Research Article

Chung-Hui Chou*

Software Cracking and Degrees of Software Protection

https://doi.org/10.1515/bejte-2021-0029
Received March 2, 2021; accepted May 23, 2022

Abstract: Progress of hardware technologies and diffusion of computer knowledge enable consumers to crack software if they decide to use software illegally. This paper constructs a software market in which consumers are horizontally differentiated in accordance with social norms of copyright protection to examine the monopolistic producer's software protection behaviors with considering partial compatibility between genuine and cracked software as well as utility loss from using cracked software. Our research presents the following results. First, when network externalities are weak, the monopolist would set a degree of protection which induces existence of software cracking to enhance consumers' willingness to pay for genuine software by improving network benefits. Conversely, if network externalities are sufficiently strong, then software producer would set a degree of protection which stop software cracking completely. This implies that stopping software cracking is not possible without network effects. Second, if utility loss from using cracked software is severe (mild), then strengthening (weakening) network externalities or lowering (raising) compatibility may reduce the number of consumers using cracked software and increase software producer's profits consequently. Finally, we show that the monopolistic producer tends to over-protect software when genuine and cracked software are highly compatible or network externalities are relatively weak which results in inadequate consumers using cracked software for social optimum.

Keywords: network externalities, partial compatibility, software cracking

JEL Classification: L11, L86, O34

***Corresponding author: Chung-Hui Chou**, Department of Finance, I-Shou University, Kaohsiung City 84001, Taiwan, ROC, E-mail: chchou@isu.edu.tw. https://orcid.org/0000-0002-4126-9263

1 Introduction

Software piracy can be observed easily in computer industries which causes enormous damage in software producers' profits. In the past decades, economists have taken various approaches to analyze software manufacturers' copyright protection behaviors and the impacts of piracy on profits. Novos and Waldman (1984), Johnson (1985), Liebowitz (1985) and Bensen and Kirby (1989) studied copyright protection in a market without network externalities which are more applicable to books or journals industries. Conner and Rulmet (1991), Takayama (1994) and Slive and Bernhardt (1998) showed that permission of software piracy may enhance monopolistic firm's profits under significant network externalities. Shy and Thisse (1999) considered a duopoly market in which firms produce horizontally differentiated software and presented that software without protection may be an equilibrium outcome under significant network externalities. Rasch and Wenzel (2013) studied the impacts of software piracy on pricing in a two-sided market. In literature, there were some papers considering software protection as a continuous degree rather than a dichotomous decision. Yoon (2002) assumed that degrees of software protection affect consumers' reproduction costs in a market without network externalities, and showed that raising copyright protection may enhance social welfare. Jain (2008) assumed that degrees of software protection determine probabilities of consumers using pirated software without incurring copy costs, and found that duopolistic software manufacturers may permit software piracy under weak network externalities. Inceoglu (2015) considered that degrees of software protection determine substitution between genuine and pirated software, and discovered that software piracy may benefit an incumbent by making entry less profitable. It is worth mentioning that in the existing literature, pirated software is generally assumed to be fully compatible with genuine one.

Progress of hardware technologies and diffusion of computer skills make drastic changes in software piracy. Nowadays consumers who want to use software illegally would crack genuine software rather than purchase pirated one. From this viewpoint, degrees of software protection determine the amount of effort required for cracking software successfully. It is reasonable to infer that consumers would suffer more disutility to crack more-protected software successfully. Besides, consumers usually feel that the malfunction risk of cracked software is higher than that of genuine one. This motivates us to consider utility loss from using cracked software. In other words, consumers gain less intrinsic utility from cracked software than they do from genuine one.

The extents of accordance with social norms of software protection also affect consumers' software adoption behaviors significantly. This induces us to consider a market in which consumers are horizontally differentiated in accordance with social norms of copyright protection. Consumers with more (less) accordance with social norms of copyright protection prefer genuine (cracked) software subjectively.

Software markets usually have some common features. First, consumers gain more utility from specific software when the number of consumers using compatible products increase. Second, consumers using cracked software usually cannot communicate with all of the consumers using genuine software, and vice versa. If genuine and cracked software become more compatible, then consumers are more-motivated to use cracked software which induces software producer to increase protection to strive for its market share. On the other hand, an increase in compatibility between genuine and cracked software enables consumers buying genuine software to enjoy more network benefits from those using cracked software which reduces software producer's motivations for striving market share. Hence, software producer has an incentive to reduce protection. From the above demonstrations, partial compatibility between genuine and cracked software is not only a phenomenon commonly-observed in computer industries, but also plays a crucial role in software producer's protection decision. We are thus motivated to construct a software market exhibiting network externalities with considering partial compatibility between genuine and cracked software to analyze the monopolistic producer's software protection behaviors as well as the determinants of existence of software cracking and the number of consumers using cracked software. It is worth demonstrating our setup with the document processing software market in which Microsoft (MS) Office owns monopoly power evidently. This market has the following features. First, consumers may decide whether to purchase genuine MS Office or use cracked MS Office illegally. Second, if consumers decide to use cracked MS Office illegally, then they not only need to devote enough effort to crack software successfully which are determined by the degrees of software protection imposed by MS, but also suffer utility loss from using cracked MS Office due to an inferior quality. Third, consumers using cracked MS Office generally cannot fully communicate with those using genuine MS Office. In other words, genuine and cracked MS Office are partially compatible.

Our research presents the following results. First, when network externalities are relatively weak, the monopolistic producer would set a degree of protection which induces existence of software cracking to enhance consumers' willingness to pay for genuine software by improving network benefits. On the contrary, if network externalities are sufficiently strong, then the monopolistic producer would set a degree of protection which stop software cracking completely. This implies that stopping software cracking is not possible without network effects.

It is worth mentioning that the lower limits of strength of network externalities of non-existence of software cracking increase with compatibility between genuine and cracked software which implies that reducing compatibility may lessen possibilities of existence of software cracking. We also show that if utility loss from using cracked software is sufficiently small with software protection costs being sufficiently large or the degrees of compatibility between software being sufficiently high, then cracked software may coexist with genuine software no matter what the strength of network externalities is which means that strong network externalities cannot stop software cracking in these two situations. Second, if utility loss from using cracked software is serious (mild), then stronger network externalities or reducing compatibility between genuine and cracked software induce monopolistic producer to set a higher (lower) degree of software protection; as a consequence, strengthening (weakening) network externalities or lowering (raising) compatibility may reduce the number of consumers using cracked software and increase software producer's profits under this situation. To sum up the above findings, even though reducing compatibility may lessen possibilities of existence of software cracking; it does not necessarily reduce the number of consumers using cracked software. In other words, lowering compatibility between genuine and cracked software exerts different effects in *lessening possibilities of existence of software cracking* and *reducing the number of consumers using cracked software*. Finally, we show that if genuine and cracked software are highly compatible or network externalities are relatively weak, then the monopolistic producer tends to over-protect software which implies that the number of consumers buying genuine software under the market equilibrium exceeds that under the social optimum.

