About Inspection Games

Nikolaos Karagiannis-Axypolitidis
R. N.: A0025

Examination committee:
Melolidakis Costis, Department of Mathematics, National and Kapodistrian University of Athens.

Burnetas Apostolos, Department of Mathematics, National and Kapodistrian University of Athens.

Markakis Evangelos, Department of Informatics, Athens University of Economics and Business.

Supervisor:
Melolidakis Costis, Associate Professor, Department of Mathematics, National and Kapodistrian University of Athens.

Μεταπτυχιακό Πρόγραμμα «Αλγόριθμοι, Λογική και Διακριτά Μαθηματικά» – 2016

Η παρούσα Διπλωματική Εργασία
εκπονήθηκε στα πλαίσια των σπουδών
για την απόκτηση του
Μεταπτυχιακού Διπλώματος Ειδίκευσης
«Αλγόριθμοι, Λογική και Διακριτά Μαθηματικά»
που απονέμει το
Τμήμα Πληροφορικής και Τηλεπικοινωνιών
του
Εθνικού και Καποδιστριακού Πανεπιστημίου Αθηνών

Εγκρίθηκε την .... από Εξεταστική Επιτροπή
αποτελούμενη από τους:

Ονοματεπώνυμο	Βαθμίδα	Υπογραφή
1. Μηλολιδάκης Κωστής	Αναπληρωτής Καθηγητής	. . . .
2. Μπουρνέτας Απόστολος	Καθηγητής	. . . .
3. Μαρκάκης Ευάγγελος	Αναπληρωτής Καθηγητής	. . . .

ABSTRACT

Inspection games are an area of application of Game Theory. An inspection game is a mathematical model of a situation where an inspector verifies if one other party, called violator, adheres to certain legal rules. Typically, the inspector's resources are limited so that verification can only be partial. A game-theoretic model may contribute in designing an optimal inspection scheme. In this scheme, we assume that an illegal action represents a strategic choice of the violator and an inspection represents a strategic choice of the inspector. Thus, one may define a game-theoretic problem, played in stages, with two players, the inspector and the violator.

The inspection game was first studied by Melvin Dresher and Michael Mashler in the 1960s. Several variants of inspection games have been studied since then. Besides the fundamental work of M. Dresher (1962) and M. Mashler (1966), we make a short presentation of relevant works of Bernhard von Stengel (2016), Minoru Sakaguchi (1977 and 1994) and Thomas Ferguson-Costis Melolidakis (1998).

The main object of the present thesis is to study the model introduced by Ferguson and Melolidakis (1998) and an open problem mentioned there. We came up with some interesting results in relation to the behavior of the players and, as a consequence, to the value of the game.

ΣΥΝΟΨΗ

Τα παίγνια επιθεώρησης αποτελούν ένα πεδίο εφαρμογών της Θεωρίας Παιγνίων. Το παίγνιο επιθεώρησης είναι η μαθηματική μοντελοποίηση της κατάστασης στην οποία ένας επιθεωρητής εξακριβώνει αν ένα αντίπαλο μέλος, ο παραβάτης, υπακούει σε συγκεκριμένους νόμιμους κανόνες. Οι πόροι του επιθεωρητή είναι περιορισμένοι, επομένως, η εξακρίβωση μπορεί να γίνει μόνο μερικώς. Ένα παιγνιοθεωρητικό μοντέλο μπορεί να συμβάλει στο σχεδιασμό ενός βέλτιστου σχήματος επιθεώρησης. Σε αυτό το σχήμα υποθέτουμε ότι μια παράνομη δράση και μια επιθεώρηση αντιπροσωπεύουν στρατηγικές επιλογές του παραβάτη και του επιθεωρητή, αντίστοιχα. Κατά αυτόν τον τρόπο, μπορού-με να ορίσουμε ένα παιγιοθεωρητικό πρόβλημα το οποίο παίζεται σε στάδια με δύο παίκτες, τον επιθεωρητή και τον παραβάτη.

Το παίγνιο επιθεώρησης μελετήθηκε πρώτη φορά από τους Melvin Dresher και Michael Mashler τη δεκαετία του 1960. Από τότε, διαφορές παραλλαγές παιγνίων επιθεώρησης έχουν γίνει αντικείμενο μελέτης. Εκτός από τις θεμελιώδεις εργασίες των M. Dresher (1962) και M. Mashler (1966), παρουσιάζουμε συνοπτικά τις σχετικές εργασίες των Bernhard von Stengel (2016), Minoru Sakaguchi (1977 και 1994) και Thomas Ferguson-Κωστής Μηλολιδάκης (1998).

Το κύριο αντικείμενο της παρούσας εργασίας είναι η μελέτη του μοντέλου που παρουσίασαν οι Ferguson-Μηλολιδάκης (1998) και του ανοικτού προβλήματος που αναφέρθηκε στη συγκεκριμένη εργασία. Από τη μελέτη αυτή, καταλήξαμε σε κάποια ενδιαφέροντα αποτελέσματα σε σχέση με τη συμπεριφορά των παικτών και, κατά συνέπεια, την τιμή του παιχνιδιού.

ΕΥΧΑΡΙΣΤΙΕΣ

Καταρχάς θα ήθελα να ευχαριστήσω από καρδιάς τον δάσκαλό μου και επιβλέποντα της διπλωματικής μου, τον καθηγητή Κωστή Μηλολιδάκη. Πρώτα και κύρια για την ευκαιρία που μου έδωσε από τα προπτυχιακά μου χρόνια να έρθω σε επαφή με τη Θεωρία Παιγνίων μέσω των μαθημάτων του αλλά και της προσωπικής μας συνεργασίας. Ακόμα περισσότερο, το τελευταίο διάστημα στην εκπόνηση της διπλωματικής μου εργασίας. Η βοήθειά του, οι "πόρτες" που μου άνοιγε στα διάφορα μονοπάτια της Θεωρίας Παιγνίων αλλά και η ουσιαστική του στήριξη και βοήθεια στην προσπάθειά μου, αποτέλεσαν καταλυτικούς παράγοντες στην ύπαρξη της παρούσης εργασίας.

Θα ήθελα επίσης να ευχαριστήσω τα μέλη της τριμελούς εξεταστικής επιτροπής της διπλωματικής μου εργασίας, τους καθηγητές Απόστολο Μπουρνέτα και Βαγγέλη Μαρκάκη για τη συμμετοχή τους, τη βοήθειά τους και την εν γένει συνεργασία μας. Ιδιαίτερα, θα ήθελα να αναφερθώ στον καθηγητή Απόστολο Μπουρνέτα. Εκτός από μέλος της επιτροπής, είναι και ο δεύτερος καθηγητής μου όλα αυτά τα χρόνια στο Μαθηματικό, μαζί με τον κύριο Μηλολιδάκη, ο οποίος με την αμέριστη βοήθειά του, με τις συμβουλές του και με τη διδασκαλία του δικού του Μαθηματικού αντικειμένου, συνέβαλε καθοριστικά στα βήματά μου.

Συνολικά, νιώθω την ανάγκη να ευχαριστήσω όλο το εκπαιδευτικό προσωπικό του τμήματος Μαθηματικών και του διιδρυματικού Προγράμματος Μεταπτυχιακών Σπουδών "Α.Λ.ΜΑ." αλλά και τους συμφοιτητές μου όλα αυτά τα χρόνια για όσα μου πρόσφεραν στην περιήγηση στον πανέμορφο και παράξενο κόσμο των Μαθηματικών.

Στην ολοκλήρωση της παρούσης εργασίας έπαιξαν καθοριστικό ρόλο τρεις φίλοι και γι' αυτό τους ευχαριστώ και μέσω αυτού του κειμένου: Η Βασιλίνα Μπισμπίκη κι ο Στέλιος Νταβέας με την κρίσιμη συμβολή τους στη γνωριμία μου με τις γλώσσες προγραμματισμού και την αξιοποίησή τους. Και ο Μάνθος Ψυρράκης με την αδελφική του έγνοια και παρότρυνση όλο αυτό το διάστημα δουλειάς.

Θεωρώντας την ολοκλήρωση αυτής της εργασίας ως το αποκρυστάλλωμα όλης της μέχρι τώρα μαθηματικής μου ζωής, ξέρω με σιγουριά ότι πολλά πράγματα δεν θα ήταν τόσο όμορφα χωρίς τους ανθρώπους με τους οποίους έχω ζήσει όλα αυτά τα χρόνια μέσα και έξω από τις αίθουσες του Μαθηματικού: Την Αλεξία, τον Αριστοτέλη, την Βασιλίνα, την Εμμανουέλα, τον Θωμά και τον Παναγιώτη.

Τα Μαθηματικά είναι πανέμορφα γιατί είναι δημιούργημα του Ανθρώπου που εξελίσσεται και προχωράει. Γιατί είναι κομμάτι της Γνώσης, αυτού του τεράστιου θησαυρού. Το σεντούκι αυτού του θησαυρού μου το άνοιξαν πρώτοι από όλους οι γονείς μου. Και θα τους ευγνωμονώ γι αυτό.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

The inspection game was originally proposed by M. Dresher (1962), and was futher elaborated by M. Mashler (1966) in more general terms in the context of checking possible treaty violations in arms control agreements.

The basic game presented by M. Dresher (1962) (using our notation and inserting the amount q) is played in n stages. Player I, the inspector, must allocate k inspections in n periods, i.e. choose the stages in which he will perform an inspection. Player II, the violator, may choose one of the stages to attempt performing an illegal act. If Player I is inspecting at the same stage when Player II acts, then Player I wins 1 unit and the game ends. If Player II acts when Player I is not inspecting, then the payoff is zero. If Player II decides not to act at any of the n stages, then Player I wins an amount q between these two values, $0 \leq q \leq 1$. The game is a zero-sum game. We assume that Player II knows k, and learns of every inspection before the next stage. By denoting this game as $\Gamma(n,k)$, with $0 \leq k \leq n, \quad n \geq 1$, we have the following recursive structure

$$\Gamma(n,k) = \begin{array}{r} \\ \text{inspect} \\ \text{don't inspect} \end{array} \begin{array}{c} \begin{array}{cc} \text{act} & \text{wait} \end{array} \\ \begin{pmatrix} 1 & \Gamma(n-1,k-1) \\ 0 & \Gamma(n-1,k) \end{pmatrix} \end{array} \tag{1.1}$$

for $0 < k < n$, and with boundary conditions

$$\Gamma(n,0) = (0) \quad \text{and} \quad \Gamma(n,n) = (q) \tag{1.2}$$

The problem is to find the value, $V_q(n,k)$, and the optimal strategies of the game $\Gamma(n,k)$.

The area of variations of the inspection game is vast. We can find several types of inspection games in relation to their payoff function, rules, number of players, number of allowed violations etc. For example, there are zero-sum games and non-constant-sum games. There are games with only 1 allowed violation and games where the violator can act more than once; games with full information and games without. Some inspection games can also be considered as Games with Finite Resources as in D. Gale (1957) and Th. Ferguson - C. Melolidakis (2000).

In Chapter 2, we consider 2-person games in which the probability of detection is 1 when an inspection and an illegal act take place simultaneously. We arrange them in two categories according to their ending rule and we present some models of such games:
• Games which end after the first violation is detected
• Games which may continue after the first violation is detected

Chapter 3 contains the main object of our survey. We present the Th. Ferguson - C. Melolidakis (1998) contribution and the open problem of finding a non-recursive solution for the model they present. Afterwards, based on numerical simulations, we present some conjectures concerning the value and the optimal strategies of this model, along with the methodology followed that brought these results.

