

An Aggregation Method for Large-Scale Dynamic Games

Carlos Daniel Santos[†]

Nova School of Business and Economics

January 17, 2020

Abstract

It is a well known fact that many dynamic games are subject to the curse of dimensionality, limiting the ability to use them in the study of real-world problems. I propose a new method to solve complex large-scale dynamic games using aggregation as an approximate solution. I obtain two fundamental characterization results. First, approximations with small within-state variation in the primitives have a smaller maximum error bound. I provide numerical results which compare the exact errors and the bound. Second, I find that for monotone games, order preserving aggregation is a necessary condition of any **optimal aggregation**. I suggest using quantiles as a straightforward implementation of an order preserving aggregation architecture for industry distributions. I conclude with an illustration, by solving and estimating a stylized dynamic reputation game for the hotel industry. Simulation results show maximal errors between the exact and approximated solutions below 6%, with average errors below 1%.

Keywords: Aggregation, Curse of Dimensionality, Dynamic Games, Reputation, Markov Perfect Equilibrium.

1 Introduction

The largest challenge to studying discrete state dynamic games is the curse of dimensionality. The curse of dimensionality occurs because the number of states grows exponentially with the number of players. If there are N players, and each can be in one of K possible states, the size of the problem is K^N .

However, studying dynamic interactions is relevant for marketing scholars given the importance of persistence that naturally generates models of dynamic demand (see Nair, 2019 for a

*Nova SBE, Rua da Holanda, 1, 2775-405 Carcavelos, Portugal; carlos.santos@novasbe.pt

[†]I would like to thank Jaap Abbring, Victor Aguirregabiria, Bart Bronnenberg, Ulrich Doraszelski, Pedro Gardete, Tobias Klein, Philipp Schmidt Dengler, John Van Reenen, Gabriel Weintraub and seminar participants at Alicante, Tilburg, EC2 Rome, RNIC Structural IO workshop, ES-NAWM San Diego, JEI Murcia and EARIE Rome for useful comments on a earlier version of this article. Financial support from the Spanish Ministerio de Ciencia e Innovación FEDER funds under project SEJ-2007-62656 and from the Portuguese Fundacao para a Ciencia e Tecnologia grant CONTDOUT/114/UECE/436/10692/1/2008 and the Strategic Project UID/ECO/00124/2013, and by POR Lisboa under the project LISBOA-01-0145-FEDER-007722 are gratefully acknowledged.

review). For example, Che, Sudhir and Seetharaman (2007) find that "omission of state dependence in demand biases inference of firm behavior" while Dubé, Hitsch and Rossi (2010) "show that the finding of inertia is robust to flexible controls for preference heterogeneity and not due to autocorrelated taste shocks". Several authors address this dynamic element and study settings with dynamic competition. Shen (2014) constructs a dynamic model with endogenously expanding demand to investigate the optimal entry and exit behavior of firms. Dubé, Hitsch and Manchanda (2005) use a dynamic advertising competition model to show that pulsing (on and off) is an optimal (equilibrium) advertising strategy. Goettler and Gordon (2011, 2014) use a dynamic oligopoly framework to explain how competition can reduce innovation in an environment with persistent demand. This area is now growing but researchers face some limitations. For example, Chintagunta, Qin and Vitorino (2018) acknowledge this limitation because the "dynamic problem specified (...) is computationally infeasible due to the very large size of the state space, and the need to solve for equilibrium prices in an oligopoly setting" forcing them to "simplify the firm's optimization problem by focusing on the intertemporal pricing considerations". The same problem of large state spaces is acknowledged by Nair (2019): "A separate area where more progress will be welcome is in developing methods to handle very large state spaces in the context of dynamic games."

To alleviate the curse of dimensionality problem, I propose an approximation method using state aggregation. Let N be the number of competitors and K the number of possible states for each competitor (e.g. state 1, 2, ..., K). The proposed method allows researchers to study more complex environments with more competitors and larger state spaces. The base idea is intuitive, in a game that satisfies symmetry and anonymity, only the state of the competitors matters and not their identity (Doraszelski and Satterthwaite, 2010). That is, a situation with competitor A in state 1 and competitor B in state 2 is **identical** to a situation with competitor A in state 2 and competitor B in state 1 (**identical state**). This means that the industry state (market structure) is fully summarized by the *distribution of players* across the states. The benefit is that instead of growing exponentially, the number of possible states is now growing at a much smaller rate.¹ For example, a problem with $K = 5$ possible states and 100 players has 7.9×10^{69} industry states in the base case that can be reduced to 2.2×10^7 possible distribution configurations.² Taking this idea one step further we can now aggregate states that are close in a distribution sense, that is, quantiles. For example, the industry distribution with $K = 5$ has 350 possible states (approximated by 5 quantiles) and 3,575 possible states (approximated by 10 quantiles). The difference from aggregation of "single agent" problems to games is that besides the approximation error to the single agent dynamic problem, there is a second approximation error to the rival's equilibrium strategies. That is because we can only solve for approximated equilibrium strategies.

To illustrate the idea, consider the example in Table [1] with three firms competing against firm i and one variable that can take only two values, High (H) or Low (L). The original problem has 8 (2^3) possible industry configurations for the rival's state. The symmetric and

¹The size of the state space is reduced from K^{N-1} to $\binom{K+N-2}{K-1} = \frac{(K+N-2)!}{[(N-1)!(K-1)!]}$

²Table [2] in Section [3] reports how the size of the state space varies with the numbers of firms.

anonymous case has only four configurations $\binom{4}{1}$. Industry states 2, 3 and 5 get aggregated into state B whereas industry states 4, 6 and 7 get aggregated into state C. The solution obtained is exactly the same because the problem in state (L, L, H) is the same as in state (L, H, L) or state (H, L, L) . That is, the three states are **identical**. If we call this "macro-state" $(2, 1)$ denoting 2 competitors in L state and 1 competitor in H state we can say that this macro-state has *zero within-state variation*, where the variation is defined with respect to the equilibrium value function. That is, the equilibrium value at each of the states is the same and there is no information lost from aggregating them.³ The four resulting aggregate states can be further aggregated into only two macro-states E and F where E contains the previously defined states A and B whereas F contains the previously defined states C and D (see Table [1]). This now extends to aggregation of *non-identical* and there is some information lost, since the equilibrium value at the aggregated states is no longer the same. On the other hand, the benefit is that the number of states at which we need to solve the model is smaller. This is the trade-off of aggregation, obtaining the largest state space that is computationally feasible with the smallest loss of information possible.

Table 1: Aggregation Example.

| Industry State | Firm | | | Symmetric Anonymous Industry State | Aggregated Industry State |
|----------------|------|---|---|------------------------------------|---------------------------|
| | 1 | 2 | 3 | | |
| 1 | L | L | L | A | E |
| 2 | L | L | H | B | E |
| 3 | L | H | L | B | E |
| 4 | L | H | H | C | F |
| 5 | H | L | L | B | E |
| 6 | H | L | H | C | F |
| 7 | H | H | L | C | F |
| 8 | H | H | H | D | F |

I obtain two main results on the quality of approximate aggregation methods. First, Theorem [5.1] characterizes the sup-norm error for a general class of aggregation methods. It illustrates the three sources of error. In particular, it shows that approximate solutions with a smaller within-state variation in both primitives (i) the period returns and (ii) transition matrix, are characterized by a smaller sup-norm error. The third source of error is specific to games and is not present in single agent problems. It is due to the use of approximate, instead of the exact solution, to the rivals' strategies. Second, Theorem [6.2] provides a necessary condition of an **optimal aggregation** scheme. When the value function satisfies a state ordering property (monotonicity), an optimal aggregation must respect an order preserving condition.⁴ This is a necessary condition. There can be multiple aggregation rules that satisfy this condition. It is a rather intuitive result given that previous research already emphasizes that "similar" states

³If the problem is not truly symmetric and anonymous, the simplified aggregation comes at the cost of some loss of information. There is information loss from aggregation because the within-state variation is strictly positive (different values at different states). The information loss grows with the growth of within-state variation. See Geiger and Temmel (2014) for a discussion on lumping of Markov chains and the information loss (entropy).

⁴Section [6] contains the exact definitions of monotonicity and order preserving aggregation.

should be aggregated together (Bertsekas and Castanon, 1989). Given that, in our case, an industry state (market structure) is a distribution, two states are "similar" if the two industry distributions are close in some appropriate metric (the exact metric is unimportant for our results). For example, in Table [1] we decided to aggregate A and B into state E , instead of aggregating A and C . This is intuitive. State A has 3 firms in low state. State C has two firms in high state and one firm in low state while state B has two firms in low state and one firm in high state. Thus, state A is "closer" to state B than to state C and this is why we propose to aggregate them. This illustrates the result that optimal aggregation must be order preserving. Of course we could also aggregate A , B and C (or B and C) thus preserving the order but we should not aggregate A and C without B .

Aggregating states with similar market structures reduces within aggregate state variation. It reduces the variability of micro-states within a given macro-state, that characterized the error obtained in Theorem [5.1]. This result is important as it let us explain which approximation schemes are not optimal. From a practical perspective, if the researcher cannot solve the game at all possibly industry states, she can try and solve the game at the aggregate points defined by industry distribution quantiles. The result tells us that whatever is the optimal aggregation rule, it belongs to the same class as the selected quantiles (necessary condition).

I conclude with an illustration by solving, and estimating, a stylized dynamic reputation game for the hotel industry. The game is subject to the curse of dimensionality, that makes it a computational challenge to solve when there is more than about a dozen players. I use a least squares nested fixed point approach for estimation. Although nested fixed point is econometrically more efficient than two-step methods (e.g. Bajari, Benkard and Levin, 2007), and not subject to the potential identification problems created by the choice (perturbation) of alternative actions (Srisuma, 2013), they are often neglected in dynamic games due to the computational challenge of solving the game multiple times. I illustrate how the nested fixed point can be applied using the method developed in this article. Using it we are able to solve the game in about 460 seconds.

2 Related Literature

The method of aggregation is frequent in numerical solutions for large scale dynamic programming problems (e.g. Bertsekas, 2005; Van Roy, 2006). The aggregation method developed here is sufficiently general to be applied to any type of Markov games. Hard aggregation consists of assigning micro states to macro (or aggregate) states where the assignment is many to one, that is, many micro states are aggregated into one macro state.⁵ For a given number of aggregate nodes/states, the main decision is about which micro-states to aggregate together. The main advantage of aggregation is the reduction in the number states. In discrete state Markov games, obtaining an equilibrium solution is constrained by the 'curse of dimensionality' caused by the exponential growth of the state space as either (i) the number of players (N) or (ii) the number

⁵In hard aggregation each micro state is assigned to only one aggregate/macro state while in soft aggregation, micro states can be assigned to more than one aggregate/macro state.

of state variables (K) increase. For example, in our application the set of industry states with 77 players and a state variable that takes 5 levels, has cardinality equal to $6.6 * 10^{53}$. Using aggregation, the industry state is reduced from $6.6 * 10^{53}$ to 630.⁶

The problem with aggregation is that it works as an approximation. As such, the solution to the aggregated problem differs from the solution to the original problem. We thus need to find an aggregation solution that is optimal in some sense. For example, an aggregation is optimal if it delivers the smallest approximation error for a given size of the industry state (a number of macro states). The selection of an aggregation rule (matrix) is what determines the quality of the approximation. This is not a trivial decision because the literature on general **optimal aggregation** rules is scarce, even for simple Markov chains. For example, Deng et al. (2011) discuss optimality for simple discrete time Markov chains and show that numerically searching for the optimal aggregation is a problem itself subject to the curse of dimensionality (see their Equation [8]).

I borrow ideas from two articles: order preserving aggregation and state similarity. First, Jia (2011) shows that consecutive (order preserving) state aggregation is optimal for the case where the value function is *a priori* known. Second, Deng et al. (2011) show that for irreducible and aperiodic Markov chains, partitioning the chain according to the sign structure of the eigenvector for the second largest eigenvalue allows maximal information extraction for a given partition size. This partitioning rule aggregates highly communicating states (states with high within macro-state transition). However, this result can only be shown for the case with two macro-states (bi-partition). In Section [6] I extend these results by restricting to symmetric and anonymous monotone Markov games. For this class of games I show that a necessary condition of an optimal aggregation mechanism is order preservation. I show that one straightforward way to implement order preserving aggregation for symmetric and anonymous monotone Markov games which is the use of quantiles. However, the order preserving aggregation can take many other forms.

Several methods have been proposed to address the curse of dimensionality in discrete state Markovian games. For example, Pakes and McGuire (2001) suggest the use of a stochastic algorithm that wanders through the state space and updates states in the recurrent class, avoiding the computation of the integral for **all** the states in the state space. The computational gains are larger when the recurrent class of the state space is significantly smaller than the full state space. This algorithm is mostly fit for cases where the ergodic set has low cardinality. Weintraub et al. (2008) propose an alternative equilibrium concept, the 'oblivious equilibrium'. In the oblivious equilibrium, players' strategies depend solely on own state and long run state distribution, that is, all rivals' states (\mathbf{s}_{-i}) are aggregated into one single macro-node (the LR industry distribution). This framework is a good approximation for industries with a large number of firms and no aggregate shocks, provided the industry distribution satisfies a light tail condition (no market leaders).⁷

Ifrach and Weintraub (2016) extend this and propose a moment-based Markov equilibrium

⁶Note that the assumptions of symmetry and anonymity already reduce the size of the state (industry distribution) to 7.9 million.

⁷The oblivious equilibrium concept proposed by Weintraub et al. (2008) requires a "light tail" condition for the equilibrium state distribution to have a well defined steady state distribution.

(MME) with dominant firms, where each firm keeps track of the states of the dominant firms and a few moments for the fringe firms. It departs from the Oblivious Equilibrium proposed by Weintraub, Benkard and Van Roy (2008) by including the dominant firms's state and the moments for the fringe firms. Both the Oblivious and the MME equilibrium concepts can be analyzed as aggregation architectures. Take the MME approach, which is more general. The aggregation rule is as follows: two micro-states with the same individual states for the dominant firms and the same moment for the fringe firms, are aggregated in the same macro-node. For example, imagine a case with 9 rival firms where each firm is dominant if they reach state 5 and let F be the cumulative distribution over the non-dominant states (1 to 4). Under the moment-based approximation, facing a distribution over states 1 to 4 of $F = (.3, .6, .9, .9)$ is similar to facing a distribution of $F = (.2, .7, .9, .9)$, since the average of the 9 non dominant firms is 2 in both cases. Moment-based aggregation would thus aggregate these two states together. While an MME can be written using state aggregation, moments fall outside of the optimal class of aggregations because the MME does not satisfy the monotonicity condition that is the necessary condition for aggregation to be optimal. As we show in our application to the hotel industry, the equilibrium distribution is multimodal making it impossible to use either the oblivious or the MME equilibrium concepts.

Farias, Saure and Weintraub (2012) propose a fully separable approximation architecture and also illustrate that MME is underperforming when compared to the new approximate solution. They extend to dynamic games the work of De Farias and Van Roy (2003) on single agent problems which consists of using a linear programming approach to solve an approximated value function. The value function is approximated by basis functions, transforming the game into a much more tractable linear programming problem. They have two main contributions.

First, a mathematical programming approach to computing dynamic games that involves solving a linear program subject to non-linear constraints. This is different from the traditional approaches to solving Markov decision problems such as the one used here. Perhaps the main disadvantage of the linear programming method for solving MDPs is that it requires setting up the set of non linear constraints, one for each industry state. This is still subject to the curse of dimensionality (see their equation [4]), and is overcome by using constraint sampling. Second, the use of basis functions as an approximation to solve large scale problems. Basis functions, as well as aggregation, are part of the more general class of approximations that use lower dimensional basis to approximate higher dimensional objects. Their matrix Φ is a mapping from the number of states to the number of nodes/basis functions. In particular, their most accurate approximation method is the fully separable approximation architecture. It resembles aggregation as it involves a Φ matrix of zeros and ones. The difference is that each "macro-state" maps to more than one micro-state. Another important difference between hard aggregation and basis functions is that no disaggregation matrix is explicitly specified. This article also departs from theirs by exploiting information on the primitives to characterize optimality. In fact, the optimal choice of aggregate nodes can be combined with the linear programming method proposed by Farias et al. (2012). Furthermore, we provide theoretical foundations for why moments-based approximations underperform when compared to basis function approximations.

Something to which we return below.

3 Model

This section describes the elements of the model. In particular, I describe the period game, the transition function, the payoffs, the strategies, and the equilibrium concept. Unless otherwise stated, all proofs contained in the article are original. In what follows, boldface denotes vectors (e.g. \mathbf{s} , \mathbf{V}) while sets are denoted with an overtilde (e.g. $\tilde{S}, \tilde{A}, \tilde{F}, \dots$). To illustrate how the aggregation method can be implemented, I use a stylized model of reputation building in the hotel industry.

I consider a dynamic game with discrete time $t = 1, 2, \dots, \infty$.⁸ The number of players is N and is assumed fixed over time and a typical player is denoted by $i \in \{1, \dots, N\}$. Players can choose actions $a_{it} \in \tilde{A}$, for example, agents can decide to invest into reputation building. I will focus on stationary Markov games, so we don't need to keep track of time and I will frequently use the following short notation, $\mathbf{s}' = \mathbf{s}_{t+1}$.

A variety of problems have been studied with a similar structure. For example, capacity (investment) games (Besanko et al., 2010), innovation adoption (Schmidt-Dengler, 2006), international trade (Santos, 2017).

3.1 States, actions and state transition

States Each player is endowed with an **own state** variable, $s_{it} \in \tilde{S} = \{1, 2, \dots, K\}$ - e.g. the current rating of the hotel. The own state variable can be univariate or multivariate. Multivariate variables can be indexed along a single dimension. The **industry state** can be written as a vector, $\mathbf{s}_t = (s_{1t}, \dots, s_{Nt}) \in \tilde{S}^N$.

Actions Actions are chosen from the set $a_{it} \in \tilde{A}_i$, where \tilde{A}_i is a compact and convex subset of the Euclidean space and $\mathbf{a}_t = (a_{1t}, \dots, a_{Nt}) \in \tilde{A} = \times_{i=1}^N \tilde{A}_i$ - e.g. hotels choose a level of (investment/effort) into customer satisfaction. Compactness and convexity can be relaxed given the finite nature of the state space. For simplicity, we will further assume that the set of admissible actions is state invariant: \tilde{A}_i is the same for all s . All players choose actions simultaneously after observing the state s_t .

State transition The industry state transition is described by a probability function $q : \tilde{S}^N \times \tilde{S}^N \times \tilde{A}^N \rightarrow [0, 1]$ where a typical element $q(\mathbf{s}_{t+1}, \mathbf{s}_t, \mathbf{a}_t)$ equals the probability that industry state \mathbf{s}_{t+1} is reached from state \mathbf{s}_t when players choose actions \mathbf{a}_t . It is required that $\sum_{\mathbf{s}_{t+1} \in \tilde{S}^N} q(\mathbf{s}_{t+1}, \mathbf{s}_t, \mathbf{a}_t) = 1$.

⁸Kim, Bradlow and Iyengar (2019) show that when choosing the period of analysis (e.g. year, quarter, month) to analyze Marketing problems, incorrect time aggregation creates either bias or variance for estimation. We are assuming the frequency of the True Generation Process is known.

Information structure Assume the industry state is publicly observed - all players observe \mathbf{s} . To guarantee existence of equilibrium (Doraszelski and Satterthwaite, 2010) let there be a privately observed i.i.d. shock, $g \in \mathbb{R}$, with distribution $G(g)$ and finite absolute first moment $E(|g|) < \infty$.⁹

3.2 Pay-offs

Period pay-off Player i 's pay-offs depend on the demand, cost and pricing structure. In reduced form, this is a function of states and actions of all players and a privately observed state - $\pi(\mathbf{a}, \mathbf{s}, g)$ with $\pi : \tilde{A}^N \times \tilde{S}^N \times \mathbb{R} \rightarrow \mathbb{R}$. For simplicity, let us restrict rivals' actions to have no direct effects on period payoffs: $\pi(\mathbf{a}, \mathbf{s}, g) = \pi(a, \mathbf{s}, g)$. This simplification can be relaxed but is useful since it allows us to separate the solution to optimal period profits (static) and focus our attention in the solution to the dynamic decision problem (dynamic/investment). In terms of our main results, the simplification has no effect to the optimal aggregation architecture (Theorem [6.2]). There is a minor effect to the bounds derived in Theorem [5.1] that must now include an extra term to account for the effect of the departure from rival's equilibrium strategies to the static returns.

