not everywhere differentiable. Although we cannot define a gradient at a given point, it may be sufficient to substitute an exact gradient

[2]Reminder: A function is lower semi-continuous at x_0 if for every $\epsilon > 0$ there exists a neighborhood U of x_0 such that $f(x) \geq f(x_0) - \epsilon \ \forall x \in U$ if $f(x_0) < +\infty$. Else if $f(x_0) \to +\infty$ then $f(x) \to +\infty$ as well.

with slope-vectors that will always undeshoot the value of function in question for any given pair of points.
However informal is the initial description of the subgradient might have been, the formal definition does not fall far:

Definition 2.1.13. (Subgradient) Let a function f be convex. Any vector $\boldsymbol{p}$ is called a subgradient of f at a point $\boldsymbol{x} \in \operatorname{dom} f$ if for any $\boldsymbol{z} \in \operatorname{dom} f$:

$$f(\boldsymbol{z}) \geq f(\boldsymbol{x}) + \boldsymbol{p}^\top \boldsymbol{z} - \boldsymbol{x}$$

Definition 2.1.14. (Subdifferential) The set of every subgradient vector of a convex function f at point $\boldsymbol{x}$ is called the subdifferential, $\partial f(\boldsymbol{x})$, of f at point $\boldsymbol{x}$.

In figure 2.4 we tried to illustrate a number of subgradients for the function f that is not everywhere differentiable.

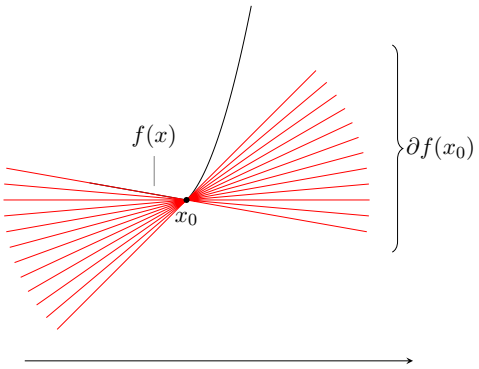


Figure 2.4: The subdifferential $\partial f(x_0)$ is the set of all subgradients which are represented by the red lines. A subgradient at point x_0 will always undershoot the value of f at any point z.

Since we have seen a quite intuitive definition of the subdifferential, why not use our intuition as a stepping stone in order to grasp a more technical definition of it? Before we move on to define subgradients alternatively, we will need to remind ourselves the notion of directional derivatives.

Definition 2.1.15 (Directional Derivative)**.** The directional derivative $f'(\cdot, \cdot)$ of a function f at point $\boldsymbol{x}$ in direction $\boldsymbol{d}$ is defined as:

$$f'(\boldsymbol{x}, \boldsymbol{d}) = \lim_{t \downarrow 0} \frac{f(\boldsymbol{x} + t\boldsymbol{d}) - f(\boldsymbol{x})}{t}$$

(Obviously, $t > 0$)

We can now define the subdifferential with respect to the directional derivative.

Definition 2.1.16 (Subdifferential–Alternative Definition)**.** The subdifferential ∂f of f at point $\boldsymbol{x}$ is the set defined as such:

$$\partial f(\boldsymbol{x}) = \{\boldsymbol{s} \,\big|\, \boldsymbol{s}^\top \boldsymbol{d} \leq f'(\boldsymbol{x}, \boldsymbol{d}),\ \forall \boldsymbol{d} \in \mathbb{R}^d\}$$

Vectors $\boldsymbol{s}$ are the subgradients of f at point $\boldsymbol{x}$.

2.1.6 Fenchel's inequality

Since we have now seen both the conjugate transform and the definition of the subgradient, we can move on to state Fenchel's inequality:

Theorem 2.1.8 (Fenchel's inequality)**.** *For any subgradient vector $\boldsymbol{p} \in f^*(\operatorname{dom} f^*)$ and any $\boldsymbol{x} \in \operatorname{dom} f$ the following inequality stands:*

$$f^*(\boldsymbol{p}) + f(\boldsymbol{x}) \geq \boldsymbol{p}^\top \boldsymbol{x}$$

2.1.7 Mean Value Theorems

Subgradients seem to magically lift restrictions placed upon the validity of known theorems when functions are non-differentiable. An instance of this is the reinstatement of the mean value theorem for functions that are convex but not everyhwere differentiable.

Theorem 2.1.9 (Differential form)**.** *Let a function $f : \mathbb{R}^d \to \mathbb{R}$ be convex. For any $\boldsymbol{x}, \boldsymbol{y} \in \operatorname{dom} f$, there will always exist a $t \in (0,1)$ for which $\boldsymbol{z} = t\boldsymbol{x} + (1-t)\boldsymbol{y}$ and $\boldsymbol{s} \in \partial f(\boldsymbol{z})$ such that:*

$$f(\boldsymbol{y}) - f(\boldsymbol{x}) = \boldsymbol{z}^\top \boldsymbol{y} - \boldsymbol{x}$$

Theorem 2.1.10 (Integral form)**.** *Let a function $f : \mathbb{R}^d \to \mathbb{R}$ be convex. For any $\boldsymbol{x}, \boldsymbol{y} \in \operatorname{dom} f$ with $\boldsymbol{z} = t\boldsymbol{x} + (1-t)\boldsymbol{y}$ and any collection of subgradients on points $\boldsymbol{z}(t)$, $\forall t \in [0,1]$ (i.e. for every point between $\boldsymbol{x}$ and $\boldsymbol{y}$):*

$$f(\boldsymbol{y}) - f(\boldsymbol{x}) = \int_{t=0}^{1} \Big(\partial f\big(\boldsymbol{z}(t)\big)\Big)^\top \boldsymbol{y} - \boldsymbol{x}\, dt$$

2.1.8 Extending convexity theorems to non-differentiable functions

We can now extend some previously stated theorems for the convexity of differentiable functions to functions that are not everywhere differentiable.

Lemma 2.1.11. *The following are equivalent:*

(a) f is convex

(b) (First Order Condition Analogue)

$$f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \boldsymbol{s}_x^\top (\boldsymbol{y} - \boldsymbol{x}) \tag{2.2}$$

(c) (Monotonicity of Subgradients)

$$(\boldsymbol{s}_y - \boldsymbol{s}_x)^\top (\boldsymbol{y} - \boldsymbol{x}) \geq 0,\ \forall \boldsymbol{x}, \boldsymbol{y} \text{ and any } \boldsymbol{s}_x \in \partial f(\boldsymbol{x}),\ \boldsymbol{s}_y \in \partial f(\boldsymbol{y}) \tag{2.3}$$

2.1.9 Strong Convexity

As we can observe, if a function is convex it means that we can bound every value of the function if it lies between two other given values of that function. There is one more – quite stronger – bound we can define for certain functions. The latter functions are the ones that we are going to call *strongly convex*. In strongly convex functions, values of the function intermediate to any pair of points $\boldsymbol{x}, \boldsymbol{y} \in \operatorname{dom} f$ can be bound with combination of the previously stated bound and an appropriate parabola (more or less what we would call a bowl).
Since there is not much consensus in the most popular texts on the matter and although some authors choose to speak of strong convexity in the context of twice differentiable, we chose to give a more liberal definition of strong convexity –following [24]– that is not immediately concerned with differentiability.

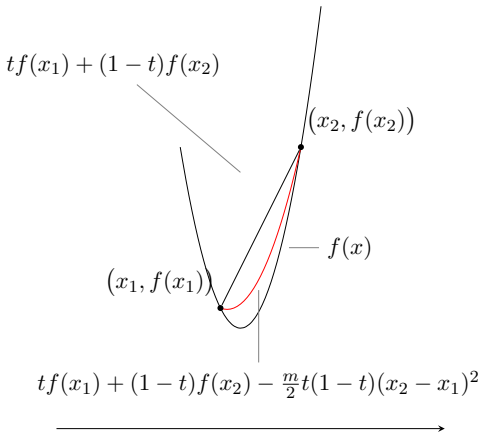


Figure 2.5: A strongly convex function is not only bounded by the straight black line but the red line as well

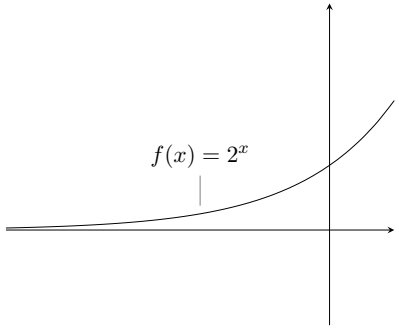


Figure 2.6: Function $f(x) = 2^x$ is not strongly convex as $f''(x)$ can be arbitrarily close to 0 for $x < 0$

Definition 2.1.17. (Strong Convexity) A function $f : \mathbb{R}^m \to \mathbb{R}$ is said to be a strongly convex function with coefficient (more precisely, modulus) m if the latter inequality holds:

$$f(t\boldsymbol{x} + (1-t)\boldsymbol{y}) \leq tf(\boldsymbol{x}) + (1-t)f(\boldsymbol{y}) - \frac{m}{2}t(1-t)\|x-y\|_2^2$$

is convex.

Lemma 2.1.12. *A function f is strongly convex if $f(\boldsymbol{x}) - \frac{m}{2}\|\boldsymbol{x}\|^2$ is convex.*

Remark 1. If $f : \mathbb{R}^d \to \mathbb{R}$ is twice differentiable, then f is strongly convex iff the next sentence holds for some $m > 0$:

$$\nabla^2 f \succeq mI$$

Lemma 2.1.13. *The following sentences are equivalent to f being strongly convex with modulus m (or m-strongly-convex):*

(a) $f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \boldsymbol{s}^\top(\boldsymbol{y}-\boldsymbol{x}) + \frac{m}{2}\|\boldsymbol{y}-\boldsymbol{x}\|_2^2$ (2.4)

(b) $(\boldsymbol{s}_y - \boldsymbol{s}_x)^\top(\boldsymbol{y}-\boldsymbol{x}) \geq m\|\boldsymbol{y}-\boldsymbol{x}\|_2^2$ (2.5)

2.1.10 Lipschitz Condition

One way to implicitly speak of functions whose (sub-)gradients are bounded is to use the definition of them being λ-Lipschitz or not, specifically when the slope of every one of its tangents is bounded by λ. Such a function is said to satisfy the Lipschitz Condition.

Definition 2.1.18. (Lipschitz Condition) Let a function $\boldsymbol{f} : \mathbb{R}^m \to \mathbb{R}^n$. The function is said to be λ-Lipschitz or to satisfy a Lipschitz Condition if there exists a constant $\lambda \in \mathbb{R}, \lambda > 0$ such that for any $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^m$:

$$\|\boldsymbol{f}(\boldsymbol{y}) - \boldsymbol{f}(\boldsymbol{x})\| \leq \lambda \|\boldsymbol{y} - \boldsymbol{x}\|$$

We complement the definition with two examples of one function that satisfies the Lipschitz Condition and is not everywhere continuous and one that is continuous everywhere and does satifisfy the Lipschitz Condition.

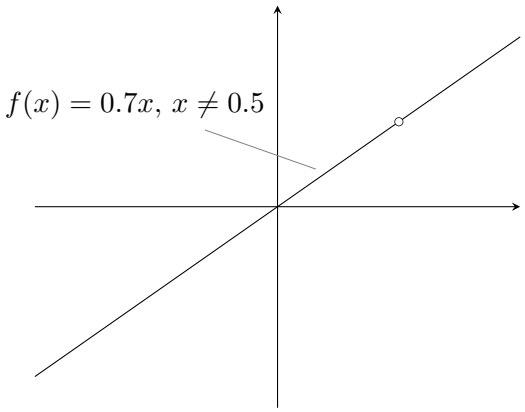


Figure 2.7: The straight line that has one point of incontinuity satisfies the Lipschitz Condition

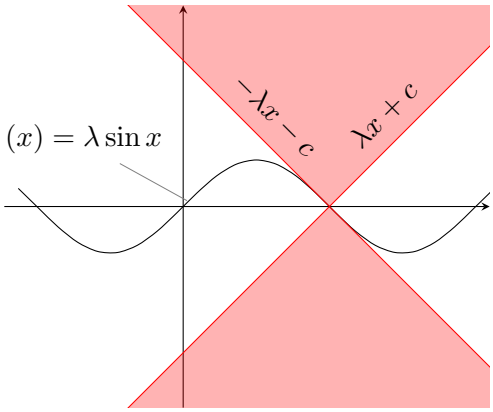


Figure 2.8: The tangents of $\lambda \sin x$ at any given point x never get into the red area defined by the two red lines defined for c's that correspond to given x.

2.2 Game Theory Basics

The term *game theory* is an almost euphemistic way of referring to the mathematical study of *interactions* between *self-interested* agents modeling settings that can span from armed-conflict to a game of rock-paper-scissors. Each agent's interest is modeled through *utility theory*. A player's *utility* is the measure that is used to quantify the compatibility of an outcome of the game with their interests.

A simple example

In order to help the reader gain some intuition, we start by stating a simple example:

Rock-Paper-Scissors Rock-paper-scissors is a game played between two players that lasts only one round. Each player has a selection of 3 moves/strategies (namely rock, paper and scissors). Each player picks their move and simultaneously reveal their choice. A player can win or lose and there can be only one winner.
In terms conventional to game theory:

- this is a 2-person game
- each one of the 3 moves is a *(pure) strategy* and can be thought of as three perpendicular unit vectors, $\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3$. (More precisely strategy is the player's all-or-nothing bet that the corresponding move will win)
- The rule by which we decide which one of the played moves wins can either be formulated as a matrix and in that case the matrix would be called a *payoff matrix*, or it could be thought of as a function, in which case it would be called a *payoff function*
- the fact that only one player can be a winner for every instance of the game makes the game a *zero-sum game*. Think about winning as having payoff 1 and losing as -1, in that case the sum of every player's payoffs equals to 0

The general case The latter may not always hold, in the sense that:

- a game may have more than two players. Such a game is obviously called a N-player game
- a player need not go *all-in* on each move but can rather place partial bets on several ones. Such a betting is called a *mixed strategy*.
- it may not always be true that the sum of the payoffs is equal to zero. Such games are called *general-sum games*

2.2.1 Normal-form Games

The *normal form* (also known as *strategic form* and *matrix form*) is a very common representation of games. This way of representing a game is its representation as the collection of the utility gained by every player for each and every combination of different decisions they are able to make. To be more precise, what we refer to as decision is more formally referred to as a (pure) *strategy*.
More formally, we move on to the mathematical definition of a normal-form game:

Definition 2.2.1 (Normal-form game)**.** A normal-form game is a tuple $\Gamma = \Gamma(\mathcal{N}, \mathcal{A}, \boldsymbol{u})$:

- a finite set $\mathcal{N}$ of N players, $\mathcal{N} = \{1, \cdots, N\}$
- a collection of finite action sets $\mathcal{A} = \{A_1, \cdots, A_N\}$ where A_i is the set of available pure *actions/strategies* available to player i
- a vector-valued function $\boldsymbol{u} = (u_1, \cdots, u_N)$ where the i-th entry is the utility function of player i, $u_i : \Delta(\mathcal{A}) \to \mathbb{R}$ that maps a distribution of actions to the utility/payoff this player enjoys .

As strategy vectors are probability vectors and probability vectors live in spaces commonly referred to as *probability simplices*, we should provide a formal definition of what a simplex is:

Definition 2.2.2 (m-simplex)**.** Consider an m-dimensional space and vectors $\boldsymbol{v}_0, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_m$ drawn from the former space. If the vectors $\boldsymbol{v}_1 - \boldsymbol{v}_0, \ldots, \boldsymbol{v}_m - \boldsymbol{v}_0$ are linearly independent, the convex hull of these vectors (namely, the intersection of all convex sets that contain the points $\boldsymbol{v}_0, \ldots, \boldsymbol{v}_m$) is said to be an m-simplex, Δ^m.

It may have become apparent that we can speak of the vectors $\boldsymbol{v}_0, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_m$ as the vertices of the simplex as well.

Definition 2.2.3. A simplex P is called a *probability simplex* or a *standard simplex* if

$$P = \{x = \theta_0 \boldsymbol{v}_0 + \ldots + \theta_m \boldsymbol{v}_m : \theta_i \geq 0, \sum_{i=0}^{m} \theta_i = 1\}$$

.

That means that all points of P lie in the convex set defined by the origin $\mathbf{0}$ and m unit vectors of $\mathbb{R}^m$. We note that one very popular way of denoting the probability simplices of the strategy-space is $\Delta(\mathcal{A})$.

2.2.2 Two-Player Zero-Sum Games

Definition The relationship between optimization and games enjoys an early start. This becomes apparent with the intimate relationship shared between two-player zero-sum games in normal-form and Linear Programming. In fact, Von Neumann's Minimax Theorem predates both the Duality Theorem of Linear Programming and Nash's Theorem itself.
This kind of games is probably the most studied one. It provides a gentle start into the theory of games and remains important to this day. A formal definition of a two-player zero-sum game is the following:

Definition 2.2.4 (Two-Player Zero-Sum Game)**.** A two-player zero-sum game in normal form is a tuple $\Gamma = \Gamma(\mathcal{N}, \mathcal{A}, \boldsymbol{u})$ with:

- the number of players $|\mathcal{N}| = 2$
- the strategy space $\mathcal{A}$ consisting of two finite sets of pure strategies for each player $\mathcal{A} = \{S_1, S_2\}$
- $u = u_2 = -u_1$ and $u = \boldsymbol{x}^\top A \boldsymbol{y}$ where $A \in \mathbb{R}^{n \times m}$ is the *payoff matrix* and vectors $\boldsymbol{x}, \boldsymbol{y}$ with $\boldsymbol{x} \in \Delta(S_1)$, $\boldsymbol{y} \in \Delta(S_2)$ being the possibly mixed strategy vectors of the *row/minimizing* player and the *column/maximizing* player respectively.

