A DYNAMIC MODEL OF MIXED DUOPOLISTIC COMPETITION:
OPEN SOURCE VS. PROPRIETARY INNOVATION

SUAT AKBULUT

BOĞAZİÇİ UNIVERSITY

2015

A DYNAMIC MODEL OF MIXED DUOPOLISTIC COMPETITION: OPEN SOURCE VS. PROPRIETARY INNOVATION

Thesis submitted to the
Institute for Graduate Studies in Social Sciences
in partial fulfillment of the requirements for the degree of

Master of Arts
in
Economics

by
Suat Akbulut

Boğaziçi University
2015

DECLARATION OF ORIGINALITY

I, Suat Akbulut, certify that

- I am the sole author of this thesis and that I have fully acknowledged and documented in my thesis all sources of ideas and words, including digital resources, which have been produced or published by another person or institution;
- this thesis contains no material that has been submitted or accepted for a degree or diploma in any other educational institution;
- this is a true copy of the thesis approved by my advisor and thesis committee at Boğaziçi University, including final revisions required by them.

Signature ....

Date ....

ABSTRACT

A Dynamic Model of Mixed Duopolistic Competition: Open Source vs. Proprietary Innovation

Open source software development has been an interesting investment model of production and innovation in recent years, especially in the last few decades. Unlike the private investment model, open source innovators freely share the software that they have developed at their private expense. Also, open source development is usually subject to certain licenses, one of which being the General Public License (GPL), the most popular open source license. In this thesis, we study the competition dynamics between a proprietary firm and an open source firm, Windows and Linux, for instance. We model the the competition between such two firms by incorporating the nature of the GPL, investment opportunities by the proprietary firm, user developers who can invest in the open source development, and a ladder type technology. We use a two period dynamic mixed duopoly model, in which a profit-maximizing proprietary firm competes with a rival, the open source firm, which prices the product at zero, with the quality levels determining their relative positions over time. We ask the following questions. How does the existence of open source firm affect the investment and the pricing behavior of the proprietary firm? Does the social welfare increase with the existence of the open source development? How are the users/consumers affected from the open source firm being available? We also discuss the limitations of our model and possible extensions.

ÖZET

Dinamik Bir Karma Düopol Modeli:
Açık Kaynağa Karşı Müseccel Yatırım

Açık kaynaklı yazılım son yıllarda, özellikle geçtiğimiz bir kaç on yılda, ilgi çekici bir üretim ve inovasyon modeli haline geldi. Özel yatırım modelinin aksine, açık kaynaklı yazılım geliştiricileri kendileri tarafından geliştirilen yazılımı ücretsiz olarak paylaşmaktalar. Bunun yanısıra açık kaynaklı yazılım geliştirmeleri bazı lisanslara bağlı olarak yapılmak durumunda ve bu lisanslardan biri ve en popüler olanı ise General Public License (GPL). Bu tezde Windows ve Linux örneğinde de olduğu gibi, müseccel firma ile açık kaynaklı firma arasındaki rekabet dinamiklerini çalışmaktayız. Bu türde iki firmanın arasındaki rekabeti, GPL lisansının özelliklerini, müseccel firmanın yatırım olanaklarını, açık kaynaklı yazılım geliştirmeye yatırım yapabilen kullanıcı-geliştiricileri ve merdiven tipi teknoloji kavramını da katarak modelliyoruz. Kar maksimizasyonu amacı güden firmanın ücret talep etmeyen açık kaynaklı yazılım firması ile rekabet ettiği ve kalite seviyelerinin zaman içersinde göreli pozisyonlarını etkilediği, iki periyotlu bir dinamik karma-düopol model kullanmaktayız. Bu bağlamda, şu soruları soruyoruz: Açık kaynaklı yazılım firmasının varlığı müseccel firmanın yatırım ve fiyatlama davranışını nasıl etkiler? Sosyal refah açık kaynaklı yazılım firmasının varlığıyla artar mı? Kullanıcılar/tüketiciler açık kaynaklı yazılım firmasının mevcudiyetinden nasıl etkilenirler? Ayrıca, modelimizin limitlerinden ve olası uzantılardan bahsetmekteyiz.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

A software is called open source, if its source code is open in the sense that anyone has free access to it. Open Source movement aims to bring programmers, who are not concerned with proprietary ownership or any financial gain, together to produce a more useful and bug-free product for everyone to use. By revealing its source code, an open source can be refined by many independent developers all around the world. The source code of an open source product is made available free of charge to the public. So, developers can read, redistribute and modify the source code, forcing an advantageous evolution of it.

Although there are many licenses used to distribute an open source projects, GNU General Public License (GPL) is the most commonly used one as of late 2014, by a share above 51%.[1] GPL has two main features. The first feature that GPL has is that although every user has the right to use and modify the code freely, the modifications must be distributed under the terms of the same license, if they are to be distributed at all. That is to say, GPL is a copylefted license. Although the rationale behind the open source movement is that a larger group of programmers who are not interested in the ownership produces a better, faster, more useful and, bug-free software, the second aspect of GPL allows the commercial exploitation of the program. Hence, the users have to sustain the free access to the source code, yet, as long as they maintain the free access, they are allowed to make profit. For example,

---

[1] See https://www.blackducksoftware.com/resources/data/top-20-open-source-licenses/

according to its 2015 income statement, Red Hat, the world's largest commercial distributor of the Linux operating system, made a total net income of $ 180.20 million in 2014.

Open source software (OSS), particularly Linux, has gained significant impact on software industry, thereby it has attracted noticeable interest of the researchers, as well. Having observed that the provision of OSS and its ongoing developments are costly, and moreover it is almost always publicly available at a price of zero, which does not reflect the economic costs that user developers incur to enhance it, economists have tried to understand the motives that might encourage the user developers to involve in such a costly activity.

This study tries to examine the effects of the existence of an open source firm that is competing with the proprietary firm on the proprietary firm's investment in innovation and production behavior, and how it affects the total welfare in the market. We set up a dynamic model with two periods, the first of which has two stages: competition and investment. In the second period, being the last period of the model, there is no investment stage. In the two competition stages, proprietary firm and open source firm compete in a mixed duopolistic industry, where the former charges a price to maximize its overall expected profit, whereas, the latter is freely available. At the beginning of each period, a new cohort of potential users enter into the model. At the beginning of the competition stage, they observe the quality levels and the price of proprietary firm's product, and they decide which operating system to use during their life time of one period. As long as they have some valuation for the open source, everyone will have an operating system, at least the free open source. In the next stage of the first period, the investment stage, while proprietary firm invests in probability to increase its products quality level, user developers' incentives for involving this costly development activity is to signal their abilities.