Game pay-off Players discount the future at rate $\rho \in [0, 1)$ and the game pay-off of player i is equal to the present discounted value of all future period payoffs. If firm i observes the industry state (\mathbf{s}) and private shock (g_i), its **ex-post** value function can be written as

$$V_i(\mathbf{s}, g_i) = \max_{a_i \in \tilde{A}} \pi(a_i, g_i, \mathbf{s}) + \rho \sum_{\mathbf{s}' \in \tilde{S}^N} V_i^e(\mathbf{s}') q(\mathbf{s}', \mathbf{s}, a_i, \mathbf{a}_{-i}), \quad \forall i, \mathbf{s}$$

where $q(\mathbf{s}', \mathbf{s}, a_i, \mathbf{a}_{-i})$ is the evolution of the industry state conditional on player i choosing action a_i , and the other players choosing actions $\mathbf{a}_{-i} = (a_1, \dots, a_{i-1}, a_{i+1}, \dots, a_N)$. $V_i^e(\mathbf{s}') = \int V_i(\mathbf{s}', g_i') dG(g')$ is the **ex-ante** value function

$$V_i^e(\mathbf{s}) = \int \left(\max_{a_i \in \tilde{A}} \pi(a_i, g_i, \mathbf{s}) + \rho \sum_{\mathbf{s}' \in \tilde{S}^N} V_i^e(\mathbf{s}') q(\mathbf{s}', \mathbf{s}, a_i, \mathbf{a}_{-i}) \right) dG(g), \quad \forall i, \mathbf{s}.$$

Conditional on the actions, privately observed i.i.d. shocks have no information content for the future of the game, and the problem is similar to the case without privately observed i.i.d. shocks. In this case, the period returns are $\pi(a_i, \mathbf{s})$ and the value function is simply

$$V_i(\mathbf{s}) = \max_{a_i \in \tilde{A}} \pi(a_i, \mathbf{s}) + \rho \sum_{\mathbf{s}' \in \tilde{S}^N} V_i(\mathbf{s}') q(\mathbf{s}', \mathbf{s}, a_i, \mathbf{a}_{-i}), \quad \forall i, \mathbf{s}.$$

Although we use the case with privately observed shocks in the empirical application, we will now assume no private information for simplicity.

⁹While we assume latter that persistent states are observed by the researcher (econometrician), Gallant et al. (2017) study an environment where this is relaxed and persistent states can be unobservable to the researcher. Note that, like here, in their case all states are still observed by all players.

3.3 Strategies and equilibrium

Strategies Players use pure Markovian strategies, mapping the set of pay-off relevant states onto the action set $\alpha_i(\mathbf{s}) : \tilde{S}^N \rightarrow \tilde{A}_i$.

Equilibrium The equilibrium concept is Markov Perfect Equilibrium in the sense of Maskin and Tirole (1988, 2001). As the focus of this article is on approximation methods, I will abstract from problems related with existence that have been studied in the literature, and assume that the conditions for equilibrium existence are verified. For example, Doraszelski and Satterthwaite (2010) or Schmidt-Dengler and Pesendorfer (2008) provide existence proofs for a similar class of models.

Definition 1 (Equilibrium) A collection of strategies $(\alpha_i^*(\mathbf{s}))$ form an equilibrium if for any i and \mathbf{s}

$$\alpha_i^*(\mathbf{s}) = \arg \max_{a_i \in \tilde{A}} \pi(a_i, \mathbf{s}) + \rho EV_i(\mathbf{s}'), \quad \forall i, \mathbf{s}, \quad (1)$$

where $EV_i(\mathbf{s}') = \sum_{\mathbf{s}' \in \tilde{S}^N} V_i(\mathbf{s}') q(\mathbf{s}', \mathbf{s}, a_i, \boldsymbol{\alpha}_{-i})$ and the value function $V_i(\mathbf{s}')$ satisfies

$$V_i(\mathbf{s}) = \max_{a_i \in \tilde{A}} \pi(a_i, \mathbf{s}) + \rho \sum_{\mathbf{s}' \in \tilde{S}^N} V_i(\mathbf{s}') q(\mathbf{s}', \mathbf{s}, a_i, \boldsymbol{\alpha}_{-i}), \quad \forall i, \mathbf{s}, \quad (2)$$

where $\boldsymbol{\alpha}_{-i} = (\alpha_1(\mathbf{s}), \dots, \alpha_{i-1}(\mathbf{s}), \alpha_{i+1}(\mathbf{s}), \dots, \alpha_N(\mathbf{s}))$ is the vector of rivals' strategies.

An equilibrium is the solution to the non-linear system of equations formed by Equations 1 and 2 (one for each i, \mathbf{s}). In general, we might have several solutions/multiple equilibria. Optimal strategies $(\boldsymbol{\alpha}^*)$ generate an equilibrium industry state transition

$$q(\mathbf{s}', \mathbf{s}, \boldsymbol{\alpha}^*) = q[(s'_1, s'_2, \dots, s'_N), (s_1, s_2, \dots, s_N), (\alpha_1^*, \alpha_2^*, \dots, \alpha_N^*)] \quad (3)$$

The equilibrium solution is thus a vector of value functions $(V_i^*(\mathbf{s}))$ and optimal strategies $(\alpha_i^*(\mathbf{s}))$ for each firm i at each industry state \mathbf{s} . In the case of multiple equilibria there is more than one solution to the system of equations of optimal strategies/value functions, one for each equilibrium.

3.4 Dimensionality concerns

The problem is computationally demanding as it requires the numerical solution to a dynamic programming problem with cardinality K^N .¹⁰ One possible solution to the curse of dimensionality is to impose restrictions on the primitives, $\pi(a_{it}, \mathbf{s})$ and $q(\mathbf{s}', \mathbf{s}, \boldsymbol{\alpha}^*)$. Anonymity and symmetry allow the problem to be formulated in a smaller set of industry states (Ericson and Pakes, 1995; Doraszelski and Satterthwaite, 2010). Anonymous sequential games have a long tradition in the literature (e.g. Jovanovic and Rosenthal, 1988). Anonymity implies that firm i does not care

¹⁰There are two sources of computational complexity for dynamic problems, the size of the state space and integration (computing the expectation). Using hard aggregation addresses the first problem while the second can be addressed with randomization (Rust, 1996, 1997).

about the identity of its competitors, and its decision depends solely on the industry structure (distribution) it faces, or

$$\begin{aligned} V_i(s_1, \dots, s_i, \dots, s_j = x, \dots, s_l = y, \dots, s_N) &= & \text{(Anonymity)} \\ V_i(s_1, \dots, s_i, \dots, s_j = y, \dots, s_l = x, \dots, s_N) &\forall j, l \neq i, \end{aligned}$$

and the value of the firm is the same for any two industry states with identical distribution. Players are not atomistic.¹¹

Under symmetry, the problem of a firm with **own state** s_i and **rivals' state** $\mathbf{s}_{-i} = (s_1, \dots, s_{i-1}, s_{i+1}, s_N)$ is independent of the firm i itself and solely dependent on the two states. Firms are symmetric when all heterogeneity is included in the state vector (s_i, \mathbf{s}_{-i}) , or

$$V_j(s_j, \mathbf{s}_{-i}) = V_i(s_i, \mathbf{s}_{-i}) = V(s_i, \mathbf{s}_{-i}) \text{ if } s_i = s_j. \quad \text{(Symmetry)}$$

Symmetry allows us to focus on the problem of solely one firm. Proposition 2 in Doraszelski and Satterthwaite (2010), provides sufficient conditions on the primitives, namely, that both period returns and the transition function are symmetric and anonymous. In this case, restricting to symmetric strategies, the value function inherits the two properties.

Identical states We say that two states are *identical* if the primitives are the same at each of those states, or $\mathbf{s}_{-i}(k)$ and $\mathbf{s}_{-i}(j)$ are *identical* if $\pi(s_i, \mathbf{s}_{-i}(j); a) = \pi(s_i, \mathbf{s}_{-i}(k); a)$ **and** $q(s'_i, \mathbf{s}'_{-i}, s_i, \mathbf{s}_{-i}(j), \boldsymbol{\alpha}) = q(s'_i, \mathbf{s}'_{-i}, s_i, \mathbf{s}_{-i}(k), \boldsymbol{\alpha})$ for any a and any $\boldsymbol{\alpha}_{-i}$. Two identical states can be aggregated without information loss, and the solutions to the original and the aggregated problems is the same. Symmetry and anonymity implies that symmetric and anonymous states are *identical*.

Under symmetry and anonymity we can redefine the domain of the industry state. Let $n_k = \sum_{j \neq i} \mathbf{1}(s_j = k)$ be the count of firms at each possible value $s_j = k$, $k = 1, \dots, K$. Let \mathbf{F} be the cumulative relative frequency of firms at each state $k = 1, \dots, K - 1$. We can ignore the last (K th) element because the cumulative fraction of firms in the highest state (K) is always equal to one. The $K - 1$ vector $\mathbf{F}_{\mathbf{s}_{-i}} = (\frac{n_1}{N-1}, \frac{n_1+n_2}{N-1}, \dots, \frac{n_1+n_2+\dots+n_{K-1}}{N-1})$ maps each particular rivals' state, \mathbf{s}_{-i} , into a probability space, where $\mathbf{F} : \tilde{S}^{N-1} \rightarrow \tilde{F} \equiv \{0, \frac{1}{N-1}, \dots, \frac{N-2}{N-1}, 1\}^{K-1}$ and N is the total number of players including player i ($N - 1$ is the number of rivals).¹² As an example, imagine the simplest case with two rivals and two possible states ($N - 1 = 2$ and $K = 2$). The industry distribution (cumulative frequency) can take the following values:

¹¹In the atomistic case and in the absence of aggregate uncertainty, the industry state distribution becomes deterministic, producing a perfect foresight equilibrium (Jovanovic and Rosenthal, 1988).

¹²The mass of players is constant at N . If the mass of incumbents evolves over time (as in exit/entry games) we can let $k = 0, 1, \dots, K$ where $k = 0$ now denotes the mass of potential entrants and $\sum_{k=1}^K n_k$ the mass of incumbents.

| | |
|----------------|--|
| $(n_1, n_2)/2$ | |
| (1, 1) | if both players are in state 1 |
| (0.5, 1) | if player 1 in state 1 and player 2 in state 2 |
| (0.5, 1) | if player 1 in state 2 and player 2 in state 1 |
| (0, 1) | if both players are in state 2 |

The subscript is used to emphasize that the \mathbf{F} states are "linked" to \mathbf{s}_{-i} , that is, it assigns (rivals') micro-states (\mathbf{s}_{-i}) to macro-state (\mathbf{F}), with \mathbf{F} being the cumulative relative frequency of the number of competitors at each state $\{1, 2, \dots, K - 1\}$.¹³ The redefined industry state is $(s_i, \mathbf{F}_{\mathbf{s}_{-i}}) \in \tilde{S} \times \tilde{F}$, where the cardinality of \tilde{F} is now the combination of possible cumulative frequency combinations (combinations of $N - 1$ from each of the $K + N - 2$ possibilities) which gives the binomial coefficient

$$|\tilde{F}| = \binom{K + N - 2}{N - 1}. \quad (4)$$

In the case presented above, $N - 1 = 2$ ($N - 1$ is the number of rival players) and $K = 2$ so that the number of combinations is $\binom{3}{2} = 3$.

The value function and strategies can be written as

$$V(s_i, \mathbf{s}_{-i}) = V_F(s_i, \mathbf{F}_{\mathbf{s}_{-i}}), \text{ and } \alpha^*(s_i, \mathbf{s}_{-i}) = \alpha_F^*(s_i, \mathbf{F}_{\mathbf{s}_{-i}})$$

where the subscript F emphasizes the new domain of the functions. These equalities are only true when $V(s_i, \mathbf{s}'_{-i}) = V(s_i, \mathbf{s}''_{-i})$ for any two states \mathbf{s}'_{-i} and \mathbf{s}''_{-i} such that $\mathbf{F}_{\mathbf{s}'_{-i}} = \mathbf{F}_{\mathbf{s}''_{-i}}$. In the example above, $\mathbf{F}_{\mathbf{s}'_{-i}} = (0.5, 1)$ when either $\mathbf{s}_{-i} = (1, 2)$, or $\mathbf{s}_{-i} = (2, 1)$. That is, the industry distribution is the same in the two industry states.

The symmetric and anonymous game successfully becomes *tractable*. The cardinality of the rivals' state is reduced from K^{N-1} ($\mathbf{s}_{-i} \in \tilde{S}^{N-1}$) to $\binom{K+N-2}{N-1}$ ($\mathbf{F} \in \tilde{F}$). Note that *intractable* is a problem that grows exponentially. Unfortunately, a problem that is *tractable* might still not be *computationally feasible* to solve. Table [2] reports state size comparisons for $K = 5$.¹⁴

4 Hard Aggregation Methods

We start this section by first defining hard aggregation. Let \mathbf{V} be a $|\tilde{S}|^N \times 1$ vector, Φ be a $|\tilde{S}|^N \times |\tilde{S}| \times |\tilde{P}|$ matrix, and \mathbf{r} a $|\tilde{S}| \times |\tilde{P}| \times 1$ vector. The dimensions of the objects are the number of possible industry states ($|\tilde{S}|^N$) and the number of approximated states ($|\tilde{S}| \times |\tilde{P}|$). In the approximated case we always keep track of the individual state and that is why we use ($|\tilde{S}| \times |\tilde{P}|$) instead of

¹³Note that \mathbf{F} is defined over the vector space \tilde{S}^{N-1} and not solely over the scalar space \tilde{S} . So, \mathbf{F} is not a probability measure because \mathbf{F} is (i) a vector and (ii) cannot add up to 1. Alternatively, we could define a proper cumulative probability function $\hat{F}_{\mathbf{s}_{-i}}$ that assigns to each $s \in \tilde{S}$ a probability on the set $\{0, \frac{1}{N-1}, \dots, \frac{N-2}{N-1}, 1\}$ and our previous function can be defined as $\mathbf{F}_{\mathbf{s}_{-i}} = (\hat{F}_{\mathbf{s}_{-i}}(1), \dots, \hat{F}_{\mathbf{s}_{-i}}(K-1))$, where $\sum_{k=1}^K \hat{F}_{\mathbf{s}_{-i}}(k) = 1$ and \hat{F} is a proper probability measure. The reason not to write the state this way is to make clear that s is not a random variable. The random variable here is the rivals' state, that is, the whole vector \mathbf{s}_{-i} (as well as $\mathbf{F}_{\mathbf{s}_{-i}}$).

¹⁴Although symmetry and anonymity successfully break the curse of dimensionality, it does not make problems *computationally feasible* because they are still too large to be solved. For example, the size of the state space with 100 firms is reduced from 7.9×10^{69} to 2.2×10^7 .

Table 2: Dimensionality of the industry state space for different number of firms with and without the symmetry and anonymity when $K = 5$.

| N | K^N | Symmetric |
|-----|----------------|------------|
| | | Anonymous |
| 1 | 5 | 5 |
| 2 | 25 | 25 |
| 3 | 125 | 75 |
| 4 | 625 | 175 |
| 5 | 3,125 | 350 |
| 6 | 15,625 | 630 |
| 7 | 78,125 | 1,050 |
| 8 | 390,625 | 1,650 |
| 9 | 1,953,125 | 2,475 |
| 10 | 9,765,625 | 3,575 |
| 15 | 30,517,578,125 | 15,300 |
| 20 | 9.54E+13 | 44,275 |
| 25 | 2.98E+17 | 102,375 |
| 50 | 8.88E+34 | 1,464,125 |
| 100 | 7.89E+69 | 22,106,375 |

($|\tilde{P}|$) as this way we emphasize that the approximated case is solved at each of the $|\tilde{S}|$ individual states. Aggregation is part of a more general class of approximation architectures where the vector \mathbf{V} in high dimension is approximated by $\Phi \mathbf{r}$ where \mathbf{r} is of much smaller dimensionality than \mathbf{V} . In this sense we try to find a lower dimensional space span that approximates reasonably well the higher dimensional space. The basis functions proposed by Farias et al. (2012) is part of the same class of approximation architectures. Aggregation is sufficiently general and can be applied to any type of Markov decision problem (see Van Roy, 2006). This incorporates both perfect and imperfect information Markov games (hidden Markov model when strategies are restricted to be Markov). For this section we will separate the own state (s_i) from the rivals' vector of states (\mathbf{s}_{-i}) and perform the aggregation over the rivals' state. The analysis is thus "conditional" on s_i .

Aggregation An aggregation can be described by two matrices: the aggregation matrix (the rule allocating micro states to macro states), and the disaggregation matrix (the rule splitting macro states into micro states). Take our main object of interest, the value function $V(s_i, \mathbf{s}_{-i})$, and let $\mathbf{V} = V(.,.)$ be the vector containing all the values V for each firm i , stacked over each possible state (s_i, \mathbf{s}_{-i}) . All possible (s_i, \mathbf{s}_{-i}) states are indexed along a single dimension. An aggregation, maps the vector \mathbf{V} to another vector \mathbf{V}_P at each alternative macro-state $(s_i, P) \in \tilde{S} \times \tilde{P}$ where \tilde{P} is the set of possible macro-states, and $|\tilde{P}| \ll |\tilde{S}|^{N-1}$. Let \mathbf{V} be the $|\tilde{S}|^N \times 1$ vector and Φ be a $|\tilde{S}|^N \times |\tilde{S}||\tilde{P}|$ (binary) **aggregation matrix** where there is a "1" in entry (j, k) when micro-state $(s_i, \mathbf{s}_{-i}(j))$ is aggregated in macro-state $(s_i, P_{\mathbf{s}_{-i}}(k))$, and a "0" otherwise. Again, the subscript is used to emphasize that the aggregation matrix links each \mathbf{s}_{-i} to each P . This (hard aggregation) matrix satisfies some properties: each row adds up to one, it is a binary matrix, each row contains only one element different from zero, and it has (full column) rank $|\tilde{S}||\tilde{P}|$.

For example, the case reported in Table [1] corresponds to the following aggregation matrices:

$$\hat{\Phi}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T, \hat{\Phi}_2 = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}^T$$

The difference between hard and soft aggregation is the restriction to mutually exclusive subsets. Soft aggregation schemes allow micro-states to belong to more than one macro-state, which is similar to basis function approximation. In this case, the rows of Φ add up to 1 but can have more than one element different from zero.

Ideally, we would like to find an aggregation matrix which would allow us to "recover" the original value function

$$\mathbf{V} = \Phi \mathbf{V}_P.$$

Unless $|\tilde{P}| = |\tilde{S}|^{N-1}$ (no dimension reduction), this equality cannot hold with hard aggregation (see below), except for the case where all the aggregated states are *identical* and have the same value, V (zero within macro-state variation). Also, the original function belongs in the space spanned by the columns of Φ . It is thus important that Φ does not include linearly dependent columns (full rank). As long as the aggregation matrix, Φ , is full rank and its columns form a basis (linear independence), soft or hard aggregation will span the same vector space, $\mathbb{R}^{|\tilde{S}||\tilde{P}|}$.

Definition 2 (Hard Aggregation) *Let \tilde{S}^N be the discrete set of industry states. An aggregation matrix, Φ maps any function with domain \tilde{S}^N to another function with domain $\tilde{S} \times \tilde{P}$, where \tilde{P} is the set of macro-states. An aggregation scheme is **hard** if the implied mapping, Φ is surjective, that is, every element in \tilde{S}^N maps to only one element in $\tilde{S} \times \tilde{P}$ (each row of Φ contains one and only one 1 and the matrix is full column rank).*

Disaggregation Paired with the **aggregation matrix**, there is a **disaggregation matrix**. It reassigns each macro-state to the micro-states. Let D^T be a $|\tilde{S}|^N \times |\tilde{S}||\tilde{P}|$ disaggregation matrix (where T denotes the matrix transpose). We can write

$$\mathbf{V}_P = D\mathbf{V},$$

and obtain a smaller vector \mathbf{V}_P out of the original vector \mathbf{V} . The disaggregation matrix satisfies the following properties: its columns add up to one, each element belongs to the unit interval, and each row contains only one element different from zero. It is similar to the probability of being in a given rivals state (\mathbf{s}_{-i}) conditional on observing the macro-state (P).

As disaggregation matrices we select the generalized inverses of Φ . That is, matrices D such that $\Phi D \Phi = \Phi$, where $D\Phi$ is an orthogonal projection ($D = (\Phi^T \Phi)^{-1} \Phi^T$). In general, there is an infinity of generalized inverse matrices of Φ . The three most common are (i) uniform, that is, equal weight to each micro-state within the macro state, $weight_i(\mathbf{s}_{-i}|P_{\mathbf{s}_{-i}}) = 1/(\#\mathbf{s}_{-i} \text{ in } P_{\mathbf{s}_{-i}})$;

(ii) invariant, that is, equal to the invariant long run equilibrium probability, conditional on being in the macro-state, $weight_i(\mathbf{s}_{-i}|P_{\mathbf{s}_{-i}}) = \Pr_i(\mathbf{s}_{-i}|P_{\mathbf{s}_{-i}})$, and (iii) mass point where one of the micro-states has probability one and all the remaining micro-states have probability zero.¹⁵

For the example reported in Table [1], using uniform weights we obtain

$$\hat{D}_1^T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{3} & \frac{1}{3} & 0 & \frac{1}{3} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{3} & 0 & \frac{1}{3} & \frac{1}{3} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T, \hat{D}_2^T = \begin{bmatrix} \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \end{bmatrix}^T,$$

In particular cases invariant weights have been shown to give a better approximation. Van Roy (2006) shows that the invariant weights reduce the error bound by a factor of $2/(1-\rho)$ for ρ close to 1 and Deng et al. (2011) formally prove the optimality of invariant weights under the Kullback Leibler metric, for the case of simple irreducible and aperiodic Markov Chains. However, the invariant disaggregation probabilities are difficult to derive as they require knowledge about the conditional probabilities originated from the invariant distribution, which are equilibrium objects that we do not know. To obtain them, requires knowing the solution to the dynamic game. One can still obtain approximate invariant disaggregation probabilities via simulation but this is subject to the curse of dimensionality. We will discuss this below.