Linear Programming Here we demonstrate how a two-player zero-sum game can be formulated as a linear program and consecutively be "solved" in polynomial time. Without invoking Nash's theorem, by Strong Duality we can demonstrate that there exists a (possibly mixed) strategy profile $\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}$ from which no player can improve their utility by unilaterally deviating. We refer the interested reader to the standard reference for Linear Programming[29]. Let the two-person zero-sum game with a payoff matrix $A \in \mathbb{R}^{n \times m}$ and vectors $\boldsymbol{x} \in \Delta(S_1), \boldsymbol{y} \in \Delta(S_2)$ controlled by the minimizing player and the maximizing player respectively.
What the minimizing player[3] seeks to succeed is to minimize the value u that the maximizing player can take from them. This value has to be greater or equal than the value the maximizer[4] can accomplish with any one of their pure strategies which translates to the elementiwise vector inequality $\boldsymbol{x}^\top A \leq u$. Keeping in mind that $\boldsymbol{x}$ has to remain in the simplex, we quickly derive the following linear program:

$$
\begin{aligned}
& & \min \sum u & & & (2.6a)\\
& \text{s.t.} & \boldsymbol{x}^\top A \leq u & & & (2.6b)\\
\text{(P)} & & \boldsymbol{x}^\top \mathbf{1} = 1 & & & (2.6c)\\
& & x_a \geq 0, & & \forall a \in A & (2.6d)
\end{aligned}
$$

It is rather easy to verify that the dual of the latter program should be the following:

$$
\begin{aligned}
& & \max \sum z & & & (2.7a)\\
& \text{s.t.} & A\boldsymbol{y} \geq z & & & (2.7b)\\
\text{(D)} & & \boldsymbol{y}^\top \mathbf{1} = 1 & & & (2.7c)\\
& & y_b \geq 0, & & \forall b \in B & (2.7d)
\end{aligned}
$$

We observe that the dual program is precisely the (symmetrical) objective of the maximizing player; maximizing a value z that will be less or equal to every entry of the vector of payoffs $A\boldsymbol{y}$.
It is straightforward to see that that the primal program is feasible and bounded, hence the dual program will be feasible and bounded and the two programs will share the same value, i.e. let $\hat{\boldsymbol{x}}, \hat{u}$ and $\hat{\boldsymbol{y}}, \hat{z}$ be the respective solutions of the two programs then:

$$\hat{u} = \hat{\boldsymbol{x}}^\top A \hat{\boldsymbol{y}} = \hat{z}. \tag{2.8}$$

By the Weak Duality of Linear Programming we also have that:

$$\min_{\boldsymbol{x} \in \Delta(S_1)} \max_{\boldsymbol{y} \in \Delta(S_2)} \boldsymbol{x}^\top A \boldsymbol{y} \geq \max_{\boldsymbol{y} \in \Delta(S_2)} \min_{\boldsymbol{x} \in \Delta(S_1)} \boldsymbol{x}^\top A \boldsymbol{y}. \tag{2.9}$$

We can now observe that there is no unilateral deviation from $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}})$ that yields a utility better for one of the two players:

$$\hat{\boldsymbol{x}}^\top A \boldsymbol{y} \leq \hat{\boldsymbol{x}}^\top A \hat{\boldsymbol{y}} \leq \boldsymbol{x}^\top A \hat{\boldsymbol{y}}. \tag{2.10}$$

As we will see, this is the definition of the Nash equilibrium in two-person zero-sum games.
We conclude this part by stating Von Neumann's Minimax Theorem:

[3] or row player since they control the row of matrix A
[4] respectively, column player as they control the column of matrix A

Theorem 2.2.1 (Von Neumann's Minimax Theorem)**.** *For any two-player zero-sum game with a $n \times m$ payoff matrix A, there exist a value V – the value of the game – satisfying:*

$$\min_{\boldsymbol{x}\in\mathcal{X}} \max_{\boldsymbol{y}\in\mathcal{Y}} \boldsymbol{x}^\top A\boldsymbol{y} = V = \max_{\boldsymbol{y}\in\mathcal{Y}} \min_{\boldsymbol{x}\in\mathcal{X}} \boldsymbol{x}^\top A\boldsymbol{y} \tag{2.11}$$

2.2.3 Solution Concepts

Apart form the concept of the Nash equilibrium that we roughly touched upon in 2.2.2 we will discuss a number of different solution concepts starting from a formal definition of the Nash equilibrium.
Different solution concepts become useful depending on the setting we seek to analyze and the computational constraints we are subject. For example, in transportation networks inside a city, the governing body of the city does not trust self-interested drivers for the reduction of overall traffic and seeks to coordinate the *congestion game* played among car drivers through traffic lights, mass transportation, closing certain parts of the network etc.
With regards to computation of a "solution" to a game, we will not be able to compute a Nash equilibrium in polynomial time as seen in 2.2.2.

2.2.3.1 Pure & Mixed Nash Equilibria

The first solution concept that we will discuss is that of the *pure Nash equilibrium*. A pure Nash equilibrium is a strategy profile consisting of only pure strategies from which no player can improve their utility through unilaterally deviating.
Not all games have a pure Nash equilibrium. One that has is shown in table 2.1. The Nash strategy is for both players to play action A.

		Player 2	
		A	B
Player 1	A	$(2, -2)$	$(3, -3)$
	B	$(1, -1)$	$(0, 0)$

Table 2.1: A zero-sum game with a pure Nash equilibrium

The formal definition of a pure Nash equilibrium is the following:

Definition 2.2.5 (Pure Nash equilibrium)**.** A pure strategy profile $a = (a_1, \ldots, a_n)$ of pure strategies $a_i \in A_i$ in a utility maximization game is a Nash equilibrium (PNE) if for for every player $i \in [n]$ and every pure strategy $a'_i \in A_i$ the following inequality holds:

$$u_i(a) \geq u_i(a'_i; a_{i-1}) \tag{PNE}$$

Next, we consider *mixed* Nash equilibria which are guaranteed to always exist [42].

Definition 2.2.6 (Nash equilibrium)**.** A product-distribution $x = (x_1, \ldots, x_n)$ of distributions x_i on A_i of strategy profiles in a utility maximization game is a Nash equilibrium (NE) if for for every player $i \in [n]$ and every pure strategy $a'_i \in A_i$ the following inequality holds:

$$\mathbb{E}_{a\sim x}\left[u_i(a)\right] \geq \mathbb{E}_{a\sim x}\left[u_i(a'_i; a_{i-1})\right] \tag{NE}$$

For this case we give the example of the game of *matching pennies*. Two players, Max and Rose, have a penny (coin) and pick either one of its side. Max wins when he draws the same side as Rose, i.e.when he matches his side with hers; Rose on the other hand wins when she draws the side that Max will not. The payoff matrix is given in table 2.2. The unique Nash equilibrium is attained when both players pick each side of the coin uniformly at random, i.e. each with probability $\frac{1}{2}$.

		Player 2	
		A	B
Player 1	A	$(1,-1)$	$(-1,1)$
	B	$(-1,1)$	$(1,-1)$

Table 2.2: The game of matching pennies

2.2.3.2 Correlated and Coarse-Correlated Equilibria

The next solution concept is due to [3]. It offers an alternative that steers off the complexity of computing a Nash equilibrium along. Players are able to coordinate their actions with respect to a source of randomness that is external to them. As such, we are not constrained to product distribution of randomized strategies, but rather on a distribution on every pure strategy combination. Formally, a correlated distribution belongs in the simplex defined on $\mathcal{A}$, $\Delta(\mathcal{A})$ and is not constrained in the product of simplices of every player's actions $\Delta(A_1)\times\cdots\times\Delta(A_n)$.

Definition 2.2.7 (Correlated equilibrium)**.** A distribution x on the set $A_1\times\cdots\times A_n$ of strategy profiles in a utility maximization game is a correlated equilibrium (CCE) if for for every player $i\in[n]$ and every pure strategy $a_i'\in A_i$ the following inequality holds:

$$\mathbb{E}_{a\sim x}\left[u_i(a)\right]\geq\mathbb{E}_{a\sim x}\left[u_i(a_i';a_{i-1})|a_i\right] \tag{CE}$$

This solution concept has a very intuitive description. Instead of allowing players to randomize their strategies independently, there exists a benevolent dictator that is trusted by all players. The dictator samples a pure strategy profile a from x and reveals every a_i privately to player i. Given that player i has observed the suggestion a_i, they have no incentive to deviate given that everyone else plays according to a. Moreover, Nash equilibria are also a special case of correlated equilibria, hence CE are always guaranteed to exist.
A very natural example is the game played between two drivers on a road conjunction. The players can select to cross the road (GO) or stay put (STOP). When they both cross the road, a terrible accident takes place taking a toll on both players' utilities. If the one of them stops and the other one passes, they get a utility of 0 and 1 respectively. The payoff matrix is seen in table 2.3. This game has two pure strategy Nash equilibria (STOP,GO), (GO,STOP) and one mixed Nash equilibrium where both players play STOP with probability $\frac{100}{101}$. In general, this is by no means an favorable state of affairs. It is only through the use of a traffic light that a much preferable situation rises. Namely one where only one of the two drivers is instructed to go each time and this driver is picked uniformly at random.

		Driver 2	
		GO	STOP
Driver 1	GO	$(0,0)$	$(0,1)$
	STOP	$(1,0)$	$(-100,-100)$

		Driver 2	
		GO	STOP
Driver 1	GO	0	1/2
	STOP	1/2	0

Table 2.3: A game between two drivers & and a CE

One more solution concept we define is that of the Coarse Correlated Equilibrium. Similar to a correlated equilibrium, in a coarse correlated equilibrium there will be a suggested action a_i when a is drawn from x. But a_i will only be required to be a best response in expectation *before* observing the instruction to play a_i. The player must decide whether they will play according to the instruction a_i before seeing it. In the case that they were allowed to observe it before deciding, there could very well exist a best-response other than a_i. Formally:

Definition 2.2.8 (Coarse-correlated equilibrium)**.** A distribution x on the set $A_1 \times \cdots \times A_n$ of strategy profiles in a utility maximization game is a coarse-correlated equilibrium (CCE) if for for every player $i \in [n]$ and every pure strategy $a_i' \in A_i$ the following inequality holds:

$$\mathbb{E}_{a\sim x}\left[u_i(a)\right] \geq \mathbb{E}_{a\sim x}\left[u_i(a_i'; a_{i-1})\right] \tag{CCE}$$

An example is given in tables 2.4. Let x be a distribution as the one seen on the right table and a be one of its realizations. If both players decide to commit to instruction a without seeing it, the expected payoff they will get is $\frac{1}{3}\cdot(1+1-1.1)=0.3$. Now, let one player play according to a and the other one deviate from a. If the deviating player plays actions A or B, they will experience an expected utility of 0 while if they play C they will experience a utility $-\frac{1.1}{3}$. No convex combination of 0 and $\frac{-1.1}{3}$ is better than 0.3 hence no player has an incentive to deviate from a without seeing it.

		Player 2		
		A	B	C
Player 1	A	$(1,1)$	$(-1,-1)$	$(0,0)$
	B	$(-1,-1)$	$(1,1)$	$(0,0)$
	C	$(0,0)$	$(0,0)$	$(-1.1,-1.1)$

		Player 2		
		A	B	C
Player 1	A	1/3		
	B		1/3	
	C			1/3

Table 2.4: A general-sum game and a CCE

2.2.3.3 Approximate Nash equilibrium

Finally, we define the notion of an approximate Nash equilibrium that becomes relevant when we try to compute Nash equilibria in games that are not bi-linear. An ϵ-apprxoimate Nash strategy yields an outcome that any deviation is only better by at most a margin of ϵ.

Definition 2.2.9 (ϵ-approximate Nash equilibrium)**.** A mixed strategy profile $x = (x_1, \ldots, x_n)$ in a utility maximization game is an ϵ-approximate Nash equilibrium (ϵ-NE) if for for every

player $i \in [n]$ and every deviation $x_i' \in \Delta(A_i)$ the following inequality holds:

$$u_i(x) \geq u_i(x_i'; x_{i-1}) - \epsilon \tag{ϵ-NE}$$

2.2.4 The Nash Theorem

The proof of the existence of Nash equilibria in all finite games[42] is cornerstone not only to game theory but also to an array of different scientific fields as well. It most certainly deserves discussion on its own, and this is something we will do now.

Preliminaries In order to start talking about equilibrium points in games we need to familiarize ourselves with a couple of tools. In order to prove the existence of an equilibrium point in a finite N-person game we will need the Brouwer Fixed Point Theorem which in turn needs Sperner's Lemma.
We will use an analogy from a common practice in building wire-frames in order to get into graph theory gently. For this purpose a graph's edge will be represented by a stick and a joint made out of playdough that will stand in the place of a graph's vertex.

On Sticks, Colored Playdough, and Triangles (Sperner's Lemma)

One dimension and Two dimensions We are given n buckets of playdough – each bucket's content is colored uniquely – and as many identical sticks as we like. Using playdough to construct joints we connect sticks together. The joints are what we refer to as nodes or vertices in graph theory and the sticks are called edges. The color of the playdough will remain named as such.
Let us imagine a simple recipe for constructing however large structures starting from a single dimension structure. Consider a line consisting by a series of edges connected consecutively by edges. We pick 2 different colors to label with the rightmost and the leftmost vertices respectively. Now, we can color the rest of the vertices using the same 2 colors.

Lemma 2.2.2 (Sperner's lemma in a single dimension)**.** *Given a set of vertices and edges that alternate forming a straight line with one vertex on each end of the line and two distinct colors; if the two ends are colored differently and the intermediate vertices can be colored only by the same two colors used for the two ends, then the number of color changes between the vertices of line will be an odd number.*

Of course, the latter are trivial. We shall now state the rule for building greater structures as promised.

Top-Down Approach Form a large triangle consisting of 3 big sticks and 3 distinctly colored pieces of playdough. We will now break every edge to as many pieces we may in order to be able to create more connections within the initial triangle. But there are some rules to that; every time you break one of the exterior edges you have to stick them back together using playdough of the same color found on its ends. Also, every time you break one of the exterior sticks you have to make sure that no other polygon than a triangle is formed, you will have to break any other edge you find necessary while respecting the rule that every node of each side of the exterior triangle is colored with either one of the colors found on the two respective ends.

The top-down approach may – in all honesty – seem a bit cluttered, so let's try a bottom-up one.

Bottom-Up Approach Consider a new bucket of playdough. The playdough is colored gray and we will consider it as a colorless placeholder for replacing it later with one that is colored. We construct a simple triangle with 3 sticks and gray playdough. We start building up from the initial simple triangle by making sure the outer shape remains a triangle by either adding two new sticks and a new joint. When and if - as required - the outer edges still form a triangle we may stop the building phase and initiate the coloring phase. For the coloring phase we pick a unique color for every one of the vertices of the outermost triangle. We color every joint of the outside triangle – color as in replace the gray plaster with a colored one – with strictly either one of the colors found on one of the two ends. The interior vertices need to be colored as well, but we can do so randomly.

We remind the reader that such a wire-frame made of playdough and sticks is a graph $G(V, E)$, whose vertices V are the joints made of playdough and its E are the sticks we used. The color of the playdough is a label we assign to every vertex $v \in V$. The colors can be equivalently thought of as an integer.

Definition 2.2.10. A triangle is called distinguished if its vertices are colored by all 3 colors.

Lemma 2.2.3 (Sperner's lemma on a planar graph)**.** *Every properly colored triangular subdivision (or triangulation) of a triangle has an odd number of 3-colored triangles.*

More than two dimensions In order to move on to more than two dimensions one should seek for a generalization of the notion of a triangle. The generalization in question is a simplex. In the 0-dimensional case a simplex reduces to a point, in the 1-dimensional case a simplex reduces to a line, in the 2-dimensional case a simplex is a triangle, in the 3-dimensional case a simplex is a tetrahedron (a pyramid whose every side is a triangle) and lastly a 4-dimensional space is called a 5-cell (as it consists of 5 tetrahedra). Some authors symbolize m-dimensional simplices (or m-simplices) with Δ^m. We also define the "faces" of a given simplex as:

Definition 2.2.11 (k-face)**.** A k-face of an m-simplex, is merely a k-simplex defined by a subset of the $m + 1$ vertices of the initial m-simplex such that $k \leq m$.

We name P the m-simplex defined by $\boldsymbol{v}_0, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_m$.

Sperner coloring on an m-simplex Let $\mathcal{S}$ be a set of vertices and edges lying inside an m-simplex including vertices and the edges of the m-simplex itself. The points of $\mathcal{S}$ define smaller m-simplices that divide the larger simplex into disjoint spaces. $\mathcal{S}$ is also called a triangulation of P. Keeping in mind that colors are refered to uniquley as color $0, 1, 2, \ldots, m$, a valid Sperner coloring is a coloring of every vertex for which the following hold:

- Every one of the $m + 1$ vertices that define the m-simplex get a unique color 1 through $m + 1$. (With no loss of generality and for the sake of clarity, the vertex v_i gets the i-th color.)
- Vertices of $\mathcal{S}$ that lie on a k-simplex defined by a subset of $\{v_0, \ldots, v_m\}$ can only receive colors whose numbers coincide with the indices of the $k + 1$ vertices that define the k-simplex in question. (e.g. vertices lying on a 4-simplex defined by $\{v_0, v_2, v_4, v_5\}$ can only be colored with colors $\{0, 2, 4, 5\}$)

Lemma 2.2.4 (Sperner's lemma on an m-simplex)**.** *Given an m-simplex P, a division $\mathcal{S}$ of P that retains a valid Sperner coloring with $m+1$ colors, there exists an odd number of m-simplices inside P that carry all $m+1$ colors.*

Brouwer Fixed Point Theorem

Theorem 2.2.5 (Brouwer Fixed Point Theorem)**.** *Let a set $K \subseteq \mathbb{R}^d$ be convex, closed and bounded. A function $\boldsymbol{T} : K \to K$ that is continuous has at least one $\boldsymbol{x} \in K$ such that $\boldsymbol{T}(\boldsymbol{x}) = \boldsymbol{x}$.*

We prove the Brouwer theorem on $K = \Delta^m$ by means of the Sperner Lemma. We show that there is a relation between entries of $\boldsymbol{Q} \in K$ and $\boldsymbol{T}(\boldsymbol{Q})$ that can be considered as a valid Sperner coloring. We will talk of k-faces of Δ^m as being constituted by k vectors v_r vectors or the indices of these vectors interchangeably.

Proof. We consider the case that $K = \Delta^m$. Since every $\boldsymbol{Q} = (q_1, \ldots, q_m) \in K$ and $\boldsymbol{T}(\boldsymbol{Q}) = (T_1(\boldsymbol{Q}), \ldots, T_m(\boldsymbol{Q})) \in K$ we know that:

$$\sum_i q_i = \sum_i T_i(\boldsymbol{Q}) = 1$$

By means of the pigeon hole principle we can conclude that there exists at least one index j such that $q_j \geq T_j(\boldsymbol{Q})$.
If $\boldsymbol{Q}$ happens to lie on a k-face of the larger Δ^m simplex, we know that there are several non-zero $q_i \geq 0$ with $i \in I$, where I signifies the indices of the vectors v_r that constitute the k-face in question. Apparently, for the rest q_i s.t. $i \notin I$, $q_i = 0$.
Combining the two latter propositions, since the entries q_i that refer to the vertices perpendicular to the current k-face are zero, we can pick a j s.t. $q_j \geq T_j(\boldsymbol{Q})$ from only the set of indices that constitute the current k-face.
Picking an index from those that correspond to the set of vectors that construct a k-face is directly equivalent to picking a color that is compatible to a Sperner coloring – remember that a vertex's color can only come from the set of colors of the vertices that constitute the k-face in question.
We firstly use this rule on the vertices of Δ^m as to assign a number 0 through $m+1$ to each color.
Picking arbitrarily many points Q_i we can create an arbitrarily small triangulation $\mathcal{S}$ of P that has a valid Sperner coloring. Thanks to the Sperner lemma for the multi-dimensional case, we know that there is an odd number of cells (smaller m-simplices) of Δ^m such that every vertex is uniquely colored. But, having a cell being uniquely colored means that for every vertex $Q^{(r)}, r \in \{0, 1, \ldots m\}$:

$$q_i^{(r)} \geq T_i(\boldsymbol{Q}^{(r)})$$

If all $Q^{(r)}$ can come infinitesimally close to each other depending on the number of initial points in $\mathcal{S}$, we can conclude that there will be such a point $\boldsymbol{Q}^* = (q_1^*, \ldots, q_m^*)$, infinitesimally close to the points $\boldsymbol{Q}^{(r)}$ of the distinguished simplex that since $\sum_i q_i^* = 1 = \sum_i T_i(\boldsymbol{Q}^*)$ and $q_i \geq T_i(\boldsymbol{Q}^*)$ it must inevitably hold that $\boldsymbol{Q}^* = \boldsymbol{T}(\boldsymbol{Q}^*)$ ■

2.2.4.1 Nash Equilibrium

We will consider a game in normal-form with finite players and finite moves for each player. Just to be clear, a lowercase Latin letter (e.g. i, j, k) signifies one out of n players and a

lowercase Greek letter signifies one of m moves. Each player's strategy can be represented by a vector $\boldsymbol{x}_i = (x_{i\alpha}, x_{i\beta}, \ldots, x_{i\kappa}, \ldots) \in \Delta^{m-1} \subset \mathbb{R}^d$ whose every entry i holds the player's betting on the i-th move. It may be obvious that we consider that $\sum_\lambda x_{i\lambda} = 1, x_{i\lambda} \geq 0$. More formally:

Definition 2.2.12 (Finite Game)**.** [5] A finite n-person game is:

- a *general-sum game*
- consisting of n players
- each player i is associated with an m-dimensional *strategy vector* $\boldsymbol{x}_i \in \Delta^m$. A *mixed strategy* vector is a vector whose entries sum to one. *Pure strategies* (i.e. $\boldsymbol{x}_i = \boldsymbol{e}_{i\kappa}$ for some $\kappa \in \{1, \ldots, m\}$) are special cases of mixed strategies.
- each player i is associated with a *payoff function* $p_i : \underbrace{\Delta^m \times \cdots \times \Delta^m}_{n} \to [0, 1]$. Payoff, the scalar value of a payoff function, could be referred to as *utility* as well.