We find that under some circumstances, the proprietary firm supplies less and invests more in the presence of an open source rival, relative to the monopoly outcomes. This causes the proprietary firm to make less profit in the duopolistic industry compared to its monopoly. We show that, under some conditions, the total welfare generated under the monopoly case may be larger than the one generated under the duopolistic competition. However, when the conditions are not met it is possible that it might be better to have the duopolistic competition rather than a monopoly, when total welfare is concerned.

The success of open source software has led a literature on it, which has been flourishing since early 2000s. Lerner and Tirole (2002) introduce a broad sense discussion on economics of open source development. They indicate two reasons that might lead the developers to contribute to open source evolution. First reason that might make developers involve in this costly activity is that they receive a direct benefit in the form of improved software. Secondly, they get an indirect benefit by signaling their abilities in the job market. They also point out that the literature mostly considers individual incentives to adopt open source software.

Contributing to open source innovations brings the public good nature onto the surface. A considerable amount of literature focuses on open source development as public good in a static approach. Johnson (2002) uses public good approach in a static environment, where private provisions of user developers to a public good -the open source- diminishes as the number of user developers increases because of the free riding problem, and presents some comparative statistics and welfare results.

Modica (2012) takes a two period oligopoly game using a circular city approach in order to model the open source innovations from a public good perspective. Atal and Shankar (2014) introduce quality competition in order to model the open source development by treating the open source product as a public good. Llanes and Elejalde (2013), in a closely related study, analyze the participation decision to open source. In a two-stage game, they explore open source participation decision of n firms, the prices they choose to charge and their R&D investment decisions. They describe the conditions for coexistence of opens source firms and proprietary firms. In a similar study, Suh and Yılmaz (forthcoming) model the open source participation decision in a dynamic environment. They show that as long as a firm has the same or higher technology level as the open source, it does not join to the open source.

Some of the open source literature focuses on the competition between proprietary firm and open source firm. Casadesus-Masanell and Ghemawat (2006) study the competition between proprietary firm and open source firm in a dynamic mixed duopolostic industry with the demand side learning, and show that it is better to have the proprietary firm as a monopoly when the total welfare is considered. Casadesus-Masanell and Llanes (2011) use a mixed duopoly structure, where a for-profit proprietary firm competes with an open source firm, which tries to maximize the value of its open software. Our model differs from these studies in the way that it combines the open source innovation and the competition between a proprietary firm and an open source firm in a dynamic environment.

Economides and Katsamakas (2006) consider the variety of industry structures, and find that the profits are the highest for the vertically integrated proprietary industry structure compared to vertically disintegrated proprietary and open source firm with proprietary applications. Jaisingh, See-To and Tam (2014) study the response of a firm's choice of resource investments to improve quality and the firm's pricing decisions to the presence of an open source firm in its market through a duopolistic framework. Li and Ji (2010) study the welfare effects of price and quantity competition in the presence of technology licensing be in a duopoly industry setting where firms use R&D to reduce their cost. They show that Cournot competition yields lower prices, lower industry profit, higher consumer surplus as well as higher social welfare than Bertrand competition. Mustonen (2005) considers a firm's decision of supporting an existing open source program. It either chooses to support an existing open source program or not, and it the firm chooses to support, the programs become compatible and there are network effects.[2]

Yıldırım (2006) is one of the some other dynamic models studied. He explores the free-rider problem when contributing more is what induce the others to contribute.[3] Caulkins et al. (2013) study question of when does a proprietary firm release its source codes to be open source through a continuous time dynamic model where firms invest to increase their qualities and pick their own price for its software. Kort and Zaccour (2011) consider a similar problem through a three-stage duopoly game. They characterize the conditions for a firm relasing its source codes.[4]

Among other innovation related studies, Reisinger, Ressner, Schmidtke and Thomes (2014) construct a model in which firms produce a private good and invest in

---

[2] He characterizes the conditions under which supporting is optimal and also shows that a larger open source programmers' community does not mean an increase the welfare.

[3] For more on dynamic voluntary contribution games see, for instance, Admati and Perry (1991) and Marx and Matthews (2000).

[4] For more on duopolistic competition related studies, see also Haruvy, Sethi and Zhou (2008), Bitzer (2004) and Hasnas, Lambertini and Palestini (2014).

the quality of a public good, whish is considered as an open source project.[5] In another dynamic games of innovation, Erkal and Minehart (2014) explore the impact of knowledge sharing and incentives to license the intermediate steps in the production.[6]

Our paper also differs from these papers, in either modeling aspects or in the main question studied or the user developers being permitted to generate profit or not. We believe that it is important to characterize the behavior of a firm when there is open source subject to GPL, in terms of its investment in innovation and its pricing. Our model is both dynamic and tractable, and it is capable of capturing the essence of the GPL licensing.

---

[5] They find that, when investment cost function is assumed to be super additive, the presence of an additional firm o a government contribution to the public good increases the firms' contributions, that is, they get a crowd-in effect.

[6] See also Grossman and Shapiro (1987), Aoki (1991) and Fershtman and Malkovich (2010).

CHAPTER 2

A MODEL OF OPEN SOURCE VS. PROPRIETARY SOFTWARE

We study the competition between a proprietary firm and an open source firm. In order to model such a competition, where the competitors have heterogeneous objectives, we will make use of the literature on ”mixed duopolies”.

There are two firms, a proprietary firm and an open source firm, represented by Windows and Linux, respectively. Throughout this study, w and ℓ will stand for Windows and Linux, respectively.