4.1 The transition matrix

We need to introduce some further notation. Let each rivals' state $\mathbf{s}_{-i}(j) = (s_{-1}, \dots, s_{-N})$ be indexed along one single dimension $j = \{1, \dots, K^{N-1}\}$, and $Q_{\mathbf{s}}$ be the transition **matrix** for player i where a (k, j) entry

$$q(s'_i, \mathbf{s}'_{-i}(j), s_i, \mathbf{s}_{-i}(k), \boldsymbol{\alpha}_{-i}, a),$$

is the probability that industry state $(s_i, \mathbf{s}'_{-i}(j))$ is reached from industry state $\mathbf{s}_{-i}(k)$ when a player in state s_i chooses action a conditional on rivals' strategy profile $\boldsymbol{\alpha}_{-i}(\mathbf{s})$. A subscript \mathbf{s} indicates that the matrix is defined over the original set of rivals' states. Matrix $Q_{\mathbf{s}}$ has dimension K^N by K^N , which grows exponentially in N . The final "aggregated" transition matrix becomes

$$Q_P = DQ_{\mathbf{s}}\Phi$$

where a typical entry $q_P(s'P'; s, P, \boldsymbol{\alpha}_{-i}, a)$ is the probability that macro-state (s', P') is reached from macro-state (s, P) when player i chooses action a and rivals adopt the vector of strategies $\boldsymbol{\alpha}_{-i}(\mathbf{s})$. We now explain how the aggregated matrix is obtained using a simple example.

Aggregation The case reported in Table [1] corresponds to the following aggregation matrices:

$$\Phi_l = \begin{bmatrix} \hat{\Phi}_l & \mathbf{0} \\ \mathbf{0} & \hat{\Phi}_l \end{bmatrix} \text{ for } l = 1, 2$$

¹⁵This last one is the disaggregation matrix in the case of discretization methods (see below).

where the aggregation matrix is the same at each possible own state of firm i , $s_i = H, L$. That is why we have to "double" $\hat{\Phi}_i$, one for each possible $s_i = H, L$. Note that both Φ_1 and Φ_2 form a basis as there are no linearly dependent columns. Adding linearly dependent columns would be an inefficient way to organize information. Using the aggregation matrix we can construct the "aggregated" transition matrix $Q_s\Phi$. A typical element of matrix $Q_s\Phi$ is $\Pr(s'_i, P'_{s_{-i}}; s_i, \mathbf{s}_{-i}, \boldsymbol{\alpha}_{-i}, a)$, the probability that the macro-state $(s'_i, P'_{s_{-i}})$ is reached from the micro-state (s_i, \mathbf{s}_{-i}) when rivals use a given strategy profile $\boldsymbol{\alpha}_{-i}$ and player i chooses action a . As the original transition matrix (Q_s) is of dimension K^N by K^N , the "aggregated" transition matrix ($Q_s\Phi$) is of dimension K^N by $K|\tilde{P}|$.

Let $q_{(LLL)',(LLL)} = q(s'_i, LLL'; s_i, LLL, \boldsymbol{\alpha}_{-i}, a)$ be (s'_i, s_i) -specific and denotes the probability that rivals' state \mathbf{s}'_{-i} is reached from state \mathbf{s}_{-i} when players use a given strategy profile $\boldsymbol{\alpha}_{-i}$. The 16 by 16 transition matrix in our example is

$$Q_s = \begin{bmatrix} Q_{\mathbf{s}_{-i}}(s'_i = L, s_i = L) & Q_{\mathbf{s}_{-i}}(s'_i = H, s_i = L) \\ Q_{\mathbf{s}_{-i}}(s'_i = L, s_i = H) & Q_{\mathbf{s}_{-i}}(s'_i = H, s_i = H) \end{bmatrix}$$

where each 8 by 8 transition sub-matrix for each possible (s'_i, s_i) is

$$Q_{\mathbf{s}_{-i}}(s'_i, s_i) = \begin{bmatrix} q_{(LLL)',(LLL)} & q_{(LLH)',(LLL)} & \cdots & q_{(HHH)',(LLL)} \\ q_{(LLL)',(LLH)} & q_{(LLH)',(LLH)} & \cdots & q_{(HHH)',(LLH)} \\ \cdots & \cdots & \cdots & \cdots \\ q_{(LLL)',(HHH)} & q_{(LLH)',(HHH)} & \cdots & q_{(HHH)',(HHH)} \end{bmatrix},$$

The 16 by 8 transition matrix using aggregation matrix, Φ_1 , is

$$Q_s\Phi_1 = \begin{bmatrix} Q_{\mathbf{s}_{-i}}\Phi_1(s'_i = L, s_i = L) & Q_{\mathbf{s}_{-i}}\Phi_1(s'_i = H, s_i = L) \\ Q_{\mathbf{s}_{-i}}\Phi_1(s'_i = L, s_i = H) & Q_{\mathbf{s}_{-i}}\Phi_1(s'_i = H, s_i = H) \end{bmatrix},$$

and each 8 by 4 transition sub-matrix for each possible (s'_i, s_i) is

$$Q_{\mathbf{s}_{-i}}\Phi_1(s'_i, s_i) = \begin{bmatrix} q_{(LLL)',(LLL)} & \boxed{q_{(LLH)',(LLL)} + q_{(LHL)',(LLL)} + q_{(HLL)',(LLL)}} & \cdots & q_{(HHH)',(LLL)} \\ q_{(LLL)',(LLH)} & q_{(LLH)',(LLH)} + q_{(LHL)',(LLH)} + q_{(HLL)',(LLH)} & \cdots & q_{(HHH)',(LLH)} \\ \cdots & \cdots & \cdots & \cdots \\ q_{(LLL)',(HHH)} & q_{(LLH)',(HHH)} + q_{(LHL)',(HHH)} + q_{(HLL)',(HHH)} & \cdots & q_{(HHH)',(HHH)} \end{bmatrix}.$$

Each sub-matrix illustrates where information is lost during aggregation. Take the cell in row one, column two, which aggregates three continuation probabilities. If continuation values are the same at each of these three probabilities, the aggregation will have no "loss" because the sum of each probability times the continuation value equals the sum of the probabilities times the continuation value (which is the same for each micro-state). The approximation will deteriorate as the within-state variation increases, that is, as the range of continuation values of the aggregated micro-states becomes wider.

Disaggregation As for the disaggregation probability matrices, using uniform weights

$$D_l^T = \begin{bmatrix} \hat{D}_l & \mathbf{0} \\ \mathbf{0} & D_l \end{bmatrix} \text{ for } l = 1, 2$$

and the disaggregation is the same at each possible own state of firm i , $s_i = H, L$.

The "aggregated" 8 by 8 matrix is

$$D_1 Q_{\mathbf{s}} \Phi_1 = \begin{bmatrix} D_1 Q_{\mathbf{s}_{-i}} \Phi_1(s'_i = L, s_i = L) & D_1 Q_{\mathbf{s}_{-i}} \Phi_1(s'_i = H, s_i = L) \\ D_1 Q_{\mathbf{s}_{-i}} \Phi_1(s'_i = L, s_i = H) & D_1 Q_{\mathbf{s}_{-i}} \Phi_1(s'_i = H, s_i = H) \end{bmatrix}$$

where each 4 by 4 transition sub-matrix for each possible (s'_i, s_i) is

$$D_1 Q_{\mathbf{s}_{-i}} \Phi_1(s'_i, s_i) = \begin{bmatrix} q_{(LLL)',(LLL)} & q_{(LLH)',(LLL)} & q_{(LHH)',(LLL)} & q_{(HHH)',(LLL)} \\ +q_{(LHL)',(LLL)} & +q_{(HLH)',(LLL)} & +q_{(HHL)',(LLL)} & \\ \left(\begin{array}{c} q_{(LLL)',(LLH)} \\ +q_{(LLL)',(LHL)} \\ +q_{(LLL)',(HLL)} \end{array} \right) / 3 & \dots & \dots & \left(\begin{array}{c} q_{(HHH)',(LLH)} \\ +q_{(HHH)',(LHL)} \\ +q_{(HHH)',(HLL)} \end{array} \right) / 3 \\ \left(\begin{array}{c} q_{(LLL)',(LHH)} \\ +q_{(LLL)',(HLH)} \\ +q_{(LLL)',(HHL)} \end{array} \right) / 3 & \dots & \dots & \left(\begin{array}{c} q_{(HHH)',(LHH)} \\ +q_{(HHH)',(HLH)} \\ +q_{(HHH)',(HHL)} \end{array} \right) / 3 \\ q_{(LLL)',(HHH)} & q_{(LLH)',(HHH)} & q_{(LHH)',(HHH)} & q_{(HHH)',(HHH)} \\ +q_{(LHL)',(HHH)} & +q_{(HLH)',(HHH)} & +q_{(HHL)',(HHH)} & \end{bmatrix}$$

Again, each sub-matrix at state (s'_i, s_i) illustrates where information is "lost" during aggregation. Consider element in row two, column one. The aggregate transition probability from macro-state B to macro-state A (as defined in Table [1]) is $\left(q_{(LLL)',(LLH)} + q_{(LLL)',(LHL)} + q_{(LLL)',(HLL)} \right) / 3$. The true transition probability is

$$\left(q_{(LLL)',(LLH)} \Pr_i(LLH|B) + q_{(LLL)',(LHL)} \Pr_i(LHL|B) + q_{(LLL)',(HLL)} \Pr_i(HLL|B) \right)$$

where $\Pr_i(s_{-i}|P)$ are the invariant distribution weights, that is, the equilibrium probability conditional on being in the macro-state. In the symmetric and anonymous case, the equilibrium conditional probability is uniform, $\Pr(HLL|B) = 1/3$ and the uniform and invariant probabilities are equivalent. However, in the general case the uniform and invariant probabilities will differ. This suggests a preference for invariant weights, as it matches the true transition probabilities exactly.

Rivals' strategies The rivals follow the vector of strategies $\alpha_{-i}(s_i, \mathbf{s}_{-i})$ at each industry state (s_i, \mathbf{s}_{-i}) . However, such strategies cannot be calculated if we solve the aggregated problem. Instead, we obtain the approximated strategies in the set $\tilde{S} \times \tilde{P}$. Let the vector with these strategies be denoted α_i^P mapping the set of own and macro-states $\tilde{S} \times \tilde{P}$ to the set of possible

actions $\tilde{A}(\boldsymbol{\alpha}_i^P(s, P) : \tilde{S} \times \tilde{P} \rightarrow \tilde{A})$.¹⁶ Under the approximated strategies, the respective transition probability is $q(s'_i, \mathbf{s}'_{-i}; s_i, \mathbf{s}_{-i}, \boldsymbol{\alpha}_{-i}^P, a)$ and the transition matrix is.

$$Q_{\alpha_{-i}^P, P} = DQ_{\alpha_{-i}^P, \mathbf{s}} \Phi$$

where it is now made explicit the dependence between the transition matrix and the "approximated" rivals' strategies. The fact that we cannot solve for the exact strategies in the approximated problem will result in a second (second as compared to a single agent problem) source of "error" due to the effect on the rivals' strategies (see the bounds in Theorem [5.1]). This distinction is only relevant at equilibrium play.

Aggregation is not restricted to univariate own states. The aggregation rule defines how each micro-state is assigned to a macro-state. This can be done for any number of own states. In Appendix A.3 we provide an example with two states per player.

4.2 The Bellman operator

We now have all the elements to introduce the Bellman operator for both the original and the approximated problems. This representation is very convenient since these operators are conditional on the strategies of the other players and are thus not the equilibrium objects. For the original problem, let the Bellman operator **conditional on the strategies of the other players** ($\boldsymbol{\alpha}_{-i}$) be

$$(T_{\mathbf{s}, \alpha_{-i}} \mathbf{V}) = \max_a (T_{\mathbf{s}, \alpha_{-i}}^a \mathbf{V}) \quad \text{where } (T_{\mathbf{s}, \alpha_{-i}}^a \mathbf{V}) = \Pi_{\mathbf{s}} + \rho Q_{\mathbf{s}} \mathbf{V}, \quad \forall s, \mathbf{s}_{-i} \in \tilde{S}^N \quad (5)$$

and $\Pi_{\mathbf{s}}$ is the vector stacking all possible industry states (s, \mathbf{s}_{-i}) for a given action a . Note also that $Q_{\mathbf{s}}$ is a and $\boldsymbol{\alpha}_{-i}$ specific. For the approximated problem, let the Bellman operator conditional on the strategies of the other players ($\boldsymbol{\alpha}_{-i}^P$) be

$$(T_{P, \alpha_{-i}^P} \mathbf{V}_P) = \max_a (T_{P, \alpha_{-i}^P}^a \mathbf{V}_P) \quad \text{where } (T_{P, \alpha_{-i}^P}^a \mathbf{V}_P) = \Pi_P + \rho Q_{\alpha_{-i}^P, P} \mathbf{V}_P, \quad \forall s, P \in \tilde{S} \times \tilde{P} \quad (6)$$

and $\Pi_P = D\Pi_{\mathbf{s}}$. The transition matrix, as defined above, is $Q_{\alpha_{-i}^P, P} = DQ_{\alpha_{-i}^P, \mathbf{s}} \Phi$. Notice that $Q_{\alpha_{-i}^P, P}$ is the matrix defined over the aggregated set of states $(\tilde{S} \times \tilde{P})$ when rivals' strategies are also defined in the set $(\tilde{S} \times \tilde{P})$ whereas $Q_{\alpha_{-i}^P, \mathbf{s}}$ is the matrix defined over the original set of states (\tilde{S}^N) when rivals' strategies are defined over the set $(\tilde{S} \times \tilde{P})$.

Write $\mathbf{V}_{\Phi} = \Phi \mathbf{V}_P$ where the aggregation matrix Φ "expands" the vector \mathbf{V}_P in the aggregated set to the original set. The two problems relate to each other by

$$T_{P, \alpha_{-i}^P}^a \mathbf{V}_P = D \left(T_{\mathbf{s}, \alpha_{-i}^P}^a \mathbf{V}_{\Phi} \right).$$

¹⁶We can scale the strategies back to the original domain using the disaggregation matrix, as we did for the value function. If $\boldsymbol{\alpha}$ is the vector of strategies at each possible state and similarly $\boldsymbol{\alpha}^P$ is the vector of strategies at each possible aggregated state, we obtain $\boldsymbol{\alpha}^P = D\boldsymbol{\alpha}$. This simply sets the strategy at each macro-state (s, P) equal to an "average" of the strategies followed at each individual state (s, \mathbf{s}_{-i}) . If "similar" states are aggregated, the strategies at each micro-state belonging to a given macro-state will also be "similar".

In equilibrium, the optimal set of strategies are α^* for the original equilibria and α^{*P} for the approximated equilibria.

4.3 Implementation

To implement aggregation we can use the following step-by-step guide.

1. Define the aggregation architecture, Φ and solve the period-payoff of the full game, $\Pi_{\mathbf{s}}$.
2. Specify the initial disaggregation matrix D^{it} . If invariant weights are not available, use uniform weights at $it = 0$. These can be altered in step 5.
3. Specify some starting transition matrix $Q_{\alpha_{-i}, P}^{it}$, e.g. the identity matrix. Note: $Q_{\alpha_{-i}, \mathbf{s}}$ cannot be obtained due to the curse of dimensionality.
4. Solve the Bellman Equation [6] and obtain the solution $V_P^{it}(s, P)$ and $\alpha^{it}(s, P)$
5. Update $Q_{(\alpha_{-i}^p)^j, P}$. We cannot obtain $Q_{(\alpha_{-i}^p)^{it}, P}^{it} = D^{it}Q_{(\alpha_{-i}^p)^{it}, \mathbf{s}}\Phi$ or its invariant distribution numerically. Use simulation. Start from industry configurations $s \in \tilde{S}^N$ and simulate industry paths for T periods. Construct the observed transition matrix for the industry paths $Q_{(\alpha_{-i}^p)^{it}, P}^{it}$. The disaggregation matrix (D^{it}) determines how to perform the simulations. For example, with uniform weights, we let $T = 2$ and draw s uniformly from \tilde{S}^N . With invariant weights, we let $T \rightarrow \infty$ and calculate $Q_{(\alpha_{-i}^p)^{it}, P}$ from this sample. The two approaches work as follows:
 - a) **Uniform:** For each \mathbf{s} (or sampling uniformly if K^N is prohibitively large) use the optimal policies from step 4. and construct $\tilde{q}(s'_i, \mathbf{s}'_{-i}, \mathbf{s}, \alpha^P)$ via simulation. Aggregate by summing over all industry states

$$\tilde{q}(s'_i, P', \mathbf{s}, \alpha^P) = \sum_{\mathbf{s}'} \phi_{\mathbf{s}'(l), (s'_i, P')(r)} \tilde{q}(\mathbf{s}', \mathbf{s}, \alpha^P)$$

where $\phi_{\mathbf{s}'(l), (s'_i, P')(r)}$ is element (l, r) from matrix Φ . Disaggregate by taking the average across

$$\tilde{q}(s'_i, P', s_i, P, \alpha^P) = \sum_{\mathbf{s}} d_{\mathbf{s}(l), (s_i, P_{-i})(r)} \tilde{q}((s'_i, P_{-i}), \mathbf{s}, \alpha^P)$$

where $d_{\mathbf{s}(l), (s_i, P_{-i})(r)}$ is element (l, r) from (uniform) matrix D .

b) Invariant: Sample an element \mathbf{s} from K^N . Use the optimal policies from step 4. and draw from $q(\mathbf{s}', \mathbf{s}, \alpha^P)$. Repeat this for T periods (for $T \rightarrow \infty$) and obtain a series $(\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_T)$. Drop the first T_{start} periods. Use the remaining $T - T_{start}$ periods to calculate the transition under policies α_i^p , $Q_{(\alpha_{-i}^p)^j, P}$ and the invariant distribution.

6. Update $\Pi_P = D^{it}\Pi_{\mathbf{s}_{-i}}$ with the according disaggregation matrix. No update needed with uniform weights.

7. Iterate on 4-6 until $\|V_P^{it+1}(s, P) - V_P^{it}(s, P)\|_\infty < \varphi$, where φ is the tolerance level.

Step 5 calculates the transition matrix that lets us integrate over all the possible future industry states and calculate the continuation value when players behave according to the prescribed strategies $\alpha^j(s, P)$ at each macro-node, P . While it is conceptually straightforward to apply the aggregation and disaggregation matrices and obtain the transition analytically, this is subject to the curse of dimensionality. Instead, we use simulations to compute the transition matrix. Obviously this could run counter to the benefits from aggregation. However, although randomization is not successful for general nonlinear optimization, to deal with the curse of dimensionality arising from numerical integration of the expected continuation value we can use randomization, shown in Rust (1997) to break the curse of dimensionality of integration over the future states.¹⁷

Convergence The inner loop in Step 4, can be solved using value function iteration which has guaranteed convergence. It is a well known fact that dynamic games are prone to have multiple equilibria (e.g. Borkovsky, Doraszelski and Kryukov, 2010). Due to multiplicity, there may be multiple fixed points (multiple equilibria) which results in, not one, but a set of equilibrium strategies. A proof of convergence is thus not possible for the outer loop that finds the equilibrium strategies (fixed point) without further assumptions that guarantee uniqueness (e.g. concavity restrictions can guarantee a unique equilibrium).