We will also define some auxiliary observations and notations; the κ-th pure strategy of the i-th player will be denoted by $\boldsymbol{e}_{i\kappa}$ and a mixed strategy $\boldsymbol{x}_i = (x_1, \ldots, x_m)$ can equivalently be written as $\sum_\kappa x_{i\kappa}\boldsymbol{e}_{i\kappa}$. Also, $\boldsymbol{s}$ will be the tuple of all mixed strategies $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m)$ and a swapping of i-th player's strategy vector $\boldsymbol{x}_i$ with a new one $\boldsymbol{T}_i$ will conveniently be noted as $(\boldsymbol{s}; \boldsymbol{T}_i)$. Successive substitutions $((\boldsymbol{s}; \boldsymbol{T}_i); \boldsymbol{r}_j)$ will be indicated as $(\boldsymbol{s}; \boldsymbol{T}_i; \boldsymbol{r}_j)$. A pure strategy κ is said to be *used* by player i when the coefficient $x_{i\kappa} > 0$. We also define $p_{i\kappa}$ as replacing player i's mixed strategy with the pure strategy $e_{i\kappa}$.

Theorem 2.2.6 (Nash Theorem[42])**.** *Every finite N-person game has a Nash Equilibrium.*

In plain English, the latter means that an equilibrium point is an n-tuple of mixed strategies such that no player can unilaterally (i.e. by themselves) increase their payoff if the strategies of all other players are held fixed.

Theorem 2.2.7 (Existence of Equilibrium Points)**.** *Every finite general-sum game has an equilibrium point.*

Proof. The proof we selected relies on the Brouwer Fixed Point Theorem. We will define a function $\boldsymbol{T} : \boldsymbol{s} \to \boldsymbol{s}'$, that maps a strategy $\boldsymbol{s}$ to a new one $\boldsymbol{s}'$.
First off, we observe that p_i is linear with respect to every strategy vector $\boldsymbol{x}_i$ and n-linear with respect to all n of them. Consequently:

$$\max_{\forall \boldsymbol{r}_i \in \Delta^m} p_i(\boldsymbol{s}; \boldsymbol{r}_i) = \max_\kappa p_i(\boldsymbol{s}; \boldsymbol{e}_{i\kappa})$$

Then we let a function $\phi_{i\kappa}$ be a *continuous* function used as an indicator of how much would a player's payoff would improve if they were to play a pure strategy κ instead (deteriorating is as good as no change at all):

$$\phi_{i\kappa} = \max\{0, p_{i\kappa}(\boldsymbol{s}) - p_i(\boldsymbol{s})\}$$

With the help of the ϕ_i functions we move on to define a function $\boldsymbol{T}(\boldsymbol{s}) = \boldsymbol{s}'$ that will perform an one-step payoff increase for every player's strategy (if no increase is possible, payoff stays the

[5] In a more general case, not all players need to have the same number of moves but this will not concern us at the moment and we will not lose any of the generality from the conclusions we are going to draw.

same). The function in question can be seen as a two step process; namely it performs a step in every dimension from $x_{i\kappa}$ towards the pure strategy that improves the payoff for the given player and then a scaling of every dimension in order to keep the sum of all the coefficients $\sum_\kappa x_{i\kappa} = 1$. So for every strategy $\boldsymbol{x}$, function $\boldsymbol{T}$ outputs a new one $\boldsymbol{x}'$ s.t.:

$$x_i' = \overbrace{(x_i + \sum \phi_{i\kappa} \boldsymbol{e}_{i\kappa})}^{step} \cdot \overbrace{\frac{1}{1 + \sum_\kappa \phi_{i\kappa}}}^{scaling}$$

If $\boldsymbol{s}' = \boldsymbol{s}$ stays the same under $\boldsymbol{T}$ it means that no player can shift to a strategy that will make their payoff better which makes $\boldsymbol{s}$ an equilibrium point. Conversely, if $\boldsymbol{s}$ is an equilibrium point we need to observe that all $\phi_\cdot(\cdot)$ turn to zero which makes $x_i' = (x_i + 0)\frac{1}{1} = x_i$
Since $\boldsymbol{T}$ is continuous, bounded and closed on $S = \underbrace{\Delta^m \times \cdots \times \Delta^m}_{n}$ and S is convex, thanks to the Brouwer Fixed Point Theorem, we can decide that there exists an equilibrium point, or $\boldsymbol{T}(\boldsymbol{x}) = \boldsymbol{x}$ for every finite game. ■

2.2.5 Potential games

A class of games that enjoys widespread popularity and has been thoroughly studied is the class of *potential games*. Potential games are games where a single player's deviation affects the utility of all players. Moreover, there exists a single function that can track the deviation of any player's utility given their strategy deviation.

Definition 2.2.13 (Potential Game)**.** Let a normal-form game $\Gamma = \Gamma(\mathcal{N}, \mathcal{A}, \boldsymbol{u})$. The game Γ is said to be an *(exact)* potential game if there exists a function $\Phi : \mathcal{A} \to \mathbb{R}$ such that for every player i, any strategy profile $\boldsymbol{a} \in \mathcal{A}$ and any two different strategies $a_i', a_i'' \in A_i$:

$$\Phi(a_i', \boldsymbol{a}_{-i}) - \Phi(a_i'', \boldsymbol{a}_{-i}) = u(a_i', \boldsymbol{a}_{-i}) - u(a_i'', \boldsymbol{a}_{-i}) \tag{2.12}$$

Definition 2.2.14 (Weighted Potential Game)**.** Let a normal-form game $\Gamma = \Gamma(\mathcal{N}, \mathcal{A}, \boldsymbol{u})$. The game Γ is said to be a *weighted* potential game if there exists a function $\Phi : \mathcal{A} \to \mathbb{R}$ and a vector $\boldsymbol{w} \in \mathbb{R}^N, w_i > 0$ such that for every player i, any strategy profile $\boldsymbol{a} \in \mathcal{A}$ and any two different strategies $a_i', a_i'' \in A_i$:

$$\Phi(a_i', \boldsymbol{a}_{-i}) - \Phi(a_i'', \boldsymbol{a}_{-i}) = w_i\big(u(a_i', \boldsymbol{a}_{-i}) - u(a_i'', \boldsymbol{a}_{-i})\big) \tag{2.13}$$

Definition 2.2.15 (Ordinal Potential Game)**.** Let a normal-form game $\Gamma = \Gamma(\mathcal{N}, \mathcal{A}, \boldsymbol{u})$. The game Γ is said to be an *ordinal* potential game if there exists a function $\Phi : \mathcal{A} \to \mathbb{R}$ such that for every player i, any strategy profile $\boldsymbol{a} \in \mathcal{A}$ and any two different strategies $a_i', a_i'' \in A_i$:

$$\Phi(a_i', \boldsymbol{a}_{-i}) - \Phi(a_i'', \boldsymbol{a}_{-i}) > 0 \Leftrightarrow u(a_i', \boldsymbol{a}_{-i}) - u(a_i'', \boldsymbol{a}_{-i}) > 0 \tag{2.14}$$

2.3 Short note on optimization

Before discussing min-max optimization we need to discuss some rudimentary optimization theory concepts. We consider a template *constrained minimization problem* (MP) with a convex feasible set $\mathcal{X}$. We define the tangent and normal cones at a given feasible point in order to define first-order stationarity in the constrained setting.
Next, we will define a template min-max optimization problem that is known in optimization literature as a constrained *saddle-point problem* (SP) and discuss some notions such as the monotonicity of an operator that will prove useful in the discussion of the analysis of some algorithms.

2.3.1 Optimization

Let f be an *objective function* that we seek to *minimize* for a variable $\boldsymbol{x}$ that is constrained inside the feasibility set $\mathcal{X}$. We can describe this problem succinctly as:

$$\min_{\boldsymbol{x}\in\mathcal{X}} f(\boldsymbol{x}) \tag{MP}$$

When $\mathcal{X} = \mathbb{R}^d$ we say that the problem is *unconstrained*. Unless certain properties hold for the objective function (e.g. convexity), even finding a local minimum of the objective function f can be NP-hard [41]. What we can only hope for is stationary points. More precisely, *approximate* stationary points. But, before defining stationarity in a constrained optimization problem, the definitions of the *tanget cone* and the *normal cone* are in order. We note that a *cone* is a set S such that for all $s \in S \subseteq \mathbb{R}^d$ and every scalar $\lambda \geq 0$ then also λs belongs to S.

Tangent Cone The tangent cone is the set that contains every *tangent vector* of $\mathcal{X}$ to a given point $\hat{\boldsymbol{x}}$. A *tangent vector* is defined as any $\boldsymbol{\delta}$ for which $\boldsymbol{\delta} = \lim_{k\to\infty} \frac{\boldsymbol{z}_k - \hat{\boldsymbol{x}}}{t_k}$.

$$T_{\mathcal{X}}(\hat{\boldsymbol{x}}) = \left\{ \boldsymbol{\delta} \;\middle|\; \boldsymbol{\delta} = \lim_{k\to\infty} \frac{\boldsymbol{z}_k - \hat{\boldsymbol{x}}}{t_k},\ \forall k\ \boldsymbol{z}_k \in \mathcal{X} \right\} \tag{2.15}$$

Normal Cone The normal cone is the set that contains all vectors that are negatively correlated with every tangent vector, i.e. all vectors $\boldsymbol{\delta}$ for which $\langle \boldsymbol{\delta}, \boldsymbol{\zeta} \rangle \leq 0,\ \boldsymbol{\zeta} \in T_{\mathcal{X}}(\hat{\boldsymbol{x}})$.

$$N_{\mathcal{X}}(\hat{\boldsymbol{x}}) = \left\{ \boldsymbol{\delta} \;\middle|\; \langle \boldsymbol{\delta}, \boldsymbol{\zeta} \rangle \leq 0, \forall \boldsymbol{\zeta} \in T_{\mathcal{X}}(\hat{\boldsymbol{x}}) \right\} \tag{2.16}$$

Effectively, we use these sets (the tangent and normal cone) in order to develop a constrained optimization first-order stationarity condition that is equivalent to the stationarity condition $\nabla f(\boldsymbol{x}) = 0$ of unconstrained optimization.
Finally, a first-order stationary point is a point $\hat{\boldsymbol{x}}$ such that every direction in the tangent cone is negatively correlated with the negative gradient of f. Intuitively, starting from $\hat{\boldsymbol{x}}$ there is no direction that we can follow that will lead to a new *feasible point* with which direction the negative of the gradient is forming an acute angle. If there were such a direction, this would lead to a reduction in the value of the objective function for a sufficiently small step towards said direction.

Definition 2.3.1 (FOSP)**.** Let $f : \mathcal{X} \to \mathbb{R}$ be a differentiable functions and $\hat{\boldsymbol{x}} \in \mathcal{X}$. The point $\hat{\boldsymbol{x}}$ is a first-order stationary point of f:

$$\langle -\nabla f(\hat{\boldsymbol{x}}), \boldsymbol{x} - \hat{\boldsymbol{x}} \rangle \leq 0, \quad \forall \boldsymbol{x} \in \mathcal{X} \tag{2.17}$$

or equivalently:

$$-\nabla f(\hat{\boldsymbol{x}}) \in N(\hat{\boldsymbol{x}}). \tag{2.18}$$

Further, we relax the latter condition for points where there is no significant decrease in the objective function. This is the main definition of stationarity used in the analysis of iterative optimization methods.

Definition 2.3.2 (ϵ-FOSP)**.** Let $f : \mathcal{X} \to \mathbb{R}$ be a differentiable functions and $\hat{\boldsymbol{x}} \in \mathcal{X}$. The point $\hat{\boldsymbol{x}}$ is an ϵ-approximate first-order stationary point of f:

$$\langle -\nabla f(\hat{\boldsymbol{x}}), \boldsymbol{x} - \hat{\boldsymbol{x}} \rangle \leq \epsilon, \quad \forall \boldsymbol{x} \in \mathcal{X}, \|\boldsymbol{x} - \hat{\boldsymbol{x}}\| = 1 \tag{2.19}$$

or equivalently:

$$-\nabla f(\hat{\boldsymbol{x}}) \in N_{\mathcal{X}}(\hat{\boldsymbol{x}}) + \epsilon B_2(1) \tag{2.20}$$

where $B_2(1)$ is a d-dimensional ball with a radius measured in ℓ_2-norm equal to 1.

2.3.2 Saddle-Point Problems

We let $\mathcal{X} \subseteq \mathbb{R}^n$, $\mathcal{Y} \subseteq \mathbb{R}^m$ be two compact convex feasible regions and $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ a (possibly nonconvex-nonconcave) function. The following problem is referred to as a *saddle-point* problem with f being the value function:

$$\min_{\boldsymbol{x} \in \mathcal{X}} \max_{\boldsymbol{y} \in \mathcal{Y}} f(\boldsymbol{x}, \boldsymbol{y}). \tag{SP}$$

In the case that f denotes the utility function of the maximizing agent in two-person zero-sum game, the saddle-point problem is equivalent to computing a Nash equilibrium of the corresponding game.
Solutions to a saddle-point point problem can be represented as solutions to the Stampacchia variational inequality:

$$\langle F(\hat{\boldsymbol{z}}), \boldsymbol{z} - \hat{\boldsymbol{z}} \rangle \tag{SVI}$$

where $\boldsymbol{z}^\top = (\boldsymbol{x}^\top, \boldsymbol{y}^\top)$ and $F(\boldsymbol{z})$ is the operator $F(\boldsymbol{z}) = (\nabla_{\boldsymbol{x}} f(\boldsymbol{x}, \boldsymbol{y}), -\nabla_{\boldsymbol{y}} f(\boldsymbol{x}, \boldsymbol{y}))$.
In the special case that f is convex-concave, the (SVI) is equivalent to the Minty variational inequality:

$$\langle F(\boldsymbol{z}), \boldsymbol{z} - \hat{\boldsymbol{z}} \rangle. \tag{MVI}$$

It should be noted that convexity-concavity of f is only a sufficient condition for the Minty variational inequality to hold and that the inequality can hold in a nonconvex-nonconcave f. In fact, [40] demonstrate how to exploit the (MVI) to prove last-iterate convergence to a saddle-point/Nash equilibrium for a certain first-order algorithm.
Further, we say that an operator F is *monotone* when for any two points $\boldsymbol{z}, \boldsymbol{z}' \in \mathcal{X} \times \mathcal{Y}$:

$$\langle F(\boldsymbol{z}) - F(\boldsymbol{z}'), \boldsymbol{z} - \boldsymbol{z}' \rangle \leq 0. \tag{2.21}$$

On the Duality gap The duality gap is the value Gap defined as :

$$\text{Gap} = \min_{\boldsymbol{x} \in \mathcal{X}} \max_{\boldsymbol{y} \in \mathcal{Y}} f(\boldsymbol{x}, \boldsymbol{y}) - \max_{\boldsymbol{y} \in \mathcal{Y}} \min_{\boldsymbol{x} \in \mathcal{X}} f(\boldsymbol{x}, \boldsymbol{y}) \tag{Duality Gap}$$

and is always non-negative be the weak duality theorem.
It is rather favorable to have a duality gap equal to zero in optimization and game theory. A zero duality gap in linear programming allows us to use the dual of a given linear program that can be potentially easier to solve by having less variables all the while having the same optimal value as the primal.
By theorem 2.2.1 we are guaranteed to have a zero duality-gap in bi-linear functions (hence, linear programming and two-player zero-sum games). But, it turns out that a zero duality-gap exists in every convex-concave $f(\boldsymbol{x}, \boldsymbol{y})$. This theorem is due to Maurice Sion[6][58].

[6]Maurice Sion was a mathematician born in Skopje (now in North Macedonia). He came from a Ladino-speaking Sephardic Jewish family and spent his early years in Salonica, Izmir, and Beirut before migrating to Canada.

Theorem 2.3.1 (Sion's Minimax Theorem)**.** *Let $\mathcal{X} \subset \mathbb{R}^n$ and $\mathcal{Y} \subset \mathbb{R}^m$ be compact convex sets and a real-valued function $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ for which:*

- *$f(\cdot, \boldsymbol{y})$ is lower-semicontinuous and quasi-convex for every fixed $\boldsymbol{y}$*
- *$f(\boldsymbol{x}, \cdot)$ is upper-semicontinuous and quasi-concave for every fixed $\boldsymbol{x}$.*

Then:

$$\inf_{\boldsymbol{x}\in\mathcal{X}} \sup_{\boldsymbol{y}\in\mathcal{Y}} f(\boldsymbol{x}, \boldsymbol{y}) = \sup_{\boldsymbol{y}\in\mathcal{Y}} \inf_{\boldsymbol{x}\in\mathcal{X}} f(\boldsymbol{x}, \boldsymbol{y}). \tag{2.22}$$

We can also deduce the following corollary for convex-concave functions:

Corollary 2.3.1. *Let $\mathcal{X} \subset \mathbb{R}^n$ and $\mathcal{Y} \subset \mathbb{R}^m$ be compact convex sets and function $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ be a continuous function that is convex-concave, i.e.:*

- *$f(\cdot, \boldsymbol{y})$ is convex for every fixed $\boldsymbol{y}$*
- *$f(\boldsymbol{x}, \cdot)$ is concave for every fixed $\boldsymbol{x}$.*

Then:

$$\min_{\mathcal{X}} \max_{\mathcal{Y}} f(\boldsymbol{x}, \boldsymbol{y}) = \max_{\mathcal{Y}} \min_{\mathcal{X}} f(\boldsymbol{x}, \boldsymbol{y}). \tag{2.23}$$

2.4 Learning in Games & Online Convex Optimization

Minimizing Regret Both the field of Learning in Games [11] and the field of Online Convex Optimization [22] are concerned with taking decisions on the fly in an optimal, mathematically informed manner. In every day life, a decision is deemed as a good one when you least regret in hindsight. In fact, both aforementioned fields formalize this intuitive criterion into the notion of *regret*. Regret in these fields can come in a couple of flavors. One of them, also known as *external regret*, is defined as the difference between the accumulated payoff of one's decisions compared to the payoff of the best decision they could have made in hindsight.

2.4.1 The Repeated Game Model

Consider a finite normal-form game $\Gamma = \Gamma(\mathcal{N}, \mathcal{A}, \boldsymbol{u})$ and a natural number T denoting the number of times the same game Γ is played. Without loss of generality, we assume that for every player i, their utility u_i is normalized $u_i : \Delta(\mathcal{A}) \to [0, 1]$. At each round t, every player i selects a possibly mixed strategy $\boldsymbol{x}_i^{(t)}$ ultimately yielding the strategy profile $\boldsymbol{x}^{(t)} = (\boldsymbol{x}_1^{(t)}, \ldots, \boldsymbol{x}_N^{(t)})$. It follows that, $u_i(\boldsymbol{x}^{(t)}) = \mathbb{E}_{\boldsymbol{a}\sim\boldsymbol{x}^{(t)}}[u_i(\boldsymbol{a})]$. Furthermore, the sequence of strategy profile $\left\{\boldsymbol{x}^{(t)}\right\}_{0\le t\le T}$ induces a sequence of utilities $\left\{\boldsymbol{u}^{(t)}\right\}_{0\le t\le T}$.