In the Appendix, we cite the code (written in Python) used to build the program that ran instances of the Th. Ferguson - C. Melolidakis (1998) model.

CHAPTER 2

SOME INSPECTION GAMES

Some very interesting results for 2-person inspection games are produced by Michael Mashler (1966), Bernhard von Stengel (2016) and Minoru Sakaguchi (1977,1994).

Mashler applied the idea of $inspector's\ leadership$ to sequential inspections in a non-zero-sum game. It states that the inspector may commit himself to his inspection strategy in advance, and thereby gain an advantage compared to the situation where both players choose their actions simultaneously. It is of interest for future work to apply the $inspector's\ leadership$ to our model (see Chapter 3) changing the payoffs to a non-zero-sum game.

2.1 Games which end after the first violation is detected

2.1.1 Mashler model

Michael Mashler (1966) introduces an inspection game $\Gamma(n, k)$ on n stages as follows:

- non-constant-sum game.
- the inspector is permitted to commit k inspections, $0 \leq k \leq n$.
- the violator can attempt 1 illegal act at most.
- the violator learns whether an inspection has occurred on the event i after the event i and prior to the event $i + 1$.
- the inspector has the option to announce and commit himself to a specific mixed strategy, before playing the game ($inspector's\ leadership$).
- about payoffs:
  - If the violator violates on the event i and the inspector inspects, then the payoff is 0 to the violator and 1 to the inspector.
  - If the violator violates on the event i and the inspector does not inspect, then the payoff is 1 to the violator and 0 to the inspector.
  - If the violator does not act at any stage of the game, the payoffs are α for the

inspector and and β for the violator, $\alpha > 1$, $0 < \beta < 1$.[1] Thus, $\gamma = 1 - \beta$ measures how important it is to the violator to conduct a secret violation.

- **Notation:** denote $v_{n,k}$ and $w_{n,k}$ the payoff for Player I and Player II, respectively. Denote $h(r)$ the history of r events, i.e. $h(r) = \{i \mid \text{Player I inspects on stage i}\} \subseteq \{1, ..., r\}$. Accordingly, if history $h(r)$ has taken place, denote $q_{n^*,k^*}[h(r)]$ and $l_{n^*,k^*}[h(r)]$ the probability of inspecting and violating, respectively, on $(r+1)$-stage with n^* stages and k^* inspections remaining.

According to the assumptions above, the payoff matrix is:

$$\Gamma(n,k) = \begin{array}{r} \\ \text{inspect} \\ \text{don't inspect} \end{array} \begin{array}{c} \begin{array}{cc} \text{violate} & \text{don't violate} \end{array} \\ \begin{pmatrix} (1,0) & \Gamma(n-1,k-1) \\ (0,1) & \Gamma(n-1,k) \end{pmatrix} \end{array} \tag{2.1}$$

with boudary conditions

$$v_{n,0} = 0, \;\; v_{n,n} = \alpha, \;\; w_{n,0} = 1, \;\; w_{n,n} = 1 - \gamma \tag{2.2}$$

Mashler proved that Player I should announce his mixed strategy, i.e. $inspector's\ leadership$ leads to a better payoff for the inspector. As a result, both players will choose strategies for the zero-sum game based on the payoff to Player II. Mashler proved the followings in relation to the values and the optimal[2] strategies of $\Gamma(n,k)$:

Theorem 2.1. The expected payoffs $\mathrm{v}_{n,k}$ and $\mathrm{w}_{n,k}$ for the inspector and the violator, respectively, obtained by using optimal strategies in the game $\Gamma(n,k), 0 < k < n$ are

$$\mathrm{v}_{n,k} = \frac{\alpha P_{n-1,k-1}(\gamma)}{P_{n,k}(\gamma)} \tag{2.3}$$

$$\mathrm{w}_{n,k} = \frac{P_{n-1,k}(\gamma)}{P_{n,k}(\gamma)} \tag{2.4}$$

where

$$P_{n,0}(\gamma) = 1, \quad P_{n,n}(\gamma) = 1 \tag{2.5}$$

$$P_{n,k}(\gamma) = \sum_{i=0}^{k} \binom{n-k-1+i}{i} \gamma^i \tag{2.6}$$

[1]The assumption $\alpha > 1$ means that the inspector would prefer that no violation takes place than a situation in which a violation occurs and it is inspected. By $\beta < 1$ we mean that the violator prefers to act illegaly, instead of avoiding acting if he is sure that his violation will not be inspected. An inspection has a deterrent power, however, and the violator would change his preference if he were sure that an inspection would follow a violation. This accounts for $\beta > 0$.

[2]Notice that since this is a leader-follower game, we may speak of "optimal strategies" instead of just "equilibrium strategies".

Theorem 2.2. There is a unique pair of behavioral optimal strategies $q_{n^*,k^*}[h(r)]$ and $l_{n^*,k^*}[h(r)]$ to the game $\Gamma(n,k)$, of the type described by 2.1. Namely,

$$q_{n^*,k^*}[h(r)] = \begin{cases} 1 - \mathrm{w}_{n^*,k^*} & \text{if } 0 \le k^* < n \\ 1 & \text{if } k^* = n^* \end{cases} \tag{2.7}$$

$$l_{n^*,k^*}[h(r)] = \begin{cases} 1 & \text{if } k^* = 0 \\ 0 & \text{if } 0 < k^* \le n \end{cases} \tag{2.8}$$

Observation. *Theorems 2.1 and 2.2 lead to the solution of the Dresher game presented in equations 1.1 and 1.2 in the following way: If we remove the restriction $\alpha > 1$ and require $\alpha + \beta = 1$, then the equations 2.1 and 2.2 describe a constant-sum game (payoffs add to 1). Hence, finding the minimax strategies for the matrix game of Player's II payoffs (which according to Mashler's Theorem are optimal in his game and are provided in Theorems 2.1 and 2.2) is equivalent to solving the matrix game of Player's I payoffs. But that game is the Dresher game of equations 1.1 and 1.2 and hence Mashler's theorems solve that game also.*

Denoting by $V_q(n,k)$ the value of this game, equation 2.3 gives

$$V_q(n,k) = q\left(1 - \frac{\binom{n-1}{k} q^k}{\sum_{j=0}^{k} \binom{n-k-1+j}{j} q^j}\right) \tag{2.9}$$

The optimal mixed strategy for Player I is $\Big(V_q(n,k),\ 1 - V_q(n,k)\Big)$ and for Player II is $(1-Q,\ Q)$ where $Q = \dfrac{V_q(n,k)}{V_q(n-1,k)}$.

As we will see later, M. Sakaguchi (1977,1994) and Th. Ferguson - C. Melolidakis (1998) used this form to study their model of inspection game.

2.1.2 von Stengel's work

B. von Stengel (2016) works on an inspection game with multiple violations where every violation may have a different reward or penalty for the inspector and the violator. In his notation, we have:
- n stages
- k inspections, $0 \le k \le n$
- l intended[3] violations, $0 \le l \le n$
- a penalty parameter b
- reward parameters $r_l, ..., r_1, \quad r_i \ge 0$, where r_i is the reward for violating while the remaining violations are i.
- denote $v(n,k,l)$ and $w(n,k,l)$ the payoffs for the inspector and the violator, respectively.

Von Stengel introduces three variants of this game:

[3] The violator may not act at any stage or may attempt less than l violations

1. zero-sum game
2. non-zero-sum game where both the inspector and the violator receive negative payoff when a violation is caught
3. non-zero-sum game with $inspector's\ leadership$

- zero-sum case

  Considering the zero-sum model, we have the following game

$$\Gamma(n,k,l) = \begin{array}{c} \\ \text{inspection} \\ \text{no inspection} \end{array} \begin{array}{cc} \text{legal action} & \text{violation} \\ \left(\begin{array}{c} \Gamma(n-1,k-1,l) \\ \Gamma(n-1,k,l) \end{array}\right. & \left.\begin{array}{c} b \cdot r_l \\ \Gamma(n-1,k,l-1) - r_l \end{array}\right) \end{array} \tag{2.10}$$

  with boundary conditions

$$v(n,n,l) = 0, \quad v(n,0,l) = -\sum_{i=1}^{min\{l,n\}} r_{l+1-i} \tag{2.11}$$

  and $b > -1$.

  Then, the value of $\Gamma(n,k,l)$ is

$$v(n,k,l) = \frac{-t(n,k,l)}{s(n,k)}, \quad n > k, l > 0 \tag{2.12}$$

  where

$$s(n,k) = \sum_{i=0}^{k} \binom{n}{i} b^{k-i} \text{ and } t(n,k,l) = \sum_{i=1}^{l} r_{l+1-i} \binom{n-i}{k} \tag{2.13}$$

  The optimal behavioral strategies are $(p, 1-p)$ for the inspector and $(q, 1-q)$ for the violator, where

$$p = \frac{s(n-1,k-1)}{s(n,k)} \text{ and } \tag{2.14}$$

$$q = \frac{v(n-1,k,l) - v(n-1,k-1,l)}{v(n-1,k,l) - v(n-1,k-1,l) + b * r_l - v(n-1,k,l-1) + r_l}$$

  Another interesting result for the zero-sum case is the following:
  Let $\Gamma'(n,k,l)$ be the zero-sum game without full information where the inspector is not informed about the action of the violator after a time period without inspection. In $\Gamma(n,k,l)$, the inspector's strategy is independent of l (number of available violations). Hence, both the equilibrium payoff and the strategies of the $\Gamma(n,k,l)$ are valid in $\Gamma'(n,k,l)$ also.

- non-zero-sum case

  In the non-zero-sum case, von Stengel uses a penalty parameter $a \in (0,1)$ to model the cost that a caught violation has on the inspector. As in Mashler (1966), the inspector prefers that the violator does not act illegally instead of catching him. Von Stengel's non-zero-sum inspection game is given by

$$\hat{\Gamma}(n,k,l) =$$

$$\begin{array}{c} \\ \text{inspection} \\ \text{no inspection} \end{array} \begin{array}{c} \text{legal action} \qquad\qquad\qquad\qquad\qquad\qquad \text{violation} \\ \begin{pmatrix} \big(v(n-1,k-1,l), w(n-1,k-1,l)\big) & \big(-a \cdot r_l,\ -b \cdot r_l\big) \\ \big(v(n-1,k,l), w(n-1,k,l)\big) & \big(v(n-1,k,l-1) - r_l, w(n-1,k,l-1) + r_l\big) \end{pmatrix} \end{array} \tag{2.15}$$

  with boundary conditions

$$v(n,n,l) = w(n,n,l) = 0, \quad -v(n,0,l) = w(n,0,l) = \sum_{i=1}^{min\{l,n\}} r_{l+1-i} \tag{2.16}$$

  and $0 < a < 1, \quad b \geq 0$. The payoffs under optimal strategies for the inspector and the violator are given by

$$v(n,k,l) = \frac{-t(n,k,l)}{\hat{s}(n,k)} \quad \text{and} \quad w(n,k,l) = \frac{t(n,k,l)}{s(n,k)} \tag{2.17}$$

  where

$$\hat{s}(n,k) = \sum_{i=0}^{k} \binom{n}{i} (-a)^{k-i} \quad \text{and} \quad t(n,k,l),\ s(n,k) \ \text{ are as in 2.13} \tag{2.18}$$

  Optimal strategies are the same as with the zero-sum game (equations 2.14).