4.4 Discretization vs. aggregation

The parallel between discretization and aggregation is known for MDPs (Markov decision processes - see Gordon, 1999, page 114 or Van Roy, 2006, page 243). "Discretization is closely related to state-aggregation; states are essentially partitioned into subsets, each of which is represented by a point in a grid". Consider the case with one continuous state taking values in the unit interval, $s \in [0, 1]$. A possible discrete grid with 5 points is $P \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$. This grid has an implicit aggregation rule. While there is an infinity of possible aggregation rules let it be uniform:

$$\begin{array}{ll} [0, 0.2] & P = 0.1 \\ (0.2, 0.4] & P = 0.3 \\ \text{if } s \in (0.4, 0.6] & P = 0.5 . \\ (0.6, 0.8] & P = 0.7 \\ (0.8, 1] & P = 0.9 \end{array}$$

Discretization imposes two constraints on the (infinite dimensional) aggregation matrix. First, the macro-states have to be members of set of micro-states. This is not required for general aggregation where the macro-states do not need to be members of the set of micro-states. Second, the disaggregation operator is mass point: equal to one when the micro-state

¹⁷Formulating the game in continuous time (Doraszelski and Judd, 2011) also breaks the curse of dimensionality of integration.

equals the macro-state and zero for all other micro-states. Some rows of matrix D^T are vectors of zeros, e.g., for any $s \in [0, 1] \setminus \{0.1\}$. This restriction simplifies the problem as all other micro-states can be ignored. Hybrid methods that use interpolation relax the hard aggregation assumption on the aggregation matrix and allow micro-states to be recovered as a weighted average of macro-states. For example $V(0.2) = \frac{1}{2}(V(P = 0.1) + V(P = 0.3))$.¹⁸

Aggregation combined with discretization can be applied to the cases with continuous states. For example, the actual game is defined over the continuous state s and instead we use the discrete grid P . There is an extra source of approximation error that emerges from using the discrete approximation to the continuous case and occurs even if we could solve the game exactly at the discrete grid states.¹⁹

5 Characterizing the aggregation: The bounds

We now use the framework outlined to characterize the error bound by decomposing it as a function of the primitives. This characterization is useful as it explains why some approximations work well while others do not. The error bound depends on the "difference" in the primitives of the aggregated micro-states, namely the profit and the transition functions. When *identical* states are aggregated (as is the case with symmetric and anonymous industry states), the bound is zero and the solution becomes exact. Furthermore, the matrix $(I - \Phi D)$ plays a central role. This matrix is a measure of within macro-state variability (see Equation [8] below). All proofs of this section are contained in the Supplementary Material online appendix.

We first establish two preliminary lemmas. The first lemma is similar to Theorem 3.1 in Whitt (1978) and establishes that an approximation is bounded above by the optimal greedy strategy, with the optimal greedy strategy being a one-step application of the Bellman operator to Equation [5].

Lemma 5.1 *Let Φ and D be the aggregation and disaggregation matrices. Let $T_{\mathbf{s}, \alpha_{-i}^*}$ be the Bellman operator in Equation [5] with solution \mathbf{V}^* , let \mathbf{V}_P^* be the solution to approximated problem in Equation [6] and $\mathbf{V}_\Phi^* = \Phi \mathbf{V}_P^*$. The error bound for the two solutions is*

$$\|\mathbf{V} - \mathbf{V}_\Phi\|_\infty \leq \frac{1}{1 - \rho} \|T_{\alpha_{-i}^*} \mathbf{V}_\Phi^* - \mathbf{V}_\Phi^*\|_\infty.$$

The Bellman operator conditions on the competitor's strategies. The difference between the two solutions is bounded by the one step greedy strategy of the full solution. The second lemma "divides" this unilateral deviation to the optimal Bellman operator ($T_{\alpha_{-i}^*} V_\Phi^* - V_\Phi^*$) into its individual components.

¹⁸Interpolation is regularly used with discretization to improve the quality of the approximation in cases with continuous states.

¹⁹Alternative methods to discretization can be used to approximate functions of continuous variables (see Judd, 1998 for a review of these alternatives). As long as there is sufficient smoothness, computational tools to solve continuous state problems are very different from the tools to solve discrete state problems. For example, Doraszelski (2003) uses projection methods.

Lemma 5.2 Let Φ and D be the aggregation and disaggregation matrices. Let Π_s be the profit function, $Q_{\alpha_{-i},s}$ the transition matrix when competitors follow strategies, α_{-i} and $Q_{\alpha_{-i}^P,s}$ the transition matrix when competitors follow strategies, α_{-i}^P . Let $T_{\alpha_{-i}}$ be the Bellman operator in Equation [5] with optimal solution \mathbf{V}^* , $T_{\alpha_{-i}}^a$ be the Bellman operator in Equation [6] with optimal solution \mathbf{V}_P^* and $\mathbf{V}_\Phi^* = \Phi \mathbf{V}_P^*$. The one step unilateral deviation is

$$T_{\alpha_{-i}} \mathbf{V}_\Phi^* - \mathbf{V}_\Phi^* = \max_a \left\{ \begin{array}{l} (I - \Phi D) \Pi_s \\ + \rho [(I - \Phi D) Q_{\alpha_{-i},s} \mathbf{V}_\Phi^*] \\ + \rho \left[\Phi D \left(Q_{\alpha_{-i},s} - Q_{\alpha_{-i}^P,s} \right) \mathbf{V}_\Phi^* \right] \\ + \Phi \left(T_{\alpha_{-i}}^a \mathbf{V}_P^* - \mathbf{V}_P^* \right) \end{array} \right\}.$$

The one step unilateral deviation to the full solution can be divided into three "gains": (i) from the approximated $(\Phi D \Pi_s)$ to the full profit (Π_s) ; (ii) from the approximated $(\Phi D Q_{\alpha_{-i},s})$ to the full transition $(Q_{\alpha_{-i},s})$ and; (iii) from the approximated $(Q_{\alpha_{-i}^P,s})$ to the full optimal rivals' strategies $(Q_{\alpha_{-i},s})$. The last (fourth) term is the one step operator to the full solution and it will be zero at the maximum, $\max_a T_{\alpha_{-i}}^a \mathbf{V}_P^* = \mathbf{V}_P^*$. With the two lemmas in hand, we can now characterize the three components of the error bound).

Theorem 5.1 Let Φ and D be the aggregation and disaggregation matrices. Let Π_s be the profit function, $Q_{\alpha_{-i},s}$ the transition matrix when competitors follow strategies, α_{-i} and $Q_{\alpha_{-i}^P,s}$ the transition matrix when competitors follow strategies, α_{-i}^P . Let \mathbf{V}^* be the optimal solution to Equation [5], \mathbf{V}_P^* the optimal solution to the aggregated problem in Equation [6] and $\mathbf{V}_\Phi^* = \Phi \mathbf{V}_P^*$. The approximation error is bounded as follows

$$\|\mathbf{V}^* - \mathbf{V}_\Phi^*\|_\infty \leq \frac{1}{1-\rho} \left\| \max_a \left\{ \begin{array}{l} (I - \Phi D) \Pi_s \\ + \rho [(I - \Phi D) Q_{\alpha_{-i},s} \mathbf{V}_\Phi^*] \\ + \rho \left[\Phi D \left(Q_{\alpha_{-i},s} - Q_{\alpha_{-i}^P,s} \right) \mathbf{V}_\Phi^* \right] \end{array} \right\} \right\|_\infty.$$

The error bound does not impose any restrictions to the primitives. It is thus fairly general. The result illustrates the three sources of approximation error: (i) the period returns $((I - \Phi D) \Pi_s)$, (ii) the transition matrix $((I - \Phi D) Q_{\alpha_{-i},s})$, and (iii) the difference in rivals' strategies $(Q_{\alpha_{-i},s} - Q_{\alpha_{-i}^P,s})$. The last term is specific to games and is not present in single agent problems.²⁰ There are cases where the bound is tight. If both primitives are perfectly aggregated $(\Pi_s = \Phi D \Pi_s)$, and $Q_{\alpha_{-i},s} = \Phi D Q_{\alpha_{-i}^P,s})$, the approximation is exact and the bound is tight (= 0). This is the case when aggregated states are *identical*. Another case occurs when the law of motion is independent from the actions of the players. In such case $Q_{\alpha_{-i},s} = Q_s = I$ is the same for all a and all α_{-i} so that $\mathbf{V}^* - \mathbf{V}_\Phi^* = \frac{1}{1-\rho} (I - \Phi D) \Pi_s + \frac{\rho}{1-\rho} (I - \Phi D) \mathbf{V}_\Phi^*$, and the bound holds with equality. Finally, when $\rho = 0$, then $\mathbf{V}^* - \mathbf{V}_\Phi^* = (I - \Phi D) \Pi_s$ which again delivers a tight bound. Section [A.4], reports numerical comparisons for this error bound.

²⁰One important note that this last difference holds for any set of rivals strategies, including the set of equilibrium strategies. In games with multiple equilibria there is more than one such equilibrium strategies and thus one bound for each of them. This last term only emerges because in solving the approximated game we obtain an approximate solution $(\alpha_{-i,s}^P)$ which is different from the exact solution $(\alpha_{-i,s}^*)$.

Theorem [5.1] illustrates the importance of within macro-state variation ($I - \Phi D$) for optimal aggregation. To minimize the first two terms, optimal aggregation has to minimize within macro-state variation ($I - \Phi D$). The next section discusses optimal aggregation. To do so, we have to impose structure and restrict to monotone symmetric and anonymous Markov games.

6 Monotonicity and Order Preserving Aggregation

An aggregation architecture requires the choice of two elements: (i) the number of macro-states/nodes ($|\tilde{P}|$) and (ii) the rule assigning each micro-state to one macro-state, that is, the aggregation matrix (Φ). We now restrict attention to symmetric and anonymous games where the value function is a function of own state and the distribution of rivals' states, $V_F(s_i, \mathbf{F}_{s_{-i}})$. When the value function satisfies monotonicity, we show that for a given number of macro-nodes, an optimal aggregation matrix (Φ) must be order preserving. That is, the rule assigning micro-states to macro states should preserve monotonicity. Monotonicity is not a restrictive condition since the equilibrium value function can always be sorted along any indexed set of states. Nonetheless, characterizing monotonicity from primitive conditions is very useful to guide the choice of the aggregation architecture and monotone games are part of a wide class of games (see Amir (2010), Milgrom and Roberts (1990) and Vives (1990)).

This section starts with the definitions of monotonicity and order preserving aggregation. It then shows that order preserving is a necessary condition of an optimal aggregation scheme (Theorem [6.2]). Finally, sufficient conditions are discussed.

6.1 Definitions

Let (\tilde{F}, \geq) be a partially ordered set where the order relation (\geq) is defined as follows.

Definition 3 (Order) For two elements $\mathbf{F} = (F_1, F_2, \dots, F_k), \mathbf{F}' = (F'_1, F'_2, \dots, F'_k) \in \tilde{F}$, $F_k \geq F'_k$ for all $k = 1, \dots, K - 1$.

As \mathbf{F} is a cumulative distribution, we can also say that \mathbf{F}' (weakly) first order stochastically dominates \mathbf{F} .

Definition 4 (Monotonicity) Let two states $\mathbf{F}, \mathbf{F}' \in \tilde{F}$. The value function is monotone in \mathbf{F} if for $\mathbf{F} \geq \mathbf{F}'$, $V_F(s, \mathbf{F}) \geq V_F(s, \mathbf{F}')$.

Assumption 1 The value function $V_F(s, \mathbf{F})$ is monotone in \mathbf{F} .

Theorem 2 in Amir (2010) provides particular conditions under which the value function is monotone. The result is an extension of the results on supermodular static games from Milgrom and Roberts (1990) and Vives (1990). The analysis is conducted in ordered topological spaces (lattices) and restricts to games with non deterministic transition (Q) and profit (Π) functions that are monotone, supermodular and have increasing differences (Assumptions **R3** and **T3**). This restriction excludes several games of interest. For example, the model in Besanko et al.

(2010) violates the supermodularity condition.²¹ An alternative to checking the conditions on the primitives is to directly check monotonicity for any particular value function.

Finally, the order is preserved under aggregation if micro-states maintain the same order relation in the macro-state:

Definition 5 (Order Preserving Aggregation) Let (\tilde{F}, \geq) and (\tilde{P}, \geq) be two partially ordered sets. An aggregation scheme between partially ordered sets $(\Phi : \mathbf{V}_\Phi = \Phi \mathbf{V}_P)$ is order preserving if for any $\mathbf{s}_{-i}, \mathbf{s}'_{-i}$ such that $\mathbf{F} \geq \mathbf{F}' \implies P \geq P'$.

6.2 Optimal Aggregation

The aggregation literature suggests aggregating micro-states that are **similar**. Two micro-states are **similar** if they have a **similar** value (Rogers et al., 1991). Intuitively, when **similar** micro-states are aggregated, the within macro-state variation is minimized. The same principle is used, for example, in statistical cluster analysis. More formally, an optimal aggregation minimizes the difference between the exact and the approximate solutions. This error difference can be decomposed as follows

$$\mathbf{V}_F^* - \mathbf{V}_\Phi^* = \Phi D (\mathbf{V}_F^* - \mathbf{V}_\Phi^*) + (I - \Phi D) \mathbf{V}_F^*, \quad (7)$$

because $\mathbf{V}_\Phi = \Phi \mathbf{V}_P$. So the error can be divided into an average component $(\Phi D (\mathbf{V}_F^* - \mathbf{V}_\Phi^*))$ and a variation $((I - \Phi D) \mathbf{V}_F^*)$. This difference is bounded by a term on the second component, when rivals' strategies are fixed. If rivals strategies are fixed, the bound is a function of the value function and the aggregation matrix.

Proposition 6.1 Let $D = (\Phi^T \Phi)^{-1} \Phi^T$ and $\alpha_{-i}^* = \Phi \alpha_{-i}^{*P}$,

$$\|\mathbf{V}_F^* - \mathbf{V}_\Phi^*\| \leq \frac{1}{1-\rho} \|(I - \Phi D) \mathbf{V}_F^*\|.$$

Ideally we would like to minimize the error instead of the bound. There are cases where the bound is sharp. If both primitives are perfectly aggregated ($\Pi_s = \Phi D \Pi_s$ and $Q_{\alpha_{-i}, s} = \Phi D Q_{\alpha_{-i}, s}^P$), the approximation is exact and the bound is sharp ($(I - \Phi D) \mathbf{V}_F^* = 0$). This is the case when aggregated states are *identical*. Also, when $\rho = 0$ the bound is sharp because $\mathbf{V}_F^* - \mathbf{V}_\Phi^* = (I - \Phi D) \Pi_s = (I - \Phi D) \mathbf{V}_F^*$. In general, however, the bound can be relatively loose. For example, when the law of motion is independent from the actions of the players, $Q_{\alpha_{-i}, s} = Q_s$ is the same for all a and all α_{-i} . In such case $\Phi D \mathbf{V}_F^* = \mathbf{V}_\Phi^*$ and the error becomes $\mathbf{V}_F^* - \mathbf{V}_\Phi^* = (I - \Phi D) \mathbf{V}_F^*$, so our error is $\frac{1}{1-\rho}$ times larger than the true bound. For a usual ρ close to 1, this

²¹A more subtle example is an entry game with two quality types (high and low). The value of being a high type monopolist is highest (say $V=3$). Having an extra low type competitors lowers my value (say $V=2$), while having two low type competitors increases my value (say $V=2.5$) so the value function is non-monotone in the number of low type competitors. This can be explained by a violation of increasing differences in the transition function. That is, with a low type competitor, there might be a high (transition) probability that second competitor enters the market with a high type, leaving the incumbent worse than when facing two low type competitors. On the other hand, having two low type competitors might keep the potential entrant outside, leaving the incumbent as the sole high type competitor.

constant term is considerably large. In Appendix [A.4], we study the relation between the error and the bound. The results suggest that the error regularly holds with a constant smaller than $\frac{1}{1-\rho}$. In this case, we can use a tighter bound $\|\mathbf{V}_F^* - \mathbf{V}_\Phi\| \approx f(\rho, \mathbf{V}_F^*)\|(I - \Phi D)\mathbf{V}_F^*\|$, where $f(\cdot)$ is some scaling constant, in many cases one order of magnitude smaller than $1/(1-\rho)$.

Our bounds are in line with the literature on state aggregation. For example, Tsitsiklis and Van Roy (1996) establish that for single agent aggregation problem, $\|\mathbf{V}_F^* - \mathbf{V}_\Phi^*\| \leq \frac{4\rho}{(1-\rho)^2} \min_{V_{\tilde{P}}^* \in \mathbb{R}^{|\tilde{P}|}} \|\mathbf{V}_F^* - \mathbf{V}_\Phi^*\|$. The minimization term makes this bound smaller, whereas the scaling constant makes this bound larger than the ones derived here. In Section [A.4] we provide some computational comparisons which suggest that both the bounds derived in Theorem [5.1] and Proposition [6.1] are overall smaller. This is because their bound is proportional to $\frac{4\rho}{(1-\rho)^2}$ although ours is proportional to $\frac{1}{1-\rho}$, and the proportional factor dominates for large ρ . Van Roy (2006) shows that if one uses the invariant distribution for constructing the disaggregation matrix, the proportionality factor is reduced by a constant $\frac{2}{(1-\rho)}$ for ρ close to 1. Of course this leads to a drastic improvement for large ρ , similar to our bounds.

Using Proposition [6.1] and following the same approach as Bertsekas and Castanon (1989, pp 591-592) we fix the number of nodes/macro-states, $|s|\tilde{P}|$ and select the aggregation scheme, Φ , that minimizes²² the within-state variation

$$\min_{\Phi \in \{0,1\}^{|\tilde{P}| \times |s|\tilde{P}|}} \|(I - \Phi D)\mathbf{V}_F^*\| \quad (8)$$

where for any vector \mathbf{v} , $(I - \Phi D)\mathbf{v}$ is a vector of residuals orthogonal to Φ . Let j be the j th element of vector \mathbf{V}_F^* (or $(s, \mathbf{F})(j)$) and assume uniform weights. Each j element of $(I - \Phi D)\mathbf{V}_F^*$ is

$$\begin{aligned} & ((I - \Phi D)\mathbf{V}_F^*)(j) \\ = & V_F^*((s, \mathbf{F}_{s-i})(j)) - \frac{\sum_{l,r} \phi_{(s, \mathbf{F}'_{s-i})(j), (s'_i, P')(r)} \phi_{(s, \mathbf{F}'_{s-i})(l), (s'_i, P')(r)} V_F^*((s, F)(l))}{\sum_{l,r} \phi_{(s, \mathbf{F}'_{s-i})(j), (s'_i, P')(r)} \phi_{(s, \mathbf{F}'_{s-i})(l), (s'_i, P')(r)}} \end{aligned}$$

where $\phi_{(s'_i, \mathbf{F}'_{s-i})(l), (s'_i, P')(r)}$ is element (l, r) from matrix Φ and is equal to 1 if state $(s'_i, \mathbf{F}'_{s-i})(l)$ is aggregated into macro-state $(s'_i, P')(r)$ and zero otherwise.²³ The previous expression shows that choosing the aggregation matrix is equivalent to choosing an aggregation architecture that minimizes the within macro-state variation. It is optimal to aggregate micro-states that have similar value, V_F^* . Thus, when the value function is monotone, any optimal aggregation must be order preserving, which explains the following result.

²²As explained in Deng et al. (2011) this optimization problem is itself subject to the curse of dimensionality. This means that even if we knew \mathbf{V}^* (which we don't), numerically finding the exact solution seems to be computationally impossible. For the case where \mathbf{V}^* is known, Jia (2011) derives an algorithm that finds the optimal state aggregation in polynomial time of the size of the state space. We follow an alternative route to numerical optimization, which is to establish necessary conditions on state aggregation.

²³ $I - \Phi D = I - \Phi(\Phi^T \Phi)^{-1} \Phi^T$ is the (idempotent) orthogonal projection matrix generated by the (full column rank) matrix Φ .

6.2.1 Necessary condition

Theorem 6.2 *Let (\tilde{F}, \geq) and (\tilde{P}, \geq) be two partially ordered sets. Let the value function, $V_F(s, F)$ be monotone in $F \in \tilde{F}$. The aggregation scheme between partially ordered sets $(\Phi : \mathbf{V}_\Phi = \Phi \mathbf{V}_P)$ is optimal only if it is order preserving.*

Theorem [6.2] lets us restrict attention to order preserving aggregation schemes in monotone games. The intuition is simple. An optimal architecture has to approximate well the short run profits, and the transition matrix (Theorem [5.1]). Order preserving aggregation lumps together micro-states that are "close" to each other, having a small within macro-state variation. Although this condition is necessary, it is not sufficient, as there is typically more than one order preserving aggregation architecture. For example, let there be five ordered micro-states and two macro-nodes. These can be aggregated as $\{1,2,3\}$ and $\{4,5\}$, thus satisfying order preserving. However, the fully optimal scheme could be $\{1,2\}$ and $\{3,4,5\}$. Also order preserving.