The remainder of the paper is organized as follows. Section 2 introduces the basic model. Section 3 analyzes the monopolistic manufacturer's software protection decision. Section 4 examines social optimal software protection. Section 5 concludes and discusses robustness of our results which are derived under a specific utility function.

2 The Model

This section proposes a model in which the monopolistic software producer and consumers act to maximize profits and utility respectively. Consumers will base software adoption decisions on their expected network sizes of each software. To be specific, we assume that consumers must make their adoption decisions before the actual network sizes are known. In other words, given that the expected number of consumers buying genuine software and using cracked software are

x_g^e and x_c^e respectively, consumers make their software adoption decisions to maximize their utility.

Consumers may either buy genuine software or use cracked one illegally. Consumers' utility levels are determined by network benefits, software adoption costs, and their subjective preferences for genuine and cracked software which depend on the extents of accordance with social norms of copyright protection.

First, regarding network benefits, consumers gain more utility from software when the number of consumers using compatible products increase. We assume that network benefits increase with the number of consumers adopting (partially) compatible software linearly. Genuine and cracked software are partially compatible, and the degree of compatibility is $0 < \beta < 1$. From this viewpoint, given that the expected number of consumers buying genuine software and using cracked one are x_g^e and x_c^e respectively, if consumers buy genuine software, then they gain network benefits, $\alpha\left(x_g^e + \beta x_c^e\right)$. On the other hand, consumers may gain network benefits, $\alpha\left(x_c^e + \beta x_g^e\right)$ from using cracked software. Here, $\alpha < 1$ reflects strength of network externalities.

The second determinant is software adoption costs. If consumers purchase genuine software, then they pay software producer the price, p_g. Conversely, if consumers use cracked software illegally, then they suffer two kinds of utility losses. The first kind of utility loss comes from devoting effort for cracking software successfully which intuitively increase with degrees of software protection. To be specific, if the monopolistic producer sets the degree of software protection to be s_g, then consumers suffer disutility rs_g to crack software successfully. Here, $r > 0$ is marginal disutility of cracking more-protected software. A larger r means that cracking more-protected software causes a drastic increase in disutility. A smaller r alternatively means that cracking more-protected software causes a mild increase in disutility. The second kind of utility loss results from an inferior quality of cracked software. In real life, due to some unexpected problems, consumers usually feel that the malfunction risk of cracked software is higher than that of genuine software. Moreover, software producer may use some strategic tools to affect the extents of utility loss from using cracked software such as providing excellent after-sale services to the consumers buying genuine software exclusively to make consumers suffer serious utility loss from using cracked software. Intuitively, cracked software is unattractive if consumers would suffer serious utility loss from using it which may induce software producer to invest less in protection. From this viewpoint, the extents of utility loss from using cracked software affect software producer's protection decision essentially. Hence, we consider that consumers suffer utility loss, f from using cracked software in our analysis.

The third determinant is consumers' subjective preferences for genuine and cracked software which are determined by the extents of accordance with social norms of copyright protection. We assume that consumers are horizontally differentiated in accordance with social norms of copyright protection, and are uniformly distributed on the interval $[0, 1]$ with density 1. Let t denote consumers' locations which represent their accordance with social norms of copyright protection. Genuine and cracked software are located at $t = 0$ and $t = 1$ respectively. Consumers with smaller t have more accordance with social norms of copyright protection and prefer genuine software subjectively. Conversely, consumers with larger t have less accordance with social norms of copyright protection and prefer cracked software subjectively.[1]

From the above demonstrations, the utility function of a consumer located at t can be written as follows.

$$u(t) = \begin{cases} u + \alpha\left(x_g^e + \beta x_c^e\right) - p_g - t & \text{if he buys genuine software,} \\ u + \alpha\left(x_c^e + \beta x_g^e\right) - (rs_g + f) - (1 - t) & \text{if he uses cracked software.} \end{cases} \tag{1}$$

Here, u is the standalone utility of software which is sufficiently large that the market is covered.

The participants' interactions take place in the following three-stage game. In the first stage, the monopolistic producer determines a degree of software protection and incurs investment costs. In the second stage, the monopolist sets a price of genuine software to maximize profits. In the third stage, consumers decide whether to buy genuine software or use cracked one. In the following analysis, we use backward induction to characterize the subgame-perfect Nash equilibrium.

The marginal costs of producing genuine software are normalized to be 1, and investment costs for software protection is cs_g^2 if software producer sets the degree of protection to be s_g. In order to satisfy the second-order conditions and the number of consumers buying genuine software is positive, parameters are assumed to satisfy the following conditions.

(A1) $f > \alpha(1 - \beta)$.

(A2) $c > \frac{r^2}{8[1-\alpha(1-\beta)]}$.

1 The literature of public finance including de Juan, Lasheras, and Mayo (1994), Alm, McCelland, and Schulze (1999), and Wenzel (2005) discussed tax compliance from the viewpoint of social norms.

(A1) and (A2) mean that utility loss from using cracked software is sufficiently serious, and software protection costs are sufficiently high respectively.

It is worth demonstrating our setup with the documents processing software market in which MS Office owns monopoly power evidently. u is the standalone utility of MS Office, x_g^e and x_c^e are the expected number of consumers buying genuine MS Office and using cracked MS Office respectively. $0 < \beta < 1$ is the degree of compatibility between genuine and cracked MS Office. If consumers decide to use cracked MS Office, then they suffer disutility, rs_g to crack MS Office successfully; and utility loss, f due to an inferior quality and lack of after-sales supports cracked MS Office.

3 Equilibrium Analysis

Following Katz and Shapiro (1985) and Economides (1996), we adopt fulfilled expectations equilibrium which means that the expected number of consumers using each kind of software are the realized ones.

From (1), the consumer who is indifferent between buying genuine software and using cracked one is derived as follows.

$$x_g \equiv t = \frac{1 - p_g + \left(rs_g + f\right) - \alpha\left(x_c^e + \beta x_g^e\right) + \alpha\left(x_g^e + \beta x_c^e\right)}{2}. \tag{2a}$$

$$x_c \equiv 1 - t = 1 - \frac{1 - p_g + \left(rs_g + f\right) - \alpha\left(x_c^e + \beta x_g^e\right) + \alpha\left(x_g^e + \beta x_c^e\right)}{2}. \tag{2b}$$

Inserting $x_g^e = x_g$ and $x_c^e = x_c$ into (2a) and (2b), and solving them simultaneously yield the number of consumers buying genuine software and using cracked one as follows respectively.

$$x_g = \frac{(rs_g - p_g + f + \alpha\beta - \alpha + 1)}{2[1 - \alpha(1 - \beta)]}. \tag{3a}$$

$$x_c = 1 - \frac{\left(rs_g - p_g + f + \alpha\beta - \alpha + 1\right)}{2\left[1 - \alpha(1 - \beta)\right]} \tag{3b}$$

It is worth noting that the above demand functions are valid only when $0 < x_g < 1$. Moreover, $\alpha < 1$ ensures stability of the demand functions.