  Note that if we allow more general conditions $a < 1$ and $b > -1$, the zero-sum $\Gamma(n,k,l)$ is a special case of $\hat{\Gamma}(n,k,l)$ when $a = -b$.
  As in zero-sum, $\hat{\Gamma}'(n,k,l)$ without full information has the same equilibrium payoffs and strategies as $\hat{\Gamma}(n,k,l)$.

- $\hat{\Gamma}(n,k,l)$ with $inspector's\ leadership$

  As in the case of Mashler, von Stengel shows that the $inspector's\ leadership$ in $\hat{\Gamma}(n,k,l)$ leads to a better payoff $\big(u(n,k,l)\big)$ for the inspector.
  In particular,

$$u(n,k,l) = \frac{-t(n,k,l)}{s(n,k)} \quad \text{where} \quad t(n,k,l),\ s(n,k) \ \text{ are as in 2.13} \tag{2.19}$$

  As for the optimal strategies, Player I plays the optimal strategy of the simultaneous game $\hat{\Gamma}(n,k,l)$, $(p, 1-p)$ (as in 2.14). Player II acts legally while $k > 0$ and violates in every remaining stage if $k = 0$.

2.2 Games which may continue after the first violation is detected

Sakaguchi (1977,1994) considered a generalisation of Dresher's model (see Chapter 1). He studied a zero-sum inspection game in which Player II may act l times and the game stops after n periods. Sakaguchi's generalization of Dresher's problem may be described by having Player II pay an amount $q \in [0,1]$ for each non-action among the l. For example, we may think of Player II as having l loads of toxic waste that he may either despose legally of (cost q per load) or attempt to drop to a river. Denoting this game by $\Gamma(n,k,l)$, we have

$$\Gamma(n,k,l) = \begin{array}{c} \\ \text{inspect} \\ \text{don't inspect} \end{array} \begin{array}{c} \begin{array}{cc} \text{act} & \text{wait} \end{array} \\ \begin{pmatrix} 1+\Gamma(n-1,k-1,l-1) & \Gamma(n-1,k-1,l) \\ \Gamma(n-1,k,l-1) & \Gamma(n-1,k,l) \end{pmatrix} \end{array} \tag{2.20}$$

for $1 \leq k, l \leq n-1$. The boundary conditions are:

$$\Gamma(n,0,l) = (0) \quad \text{for } 0 \leq l \leq n,$$

$$\Gamma(n,k,0) = (0) \quad \text{for } 0 \leq k \leq n,$$

$$\Gamma(n,n,l) = (l \cdot q) \quad \text{for } 0 \leq l \leq n,$$

$$\Gamma(n,k,n) = (?) \quad \text{for } 0 \leq k \leq n-1. \tag{2.21}$$

Sakaguchi (1977,1994) solved this problem for both $q = 1$ and $q = \frac{1}{2}$. In the case where $q = 1$, the question mark in 2.21 is naturally replaced by k. In the case where $q = \frac{1}{2}$, Sakaguchi studies the solution of the game according to the implied boundary condition

$$\Gamma(n,k,n) = \left(n \cdot V_{\frac{1}{2}}(n,k)\right) \quad \text{for } 1 \leq k \leq n-1 \tag{2.22}$$

where $V_{\frac{1}{2}}(n,k)$ is as in 2.9.

Sakaguchi finds that the value of this game is l times the value when $l = 1$, for both cases where $q = 1$ and $q = \frac{1}{2}$, the latter under the conjecture that all the games have completely mixed optimal behavioral strategies (both players give positive weight to both actions). i.e., he states that for $q \in \left\{\frac{1}{2}, 1\right\}$ and question mark in 2.21 given by $n \cdot V_q(n,k)$,[4] the value of $\Gamma(n,k,l)$ is

$$V_q(n,k,l) = l \cdot V_q(n,k) \tag{2.23}$$

Th. Ferguson - C. Melolidakis (1998) have proven (see the following chapter) that Sakaguchi's conjecture about the games $\Gamma(n,k,l)$ being completely mixed is true. Moreover, they have proven that 2.23 holds for an arbitrary $q \in [0,1]$ when the boundary condition $\Gamma(n,k,n) = \left(n \cdot V_q(n,k)\right)$ is used.

[4] It can be easily seen that $n \cdot V_1(n,k) = k$. Also, Sakaguchi doesn't make clear what boundary condition he is using.

CHAPTER 3

FERGUSON-MELOLIDAKIS MODEL AND RESULTS

Th. Ferguson and C. Melolidakis (1998) clarified Sakaguchi's model and proved his result for an arbitrary $q \in [0,1]$. Furthermore, they suggested a more natural way of specifying the values of $\Gamma(n,k,n)$ by a 2x2 matrix game. This approach leads to the open problem of finding a non-recursive solution for $\Gamma(n,k,l)$ under this natural boundary condition.

In section 3.1 we present Ferguson-Melolidakis (1998) results. In section 3.2 we describe our simulation approach. In sections 3.3 and 3.4 we present the conclusions we draw while studying this open problem.

3.1 Model

The following scenario interprets the Ferguson-Melolidakis model: Player II (violator) has l truckloads of toxic waste to dispose of. There is no cost for dumbing a truckload of toxic waste in the river unless he gets caught by the inspector. In this case he loses +1 every time he is caught. Instead Player II may dispose of any truckload in a legal way at a cost of $q \in (0,1)$ per truckload. However, after n days, the inspector will inspect Player II homebase and force him to dispose legally of any waste found. In the meantime, Player II may try to dumb one truckload in the river every day. However, the inspector only has stuff enough to watch him on k of those days.

Denoting this game by $\Gamma(n,k,l)$, we have the matrix game:

$$\Gamma(n,k,l) = \begin{array}{c} \\ \text{inspect} \\ \text{don't inspect} \end{array} \begin{array}{cc} \text{act} & \text{wait} \end{array} \begin{pmatrix} 1+\Gamma(n-1,k-1,l-1) & \Gamma(n-1,k-1,l) \\ \Gamma(n-1,k,l-1) & \Gamma(n-1,k,l) \end{pmatrix} \quad (3.1)$$

for $1 \leq k, l \leq n-1$. The boundary conditions are:

$\Gamma(n,0,l) = (0)$ for $0 \le l \le n$,
$\Gamma(n,k,0) = (0)$ for $0 \le k \le n$,
$\Gamma(n,n,l) = (l \cdot q)$ for $0 \le l \le n$,

$\Gamma(n,k,n) =$

$$\begin{array}{c} \\ \text{inspect} \\ \text{don't inspect} \end{array} \begin{array}{c} \begin{array}{cc} \text{act} & \text{wait} \end{array} \\ \begin{pmatrix} 1+\Gamma(n-1,k-1,n-1) & q+\Gamma(n-1,k-1,n-1) \\ \Gamma(n-1,k,n-1) & q+\Gamma(n-1,k,n-1) \end{pmatrix} \end{array} \tag{3.2}$$

for $0 \le k \le n-1$.
In Theorem 3.3 below, it is shown that $\Gamma(n,k,n) = \Big(v(n,k)\Big)$, where

$$v(n,k) = k - (1-q)^{(n-k+1)} \sum_{j=0}^{k-1} (k-j) \binom{n-k-1+j}{j} q^j \tag{3.3}$$

3.1.1 Solution to Sakaguchi approach for an arbitrary q

Remember that Sakaguchi (1977,1994) studied the game $\Gamma(n,k,l)$ under the boundary condition $\Gamma(n,k,n) = \Big(n \cdot V_q(n,k)\Big)$ and $q \in \big\{1, \frac{1}{2}\big\}$.

Ferguson and Melolidakis (1998) first proved a Lemma which is used to show that all games $\Gamma(n,k,l)$, for $n \ge 2,\ \ 1 \le k,l \le n-1$, are completely mixed:

Lemma 3.1. Let $V_q(n,k)$, as described by 2.9, the value of $\Gamma(n,k)$. For all $n \ge 2$ and $1 \le k \le n$, we have

$$V_q(n,k) - V_q(n,k-1) \le \frac{1}{n}$$

Using the Lemma above, they proved the following result:

Theorem 3.2. If we choose as boundary condition $\Gamma(n,k,n) = \Big(n \cdot V_q(n,k)\Big)$, then for the value of the game $\Gamma(n,k,l)$ the following holds:

$$V_q(n,k,l) = l \cdot V_q(n,k) \ \text{ for } \ 0 \le k,l \le n \ \text{ and } \ q \in (0,1) \tag{3.4}$$

Proof. The boundary conditions 3.2 give the result for $k=0,\ k=n,\ l=0$ and $l=n$, for all $n \ge 2$. We must show 3.4 for all $n \ge 2,\ 1 \le k,l \le n-1$. We prove it by induction on n. The case $n=2$ follows from Mashler Theorem (equation 2.9). As the induction hypothesis, we assume 3.4 is true with n replaced by $n-1$. Now consider the case n and arbitrary $1 \le k,l \le n-1$. From 3.1 we have

$$V_q(n,k,l) = Value \begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix} \tag{3.5}$$

where

$$\begin{aligned} \alpha &:= 1 + V_q(n-1, k-1, l-1), \\ \beta &:= V_q(n-1, k-1, l), \\ \gamma &:= V_q(n-1, k, l-1), \\ \delta &:= V_q(n-1, k, l). \end{aligned}$$

The game is completely mixed if $\alpha > \beta,\ \beta < \delta,\ \delta > \gamma,\ \gamma < \alpha$. The first three of these inequalities follow easily from the induction hypothesis. The last inequality may be written by using the induction hypothesis as

$$V_q(n-1, k, l-1) = (l-1) \cdot V_q(n-1, k) < 1 + (l-1) \cdot V_q(n-1, k-1). \quad (3.6)$$

From Lemma 3.1, inequality 3.6 holds for all $1 \leq k, l \leq n-1$. Thus, the game is completely mixed and 3.5 reduces to

$$V_q(n, k, l) = \frac{\alpha\delta - \beta\gamma}{\alpha - \beta - \gamma + \delta} = \frac{l \cdot V_q(n-1, k)}{1 - V_q(n-1, k-1) + V_q(n-1, k)} \quad (3.7)$$

From the induction hypothesis we have: $V_q(n, k, 1) = V_q(n, k)$. From 3.7 we have

$$\begin{aligned} V_q(n, k, 1) = \frac{V_q(n-1, k)}{1 - V_q(n-1, k-1) + V_q(n-1, k)} &\Leftrightarrow \\ V_q(n, k) = \frac{V_q(n-1, k)}{1 - V_q(n-1, k-1) + V_q(n-1, k)} &\Leftrightarrow \\ V_q(n-1, k) = V_q(n, k)\Big(1 - V_q(n-1, k-1) + V_q(n-1, k)\Big)&. \end{aligned}$$