Monotonicity is in general not a restrictive condition since we can always sort the vector \mathbf{V}_F along any indexed set of states. However, given that we do not know \mathbf{V}_F , using the properties of the primitives to obtain such monotonicity is important so that we can check if a proposed aggregation is order preserving. Consider one particular, and intuitive, order preserving scheme for monotone symmetric and anonymous games: quantiles. Let $S_{\mathbf{s}_{-i}}^q$ be the q th quantile of the rivals' distribution

$$\{\min S^q \in \tilde{S} : \frac{1}{N-1} \sum_{k=1}^{S^j} n_k \geq q, \forall q \in [0, 1]\}.$$

There is a continuum of quantiles ($q \in [0, 1]$). For any finite set of R quantiles, the R -dimensional quantile **vector** is

$$\mathbf{P} = (S_{\mathbf{s}_{-i}}^{q1}, S_{\mathbf{s}_{-i}}^{q2}, \dots, S_{\mathbf{s}_{-i}}^{qR}) \in \tilde{P}, \text{ with cardinality } |\tilde{P}| = \binom{K+R-1}{R}. \quad (9)$$

that takes values from the set $\tilde{P} \subset \tilde{S}^R$. The dimensionality of the problem is unrelated with the number of firms because the cardinality of set, \tilde{P} , solely depends on K and R . The size grows with either R or K .²⁴

Definition 6 (Quantile Aggregation) *Let $\mathbf{F} \in \tilde{F}$ be the cumulative distribution of s_{-i} . The quantile aggregation is a matrix $\Phi : \mathbf{V}_F = \Phi \mathbf{V}_P$, and the macro-state, P , is the quantile of s_{-i} . $\mathbf{P}_{\mathbf{s}_{-i}}$, is the R dimensional vector $\mathbf{P} : \tilde{S}^{N-1} \rightarrow \tilde{P}$ as defined in equation [9].*

A quantile aggregation scheme \mathbf{P} is order preserving. To see this let us use a state definition that is slightly different. Let $\vec{\mathbf{s}}_{-i}$ be the $N-1$ -dimensional rivals' state, s_{-i} , sorted from smallest to largest. This is identical to \mathbf{F} because for any two states \mathbf{s}_{-i} and \mathbf{s}'_{-i} with $\mathbf{F}_{\mathbf{s}_{-i}} = \mathbf{F}'_{\mathbf{s}'_{-i}}$, $\vec{\mathbf{s}}_{-i} = \vec{\mathbf{s}}'_{-i}$. Now take two different states. If $\vec{\mathbf{s}}_{-i} \geq \vec{\mathbf{s}}'_{-i} \implies \mathbf{P}_{\vec{\mathbf{s}}_{-i}} \geq \mathbf{P}_{\vec{\mathbf{s}}'_{-i}}$. Note that, while not necessary, in some cases the quantiles (macro-states) can be a subset of the rivals states

²⁴We consider games with many firms, so that $N-1 > K$. If $N-1 < K$ the approximation becomes exact at $R = K$. However, in this case the approximation becomes exact at $R = N-1 < K$. See the discussion in Section 3 of Pakes, Gowrisankaran and McGuire, 1993, in particular their footnote 4.

(micro-states), $\tilde{P} \subset \tilde{F}$. Also, the quantile aggregation set can be identical to \tilde{F} , by choosing $N - 1$ equally spaced quantiles (no aggregation). This solves the problem exactly.

Example (quantiles) Let each player be in one of 3 possible states. Select the 50th and the 100th percentile of the industry distribution. Now imagine that our industry has 40% of firms in state 1, 30% are in state 2 and the remaining 30% are in state 3. The cumulative frequency is (0.4, 0.7, 1) and, as usual, the last state (3) has a cumulative frequency of 1. We can have several macro-states under the 50th and the 100th percentile. For example, (1, 2) means that 50% of the firms are in state 1 (or below) and 100% of the firms are in state 2 (or below). We can also have (1, 3) where the difference is that we now know that 100% of the firms are in state 3 (or below). Alternatively in state (2, 2) 50% of the firms are in state 2 (or below). To which macro-state should we assign our industry distribution (0.4, 0.7, 1)? Well, 50% of the firms are in state 2 or below (we have 40% in state 1 and 70% in state 2) while 100% of the firms are in state 3 or below so that the macro-state to which we should assign (0.4, 0.7, 1) is macro-state (2, 3) where 50% of the firms are in state 2 (or below) and 100% of the firms are in state 3 (or below). If our industry distribution was (0.2, 0.5, 1) it would be assigned to the same macro-state (2, 3), while if it was (0.4, 1, 1) it would be assigned to macro-state (2, 2).

Multiple own states The notion of quantile cannot be directly extended to the case with multiple states per firm. The reason is the lack of a natural order in multiple dimensions. Following the same reasoning, instead of defining the rivals' state with $N - 1$ rivals, we can define it with R rivals, attributing a weight of $\frac{N-1}{R}$ to each rival. By this, we effectively solve a game with R rivals instead of $N - 1$, each with a $\frac{N-1}{R}$ weight. This is the advantage of symmetry and anonymity. Let $F_{\mathbf{s}_{-i}}^R$ denote the industry distribution when we use R players, that can take $\binom{K+R-1}{R}$ values, whereas $F_{\mathbf{s}_{-i}}$ can take $\binom{K+N-2}{N-1}$ values. For example, let the industry state with 6 firms and two binary variables be $\mathbf{F}_{\mathbf{s}_{-i}} = \frac{1}{6} \begin{bmatrix} 2 & 3 \\ 3 & 6 \end{bmatrix}$. There are two firms in state (1,1), one firm in states (1,2), and (2,1) and two firms in state (2,2). The macro-state (with $R = 3$) to which this micro-state can be aggregated is $\mathbf{F}_{\mathbf{s}_{-i}}^R = \frac{1}{3} \begin{bmatrix} 1 & 2 \\ 2 & 3 \end{bmatrix}$ where each firm in the macro-state would get a weight of $\frac{N-1}{R} = 2$. This macro-state is "equivalent" to micro-state $\mathbf{F}_{\mathbf{s}_{-i}} = \frac{1}{6} \begin{bmatrix} 2 & 4 \\ 4 & 6 \end{bmatrix}$.

This R -player aggregation is a particular case of quantiles when R are equally spaced quantiles. It is less flexible, for one reason. By imposing equal spacing, it may prevent separating interesting market structures, such as the case of leading firms. Such restriction relates to the sufficiency condition. Equally spaced quantiles is one of many aggregation architectures. While non-equally spaced quantiles are likely to perform better, choosing such quantiles is a difficult (problem-specific) question that we now discuss.

6.2.2 Sufficient Conditions

Sufficient conditions let us select the best out of all order preserving schemes. For example, any quantile aggregation is order preserving, while there is an infinity of quantiles to select from. Drawing a parallel with other methods, in discretization we do not know which nodes are the best points to select in a discrete grid, or in basis functions we do not know which basis to select. Such choices depend on the shape of the value function, which is unknown. However, information about the primitives of the model (e.g. shape restrictions) can be used to improve the solution. For example, as we know Π_s , the quantiles can be selected to capture relevant industry features from the period returns by solving $\Phi^* = \arg \min_{\Phi \in \{0,1\}^{|\mathcal{S}||\tilde{P}| \times |\mathcal{S}||\tilde{P}|}} \|(I - \Phi D)\Pi_s\|$. This is the first, and normally the most significant, of the three sources of error in Theorem [5.1]. When highly concentrated industry structures have a very strong (steep) effect on profits, top quantiles will capture rivals' states with high concentration. This is the logic behind Ifrach and Weintraub's (2016) use of dominant firms. The reason is intuitive, in the regions where the value function is very steep we should aggregate fewer micro-states than in the regions where the value function is flat, thus minimizing the within macro-state variation. Again drawing a parallel with discretization, it is best to select non-uniform grids with more nodes in areas where the value function is steeper and less nodes in areas where the value function is flatter.²⁵

7 Empirical Illustration

To illustrate the applicability of the previous results we solve a dynamic reputation game for the Hotel industry in the city of Porto, Portugal. Hotels compete to attract customers and fill up rooms. Investing in customer satisfaction can help to build a visible reputation.

Reputation building has received increasing attention in online markets (for a review, see Tadelis, 2016). Reputation is a persistent demand element captured here by online user hotel ratings. User ratings can be divided into flow and stock. The **ratings** (flow) are the posted user assessments in any given period of time. The **ratings** (stock), on the other hand, represents the accumulated reviews over time. This distinction is important because the stock of user ratings is an important strategic asset that hotels can exploit but takes time, and effort to build. We thus model ratings (stock) as a measure of accumulated past investments into quality, and the flow as a measure of current investments into quality. This formulation is borrowed from the standard capital accumulation/quality ladder models with reviews playing the role of investment and ratings' stock playing the role of the stock of capital.

Accounting for persistence in strategic variables involves setting up a dynamic model of competition with heterogeneous players (Ericson and Pakes, 1995). Since both prices and ratings are perfectly observed by all the players, the perfect information MPE is a valid equilibrium concept. Furthermore, the Markovian nature of hotel ratings (where rating stock today is a function of

²⁵As an illustrative example take the following function, $v = \log(x)$. To approximate this function at some K aggregate states $x_k \in (0, 1]$ we should select states that are equidistant in v , not in x . For a concave function this means selecting more states close zero where the function is very steep and less states close to one where the function is less steep.

the stock in the previous period and the flow of new reviews) make this a plausible dynamic setting to adopt. In the hotel industry, the number of players can easily reach the hundreds, meaning that standard dynamic models cannot be solved due to the curse of dimensionality.

To estimate the game I collected individual rating and review evaluations for all hotels from online websites. Table 3 contains the descriptive statistics for the variables. A description of the industry evolution, data collection, treatment and statistics is reported in the Appendix.

Table 3: Descriptive statistics.

| | Obs | Mean | Std. Dev. | Min | Max |
|--------------------------|------|------|-----------|-----|-----|
| Rating (stock) 2005-2014 | 6403 | 3.0 | 1.4 | 1 | 5 |
| Rating (flow) 2005-2014 | 3812 | 3.9 | 0.8 | 1 | 5 |

Notes: Summary statistics for online ratings (stock and flow).

7.1 Parameterization

7.1.1 Static Demand

Several characteristics can influence the utility a customer obtains from staying at a hotel, such as its location, the number of stars, its price or its rating. The first two are fixed hotel characteristics and we will instead focus on the remaining: price and ratings. Price is used to fill up the hotel in any given month. Companies most commonly use BAR (Best Available Rate) pricing with the objective to maximize short term (month) profits. There are five possible rating levels ($K = 5$): very low, low, medium, high and very high.

Let us abstract from time t for the moment. Consumer $m = 1, \dots, M$ receives utility u_{im} from staying at hotel i

$$u_{im} = \beta s_i - \gamma p_i + v_{im}$$

where v_{im} are extreme value independent and identically distributed preference shocks, $p_i \in [0, \infty)$ is the hotel price and $\beta, \gamma \in [0, \infty)$ are the rating and price sensitivities, respectively. Integrating over v_{im} , market shares can be written as

$$\mu_i = \frac{\exp(\beta s_i - \gamma p_i)}{1 + \sum_{n=1}^N \exp(\beta s_n - \gamma p_n)} \quad (10)$$

where the market share, μ_i , is a function of $(s_i, \mathbf{s}_{-i}, p_i, \mathbf{p}_{-i})$. Marginal costs of production, c , are constant and equal for all firms. The Nash equilibrium for the pricing game²⁶ satisfies the following equation:

$$\max_{p_i} \frac{\exp(\beta s_i - \gamma p_i)}{1 + \sum_{n=1}^N \exp(\beta s_n - \gamma p_n)} M(p_i - c),$$

²⁶While prices can be used strategically and become a choice variable of the dynamic game, we will abstract from this and assume static pricing. This is also in line with pricing policies used in the hotel industry, where the most common is the BAR (best available rate) pricing. Prices are used to maximize expected short run monthly profits.

and equilibrium prices are implicitly defined by

$$p_i^* = c + \frac{1}{\gamma(1 - \mu_i^*)}. \quad (11)$$

Letting the cost of investment, g , be a privately observed i.i.d. cost shock drawn from a distribution $G(g)$ with $E(|g|) < \infty$, period profits are

$$\pi_i = \mu_i M \left(\frac{1}{\gamma(1 - \mu_i)} \right) - ga_i.$$

To estimate the cost of investing in reputation building we have to calibrate the parameters for the demand: β and γ . We borrow some parameters from Santos (2019). The price coefficient is set to $\beta = -0.085$. With 77 hotels this implies an average markup of about 12 euros per hotel room/night ($p_i^* - c = [\gamma(1 - \mu_i)]^{-1}$ where $\mu_i = 1/77$). This markup is about 20% when compared to the average room price of 60 euros reported in the industry (ahp-monitor.pt). The rating coefficient is set to $\gamma = 0.56$. It implies that an increase of one unit in the rating generates a 56% increase in demand. A one standard deviation is 0.8 units (Table 3) which corresponds to a 45% increase in demand.

7.1.2 Transitions

Individual ratings exhibit a very strong persistence, as illustrated by the transition matrix reported in Table [4]. Firms can improve the rating by investing $a_{it} \in [0, \bar{a}]$ units and this way influencing customer reviews. Although the map from reviews to ratings is deterministic (yet potentially unknown to the firm), the map from investment to reviews is stochastic, because firms cannot control what their customers will post (see the online appendix A.1 for a more detailed explanation). Companies expect investments to generate good on-line customer reviews.

Table 4: Transition matrix for individual ratings.

| | | Rating at t | | | | |
|-----|---|-------------|----|----|----|----|
| | | 1 | 2 | 3 | 4 | 5 |
| t-1 | 1 | 87 | 13 | 0 | 0 | 0 |
| | 2 | 3 | 94 | 3 | 0 | 0 |
| | 3 | 0 | 3 | 93 | 4 | 0 |
| | 4 | 0 | 0 | 4 | 93 | 2 |
| | 5 | 0 | 0 | 0 | 3 | 97 |

Number of Observations: 5842

We model the rating transition as a simple stochastic function of the investment. Let $h \in [0, \infty)$ be a parameter. The rating can increase to the next level with probability $\frac{ha_{it}}{1+ha_{it}}$ and will stay the same with probability $\frac{1}{1+ha_{it}}$. If the hotel does not invest in customer satisfaction ($a_{it} = 0$), the rating will decrease with (exogenous) probability parameter $\delta \in [0, \infty)$. The own

Table 5: Maximum likelihood estimates for the transition parameters.

| | Coef. | s.e. | pval |
|----------|----------|-------|------|
| δ | 0.96 | 0.004 | 0.00 |
| h | 12.04 | 0.008 | 0.00 |
| Obs. | 3,226 | | |
| Log-Lik. | -1081.72 | | |

Note: Maximum Likelihood estimates.

state transition $p(s'_i|s_i, a_i)$ is thus

$$p(s'_i = y|s_i = s, x) = \begin{cases} \frac{(1-\delta)ha}{1+ha} & \text{if } y = s + 1 \\ \frac{(1-\delta)+\delta ha}{1+ha} & \text{if } y = s \\ \frac{\delta}{1+ha} & \text{if } y = s - 1 \end{cases} . \quad (12)$$

We use this transition probabilities to obtain a maximum likelihood estimator for δ and h . Table [5] reports the results. The probability of moving down one level if no investment is made ($a = 0$) is estimated at $\hat{\delta} = 96\%$. On the other hand, the returns to investment are estimated to be $\hat{h} = 12.04$. This number implies that at the mean investment, the probability of remaining in the same own state is 91.9% (4.1% up and 4% down), whereas a one standard deviation increase in investment increases this probability to 92.8% (4.2% up and 3% down), and a one standard deviation decrease in investment decreases this probability to 89.8% (4.0% up and 6.1% down).

7.2 Estimator

We now introduce an estimator for the model. Solving the model even once is impossible due to the curse of dimensionality (the size of the state space is 6.6×10^{53}). The approximate solution will be fundamental at breaking the curse of dimensionality so we are able to solve the game numerically. Let us start by introducing the value for a firm in own state s_i and facing rivals' state \mathbf{s}_{-i} is

$$V_i^e(\mathbf{s}) = \int \left[\max_{a_i \in \bar{A}} \mu_i \left(\frac{1}{\gamma(1 - \mu_i)} \right) M - ga_{it} + \rho \sum_{\mathbf{s}' \in \tilde{S}^N} V_i^e(\mathbf{s}') q(\mathbf{s}', \mathbf{s}, a_i, \boldsymbol{\alpha}_{-i}) \right] dG(g), \quad \forall i, \mathbf{s}, \quad (13)$$

where M is the market size. The market share of hotel i (μ_i) is a function of (s_i, s_{-i}) and solves the system of equations [10] and [11].

The industry transition when firm i chooses action a_i , and the competitors adopt strategies $\boldsymbol{\alpha}_{-i}$ is $q(\mathbf{s}', \mathbf{s}, a_i, \boldsymbol{\alpha}_{-i})$. The full set of parameters in the problem are the price and rating sensitivity (γ and β , respectively), the transition parameters (δ and h), the discount factor (ρ), market size (M) and our main parameter of interest, the cost parameter, g .

Data is at a monthly frequency and the discount factor is set to $\rho = 0.98$, per month. The discount factor is normally not identified in dynamic models. A larger/smaller discount factor is directly translated into a larger/smaller estimated cost, g . If the discount factor is larger (closer to 1), it must be rationalized by larger costs, and vice-versa. Market size, M , is defined in terms

of rooms in a given month ($M = 428,571$). It equals the maximum number of people sleeping in the northern region (750,000 in August 2014) divided by 1.75 that is the average number of persons per room.²⁷

To recover the cost function we use revealed preferences and find the cost structure that rationalizes the observed decisions at the estimated static profits. We can write down a Least Square estimator for the cost parameter, g . First note that the continuation value has a convenient analytic expression

$$\begin{aligned} & \sum_{\mathbf{s}' \in \bar{S}^N} V_i^e(\mathbf{s}') q_i(\mathbf{s}', \mathbf{s}, a_i, \alpha_{-i}) \\ = & \sum_{\mathbf{s}_{-i,t+1}} \tilde{q}_{\alpha_{-i}, \mathbf{s}_{-i}, s_i} \left[\frac{(1-\delta)ha_{i,t}}{1+ha_{i,t}} V^e(s_{i,t} + 1, \mathbf{s}_{-i,t+1}) + \frac{1-\delta+\delta ha_{i,t}}{1+ha_{i,t}} V^e(s_{i,t}, \mathbf{s}_{-i,t+1}) \right. \\ & \left. + \frac{\delta}{1+ha_{i,t}} V^e(s_{i,t} - 1, \mathbf{s}_{-i,t+1}) \right], \end{aligned}$$

with $\tilde{q}_{\alpha_{-i}, \mathbf{s}_{-i}, s_i} = \Pr(\mathbf{s}_{-i,t+1} | s_{i,t}, \mathbf{s}_{-i,t}, \alpha_{-i})$ being the transition for the rivals' state of competition faced by firm i when rivals use strategy α_{-i} . The transition for s_i and \mathbf{s}_{-i} are conditionally independent given the transition structure formulated in Equation [12]. Assume for the moment that rivals use equilibrium strategies, $\alpha_{-i} = \alpha_{-i}^*$. Combining with Equation [13], we obtain the optimal interior solution to the problem, conditional on equilibrium play:

$$g = \rho \sum_{\mathbf{s}_{-i,t+1}} \tilde{q}_{\alpha_{-i}^*, \mathbf{s}_{-i}, s_i} \left[\frac{\partial \frac{(1-\delta)ha_{i,t}}{1+ha_{i,t}} V^e(s_{i,t} + 1, \mathbf{s}_{-i,t+1}) + \frac{\partial \frac{1-\delta+\delta ha_{i,t}}{1+ha_{i,t}} V^e(s_{i,t}, \mathbf{s}_{-i,t+1})}{\partial a}}{\partial a} \right. \\ \left. + \frac{\partial \frac{\delta}{1+ha_{i,t}} V^e(s_{i,t} - 1, \mathbf{s}_{-i,t+1})}{\partial x} \right].$$

This equation has one very convenient analytical expression that allows us to solve it as a function of a . Taking the logarithm we obtain

$$\begin{aligned} \ln g = \ln & \left[\rho h \sum_{\mathbf{s}_{-i,t+1}} \tilde{q}_{\alpha_{-i}^*, \mathbf{s}_{-i}, s_i} \left[(1-\delta) (V^e(s_{i,t} + 1, \mathbf{s}_{-i,t+1}) - V^e(s_{i,t}, \mathbf{s}_{-i,t+1})) \right. \right. \\ & \left. \left. + \delta (V^e(s_{i,t}, \mathbf{s}_{-i,t+1}) - V^e(s_{i,t} - 1, \mathbf{s}_{-i,t+1})) \right] \right] \quad (14) \\ & - 2 \ln(1 + ha_{i,t}). \end{aligned}$$

If V^e and α^* were known, all the elements on the right hand side of Equation [14] would be known and we could recover the distribution of g , using for example a non-parametric kernel density estimator. However, V^e is not known and depends on the distribution of g . We will assume $\ln g$ to be normally distributed²⁸ with mean $\overline{\ln g}$ and standard deviation $\sigma_{\ln g}$ and use the following least squares estimator

$$\min_{\ln g, \sigma_{\ln g}} \sum_{i=1}^N \sum_{t=1}^{T_i} (\ln g)_{it}^2. \quad (15)$$

Note that even though this is a least squares estimator, it is still computationally demanding

²⁷In the section with computational comparisons, error and bound calculations, market size is scaled by the number of firms $M = 428,571 * \frac{N}{77}$

²⁸We assume a parametric distribution because we have to take random draws from that distribution to solve the game numerically.