There are two periods, the first of which has two stages: competition and investment. There is only one stage in the second period, which is competition stage. At the beginning of period t, for $t = 1, 2$, the quality level of an operating system (OS) $s \in \{w, \ell\}$, is denoted as $k_t^s \in \mathbb{Z}_+$. Although the initial quality levels k_1^w and k_1^ℓ will be given, their levels at the beginning of the second period, k_2^w and k_2^ℓ, are determined endogenously by the investment decisions of both Windows and Linux users, namely user developers. The evolution of quality levels follow a ladder type technology. For this reason, in the second stage of the first period, investment stage, Windows invests in probability in order to climb one step up on the technology ladder, whereas user developers involve in this costly development activity to signal their abilities to the job market. The realizations of the developments occur at the end of the period. Hence, if Windows invests $i_w \in [0, 1]$, its quality level at the beginning of the second period will be be the given by the following piece wise function:

$$k_2^w = \begin{cases} k_1^w + 1 & \text{with prob. } i_w \\ k_1^w & \text{with the remaining prob. } (1 - i_w) \end{cases}$$

During the same stage, those user developers, who decided to get a free copy of Linux in the previous stage, simultaneously with Windows' investment decision, decide whether and how much to invest in success probability to get the exogenous bonus $b \in [0, 1]$. Let i_j be the user developer j's investment level. Assuming the open source does not have the modular nature,[7] if at least one user developer succeeds in development stage, because of the terms of GPL, Linux will move up one step in the technology ladder. Hence, then Linux' quality level at the beginning of the second period will be:

$$k_2^\ell = \begin{cases} k_1^\ell + 1 & \text{if at least one user developer succeeds} \\ k_1^\ell & \text{if no user developers succeed} \end{cases}$$

However, investment is a costly activity for both Windows and user developers. In order to invest i, they must incur a cost of $\frac{1}{2}i^2$. Here we assume that users developers' skills are homogeneous. Yet, in Section 6.3, we introduce heterogeneity among user developers in their development skills.

---

[7] If a software has modular nature, then it is possible to break up the large projects, such as developing it, into small modules. Improving the modules independently will accomplish the project. However, it the software does not have the modular nature, then its development cannot be divided into pieces, and consequently, there may not be sufficient sophisticated user developers who can customize the software to their own needs. Section 6.3 tries to deal with the question when the open source software has the modular nature by introducing a contribution dimension to the model. You can see Lerner and Tirole (2002) for details about modules.

To specify the demand side, in each period, a new cohort of N potential users enter into the market. They observe the quality levels of both proprietary software and open source. Let k_t denote the quality differences between Windows and Linux, thus $k_t = k_t^w - k_t^\ell$. Let $\alpha_s(k_t) > 0$ denote the value of OS s given by the cohort entering at time t. $\alpha_s(k_t)$ is called as OS s' technological trajectory, which is a function of the quality level of OS s, k_t^s, and the quality level of the competing OS, k_t^{-s}. Even though the initial levels of these technological trajectories are exogenously given in the model, how they evolve while moving from period 1 to period 2 is endogenous. In the beginning of the first stage of period 1, and period 2, Windows charges a price of P_t, where $t \in \{1, 2\}$, to attract new customers. Assume that marginal cost of an extra copy of an operating system is zero. Let q_t be the number of users in period t, who buy Windows, then $N - q_t$ is the number of user developers in the same period since Linux is freely available and $\alpha_\ell(\cdot)$ is positive.

Assumption 1. *We assume linear demand function. For* $t = 1, 2$*, Windows' value for a user* $q_t \in \{1, 2, \ldots, N\}$ *in period* t *is:*

$$\alpha_w(k_t)\frac{N - q_t}{N} \tag{1}$$

and, similarly, let the value of Linux be:

$$\alpha_\ell(k_t)\frac{N - q_t}{N}$$

Figure 1 illustrates the Assumption 1. The demand is drawn as a straight line as if N was a real number, but the readers are well aware of that it must be a set of N points on that line instead.

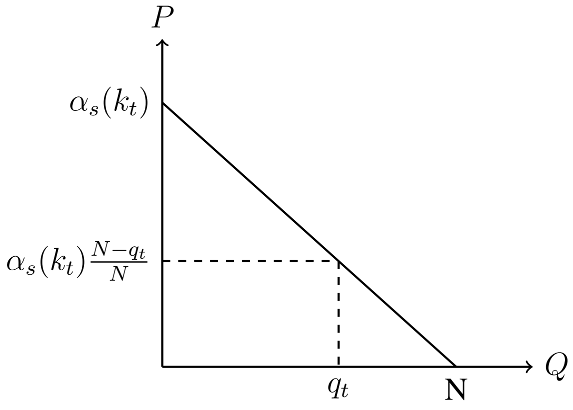

Figure 1. Demand of operating system s by cohort t.

Assumption 2. $\alpha_s(k_t) \geq 0$. *Since we assume OS s' value is never negative, technological trajectories are to be bounded below, which is required to have well defined demand functions.*

Assumption 3. $\alpha_w(k_t)$ *is increasing in* k_t*, whereas, as* k_t *increases,* $\alpha_\ell(k_t)$ *decreases. Holding the competent OS' quality level constant,* $\alpha_j(\cdot)$*, where* $j \neq i$*, the value of OS* i*,* $\alpha_i(\cdot)$*, will increase as its quality level increases.*

Let β be defined as $\beta(k_t) = \alpha_w(k_t) - \alpha_\ell(k_t)$, representing the difference between the trajectories of w and ℓ.

Assumption 4. $\beta(k_t)$ *is assumed to be non-negative, and to be concave in* k_t*.*

Assumption 4 is needed to be assumed to have a well defined maximization problem for Windows. It ensures that the difference between the technological trajectories of Windows and Linux would not explode. Otherwise, after some level of the quality difference, Windows would become a monopoly-like firm in the market. A negative value of β means the value of Linux is higher for all users. Taking it being freely available in the market, there would be no user buying Windows.

CHAPTER 3

MONOPOLY

In a market, where there is no substitute for Windows, and every user of any cohort has positive willingness to pay, inverse demand function is directly obtained by Equation 2. Figure 2 illustrates the timing of the events in a monopoly industry. To solve the equilibrium of monopolistic structure, we employ the Backward Induction methodology in this two-period game.