Next, we analyze the situation under which no consumers decide to use cracked software (a corner solution) by examining software adoption decision of the consumer with least accordance with social norms of copyright protection ($t = 1$).

$$u(1) = \begin{cases} u + \alpha \left[x_g^e + \beta\left(1 - x_g^e\right)\right] - p_g - 1 & \text{if he buys genuine software,} \\ u + \alpha \left[\left(1 - x_g^e\right) + \beta x_g^e\right] - \left(rs_g + f\right) & \text{if he uses cracked software.} \end{cases}$$

Given that the monopolistic producer sets the degree of protection and price of genuine software to be s_g and p_g respectively, and the expected number of consumers buying genuine software and using cracked software are x_g^e and x_c^e respectively, if the following condition is satisfied, then the consumer with $t = 1$ would buy genuine software rather than use cracked one which implies that there are no consumers using cracked software. In other words, software cracking is stopped completely under this situation.

$$\alpha \left[x_g^e + \beta\left(1 - x_g^e\right)\right] - p_g - 1 > \alpha \left[\left(1 - x_g^e\right) + \beta x_g^e\right] - \left(rs_g + f\right). \tag{4}$$

We analyze the monopolistic software producer's pricing behaviors. From demand function of genuine software derived in (3a), the monopolistic software producer's profit function can be written as follows

$$\pi_g = \frac{\left(p_g - 1\right)\left(rs_g - p_g + f + \alpha\beta - \alpha + 1\right)}{2\left[1 - \alpha\left(1 - \beta\right)\right]} - cs_g^2 \tag{5}$$

Solving the first-order condition yields monopolistic producer's optimal pricing of genuine software as follows.

$$p_g = \frac{\left(rs_g + f + \alpha\beta - \alpha + 2\right)}{2}. \tag{6}$$

We can analyze the impacts the degrees of software protection and utility loss from using cracked software on prices of genuine software which are stated in the following lemma.

Lemma.
(1) $\frac{dp_g}{ds_g} > 0.$
(2) $\frac{dp_g}{df} > 0.$

First, regarding the impacts of protection degrees of genuine software on its price, if the monopolistic producer increases protection, then consumers must suffer more disutility to crack software successfully, and are willing to pay a higher price for genuine software. Hence, the monopolistic producer has an incentive to sell more-protected software for a higher price under this situation.

Next, we discuss the impacts of utility losses from using cracked software on prices of genuine software. Consumers are similarly willing to pay a higher price for genuine software if they would suffer more utility loss from using cracked software, which induces the monopolistic producer to ask a higher price for genuine software under this situation. This result implies that providing more after-sale services to consumers buying genuine software may induce the monopolistic producer to sell genuine software for a higher price.

Inserting (6) into (3a) yields the number of consumers using genuine software as follows under the monopolistic producer setting degree of software protection to be s_g.

$$x_g = \frac{\left(rs_g + f + \alpha\beta - \alpha\right)}{4\left[1 - \alpha\left(1 - \beta\right)\right]}. \tag{7a}$$

Hence, the number of consumers using cracked software is

$$x_c = 1 - \frac{\left(rs_g + f + \alpha\beta - \alpha\right)}{4\left[1 - \alpha\left(1 - \beta\right)\right]}. \tag{7b}$$

From (7a), $\frac{dx_g}{ds_g} = \frac{r}{4[1-\alpha(1-\beta)]} > 0$. This means that raising protection enables software producer to gain a larger number of consumers buying genuine software.

Inserting (6) into (5) derives monopolistic software producer's profit function as follows.

$$\pi_g\left(s_g\right) = \frac{\left(rs_g + f + \alpha\beta - \alpha\right)^2}{8\left[1 - \alpha\left(1 - \beta\right)\right]} - cs_g^2 \tag{8}$$

Solving the first-order condition of the above profit function yields the monopolistic producer's optimal software protection as follows.

$$s_g^* = \frac{(\alpha\beta - \alpha + f)\,r}{-r^2 + 8c\left[1 - \alpha(1 - \beta)\right]} > 0. \tag{9}$$

Hence, the monopolistic producer's profit can be derived as follows which is positive due to assumption (A2).

$$\pi_g^* = \frac{c\left(\alpha\beta - \alpha + f\right)^2}{-r^2 + 8c[1 - \alpha(1 - \beta)]} > 0. \tag{10}$$

Inserting (9) into (7a) and (7b) derives the equilibrium number of consumers buying genuine software and using cracked one as follows respectively.

$$x_g^* = \frac{2c\left(\alpha\beta - \alpha + f\right)}{-r^2 + 8c[1 - \alpha(1 - \beta)]} > 0. \tag{11a}$$

$$x_c^* = 1 - \frac{2c\,(f + \alpha\beta - \alpha)}{-r^2 + 8c[1 - \alpha(1-\beta)]} = \frac{-r^2 - 2cf + 6\alpha\beta c - 6\alpha c + 8c}{-r^2 + 8c[1 - \alpha(1-\beta)]}. \tag{11b}$$

From (11a),

$$1 - x_g^* = \frac{-r^2 - 2cf + 6\alpha\beta c - 6\alpha c + 8c}{-r^2 + 8c[1 - \alpha(1-\beta)]}. \tag{11c}$$

From (11c), $0 < x_g^* < 1$ if and only if $-r^2 - 2cf + 6\alpha\beta c - 6\alpha c + 8c > 0$. This implies that if $-r^2 - 2cf + 8c < 0$, then $-r^2 - 2cf + 6\alpha\beta c - 6\alpha c + 8c < 0$ for all α which implies all consumers would buy genuine software in this situation. This means that if $-r^2 - 2cf + 8c < 0$, no matter what the strength of network externalities is, cracked software cannot coexist with genuine software in the market. Alternatively, if $-r^2 - 2cf + 8c > 0$, then the equilibrium outcome is determined by the sign of $-r^2 - 2cf + 6\alpha\beta c - 6\alpha c + 8c$.

(1) If $-r^2 - 2cf + 6\alpha\beta c - 6\alpha c + 8c > 0$, then $0 < x_g^* < 1$ which means that cracked software may coexist with genuine software in the market.
(2) If $-r^2 - 2cf + 6\alpha\beta c - 6\alpha c + 8c < 0$, then $x_c^* = 0$ which means that cracked software is stopped.

The above discussions imply that the strength of network externalities, α plays a role in existence of software cracking under $-r^2 - 2cf + 8c > 0$. Hence, we assume $-r^2 - 2cf + 8c > 0$ in the following analysis in existence of software cracking as well as its impacts on market performances.

Inserting s_g^* into (6) yields the price of genuine software price as follows.

$$p_g^* = 1 + \frac{4c(\alpha\beta - \alpha + 1)\,(\alpha\beta - \alpha + f)}{-r^2 + 8c[1 - \alpha(1-\beta)]} > 1. \tag{12}$$

Proposition 1.