Hence,

$$V_q(n, k, l) = l \cdot \frac{V_q(n, k)\Big(1 - V_q(n-1, k-1) + V_q(n-1, k)\Big)}{1 - V_q(n-1, k-1) + V_q(n-1, k)} = l \cdot V_q(n, k)$$

and the induction is complete. □

3.1.2 $\Gamma(n, k, n)$ and the open problem

Ferguson and Melolidakis (1998) suggest to specify the values of $\Gamma(n, k, n)$ by assuming that the cost q is always assessed for every one of the l actions that are not taken. In particular, for a truckload the violator has now the following choices: either to act illegally and risk getting caught or to act legally and pay the taxes q. Therefore, $\Gamma(n, k, n)$ is specified through the recursive equations

$$\Gamma(n, k, n) =$$

$$\begin{array}{r} \\ \text{inspect} \\ \text{don't inspect} \end{array} \begin{pmatrix} \text{act} & \text{wait} \\ 1 + \Gamma(n-1, k-1, n-1) & q + \Gamma(n-1, k-1, n-1) \\ \Gamma(n-1, k, n-1) & q + \Gamma(n-1, k, n-1) \end{pmatrix} \quad (3.8)$$

for $1 \leq k \leq n-1$, with boundary conditions

$$\Gamma(n,0,n) = (0), \qquad \Gamma(n,n,n) = (n \cdot q) \tag{3.9}$$

They proved the following for the value of $\Gamma(n,k,n)$:

Theorem 3.3. Let $v(n,k)$ denote the value of $\Gamma(n,k,n)$. Then, for $0 \leq k \leq n$,

$$v(n,k) = k - (1-q)^{(n-k+1)}u(n,k), \tag{3.10}$$

where

$$u(n,k) = \sum_{j=0}^{k-1}(k-j)\binom{n-k-1+j}{j}q^j \tag{3.11}$$

For $0 < k < n$, the optimal mixed strategy for Player I is $(q, 1-q)$ and the optimal mixed strategy for Player II is $(Q, 1-Q)$, where $Q = v(n-1,k) - v(n-1,k-1)$.

Proof. Equations 3.8 and 3.9 become

$$v(n,k) = Value\begin{pmatrix} 1+v(n-1,k-1) & q+v(n-1,k-1) \\ v(n-1,k) & q+v(n-1,k) \end{pmatrix} \quad 1 \leq k \leq n-1 \tag{3.12}$$

subject to the boundary conditions

$$v(n,0) = 0 \text{ for } n \geq 1 \quad \text{and} \quad v(n,n) = n \cdot q \text{ for } n \geq 0 \tag{3.13}$$

It is easy to argue directly that $v(n,k-1) \leq v(n,k) \leq 1 + v(n,k-1)$ for $1 \leq k \leq n-1$, so the game is completely mixed. Therefore, Player I has the optimal mixed strategy $(x, 1-x)$ where

$$\begin{aligned} x &= \frac{q+v(n-1,k)-v(n-1,k)}{1+v(n-1,k-1)-\big(q+v(n-1,k-1)\big)-v(n-1,k)+\big(q+v(n-1,k)\big)} \\ &= q \end{aligned}$$

and Player II has the optimal mixed strategy $(y, 1-y)$ where

$$\begin{aligned} y &= \frac{q+v(n-1,k)-\big(q+v(n-1,k-1)\big)}{1+v(n-1,k-1)-\big(q+v(n-1,k-1)\big)-v(n-1,k)+\big(q+v(n-1,k)\big)} \\ &= v(n-1,k)-v(n-1,k-1) \end{aligned}$$

Equation 3.12 reduces to

$$v(n,k) = q\Big(1+v(n-1,k-1)\Big) + (1-q)v(n-1,k) \quad \text{for } 1 \leq k \leq n-1 \tag{3.14}$$

We can simplify the equations 3.13 and 3.14 by changing the functions 3.11 from $v(n,k)$ to $u(n,k)$. Equations 3.13 and 3.14 reduce to

$$u(n,k) = q \cdot u(n-1,k-1) + u(n-1,k) \quad \text{for } 1 \leq k \leq n-1, \tag{3.15}$$

subject to the boundary conditions

$$u(n,0) = 0 \text{ for } n \geq 1 \quad \text{and} \quad u(n,n) = n \text{ for } n \geq 0. \tag{3.16}$$

The solution is given by 3.10 for $0 \leq k \leq n$. This can be easily seen by checking that it satisfies 3.15 and 3.16. □

It is important to notice that, for $q \in (0,1)$, if we choose $\Gamma(n,k,n) = \Big(v(n,k)\Big)$ as the boundary condition, it is an open problem to find a non-recursive solution for $\Gamma(n,k,l)$.

3.2 Our method

To study properties of the values and of the optimal strategies for $\Gamma(n,k,l)$, we use numerical methods for various ranges of n, k, l and q.

An algorithm using Python was created to export the results. The code can be found in the appendix.

The output for the code are the values and the optimal strategies of the selected $\Gamma(n,k,l)$ according to n, k, l, q.

We study the results for all $\Gamma(n,k,l)$ with $n \leq 200,\ 1 \leq k, l \leq n-1,\ q \in \{0.1, 0.15, 0.2, 0.25, ..., 0.9\}$. That makes 45.667.117 values and 45.328.817 pairs of optimal strategies (there are no strategies for boundary conditions).

Notation:

- $\tilde{V}_q(n,k,l)$: value of $\Gamma(n,k,l)$ defined by 3.1 and 3.2.
- $(x, 1-x)$ the optimal mixed strategy for the inspector.
- $(y, 1-y)$ the optimal mixed strategy for the violator.

Recall that

Definition 3.4 (Correlation Coefficient)**.** Given a pair of random variables (X,Y) the *correlation coefficient between X and Y* is:

$$r_{X,Y} = \frac{\text{cov}(X,Y)}{\sigma_X \cdot \sigma_Y}$$

where $cov(X,Y)$ is the *covarience* of X and Y, σ_X and σ_Y are the *standard deviations* of X and Y, respectively.

The correlation coefficient ranges from -1 to 1. A value of 1 implies that a linear equation describes the relationship between X and Y perfectly, with all data points lying on a line for which Y increases as X increases. A value of -1 implies that all data points lie on a line for which Y decreases as X increases.

We find the correlation coefficients between the variables of the following pairs: $\left(\tilde{V}_q(n,k,l), k\right)$, $\left(\tilde{V}_q(n,k,l), l\right)$, (x,k), (x,l), (y,k), (y,l).

Correlation coefficients are used to study the relation between value/strategies and inspections/violations. The values n and q are kept constant and for all q we study the cases $\Gamma(10,k,l),\ \Gamma(50,k,l),\ \Gamma(100,k,l),\ \Gamma(200,k,l),\ 1 \leq k, l \leq n-1$.

We created two kinds of graphs to study the outputs. One refers to the values of the game (value graphs) and the other refers to the optimal strategies of the game (strategy graphs).

The vertical axis of the graphs refers to values (for value graphs) and probability of every player's 1st strategy (for strategy graphs).

The horizontal axis refers to the output results generated by the code. That means, depending on the order of the for-loops, that the $x_0 - 1$ point of the horizontal axis represents the x_0 in the row result.

As an example let us see the representation of $\Gamma(3,k,l)$ with $q = 0.4$ (the results are shown in tables 4.1 and 4.2 of the Appendix):
Let

```
for k in range(1,2):
        for l in range(1,2):
```

be the order of for-loops in the code. Then, the results are encountered in sequence as follows:
$\Gamma(3,1,1)$
$\Gamma(3,1,2)$
$\Gamma(3,2,1)$
$\Gamma(3,2,2)$

The corresponding graphs are:

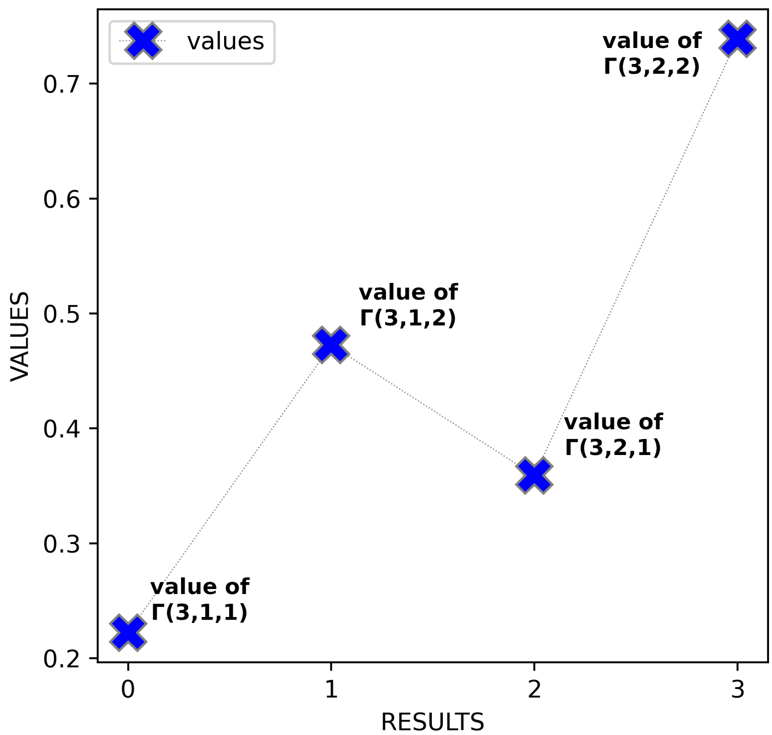

Figure 3.1: values for $\Gamma(3,k,l),\ q = 0.4,\ k,l \in \{1,2\}$ with k-loop prior to l-loop

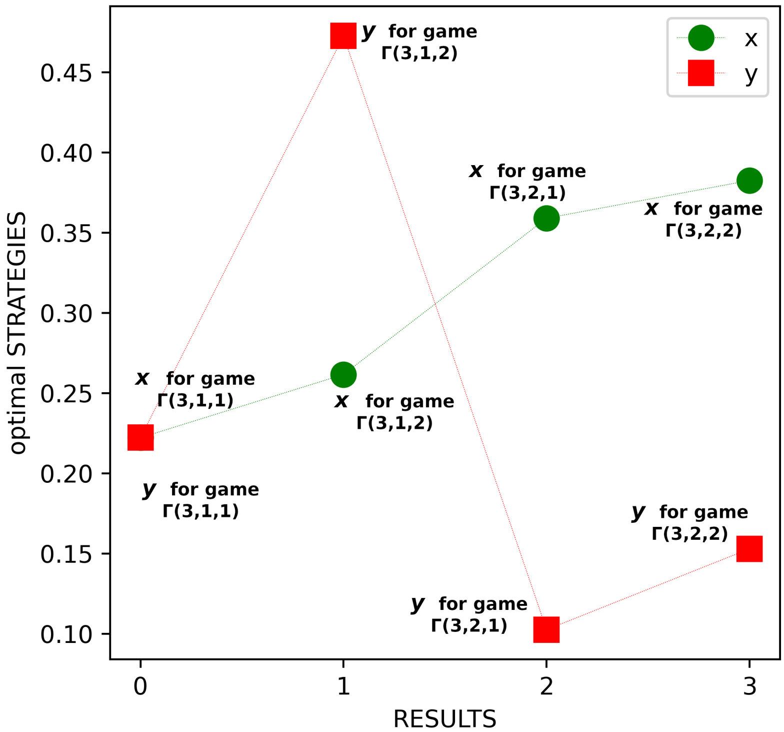

Figure 3.2: optimal strategies for $\Gamma(3, k, l),\ q = 0.4,\ k, l \in \{1, 2\}$ with k-loop prior to l-loop. x is illustrated with green circles and y is illustrated with red squares

In Figure 3.1, we can see that the value of the game $\Gamma(3, 2, 1)$ is $\tilde{V}_{0.4}(3, 2, 1) = 0.35897$. Optimal strategies of this game can be shown in Figure 3.2: x=0.3589743590, i.e. Player I "inspects" with a probability of 0.3589743590 and y=0.1025641026, i.e. Player II "acts" with a probability of 0.1025641026 in the game $\Gamma(3, 2, 1)$.