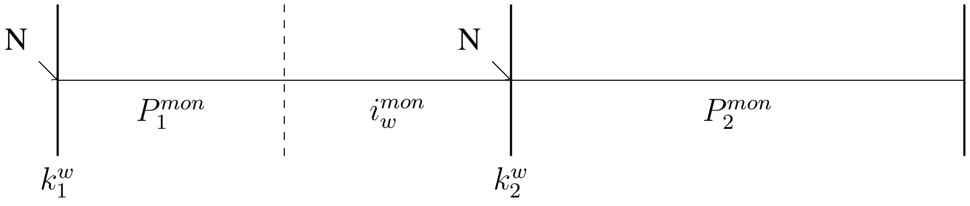

Figure 2. Timing of events in monopoly benchmark.

3.1 Second period

Since Windows is the only operating system producer and this is the last period of the game, having monopoly power, it produces the profit maximizing amount of $N/2$, and sets its price to $\alpha_w(k_2^w)/2$ in accordance with the demand structure. As a result, it generates a profit of:

$$\pi_2^{mon}(k_2^w) = \frac{N}{4}\alpha_w(k_2^w) \tag{2}$$

3.2 Second stage of the first period

How much investment is optimal for Windows? Investment affects Windows' profit through the quality level, which determines the valuations of the users, with trajectory function. An investment level i_w will increase its quality level by 1 with the probability i_w. Having known it will generate a profit of $\pi_2^{mon}(k_2^w)$ in the next period by Equation 2, the optimal monopoly investment strategy for Windows require it to choose an investment level, i_w^{mon} be in the following set:

$$i_w^{mon} \in \underset{i_w}{\text{argmax}} \left\{ i_w \frac{N}{4} \alpha_w(k_1^w + 1) + (1 - i_w) \frac{N}{4} \alpha_w(k_1^w) - \frac{1}{2} (i_w^{mon})^2 \right\}$$

Since the above term is linear in i_w, the monopoly investment level will be:

$$i_w^{mon} = \begin{cases} \frac{N}{4} \left[\alpha_w(k_1^w + 1) - \alpha_w(k_1^w) \right] & \text{if } \alpha_w(k_1^w + 1) - \alpha_w(k_1^w) \leq \frac{4}{N} \\ 1 & \text{otherwise} \end{cases} \tag{3}$$

3.3 First stage of the first period

At the beginning of the game, knowing its optimal strategies for the second phase and the next period, Windows chooses a price level, P_1^{mon} (or, equivalently, quantity level, q_1^{mon}) that maximizes its following overall expected profit:

$$\max_{q_1} \left\{ \alpha_w(k_1^w) \frac{(N - q_1)}{N} q_1 - \frac{1}{2} (i_w)^2 + i_w \frac{N}{4} \alpha_w(k_1^w + 1) + (1 - i_w) \frac{N}{4} \alpha_w(k_1^w) \right\}$$

When we take the first order derivative with respect to q_1, we obtain the following optimality condition since second order condition holds:

$$0 = \frac{\alpha_w(k_1^w)}{N}(N - 2q_1)$$

which implies $q_1^{mon} = \frac{N}{2}$ and $P_1^{mon} = \frac{\alpha_w(k_1^w)}{2}$.

Hence, Window chooses to sell to exactly one half of the number of consumers in each period, and the users should pay a price that is equal to the one half of the maximum value given to itself by their cohort.

CHAPTER 4

DUOPOLY

When we introduce Linux into the market, Windows no longer has its monopoly power. It has to consider the presence of Linux and the user developers' investment decisions while deciding how much to produce in each period, and to invest in the investment stage. The timing of events in duopoly industry is described in Figure 3.

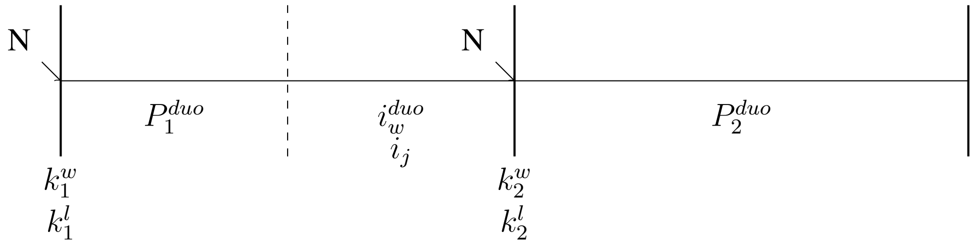

Figure 3. Timing of events in duopoly industry.

To solve the duopoly equilibrium, we use the same methodology, backward induction technique, as we utilize in the monopoly case. Since Linux can be downloaded freely and $\alpha_\ell(\cdot) > 0$, that is, all users are willing to pay a positive amount (even only a small amount) for a product that they can get without paying anything, it is guaranteed that every user will get one operating system, at least Linux. Hence, if q_t is the number of users who buy Windows, then the remaining users of cohort t, $N - q_t$, obtain the Linux at no price, and they become user developers.

4.1 Second period

When Windows is sold at price P_2^{duo} at period 2, the indifferent user between Windows and Linux, q_2, is found by the following equation:

$$\alpha_w(k_2)\frac{N - q_2^{duo}}{N} - P_2^{duo} = \alpha_\ell(k_2)\frac{N - q_2^{duo}}{N}$$

Remembering $\beta(k_t) = \alpha_w(k_t) - \alpha_\ell(k_t)$, the inverse demand function for Windows in period 2 would be the following:

$$P_2^{duo} = \beta(k_2)\frac{N - q_2^{duo}}{N} \tag{4}$$

As a rational profit maximizer agent, Windows produces q_2^{duo} such that:

$$q_2^{duo} \in \underset{q_2}{\text{argmax}} \left\{ \left(\beta(k_2)\frac{N - q_2}{N} \right) \cdot q_2 \right\}$$

Taking the first order derivative with respect to q_2, we get quantity and price levels for the second period as follows:

$$0 = \frac{\beta(k_2)}{N}\left(N - 2q_2\right)$$

implying $\quad q_2^{duo} = \frac{N}{2} \quad$ and $\quad P_2^{duo} = \frac{\beta(k_2)}{2}$. Thus, Windows will make a profit of

$$\pi_2^{duo} = P_2^{duo} \cdot q_2^{duo} = \frac{N}{4}\beta(k_2)$$

in the second period.

4.2 Second stage of the first period

In the investment stage, the actions of the user developers have impact on Windows' objectives. However, Windows' investment decision does not affect the user developers' investment strategies since they invest only for the purpose of getting the bonus, b. There would be some other version of the model, in which Windows' and user developers' decisions affect both their investment strategies. However, Chapter 6 will be discussing the issues that such models could create and how these issues could be handled. For now, we will analyze the investment decisions of user developers and Windows separately.