Table 6: Least squares estimates for the cost parameters.

| | Coef. | s.e. |
|------------|--------|------|
| \bar{g} | 8,289 | 236 |
| σ_g | 14,629 | . |
| Obs. | 3,846 | |
| R^2 | 0.56 | |

Note: Least sq. estimates.

because we do not know V^e and α^* . Both the transition and the value functions ($\tilde{q}_{\alpha^*_{-i}, \mathbf{s}_{-i}, s_i}$ and V^*) are unknown and vary with g . As such, we need to solve for the equilibrium of the game in order to recover the two objects of interest by minimizing Equation [15] over $\overline{\ln g}$ and $\sigma_{\ln g}$.²⁹ The heavy computational part of the estimation is to solve the dynamic game for each candidate solution $(\overline{\ln g}, \sigma_{\ln g}^2)$.

The approximate solution is fundamental to let us solve the game numerically. For the case with $N = 77$ hotels and $K = 5$ ratings levels, the total number of industry states $(s_{i,t}, s_{-i,t})$ is $5^{77} = 6.6 * 10^{53}$. Under symmetry and anonymity the number of possible industry distributions $(s_{i,t}, \mathbf{F}_{\mathbf{s}_{-i,t}})$ is $7.9 * 10^6$, about 8 million states. Using 5 quantiles (16.6, 33.3, 50, 66.6, 83.3) the size of $(s_{i,t}, \mathbf{P}_{\mathbf{s}_{-i,t}})$ is 630. Using the approximation we are able to solve the game in about 460 seconds.³⁰

7.3 Results

The results are presented in Table [6]. Average cost parameter is estimated at $\bar{g} = 8,289$. Combining with the review data a_{it} , this number is equivalent to an average of 32,878 euros spent per month ($\frac{1}{N} \sum_{i=1}^N \frac{1}{T_i} \sum_{t=1}^{T_i} \hat{g} a_{it} = 32,878$). We can express this in euros per room-day by considering an average of 30 days and 140 rooms. This average daily cost per room is about 7 euros and 82 cents, that is 14% of the average room price per day of 57 euros.

²⁹Optimization of Equation [15] is not trivial. To find the minimum to Equation [15] we start with a candidate pair $(\ln g^0, \sigma_{\ln g}^0)$, solve the game for this candidate parameter and update using the following update rule

$$\begin{aligned} \overline{\ln g}^{s+1} &= \overline{\ln g}^s + \frac{1}{N} \sum_{i=1}^N \frac{1}{T_i} \sum_{t=1}^{T_i} 2 \ln \left(\frac{1 + \hat{h} a^*(s_{i,t}, \mathbf{s}_{-i,t})}{1 + \hat{h} a_{i,t}} \right), \text{ and} \\ (\sigma_{\ln g}^2)^{s+1} &= (\sigma_{\ln g}^2)^s + \frac{1}{N} \sum_{i=1}^N \frac{1}{T_i} \sum_{t=1}^{T_i} 4 \ln \left(\frac{1 + \hat{h} a^*(s_{i,t}, \mathbf{s}_{-i,t})}{1 + \hat{h} a_{i,t}} \right)^2 \end{aligned}$$

where $(\overline{\ln g}, \sigma_{\ln g}^2)^{s+1}$ is the new estimate, $(\overline{\ln g}, \sigma_{\ln g}^2)^s$ is the old estimate, a_{it} is the reviews data and $\alpha^*(s_{i,t}, \mathbf{s}_{-i,t})$ is the equilibrium solution predicted by the model at each industry state $(s_{i,t}, \mathbf{s}_{-i,t})$.

³⁰Obtaining the transition matrix (kernel), requires the use of simulations. As such, the time spent can vary as we increase or decrease the number of draws. The reported figure uses a total of 44 million draws. We should emphasize that the absolute numbers depend on hardware, software, and code and should be interpreted with this in mind. Section A.4 contains the precise details.

8 Conclusion

In this article, I have developed an approximation method for dynamic games, based on state aggregation. The main advantage is that the cardinality of the problem becomes unrelated with the number of firms, letting us solve models with many players. I provide two main theorems on the performance of approximate aggregation methods and a necessary condition for any aggregate approximation method. I illustrate the method's applicability by building and estimating a dynamic reputation game for the Hotel industry, where there are 77 players. I hope the results open the venue for the application (and estimation) of dynamic games to a set of situations where it was previously not possible.

There are two promising extensions that I have not considered. First, Van Roy (2006) shows that using the invariant distribution allows us to establish a lower error bound for single agent MDPs. It is possible that this result can be extended to games. Second, instead of defining the aggregation matrix beforehand, Bertsekas and Castañon (1989) use an adaptive aggregation algorithm that iteratively selects states to aggregate using the transition matrices. This is expected to lead to an improvement in performance.

References

- [1] Aguirregabiria, V. and Mira, P. (2007) "Sequential Estimation of Dynamic Discrete Games", *Econometrica*, 75(1), 1-53.
- [2] Amir, R. (2010) "Discounted Supermodular Stochastic Games: Theory and Applications", manuscript
- [3] Bajari, P. Benkard, C. and Levin, J. (2007) "Estimating Dynamic Models of Imperfect Competition", *Econometrica*, 75(5), 1331-1370
- [4] Bertsekas, D. (2005) *Dynamic Programming and Optimal Control*, vols I and II, 3rd edition, Athena Scientific
- [5] Bertsekas, D. and Castanon, D. (1989) "Adaptive Aggregation Methods for Infinite Horizon Dynamic Programming", *IEEE Transactions on Automatic Control*, 34 (6), 589-598
- [6] Besanko, D., Doraszelski, U., Lu, L. and Satterthwaite, M. (2010), "Lumpy Capacity Investment and Disinvestment Dynamics", *Operations Research*, 58(4), 1178-1193
- [7] Borkovsky, R., Doraszelski, U. and Kryukov, Y. (2010) "A User's Guide to Solving Dynamic Stochastic Games Using the Homotopy Method", *Operations Research*, 58(4), 1116-1132
- [8] Che, H., Sudhir, K. and Seetharaman, P.B. (2007), "Bounded Rationality in Pricing under State-Dependent Demand: Do Firms Look Ahead, and if So, How Far?" *Journal of Marketing Research*, 44(3), 434-449

- [9] Chevalier J. and D. Mayzlin (2006), "The effect of word of mouth on sales: Online book reviews", *Journal of Marketing Research*, 43(3), 345–354
- [10] Chintagunta, P., Qin, M. and Vitorino, M.A. (2018) "Licensing and Price Competition in Tied-Goods Markets: An Application to the Single-Serve Coffee System Industry", *Marketing Science*, 37(6), 855-1052
- [11] Chow, C. and Tsitsiklis, J. (1991) "An Optimal One-Way Multigrid Algorithm for Discrete-Time Stochastic Control", *IEEE Transactions on Automatic Control*, 36 (8), 898-914
- [12] Deng, K., Mehta, P., and Meyn, S. (2011) "Optimal Kullback-Leibler aggregation via spectral theory of Markov chains," *IEEE Trans. Autom. Control*, 56 (12), 2793–2808
- [13] Doraszelski, U. (2003) "An R&D race with knowledge accumulation", *RAND Journal of Economics*, 34(1), 20-42
- [14] Doraszelski, U. and Satterthwaite, M. (2010) "Computable Markov-Perfect Industry Dynamics", *Rand Journal of Economics*, 41(2), 215-243
- [15] Dubé, J-P., Hitsch, G. and Manchanda, P. (2005) "An empirical model of advertising dynamics". *Quant. Marketing Econom*, 3(2), 107 –144
- [16] Dubé, J-P., Hitsch, G. and Rossi, P. (2010) "State dependence and alternative explanations for consumer inertia", *The RAND Journal of Economics*, 41(3), 417-445
- [17] Ellickson, P. Misra, S. and Nair, H. (2012) "Repositioning Dynamics and Pricing Strategy", *Journal of Marketing Research*, 49(6), 750-772
- [18] Ericson, R. and Pakes, A. (1995) "Markov-Perfect Industry Dynamics: A Framework for Empirical Work", *Review of Economic Studies*, 62 (1), 53-82
- [19] Fershtman, C. and Pakes, A. (2012) "Dynamics Games with Asymmetric Information: A Framework for Empirical Work", mimeo, Harvard University
- [20] Farias, V. Saure, D. and Weintraub, G. (2012) "An Approximate Dynamic Programming Approach to Solving Dynamic Oligopoly Models", *Rand Journal of Economics*, 43(2), 253-282
- [21] Farias, D. and Van Roy, B. (2003) "The Linear Programming Approach to Approximate Dynamic Programming", *Operations Research*, 51 (6), 850-865
- [22] Gallant, A. R., Hong, H. and Khwaja, A. (2018) "The Dynamic Spillovers of Entry: An Application to the Generic Drug Industry", *Management Science*, 64(3), 1189-1211
- [23] Geiger, B. and Temmel, C. (2014) Lumpings of Markov chains, entropy rate preservation, and higher-order lumpability, *Journal of Applied Probability*, 51 (4), 1114-1132

- [24] Goettler, R. and Gordon, B. (2011), "Does AMD spur Intel to innovate more?" *Journal of Political Economy*, 119(6), 1141-120
- [25] Goettler, R. and Gordon, B. (2014), "Competition and Product Innovation in Dynamic Oligopoly" *Quantitative Marketing and Economics*, 12(1), 1-42
- [26] Gordon, G. (1999) "Approximate solutions to Markov decision processes", Phd thesis, Carnegie Mellon University
- [27] Ifrach, B. and Weintraub, G. (2016) "A Framework for Dynamic Oligopoly in Concentrated Industries", manuscript
- [28] Jia, Q. (2011) "On State Aggregation to Approximate Complex Value Functions in Large-Scale Markov Decision Processes", *IEEE Transactions on Automatic Control*, Vol. 56(2), 333-344
- [29] Jovanovic, B., and Rosenthal, R. (1988) "Anonymous Sequential Games," *Journal of Mathematical Economics*, 17, 77–87.
- [30] Judd, K. (1998) *Numerical Methods in Economics*, MIT Press
- [31] Kim, M., Bradlow, E. and Iyengar, R. (2019) "Selecting Data Granularity Using the Power Likelihood", SSRN manuscript 3453170
- [32] Liu, Y. (2006), "Word of Mouth for Movies: Its Dynamics and Impact on Box Office Revenue," *Journal of Marketing*, 70 (July), 74–89.
- [33] Maskin, E. and Tirole, J. (1988) "A Theory of Dynamic Oligopoly, I: Overview and Quantity Competition with Large Fixed Costs", *Econometrica* 56(3), 549-569
- [34] Maskin, E. and Tirole, J. (2001) "Markov Perfect Equilibrium: I. Observable Actions"; *Journal of Economic Theory*, 100 (2), 191-219
- [35] Milgrom, P. and J. Roberts (1990) "Rationalizability and Learning in Games with Strategic Complementarities" *Econometrica*, 58, 1225-1278
- [36] Nair, H. (2019) "Diffusion and Pricing Over the Product Life Cycle", manuscript
- [37] Pakes, A., Gowrisankaran, G. and McGuire, P. (1993) "Implementing the Pakes-McGuire Algorithm for Computing Markov Perfect Equilibria in Gauss, mimeo, Harvard University
- [38] Pakes, A. and McGuire, P. (1994) "Computing Markov-perfet Nash Equilibria: Numerical Implications of a Dynamic differentiated Product Model", *RAND Journal of Economics*, 25(4)
- [39] Pakes, A. and McGuire, P. (2001) "Stochastic Algorithms, Symetric Markov Perfect Equilibrium, and the 'Curse' of Dimensionality", *Econometrica*, 69 (5), 1261-1281

- [40] Pakes, A. Ostrovsky, M. and Berry, S. (2007) "Simple Estimators for the Parameters of Discrete Dynamic Games, with Entry/Exit Examples", *Rand Journal of Economics*, 38(2), 373-399
- [41] Pesendorfer, M. and Schmidt-Dengler, P. (2008) "Asymptotic Least Squares Estimators for Dynamic Games", *Review of Economic Studies*, 75(3), 901-928
- [42] Porteus, E. (1980), "Overview of Iterative Methods for Discounted Finite Markov and Semi-Markov Decision Chains," in *Rec. Developments in Markov Decision Processes*, R. Hartley, L.C. Thomas and D. J. White (eds.), Academic Press, London
- [43] Rogers, D., Plante, R., Wong, R. and Evans. J. (1991) "Aggregation and Disaggregation Techniques and Methodology in Optimization, *Operations Research*, Vol. 39 (4), 553-581
- [44] Rust, J. (1996), "Numerical Dynamic Programming in Economics," *Handbook on Computational Economics*, vol 1, ch 14, 619–729
- [45] Rust, J. (1997), "Using Randomization to Break the Curse of Dimensionality", *Econometrica*, vol 65 (3), 487–516
- [46] Santos, C. (2017), "Sunk Costs of R&D, Trade and Productivity: The moulds industry case", *The Economic Journal*, 127(603), 1626-1664
- [47] Santos, C. (2019) "Online Word-of-Mouth and Market Structure", manuscript
- [48] Schmidt-Dengler, P. (2006) "The timing of New Technology Adoption: The case of the MRI", manuscript
- [49] Serfling, R. (2002) "Quantile functions for multivariate analysis: approaches and applications", *Statistica Neerlandica*, 56 (2), 214-232
- [50] Shen, Q. (2014) "A Dynamic Model of Entry and Exit in a Growing Industry", *Marketing Science*, 33(5), 712-724
- [51] Srisuma, S. (2007) "Minimum Distance Estimators for Dynamic Games", *Quantitative Economics*, 4, 549-583
- [52] Tadelis, S. (2016) "Reputation and Feedback Systems in Online Platform Markets", *Annual Review of Economics*, 8
- [53] Topkis, D. (1998) *Supermodularity and Complementarity*, Princeton University Press
- [54] Tsitsiklis, J. and Van Roy, B. (1996) "Feature-Based Methods for Large Scale Dynamic Programming", *Machine Learning*, 22, 59-94
- [55] Van Roy, B. (2006) "Performance Loss Bounds for Approximate Value Iteration with State Aggregation", *Mathematics of Operations Research*, 31 (2), 234-244

- [56] Vitorino, M. A. (2012) "Empirical Entry Games with Complementarities: An Application to the Shopping Center Industry", *Journal of Marketing Research*, 49(2), 175-191
- [57] Vives, X. (1990) "Nash Equilibrium with Strategic Complementarities", *Journal of Mathematical Economics*, 19, 305-321
- [58] Weintraub, G. Benkard, L. and Van Roy, B. (2008) "Markov Perfect Industry Dynamics with Many Firms", *Econometrica*, 76 (6), 1375-1411
- [59] Whitt, W. (1978) "Approximations of Dynamic Programs, I", *Mathematics of Operations Research*, 3 (3), 231-243
- [60] Whitt, W. (1980) "Representation and Approximation of NonCooperative Sequential Games", *Siam Journal of Control and Optimization*, 18 (1), 33-48

A Online Appendix

A.1 Data Appendix

Industry evolution Following the entry of low cost airline carriers in 2009, **average** industry ratings recovered from a steady decline (Figure [A1]). Both the number of hotels and their quality (measured by ratings) have increased, as a result of the larger number of tourists in town. Although it is easy to explain the increase in the number of hotels as a response to the increase in the number of tourists, it is less clear why ratings have increased. To understand this, notice that user ratings (both the flow and the stock) exhibit a skewed distribution - Figure [A2], with a larger fraction of ratings at higher values (4 and 5). This fact has been documented in previous studies, e.g. Chevalier and Mayzlin, 2006; Liu, 2006. The observed changes in average ratings are explained by the shifting distribution at the tails, as reported in Figure [A3].

Data Collection User generated content data is collected from Tripadvisor. This involves collecting individual hotel information together with all the individual reviews. We obtain a total of 20,388 reviews. The second source of data is aggregate (market level) stays at all hotels in the northern region of Portugal obtained from <http://www.turismodeportugal.pt/Português/ProTurismo/estatísticas/quadrosestatisticos/dormidas/Pages/Dormidas.aspx>. We define the market to be the whole northern region for convenience because municipality (Porto) level data is not available with a monthly frequency (only yearly). The city of Porto represents about 40% of the total stays in the northern region (Pordata.pt).

Data description and treatment There is a total of 89 hotels in Porto. From the latitude and longitude data we identify seven hotels that have either changed name and code or have double entries on tripadvisor and merge them together. We also drop two hotels that located outside of Porto (AS Hotel Agua Santas and Trajano) and three other hotels for which there is no information besides the its name (Residencial Santo Antonio, Residencial S. Marcos and Porto Center Hotel). We end up with a total of 77 hotels. Two hotels have no customer reviews, that is, no rating (Morro do Sol, and HF Tuela Porto Ala Sul).

We do not observe investment directly. Instead, we observe the reviews as posted by each individual customer. We model a directed relation from investment to reviews to ratings. To obtain a measure of the investment a , we use the monthly average of these reviews that take values on $[1,5]$.

We can write ratings as a stochastic function of investment. In particular we use the following parametric function

$$review = \frac{5 \exp(a + \zeta) - 3}{\exp(a + \zeta) + 1}$$

where $\zeta \in [0, \infty)$ is a stochastic component. The choice of this parametric function is related to the fact that investment a can take any non-negative value $a \in [0, \infty)$ whereas the reviews

take values on the interval $[1, 5]$. This parametric function maps one into the other. Because we would like to have a measure of investment defined on $[0, \infty)$, we let $\tilde{a} = a + \zeta$ and use the following transformation

$$\tilde{a} = -\ln\left(\frac{8}{\text{review} + 3} - 1\right)$$

This approach is similar to the innovation literature when only the output measures of innovation are observed whereas the actual innovative efforts are unobserved.

Figure A1: User ratings: time series average for 2004-2014.

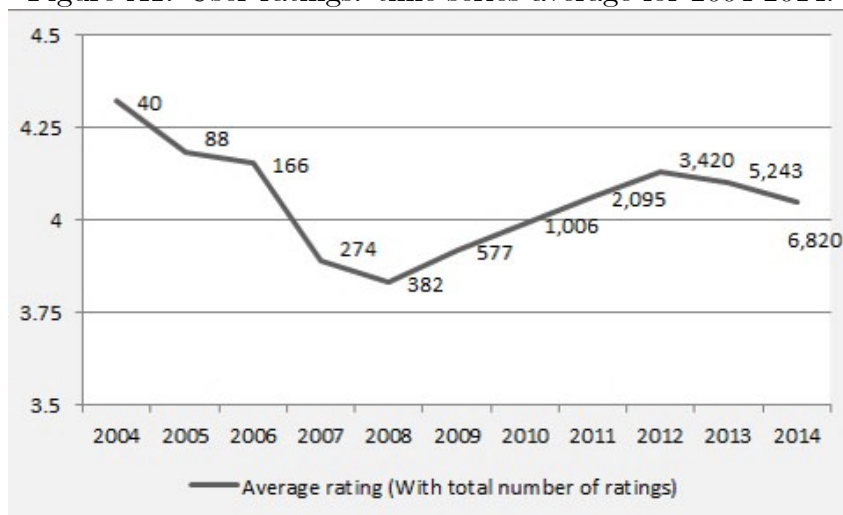


Figure A2: User ratings: Histogram for 2004-2014.