2.4.2 Online Convex Optimization Model

In the setting of online convex optimization, a *player* or *decision-maker*, is called to take a series of sequential decision on-the-go – *online* – against an *adversary* or *nature*. At each step t, the player commits to a decision $\boldsymbol{x}_t \in \mathcal{X}$ and suffers a loss ℓ_t due to the adversary's/nature's selection of a convex function $f_t \in \mathcal{F}$. Of course, the losses to be incurred by each decision cannot be known beforehand as it would beat the purpose of studying *online* decision-making. We also have to note that, the losses and the decision set are bounded.
Formally, the protocol is as follows:
Additionally, due to the convexity of any $f_t(\cdot)$ we can equivalently assume that $f_t(\boldsymbol{x}_t) = \langle \boldsymbol{y}_t, \boldsymbol{x}_t \rangle$ for some vector $\boldsymbol{y}_t$ chosen by the adversary.

Online Convex Optimization

Require: Convex set $\mathcal{X}$
  for t = 1, 2, ..., T **do**
    Player picks a decision $\boldsymbol{x}_t$
    Nature picks a convex loss function f_t
    Player experiences loss $\ell_t = f_t(\boldsymbol{x}_t)$
  end for

2.4.3 Regret and two ways to minimize it

In both models previously referenced, the objective we mentioned we would like to optimize our choices against is a measure of *regret*. This comes to contrast the objective of (convex) optimization that strives to minimize a given objective function. Particularly, we would like minimize the *external regret* of our choices that we formally define as such:

2.4.3.1 Follow-The-Regularized-Leader

Follow-The-Leader Before defining the regularized version of this framework, we shall define the non-regularized Follow-The-Leader approach, also known as *fictitious play* in Economics.
The decision-maker assumes that they can minimize future regrets by making a decision in the present that is optimal for the past. I.e.:

$$\boldsymbol{x}_{t+1} = \arg\min_{\boldsymbol{x}} \sum_{\tau=1}^{t} f(\boldsymbol{x}_\tau) \tag{FTL}$$

It is demonstrable that regret can be linear in the number of iterations. Suppose $\mathcal{X}$ to be $\mathcal{X} = [-1, 1]$ and the adversarial function space $\mathcal{F} = \{-1/2x, 1/2x\}$. Say that the adversary picks $-1/2x$ for even t and $1/2x$ for odd t. If the player follows FTL, they will fluctuate between decisions -1 and 1 and the regret will grow linearly through time.
A natural approach is to help the decision-maker fluctuate less and manage to form more stable trajectories. This can be achieved by means of regularization which informally means taking into consideration previous choices and discouraging abrupt changes between two time steps.

The regularized version First, we observe from a standard convex analysis argument that regret can be bounded by a linear function. Remember that all f_t are convex functions, as such for any two points $\boldsymbol{x}_t, \hat{\boldsymbol{x}}$:

$$f_t(\boldsymbol{x}_t) - f_t(\hat{\boldsymbol{x}}) \leq \nabla f(\boldsymbol{x}_t)(\boldsymbol{x}_t - \hat{\boldsymbol{x}}). \tag{2.24}$$

Define $\nabla_t := \nabla f(\boldsymbol{x}_t)$ and sum the latter inequality for all iterates $\boldsymbol{x}_t, f_t$ to get:

$$\sum_{\tau} f_t(\boldsymbol{x}_\tau) - f_t(\hat{\boldsymbol{x}} \leq \sum_{\tau} \nabla_\tau^\top (\boldsymbol{x}_\tau - \hat{\boldsymbol{x}}). \tag{2.25}$$

Let a regularization function $R(\boldsymbol{x})$ whose properties we shall shortly discuss and a hyper-parameter η. For the Follow-The-Regularized-Leader strategy, the decision-makes at each time step t the player picks a decision that optimizes the objective:

$$\boldsymbol{x}_{t+1} = \arg\min_{\boldsymbol{x} \in \mathcal{X}} \left\{ \sum_{\tau=1}^{t} \eta \nabla_\tau^\top \boldsymbol{x} + R(\boldsymbol{x}) \right\} \tag{FTRL}$$

Meta-Algorithm 1 Follow-The-Regularized-Leader

Require: Convex set $\mathcal{X}$, parameter η, regularization function R
1: $\boldsymbol{x}_1 = \arg\min_{\boldsymbol{x}\in\mathcal{X}}\{R(\boldsymbol{x})\}$
2: **for** t = 1, 2, ..., T **do**
3: Player plays $\boldsymbol{x}_t$ and experiences loss $\ell_t = f_t(\boldsymbol{x}_t)$
4: $\boldsymbol{x}_{t+1} \leftarrow \arg\min_{\boldsymbol{x}\in\mathcal{X}} \left\{\eta \sum_{\tau}^{t} \nabla_{\tau}^{\top}\boldsymbol{x} + R(\boldsymbol{x})\right\}$
5: **end for**

Theorem 2.4.1. *The FTRL algorithm succeeds a bound on regret at time T for every $\boldsymbol{u} \in \mathcal{X}$ as follows:*

$$\mathrm{Reg}^{(T)} \leq 2\eta \sum_{t=1}^{T} \|\nabla_t\|_*^2 + \frac{\mathrm{Reg}(\boldsymbol{u}) - \mathrm{Reg}(\boldsymbol{x}_1)}{\eta}. \tag{2.26}$$

Regularization functions In the context of the present work, we assume that the regularizing function $R : \mathcal{X} \to \mathbb{R}$ is strongly-convex, smooth and twice-differentiable.
We define the diameter of a set $\mathcal{X}$ relative to a function R as:

$$D_R = \sqrt{\max_{\boldsymbol{x},\boldsymbol{y}}\{R(\boldsymbol{x}) - R(\boldsymbol{y})\}} \tag{Diameter}$$

Further, we should define a *dual norm*. Given a general norm $\|\cdot\|$, the dual norm is defined as:

$$\|\boldsymbol{y}\|_* = \sup_{\|\boldsymbol{x}\|\leq 1} \{\boldsymbol{x}^{\top}\boldsymbol{y}^{\top}\}. \tag{2.27}$$

Moreover, we can define a *matrix norm* to be $\|\boldsymbol{x}\|_A = \sqrt{\boldsymbol{x}^{\top}A\boldsymbol{x}}$. The dual matrix norm is $\|\boldsymbol{x}\|_{A,*} = \|\boldsymbol{x}\|_{A^{-1}}$.
A quantity that is crucial in optimization is the residue of a first order Taylor approximation also known as *Bregman divergence*.

Definition 2.4.1 (Bregman Divergence)**.** The Bregman divergence with respect to a function R is defined as:

$$B_R(\boldsymbol{x},\boldsymbol{y}) = R(\boldsymbol{x}) - R(\boldsymbol{y}) - \nabla R(\boldsymbol{y})^{\top}(\boldsymbol{x}-\boldsymbol{y}). \tag{2.28}$$

By the mean value theorem, the twice-differentiability of R, and the Taylor approximation we can prove that there exists an intermediate point $\boldsymbol{z} = \lambda\boldsymbol{x} + (1-\lambda)\boldsymbol{y}$ for some $\lambda \in [0,1]$, such that

$$B_R(\boldsymbol{x},\boldsymbol{y}) = \frac{1}{2}\|\boldsymbol{x}-\boldsymbol{y}\|_{\boldsymbol{x},\boldsymbol{y}}^2 = \frac{1}{2}\|\boldsymbol{x}-\boldsymbol{y}\|_{\boldsymbol{z}}^2, \tag{2.29}$$

where $\|\cdot\|_{\boldsymbol{z}} = \|\cdot\|_{\nabla^2 R(\boldsymbol{z})}$. Since the function is assumed to be strongly convex (hence the matrix $\nabla^2 R(\cdot)$ is positive definite ergo reversible), a dual norm can also be defined: $\|\boldsymbol{x}\|_{\boldsymbol{x},\boldsymbol{y},*} = \|\boldsymbol{x}\|_{\boldsymbol{z},*} = \|\boldsymbol{x}\|_{(\nabla^2(\boldsymbol{z}))^{-1}}$. As such, the Bergman divergence defines a local norm amenable to a dual which is defined as: $\|\cdot\|_{\boldsymbol{x},\boldsymbol{y},*} = \|\cdot\|_{\boldsymbol{z},*} = \|\cdot\|_{(\nabla^2 R(\boldsymbol{z}))^{-1}}$.
Furthermore, it should be noted that the inverse operator of the gradient $(\nabla R)^{-1}$ which we use is equal to the operator of the gradient of the convex conjugate R^* of the given convex function, i.e.:

$$\nabla R^{-1} = \nabla R^*. \tag{2.30}$$

2.4.3.2 (Online) Mirror Descent

An alternative meta-algorithm to minimize regret builds upon the gradient descent method [35]. Mirror descent generalizes gradient descent to making it a lot more versatile. Instead of performing updates directly in the feasible set (in that case the decision set), updates are realized inside a *dual space*. The dual space is intrinsically defined through the choice of the *regularization function* R. The gradient of the regularization maps $\mathbb{R}^d$ into itself. An update is performed inside this vector field and the next iterate is acquired after an appropriate projection.
There are two versions of Online Mirror Descent, an agile and a lazy one. For the agile one, at every step t a feasible point $\boldsymbol{x}_t$ is used. For the lazy one (also known as Nesterov's dual averaging), while the algorithm runs, a feasible point $\boldsymbol{x}_t$ is only retrieved at decision time but the update of $\boldsymbol{y}_{t+1}$ uses the (possibly not feasible) previous iterate $\boldsymbol{y}_t$. The vector $\boldsymbol{y}_0$ is initialized in such a way that $\nabla R(\boldsymbol{y}_0) = 0$

$$\begin{cases} \nabla R(\boldsymbol{y}_{t+1}) &= \nabla R(\boldsymbol{y}_t) - \eta \nabla_t \quad \text{(Lazy Version)} \\ \nabla R(\boldsymbol{y}_{t+1}) &= \nabla R(\boldsymbol{x}_t) - \eta \nabla_t \quad \text{(Agile Version)} \end{cases} \tag{2.31}$$

$$\boldsymbol{x}_{t+1} = \arg\min_{\boldsymbol{x}} B_R(\boldsymbol{x}, \boldsymbol{y}_{t+1}) \tag{2.32}$$

Meta-Algorithm 2 Online Mirror Descent

Require: Convex set $\mathcal{X}$, parameter η, regularization function R
1: $\boldsymbol{y}_1 = \arg_{\boldsymbol{y}}\{\nabla R(\boldsymbol{y}) = 0\}$ and $\boldsymbol{x}_1 \leftarrow \arg\min_{\boldsymbol{x}\in\mathcal{X}}\{B_R(\boldsymbol{x}, \boldsymbol{y}_1)\}$
2: **for** t = 1, 2, ..., T **do**
3: Player plays $\boldsymbol{x}_t$ and experiences loss $\ell_t = f_t(\boldsymbol{x}_t)$
4: $\boldsymbol{y}_{t+1} \leftarrow \left\{ (\nabla R)^{-1}\left(\nabla R(\boldsymbol{y}_t) - \eta\nabla_t\right) \text{ or } (\nabla R)^{-1}\left(\nabla R(\boldsymbol{x}_t) - \eta\nabla_t\right) \right\}$
5: $\boldsymbol{x}_{t+1} \leftarrow \arg\min_{\boldsymbol{x}\in\mathcal{X}} B_R(\boldsymbol{x}, \boldsymbol{y}_{t+1})$
6: **end for**

For the Online Mirror Descent we have a regret on bound Reg after T iterations:

Theorem 2.4.2. *The Online-MD algorithm succeeds a bound on regret at time T for every $\boldsymbol{u} \in \mathcal{X}$ as follows:*

$$\text{Reg}^{(T)} \leq \frac{\eta}{4}\sum_{t}^{T} \|B_R(\boldsymbol{x}_t, \boldsymbol{x}_{t+1})\|_*^2 + \frac{R(\boldsymbol{u}) - R(\boldsymbol{x}_1)}{2\eta} \tag{2.33}$$

Relation between FTRL and Online-MD We will not elaborate much on this, but we will not that Follow-The-Regularized-Leader and the Lazy version of Online Mirror Descent are equivalent.

Lemma 2.4.3 (FTRL Lazy Online-MD equivalence)**.** *Follow-The-Regularized-Leader is equivalent to the Lazy version of Online Mirror-Descent given that $\{f_t\}_{1\leq t\leq T}$ are linear functions.*

2.4.4 Minimizing regret in predictable sequences

There exists a very interesting line of research that has risen in optimization that has predecessors in Popov, Nemirovski's Mirro-Prox and was ultimately formalized into a framework by

[48]. We suppose the Online Convex Optimization protocol and the existence of a sequence of reliable hints $\boldsymbol{m}_t$ for the next loss gradient ∇_t either due to estimation or side information. This allows to use an *optimistic variant* of Mirror Descent (OMD).

Meta-Algorithm 3 Optimistic Mirror Descent

Require: Convex set $\mathcal{X}$, parameter η, regularization function R

1: $\boldsymbol{y}_1 = \arg_{\boldsymbol{y}}\{\nabla R(\boldsymbol{y}) = 0\}$ and $\boldsymbol{x}_1 \leftarrow \arg\min_{\boldsymbol{x}\in\mathcal{X}}\{B_R(\boldsymbol{x}, \boldsymbol{y}_1)\}$
2: **for** t = 1, 2, ..., T **do**
3: $\quad \boldsymbol{y}_t = \arg\min_{\boldsymbol{y}} \eta_t \nabla_t^\top \boldsymbol{y} + B_R(\boldsymbol{y}, \boldsymbol{y}_t)$
4: $\quad \boldsymbol{x}_{t+1} \leftarrow \arg\min_{\boldsymbol{x}\in\mathcal{X}} \eta_t \boldsymbol{m}_t^\top \boldsymbol{x} + B_R(\boldsymbol{x}, \boldsymbol{y}_t)$
5: **end for**

2.4.4.1 Certain instantiations of the Meta-Algorithms

The aforementioned meta-algorithms can assert more concrete instantiations. For example, when we would like to perform (online) optimization over a feasibility set that is a (probability) simplex, we can choose a (neg-)entropic regularization function. Of course, if we use the ℓ^2-norm we can retrieve the gradient descent algorithm.

Negentropic regulizer With the regularization set to be the negentropic function $R(\boldsymbol{x}) = \sum_i \boldsymbol{x}(i) \log\big(\boldsymbol{x}(i)\big)$, the FTRL and lazy MD meta-algorithms give rise to the *multiplicative weights algorithm*:

$$\boldsymbol{x}_{t+1}(i) = \boldsymbol{x}_t(i) \frac{\exp\left(-\eta \nabla_t(i)\right)}{\sum_j \exp\left(-\eta \nabla_t(j)\right)} \tag{2.34}$$

ℓ^2-norm regulizer Respectively, with the ℓ^2 norm set to be the regularization function $R(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{x}\|^2$, the FTRL and lazy MD meta-algorithms give rise to the *gradient descent algorithm*:

$$\boldsymbol{x}_{t+1} = \Pi_{\mathcal{X}}\left\{\boldsymbol{x}_t - \eta \nabla_t\right\} \tag{2.35}$$

Similarly, the same regularizers applied to the Optimistic Mirror Descent template along with the "hint" sequence set to be $\boldsymbol{m}_t = \boldsymbol{u}_{t-1}$ give place to the *optimistic multiplicative weight updates* and *optimistic gradient descent* algorithms.

2.5 First-Order Methods: Conditions and rates of convergence

2.5.1 Standard min-max optimization first-order methods

As we have established the meta-algorithms from which common first-order methods originate, we are ready for their explicit mathematical definition for the case of two-player zero-sum games. Assume the function $f : \times_i \mathcal{X}_i \times \times_j \mathcal{Y}_j \to \mathbb{R}$ and let x_i, y_j denote the i-th minimizer's and j-th maximizer's strategy vectors respectively belonging to the sets $\mathcal{X}_i$, $\mathcal{Y}_i$ accordingly. Also, let $\boldsymbol{x} = (\boldsymbol{x}_1, \cdot, \boldsymbol{x}_n)$ and $\boldsymbol{y} = (\boldsymbol{y}_1, \cdot, \boldsymbol{y}_m)$ denote the concatenation of the strategy vectors of the n minimizers and m maximizers accordingly and $\mathcal{X} = \times_i \mathcal{X}_i, \mathcal{Y} = \times_j \mathcal{Y}_j$. Obviously, for

two-player zero-sum games $n = m = 1$, $\mathcal{X} = \mathcal{X}_1, \mathcal{Y} = \mathcal{Y}_1$. The optimization objective is the following problem:

$$\min_{\boldsymbol{x}\in\mathcal{X}} \max_{\boldsymbol{y}\in\mathcal{Y}} f(\boldsymbol{x}, \boldsymbol{y}) \tag{2.36}$$

The operators $\Pi_{\mathcal{X}_i}, \Pi_{\mathcal{Y}_j}$ are the projection operators to the corresponding sets $\mathcal{X}_i, \mathcal{Y}_j$. We now list the iterative rule of the algorithms that are of concern:

Gradient Descent-Ascent

$$\begin{cases} \boldsymbol{x}_i^{(t+1)} = \Pi_{\mathcal{X}_i}\left\{\boldsymbol{x}_i^{(t)} - \eta\nabla_{\boldsymbol{x}_i} f(\boldsymbol{x}^{(t)}, \boldsymbol{y}^{(t)})\right\} \\ \boldsymbol{y}_j^{(t+1)} = \Pi_{\mathcal{Y}_j}\left\{\boldsymbol{y}_j^{(t)} + \eta\nabla_{\boldsymbol{y}_j} f(\boldsymbol{x}^{(t)}, \boldsymbol{y}^{(t)})\right\} \end{cases}$$

Optimistic Gradient Descent-Ascent

$$\begin{cases} \boldsymbol{x}_i^{(t+1)} = \Pi_{\mathcal{X}_i}\left\{\boldsymbol{x}_i^{(t)} - 2\eta\nabla_{\boldsymbol{x}_i} f(\boldsymbol{x}^{(t)}, \boldsymbol{y}^{(t)}) + \eta\nabla_{\boldsymbol{x}_i} f(\boldsymbol{x}^{(t-1)}, \boldsymbol{y}^{(t-1)})\right\} \\ \boldsymbol{y}_j^{(t+1)} = \Pi_{\mathcal{Y}_j}\left\{\boldsymbol{y}_j^{(t)} + 2\eta\nabla_{\boldsymbol{y}_j} f(\boldsymbol{x}^{(t)}, \boldsymbol{y}^{(t)}) - \eta\nabla_{\boldsymbol{y}_j} f(\boldsymbol{x}^{(t-1)}, \boldsymbol{y}^{(t-1)})\right\} \end{cases}$$

Extra Gradient Method

$$\begin{cases} \boldsymbol{x}_i^{(t+\frac{1}{2})} = \Pi_{\mathcal{X}_i}\left\{\boldsymbol{x}_i^{(t)} - \eta\nabla_{\boldsymbol{x}_i} f(\boldsymbol{x}^{(t)}, \boldsymbol{y}^{(t)})\right\}, & \boldsymbol{x}_i^{(t+1)} = \Pi_{\mathcal{X}_i}\left\{\boldsymbol{x}_i^{(t)} - \eta\nabla_{\boldsymbol{x}_i} f(\boldsymbol{x}^{(t+\frac{1}{2})}, \boldsymbol{y}^{(t+\frac{1}{2})})\right\} \\ \boldsymbol{y}_j^{(t+\frac{1}{2})} = \Pi_{\mathcal{Y}_j}\left\{\boldsymbol{y}_j^{(t)} + \eta\nabla_{\boldsymbol{y}_j} f(\boldsymbol{x}^{(t)}, \boldsymbol{y}^{(t)})\right\}, & \boldsymbol{y}_j^{(t+1)} = \Pi_{\mathcal{Y}_j}\left\{\boldsymbol{y}_j^{(t)} + \eta\nabla_{\boldsymbol{y}_j} f(\boldsymbol{x}^{(t+\frac{1}{2})}, \boldsymbol{y}^{(t+\frac{1}{2})})\right\} \end{cases}$$