4.2.1 User developers' investment decisions

We have already mentioned that user developers are only interested in the bonus, b, when deciding whether and how much to invest in Linux' development. Hence, a user developer j, for $j \in \{q_1^{duo}, (q_1^{duo}+1), \ldots, N\}$, chooses an investment level i_j, which maximizes her expected net benefit. Hence, i_j solves the following maximization problem

$$\max_{i_j \in [0,1]} \left\{ i_j \cdot b - \frac{1}{2} {i_j}^2 \right\}$$

In order to find the optimal solution for the problem above, we take the first order derivative of it with respect to i_j. Hence we get:

$$b - i_j = 0 \quad \Rightarrow \quad i_j^* = b \qquad \text{for} \quad j = q_1^{duo}, (q_1^{duo}+1), \ldots, N$$

Since her expected net benefit, $\frac{b^2}{2}$, is positive, in the equilibrium, she will choose to invest $i_j^* = b$. Due to the symmetry, $i_j^* = b$ for all $j \in \{q_1 + 1, \ldots, N\}$. So, Linux will be developed with probability $1 - (1 - b)^{N - q_1^{duo}}$, which is the probability that at least one user developer succeeds.

4.2.2 Windows' investment decision

In contrast to investment decision of user developers, Windows takes into account what each user developer's strategy is. Thus, it chooses the investment level i_w^{duo} so that i_w^{duo} maximizes its expected future profit $\mathbb{E}\left[\pi_2^{duo}(k_2)\right]$.

$$\max_{i_w}\left\{\begin{array}{c} i_w\left((1-b)^{(N-q_1)}\cdot\beta(k_1+1)+\left(1-(1-b)^{(N-q_1)}\right)\cdot\beta(k_1)\right) \\ +(1-i_w)\left((1-b)^{(N-q_1)}\cdot\beta(k_1)+\left(1-(1-b)^{(N-q_1)}\right)\cdot\beta(k_1-1)\right)-\frac{1}{2}{i_w}^2 \end{array}\right\}$$

By taking the first order derivative with respect to i_w, we obtain

$$i_w^{duo} = \frac{N}{4}\left[(1-b)^{N-q_1^{duo}}\cdot\left(\beta(k_1+1)+\beta(k_1-1)-2\beta(k_1)\right)+\left(\beta(k_1)-\beta(k_1-1)\right)\right]$$

or in a more proper way:

$$i_w^{duo} = \min\left(\frac{N}{4}\left[(1-b)^{N-q_1^{duo}}\cdot C+\Delta\right], 1\right) \tag{5}$$

where $C = \beta(k_1+1)+\beta(k_1-1)-2\beta(k_1)$ and $\Delta = \beta(k_1)-\beta(k_1-1)$.

4.3 First stage of the first period

In the competition stage of the first period, in order for the user q_1 to be indifferent between Windows and Linux, her net benefit from buying Windows and downloading a free copy of Linux must be equal. In the equilibrium, choosing to get a free copy of Linux ensures a user to get an expected benefit of $\frac{b^2}{2}$ in the investment stage. So, when Windows is sold at price P_1^{duo} at period 1, the indifferent user between Windows and Linux, q_1, is found by the following equation:

$$\alpha_w(k_1)\frac{N-q_1}{N} - P_1 = \alpha_\ell(k_1)\frac{N-q_1}{N} + \frac{b^2}{2}$$

Hence, the inverse demand for Windows in period 1 would look like the following:

$$P_1^{duo} = \beta(k_1)\frac{N-q_1^{duo}}{N} - \frac{b^2}{2} \tag{6}$$

The optimal pricing/quantity strategy for Windows must be a solution of the following maximization problem, which simply is Windows' overall expected profit when it chooses to produce q_1.

$$\max_{q_1} \left\{ \begin{array}{rcl} P_1 \cdot q_1 - \frac{1}{2}\left(i_w\right)^2 & + & i_w \cdot (1-b)^{(N-q_1)} \cdot \frac{N}{4} \cdot \beta(k_1+1) \\ & + & i_w \cdot \left(1-(1-b)^{(N-q_1)}\right) \cdot \frac{N}{4} \cdot \beta(k_1) \\ & + & (1-i_w) \cdot (1-b)^{(N-q_1)} \cdot \frac{N}{4} \cdot \beta(k_1) \\ & + & (1-i_w) \cdot \left(1-(1-b)^{(N-q_1)}\right) \cdot \frac{N}{4} \cdot \beta(k_1-1) \end{array} \right\}$$

The first order condition of above maximization problem is

$$-\beta(k_1)\tfrac{2q_1^{duo}}{N} + \beta(k_1) - \tfrac{b^2}{2} - \tfrac{N^2}{16}C^2\left((1-b)^{N-q_1^{duo}}\right)^2 \ln(1-b) - \tfrac{N}{4}\Delta(1-b)^{N-q_1^{duo}} \ln(1-b)\left(1+\tfrac{N}{4}C\right) = 0$$

Although this condition does not have a closed form analytical solution in q_1^{duo}, we are capable of comparing it with the first period quantity in the monopoly case,

q_1^{mon} since we know that the overall expected profit function is concave in q_1, and maximized at q_1^{duo}. When we evaluate the above first order condition at $q_1^{mon} = N/2$, we obtain the following function:

$$f(b) = -\frac{b^2}{2} - \frac{N^2C^2}{16}(1-b)^N \ln(1-b) - \frac{N}{4}\Delta(1-b)^{N/2}\ln(1-b)\left(1+\frac{N}{4}C\right)$$

Proposition 1. *For large enough bonus b, proprietary firm produces less in the first period of the duopolistic competition as opposed to the case in which it is a monopoly.*

Proof. $f(b)$ *is continuous in* $[0,1)$ *with* $f(0) = 0$ *and* $\lim_{b\to 1} f(b) < 0$. *Therefore, there exists a* $\hat{b} \in [0,1)$ *such that* $f(b) < 0$, *for all* $b \in [\hat{b}, 1)$. *Because the first order condition is negative at point* $q_1 = N/2$ *for large* b*'s and, the overall expected profit function is concave in* q_1, *we get:*

$$q_1^{duo} < q_1^{mon} = \frac{N}{2} \tag{7}$$

Proposition 1 shows that the existence of an open source rival affects the monopoly production as existence of any other rivalry for-profit firm, in the sense that the monopoly firm decreases its production level.