(1) *If utility loss from using cracked software is serious* ($f > 1 - \frac{r^2}{8c}$)*, then* $\frac{ds_g^*}{d\alpha} > 0$ *and* $\frac{ds_g^*}{d\beta} < 0$.
(2) *If utility loss from using cracked software is mild* ($f < 1 - \frac{r^2}{8c}$)*, then* $\frac{ds_g^*}{d\alpha} < 0$ *and* $\frac{ds_g^*}{d\beta} > 0$.
(3) $\frac{ds_g^*}{df} > 0$.

Proof. See Appendix I.

First, we examine the impacts of strength of network externalities on software protection. Strengthening network externalities arises two effects on software producer's protection decision. First, given the network size of genuine software, stronger network externalities improve network benefits and consumers' willingness to pay for genuine software which induces the monopolistic producer to raise protection to sell genuine software for a higher price. Second, stronger network externalities encourage the monopolistic producer to strive for market share which induces it to reduce protection to sell genuine software for a lower price. The above results show that if utility loss from using cracked software is serious, then the former effect offsets the latter one and the degrees of software protection increase with strength of network externalities. On the contrary, if utility loss from using cracked software is mild, then the former effect is offset by the latter one. Hence, the degrees of software protection decrease with strength of network externalities.

Second, we discuss the impacts of compatibility between genuine and cracked software on software protection. Raising compatibility between genuine and cracked software similarly arises two effects on software producer's protection decision. First, an increase in compatibility enables cracked software users to free-ride genuine software buyers more and discourages the monopolistic producer to expand market share which induces it to reduce software protection which incurs less protection costs. Second, raising compatibility enables genuine software buyers to free-ride cracked software users more and enhances network benefits of genuine software which induces the monopolistic producer to increase software protection to set a higher price of genuine software. The above results show that if utility loss from using cracked software is serious, then the former effect offsets the latter one and the degrees of software protection decrease with compatibility between genuine and cracked software. Conversely, if utility loss from using cracked software is mild, then the former effect is offset by the latter one. Hence, the degrees of software protection increase with compatibility between genuine and cracked software.

Third, we analyze the impacts of utility loss from using cracked software on software protection. An increase in utility loss from using cracked software also arises two effects on software producer's protection decision. First, cracked software becomes less attractive and software producer has an incentive to reduce protection which incurs less protection costs. Second, consumers are willing to pay a higher price for genuine software which induces software producer to raise software protection to increase prices. The above result indicates that the latter effect offsets the former one; as a consequence, degrees of software protection increase with utility loss from using cracked software.

Inserting (9), (11a), and (12) into (4) yields the condition of non-existence of software cracking as follows.

$$\frac{2(r^2 + 2cf - 6\alpha\beta c + 6\alpha c - 8c)}{-r^2 + 8c[1 - \alpha(1 - \beta)]} > 0.$$

Due to the denominator being positive, the above condition implies that non-existence of software cracking is an equilibrium outcome if and only if $\left(r^2 + 2cf - 6\alpha\beta c + 6\alpha c - 8c\right) > 0$. Hence, we state the equilibrium outcomes about existence of software cracking as the following proposition.

Proposition 2.

(1) *If* $\alpha < \frac{-r^2+8c-2cf}{6c(1-\beta)}$, *then* $0 < x_c^* < 1$. *In other words, existence of software cracking is an equilibrium outcome.*

(2) *If* $\alpha > \frac{-r^2+8c-2cf}{6c(1-\beta)}$, *then* $x_c^* = 0$. *In other words, non-existence of software cracking is an equilibrium outcome.*

(3) *Cracked software may coexist with genuine software no matter what the strength of network externalities is under the following two circumstances.*

(3.1) *Utility loss from using cracked software is sufficiently small and software protection costs are sufficiently high i.e.* $f < 1 + 3\beta - \frac{r^2}{2c}$ *and* $c > \frac{r^2}{2(1-\alpha)}$.

(3.2) *Utility loss from using cracked software is sufficiently small with software protection costs being moderate and the degrees of compatibility between software being sufficiently high i.e.* $f < 1 + 3\beta - \frac{r^2}{2c}$ *and* $\frac{r^2}{2(1+3\beta)} < c < \frac{r^2}{2(1-\alpha)}$ *and* $\beta > \frac{r^2-2c(1-\alpha)}{2c(3+\alpha)}$.

Proof. See Appendix II.

The monopolistic software producer confronts the trade-off between improving network benefits of genuine software and suffering profit loss due to existence of software cracking. When network externalities are relatively weak, software producer would set a degree of protection which induces existence of software cracking to enhance network benefits of genuine software rather than sets a degree of protection which makes software cracking disappear.

Next, we discuss the situation with strong network externalities, $\alpha > \frac{-r^2+8c-2cf}{6c(1-\beta)}$. Given that consumers' expected number of consumers buying genuine software and using cracked software are fulfilled $\left(x_g^e = x_g^*, x_c^e = x_c^*\right)$, all consumers would purchase genuine software rather than use cracked one. In other words, the monopolistic producer sets a degree of protection, s_g^* which stops software cracking completely under this situation. This result also implies that it is not possible to stop software cracking without network effects.

Third, if consumers suffer mild utility loss from using cracked software as well as software protections costs being sufficiently high or the extent of compatibility between software being sufficiently strong, then cracked software may coexist with genuine software no matter what the strength of network externalities is. In other words, strong network externalities cannot stop software cracking in these two circumstances.

From Proposition 2, compatibility between genuine and cracked software, software protection costs, and utility loss from using cracked software affect existence of software cracking through the lower limit of strength of network externalities of non-existence of software cracking. In other words, the above three factors affect existence of software cracking indirectly.

$$\frac{\mathrm{d}}{\mathrm{d}\beta}\frac{-r^2+8c-2cf}{6c(1-\beta)} = \frac{-r^2+8c-2cf}{6c(1-\beta)^2} > 0. \tag{13a}$$

$$\frac{\mathrm{d}}{\mathrm{d}f}\frac{-r^2+8c-2cf}{6c(1-\beta)} = -\frac{1}{3(1-\beta)} < 0. \tag{13b}$$

$$\frac{\mathrm{d}}{\mathrm{d}c}\frac{-r^2+8c-2cf}{6c(1-\beta)} = \frac{r^2}{6c^2(1-\beta)} > 0. \tag{13c}$$

The above derivations indicate that the lower limit of strength of network externalities for non-existence of software cracking increases with compatibility between genuine and cracked software and software protection costs, but decreases in utility loss from using cracked software. This implies that reducing compatibility may lessen possibilities of *existence* of software cracking. However, it is worth mentioning that the impacts of reducing compatibility on the *number of consumers using cracked software* depend on utility loss from using cracked software which are presented in the following analysis.

The above derivations also indicate that reducing software protection costs or increasing utility loss from using cracked software may lessen possibilities of existence of software cracking as well.