Multiplicative Weights Update Method

$$\begin{cases} \boldsymbol{x}_{i,k}^{(t+1)} = \boldsymbol{x}_i^{(t)} \dfrac{\exp\left(-\eta\nabla_{\boldsymbol{x}_{i,k}} f(\boldsymbol{x}^{(t)}, \boldsymbol{y}^{(t)})\right)}{\sum_j x_j^{(t)} \exp\left(-\eta\nabla_{\boldsymbol{x}_{i,k}} f(\boldsymbol{x}^{(t)}, \boldsymbol{y}^{(t)})\right)} \\ \boldsymbol{y}_{j,k}^{(t+1)} = \boldsymbol{y}_j^{(t)} \dfrac{\exp\left(\eta\nabla_{\boldsymbol{y}_{j,k}} f(\boldsymbol{x}^{(t)}, \boldsymbol{y}^{(t)})\right)}{\sum_l \boldsymbol{y}_l^{(t)} \exp\left(\eta\nabla_{\boldsymbol{y}_{l,k}} f(\boldsymbol{x}^{(t)}, \boldsymbol{y}^{(t)})\right)} \end{cases}$$

Optimistic Multiplicative Weights Update Method

$$\begin{cases} \boldsymbol{x}_{i,k}^{(t+1)} = \boldsymbol{x}_{i,k}^{(t)} \dfrac{\exp\left(-2\eta\nabla_{\boldsymbol{x}_{i,k}} f(\boldsymbol{x}^{(t)}, \boldsymbol{y}^{(t)}) + \eta\nabla_{\boldsymbol{x}_{i,k}} f(\boldsymbol{x}^{(t)}, \boldsymbol{y}^{(t-1)})\right)}{\sum_j x_j^{(t)} \exp\left(-2\eta\nabla_{\boldsymbol{x}_{i,k}} f(\boldsymbol{x}^{(t)}, \boldsymbol{y}^{(t)}) + \eta\nabla_{\boldsymbol{x}_{i,k}} f(\boldsymbol{x}^{(t-1)}, \boldsymbol{y}^{(t-1)})\right)} \\ \boldsymbol{y}_{j,k}^{(t+1)} = \boldsymbol{y}_{j,k}^{(t)} \dfrac{\exp\left(2\eta\nabla_{\boldsymbol{y}_j} f(\boldsymbol{x}^{(t)}, \boldsymbol{y}^{(t)}) - \eta\nabla_{\boldsymbol{y}_{j,k}} f(\boldsymbol{x}^{(t-1)}, \boldsymbol{y}^{(t-1)})\right)}{\sum_l \boldsymbol{y}_l^{(t)} \exp\left(2\eta\nabla_{\boldsymbol{y}_{l,k}} f(\boldsymbol{x}^{(t)}, \boldsymbol{y}^{(t)}) - \eta\nabla_{\boldsymbol{y}_{l,k}} f(\boldsymbol{x}^{(t-1)}, \boldsymbol{y}^{(t-1)})\right)} \end{cases}$$

2.5.2 Conditions and rates of Convergence

Conditions of Convergence For GDA and MWU a sufficient condition on convergence to a NE is the convexity-concavity of the utility function.

The convergence of OGDA has been examined by an array of papers. It was initiated by [48] that provided the condition that for the latter inequalities hold for the regret of the iterates:

$$\text{Reg}^{(T)} \leq \frac{R}{\eta} + \eta \sum_{t=1}^{T} \|\boldsymbol{\ell}^{(t)} - \boldsymbol{m}^{(t)}\|_*^2 - \frac{1}{8\eta} \sum_{t=2}^{T} \|\boldsymbol{x}^{(t)} - \boldsymbol{x}^{(t-1)}\| \tag{2.37}$$

Next, [62] generalized the latter to the more general *regret bounded by the variance of utilities* condition:

$$\text{Reg}^{(T)} \leq \alpha + \beta \sum_{t=1}^{T} \|\boldsymbol{u}^{(t)} - \boldsymbol{u}^{(t-1)}\|_*^2 - \gamma \sum_{t=2}^{T} \|\boldsymbol{x}^{(t)} - \boldsymbol{x}^{(t-1)}\| \tag{2.38}$$

Mertikopoulos et al. have proven that for a (possibly nonconvex-nonconcave) function for which a Nash equilibrium satisfies the Minty variational inequality the Extragradient Method converges to a NE with a diminishing step-size. Recently, Cai, Oikonomou, and Zheng showed tight last-iterate rates for both OGDA, EG in the constrained and the unconstrained setting for the case of a monotone variational inequality for a constant step size.

	minimum smooth conv. function	NE two-pl. zero-sum normal-form	NE two-pl. zero-sum non conv.-conc.
GDA[ai]	$\mathcal{O}(1/T)$	$\mathcal{O}(1/\sqrt{T})$	–
MWU[ai]	$\mathcal{O}(1/T)$	$\mathcal{O}(1/\sqrt{T})$	–
OGDA[li]	–	$\mathcal{O}(1/T)$	$\mathcal{O}(1/\sqrt{T})$: monotonicity
EG[li]	–	$\sim$	$\mathcal{O}(1/\sqrt{T})$: monotonicity
OMWU[li]	–	$\mathcal{O}(1/T)$	

Table 2.5: [ai]:average-iterate convergence, [li]:last-iterate convergence

Rates of convergence

2.6 Dynamical Systems

Dynamical systems are omnipresent in the Sciences and are used to model an abundance of phenomena both in Nature and Engineering. Of course, since optimization encompasses iterative processes, the theory of dynamical systems comes in more than handy in analyzing optimization algorithms. A broad and rather vague way to describe dynamical systems consists of 3 elements:

- a *phase space*, or the set of all possible states the system can be in
- *time* which may be continuous or discrete
- *a law of evolution across time* that dictates how the system transitions from one state to another

There are two mainly used types of dynamical systems: *differential equations* and *iterated maps/difference equations* describing the evolution of systems in continuous and discrete time respectively.

Continuous time dynamical systems Let the *phase space* $\mathcal{S} \subset \mathbb{R}^n$ be an open set and f be a continuously differentiable map with $f : \mathcal{S} \to \mathcal{S}$. Then the following form denotes an *autonomous continuous (time)* dynamical system:

$$\frac{d\boldsymbol{x}}{dt} = f(\boldsymbol{x}). \tag{2.39}$$

We have assumed f to be continuously differentiable hence the ordinary differential equation (2.39) with an initial condition $\boldsymbol{x}(0) = \boldsymbol{x}_0 \in \mathcal{S}$ has a unique solution in a time interval $t \in \mathcal{I}(\boldsymbol{x}_0)$. We represent this solution as $\phi(t, \boldsymbol{x}_0)$ and refer to it as the *flow* of the system. This happens to be a generalization of the Picard-Lindelöf theorem. By setting $\phi_t(\boldsymbol{x}_0) := \phi(t, \boldsymbol{x}_0)$ we get a function that describes the *trajectory* of the system with a given starting point at $\boldsymbol{x}_0$. Moreover, we observe that the flow is continuously differentiable and its inverse $\phi_{-t}(\boldsymbol{x})$ exists and is also continuously differentiable. Hence, the flow is a *diffeomorphism* in the maximal interval of existence. Further, we note that $\phi_t \circ \phi_s = \phi_{t+s}$ for any $t, s, t + s \in \mathcal{I}$.

Discrete time dynamical systems Again, let the phase space $\mathcal{S} \subset \mathbb{R}^n$ be an open set and f a map $f : \mathcal{S} \to \mathcal{S}$. Then the *autonomous discrete time* dynamical system can be described as:

$$\boldsymbol{x}^{(k+1)} = f(\boldsymbol{x}^{(k)}). \tag{2.40}$$

There are certain points of the phase-space of a system that draw a surplus of interest. These are the so-called *equilibrium points* of a system, i.e. points $\hat{\boldsymbol{x}}$ where the system will not move from once it has reached them ($f(\hat{\boldsymbol{x}}) = 0$ or $\phi(\hat{\boldsymbol{x}}) = \hat{\boldsymbol{x}}$). A very intriguing question is whether the system will converge to such a given its trajectory starts in a neighborhood around it. The theory of Lyapunov Stability is concerned precisely with this question.

2.6.1 Lyapunov Stability

Definition 2.6.1 (Equilibrium/Fixed point)**.** A point $\hat{\boldsymbol{x}} \in \mathcal{S}$ is said to be an *equilibrium* of the dynamical system $f : \mathcal{S} \to \mathcal{S}$ if:

$$\mathbf{0} = f(\hat{\boldsymbol{x}}) \tag{2.41}$$

Equivalently, $\hat{\boldsymbol{x}}$ is a *fixed point* of the system's flow function

$$\phi_t(\boldsymbol{x}) = \boldsymbol{x}, \ \forall t \in \mathcal{I} \tag{2.42}$$

We say that a fixed point $\hat{\boldsymbol{x}}$ is isolated if there exists a neighborhood around $\boldsymbol{x}$ such that $\hat{\boldsymbol{x}}$ is a the unique fixed point in U.

Definition 2.6.2. Consider the autonomous system f:

$$\boldsymbol{x}^{(k+1)} = f(\boldsymbol{x}^{(k)})$$

The fixed point $\hat{\boldsymbol{x}} = \mathbf{0}$ is:

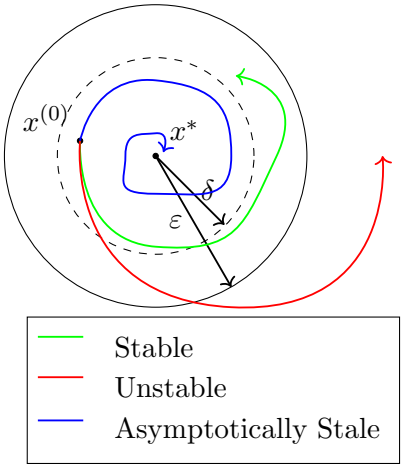
- **stable** if, for each $\varepsilon > 0$, $\exists \delta = \delta(\varepsilon) > 0$ such that:

$$\|\boldsymbol{x}^{(0)}\| < \delta \ \Rightarrow \|\boldsymbol{x}^{(k)}\| < \varepsilon, \ \forall k > 0$$

- **unstable** if it is not stable
- **asymptotically stable** if it is stable and δ can be chosen such that:

$$\|\boldsymbol{x}^{(0)}\| < \delta \;\Rightarrow\; \lim_{k\to\infty} \boldsymbol{x}^{(k)} = \mathbf{0}$$

We will use the terms stabilize/converge interchangeably as the *convergence* of an iterative process corresponds to the *asymptotic stability* of the fixed point it converges to.



Lyapunov's first method How do we test the stability of a (possibly nonlinear) dynamical system at a given equilibrium point $\hat{\boldsymbol{x}}$? A straightforward way of asserting the stability of a given fixed point $\hat{\boldsymbol{x}} \in \mathcal{S}$ of f is testing the eigenvalues of the Jacobian matrix of f evaluated at $\hat{\boldsymbol{x}}$. Put otherwise, we *linearize* the system building upon the Taylor theorem for multivariable functions:

$$f(\boldsymbol{x}) = f(\hat{\boldsymbol{x}}) + \underbrace{\mathbf{J}_f(\hat{\boldsymbol{x}})}_{A}(\boldsymbol{x} - \hat{\boldsymbol{x}}) + O(\|\boldsymbol{x} - \hat{\boldsymbol{x}}\|^2). \tag{2.43}$$

Theorem 2.6.1 (Lyapunov's 1st Method – Continuous time)**.** *Let a continuous time, autonomous dynamical system f and a point with $\hat{\boldsymbol{x}}$ and matrix $A = \mathbf{J}_f(\hat{\boldsymbol{x}})$:*

- *if all eigenvalues of A have a real part negative, then the system is (locally) stable at $\hat{\boldsymbol{x}}$*
- *if there exists an eigenvalue of A with a real part positive, then the the system is unstable at $\hat{\boldsymbol{x}}$*
- *if there exists an eigenvalue of A with a real part equal to 0, then the method is inconclusive about the stability of the system at $\hat{\boldsymbol{x}}$.*

When the system is a discrete time one an analogous theorem holds with respect to the eigenvalues of the Jacobian matrix of f.

Theorem 2.6.2 (Lyapunov's 1st Method – Discrete time)**.** *Let a discrete time, autonomous dynamical system f and a point with $\hat{\boldsymbol{x}}$ and matrix $A = \nabla^2 f|_{\boldsymbol{x}=\hat{\boldsymbol{x}}}$:*

- *if all eigenvalues of A have a magnitude less than* 1*, then the system is (locally) stable at* $\hat{\boldsymbol{x}}$
- *if there exists an eigenvalue of A with a magnitude greater than* 1*, then the the system is unstable at* $\hat{\boldsymbol{x}}$
- *if there exists an eigenvalue of A with a magnitude equal to* 1*, then the method is inconclusive about the stability of the system at* $\hat{\boldsymbol{x}}$*.*

Lyapunov's second method As we have seen, Lyapunov's first method might not be conclusive in some cases and it is also not possible to draw conclusion about stability of non-linear dynamics for a global scale. A second way that at times circumvents the aforementioned caveats for checking the stability of a fixed point stems from the observation of mechanical systems where when the system's *energy/potential* is dissipated and reduced to zero, the system is lead to a stabilize at an equilibrium. This way, through designing an appropriate scalar function that plays the role of *energy* in a given dynamical system, we are able to decide upon its stability. The function that we are searching for is called a *Lyapunov function.*
Before stating the method, we need to define what a postive semi-definite function is:

Definition 2.6.3 (Positive semi-definite function)**.** A scalar continuous function $V : \mathcal{S} \cap N \to \mathbb{R}$ is said to be locally positive semi-definite in N when:

- $V(\hat{\boldsymbol{x}}) = 0$ for a given $\hat{\boldsymbol{x}}$
- $V(\boldsymbol{x}) > 0,\ \forall \boldsymbol{x} \in N, \boldsymbol{x} \neq \hat{\boldsymbol{x}}$.

Of course, if $N = \mathcal{S}$, the function is *globally* positive semi-definite. It is now possible to formally define what a Lyapunov function is:

Definition 2.6.4 (Lyapunov function)**.** A scalar function $V : \mathcal{S} \cap D \to \mathbb{R}$ is said to be a Lyapunov function for the system f and its equilibrium $\hat{\boldsymbol{x}} \in D$ if:

- it is positive semi-definite in a domain D containing $\hat{\boldsymbol{x}}$
- it has continuous partial derivatives
- $V(\hat{\boldsymbol{x}}) = 0$ and $V(\boldsymbol{x}) > 0,\ \forall \boldsymbol{x} \in D - \{\hat{\boldsymbol{x}}\}$
- it has a time derivative that is negative semi-definite for any $\boldsymbol{x}$ in the trajectory of the system f, i.e.: $\dot{V}(\boldsymbol{x}) \leq 0,\ \forall \boldsymbol{x} \in D$.

Lyapunov's second method (or direct method) is the following:

Theorem 2.6.3 (Lyapunov's second method – Continuous time)**.** *Let a continuous time, autonomous system f, an equilibrium $\hat{\boldsymbol{x}}$, and a domain D that contains it $\hat{\boldsymbol{x}} \in D \subset \mathcal{S}$. If there exists a Lyapunov function $V(\boldsymbol{x}) : D \to \mathbb{R}$ with:*

- $V(\hat{\boldsymbol{x}}) = 0$ *and* $V(\boldsymbol{x}) > 0,\ \forall \boldsymbol{x} \in D - \{\hat{\boldsymbol{x}}\}$
- $\dot{V}(\boldsymbol{x}) \leq 0,\ \forall \boldsymbol{x} \in D$

then the equilibrium point $\hat{\boldsymbol{x}}$ is stable.
Furthermore, if $\dot{V}(\boldsymbol{x}) < 0,\ \forall \boldsymbol{x} \in D - \{\boldsymbol{x}\}$, then the system is asymptotically stable.

As for discrete time systems, there exists an analogous theorem:

Theorem 2.6.4 (Lyapunov's second method – Discrete time)**.** *Let a discrete time, autonomous system f, an equilibrium $\hat{\boldsymbol{x}}$, and a domain D that contains it $\hat{\boldsymbol{x}} \in D \subset \mathcal{S}$. If there exists a Lyapunov function $V(\boldsymbol{x}) : D \to \mathbb{R}$ with:*

- $V(\hat{\boldsymbol{x}}) = 0$ *and* $V(\boldsymbol{x}) > 0,\ \forall \boldsymbol{x} \in D - \{\hat{\boldsymbol{x}}\}$
- $V(\boldsymbol{x}^{(t+1)}) - V(\boldsymbol{x}^{(t)}) \leq 0$

then the equilibrium point $\hat{\boldsymbol{x}}$ is stable.
Furthermore, if $V(\boldsymbol{x}^{(t+1)}) - V(\boldsymbol{x}^{(t)}) < 0,\ \forall \boldsymbol{x}^{(t)} \in D - \{\hat{\boldsymbol{x}}\}$, then the system is asymptotically stable.

2.6.2 The stable-manifold theorem

The Center-stable Manifold Theorem is of utmost importance in the theory of dynamical systems. It relates the subspaces spanned by the eigenvectors corresponding to eigenvalues of magnitude less than 1, equal to 1 and greater than 1 (negative real part, real part zero, and positive real part respectively for continuous time systems) at the linearization of the system with the stable, center, and unstable manifolds of the phase space. In this way, we can prove that the set of the initial conditions such that a given algorithm reaches an unstable (Nash) equilibrium is of Lebesgue measure zero.

Theorem 2.6.5 (Center-stable Manifold Theorem (incomplete))**.** *Let $\hat{\boldsymbol{x}}$ be a fixed point for the C^r local diffeomorphism $f : D \to \mathbb{R}^n$ where D is a neighborhood of $\hat{\boldsymbol{x}}$ in $\mathbb{R}^n$ and $\infty > r \geq 1$. Let $E^s \oplus E^c \oplus E^u$ be the invariant splitting of $\mathbb{R}^n$ into the generalized eigenspaces of $\mathbf{J}_f(\hat{\boldsymbol{x}})$ corresponding to eigenvalues of absolute value less than one, equal to one, and greater than one respectively. To each of the five $\mathbf{J}_f(\hat{\boldsymbol{x}}$ invariant subspaces $E^s, E^s \oplus E^c, E^c, E^c \oplus E^u$, and E^u there is associated a local f invariant C^r embedded disc $W^s_{loc}, W^{sc}_{loc}, W^c_{loc}, W^{cu}_{loc}$, and W^u_{loc} tanget to the linear subspace 0 and a ball B around zero in an adapted norm such that:*

$$f(W^{sc}_{loc}) \cap D \cap W^s_{loc} \text{ and if } f^n(\boldsymbol{x}) \in D,\ \forall n \geq 0,\ \text{then } \boldsymbol{x} \in W^{sc}_{loc}. \tag{2.44}$$

2.6.3 An example: leveraging dynamical systems theory in min-max optimization

A simple (continuous) game of two Consider two players, $x, y \in \mathbb{R}$. They both pick an arbitrary value in the hope of turning a utility function $u : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ to their favor. Player 1, corresponding to x, tries to minimize u while player 2, y, tries to maximize it.