Proposition 2. *Proprietary firm makes more investment in the duopoly industry competition as opposed to the case where it is a monopoly.*

Proof. *Let* $(1-b)^{(N-q_1)} = x$. *Observe that* $x \in (0,1)$. *Since* $\beta(\cdot)$ *is concave,* $C = \beta(k_1+1) + \beta(k_1-1) - 2\beta(k_1)$ *is negative, and* $\Delta = \beta(k_1) - \beta(k_1-1)$ *is positive. Hence,*

$$\begin{aligned} & \frac{N}{4} \cdot (x-1) \cdot (\beta(k_1+1) + \beta(k_1-1) - 2\beta(k_1)) \ > \ 0 \\ \Rightarrow \quad & \frac{N}{4} \left(x \cdot C + \Delta - (\beta(k_1+1) - \beta(k_1))\ \right) \ > \ 0 \\ \Rightarrow \quad & \frac{N}{4} \left(x \cdot C + \Delta\right) \ > \ \frac{N}{4} \left(\beta(k_1+1) - \beta(k_1)\right) \\ \Rightarrow \quad & i_w^{duo} \ > \ i_w^{mon} \end{aligned}$$

As Proposition 2 suggests, competition results in Windows increasing its investment level. However, this result is not special to having an open source rival.

CHAPTER 5

WELFARE COMPARISON

Proposition 1 & 2 concludes that the proprietary firm makes less profit in the duopoly industry, which suggests that a duopoly is likely to dominate proprietary firm's monopoly in terms of total welfare generation. In this section, we analyze the welfare implications of the two industry structure that we studied above. Instead of finding the absolute level of total welfare in the duopoly industry, we will compare the total welfare levels under the assumptions that $\alpha_w(\cdot)$ and $\alpha_\ell(\cdot)$ are linear with slope γ_w and γ_ℓ, respectively. $\alpha_w(\cdot)$ and $\alpha_\ell(\cdot)$ being linear with slope γ_w and γ_ℓ causes $\beta(\cdot)$ to be a linear function, as well, with slope $\gamma_w - \gamma_\ell$, that is, $C = 0$ and $\Delta = \gamma_w - \gamma_\ell$. Assumption 3 ensures that $\gamma_w > 0$ and $\gamma_\ell < 0$. Thereby, Δ is positive.

Proposition 3. *If* $\frac{N}{4}\left(|\gamma_\ell| + 2\gamma_w\right) < 1$ *and* $\frac{N}{4}\gamma_w > (1-b)^{N/2}$*, then total welfare is higher in proprietary firm's monopoly than the total welfare in duopoly industry.*

Proof. *To prove Proposition 3, we divide the total welfare into pieces and compare them piece-wise instead of measuring them as wholes. And, when comparing the two welfare levels, we interpret the absence of Linux in the monopoly industry as* k_t^ℓ *and* $\alpha_\ell(k_t)$ *being zero. Therefore,* $\beta(k_t) = \alpha_w(k_t)$*.*

Expected total welfare in the Windows' monopoly, $\mathbb{E}\left[W^m\right]$ is:

$$\mathbb{E}\left[W^m\right] = \sum_{j=1}^{N/2}\left(\alpha_w(k_1)\frac{N-j}{N}\right) - \frac{(i_w^{mon})^2}{2} + i_w^{mon}\sum_{j=1}^{N/2}\left(\alpha_w(k_1+1)\frac{N-j}{N}\right) + (1-i_w^{mon})\sum_{j=1}^{N/2}\left(\alpha_w(k_1)\frac{N-j}{N}\right)$$

$$= \underbrace{\overbrace{\alpha_w(k_1)\left(\frac{3N-2}{8}\right)}^{a^{mon}} \overbrace{-\frac{(i_w^{mon})^2}{2}}^{c^{mon}}}_{\text{first period welfare}} + \underbrace{\overbrace{i_w^{mon}\alpha_w(k_1+1)\left(\frac{3N-2}{8}\right) + (1-i_w^{mon})\,\alpha_w(k_1)\left(\frac{3N-2}{8}\right)}^{d^{mon}}}_{\text{second period welfare}}$$

Expected welfare in the first period of the duopoly industry, $\mathbb{E}\left[W^d\right]$:

$$\mathbb{E}\left[W_1^d\right] = \sum_{j=1}^{q_1}\left(\alpha_w(k_1)\frac{N-j}{N}\right) + \sum_{j=q_1+1}^{N}\left(\alpha_\ell(k_1)\frac{N-j}{N}\right) - \frac{\left(i_w^{duo}\right)^2}{2}$$

$$= \overbrace{\beta(k_1)\left(q_1 - \frac{1}{N}\frac{q_1(q_1+1)}{2}\right)}^{a^{duo}} + \overbrace{\alpha_\ell(k_1)\frac{N-1}{2}}^{e} \overbrace{-\frac{\left(i_w^{duo}\right)^2}{2}}^{c^{duo}}$$

And the expected total welfare generated in the second period of the duopoly will be:

$$\mathbb{E}\left[W_2^d\right] = \begin{array}{lcll} i_w\cdot(1-b)^{(N-q_1)} & \cdot & \left(\beta(k_1+1)\left(\frac{3N-2}{8}\right)\right. & \left.-\alpha_\ell(k_1+1)\frac{N+1}{2}\right) \\ +i_w\cdot\left(1-(1-b)^{(N-q_1)}\right) & \cdot & \left(\beta(k_1)\left(\frac{3N-2}{8}\right)\right. & \left.-\alpha_\ell(k_1)\frac{N+1}{2}\right) \\ +(1-i_w)\cdot(1-b)^{(N-q_1)} & \cdot & \left(\beta(k_1)\left(\frac{3N-2}{8}\right)\right. & \left.-\alpha_\ell(k_1)\frac{N+1}{2}\right) \\ +(1-i_w)\cdot\left(1-(1-b)^{(N-q_1)}\right) & \cdot & \left(\beta(k_1-1)\left(\frac{3N-2}{8}\right)\right. & \left.-\alpha_\ell(k_1-1)\frac{N+1}{2}\right) \end{array}$$