Next, we discuss the determinant of the number of consumers using cracked software. From (11b),

$$\frac{\mathrm{d}x_c^*}{\mathrm{d}\alpha} = -\frac{2c(1-\beta)(r^2-8c+8cf)}{\{r^2-8c[1-\alpha(1-\beta)]\}^2}\begin{cases} <0 & \text{if } f > 1-\dfrac{r^2}{8c}. \\ >0 & \text{if } f < 1-\dfrac{r^2}{8c}. \end{cases} \tag{14a}$$

$$\frac{\mathrm{d}x_c^*}{\mathrm{d}\beta} = \frac{2\alpha c(r^2-8c+8cf)}{\{r^2-8c[1-\alpha(1-\beta)]\}^2}\begin{cases} >0 & \text{if } f > 1-\dfrac{r^2}{8c}. \\ <0 & \text{if } f < 1-\dfrac{r^2}{8c}. \end{cases} \tag{14b}$$

$$\frac{\mathrm{d}x_c^*}{\mathrm{d}c} = \frac{2r^2(\alpha\beta-\alpha+f)}{\{r^2-8c[1-\alpha(1-\beta)]\}^2} > 0. \tag{14c}$$

$$\frac{\mathrm{d}x_c^*}{\mathrm{d}f} = \frac{2c}{\{r^2-8c[1-\alpha(1-\beta)]\}} < 0. \tag{14d}$$

From the above derivations, the impacts of strength of network externalities and compatibility between genuine and cracked software on the number of consumers using cracked software are stated as the following proposition.

Proposition 3.
(1) *If* $f > 1 - \frac{r^2}{8c}$, *then* $\frac{\mathrm{d}x_c^*}{\mathrm{d}\alpha} < 0$ *and* $\frac{\mathrm{d}x_c^*}{\mathrm{d}\beta} > 0$.
(2) *If* $f < 1 - \frac{r^2}{8c}$, *then* $\frac{\mathrm{d}x_c^*}{\mathrm{d}\alpha} > 0$ *and* $\frac{\mathrm{d}x_c^*}{\mathrm{d}\beta} < 0$.

First, we discuss the impacts of strength of network externalities on the number of consumers using cracked software.

$$\frac{\mathrm{d}x_c^*}{\mathrm{d}\alpha} = \frac{\partial x_c}{\partial \alpha} + \frac{\partial x_c}{\partial s_g}\frac{\mathrm{d}s_g^*}{\mathrm{d}\alpha}. \tag{15}$$

From (7b),

$$\frac{\partial x_c}{\partial \alpha} = -\frac{(1-\beta)\left(rs_g^* + f - 1\right)}{4[1-\alpha(1-\beta)]^2}\begin{cases} < 0 & \text{if } f > 1 - \dfrac{r^2}{8c}. \\ > 0 & \text{if } f < 1 - \dfrac{r^2}{8c}. \end{cases} \tag{16a}$$

$$\frac{\partial x_c}{\partial s_g} = -\frac{r}{4[1-\alpha(1-\beta)]} < 0. \tag{16b}$$

Here, from (9),

$$\left(rs_g^* + f - 1\right) = \frac{(\alpha\beta - \alpha + 1)\left(r^2 + 8cf - 8c\right)}{-r^2 + 8c[1-\alpha(1-\beta)]}\begin{cases} > 0 & \text{if } f > 1 - \dfrac{r^2}{8c} \\ < 0 & \text{if } f < 1 - \dfrac{r^2}{8c} \end{cases} \tag{17}$$

Stronger network externalities arise direct and indirect effects on the number of consumers using cracked software. When utility loss from using cracked software is serious, stronger network externalities reduce the number of consumers using cracked software directly. On the other hand, due to raising software protection, stronger network externalities reduce the number of consumers using cracked software indirectly as well. As a consequence, the number of consumers using cracked software decrease with the extents of network externalities. From this viewpoint, strengthening network externalities may reduce software cracking under this situation. Conversely, if consumers suffer mild utility loss from using cracked software, then strengthening network externalities yields more consumers using cracked software.

Next, we study the impacts of compatibility between genuine and cracked software on the number of consumers using cracked software.

$$\frac{\mathrm{d}x_{\mathrm{c}}^*}{\mathrm{d}\beta} = \frac{\partial x_{\mathrm{c}}}{\partial \beta} + \frac{\partial x_{\mathrm{c}}}{\partial s_{\mathrm{g}}} \frac{\mathrm{d}s_{\mathrm{g}}^*}{\mathrm{d}\beta}. \tag{18}$$

From (7b),

$$\frac{\partial x_{\mathrm{c}}}{\partial \beta} = \frac{\alpha\left(rs_{\mathrm{g}}^* + f - 1\right)}{4[1-\alpha(1-\beta)]^2} \begin{cases} > 0 & \text{if } f > 1 - \dfrac{r^2}{8c}. \\ < 0 & \text{if } f < 1 - \dfrac{r^2}{8c}. \end{cases} \tag{19}$$

Raising compatibility similarly arises direct and indirect effects on the number of consumers using cracked software. When utility loss from using cracked software is serious, raising compatibility yields more consumers using cracked software directly. On the other hand, due to reducing software protection, raising compatibility increases the number of consumers using cracked software indirectly as well. Hence, the number of consumers using cracked software increase with compatibility between genuine and cracked software. This implies that reducing compatibility may reduce software cracking under this situation. Conversely, if consumers suffer mild utility loss from using cracked software, then raising compatibility may reduce the number of consumers using cracked software.

As demonstrated above, reducing compatibility may lessen possibilities of existence of software cracking. However, from Proposition 3, the impacts of reducing compatibility on the number of consumers using cracked software are dependent on utility loss from using cracked software. The second part of Proposition 3 further indicates that reducing compatibility may yield more consumers using cracked software when utility loss from using cracked software is mild. From this viewpoint, reducing compatibility exerts distinct effects on possibilities of existence of software cracking and the number of consumers using cracked software.

(14c) and (14d) deliver intuitive results that the number of consumers using cracked software increase with software protection costs and decrease with utility loss from using cracked software respectively.

We close our analysis about firms' behaviors with examining the impacts of strength of network externalities and compatibility between genuine and cracked software on software producer's profits.

From (10), it is straightforward to derive the following results.

$$\frac{d\pi_g^*}{d\alpha} = \frac{2c(1-\beta)(\alpha\beta-\alpha+f)[r^2-4c(2-\alpha+\alpha\beta-f)]}{\{r^2-8c[1-\alpha(1-\beta)]\}^2} \begin{cases} > 0 & \text{if } f > 2-\alpha+\alpha\beta-\frac{r^2}{4c}. \\ < 0 & \text{if } f < 2-\alpha+\alpha\beta-\frac{r^2}{4c}. \end{cases} \tag{20a}$$

$$\frac{d\pi_g^*}{d\beta} = -\frac{2\alpha c(\alpha\beta-\alpha+f)[r^2-4c(2-\alpha+\alpha\beta-f)]}{\{r^2-8c[1-\alpha(1-\beta)]\}^2} \begin{cases} < 0 & \text{if } f > 2-\alpha+\alpha\beta-\frac{r^2}{4c}. \\ > 0 & \text{if } f < 2-\alpha+\alpha\beta-\frac{r^2}{4c}. \end{cases} \tag{20b}$$

We state comparative statics of software producer's profit as Proposition 4. It is worth noting that the critical value indicated in the above expressions does not contradict with assumption (A1). Moreover, the condition $f > 2-\alpha+\alpha\beta-\frac{r^2}{4c}$ presented in Proposition 4 is compatible with (A1) as well as $-r^2+8c-2cf > 0$.[2]

Proposition 4.