Gradient Flow We allow every player to move with the same rate towards their goal. Player x will want to minimize u (conversely, maximize $-u$), which means they will move for every dt towards the direction $-\nabla_x u$, while player y who seeks to maximize u will move towards $\nabla_y(u)$.
For x, we deduce:

$$\begin{aligned} x' = x - dt\nabla_x u \Rightarrow \\ x' - x = -dt\nabla_x u \Rightarrow \\ \frac{x' - x}{dt} = -\nabla_x u \Rightarrow \\ \frac{dx}{dx} = -\nabla_x u \end{aligned} \tag{2.45}$$

Similarly for y, we can write:

$$\frac{du}{dy} = \nabla_y(u) \tag{2.46}$$

Conclusively, we define a first order system of differential equations:

$$\begin{cases} \dot{x} &= -\nabla_x u(x,y) \\ \dot{y} &= \nabla_y u(x,y) \end{cases} \Rightarrow$$

$$\begin{cases} \dot{x} &= -\nabla_x (xy) \\ \dot{y} &= \nabla_y (xy) \end{cases} \Rightarrow$$

$$\begin{cases} \dot{x} &= -y \\ \dot{y} &= x \end{cases}$$

$$\begin{pmatrix} \dot{x} \\ y \end{pmatrix} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \tag{2.47}$$

Uniqueness of $(0,0)$ as a Nash Equilibrium Point We remind ourselves that a Nash Equilibrium point is a point (point meaning the tuple of strategy vectors of all players) such that no player can unilaterally alter their strategy to turn utility to their favor if the strategy vectors of all other players were to be fixed.
In our example, the strategy vector of each player is reduced to a scalar real variable. We now move on to prove that only $(x^*, y^*) = (0,0)$ can hold the prerequisites for it to be an equilibrium point.
We observe that if $x = c \neq 0$, player y can alter their strategy to $y \to \infty$ (or $y \to -\infty$) if $c < 0$ (or $c > 0$). The converse holds for $y = c' \neq 0$.
But, if $(x,y) = (0,0)$, we observe that no matter what each player alter their strategy to, utility will stay the same. Hence, $(0,0)$ is a unique equilibrium point for the game in question.

Distance from origin Let us define $V(x,y) = x^2 + y^2$. We observe that V is the squared distance of the point (x,y) from $(0,0)$. We derive with respect to time:

$$\begin{aligned} &\frac{d}{dt}V(x,y) = \frac{d}{dt}(x^2 + y^2) \Rightarrow \\ &\frac{d}{dt}V(x,y) = 2x\frac{dx}{dt} + 2y\frac{dy}{dt} \Rightarrow \\ &\frac{d}{dt}V(x,y) = 2x(-y) + 2y(x) \Rightarrow \\ &\frac{d}{dt}V(x,y) = 0 \Rightarrow \\ &V(x,y) = \text{const.} \end{aligned}$$

We can thus conclude that the distance of (x,y) remains constant through time.
Alternatively, we can find a closed form solution to 2.47:

$$\begin{cases} x = c \cdot \cos(t) \\ y = -c \cdot \sin(t) \end{cases} \quad \text{and} \quad \begin{cases} x = c' \cdot \cos(t) \\ y = c' \cdot \sin(t) \end{cases} \tag{2.48}$$

From either one of the solutions we get the following plot 2.9 of trajectories, either clockwise or anti-clockwise:

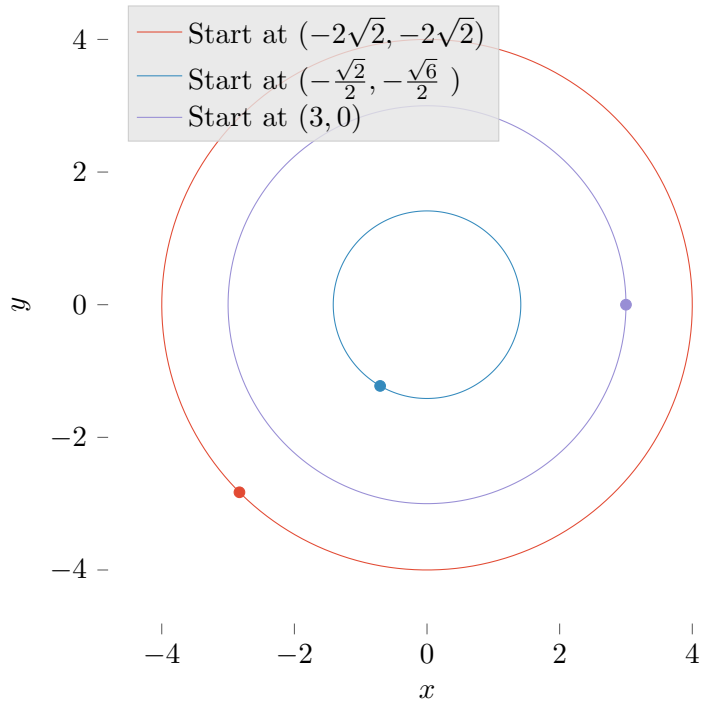


Figure 2.9: Example: Gradient flow trajectories of (x, y) starting from 3 different initial points

Discretization Let us now consider the case of a discrete time version of the latter game. Players still play their moves simultaneously. We will now demonstrate that the vanilla Gradient Descent/Ascent based dynamic won't converge. Not only does it not converge, but the distance of (x, y) from $(0, 0)$ will grow exponentially! Players are notated with variables x, y, which in our case happen to be scalars but in the general can be vectors. Notation $z^{(n)}$ denotes the value of z at step n. The operator $\nabla_{\boldsymbol{z}}$ stands for the operator of a gradient with respect to $\boldsymbol{z}$ of a given function. (e.g. $\nabla_{\boldsymbol{z}}(zw) = w$). If $\boldsymbol{z}$ is a vector in $\mathbb{R}^d$, $\nabla_{\boldsymbol{z}}$ is a vector with entries $(\frac{\partial}{\partial z_1}, \ldots, \frac{\partial}{\partial z_d})$ and finally η denotes what is called a step size. It merely is a scalar for which $0 < \eta \leq 1$.

Vanilla Gradient Descent/Ascent

$$\begin{cases} x^{(n+1)} = x^{(n)} - \eta \nabla_x u \\ y^{(n+1)} = y^{(n)} + \eta \nabla_y u \end{cases} \Rightarrow$$
$$\begin{cases} x^{(n+1)} = x^{(n)} - \eta y^{(n)} \\ y^{(n+1)} = y^{(n)} + \eta x^{(n)} \end{cases}$$

Thus we derive the following iterative process:

$$\begin{cases} x^{(n+1)} = x^{(n)} - \eta y^{(n)} \\ y^{(n+1)} = y^{(n)} + \eta x^{(n)} \end{cases} \tag{2.49}$$

Non-convergence of vanilla Gradient Descent/Ascent We define the ratio of the norm of the (x, y) vector between two sequential iteration of the process as ρ:

$$\rho = \frac{\|(x^{(n+1)}, y^{(n+1)})\|_2}{\|(x^{(n)}, y^{(n)})\|_2}$$

Elaborating more on ρ:

$$\begin{aligned}
&\rho = \frac{\sqrt{(x^{(n+1)})^2 + (y^{(n+1)})^2}}{\sqrt{(x^{(n)})^2 + (y^{(n)})^2}} \Rightarrow \\
&\rho^2 = \frac{(x^{(n+1)})^2 + (y^{(n+1)})^2}{(x^{(n)})^2 + (y^{(n)})^2} \overset{2.49}{\Longrightarrow} \\
&\rho^2 = \frac{(x^{(n)} - \eta y^{(n)})^2 + (y^{(n)} + \eta x^{(n)})^2}{(x^{(n)})^2 + (y^{(n)})^2} \Rightarrow \\
&\Rightarrow \cdots \Rightarrow \\
&\rho = \sqrt{1 + \eta^2}
\end{aligned}$$

It is obvious that $\rho > 0$ and more precisely if we define $\rho = 1 + \epsilon$:

$$\begin{aligned}
\rho = 1 + \epsilon \Rightarrow \\
\sqrt{1 + \eta^2} = 1 + \epsilon \Rightarrow \\
\epsilon = \sqrt{1 + \eta^2} - 1 \geq 0 = O(\eta)
\end{aligned}$$

In the following figure 2.10 we can observe how will the vector (x, y) move trough time if it were to start from 3 different random initial points.

Non-convergence through a Lyapunov stability argument Observe that the latter dynamic can be written as a linear time invariant system in the following way:

$$\begin{pmatrix} x^{(n+1)} \\ y^{(n+1)} \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & -\eta \\ \eta & 1 \end{pmatrix}}_{A} \begin{pmatrix} x^{(n)} \\ y^{(n)} \end{pmatrix} \tag{2.50}$$

The eigenvalues of A are $1 + j\eta$ and $1 - j\eta$. Observe that $|1 + j\eta| = |1 - j\eta| > 1, \ \forall \eta > 0$. Hence, for every $\eta > 0$ the Nash equilibrium point $(0, 0)$ is (globally) unstable for the dynamical system that corresponds to gradient descent-ascent. Hence GDA does not converge to a Nash equilibrium.

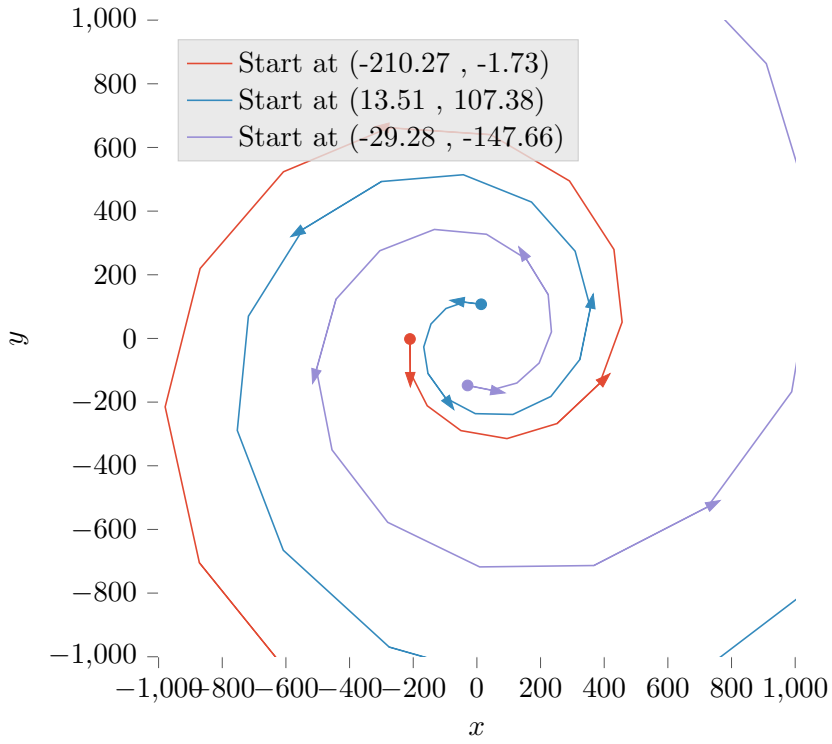


Figure 2.10: Example: GDA trajectories of (x, y) starting from 3 different initial points

Chapter 3

Min-max optimization in two-team zero-sum games

3.1 Two-team zero-sum games

Formally, a *two-team game* in normal form is defined as a tuple $\Gamma = \Gamma(\mathcal{N}, \mathcal{A}, u)$ consisting of

- a finite set of *players* $\mathcal{N}$, split into two teams A, B with k_A and k_B players correspondingly such that: $\mathcal{N} = \mathcal{N}_A \cup \mathcal{N}_B = \{A_1, \cdots, A_{k_A}, B_1, \cdots, B_{k_B}\}$
- a finite set of *actions* (or *pure strategies*) $\mathcal{A}_i = \{\alpha_1, \ldots, \alpha_{n_i}\}$ per player $i \in \mathcal{N}$
- each team's *payoff* function $u_A, u_B : \mathcal{A} \to \mathbb{R}$, where $\mathcal{A} := \prod_i \mathcal{A}_i$ denotes the ensemble of all possible action profiles $\alpha = (\alpha_{A_1}, \ldots, \alpha_{A_{k_A}}, \alpha_{B_1}, \ldots, \alpha_{B_{k_B}})$ while the *individual utility* of a player is identical to her teammates, i.e., $u_i = u_A \ \& \ u_j = u_B \ \forall (i,j) \in \mathcal{N}_A \times \mathcal{N}_B$.

In this general context, players could also submit *mixed strategies*, i.e, probability distributions $s_k \in \Delta(\mathcal{A}_k)$ over actions $\alpha_k \in \mathcal{A}_k$. Correspondingly, we define the product distributions $\boldsymbol{x} = (s_{A_1}, \ldots, s_{A_{k_A}})$, $\boldsymbol{y} = (s_{B_1}, \ldots, s_{B_{k_B}})$ as the teams' strategies. Conclusively, we will write $\mathcal{X} := \prod_{i \in \mathcal{N}_A} \mathcal{X}_i = \prod_{i \in \mathcal{N}_A} \Delta(\mathcal{A}_i), \mathcal{Y} := \prod_{i \in \mathcal{N}_A} \mathcal{Y}_i = \prod_{i \in \mathcal{N}_B} \Delta(\mathcal{A}_i)$ the space of mixed strategy profiles of teams A, B.
Similar to bilinear two-player games, the teams' utility functions can be expressed via payoff-*tensors* $\mathbf{A}, \mathbf{B} \in \mathbb{R}^\tau$ with $\tau = \prod_{i \in \mathcal{N}} |\mathcal{A}_i|$ and acquire the form [1] :

$$u_A = \mathbf{A}_{\boldsymbol{x}}^{\boldsymbol{y}} \ \& \ u_B = \mathbf{B}_{\boldsymbol{x}}^{\boldsymbol{y}} \tag{3.1}$$

In terms of solutions, we focus on the per player *Nash equilibrium* (NE), i.e., a state strategy profile $s^* = (\boldsymbol{x}, \boldsymbol{y}) = \Big((s^*_{A_1}, \ldots, s^*_{A_{k_A}}), (s^*_{B_1}, \ldots, s^*_{B_{k_B}})\Big)$ such that

$$u_i(s^*) \geq u_i(s_i; s^*_{-i}) \text{ [2] for all } s_i \in \Delta(\mathcal{A}_i) \text{ and all } i \in \mathcal{N} \tag{NE}$$

The strategy profile s^* is called *pure* if all player of both teams choose a single action; otherwise we say that it is mixed. Finally, a two-team game is called *two-team zero-sum* if $u_A = -u_B$ or equivalently $\mathbf{A} + \mathbf{B} = \mathbf{O}$.

[1] If $\boldsymbol{x}, \boldsymbol{y}$ have shapes (i,j) and (k,l) that would be equivalent to: `u = einsum('ijkl,ij,kl', A, x, y)`

[2] We are using here the standard shorthand $(s_1, \ldots, s_i, \ldots, s_{|\mathcal{N}|})$ to highlight the strategy of a given player $i \in \mathcal{N}$ versus the rest of players $\mathcal{N} \setminus \{i\}$.

Remark 2. On a quite technical note, for the rest of this work we will assume that a succinct representation of the utility tensors of the game is available or –equivalently – that a payoff oracle efficiently provides both the value of the utility function and its derivatives for a specific input, an assumption that is consistent with the vast majority of the applications that are described in the literature [61].

A first approach to computing NE in Two-Team Zero-Sum games. Due to the multilinearity of the utility and the existence of a duality gap, the linear programming method used in two-player zero-sum games can not be used to compute a Nash equilibrium. For the goal of computing Nash equilibrium in two-team zero-sum games, we have experimented with a selection of first-order methods that have been utilized with varying success in the setting of the two-person zero-sum case. Namely, we analyze the following methods: *i) Gradient Descent-Ascent ii) Optimistic Gradient Descent-Ascent iii) Extra Gradient Method iv) Optimistic Multiplicative Weights Update Method* .
The below folklore fact will play a key role hereafter.

Fact 1. Any fixed point of the aforementioned discrete-time dynamics on the utility function necessarily corresponds to a Nash equilibrium of the game.

Hence, an important test bed for the long-run behavior of GDA, OGDA, and EG methods is to examine whether these methods stabilize around their fixed points, which effectively constitute the Nash equilibria of the game. In Section 3.2.2, we show that in the absence of pure Nash equilibria, all the above methods fail to stabilize on their fixed points even for a simple class of two-team game with $(k_A = 2, k_B = 2)$, and as a consequence they fail to discover the mixed Nash equilibria of the game.
The presence of these results demonstrates the need for a different approach that lies outside the scope of traditional optimization techniques. Inspired by the applications of washout filters to stabilize highly susceptible systems and the adaptive control generalizations of the former, we design a new variant of GDA, "vaned" with a feedback loop that is dictated by a pair of two matrices. Surprisingly, in contrast to the aforementioned conventional methods, our proposed technique accomplishes last-iterate stabilization on its fixed point, i.e., the mixed Nash equilibria of the team game.

(K, P)-Vaned GDA Method. After concatenating the vectors of the minimizing and the maximizing agents $\boldsymbol{z}^{(k)} = (\boldsymbol{x}^{(k)}, \boldsymbol{y}^{(k)})$ we can write our method, for appropriate matrices K, P:

$$\begin{cases} \boldsymbol{z}^{(k+1)} = \Pi_{\mathcal{Z}}\Big\{\boldsymbol{z}^{(k)} + \eta\big(\begin{smallmatrix} -\nabla_{\boldsymbol{x}} f(\boldsymbol{z}^{(k)}) \\ \nabla_{\boldsymbol{y}} f(\boldsymbol{z}^{(k)}) \end{smallmatrix}\big) + \eta K(\boldsymbol{z}^{(k)} - \boldsymbol{\theta}^{(k)})\Big\} \\ \boldsymbol{\theta}^{(k+1)} = \Pi_{\mathcal{Z}}\Big\{\boldsymbol{\theta}^{(k)} + \eta P(\boldsymbol{z}^{(k)} - \boldsymbol{\theta}^{(k)})\Big\} \end{cases} \tag{KPV-GDA}$$

Intuitively, the added variable $\boldsymbol{\theta}^{(k)}$ holds an estimate of the fixed point, and through the feedback law $\eta K(\boldsymbol{z}^{(k)} - \boldsymbol{\theta}^{(k)})$ the vector $\boldsymbol{z}$ stabilizes around that estimate which slowly moves towards the real fixed point of the vanilla GDA dynamic (see Figure 3.1 for a typical evolution of the state vector $\boldsymbol{z}$ and estimate vector $\boldsymbol{\theta}$). It is crucial to note that no additional fixed points are introduced to the system.

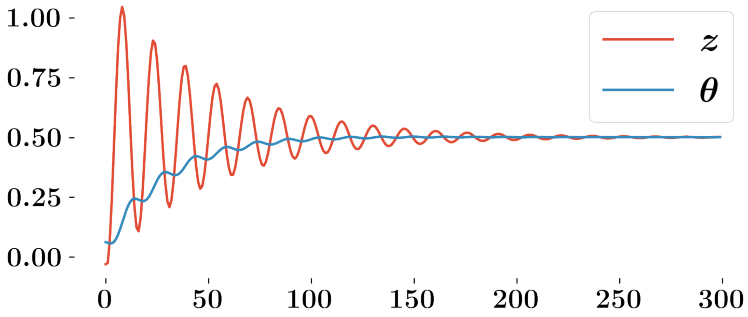


Figure 3.1: Typical KPV-GDA trajectory

3.2 Our main results

3.2.1 On the hardness of computing NE in Two-Team Zero-Sum Games

We start this section by showing that computing a Nash equilibrium in two-team zero-sum games is computationally hard and thus getting a polynomial-time algorithm that computes a Nash equilibrium is unlikely.