On the other hand, if the order of the for-loops in the code is

```
for l in range(1,2):
        for k in range(1,2):
```

the corresponding graphs are:

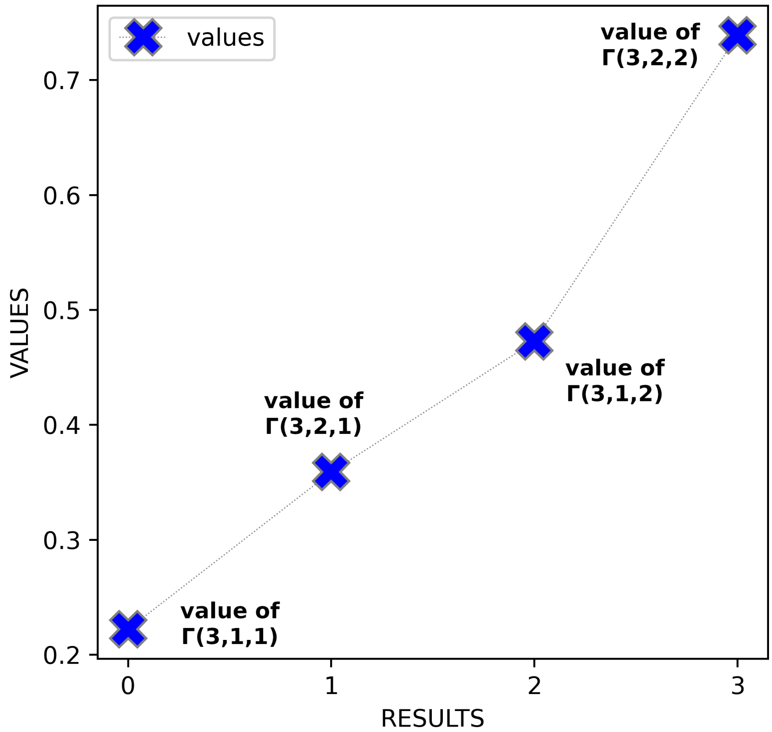

Figure 3.3: values for $\Gamma(3, k, l),\ q = 0.4,\ k, l \in \{1, 2\}$ with l-loop prior to k-loop

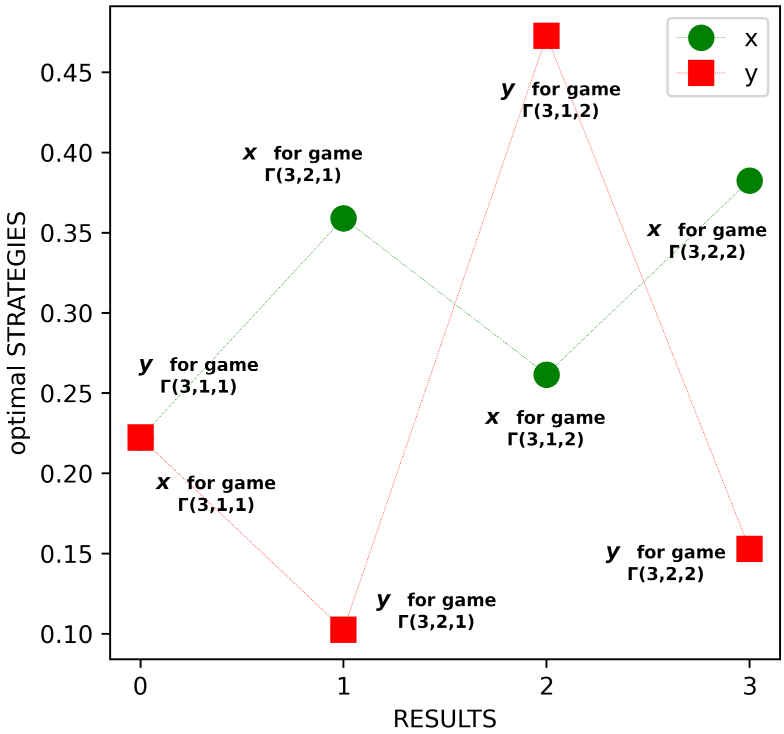

Figure 3.4: optimal strategies for $\Gamma(3, k, l),\ q = 0.4,\ k, l \in \{1, 2\}$ with l-loop prior to k-loop. x is illustrated with green circles and y is illustrated with red squares

As we can see in Figure 3.3, the value of $\Gamma(3, 2, 1)$ is the second result in the row. Accordingly, we see the optimal strategies of this game as the second result in the row in Figure 3.4.

3.3 On the threshold k*

3.3.1 Results, approach, interpretation of results

The following important conclusion was drawn by studying the produced graphs: For any game $\Gamma(n, k, l)$ there is a number of inspections (denoted by k^*) such that if the available inspections are more than k^*, players tend to act in the same fashion everyday, namely, if $k \geq k^*$ the inspector inspects with a probability very close to q and the violator acts with probability close to 0. The value of the threshold k^* depends on the number of days (n) and the value of q. Also, the threshold k^* affects the value of the game. Thus, the value tends to be $l \cdot q$ as the number of the available inspections is greater than the threshold k^*.

To further illustrate our point about the existence of this threshold k^*, let's see the values and the optimal strategies of the games $\Gamma(10, k, l),\ 1 \le k, l \le 9$ with $q = 0.1, q = 0.5$ and $q = 0.9$ [1]:

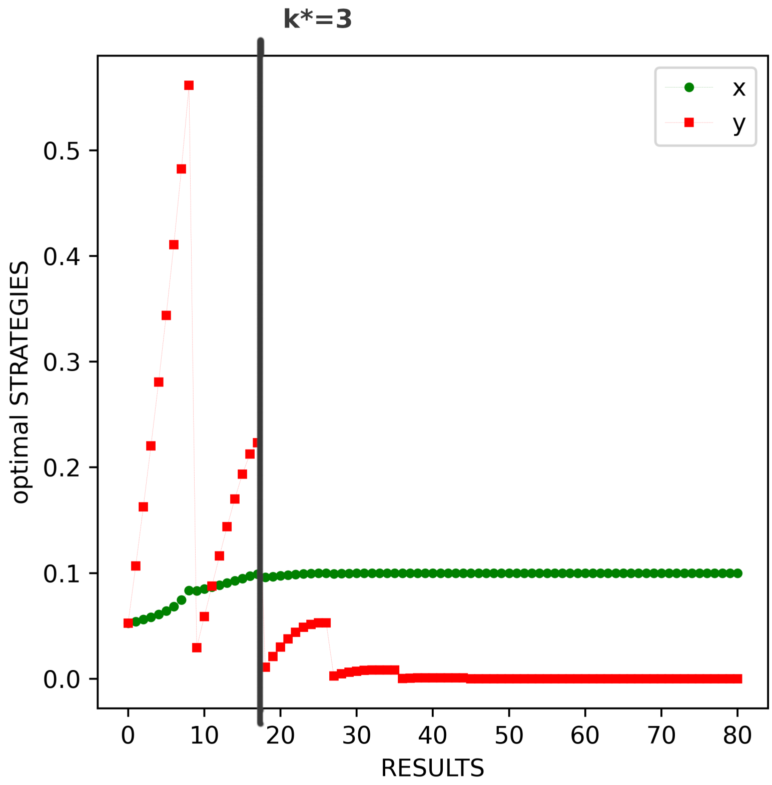

Figure 3.5: strategies for $\Gamma(10, k, l),\ q = 0.1$ (k-loop prior to l-loop)

In Figure 3.5 we see the optimal strategies of the inspector (illustrated with green circles) and the violator (illustrated with red squares) for the games $\Gamma(10, k, l),\ 1 \le k, l \le 9$ with $q = 0.1$. The games are produced in the following order:

$$\begin{array}{c} \Gamma(10,1,1),\ \ \Gamma(10,1,2), ..., \Gamma(10,1,9), \\ \Gamma(10,2,1),\ \ \Gamma(10,2,2), ..., \Gamma(10,2,9), \\ \Gamma(10,3,1),\ \ \Gamma(10,3,2), ..., \Gamma(10,3,9), \\ \vdots \\ \Gamma(10,9,1),\ \ \Gamma(10,9,2), ..., \Gamma(10,9,9). \end{array}$$

There is a point on the graph (noted with the black vertical line) after which x is stabilised apporximately at 0.1 (value of q). At this point, the number of available inspections becomes 3, i.e. we take results for game $\Gamma(10, 3, l)$ and further. Simulta-

[1]The graphs for games $\Gamma(10, k, l)$ are shown to facilitate the reading of the present thesis. Same results occur for all games $\Gamma(n, k, l),\ n \le 200,\ 1 \le k, l \le 199$.

neously, y decreases considerably and gets very close to 0. Hence, for $\Gamma(10, k, l)$ with $q = 0.1$, we have $k^* = 3$.

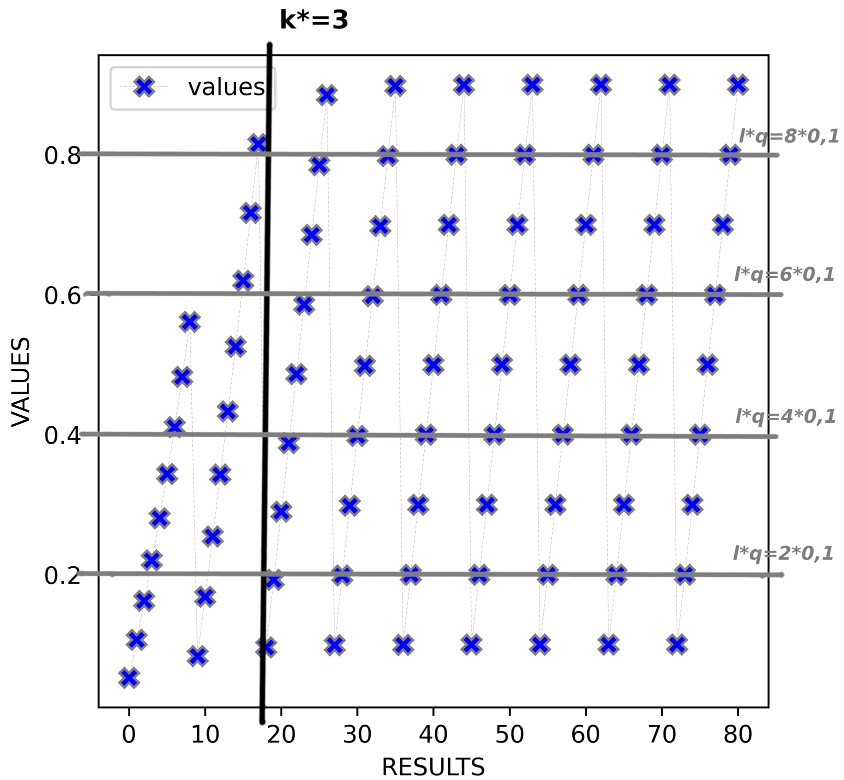

Figure 3.6: values for $\Gamma(10, k, l),\ q = 0.1$ (k-loop prior to l-loop)

In Figure 3.6, we tabulate the corresponding values for the games of Figure 3.5. The horizontal lines denote the value of $l \cdot q$ for each different value of l. We can see that if $k \geq k^* = 3$, the values of the games are approximately equal to $l \cdot q$.