(1) *If utility loss from using cracked software is serious* $(f > 2-\alpha+\alpha\beta-\frac{r^2}{4c})$, *then* $\frac{d\pi_g^*}{d\alpha} > 0$ *and* $\frac{d\pi_g^*}{d\beta} < 0$.

(2) *If utility loss from using cracked software is mild* $(f < 2-\alpha+\alpha\beta-\frac{r^2}{4c})$, *then* $\frac{d\pi_g^*}{d\alpha} < 0$ *and* $\frac{d\pi_g^*}{d\beta} > 0$.

If utility loss from using cracked software is serious, from the first part of Proposition 3, strengthening network externalities or lowering compatibility between genuine and cracked software may reduce the number of consumers using cracked software; as a consequence, software producer's profits increase with strength of network externalities, but decrease with compatibility between genuine and cracked software under this situation. On the other hand, if utility loss from using cracked software is mild, then software producer's profits conversely decrease with strength of network externalities, but increase with compatibility between genuine and cracked software due to an increase in number of consumers using cracked software under this situation.

2 The proof is offered in Appendix III.

4 Social Welfare

This section discusses social desirability of software protection. In the following analysis, we consider that in the first stage, a social planner sets a degree of software protection to maximize social welfare which is defined to be the sum of software manufacturer's profit and consumers' surplus. The participants' behaviors in stage 2 and stage 3 remain unchanged.

From the above derivations, given that degree of software protection is s, consumers' surplus can be derived as follows.

$$
\begin{aligned}
&cs\,(s) \\
&= \int_0^{x_g} \left[u + \alpha\left(x_g + \beta\left(1 - x_g\right)\right) - p_g - t\right] \mathrm{d}t + \int_{x_g}^{1} \left[u + \alpha\left(\left(1 - x_g\right) + \beta x_g\right)\right. \\
&\quad \left. - (rs + f) - (1 - t)\right] \mathrm{d}t\cdot \\
&\quad = u + \alpha\left(x_g - \beta x_g + \beta\right) x_g - p_g x_g - \frac{x_g^2}{2} + \alpha\left(1 - x_g + \beta x_g\right)\left(1 - x_g\right) \\
&\qquad - (rs + f)\left(1 - x_g\right) - \left(1 - x_g\right) + \left(\frac{1}{2} - \frac{x_g^2}{2}\right). \\
&\qquad\qquad = u + \frac{\Omega\,(\mathrm{s})}{16[1 - \alpha(1 - \beta)]^2}.
\end{aligned}
\tag{21}
$$

Here,

$$
\begin{aligned}
\Omega\,(s) = {} & r^2 s^2 + \left(-12\alpha^2\beta^2 r + 24\alpha^2\beta r - 12\alpha^2 r - 26\alpha\beta r + 2fr + 26\alpha r - 16r\right) s \\
& + \left(4\alpha^3\beta^3 + 4\alpha^3\beta^2 - 12\alpha^2\beta^2 f - 20\alpha^3\beta - 3\alpha^2\beta^2 + 24\alpha^2\beta f + 12\alpha^3\right. \\
& \left. + 38\alpha^2\beta - 12\alpha^2 f - 26\alpha\beta f - 35\alpha^2 - 16\alpha\beta + 26\alpha f + f^2 + 32\alpha - 16f - 8\right).
\end{aligned}
$$

Hence, social welfare function can be derived as follows.

$$
w\,(s_w) = u + \frac{\Omega(s_w)}{16[1 - \alpha(1 - \beta)]^2} + \frac{(rs_w + f + \alpha\beta - \alpha)^2}{8[1 - \alpha(1 - \beta)]} - cs_w^2. \tag{22}
$$

It is worth mentioning that social welfare function is derived under coexistence of genuine and cracked software.

Solving first-order condition yields social optimum degree of software protection.[3]

$$s_{\mathrm{w}}^{*}=-\frac{r\left(2\alpha\beta f-2\alpha f+3f-4\alpha^{2}\beta^{2}+8\alpha^{2}\beta-11\alpha\beta-4\alpha^{2}+11\alpha-8\right)}{2\alpha\beta r^{2}-2\alpha r^{2}+3r^{2}-16\alpha^{2}\beta^{2}c+32\alpha^{2}\beta c-32\alpha\beta c-16\alpha^{2}c+32\alpha c-16c}. \tag{23}$$

Proposition 5. *If $\beta > 1-\frac{2}{3\alpha}$, then monopolist producer over-protects software.*

Proof. See Appendix IV.

From the above analysis, reducing software protection may enhance social welfare by saving protection costs and reducing prices of genuine software. However, the monopolistic software producer earns less profits under this situation. The above proposition indicates that when genuine and cracked software are highly compatible, gains from reducing protections costs and consumers' surplus due to a lower software price offset monopolistic software producer's profit loss. Therefore, reducing software protection enhances social welfare under this situation. It is worth noting that if $\alpha < \frac{2}{3}$, then $1-\frac{2}{3\alpha} < 0$ which implies that the monopolistic manufacturer always over-protects genuine software when network externalities are relatively weak.

Regarding social optimal number of consumers buying genuine software, from (7a), $\frac{\mathrm{d}x_{\mathrm{g}}}{\mathrm{d}s_{\mathrm{g}}}=\frac{r}{2[1-\alpha(1-\beta)]}>0$ which means that the number of consumers buying genuine software increases with degrees of protection. Hence, Proposition 4 implies that if genuine and cracked software is highly compatible or network externalities are relatively weak, then the number of consumers buying genuine software under the market equilibrium are excessive. In other words, the number of consumers using cracked software under market equilibrium are inadequate for social optimum.

5 Conclusion and Discussions of Robustness

In the past decades, many economists have adopted various approaches to analyze firms' software protection behaviors and the impacts of software piracy on profits. Nevertheless, some phenomena which can be observed frequently in computer industries have not received much attention so far. First, progress of

3 $\frac{\mathrm{d}^{2}w(s_{\mathrm{w}})}{\mathrm{d}s_{\mathrm{w}}^{2}}=\frac{2\alpha\beta r^{2}-2\alpha r^{2}+3r^{2}-16\alpha^{2}\beta^{2}c+32\alpha^{2}\beta c-32\alpha\beta c-16\alpha^{2}c+32\alpha c-16c}{8[1-\alpha(1-\beta)]^{2}}<0$. Hence, the second-order condition is satisfied.

hardware technologies and diffusion of computer knowledge have made significant changes in the patterns of software piracy. Nowadays if consumers decide to use software illegally, then they would crack genuine software by themselves rather than purchase pirated one. From this viewpoint, degrees of protection determine the amount of effort required for cracking software successfully. We are thus motivated to investigate a monopolistic producer's software protection behaviors and examine the determinants of existence of software cracking as well as the number of consumers using cracked software.