Theorem 3.2.1 (CLS-hard)**.** *Computing a Nash equilibrium in two-team zero-sum games is* CLS*-hard.*

The main idea of the proof of Theorem 3.2.1 relies on a reduction of approximating Nash equilibria in congestion games, which has been shown to be complete for the interesting complexity class of CLS, which contains the problem of continuous optimization. For concision, we defer the proof of the above theorem to the paper's supplement.

3.2.2 Instability of the most-common first-order methods

The negative computational complexity result we proved for two-team zero-sum games (Theorem 3.2.1) does not preclude the prospect of attaining algorithms (learning dynamics, first-order methods) that converge to Nash equilibria and thus can approximate them well enough. Unfortunately, we can also prove negative results about convergence to Nash equilibria in two-team zero-sum games of the well-established methods broadly used in classic two-player zero-sum games.

In this section, we are going to construct a family of two-team zero-sum games with the property that GDA, OGDA, EG, and OMWU fail to stabilize to Nash equilibria. This result indicates how challenging and rich the setting of team zero-sum games can be and why provable guarantees about convergence have not been established yet. This family of games may not exhaust the hardness or richness of two-team games, but the mere fact that it exhibits such challenges for all mentioned algorithms is telling of the challenges that lie with these games both in terms of applications as well as in theory. Before defining the family of two-team zero-sum games, we prove an important theorem which states that GDA does not stabilize around mixed Nash equilibria. This fact is a stepping stone in constructing the family of team-zero

sum games later. We present the proof of all of the below statements in detail in the paper's appendix.

Weakly-stable Nash equilibrium [31, 37]. Consider the set of Nash equilibria with the property that if any single randomizing agent of one team is forced to play any strategy in her current support with probability one, all other agents of the same team must remain indifferent between the strategies in their support. This type of Nash equilibria is called weakly-stable. Note that trivially pure Nash equilibria are weakly-stable. It has been shown that mixed Nash equilibria are not weakly-stable in generic games[3] [31].
We can show that Nash equilibria that are not weakly-stable Nash are actually unstable for GDA. Moreover, through standard dynamical systems machinery, that the set of initial conditions that converges to Nash equilibria that are not weakly-stable should be of Lebesgue measure zero. Formally, we prove that:

Theorem 3.2.2 (Non weakly-stable Nash are unstable)**.** *Consider a two-team zero-sum game with utility function of Team B ($\boldsymbol{y}$ vector) being $U(\boldsymbol{x}, \boldsymbol{y})$ and Team A ($\boldsymbol{x}$ vector) being $-U(\boldsymbol{x}, \boldsymbol{y})$. Moreover, assume that $(\boldsymbol{x}^*, \boldsymbol{y}^*)$ is a Nash equilibrium of full support that is not weakly-stable. It follows that the set of initial conditions so that GDA converges to $(\boldsymbol{x}^*, \boldsymbol{y}^*)$ is of measure zero for step size $\eta < \frac{1}{L}$ where L is the Lipschitz constant of ∇U.*

3.2.3 Generalized Matching Pennies (GMP)

Inspired by Theorem 3.2.2, in this section we construct a family of team zero-sum games so that GDA, OGDA, OMWU, and EG methods fail to converge (if the initialization is a random point in the simplex, the probability of convergence of the aforementioned methods is zero). The intuition is to construct a family of games, each of which has only mixed Nash equilibria (that are not weakly-stable), i.e., the constructed games should lack pure Nash equilibria; using Theorem 3.2.2, it would immediately imply our claim for GDA. It turns out that OGDA, OMWU, and EG also fail to converge for the same family.

Definition of GMP. Consider a setting with two teams (Team A, Team B), each of which has $n = 2$ players. Inspired by the standard matching pennies game and the game defined in [54], we allow each agent i to have two strategies/actions that is $S = \{H, T\}$ for both teams with 2^4 possible strategy profiles. In case all the members of a Team choose the same strategy say H or T then the Team "agrees" to play H or T (otherwise the Team "does not agree").

	HH	HT/TH	TT
HH	$1, -1$	$\omega, -\omega$	$-1, 1$
HT/TH	$-\omega, \omega$	$0, 0$	$-\omega, \omega$
TT	$-1, 1$	$\omega, -\omega$	$1, -1$

Thus, in the case that both teams "agree", the payoff of each team is actually the payoff for the two-player matching pennies. If one team "agrees" and the other does not, the team that

[3] Roughly speaking, generic games where we add small Gaussian noise to perturb slightly every payoff only so that we preclude any payoff ties. In these games, all Nash equilibria in all but a measure-zero set of games exhibit the property that all pure best responses are played with positive probability.

"agrees" gets payoff $\omega \in (0,1)$ and the other team gets penalty ω. If both teams fail to "agree", both teams get payoff zero. Let x_i with $i \in \{1,2\}$ be the probability that agent i of Team A chooses H and $1-x_i$ the probability that she chooses T. We also denote $\boldsymbol{x}$ the vector of probabilities for Team A. Similarly, we denote y_i for $i \in \{1,2\}$ be the probability that agent i of Team B chooses H and $1-y_i$ the probability that she chooses T and $\boldsymbol{y}$ the probability vector.

Properties of GMP. An important remark on the properties of our presented game is due. Existing literature tackles settings with:

- (weak-)monotonocity [39, 15],
- cocoercivity [70],
- unconstrained solution space [20]

. Our game is carefully crafted and –although it has a *distinct structure* and is nonconvex-nonconcave only due to *multilinearity*– satisfies none of the latter properties. This makes the (local) convergence of our proposed method even more surprising. .

The first fact about the game that we defined is that for $\omega \in (0,1)$, there is only one Nash equilibrium $(\boldsymbol{x}^*, \boldsymbol{y}^*)$, which is the uniform, i.e., $x_1^* = x_2^* = y_1^* = y_2^* = \frac{1}{2}$ for all agents i.

Lemma 3.2.3 (GMP has a unique NE)**.** *The Generalized Matching Pennies game exhibits a unique Nash equilibrium which is* $(\boldsymbol{x}^*, \boldsymbol{y}^*) = ((\frac{1}{2}, \frac{1}{2}), (\frac{1}{2}, \frac{1}{2}))$.

Remark 3. The fact that the game we defined has a unique Nash equilibrium that is in the interior of $[0,1]^4$ is really crucial for our negative convergence results later in the section as we will show that it is not a weakly-stable Nash equilibrium and the negative result about GDA will be a corollary due to Theorem 3.2.2. Please also note that if $\omega = 1$ then there are more Nash equilibria, in particular the $(\mathbf{0},\mathbf{0}), (\mathbf{1},\mathbf{0}), (\mathbf{0},\mathbf{1}), (\mathbf{1},\mathbf{1})$ are also Nash equilibria (which are pure).

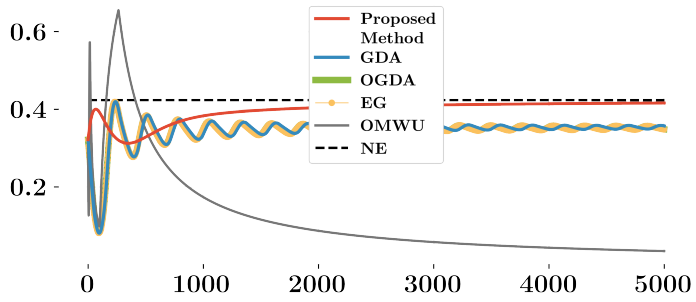
The following Theorem is the main (negative) result of this section.

Theorem 3.2.4 (GDA, OGDA, EG, and OMWU fail)**.** *Consider GMP game with* $\omega \in (0,1)$. *Assume that* $\eta_{GDA} < \frac{1}{4}$, $\eta_{OGDA} < \min(\omega, \frac{1}{8})$, $\eta_{EG} < \frac{\omega}{2}$, *and* $\eta_{OMWU} < \min\left(\frac{1}{4}, \frac{\omega}{2}\right)$ *(bound on the stepsize for GDA, OGDA, EG, and OMWU methods respectively). It holds that the set of initial conditions so that GDA, OGDA, EG, OMWU converge (stabilize to any point) is of measure zero.*

Remark 4. Theorem 3.2.4 formally demonstrates that the behavior of algorithms in **??** are not a result of bad parametrization, and in fact, the probability of the dynamics stabilizing on the NE is equal to the probability of the initial value of the parameters coincide with the NE (measure zero).

Remark 5 (Average iterate also fails). One might ask what happens when we consider average iterates instead of the last iterate. It is a well-known fact [62] that the average iterate of no-regret algorithms converges to coarse correlated equilibria (CCE) so we expect that the average iterate stabilizes. Nevertheless, CCE might not be Nash equilibria. Indeed we can construct examples in which the average iterate of GDA, OGDA, OMWU, and EG experimentally fail to stabilize to Nash equilibria. In particular, we consider a slight modification of GMP; players and strategies are the same but the payoff matrix has changed and can be found below (table on the right):

Figure 3.2 illustrates that the average iterates of GDA, OGDA, OMWU, and EG stabilize to points that are not Nash equilibria. Note that since our method (see next subsection) converges locally, the average iterate should converge locally to a Nash equilibrium.



	HH	HT/TH	TT
HH	$2,-2$	$\frac{1}{2},-\frac{1}{2}$	$-2,2$
HT/TH	$-\frac{1}{2},\frac{1}{2}$	$0,0$	$-\frac{1}{2},\frac{1}{2}$
TT	$-1,1$	$\frac{1}{2},-\frac{1}{2}$	$1,-1$

Figure 3.2: GDA, OGDA, & EG fail to converge to a Nash Equilibrium even in average

3.2.4 Wash-out Filters & Adaptive Control

The aforementioned results indicate that to answer the tantalizing question of finding NE in two-team zero-sum games, our machinery should be broadened outside the limits of the textbook optimization arsenal. The mainstay of this effort and our positive result is the KPV-GDA method defined in (KPV-GDA), inspired by the adaptive control toolbox and washout filters. Our main statement shows that KPV-GDA stabilizes around any Nash equilibrium for appropriate choices of matrices K, P. The formal theorem is given below:

Theorem 3.2.5 (KPV-GDA stabilizes)**.** *Consider a team zero-sum game so that the utility of Team B is $U(\boldsymbol{x},\boldsymbol{y})$ and hence the utility of Team A is $-U(\boldsymbol{x},\boldsymbol{y})$ and a Nash equilibrium $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ of the game. Moreover we assume*

$$\begin{pmatrix} -\nabla^2_{\boldsymbol{xx}}U(\boldsymbol{x}^*,\boldsymbol{y}^*) & -\nabla^2_{\boldsymbol{xy}}U(\boldsymbol{x}^*,\boldsymbol{y}^*) \\ \nabla^2_{\boldsymbol{yx}}U(\boldsymbol{x}^*,\boldsymbol{y}^*) & \nabla^2_{\boldsymbol{yy}}U(\boldsymbol{x}^*,\boldsymbol{y}^*) \end{pmatrix} \text{ is invertible.}$$

For any fixed step size $\eta > 0$, we can always find matrices K, P so that KPV-GDA method defined in (KPV-GDA) converges locally to $(\boldsymbol{x}^*,\boldsymbol{y}^*)$.

The latter statement concerns the existence of matrices K, P. Below, we provide a sufficient condition under which a simple parametrization of K, P (provably) guarantees convergence.

Theorem 3.2.6. *Consider a two-team zero-sum game so that the utility of Team B is $U(\boldsymbol{x},\boldsymbol{y})$ and hence the utility of Team A is $-U(\boldsymbol{x},\boldsymbol{y})$ and a Nash equilibrium $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ of the game. Moreover let*

$$H := \begin{pmatrix} -\nabla^2_{\boldsymbol{xx}}U(\boldsymbol{x}^*,\boldsymbol{y}^*) & -\nabla^2_{\boldsymbol{xy}}U(\boldsymbol{x}^*,\boldsymbol{y}^*) \\ \nabla^2_{\boldsymbol{yx}}U(\boldsymbol{x}^*,\boldsymbol{y}^*) & \nabla^2_{\boldsymbol{yy}}U(\boldsymbol{x}^*,\boldsymbol{y}^*) \end{pmatrix}.$$

and E be the set of eigenvalues ρ of H with real part positive, that is $E = \{H\text{'s eigenvalues } \rho : Re(\rho) > 0\}$. We assume that H is invertible and moreover

$$\beta = \min_{\rho\in E} \frac{Re(\rho)^2 + Im(\rho)^2}{Re(\rho)} > \max_{\rho\in E} Re(\rho) = \alpha. \tag{3.2}$$

We set $K = k \cdot \mathbf{I}$, $P = p \cdot \mathbf{I}$. *There exist small enough step size* $\eta > 0$ *and scalar* $p > 0$ *and for any* $k \in (-\beta, -\alpha)$ *so that KPV-GDA method defined in (KPV-GDA) with chosen* K, P *converges locally to* $(\boldsymbol{x}^*, \boldsymbol{y}^*)$.

Remark 6. As long as conditions in Theorem 3.2.5 are satisfied, KPV-GDA locally converges in any nonconvex-nonconcave game (hence, normal-form two-player zero-sum games as well).

Chapter 4

Multi-generator/discriminator Generative Adversarial Networks

While simple tasks such as classification can have a very straightforward definition of the quality of the resulting model (e.g. the accuracy score), generative models do have such a simple way of assessing the quality of the learned model. Nevertheless, two measures are consistently used when evaluating GANs, inception score and Fréchet inception distance [23].

4.1 Inception Score

The *inception score* metric was introduce in [52]. It gets its name from the Inception classifier [63]. The score tries to capture (i) the generation of distinct objects in each sample, (ii) the generation of multiple objects in the generated samples. Hence, the entropy of the category probability vector output $p(y|\cdot)$ for a given sample $\boldsymbol{z}$ the Inception classifier, $p(y|\boldsymbol{z})$, has to be low (distinct object in a sample gets labeled with a single category), while the entropy of the marginal distribution of labels for the generated samples has to be high (multiple categories are represented in the generated data).

$$\mathrm{IS}(p_{\mathrm{model}}) = \exp\left(\mathbb{E}_{\boldsymbol{z}\sim p_{\mathrm{model}}}[\mathrm{KL}(p(y|\boldsymbol{z}))\|p(y)]\right) \tag{IS}$$

4.2 Fréchet Inception Distance

The *Fréchet inception distance* (FID) [23] compares the activations of the hidden layers of the Inception neural network for the generated and the real data. It is the squared Wasserstein metric between the distributions of the real data represented represented a normal distribution $\mathcal{N}(\mu_{\mathrm{real}}, \Sigma_{\mathrm{real}})$ and the generated data similarly represented as $\mathcal{N}(\mu_{\mathrm{model}}, \Sigma_{\mathrm{model}})$. I.e.:

$$\mathrm{FID} = \|\mu_{\mathrm{real}} - \mu_{\mathrm{model}}\|_2^2 + \mathrm{tr}\left(\Sigma_{\mathrm{real}} + \Sigma_{\mathrm{model}} - 2\left(\Sigma_{\mathrm{real}}^{\frac{1}{2}} \cdot \Sigma_{\mathrm{model}} \cdot \Sigma_{\mathrm{real}}^{\frac{1}{2}}\right)^{\frac{1}{2}}\right) \tag{FID}$$

4.3 On the expressive power of a GAN with finitely many generators

Arora et al. examine the generalization, diversity, and expressivity of GANs in an insightful theoretical way. With regards to the expressivity, they provide a quite informative argument for the use of multiple generators instead of a single one which can be summarized in the following informal theorem:

Theorem 4.3.1 (Expressivity of multi-generator GANs – Informal)**.** *Let the discrimintor D be a neural net with p parameters. Then, a mixture of $\tilde{\mathcal{O}}(p\log(p/\epsilon)/\epsilon^2)$ generators can produce a distribution $\mathcal{D}$ that the discriminator D will not be able to distinguish from $\mathcal{D}_{real}$ with probability greater than ϵ.*

4.4 Conceptual Experiments

In this section we will try to further motivate research into multi-GANs through two conceptual experiments. The first experiment highlights the potential of our proposed method as it quickly converges to a solution. The second one illustrates the expressive superiority of multi-GANs over 1-vs.-1 architectures.

4.4.1 Learning a mixture of Gaussians using our proposed optimizer

Consider the case of $\mathcal{M}$, a mixture of gaussian distribution with two components, $C_1 \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{I})$ and $C_2 \sim \mathcal{N}(-\boldsymbol{\mu}, \boldsymbol{I})$ and mixture weights π_1, π_2 to be positive such that $\pi_1 + \pi_2 = 1$ and $\pi_1, \pi_2 \neq \frac{1}{2}$.
To learn the distribution above, we utilize an instance of a *Team*-WGAN in which there exists a generating team of agents $G_p : \mathbb{R} \to \mathbb{R}, G_{\boldsymbol{\theta}} : \mathbb{R}^n \to \mathbb{R}^n$, and a discriminating team of agents $D_{\boldsymbol{v}} : \mathbb{R}^n \to \mathbb{R}, D_{\boldsymbol{w}} : \mathbb{R}^n \to \mathbb{R}$, all described by the following equations:

$$\begin{array}{c} \text{Generators:} \quad G_p(\zeta) = p + \zeta \ , \ G_\theta(\boldsymbol{z}) = \ \boldsymbol{z} + \boldsymbol{\theta} \\ \text{Discriminators: } D_{\boldsymbol{v}}(\boldsymbol{y}) = \langle \boldsymbol{v}, \boldsymbol{y} \rangle \ , D_{\boldsymbol{w}}(\boldsymbol{y}) = \sum_i w_i y_i^2 \end{array} \tag{4.1}$$

The generating agent G_θ maps random noise $\boldsymbol{z} \sim \mathcal{N}(0, \boldsymbol{I})$ to samples while generating agent $G_p(\zeta)$, utilizing an independent source of randomness $\zeta \sim \mathcal{N}(0, 1)$, probabilistically controls the sign of the output of the generator G_θ. The probability of ultimately generating a sample $\boldsymbol{y} = \boldsymbol{z} + \boldsymbol{\theta}$ is equal to $\zeta + p$, while the probability of the sample being $\boldsymbol{y} = -\boldsymbol{z} - \boldsymbol{\theta}$ is equal to $1 - (p + \zeta)$.
On the other end, there stands the discriminating team of $D_{\boldsymbol{v}}, D_{\boldsymbol{w}}$. Discriminators, $D_v(\boldsymbol{y}), D_w(\boldsymbol{y})$ map any given sample $\boldsymbol{y}$ to a scalar value accounting for the realness or fakeness of it – negative meaning fake, positive meaning real. The discriminators are disparate in the way they measure the realness of samples as seen in their definitions.
We follow the formalism of the Wasserstein GAN to form the optimization objective:

$$\max_{\boldsymbol{v},\boldsymbol{w}} \min_{\boldsymbol{\theta},p} \left\{ \begin{array}{c} \mathbb{E}_{\boldsymbol{y}\sim real}\Big[D_{\boldsymbol{v}}(\boldsymbol{y}) + D_{\boldsymbol{w}}(\boldsymbol{y})\Big] \\ - \\ \mathbb{E}_{z\sim\mathcal{N}(0,\boldsymbol{I}),\zeta\sim\mathcal{N}(0,1)} \left[\left\{ \begin{array}{c} G_p(\zeta)\cdot\Big(D_{\boldsymbol{v}}\big(G_{\boldsymbol{\theta}}(\boldsymbol{y})\big) + D_{\boldsymbol{w}}\big(G_{\boldsymbol{\theta}}(\boldsymbol{y})\big)\Big) \\ + \\ \big(1 - G_p(\zeta)\big)\cdot\Big(D_{\boldsymbol{v}}\big(-G_{\boldsymbol{\theta}}(\boldsymbol{y})\big) + D_{\boldsymbol{w}}\big(-G_{\boldsymbol{\theta}}(\boldsymbol{y})\big)\Big) \end{array} \right\} \right] \end{array} \right\} \tag{4.2}$$