It is interesting to see what happens if we increase q. In Figures 3.7 and 3.8 below, we depict the games $\Gamma(10, k, l),\ 1 \leq k, l \leq 9$ with $q = 0.5$. In Figures 3.9 and 3.10 we depict the games $\Gamma(10, k, l)$ with $q = 0.9$.

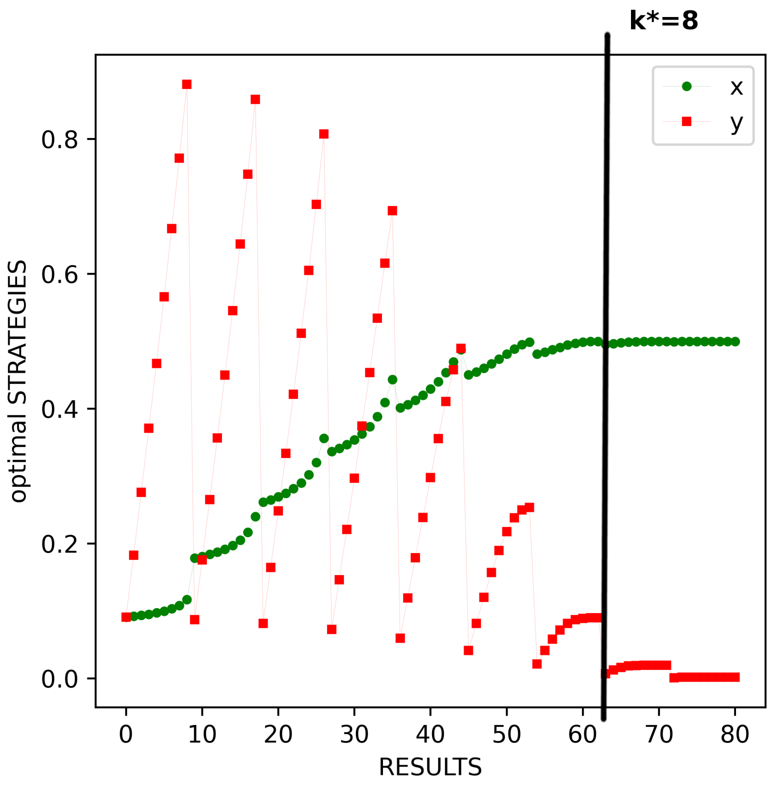

Figure 3.7: strategies for $\Gamma(10, k, l),\ q = 0.5$ (k-loop prior to l-loop)

In Figure 3.7 we present the optimal strategies for the games $\Gamma(10, k, l)$ with $q = 0.5$. The games are produced in the same order as in Figures 3.5 (k-loop prior to l-loop). We can see that k^* is larger than the case $q = 0.1$. For $q = 0.1$, we have that if $k \geq 3$, x is stabilised approximately at 0.1. Now, for $q = 0.5$ and $3 \leq k < 8$, we can see that x is considerably less than 0.5. On the other hand, if $k \geq 8$, x is stabilised approximately at $q = 0.5$. Therefore, $k^* = 8$ in that case.

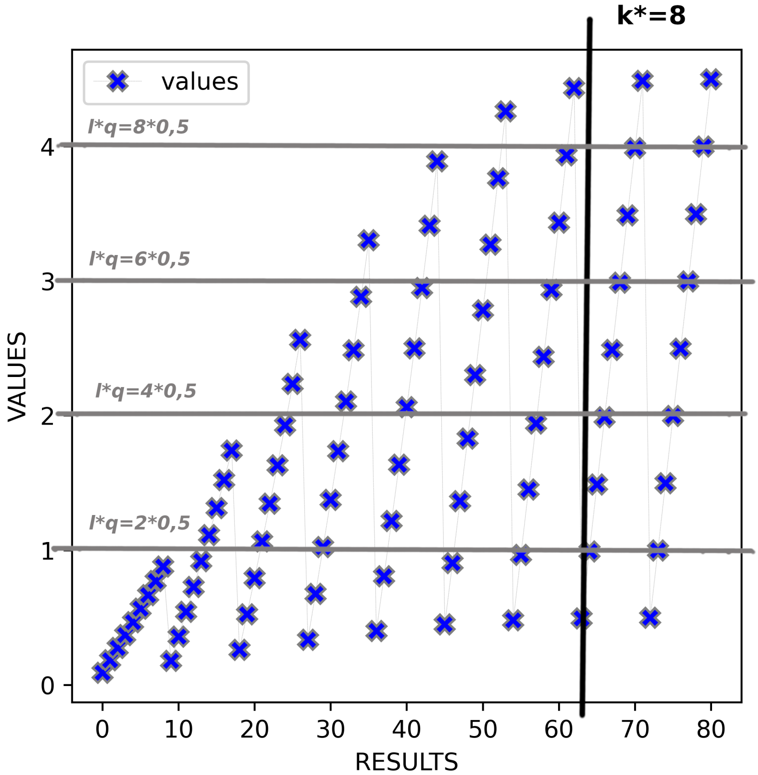

Figure 3.8: values for $\Gamma(10, k, l),\ q = 0.5$ (k-loop prior to l-loop)

Considering the values of $\Gamma(10, k, l),\ 1 \leq k, l \leq 9$ with $q = 0.5$, we see that if $k \geq 8$, they are approximately equal to $l \cdot q$. Horizontal lines on the graph above show this result.

Finally, if we take the game $\Gamma(10, k, l),\ 1 \leq k, l \leq 9$ with $q = 0.9$, there is no point in considering such a k^*. In a trivial way, we can claim that $k^* = 9$ in that case. We can see the values and the optimal strategies of these games on the graphs below.

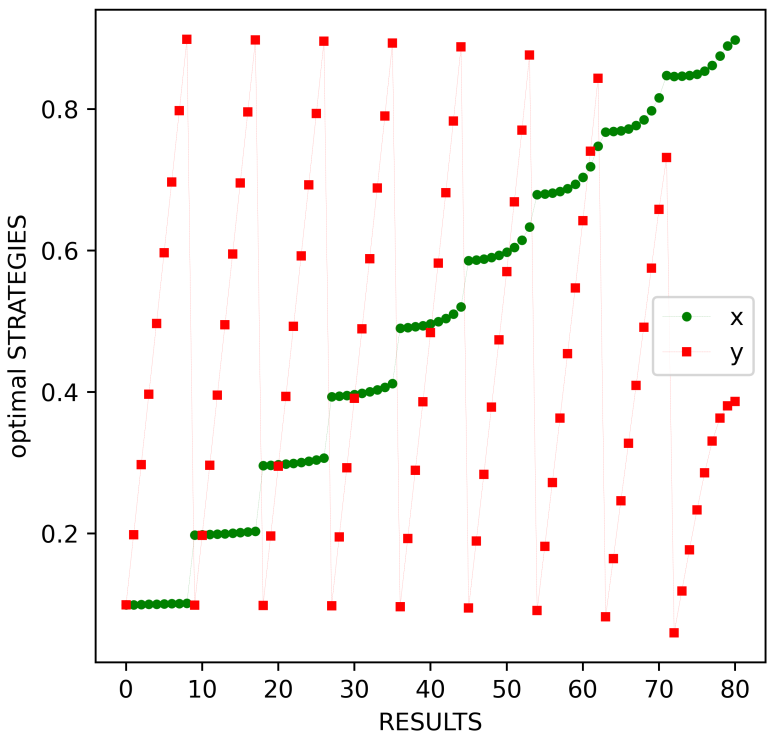

Figure 3.9: strategies for $\Gamma(10, k, l),\ q = 0.9$ (k-loop prior to l-loop)

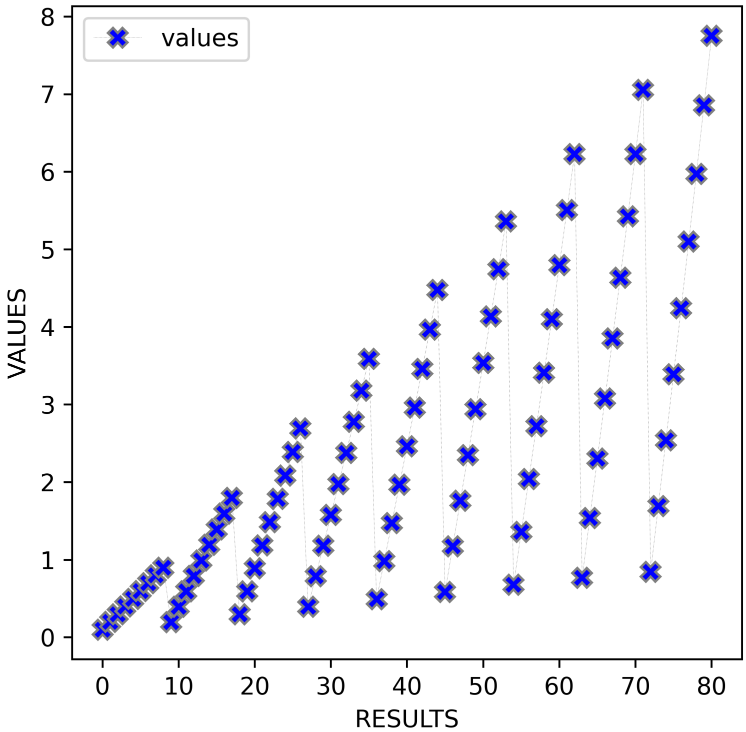

Figure 3.10: values for $\Gamma(10, k, l),\ q = 0.9$ (k-loop prior to l-loop)

Figures 3.5, 3.7 3.9 lead to the conjecture that as q increases, k^* increases. This means that the decision of the inspector is sensitive to k for a longer range of k's as q increases.

Our conjecture may be intuitively explained:

- For small value of q and large value of k, it does not make sense for the violator to risk getting caught by the inspector. Hence, he violates with a very low probability (approximately 0) and does so until the end of the game and then, he pays a tax q for every violation. Correspondingly, the inspector inspects with low probability (equal to q) in order to save inspections and not give the violator the chance to act without getting inspected. Essentially, if the inspector has more inspections than this specific value of k^*, both players try to lead the game to a boundary condition $\Gamma(n, n, l)$ waiting for the number of days to be equal to the number of available inspections. Then, the inspector gets $l \cdot q$ from the violator.

  On the other hand, if the available inspections are less than k^*, then the violator's probability for acting increases considerably since it is easy to escape detection. This probability depends on l (we can see that by the correlation coefficients of Table 3.3 in the following section (see section 3.3.2)). At the same time, the inspector inspects with an even lower probability because of the very few available inspections. Therefore, the value of the game is less than $l \cdot q$.

- As q increases, then Player II's appetite for risk increases. That explains why the value of k^* increases with q. Hence, if q is close to 1, then the violator has no gain by leading the game to the boundary condition and paying $l \cdot q$. On the other hand, the inspector's strategy depends on the number of available inspections: When k is higher than k^*, x is very close to q, i.e. the inspector inspects with a higher probability. For $k < k^*$, even when the number of available violations is high, the inspector does not risk to waste his inspections eventhough the violator violates with high probability.