Our research presents the following results. First, when network externalities are relatively weak, software producer may set a degree of protection which induces existence of software cracking as an equilibrium outcome. Conversely, when network externalities are sufficiently strong, software producer would set a degree of protection under which software cracking is stopped completely. From this viewpoint, stopping software cracking is not possible without network effects. We also discover that if utility loss from using cracked is sufficiently small with software protection costs being sufficiently large or the extent of compatibility between software being sufficiently strong, then strong network externalities cannot stop software cracking.

Second, regarding determinants of the number of consumers using cracked software, if utility loss from using cracked software is serious (mild), then the number of consumers using cracked software decrease (increase) with extents of network externalities, but increase (decrease) with compatibility between genuine and cracked software. From this viewpoint, strengthening (weakening) network externalities or reducing (raising) compatibility may reduce software cracking if consumers suffer serious (mild) utility loss from using cracked software. It is worth mentioning that even though reducing compatibility may lessen possibilities of existence of software cracking, but does not necessarily yield fewer consumers using cracked software. Finally, we show that if genuine and cracked software are highly compatible or network externalities are relatively weak, then the monopolistic producer tends to over-protect software which implies that the number of consumers buying genuine software under the market equilibrium are excessive.

Finally, we discuss robustness of the findings about existence of software cracking which are derived under a specific utility function in this paper. The monopolistic producer confronts trade-off between improving network benefits of genuine software and suffering profit loss due to existence of software cracking when setting degrees of software protection. When network effects are relatively weak, due to an increase in the number of consumers using (partially) compatible products, existence of software cracking improves consumers' willingness to pay for genuine software significantly. Hence, the monopolistic producer sets a degree of protection to accommodate software cracking under this situation. Conversely, if network effects are sufficiently strong, then network benefits of genuine software without software cracking are large enough for producer to set a genuine software price which maximizes its profit. Hence, the software producer sets a degree of protection to stop software cracking completely under this situation. From these demonstrations, we expect that similar effects may be present under a general setup.

Appendix I. Proof of Proposition 1

$$(1)\ \frac{\mathrm{d}s_{\mathrm{g}}^*}{\mathrm{d}\alpha} = \frac{(1-\beta)\,r[r^2 - 8c\,(1-f)]}{\{r^2 - 8c[1-\alpha(1-\beta)]\}^2}\begin{cases} > 0 & \text{if } f > 1 - \dfrac{r^2}{8c}. \\ < 0 & \text{if } f < 1 - \dfrac{r^2}{8c}. \end{cases}$$

$$(2)\ \frac{\mathrm{d}s_{\mathrm{g}}^*}{\mathrm{d}\beta} = -\frac{\alpha r[r^2 - 8c\,(1-f)]}{\{r^2 - 8c[1-\alpha(1-\beta)]\}^2}\begin{cases} < 0 & \text{if } f > 1 - \dfrac{r^2}{8c}. \\ > 0 & \text{if } f < 1 - \dfrac{r^2}{8c}. \end{cases}$$

$$(3)\ \frac{\mathrm{d}s_{\mathrm{g}}^*}{\mathrm{d}f} = \frac{r}{-r^2 + 8c[1-\alpha(1-\beta)]} > 0.$$

Finally, we prove that $f > 1 - \frac{r^2}{8c}$ which is equivalent to $r^2 > 8c(1-f)$ does not contradict with (A2) i.e. $c > \frac{r^2}{8[1-\alpha(1-\beta)]}$ which is equivalent to $r^2 < 8c\,[1-\alpha\,(1-\beta)]$.

$8c\,[1-\alpha\,(1-\beta)] - 8c\,(1-f) = 8c\,(\alpha\beta - \alpha + f) > 0$ due to (A1), i.e. $f > \alpha(1-\beta)$.

Hence, $f > 1 - \frac{r^2}{8c}$ does not contradict with (A2), i.e. $c > \frac{r^2}{8[1-\alpha(1-\beta)]}$.

In other words, both comparative statics results indicated above could happen in equilibrium.

This completes the proof. □

Appendix II. Proof of Proposition 2

The proof of the condition of existence of cracked software is offered in the main text.

First, we prove the condition of stopping software cracking equivalent to $r^2 > -2cf + 6\alpha\beta c - 6\alpha c + 8c$ being compatible with (A2), i.e. $c > \frac{r^2}{8[1-\alpha(1-\beta)]}$ which is equivalent to $r^2 < 8c\,[1-\alpha\,(1-\beta)]$.

$8c\,[1-\alpha\,(1-\beta)] - (-2cf + 6\alpha\beta c - 6\alpha c + 8c) = 2c\,(\alpha\beta - \alpha + f) > 0$ due to (A1), i.e. $f > \alpha(1-\beta)$.

Hence, the condition for non-existence of software cracking, $\alpha > \frac{-r^2+8c-2cf}{6c(1-\beta)}$ is compatible with (A2), i.e. $c > \frac{r^2}{8[1-\alpha(1-\beta)]}$.

In other words, both existence and non-existence of software cracking could be an equilibrium outcome.

The proof of the third part is stated as follows.

If the critical strength of network externalities of stopping software cracking is larger than 1 i.e. $\frac{-r^2+8c-2cf}{6c(1-\beta)} > 1$, then it is impossible to stop software cracking.

Hence, if $f < 1+3\beta - \frac{r^2}{2c}$, then cracked software can coexist with genuine software no matter what the strength of network externalities is. It is worth noting that due to utility loss from using cracked software cannot be negative, $c > \frac{r^2}{2(1+3\beta)}$ must be satisfied for $1+3\beta-\frac{r^2}{2c} > 0$.

Moreover, it is necessary to check compatibility between $f < 1+3\beta-\frac{r^2}{2c}$ and (A1) i.e. $f > \alpha(1-\beta)$.

$$\left(1+3\beta-\frac{r^2}{2c}\right)-\alpha\left(1-\beta\right)=\frac{-r^2+2c\left(1-\alpha\right)+2\beta c\left(3+\alpha\right)}{2c} < \frac{-r^2+8c\left[1-\alpha\left(1-\beta\right)\right]}{2c}.$$

$$\frac{-r^2+2c\left(1-\alpha\right)+2\beta c\left(3+\alpha\right)}{2c}-\frac{-r^2+8c\left[1-\alpha\left(1-\beta\right)\right]}{2c}=-3\left(1-\alpha\right)\left(1-\beta\right)<0.$$

Hence, $\frac{-r^2+8c[1-\alpha(1-\beta)]}{2c} > \frac{-r^2+2c(1-\alpha)+2\beta c(3+\alpha)}{2c}$ which implies that $\left(1+3\beta-\frac{r^2}{2c}\right)$ can be larger or smaller than $\alpha\left(1-\beta\right)$ due to $\frac{-r^2+8c[1-\alpha(1-\beta)]}{2c} > 0$.