Hence,

$$\mathbb{E}\left[W_2^d\right] = \overbrace{\frac{3N-2}{8}\left[\left(i_w^{duo}+(1-b)^{(N-q_1)}\right)(\gamma_w-\gamma_\ell)+\beta(k_1-1)\right]}^{d^{duo}}$$
$$\underbrace{-\frac{N+1}{2}\left[\left(i_w^{duo}+(1-b)^{(N-q_1)}\right)\gamma_\ell+\alpha_\ell(k_1-1)\right]}_{f}$$

Now let us start to compare the pieces marked by lower case letters. For $q_1 < N$*, we have:*

$$\frac{d\left(q_1 - \frac{1}{N}\frac{q_1(q_1+1)}{2}\right)}{dq_1} > 0$$

Thus,

$$\beta(k_1)\left(q_1^{duo} - \frac{1}{N}\frac{q_1^{duo}(q_1^{duo}+1)}{2}\right) < \beta(k_1)\left(\frac{N}{2} - \frac{1}{N}\frac{\frac{N}{2}(\frac{N}{2}+1)}{2}\right) = \beta(k_1)\frac{3N-2}{8}$$

which implies

$$a^{duo} < a^{mon} \tag{8}$$

As a consequence of Proposition 2, we have $i_w^{duo} > i_w^{mon}$ *which implies* $-\frac{1}{2}\left(i_w^{duo}\right)^2 < -\frac{1}{2}\left(i_w^{mon}\right)^2$*, which in turn implies*

$$c^{duo} < c^{mon} \tag{9}$$

We know that $i_w^{mon} > (1-b)^{(N-q_1)}$ *and* i_w^{duo} *cannot be more than 1. Therefore:*

$$i_w^{duo} + (1-b)^{(N-q_1)} - 1 < i_w^{mon}$$

$$\left(i_w^{duo} + (1-b)^{(N-q_1)} - 1\right)\left(\beta(k_1) - \beta(k_1-1)\right) < i_w^{mon}\left(\beta(k_1) - \beta(k_1-1)\right)$$

$$\frac{3N-2}{8}\left[\left(i_w^{duo} + (1-b)^{(N-q_1)}\right)(\gamma_w - \gamma_\ell) + \beta(k_1-1)\right] < \frac{3N-2}{8}\left[i_w^{mon}\left(\beta(k_1) - \beta(k_1-1)\right) + \beta(k_1)\right]$$

which implies

$$d^{duo} < d^{mon} \tag{10}$$

Combining e *and* f *we get:*

$$\begin{aligned} e + f &< \frac{N-1}{2}\left[\alpha_\ell(k_1) - \left(i_w^{duo} + (1-b)^{(N-q_1)}\right)\gamma_\ell - \alpha_\ell(k_1-1)\right] \\ &= \frac{N-1}{2}\left(1 - i_w^{duo} - (1-b)^{(N-q_1)}\right)\gamma_\ell \end{aligned}$$

Note that $\left(1 - i_w^{duo} - (1-b)^{(N-q_1)}\right)$ *is positive due to the assumptions* $\frac{N}{4}\left(|\gamma_\ell| + 2\gamma_w\right) < 1$ *and* $\frac{N}{4}\gamma_w > (1-b)^{N/2}$*. To see this, note*

$$\begin{aligned} &\frac{N}{4}\left(|\gamma_\ell| + 2\gamma_w\right) < 1 \\ &\Rightarrow \quad \frac{N}{4}|\gamma_\ell| + \frac{N}{4}\gamma_w + \frac{N}{4}\gamma_w < 1 \\ &\Rightarrow \quad \frac{N}{4}\gamma_\beta + \frac{N}{4}\gamma_w < 1 \\ &\Rightarrow \quad \frac{N}{4}\gamma_\beta + (1-b)^{N/2} < 1 \\ &\Rightarrow \quad \frac{N}{4}\gamma_\beta + (1-b)^{N-q_1^{duo}} \end{aligned}$$

Thus, the summations of the terms including α_ℓ *in the duopoly welfare is negative, that is,*

$$e + f < 0 \tag{11}$$

Combining Equations (8), (9), (10) & (11), we conclude that the total welfare that the monopoly proprietary generates is higher than the total welfare in the duopoly industry.

Proposition 3 shows that the competition does not necessarily increase the welfare in an oligopoly industry when compared to the monopoly market. This is because the presence of a rival induces the proprietary firm to set lower prices and those users who do not buy the proprietary firm's product are not left empty handed; they can get the open source freely, which increases the total surplus. However, the decrease in proprietary firm's and its users' surpluses do not, always, need to be compensated by the increase in user developers' surpluses.

CHAPTER 6

DISCUSSION

This chapter is about the alternative models that could have been used to capture the effects of an open source firm's presence on the behavior of a proprietary firm. We also tried to examine changes in our results when the question is modeled in different ways, and summarized the reasons behind the fact that why we end up with not using them. As our future work, we will improve the last alternative in order to increase the period number to finitely or even infinitely many because it might be useful to have infinitely many periods in order to study long-run behavior of the two firms and to question the lifespan of proprietary firm, whose faith might be releasing its source codes and becoming an open source, as well.

6.1 $T \geq 3$ periods

When we tried to set up a model, where the number of periods is three or more, or infinitely many, we end up with technical problems of solving the first order condition of proprietary firm's maximization problem. This problem occurs because there is no analytical solution, for sure, to the number of proprietary users at period t, q_t, when the number of potential users, N, exceeds three. Employing the known methods to solve the Bellman Equation that captures the recursive nature of the dynamic game problem is not helpful since transition matrix that should govern the evolution of the state variables are determined by the choice variables in each period, i.e the transition matrix is not stationary.