Example 3.5. *Let's examine the game* $\Gamma(7, 4, 4)$ *with* $q = 0.1$. *Then:*

- $\tilde{V}_{0.1}(7, 4, 4) = 0.39982$

- *optimal strategies:* $(x, 1 - x) = (0.1, 0.9)$ *for the inspector and* $(y, 1 - y) = (0.00127, 0.99873)$ *for the violator.*

We denote by $k^*(n)$ the threshold k^* for every n in the subgames $\Gamma(n, k, l)$ of the game $\Gamma(7, 4, 4)$ with $q = 0.1$. The optimal strategies and values of these subgames are shown on the graphs below. The subgames of $\Gamma(7, 4, 4)$ are produced (and represented on the horizontal axis of the graphs) in the same order they are encountered in the game (and k-loop prior to l-loop in the code): $\Gamma(7, 4, 4)$, $\Gamma(6, 4, 4)$, $\Gamma(6, 4, 3)$, $\Gamma(6, 3, 4)$, $\Gamma(6, 3, 3)$, $\Gamma(5, 4, 4)$ and so on.

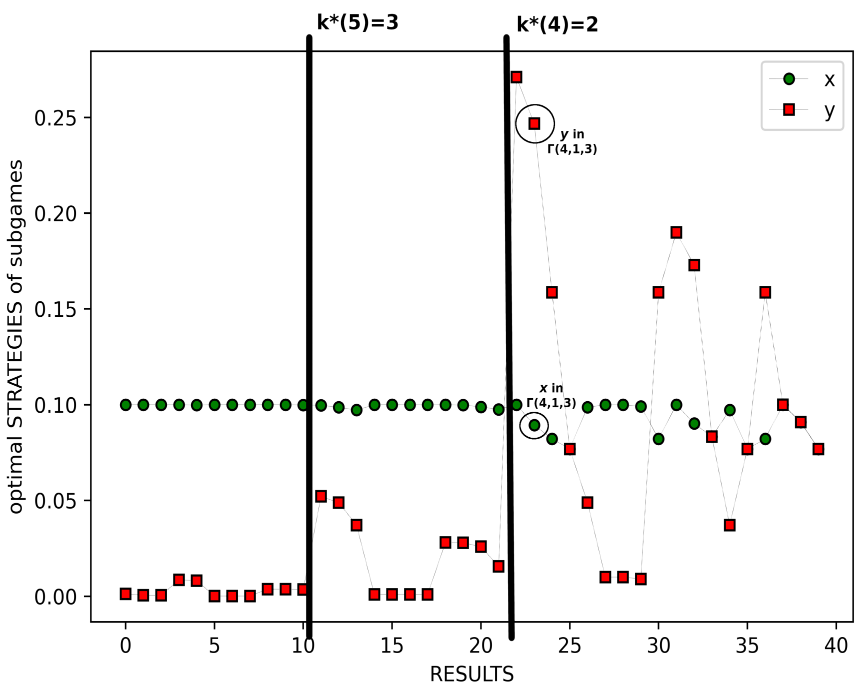

Figure 3.11: optimal strategies for subgames of the game $\Gamma(7, 4, 4)$, $q = 0.1$

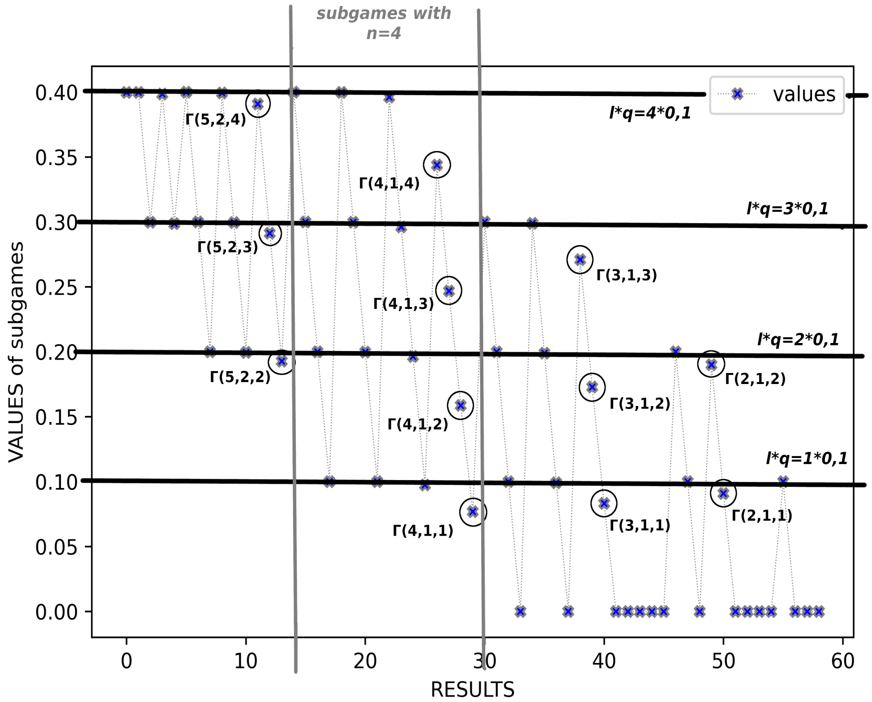

Figure 3.12: values for subgames of the game $\Gamma(7, 4, 4),\ q = 0.1$

In Figure 3.11 the probability of inspection is approximately q and the probability of violation is close to 0, unless available inspections are less than $k^*(n)$. Remember that in this example we have a single game and its subgames, thus k decreases as the game carries on. In Figure 3.12 the horizontal lines denote the value of $l \cdot q$ for each different value of l. We can see that the cases with value less than $l \cdot q$ are those where the available inspections are less than $k^*(n)$. For example, if $n = 4$ (15th to 30th result in the row), the value is less than $l \cdot q$ when $k < k^*(4) = 2$. In the game $\Gamma(7, 4, 4)$, we have $k^*(5) = 3,\ k^*(4) = 2,\ k^*(3) = 2$.

3.3.2 More numerical results

As we have already seen, the value of k^* depends on n and q.
We present some approximate values for k^* for various values of n and q. These results were tabulated using plots for games $\Gamma(n, k, l)$ for various values of n and q with $l \in \{1, ..., n-1\}$.

approximate values for k*							
	n=10	**n=20**	**n=30**	**n=40**	**n=50**	**n=100**	**n=200**
q=0,1	3	5	8	10	11	17	29
q=0,2	5	8	11	14	16	28	50
q=0,3	6	9	15	18	22	39	71
q=0,4	7	12	18	22	26	49	91
q=0,5	8	14	20	26	31	58	111
q=0,6	-	16	23	29	36	67	130
q=0,7	-	18	25	32	40	77	148
q=0,8	-	-	27	36	44	85	167
q=0,9	-	-	-	38	47	93	185

Table 3.1: approximate values for k^*

In examining the relation between the strategies/values and the days of inspections/violations, we present the correlation coefficients between them for all games $\Gamma(200, k, l),\ 1 \leq k, l \leq 199$ and various values of q:

correlation coefficients for values			
		remaining inspections	**remaining violations**
q=0,1	**value**	0.144433163230141	0.939427451863154
q=0,2	**value**	0.261318352092331	0.885272329055803
q=0,3	**value**	0.261318352092331	0.885272329055803
q=0,4	**value**	0.440868557171004	0.796003087527588
q=0,5	**value**	0.507877646443595	0.759230230615875
q=0,6	**value**	0.561466952979961	0.727168270871774
q=0,7	**value**	0.602458501171036	0.699841041048143
q=0,8	**value**	0.631412647830027	0.677721210535321
q=0,9	**value**	0.648745532969464	0.662003138331736

Table 3.2: correlation coefficients for values

Let us look at the first row of Table 3.2: For $q = 0.1$ the correlation coefficient between the value $\tilde{V}$ and the remaining inspections k is $r_{\tilde{V},k} = 0.144433163230141$ and between the value and the remaining violations l is $r_{\tilde{V},l} = 0.939427451863154$. Hence, as l increases, the value increases in an almost linear way with l ($r_{\tilde{V},l}$ is almost 1). On the other hand, an increase on k doesn't affect considerably the value of the game ($r_{\tilde{V},k}$ is very close to 0). In this case, the violator will act with low probability and, hence, the number of available inspections doesn't make a big difference to the value of the game.

For $q = 0.9$ this difference is being balanced. We could expect such a result: For a price of q close to 1, as we have already seen, the violator's appetite for risk increases. As a consequence, the number of available inspections affect the value of the game in a more decisive way due to a higher probability of violation.

correlation coefficients for strategies			
		remaining inspections	**remaining violations**
q=0,1	**x**	0.47185606343993	0.023584282595143
	y	-0.481573928613907	0.17924228643491
q=0,2	**x**	0.63767898077733	0.019363351973344
	y	-0.6147854905114	0.257822370412602
q=0,3	**x**	0.751370051197025	0.016574116448593
	y	-0.681986104322285	0.327212972550611
q=0,4	**x**	0.83476531543904	0.014361087204536
	y	-0.705353606191625	0.395192700927271
q=0,5	**x**	0.89658137265717	0.012417628145705
	y	-0.692163275319973	0.465778828305129
q=0,6	**x**	0.941366065004394	0.010579050870256
	y	-0.644144791392238	0.542350760128037
q=0,7	**x**	0.971912198227998	0.00871828983524
	y	-0.559447087949674	0.628767506712757
q=0,8	**x**	0.990276246342418	0.006687810663084
	y	-0.432505798314449	0.730295646763343
q=0,9	**x**	0.998531622666922	0.004217164396211
	y	-0.25235514785377	0.854654293359799

Table 3.3: x:inspection probability, y:violation probability

Respectively, let's take a look at the last two rows of the Table 3.3 where $q = 0.9$: $r_{x,k} = 0.998531622666922$, i.e. as k increases, the probability of inspection increases in an almost linear way with k. Simultaneously, the probability of violation decreases ($r_{y,k}$ is negative), but not decisively ($r_{y,k}$ is close to 0). On the other hand, the probability of violation increases as l increases. This result describes the fact that for a price of q very close to 1, the main thing that affects the decision of the violator is the number of the remaining violations.

These results reinforce our intuition concerning the sensitivity of the probability of inspection (and, therefore, the value of the game) to k as q increases.

3.3.3 Outline

To sum up, we conjecture that:

1. For any game $\Gamma(n, k, l)$ and any q, there exists a number k^*, which depends on n, q, such that $\forall k \geq k^*$:
   - x approximately equals to q.
   - y is very close to 0.
   - the value of the game is approximately equal to $l \cdot q$.
2. k^* is an increasing function of n, q.
3. *It seems* that $x \leq q$, i.e. Player I will never inspect with a probability higher than the amount q. We have not found an intuitive explanation yet, but this inequality holds in all our experiments.

3.4 Other results

1. Let $n \leq 200,\ 0 \leq k, l \leq n-1,\ q \in \{0.1, 0.15, 0.2, ..., 0.9\}$. There is no game $\Gamma(n, k, l)$ in which the violator has the incentive not to act at any day and pay $l \cdot q$.