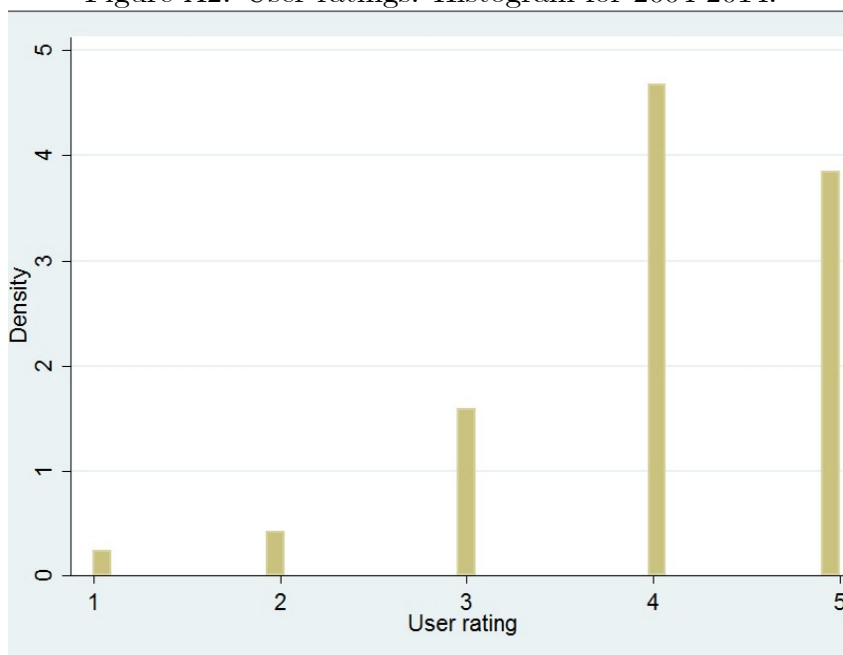
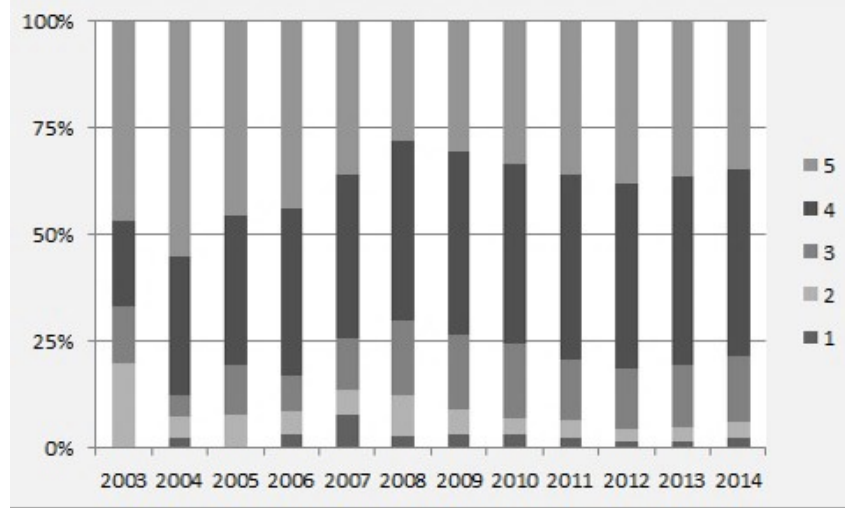


Figure A3: User ratings: Evolution of distribution for 2004-2014.



A.2 Proofs

A.2.1 Proof of Lemma [5.1]

Proof. First note that

$$\begin{aligned}
 \|\mathbf{V}^* - \mathbf{V}_\Phi^*\|_\infty &= \|T_{s,\alpha-i} \mathbf{V}^* - \mathbf{V}_\Phi^*\|_\infty \\
 &= \|T_{s,\alpha-i} \mathbf{V}_\Phi^* - T_{s,\alpha-i} \mathbf{V}_\Phi^* + T_{\alpha-i} \mathbf{V}^* - \mathbf{V}_\Phi^*\|_\infty \\
 &= \|T_{s,\alpha-i} \mathbf{V}_\Phi^* - \mathbf{V}_\Phi^* + T_{s,\alpha-i} \mathbf{V}^* - T_{s,\alpha-i} \mathbf{V}_\Phi^*\|_\infty \\
 &\leq \|T_{s,\alpha-i} \mathbf{V}_\Phi^* - \mathbf{V}_\Phi^*\|_\infty + \|T_{s,\alpha-i} \mathbf{V}^* - T_{s,\alpha-i} \mathbf{V}_\Phi^*\|_\infty \\
 &\leq \frac{1}{1-\rho} \|T_{s,\alpha-i} \mathbf{V}_\Phi^* - \mathbf{V}_\Phi^*\|_\infty
 \end{aligned}$$

where the first inequality follows from the triangle inequality. The second inequality follows from $T_{s,\alpha-i}$ being a contraction and $\|T_{s,\alpha-i} \mathbf{V}^* - T_{s,\alpha-i} \mathbf{V}_\Phi^*\|_\infty \leq \rho \|\mathbf{V}^* - \mathbf{V}_\Phi^*\|_\infty$. This intermediate result is similar to Theorem 3.1 in Whitt (1978). ■

A.2.2 Proof of Lemma [5.2]

Proof. For $Q_{\alpha_{-i},P} = DQ_{\alpha_{-i},s}\Phi$ notice that

$$\begin{aligned}
 & T_{s,\alpha_{-i}} \mathbf{V}_{\Phi}^* - \mathbf{V}_{\Phi}^* \\
 &= \max_a \left\{ \mathbf{\Pi}_s + \rho Q_{\alpha_{-i},s} \mathbf{V}_{\Phi}^* \right\} - \mathbf{V}_{\Phi}^* \\
 &= \max_a \left\{ \mathbf{\Pi}_s + \rho Q_{\alpha_{-i},s} \mathbf{V}_{\Phi}^* - \mathbf{V}_{\Phi}^* \right\} \\
 &= \max_a \left\{ \begin{array}{l} \Phi D \mathbf{\Pi}_s + \rho \Phi Q_{\alpha_{-i},P} \mathbf{V}_P^* + \mathbf{\Pi}_s - \Phi D \mathbf{\Pi}_s \\ + \rho Q_{\alpha_{-i},s} \Phi \mathbf{V}_P^* - \rho \Phi Q_{\alpha_{-i},P} \mathbf{V}_P^* - \Phi \mathbf{V}_P^* \end{array} \right\} \\
 &= \max_a \left\{ \begin{array}{l} \Phi \left(T_{P,\alpha_{-i}}^a \mathbf{V}_P^* - \mathbf{V}_P^* \right) + \mathbf{\Pi}_s - \Phi D \mathbf{\Pi}_s \\ \rho Q_{\alpha_{-i},s} \Phi \mathbf{V}_P^* - \rho \Phi Q_{\alpha_{-i},P} \mathbf{V}_P^* \end{array} \right\} \\
 &= \max_a \left\{ \begin{array}{l} (I - \Phi D) \mathbf{\Pi}_s \\ + \rho [(I - \Phi D) Q_{\alpha_{-i},s} \mathbf{V}_{\Phi}^*] \\ + \rho \left[\Phi D \left(Q_{\alpha_{-i},s} - Q_{\alpha_{-i},P} \right) \mathbf{V}_{\Phi}^* \right] \\ + \Phi \left(T_{\alpha_{-i}}^a \mathbf{V}_P^* - \mathbf{V}_P^* \right) \end{array} \right\}
 \end{aligned}$$

where in the first step we use the Bellman operator defined in Equation 5, the second step is obtained because \mathbf{V}_{Φ}^* does not depend on a , the third step we just add and subtract $\Phi D \mathbf{\Pi}_s + \rho \Phi Q_{\alpha_{-i},P} \mathbf{V}_P^*$, the fourth step uses the Bellman operator defined in Equation 6, $T_{P,\alpha_{-i}}^a \mathbf{V}_P^* = \Phi D \mathbf{\Pi}_s + \rho \Phi Q_{\alpha_{-i},P} \mathbf{V}_P^*$, the fifth step adds and subtracts $\rho \Phi D Q_{\alpha_{-i},s} \mathbf{V}_{\Phi}^*$ and uses the definition $Q_{\alpha_{-i},P} = DQ_{\alpha_{-i},s}\Phi$. ■

A.2.3 Proof of Theorem [5.1]

Proof. From Lemmas 5.1 and 5.2 ■

$$\begin{aligned}
\|\mathbf{V}^* - \mathbf{V}_\Phi^*\|_\infty &\leq \frac{1}{1-\rho} \|T_{s,\alpha_{-i}} \mathbf{V}_\Phi^* - \mathbf{V}_\Phi^*\|_\infty \\
&\leq \frac{1}{1-\rho} \left\| \max_a \left\{ \begin{array}{l} (I - \Phi D) \mathbf{\Pi}_s \\ +\rho [(I - \Phi D) Q_{\alpha_{-i},s} \mathbf{V}_\Phi^*] \\ +\rho [\Phi D (Q_{\alpha_{-i},s} - Q_{\alpha_{-i}^P,s}) \mathbf{V}_\Phi^*] \\ +\Phi (T_{P,\alpha_{-i}^P}^a \mathbf{V}_P^* - \mathbf{V}_P^*) \end{array} \right\} \right\|_\infty \\
&\leq \frac{1}{1-\rho} \left\| \max_a \left\{ \begin{array}{l} (I - \Phi D) \mathbf{\Pi}_s \\ +\rho [(I - \Phi D) Q_{\alpha_{-i},s} \mathbf{V}_\Phi^*] \\ +\rho [\Phi D (Q_{\alpha_{-i},s} - Q_{\alpha_{-i}^P,s}) \mathbf{V}_\Phi^*] \\ +\Phi (T_{P,\alpha_{-i}^P} \mathbf{V}_P^* - \mathbf{V}_P^*) \end{array} \right\} \right\|_\infty \\
&\leq \frac{1}{1-\rho} \left\| \max_a \left\{ \begin{array}{l} (I - \Phi D) \mathbf{\Pi}_s \\ +\rho [(I - \Phi D) Q_{\alpha_{-i},s} \mathbf{V}_\Phi^*] \\ +\rho [\Phi D (Q_{\alpha_{-i},s} - Q_{\alpha_{-i}^P,s}) \mathbf{V}_\Phi^*] \end{array} \right\} \right\|_\infty
\end{aligned}$$

where in the second step we use the triangle inequality and in the third step we use the Bellman operator defined in Equation 6 $\mathbf{V}_P^* = T_{\alpha_{-i}^P} \mathbf{V}_P^*$.

A.2.4 Proof of Proposition[6.1]

We first establish the following Lemma.

Lemma A.1 *Let $D = (\Phi^T \Phi)^{-1} \Phi^T$ and $\alpha_{-i}^* = \Phi \alpha_{-i}^{*P}$, then*

$$\|\Phi D (\mathbf{V}^* - \mathbf{V}_\Phi^*)\| \leq \rho \|\mathbf{V}^* - \mathbf{V}_\Phi^*\|$$

Proof.

$$\begin{aligned}
&\|\Phi D (\mathbf{V}^* - \mathbf{V}_\Phi^*)\| \\
&= \left\| \Phi D \left(\max_a \left\{ \mathbf{\Pi}_s + \rho Q_{\alpha_{-i}^*,s} \mathbf{V}^* \right\} - \max_a \left\{ \Phi D \mathbf{\Pi}_s + \rho \Phi D Q_{\alpha_{-i}^{*P},s} \mathbf{V}_\Phi^* \right\} \right) \right\| \\
&\leq \left\| \Phi D \left(\max_a \left\{ \mathbf{\Pi}_s + \rho Q_{\alpha_{-i}^*,s} \mathbf{V}^* - \Phi D \mathbf{\Pi}_s - \rho \Phi D Q_{\alpha_{-i}^{*P},s} \mathbf{V}_\Phi^* \right\} \right) \right\| \\
&= \left\| \Phi D \max_a \left\{ \begin{array}{l} (I - \Phi D) \mathbf{\Pi}_s + \rho (Q_{\alpha_{-i}^*,s} - Q_{\alpha_{-i}^{*P},s}) \mathbf{V}^* \\ +\rho (Q_{\alpha_{-i}^{*P},s} \mathbf{V}^* - \Phi D Q_{\alpha_{-i}^{*P},s} \mathbf{V}_\Phi^*) \end{array} \right\} \right\| \\
&\leq \left\| \max_a \left\{ \begin{array}{l} \Phi D (I - \Phi D) \mathbf{\Pi}_s + \rho \Phi D (Q_{\alpha_{-i}^*,s} - Q_{\alpha_{-i}^{*P},s}) \mathbf{V}^* \\ +\rho \Phi D (Q_{\alpha_{-i}^{*P},s} \mathbf{V}^* - \Phi D Q_{\alpha_{-i}^{*P},s} \mathbf{V}_\Phi^*) \end{array} \right\} \right\| \\
&= \left\| \max_a \left\{ \rho \Phi D (Q_{\alpha_{-i}^*,s} - Q_{\alpha_{-i}^{*P},s}) \mathbf{V}^* + \rho \Phi D Q_{\alpha_{-i}^{*P},s} (\mathbf{V}^* - \mathbf{V}_\Phi^*) \right\} \right\| \\
&= \rho \left\| \max_a \left\{ \Phi D Q_{\alpha_{-i}^{*P},s} (\mathbf{V}^* - \mathbf{V}_\Phi^*) \right\} \right\| \\
&\leq \rho \|\mathbf{V}^* - \mathbf{V}_\Phi^*\|
\end{aligned}$$

where the first step follows from the reverse triangle inequality, the second step is just adding and subtracting $\rho Q_{\alpha_{-i}^{*P}, \mathbf{s}} \mathbf{V}^*$, the third step follows from Jensen's inequality applied to a concave operator, the fourth from $D\Phi D = I$, the fifth from $\alpha_{-i}^{*P} = \alpha_{-i}^*$. Finally notice that for any vector v , $\left\| \Phi D Q_{\alpha_{-i}^{*P}, \mathbf{s}} v \right\| \leq \|v\|$. This is because the left hand side term is a weighted average of v , which under the supremum norm has to be $\leq \|v\|$. The only element that depends on a is $Q_{\alpha_{-i}^{*P}, \mathbf{s}}$. It thus results that the left hand side term is still a weighted average of v and $\left\| \max_a \left\{ \Phi D Q_{\alpha_{-i}^{*P}, \mathbf{s}} v \right\} \right\| \leq \|v\|$. As $\left\| \max_a \left\{ \Phi D Q_{\alpha_{-i}^{*P}, \mathbf{s}} (\mathbf{V}^* - \mathbf{V}_{\Phi}^*) \right\} \right\| \leq \|\mathbf{V}^* - \mathbf{V}_{\Phi}^*\|$, this inequality delivers the last result.

We can now prove Proposition [6.1].

$$\begin{aligned} \|\mathbf{V}^* - \mathbf{V}_{\Phi}^*\| &= \|\Phi D (\mathbf{V}^* - \mathbf{V}_{\Phi}^*) + (I - \Phi D) \mathbf{V}^*\| \\ &\leq \|\Phi D (\mathbf{V}^* - \mathbf{V}_{\Phi}^*)\| + \|(I - \Phi D) \mathbf{V}^*\| \\ &\leq \frac{1}{1 - \rho} \|(I - \Phi D) \mathbf{V}^*\| \end{aligned}$$

Where the first step follows from the triangle inequality and the second step from Lemma [A.1]. The last step results from rearranging $\|\mathbf{V}^* - \mathbf{V}_{\Phi}^*\| - \rho \|\mathbf{V}^* - \mathbf{V}_{\Phi}^*\| \leq \|(I - \Phi D) \mathbf{V}^*\|$. ■

A.2.5 Proof of Theorem [6.2]

Proof. We can prove this by contradiction. We propose an aggregation that does not satisfy the condition, claim it to be optimal and show this cannot be true. Take three micro-states $\mathbf{F}_1 \geq \mathbf{F}_2 \geq \mathbf{F}_3$ and two macro-states, $P_1 \geq P_2$. By monotonicity as defined above and by transitivity of the partially ordered set, $V_F(s, \mathbf{F}_1) \geq V_F(s, \mathbf{F}_2) \geq V_F(s, \mathbf{F}_3)$. Select a non-order preserving aggregation, where $P_{1, F_1} = P_{1, F_3} \geq P_{2, F_2}$, where for simplicity, P_{n, F_m} means that micro-state F_m is aggregated in macro-state P_n . We now show that this cannot be optimal. The following four inequalities hold by monotonicity of the value function

$$\begin{aligned} \left| \frac{1}{2}(V_F(s, \mathbf{F}_1) - V_F(s, \mathbf{F}_3)) \right| &\geq \left| \frac{1}{2}(V_F(s, \mathbf{F}_2) - V_F(s, \mathbf{F}_3)) \right| \\ \left| \frac{1}{2}(V_F(s, \mathbf{F}_1) - V_F(s, \mathbf{F}_3)) \right| &\geq \left| \frac{1}{2}(V_F(s, \mathbf{F}_3) - V_F(s, \mathbf{F}_2)) \right| \\ \left| \frac{1}{2}(V_F(s, \mathbf{F}_3) - V_F(s, \mathbf{F}_1)) \right| &\geq \left| \frac{1}{2}(V_F(s, \mathbf{F}_2) - V_F(s, \mathbf{F}_3)) \right| \\ \left| \frac{1}{2}(V_F(s, \mathbf{F}_3) - V_F(s, \mathbf{F}_1)) \right| &\geq \left| \frac{1}{2}(V_F(s, \mathbf{F}_3) - V_F(s, \mathbf{F}_2)) \right| \end{aligned}$$

and by rearranging the elements

$$\begin{aligned} |V_F(s, \mathbf{F}_1) - \frac{1}{2}(V_F(s, \mathbf{F}_1) + V_F(s, \mathbf{F}_3))| &\geq |V_F(s, \mathbf{F}_2) - \frac{1}{2}(V_F(s, \mathbf{F}_2) + V_F(s, \mathbf{F}_3))| \\ |V_F(s, \mathbf{F}_1) - \frac{1}{2}(V_F(s, \mathbf{F}_1) + V_F(s, \mathbf{F}_3))| &\geq |V_F(s, \mathbf{F}_3) - \frac{1}{2}(V_F(s, \mathbf{F}_2) + V_F(s, \mathbf{F}_3))| \\ |V_F(s, \mathbf{F}_3) - \frac{1}{2}(V_F(s, \mathbf{F}_1) + V_F(s, \mathbf{F}_3))| &\geq |V_F(s, \mathbf{F}_2) - \frac{1}{2}(V_F(s, \mathbf{F}_2) + V_F(s, \mathbf{F}_3))| \\ |V_F(s, \mathbf{F}_3) - \frac{1}{2}(V_F(s, \mathbf{F}_1) + V_F(s, \mathbf{F}_3))| &\geq |V_F(s, \mathbf{F}_3) - \frac{1}{2}(V_F(s, \mathbf{F}_2) + V_F(s, \mathbf{F}_3))| \end{aligned}$$

Putting the four inequalities together gives the final result

$$\begin{aligned} &\max(|V_F(s, \mathbf{F}_1) - \frac{1}{2}(V_F(s, \mathbf{F}_1) + V_F(s, \mathbf{F}_3))|, |V_F(s, \mathbf{F}_3) - \frac{1}{2}(V_F(s, \mathbf{F}_1) + V_F(s, \mathbf{F}_3))|, 0) \\ &\geq \max(0, |V_F(s, \mathbf{F}_2) - \frac{1}{2}(V_F(s, \mathbf{F}_2) + V_F(s, \mathbf{F}_3))|, |V_F(s, \mathbf{F}_3) - \frac{1}{2}(V_F(s, \mathbf{F}_2) + V_F(s, \mathbf{F}_3))|) \end{aligned}$$

and the within-state variation from aggregating states \mathbf{F}_1 and \mathbf{F}_3 is larger or equal to the within-state variation from aggregating states \mathbf{F}_2 and \mathbf{F}_3 . Thus, the aggregation cannot be optimal. The same holds if we selected the alternative non-order preserving aggregation $P_{1,\mathbf{F}_2} \geq P_{2,\mathbf{F}_3} = P_{2,\mathbf{F}_1}$. ■

A.3 Multiple own states

As an example, let there be two own states $s1$ and $s2$ which can take two values $\{1, 2\}$. Imagine the hotels considered the dynamic effects of pricing on ratings. In this case the two states could be the rating and the price of each hotel. Define a set of macro nodes by defining elements of the bivariate distribution, $\mathbf{F}_{(s1-i, s2-i)}$. In this setting, with two rivals there are ten micro-states. An industry state distribution like $\mathbf{F} = \begin{bmatrix} 0 & 0 \\ 1 & 2 \end{bmatrix}$, has one rival in state (1,2) and one rival in state (2,2). If $\mathbf{F} = \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix}$, the two firms are in state (1,1). So \mathbf{F} is the cumulative count of firms at each bivariate state. The micro-states can be aggregated into a specified set of macro states. If we select four macro-states, $|\tilde{P}| = 4$, a possible aggregation is

$$\begin{aligned}
 P_1 & : \begin{bmatrix} 0 & 0 \\ 2 & 2 \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ 2 & 2 \end{bmatrix}, \begin{bmatrix} 0 & 0 \\ 1 & 2 \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ 1 & 2 \end{bmatrix}, \\
 P_2 & : \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix}, \begin{bmatrix} 1 & 2 \\ 1 & 2 \end{bmatrix}, \\
 P_3 & : \begin{bmatrix} 0 & 0 \\ 0 & 2 \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ 0 & 2 \end{bmatrix}, \text{ and} \\
 P_4 & : \begin{bmatrix} 0 & 2 \\ 0 & 2 \end{bmatrix}.
 \end{aligned}$$

The aggregation rule specifies: in the first line the industry state with one firm in state (1,1) and one firm in state (1,2) is aggregated with the industry states with one firm in state (1,2) and one firm in state (2,2) and with the industry states with one firm in state (1,2) and one firm in state (2,1) and they are all aggregated with the industry state with two firms in state (2,1). The last line specifies that industry state with two firms in state (1,2) is left as a single node. The aggregation assigns each of the 10 micro-states to the 4 macro-states. This is one of several possible aggregation schemes.