6.2 Endogenous bonus with OLG

One other possibility could be utilize the first investment incentive for the user developers that Lerner and Tirole (2002) mention, i.e user developers involve in the development activity because they receive a direct benefit in the form of improved software. We set up a model, where users lived two periods. They could buy an operating system only when they are young. User developers could develop the open source when they are young, and enjoy the appreciation of its quality when they are old, if at least one of them succeeds due to General Public License. When we model the user developers investment incentives in this framework, with allowing the investment levels to be in $[0, 1]$ interval, we faced difficulties while solving the optimal investment levels of user developers since the optimal decisions include N^{th} order equations. To overcome such difficulties, one could think of forcing the possible investment level choices of the user developers to be binary, i.e they would be either 0 or 1. However, there occurs a free rider problem that Johnson (2002) finds, too. Since it is guaranteed for the open source to be improved when one user developer chooses to invest in 1, it is optimal for every user developer to let someone else do it.

6.3 Contribution game with infinitely many users

When Lerner and Tirole (2002) explain the favorable characteristics for an open source production, they mention about its modularity, whether the overall project is divided into smaller and well-defined tasks (modules) that individuals can handle independently from other modules. Sufficiently modular nature of an open source software, whose different portions can be improved by independent user developers, might turn the investment stage to a contribution game for open source user developers. To do so, one other helpful way could be having infinitely many users distributed on $[0, 1]$. Although in our original model, that would create some compatibility problems while finding the open source firm's development probability, since it has a multiplication part, which is not a good way to use when there are infinitely many users, that would provide a well defined demand, and is a better way to model the investment stage as a contribution game, where the probability of open source firms' development is affected by a fraction of the measure of user developers that contribute or all users. Such a model might also capture the direct benefit incentives of the user developers, which would result in having different optimal investment strategies for different user developers. To incorporate the direct benefit, a successful development of a user developer could be rewarded by enjoying the appreciation of her own operating system before the quality increase become public.

CHAPTER 7

CONCLUSION

It is impressive that a costly investment based upon not having the property rights has produced such a useful and reliable software. In this study, a simple two-period model of open source innovation has been presented to understand the difference of the behavior of the proprietary firm's production, pricing and investment strategies and to facilitate welfare comparisons between the presence of it and the traditional, profit driven method of development, where the quality levels of the two follow a ladder type technology framework.

It has been shown that the proprietary firm decreases its production level when there is an open source rival, and in order to better compete with the open source firm, it invests more. However, that the proprietary firm loosing some of its profit cannot be concluded as a duopoly is likely to dominate the proprietary firm's monopoly in terms of total welfare generation because it has been shown that for some levels of the linear formed technological trajectory functions' slopes, the total welfare is higher in proprietary firm's monopoly than the total welfare in duopoly industry.

REFERENCES

Admati, A. R., & Perry, M. (1991). Joint projects without commitment. *The Review of Economic Studies*, *58*, 259-276.

Aoki, R. (1991). R&D competition for product innovation: An endless race. *The American Economic Review*, *81*, 252-256.

Atal, V., & Shankar, K. (2014). Open source software: Competition with a public good. *Atlantic Economic Journal*, *42*(3), 333-345.

Bitzer, J. (2004). Commercial versus open source software: The role of product heterogeneity in competition. *Economic Systems*, *28*, 369-381.

Casadesus-Masanell, R., & Ghemawat, P. (2006). Dynamic mixed duopoly: A model motivated by Linux vs. Windows. *Management Science*, *52*(7), 1072-1084.

Casadesus-Masanell, R., & Llanes, G. (2011). Mixed source. *Management Science*, *57*(7), 1212-1230.

Caulkins, J. P., Feichtinger, G., Grass, D., Hartl, R. F., Kort, P. M., & Seidl, A. (2013). When to make proprietary software open source. *Journal of Economic Dynamics & Control*, *37*, 1182-1194.

Economides, N., & Katsamakas, E. (2006). Two-sided competition of proprietary vs. open source technology platforms and the implications for the software industry. *Management Science*, *52*(7), 1057-1071.

Erkal, N., & Minehart, D. (2014). Optimal technology sharing strategies in dynamic games of R&D. *Journal of Economics & Management Strategy*, *23*(1), 149-177.

Fershtman, C., & Markovich, S. (2010). Patents, imitation and licensing in an asymmetric dynamic R&D race. *International Journal of Industrial Organization*, *28*(2), 113-126.

Grossman, G. M., & Shapiro, C. (1987). Dynamic R&D competition. *The Economic Journal*, *97*, 372-387.

Haruvy, E., Sethi, S. P., & Zhou, J. (2008). Open source development with a commercial complementary product or service. *Production and Operations Management*, *17*(1), 29-43.

Hasnas, I., Lambertini, L., & Palestini, A. (2014). Open innovation in a dynamic cournot duopoly. *Economic Modelling*, *36*, 79-87.

Jaisingh, J, See-To, E. W. K., & Tam K. Y. (2008). The impact of open source software on the strategic choices of firms developing proprietary software. *Journal of Management Information Systems*, *25*(3), 241-276.

Johnson, J. P. (2002). Open source software: Private provision of a public good. *Journal of Economics & Management Strategy*, *11*(4), 637-662.

Kort, P. M., & Zaccour, G. (2011). When should a firm open its source code: A strategic analysis. *Production and Operations Management*, *20*(6), 877-888.

Lerner, J., & Tirole, J. (2002). Some simple economics of open source. *Journal of Industrial Economics*, *50*(2), 197-234.

Li, C., & Ji, C. (2010). Innovation, licensing, and price vs. quantity competition. *Economic Modelling*, *27*, 746-754.

Llanes, G., & de Elejalde, R. (2013). Industry equilibrium with open-source and proprietary firms. *International Journal of Industrial Organization*, *31*, 36-49.

Marx, L. E., & Matthews, S. A. (2000). Dynamic voluntary contribution to a public project. *The Review of Economic Studies*, *67*, 327-358.

Modica, S. (2012). Open source without free-riding. *Economia Politica*, *2*, 247-260.

Mustonen, M. (2005). When does a firm support substitute open source programming?. *Journal of Economics & Management Strategy*, *14*(1), 121-139.

Reisinger, M., Ressner, L., Schmidtke, R., & Thomes, T. P. (2014). Crowding-in of complementary contributions to public goods: Firm investment into open source software. *Journal of Economic Behavior & Organization*, *106*, 78-94.

Suh, J., & Yılmaz, M. (2015). *Economics of open source: A dynamic approach*. Manuscript in preparation.

Yıldırım, H. (2006). Getting the ball rolling: Voluntary contributions to a large-scale public project. *Journal of Public Economic Theory*, *8*(4), 503-528.