   Player II can guarantee a cost $l \cdot q$ if he chooses not to act at any stage. Hence, $\tilde{V}_q(n, k, l) \leq l \cdot q$. Moreover, our simulations show that this inequality is strict, i.e.
   $$\tilde{V}_q(n, k, l) < l \cdot q$$
   Thus, the violator has incentive to perform illegal actions.

2. The inequality of Lemma 3.1 does not hold for the game $\Gamma(n, k, l)$.
   For example, for $q = 0.7$:
   $\tilde{V}_{0.7}(48, 15, 10) = 3.08349$
   $\tilde{V}_{0.7}(48, 14, 10) = 2.87980$
   $$\tilde{V}_{0.7}(48, 15, 10) - \tilde{V}_{0.7}(48, 14, 10) = 0.20369 > 0.0208 \approx \frac{1}{48}$$

CHAPTER 4

APPENDIX

4.1 Code

```
import numpy as np
from decimal import *
from numpy import dtype
import pandas as pd
import matplotlib.pyplot as plt
from numpy.random import normal,rand


max_n =

min_k =
max_k =

min_l =
max_l =

min_q = Decimal('')
max_q = Decimal('')
step_q = Decimal('0.05')


#precision of decimal digits. As n gets greater,
#precision must become greater too
getcontext().prec = 75

number_of_all_results = 0
number_of_non_boundary_results = 0

#arrays in which we store results to avoid extra computations
array=np.empty((max_n,int(max_q*100),max_k+1,max_l+1),
       dtype=Decimal)
```

```
last_x = 'N/A'
last_y = 'N/A'

#definition of n!
def factorial(n):
    if n==0 or n==1:
        return Decimal(1)
    return Decimal(n) * factorial(Decimal((n)-Decimal(1)))

#definition of compination function
def combination(n,m):
    if m == 0:
        return Decimal(1)
    elif m > n:
        return Decimal(0)
    return factorial(Decimal(n))/(factorial(Decimal(n-m))*
        factorial(Decimal(m)))

#definition of function u(n,k) which is used to define value
#of G(n,k,n)
def u(n,k,q):
    s = Decimal(0)
    for j in range(0,k):
        s += (Decimal(k)-Decimal(j))*Decimal(combination(n-k-1+j,j))
            *Decimal(q)**Decimal(j)
    return s

#definition of value for G(n,k,n)
def w(n,k,q):
    m = Decimal(k)-(Decimal(1)-Decimal(q))**(Decimal(n)-Decimal(k)+
        Decimal(1))*Decimal(u(n,k,q))
    return m



#Recursive function G(n,k,l).
#The function returns the value of G(n,k,l) and stores it in array.
def G(n,k,l,q):
    q_int = int(q*100)
    global last_x
    global last_y

    #if an inappropriate instace is entered, value becomes -1
    if not (0<=k and k<=n):
        v = Decimal(-1)
    if not (0<=l and l<=n):
        v = Decimal(-1)
    if not (type(n) == int):
        v = Decimal(-1)
    if not (type(k) == int):
        v = Decimal(-1)
    if not (type(l) == int):
        v = Decimal(-1)
```

```
    #checking for boundary conditions
    if k == 0:
        v = Decimal(0)
    elif l == 0:
        v = Decimal(0)
    elif n == k:
        v = Decimal(1)*Decimal(q)
        last_x = 'N/A'
        last_y = 'N/A'
    elif n == l:
        v = Decimal(w(n,k,q))
        x = Decimal(q)
        y = Decimal(w(n-1,k,q)) - Decimal(w(n-1,k-1,q))

        last_x = Decimal(x)
        last_y = Decimal(y)

    #non boundary conditions: 0 < k,l < n
    else:
        if array[(n-2,q_int-1,k-1,l-1)] != None:
            a = array[(n-2,q_int-1,k-1,l-1)] + Decimal(1)
        else:
            a = G(n-1,k-1,l-1,q) + Decimal(1)
            array[(n-2,q_int-1,k-1,l-1)] = Decimal(a)-Decimal(1)
        if array[(n-2,q_int-1,k-1,l)] != None:
            b = array[(n-2,q_int-1,k-1,l)]
        else:
            b = G(n-1,k-1,l,q)
            array[(n-2,q_int-1,k-1,l)] = Decimal(b)
        if array[(n-2,q_int-1,k,l-1)] != None:
            c = array[(n-2,q_int-1,k,l-1)]
        else:
            c = G(n-1,k,l-1,q)
            array[(n-2,q_int-1,k,l-1)] = Decimal(c)

        if array[(n-2,q_int-1,k,l)] != None:
            d = array[(n-2,q_int-1,k,l)]

        else:
            d = G(n-1,k,l,q)
            array[(n-2,q_int-1,k,l)] = Decimal(d)

        #find value and strategy for the 2x2 matrix game
        x = (d-c)/(a-b-c+d)
        y = (d-b)/(a-b-c+d)

        v = x*(y*a+(1-y)*b)+(1-x)*(y*c+(1-y)*d)
        last_x = x
        last_y = y

    array[(n-1,q_int-1,k,l)] = v


    return v
```

```
#list for all results (including boundary)
value = []

#lists for non boundary results
days = []
inspections = []
violations = []
value_2 = []
probability_of_inspection = []
probability_of_violation = []
amount_q = []


#Part in which we run G(n,k,l) for all selected instances
#and print the results. It also stores the results in the
#produced lists to use them for plots and correlation matrix.
for q in range(int(min_q*100), int(max_q*100) + int(step_q*100),
               int(step_q*100)):
    for k in range(min_k, max_k+1):
        for l in range(min_l, max_l+1):
            for n in range(max(k,l), max_n+1):

                _q = q/100
                g = G(n,k,l,_q)
                print('G(',n,',',k,',',l,'), q=',_q)
                print("value=","{:.5f}".format(g), '\t',
                      "x=", last_x, '\t', "y=", last_y)
                print("-----------------------------------")
                number_of_all_results += 1
                value.append(g)

                if last_x != 'N/A' and last_y != 'N/A':
                    number_of_non_boundary_results += 1
                    days.append(n)
                    inspections.append(k)
                    violations.append(l)
                    value_2.append(g)
                    probability_of_inspection.append(last_x)
                    probability_of_violation.append(last_y)
                    amount_q.append(_q)


print("---------------")


#correlation matrix for non boundary conditions
#(except G(n,k,n))
data = {'n': days,
        'k': inspections,
        'l': violations,
        'q': amount_q,
        'v': value_2,
```

```
        'x': probability_of_inspection,
        'y': probability_of_violation
        }
df = pd.DataFrame(data,columns=['n','k','l','q','v','x','y'])

corrMatrix = df.astype(float).corr()
print("Matrix with correlation coefficients:")
print (corrMatrix)


print("---------------")

#plots creation
all_results = []
non_boundary_results = []

for i in range (0,number_of_all_results):
        all_results.append(i)
for i in range(0,number_of_non_boundary_results):
    non_boundary_results.append(i)



#plot for all values (including boundary conditions)
plt.xlabel('RESULTS')
plt.ylabel('VALUES')

plt.plot(all_results, value, label = "values")
plt.legend()
plt.show()


#plot for prices of x,y (excepting boundary conditions)
plt.xlabel('RESULTS')
plt.ylabel('optimal STRATEGIES')

plt.plot(non_boundary_results, probability_of_inspection,
         label= "x")
plt.plot(non_boundary_results, probability_of_violation,
         label= "y")
plt.legend()
plt.show()
```

4.2 Tables for games $\Gamma(3, k, l),\ k, l \in \{1, 2\},\ q = 0.4$

values and optimal strategies with k loop prior to l loop						
results	**n**	**k**	**l**	$\tilde{V}_{0.4}(3, k, l)$	**x**	**y**
1st	3	1	1	0.22222	0.2222222222	0.2222222222
2nd	3	1	2	0.47257	0.2616033755	0.4725738397
3rd	3	2	1	0.35897	0.3589743590	0.1025641026
4th	3	2	2	0.73880	0.3825136612	0.1530054645

Table 4.1: $\Gamma(3, k, l), q = 0.4$ with k-loop prior to l-loop

values and optimal strategies with l loop prior to k loop						
results	**n**	**k**	**l**	$\tilde{V}_{0.4}(3, k, l)$	**x**	**y**
1st	3	1	1	0.22222	0.2222222222	0.2222222222
2nd	3	2	1	0.35897	0.3589743590	0.1025641026
3rd	3	1	2	0.47257	0.2616033755	0.4725738397
4th	3	2	2	0.73880	0.3825136612	0.1530054645

Table 4.2: $\Gamma(3, k, l), q = 0.4$ with l-loop prior to k-loop

BIBLIOGRAPHY

[1] Avenhaus R., von Stengel B., Zamir S., *Inspection Games*, Handbook of Game Theory, Vol 3, (1995).

[2] Avenhaus R., Canty M.J., Kilgour D.M., von Stengel B., Zamir S., *Inspection Games in Arms Control*, European Journal of Operational Research, 31, 383-394, (1996).

[3] Avenhaus R., Canty M.J., *Playing for time: A sequential inspection game*, European Journal of Operational Research 167, 475–492, (2005).

[4] Baston V.J., Bostock F.A., *A Generalized Inspection Game*, Naval Research Logistics, 38, 171-82 (1991)

[5] Dresher M., *A Sampling Inspection Problem in Arms Control Agreements: A Game-Theoretic Analysis*, Memorandum RM-2972-ARPA, The RAND Corporation, Santa Monica, California, (1962).

[6] Ferguson Th., Melolidakis C., *On the inspection game*, Naval Research Logistics, 45(3):327–334, (1998).

[7] Ferguson Th., Melolidakis C., *Games with finite resources*, International Journal of Game Theory 29(2):289–303, (2000).

[8] Gale D., *Information in Games with Finite Resources*, Annals of Mathematical Studies, 39,141–145, (1957).

Garnaev Garnaev A. Yu, *A Remark on the Customs and Smuggler Game*, Naval Research Logistics, 41, 287-293, (1994).

[9] Hohzaki R., *An inspection game with multiple inspectees*, European Journal of Operational Research, 178, 894-906, (2007)

[10] Hohzaki R., *An Inspection Game with Smuggler's Decision on the amount of Contraband*, Journal of the Operations Research Society of Japan, Vol. 54, No. 1, 25–45, (2011).

[11] Maschler M., *A Price Leadership Method for Solving the Inspector's Non-Constant Sum Game*, Naval Research Logistics Quarterly, 13, 11–33, (1966).

[12] Ross S.M., *Goofspiel-the Game of Pure Strategy*, Journal of Applied Probability, 8, 621-625, (1971).

[13] Sakaguchi M., *A Sequential Allocation Game for Targets with Varying Values*, Journal of the Operations Research Society of Japan, 20, 182–193, (1977).

[14] Sakaguchi M., *A Sequential Game of Multi-Opportunity Infiltration*, Mathematica Japonica, 39, 157–166, (1994).

[15] Thomas M.U., Nisgav Y., *An Infiltration Game with Time Dependent Payoff*, Naval Research Logistics Quarterly, 23, 297-302, (1976).

[16] von Stengel B., *Recursive Inspection Games*, Mathematics of Operations Research 41, 935-952, (2016)

[17] von Stengel B., Zamir S., *Leadership games with convex strategy sets*, Games and Economic Behavior 69, 446–457, (2010).