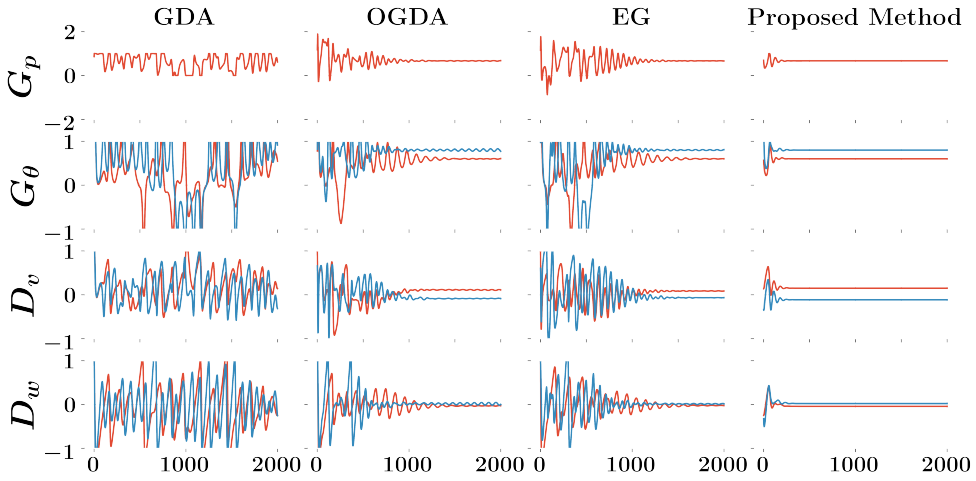


Figure 4.1: Parameter training of the configuration under different algorithms

Equation (4.2) yields the simpler form:

$$\max_{\boldsymbol{v},\boldsymbol{w}} \min_{\boldsymbol{\theta},p} (\pi_1 - \pi_2)\boldsymbol{v}^T\boldsymbol{\mu} - 2p\boldsymbol{v}^T\boldsymbol{\theta} + \boldsymbol{v}^T\boldsymbol{\theta} + \sum_i^n w_i(\mu_i^2 - \theta_i^2) \tag{4.3}$$

It is easy to check that Nash equilibria of Equation (4.2) must satisfy:

$$\begin{Bmatrix} \boldsymbol{\theta} &=& \boldsymbol{\mu}, & p = 1 - \pi_2 = \pi_1 \\ \boldsymbol{\theta} &=& -\boldsymbol{\mu}, & p = 1 - \pi_1 = \pi_2. \end{Bmatrix}$$

Figure 4.1 demonstrates both GDA's failure and OGDA, EG, and our KPV-GDA method's success to converge to the above Nash equilibria and simultaneously to discover the groud truth mixture.

4.4.2 Learning a mixture of Gaussians: a comparison between a vanilla GAN and a multi-GAN

In the previous part we compared our proposed novel optimizer with some established one. For this experiment we focus on the expressive power of multi-agent GANs. Our experiment includes a dataset of a Mixture of 2-D Gaussians with 8 modes. Our architecture includes 8 "shallow" generators and discriminators with 2 layers of 2-16-2 ReLU activations, compared with a "large" single-agent GAN with 4 layers of 2-128-256-1024-2 activations. Interestingly, the giant one fails in a double sense; It demonstrates both mode-collapsing and mode-drop phenomena without stabilizing. On the other hand, our architecture with a small number of neurons achieves to fit the data well.
The iterations illustrate the step at which the model roughly stops learning/converges. We observed no change after the demonstrated number of iterations.

Configuration	Generator(s)	Discriminator(s)	Typical outcome
Single GAN	# 1× { 2-128-256-1024-2 Linear w/ ReLU	# 1 × { 2-128-256-1024-2 Linear w/ ReLU	Mode coll. in $\sim 1 \times 10^3$ iters. Mode drop in $\sim 3 \times 10^3$ iters.
Multi-GAN	# 8× { 2-16-2 Linear w/ ReLU	# 8 × { 2-16-2 Linear w/ ReLU	Distr. learned in $\sim 3 \times 10^3$ iters.

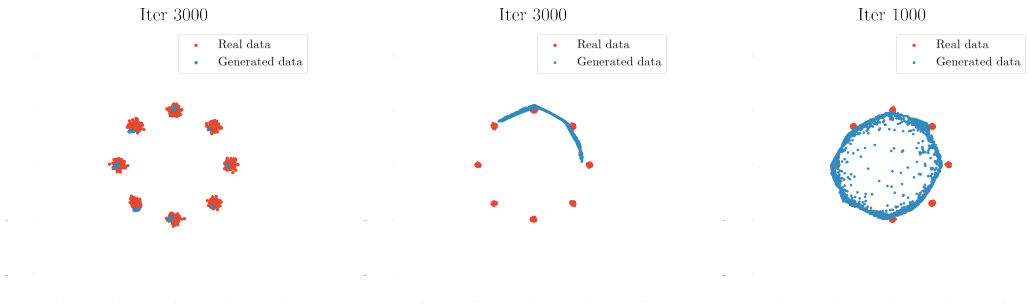


Figure 4.2: From left to right: (*i*) Each generator of MGAN learns one mode of 8-GMM, (*ii*) Mode Collapse of single-agent GANs, (*iii*) Single-agent GAN can't discriminate between the modes.

4.5 Experiments with real data

4.5.1 Mixture Generative Adversarial Nets

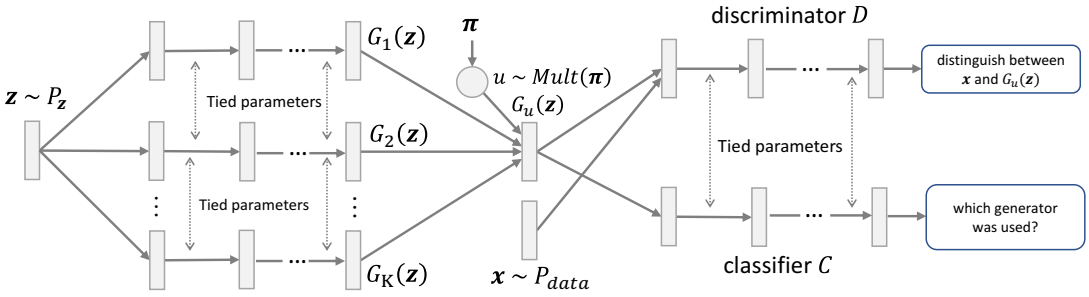


Figure 4.3: The Mixture Generative Adversarial Net.

Hoang et al. designed a multi-generator GAN that at the time of its release achieved remarkable performance in diverse datasets. The assemblage uses K generators with a fixed mixture vector $\pi = (\pi_1, \cdots, \pi_k)$, a classifier used to promote diversity of the samples (with a corresponding constant β) and a discriminator. The training objective is the following:

$$\min_{G_{1,\cdots,K}} \max_{D} \mathbb{E}_{\boldsymbol{z} \in p_{\text{data}}}[\log D(\boldsymbol{z})] - \mathbb{E}_{\boldsymbol{z} \in p_{\text{model}}}[(1 - \log D(\boldsymbol{z}))] - \beta \sum_{k} \pi_k \mathbb{E}_{\boldsymbol{z} \in p_{G_k}}[\log C_k(\boldsymbol{z})]. \tag{MGAN}$$

At the equilibrium of MGAN, the Jensen-Shannon divergence between D_{real} and D_{model} is minimized.

Table 4.1: Inception scores on different datasets. "–" denotes unavailable result.

Model	CIFAR-10	STL-10	ImageNet
Real data	11.24±0.16	26.08±0.26	25.78±0.47
WGAN [1]	3.82±0.06	–	–
MIX+WGAN [2]	4.04±0.07	–	–
Improved-GAN [52]	4.36±0.04	–	–
ALI [17]	5.34±0.05	–	–
BEGAN [7]	5.62	–	–
MAGAN [67]	5.67	–	–
GMAN [18]	6.00±0.19	–	–
DCGAN [47]	6.40±0.05	7.54	7.89
DFM [68]	7.72±0.13	8.51±0.13	9.18±0.13
D2GAN [44]	7.15±0.07	7.98	8.25
MGAN	**8.33±0.10**	**9.22±0.11**	**9.32±0.10**

Bibliography

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. «Wasserstein generative adversarial networks». In: *International conference on machine learning*. PMLR. 2017, pp. 214–223.

[2] Sanjeev Arora et al. «Generalization and equilibrium in generative adversarial nets (gans)». In: *International Conference on Machine Learning*. PMLR. 2017, pp. 224–232.

[3] Robert J Aumann. «Subjectivity and correlation in randomized strategies». In: *Journal of mathematical Economics* 1.1 (1974), pp. 67–96.

[4] Robert J Aumann and Michael Maschler. «Game theoretic analysis of a bankruptcy problem from the Talmud». In: *Journal of economic theory* 36.2 (1985), pp. 195–213.

[5] David Bellhouse. «The problem of Waldegrave». In: *Electronic Journal for the History of Probability and Statistics* 3.2 (2007), pp. 1–12.

[6] Jeremy Bentham. «An Introduction to the Principles of Morals and Legislation (1789), ed. by J». In: *H Burns and HLA Hart, London* (1970).

[7] David Berthelot, Thomas Schumm, and Luke Metz. «Began: Boundary equilibrium generative adversarial networks». In: *arXiv preprint arXiv:1703.10717* (2017).

[8] Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[9] George W Brown. *Some notes on computation of games solutions*. Tech. rep. RAND CORP SANTA MONICA CA, 1949.

[10] Yang Cai, Argyris Oikonomou, and Weiqiang Zheng. «Tight Last-Iterate Convergence of the Extragradient and the Optimistic Gradient Descent-Ascent Algorithm for Constrained Monotone Variational Inequalities». In: *arXiv preprint arXiv:2204.09228* (2022).

[11] Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

[12] Antoine Augustin Cournot. *Researches into the Mathematical Principles of the Theory of Wealth*. New York: Macmillan Company, 1927 [c1897], 1897.

[13] Constantinos Daskalakis and Ioannis Panageas. «The limit points of (optimistic) gradient descent in min-max optimization». In: *Advances in neural information processing systems* 31 (2018).

[14] Constantinos Daskalakis et al. «Training gans with optimism». In: *arXiv preprint arXiv:1711.00141* (2017).

[15] Jelena Diakonikolas, Constantinos Daskalakis, and Michael I Jordan. «Efficient methods for structured nonconvex-nonconcave min-max optimization». In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2021, pp. 2746–2754.

[16] Melvin Dresher. *Some military applications of the Theory of Games*. Tech. rep. RAND CORP SANTA MONICA CA, 1959.

[17] Vincent Dumoulin et al. «Adversarially learned inference». In: *arXiv preprint arXiv:1606.00704* (2016).

[18] Ishan Durugkar, Ian Gemp, and Sridhar Mahadevan. «Generative multi-adversarial networks». In: *arXiv preprint arXiv:1611.01673* (2016).

[19] Sigmund Freud. «Beyond the pleasure principle». In: *Psychoanalysis and History* 17.2 (2015), pp. 151–204.

[20] Noah Golowich, Sarath Pattathil, and Constantinos Daskalakis. «Tight last-iterate convergence rates for no-regret learning in multi-player games». In: *Advances in neural information processing systems* 33 (2020), pp. 20766–20778.

[21] Ian Goodfellow et al. «Generative adversarial nets». In: *Advances in neural information processing systems* 27 (2014).

[22] Elad Hazan et al. «Introduction to online convex optimization». In: *Foundations and Trends® in Optimization* 2.3-4 (2016), pp. 157–325.

[23] Martin Heusel et al. «Gans trained by a two time-scale update rule converge to a local nash equilibrium». In: *Advances in neural information processing systems* 30 (2017).

[24] Jean-Baptiste Hiriart-Urruty and Claude Lemaréchal. *Fundamentals of Convex Analysis*. Springer Berlin Heidelberg, 2001. ISBN: 978-3-540-42205-1. DOI: 10.1007/978-3-642-56468-0. URL: http://link.springer.com/10.1007/978-3-642-56468-0.

[25] Edwin Ho et al. «Game Theory in defence applications: a review». In: *Sensors* 22.3 (2022), p. 1032.

[26] Quan Hoang et al. «MGAN: Training generative adversarial nets with multiple generators». In: *International conference on learning representations*. 2018.

[27] Quan Hoang et al. «Multi-generator generative adversarial nets». In: *arXiv preprint arXiv:1708.02556* (2017).

[28] LK Jha and KN Jha. «Chanakya: the pioneer economist of the world». In: *International Journal of Social Economics* (1998).

[29] Howard Karloff. *Linear programming*. Springer Science & Business Media, 2008.

[30] Diederik P Kingma and Jimmy Ba. «Adam: A method for stochastic optimization». In: *arXiv preprint arXiv:1412.6980* (2014).

[31] R. Kleinberg, G. Piliouras, and É. Tardos. «Multiplicative Updates Outperform Generic No-Regret Learning in Congestion Games». In: *STOC*. 2009.

[32] Gerd Korman and Michael Klapper. «Game theory's wartime connections and the study of industrial conflict». In: *ILR Review* 32.1 (1978), pp. 24–39.

[33] Elias Koutsoupias and Christos Papadimitriou. «Worst-case equilibria». In: *Computer science review* 3.2 (2009), pp. 65–69.

[34] Chantale LaCasse and Don Ross. «The microeconomic interpretation of games». In: *PSA: Proceedings of the Biennial Meeting of the Philosophy of Science Association*. Vol. 1994. 1. Cambridge University Press. 1994, pp. 378–387.

[35] Claude Lemaréchal. «Cauchy and the gradient method». In: *Doc Math Extra* 251.254 (2012), p. 10.

[36] Aleksander Madry et al. «Towards deep learning models resistant to adversarial attacks». In: *arXiv preprint arXiv:1706.06083* (2017).

[37] R. Mehta, I. Panageas, and G. Piliouras. «Natural Selection as an Inhibitor of Genetic Diversity: Multiplicative Weights Updates Algorithm and a Conjecture of Haploid Genetics». In: *ITCS*. 2015.

[38] Ruta Mehta, Ioannis Panageas, and Georgios Piliouras. «Natural selection as an inhibitor of genetic diversity: Multiplicative weights updates algorithm and a conjecture of haploid genetics [working paper abstract]». In: *Proceedings of the 2015 Conference on Innovations in Theoretical Computer Science*. 2015, pp. 73–73.

[39] Panayotis Mertikopoulos et al. «Optimistic Mirror Descent in Saddle-Point Problems: Going the Extra (Gradient) Mile». In: *ICLR'19-International Conference on Learning Representations*. 2019, pp. 1–23.

[40] Panayotis Mertikopoulos et al. «Optimistic mirror descent in saddle-point problems: Going the extra (gradient) mile». In: *arXiv preprint arXiv:1807.02629* (2018).

[41] Katta G Murty and Santosh N Kabadi. *Some NP-complete problems in quadratic and nonlinear programming*. Tech. rep. 1985.

[42] John Nash. «Non-cooperative games». In: *Annals of mathematics* (1951), pp. 286–295.

[43] John F Nash et al. «Equilibrium points in n-person games». In: *Proceedings of the national academy of sciences* 36.1 (1950), pp. 48–49.

[44] Tu Nguyen et al. «Dual discriminator generative adversarial nets». In: *Advances in neural information processing systems* 30 (2017).

[45] Noam Nisan and Amir Ronen. «Algorithmic mechanism design». In: *Games and Economic behavior* 35.1-2 (2001), pp. 166–196.

[46] George R Price. «Fisher's 'fundamental theorem'made clear». In: *Annals of human genetics* 36.2 (1972), pp. 129–140.

[47] Alec Radford, Luke Metz, and Soumith Chintala. «Unsupervised representation learning with deep convolutional generative adversarial networks». In: *arXiv preprint arXiv:1511.06434* (2015).

[48] Sasha Rakhlin and Karthik Sridharan. «Optimization, learning, and games with predictable sequences». In: *Advances in Neural Information Processing Systems* 26 (2013).

[49] Julia Robinson. «An iterative method of solving a game». In: *Annals of mathematics* (1951), pp. 296–301.

[50] R Tyrrell Rockafellar. *Convex analysis*. Vol. 36. Princeton university press, 1997.

[51] Tim Roughgarden and Éva Tardos. «How bad is selfish routing?» In: *Journal of the ACM (JACM)* 49.2 (2002), pp. 236–259.

[52] Tim Salimans et al. «Improved techniques for training gans». In: *Advances in neural information processing systems* 29 (2016).

[53] Arthur L Samuel. «Some studies in machine learning using the game of checkers. II—recent progress». In: *Computer Games I* (1988), pp. 366–400.

[54] Leonard J. Schulman and Umesh V. Vazirani. «The duality gap for two-team zero-sum games». In: *Games Econ. Behav.* 115 (2019), pp. 336–345. DOI: 10.1016/j.geb.2019.03.011. URL: https://doi.org/10.1016/j.geb.2019.03.011.

[55] Claude E Shannon. «XXII. Programming a computer for playing chess». In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 41.314 (1950), pp. 256–275.

[56] David Silver et al. «A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play». In: *Science* 362.6419 (2018), pp. 1140–1144.

[57] David Silver et al. «Mastering the game of Go with deep neural networks and tree search». In: *nature* 529.7587 (2016), pp. 484–489.

[58] Maurice Sion. «On general minimax theorems.» In: *Pacific Journal of mathematics* 8.1 (1958), pp. 171–176.

[59] JMPGR Smith and George R Price. «The logic of animal conflict». In: *Nature* 246.5427 (1973), pp. 15–18.

[60] John Maynard Smith. *Evolution and the Theory of Games*. Cambridge university press, 1982.

[61] Bernhard von Stengel and Daphne Koller. «Team-maxmin equilibria». In: *Games and Economic Behavior* 21.1-2 (1997), pp. 309–321.

[62] Vasilis Syrgkanis et al. «Fast convergence of regularized learning in games». In: *Advances in Neural Information Processing Systems* 28 (2015).

[63] Christian Szegedy et al. «Rethinking the inception architecture for computer vision. 2015». In: *arXiv preprint arXiv:1512.00567* (2015).

[64] Gerald Tesauro et al. «Temporal difference learning and TD-Gammon». In: *Communications of the ACM* 38.3 (1995), pp. 58–68.

[65] Eric Van Damme. «Evolutionary game theory». In: *European Economic Review* 38.3 (1994), pp. 847–858. ISSN: 0014-2921. DOI: https://doi.org/10.1016/0014-2921(94)90121-X. URL: https://www.sciencedirect.com/science/article/pii/001429219490121X.

[66] John Von Neumann and Oskar Morgenstern. «Theory of games and economic behavior». In: (1947).

[67] Ruohan Wang et al. «Magan: Margin adaptation for generative adversarial networks». In: *arXiv preprint arXiv:1704.03817* (2017).

[68] David Warde-Farley and Yoshua Bengio. «Improving generative adversarial networks with denoising feature matching». In: (2016).

[69] Jörgen W Weibull. *Evolutionary game theory*. MIT press, 1997.

[70] Dao Li Zhu and Patrice Marcotte. «Co-coercivity and its role in the convergence of iterative schemes for solving variational inequalities». In: *SIAM Journal on Optimization* 6.3 (1996), pp. 714–726.