A.4 Simulation Results - Performance

We now compare performance in speed and accuracy across different specifications, concluding with the calculation of the bounds. **Absolute** speed figures are always dependent on the hardware, software and computational techniques used. In our case the results are obtained on a 3.2 Ghz 8-core desktop running Matlab. The code has been vectorized, and parallelized to use

the eight cores. Whenever possible we also use sparse matrix manipulation for the aggregation matrix. Given this, **absolute** numbers should vary with hardware, software and coding. We instead focus on the **relative** figures.

A.4.1 Speed

Regarding speed, Table [A1] reports an increase in computational time with both the number of players and the number of macro-nodes. The increase with the number of players is relatively small when compared to a similar increase in the number of quantiles. Solving a game with 75 firms and 5 quantiles takes about 460 seconds whereas using 9 quantiles takes about 13,895 seconds. The run times are further decomposed into load times (solving the static pricing game, constructing the aggregation and disaggregation matrices, and the profit vector), value function iterations (solving the Bellman equation conditional on a given set of beliefs for the industry transition), and the construction of the transition matrix. As reported, most time is spent in constructing the transition matrix (simulations) - a total of 264 seconds out of 460 seconds for the case with 75 firms and 5 quantiles. The time spent constructing the transition matrix changes little as we vary the number of firms. Instead as we increase the number of firms the load time (solving the static profit function, Π_s) increases substantially. We further subdivide the transition matrix time into the individual components, and find that the random draws are relatively negligible. The most time consuming procedure is to match the newly drawn industry states to the index where they belong to - "find" command in Matlab. Matlab's "find" command is a slow operation, taking about 93 seconds of the total 264 seconds spent constructing the transition matrix.

A.4.2 Accuracy

Regarding accuracy, Tables [A2] and [A3] report the exact approximation errors (supnorm) up to 15 firms for the value and investment functions, respectively. It becomes almost prohibitive, in terms of runtime, to solve the full model beyond 15 firms. Overall, the maximal errors are small, in particular for investment (under 6%). Average errors are under 1%. A more detailed analysis of Tables [A2] and [A3] gives us an understanding to how accuracy varies with the number of players and the number of quantiles. For a given equilibrium value function (fixed N), increasing R cannot make the approximation worse. On the other hand, for a fixed architecture size (fixed R) the game changes with changes to N and so does the equilibrium value function. It is unclear how the quality of the approximation is expected to change as we increase N for fixed R (N is a parameter of the game, R is not). The results in Table [A2] are consistent with a decrease in performance for fixed R as N increases³¹. However, this decrease in performance is stronger for small N than for large N , since the approximation improves as N increases for a fixed $N - R$. This is illustrated by reading the diagonals of Table [A2], over which the distance between exact and approximate $N - R$ remains fixed. From this we extrapolate that there should be some \bar{N} ,

³¹There are cases where N increases for a given R and the approximation improves. This could be due to simulation error.

Table A1: Computational time (in seconds) comparison across number of firms (N) and quantiles (R).

| N | Load | Val. Fun. | Construction of the transition matrix | | | | Total |
|-----|--------|-----------|---------------------------------------|-------|--------|----------|-----------|
| | | | Total | Draws | Sort | Match | |
| R=5 | | | | | | | |
| 10 | 0.28 | 3.67 | 144.11 | 0.70 | 1.02 | 104.68 | 157.83 |
| 15 | 0.61 | 3.45 | 142.97 | 0.58 | 1.49 | 100.37 | 147.49 |
| 20 | 1.04 | 3.79 | 147.35 | 0.59 | 2.10 | 98.72 | 152.48 |
| 25 | 1.96 | 4.15 | 160.11 | 0.60 | 2.59 | 98.04 | 166.69 |
| 50 | 35.50 | 4.49 | 250.97 | 0.60 | 7.92 | 93.84 | 291.28 |
| 75 | 189.73 | 4.30 | 264.83 | 0.56 | 18.59 | 93.15 | 459.31 |
| R=7 | | | | | | | |
| 10 | 0.40 | 30.27 | 1,062.97 | 1.47 | 2.69 | 946.20 | 1,096.18 |
| 15 | 0.79 | 28.14 | 1,033.90 | 1.35 | 3.98 | 915.79 | 1,064.84 |
| 20 | 1.22 | 30.78 | 1,020.12 | 1.54 | 5.49 | 887.41 | 1,054.45 |
| 25 | 2.62 | 25.40 | 1,066.64 | 1.39 | 6.72 | 890.67 | 1,097.57 |
| 50 | 44.18 | 33.83 | 1,821.42 | 1.35 | 20.40 | 824.34 | 1,902.03 |
| 75 | 228.81 | 33.29 | 1,902.58 | 1.37 | 50.32 | 843.52 | 2,168.87 |
| R=9 | | | | | | | |
| 10 | 0.57 | 252.03 | 5,576.22 | 3.00 | 5.81 | 5,319.27 | 5,844.67 |
| 15 | 0.95 | 248.31 | 5,638.43 | 3.02 | 9.18 | 5,144.18 | 5,915.06 |
| 20 | 1.63 | 245.68 | 5,521.16 | 3.07 | 12.43 | 5,004.03 | 5,789.87 |
| 25 | 3.47 | 236.92 | 5,569.65 | 2.97 | 15.07 | 4,904.89 | 5,838.77 |
| 50 | 59.83 | 251.13 | 7,763.94 | 2.88 | 42.34 | 4,342.80 | 8,098.42 |
| 75 | 302.06 | 253.02 | 13,312.74 | 2.96 | 132.80 | 4,795.89 | 13,895.16 |

Notes: This table reports how the run times are subdivided into the individual components. The first operation is load which consists of solving the static pricing game and building the profit function and the aggregation and disaggregation matrices. The second component is obtaining the optimal value function. The third operation is building the transition matrix (simulations). This last operation can be further subdivided into the draws, sorting of the industry vector and finding the index of industry state for each drawn state.

Table A2: Maximal relative error in the value function between the exact and approximated solutions, $\max(|V^* - V_{\Phi}^*|)/(|V^*| + |V_{\Phi}^*|)$.

| N | R | | | | | |
|----|------|------|------|------|------|------|
| | 4 | 5 | 6 | 7 | 8 | 9 |
| 6 | 3.30 | 0.00 | - | - | - | - |
| 7 | 3.87 | 2.53 | 0.00 | - | - | - |
| 8 | 4.18 | 3.01 | 2.09 | 0.00 | - | - |
| 9 | 4.70 | 3.25 | 2.91 | 1.76 | 0.00 | - |
| 10 | 4.43 | 3.80 | 2.83 | 2.09 | 1.55 | 0.00 |
| 11 | 5.33 | 3.95 | 3.28 | 2.31 | 1.97 | 1.35 |
| 12 | 5.98 | 3.72 | 3.44 | 2.66 | 2.04 | 1.61 |
| 13 | 5.84 | 4.52 | 3.68 | 3.11 | 2.44 | 1.80 |
| 14 | 6.77 | 4.92 | 3.58 | 3.13 | 2.52 | 2.05 |
| 15 | 6.43 | 5.07 | 3.97 | 3.12 | 2.63 | 2.29 |

Table A3: Maximal relative error in the investment function between the exact and approximated solutions, $\max(|a^* - a_{\Phi}^*|)/(|a^*| + |a_{\Phi}^*|)$.

| N | R | | | | | |
|----|------|------|------|------|------|------|
| | 4 | 5 | 6 | 7 | 8 | 9 |
| 6 | 1.10 | - | - | - | - | - |
| 7 | 1.45 | 0.79 | - | - | - | - |
| 8 | 2.25 | 1.33 | 0.85 | - | - | - |
| 9 | 2.50 | 2.00 | 1.25 | 0.71 | - | - |
| 10 | 1.67 | 1.78 | 1.41 | 1.29 | 0.66 | - |
| 11 | 2.10 | 1.84 | 1.58 | 1.60 | 1.03 | 0.58 |
| 12 | 2.24 | 1.62 | 1.47 | 1.57 | 1.29 | 0.96 |
| 13 | 2.32 | 1.75 | 2.14 | 1.92 | 1.39 | 1.18 |
| 14 | 2.88 | 2.15 | 1.74 | 1.49 | 1.33 | 1.28 |
| 15 | 2.86 | 2.20 | 1.67 | 1.76 | 1.24 | 1.10 |

after which performance does not deteriorate, and can even improve.

The supnorm error can be misleading about the overall accuracy of the approximation as it only considers the maximum error. Figure [A4] plots a scatter of the approximation errors of the value and investment functions. Each dot represents an individual state, \mathbf{s}_{-i} . Again the two solutions are very close, in particular for the optimal investment levels. The correlation between the approximate and exact solutions is 0.983 for $R = 5$. The figure also illustrates the within-state variation. For each approximated value (macro-state), there is a set of dots of true values (micro-states). Finally, Figure [A5] reports the steady state industry distribution. The approximated solution slightly underestimates the number of firms with high ratings and overestimates the number of firms with low ratings. This rationalizes the higher investment levels in the approximated solution. Firms expect less competition in the long run.

A.4.3 Bounds

Our previous results show that the error is a function of the approximation architecture, namely the within macro-state variance. Figure [A6] decomposes the error into its two components as specified by Equation [7]: the mean, and the variance. The mean error is relatively constant

Figure A4: Value and investment function for a firm i in state 1 ($x_i = 1$) and 14 competitors - scatter plot of exact vs approximate (5 and 7 quantiles) solutions. Log scale.

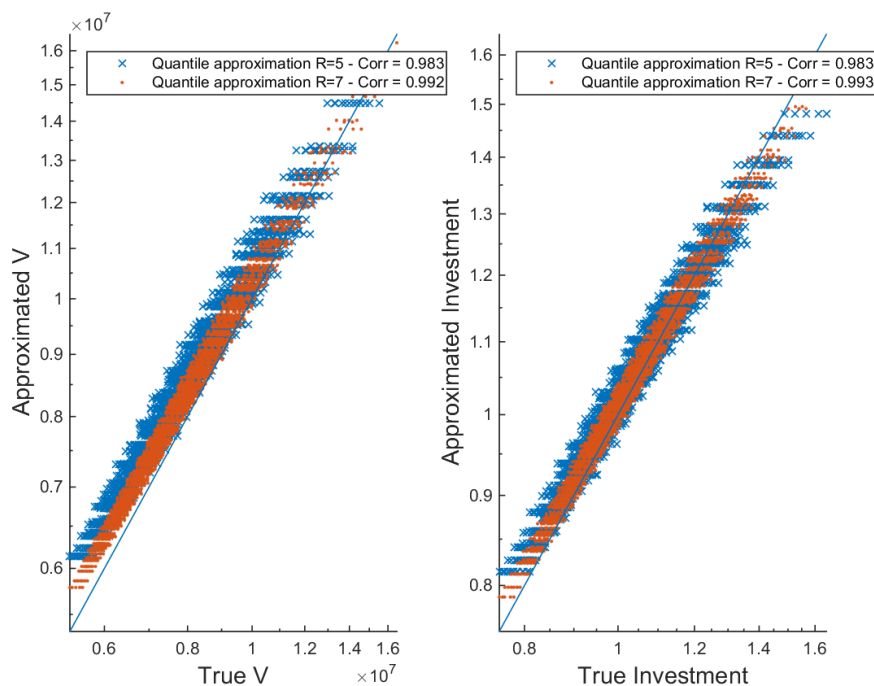
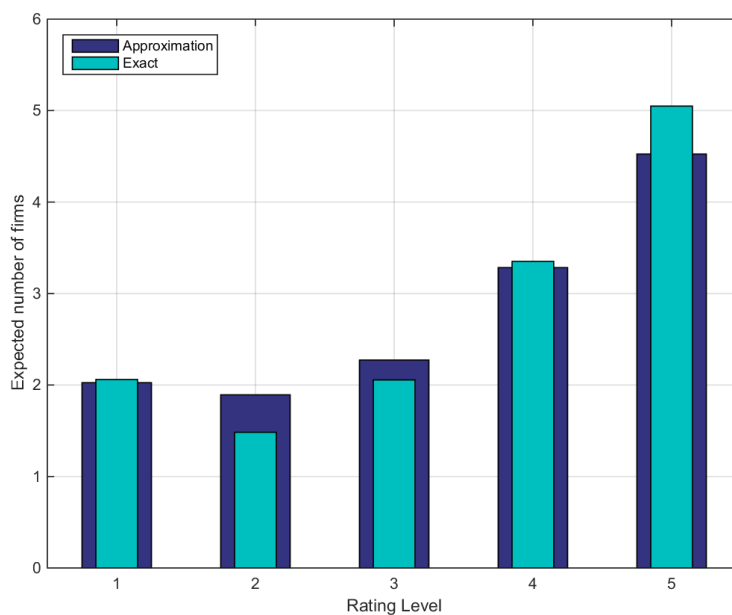


Figure A5: Long run steady state industry distribution - exact and approximated solutions ($N = 15, R = 5$).



across the different industry states, whereas the variance of the error is not. This illustrates our theoretical results derived in Proposition [6.1]. Figure [A7] reports how the maximal error and its two components (mean and variance) vary with the number of quantiles. The decrease in error computed in Table [A2] is mostly explained by the decrease in the variance component. This further corroborates the theoretical result, and illustrates how the variance component closely matches the exact error.

Figure A6: Equation [7] error decomposition. Results with $N=15$ and $R=5$.

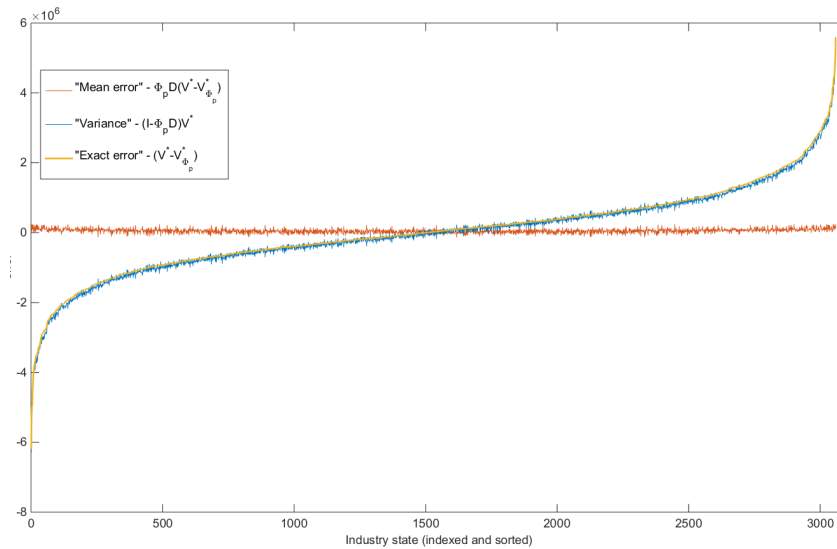
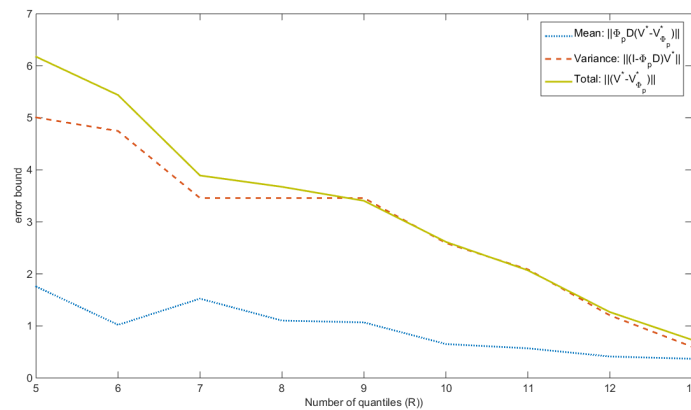


Figure A7: Maximal error (Equation [7] decomposition) with varying R ($N=15$).



Finally, Table [A4] provides a comparison across the different bounds: from Theorem [5.1], from Proposition [6.1] and the bounds in Tsitsiklis and Van Roy (1996) and Van Roy (2006). The bound obtained in Van Roy (2006) is only applicable when ρ is close to 1, under the invariant distribution. The bounds from Theorem [5.1] and Proposition [6.1] are characterization bounds that are only tight in particular cases and tend to be relatively loose when the discount factor is close to 1. First, both the error and the bound become tighter when the discount factor is smaller ($\rho = 0.33$). It seems that the bounds hold with a much smaller constant, suggesting that the error is a linear transformation of the bound (columns 2 and 8 of Table [A4] – about 40 to 50 times for $\rho = 0.98$ and about 1.5 to 1.54 times for $\rho = 0.33$). In terms of accuracy, the

Table A4: Approximation error and bounds with varying degrees of approximation quality (R) and discount factors (ρ).

| #R | Error $\ V^* - V_\Phi\ $ | Decomposition | | Theorem 5.1 | Bounds | | |
|---------------|-----------------------------|-----------------------------------|--|-------------|---------------------------|--------------------------------|---|
| | | Variance $\ (I - \Phi D)V^*\ $ | Mean Error $\ \Phi D(V^* - V_\Phi)\ $ | | Tsitsiklis and Van Roy | Van Roy valid at $\rho = 1$ | Proposition 6.1 $\frac{\ (I - \Phi D)V\ }{(1-\rho)}$ |
| $\rho = 0.98$ | | | | | | | |
| 5 | 6,173,260 | 5,007,036 | 1,756,804 | 62,252,532 | 46,724,000,000 | 467,240,000 | 250,351,792 |
| 6 | 5,437,216 | 4,745,716 | 1,020,760 | 42,669,684 | 44,895,000,000 | 448,950,000 | 237,285,792 |
| 7 | 3,890,852 | 3,457,815 | 1,524,884 | 34,121,032 | 32,895,000,000 | 328,950,000 | 172,890,752 |
| 8 | 3,673,504 | 3,457,815 | 1,102,696 | 32,136,510 | 32,895,000,000 | 328,950,000 | 172,890,752 |
| 9 | 3,405,060 | 3,457,815 | 1,067,864 | 31,970,706 | 32,895,000,000 | 328,950,000 | 172,890,752 |
| 10 | 2,614,796 | 2,593,415 | 651,808 | 27,573,418 | 24,615,000,000 | 246,150,000 | 129,670,752 |
| 11 | 2,069,256 | 2,086,781 | 569,755 | 24,013,400 | 19,611,000,000 | 196,110,000 | 104,339,048 |
| 12 | 1,266,990 | 1,203,552 | 412,736 | 16,500,475 | 10,984,000,000 | 109,840,000 | 60,177,576 |
| 13 | 734,212 | 602,183 | 368,736 | 6,768,932 | 5,575,000,000 | 55,750,000 | 30,109,150 |
| $\rho = 0.33$ | | | | | | | |
| 5 | 82,573 | 84,507 | 3,187 | 86,543 | 222,020 | 74,377 | 126,130 |
| 6 | 82,548 | 84,507 | 2,650 | 86,517 | 222,020 | 74,377 | 126,130 |
| 7 | 57,691 | 59,322 | 1,987 | 62,679 | 153,500 | 51,423 | 88,541 |
| 8 | 57,663 | 59,322 | 2,043 | 62,636 | 153,500 | 51,423 | 88,541 |
| 9 | 57,990 | 59,322 | 2,096 | 63,002 | 152,640 | 51,134 | 88,541 |
| 10 | 40,620 | 41,776 | 1,748 | 45,606 | 106,180 | 35,570 | 62,352 |
| 11 | 29,648 | 30,218 | 1,474 | 34,183 | 76,507 | 25,630 | 45,102 |
| 12 | 14,567 | 14,824 | 921 | 18,603 | 38,779 | 12,991 | 22,126 |
| 13 | 6,480 | 6,553 | 525 | 8,966 | 19,260 | 6,452 | 9,781 |

Notes: Error and bounds at the estimated parameter values with $N=15$, $\rho = 0.98$ (top panel) and $\rho = 0.33$ (bottom panel).

The bound in Tsitsiklis and Van Roy (1996) is $\frac{4\rho \min_{V_P} \|V^* - \Phi V_P\|}{(1-\rho)^2}$ while the bound in Van Roy (2006) is $\frac{2\rho \min_{V_P} \|V^* - \Phi V_P\|}{(1-\rho)}$

bound from Theorem [5.1] is much tighter than the bound from Proposition [6.1]. Comparing with the bounds derived in Tsitsiklis and Van Roy (1996) and Van Roy (2006), both of our bounds are also much tighter, in particular when compared to Tsitsiklis and Van Roy (1996). The bound obtained in Van Roy (2006) is smaller than the actual error when $\rho = 0.33$. This is because it is only applicable when ρ approaches 1. Finally, the bounds are relatively accurate when $\rho = 0.33$ (in particular the bound from Theorem [5.1]), which illustrates that the constant $\frac{1}{1-\rho}$ plays an important role in making the bounds one or more orders of magnitude larger than the true error.