(i) If $-r^2+2c\left(1-\alpha\right) > 0$ which is equivalent to $c > \frac{r^2}{2(1-\alpha)}$, then $\left(1+3\beta-\frac{r^2}{2c}\right) > \alpha\left(1-\beta\right)$ which implies that $f < 1+3\beta-\frac{r^2}{2c}$ is compatible with (A1).

(ii) If $-r^2+2c\left(1-\alpha\right) < 0$ which is equivalent to $c < \frac{r^2}{2(1-\alpha)}$, and $\beta > \frac{r^2-2c(1-\alpha)}{2c(3+\alpha)}$, then $\left(1+3\beta-\frac{r^2}{2c}\right) > \alpha\left(1-\beta\right)$ which implies that $f < 1+3\beta-\frac{r^2}{2c}$ is compatible with (A1).

(iii) If $-r^2+2c\left(1-\alpha\right) < 0$ which is equivalent to $c < \frac{r^2}{2(1-\alpha)}$, and $\beta < \frac{r^2-2c(1-\alpha)}{2c(3+\alpha)}$, then $\left(1+3\beta-\frac{r^2}{2c}\right) < \alpha\left(1-\beta\right)$ which implies that $f < 1+3\beta-\frac{r^2}{2c}$ is incompatible with (A1).

From the above derivations, cracked software can coexist with genuine software no matter what the strength of network externalities is under the following two circumstances.

(1) $c > \frac{r^2}{2(1-\alpha)}$ and $f < 1+3\beta-\frac{r^2}{2c}$.

(2) $\frac{r^2}{2(1+3\beta)} < c < \frac{r^2}{2(1-\alpha)}$ and $\beta > \frac{r^2-2c(1-\alpha)}{2c(3+\alpha)}$ and $f < 1+3\beta-\frac{r^2}{2c}$.

It is worth mentioning that $\frac{r^2-2c(1-\alpha)}{2c(3+\alpha)} < 1$ due to (A2) i.e. $c > \frac{r^2}{8[1-\alpha(1-\beta)]}$ which implies that $r^2 < 8c$.

Finally, it is also worth demonstrating the software producer's behaviors when all consumers would buy genuine software rather than use cracked software given that the software producer sells products for $p_g^* = 1 + \frac{4c(\alpha\beta-\alpha+1)(\alpha\beta-\alpha+f)}{-r^2+8c[1-\alpha(1-\beta)]}$.

If $-\left(r^2+2cf-6\alpha\beta c+6\alpha c-8c\right) < 0$, given that the expected number of consumers buying genuine software and using cracked one are fulfilled, all consumers would purchase genuine software rather than use cracked one. This means that the monopolistic software producer sets a degree of protection, $s_g^* = \frac{(\alpha\beta-\alpha+f)r}{-r^2+8c[1-\alpha(1-\beta)]}$ which may stop software cracking completely.

Hence, the monopolistic software producer's profit function can be derived as follows under this situation.

$$\pi_g = \frac{\left(p_g-1\right)\left(u-p_g\right)}{(1-\alpha)}.$$

Solving first-order condition yields optimal pricing of genuine software being $\frac{(u+1)}{2}$.

This completes the proof. □

Appendix III. Compatibility between Condition in Proposition 4 and (A1) as well as $-r^2+8c-2cf>0$

First, we prove that $f < 2-\alpha+\alpha\beta-\frac{r^2}{4c}$ is compatible with (A1) i.e. $f > \alpha(1-\beta)$.

$(2-\alpha+\alpha\beta-\frac{r^2}{4c}) - \alpha(1-\beta) = \frac{-r^2+8c[1-\alpha(1-\beta)]}{4c} > 0$ due to (A2) i.e. $c > \frac{r^2}{8[1-\alpha(1-\beta)]}$.

Hence, $f < 2-\alpha+\alpha\beta-\frac{r^2}{4c}$ and $f > \alpha(1-\beta)$ are compatible.

Second, $-r^2+8c-2cf > 0$ which is equivalent to $f < 4-\frac{r^2}{2c}$ is essentially an assumption of the analysis in the impacts of software cracking on market performances. Hence, it is worthwhile to check compatibility between $f > 2-\alpha+\alpha\beta-\frac{r^2}{4c}$ and $f < 4-\frac{r^2}{2c}$.

$$\left(4-\frac{r^2}{2c}\right)-\left(2-\alpha+\alpha\beta-\frac{r^2}{4c}\right) = \frac{-r^2-4\alpha\beta c+4\alpha c+8c}{4c}$$

$$> \frac{-r^2+8c\left[1-\alpha\left(1-\beta\right)\right]}{4c} > 0.$$

Hence, $f < 4 - \frac{r^2}{2c}$ and $f > \left(2 - \alpha + \alpha\beta - \frac{r^2}{4c}\right)$ are compatible.
This completes the proof. □

Appendix IV. Proof of Proposition 5

From (9) and (22),

$$s_{\mathrm{w}}^* - s_{\mathrm{g}}^* = \frac{2r[1-\alpha(1-\beta)]\Sigma\left(r^2\right)}{8\{r^2 - 8c[1-\alpha(1-\beta)]\}[1-\alpha(1-\beta)]^2 \frac{\mathrm{d}^2 w(s)}{\mathrm{d}s^2}}.$$

Here,

$$\Sigma\left(r^2\right) = (3\alpha\beta - 3\alpha + 4)\, r^2 + 4c\left(f - 6\alpha^2\beta^2 + 12\alpha^2\beta^2 - 13\alpha\beta - 6\alpha^2 + 13\alpha - 8\right).$$

Both $(\alpha\beta - \alpha + 1)$ and denominator are positive. Hence,

$$\operatorname{sign}\left[s_{\mathrm{w}}^* - s_{\mathrm{g}}^*\right] = \operatorname{sign}\left[\Sigma\left(r^2\right)\right].$$

From Proposition 1, genuine and cracked software coexists only when $r^2 < 6\alpha\beta c - 6\alpha c + 8c - 2cf$. In other words, the above social welfare function is valid only when $r^2 < 6\alpha\beta c - 6\alpha c + 8c - 2cf$.

From (A1), $3\alpha\beta - 3\alpha + 4 > 0$. Therefore,

$$\Sigma\left(r^2\right) < \Sigma\left(6\alpha\beta c - 6\alpha c + 8c - 2cf\right) = -2c\left(\alpha\beta - \alpha + f\right)\left(3\alpha\beta - 3\alpha + 2\right).$$

Hence, if $(3\alpha\beta - 3\alpha + 2) > 0$, then $\Sigma\left(r^2\right) < \Sigma\left(6\alpha\beta c - 6\alpha c + 8c - 2cf\right) < 0$.
In other words, if $(3\alpha\beta - 3\alpha + 2) > 0$, then $\Sigma\left(r^2\right)$ is always negative.
Therefore, when $(3\alpha\beta - 3\alpha + 2) > 0$, $s_{\mathrm{w}}^* < s_{\mathrm{g}}^*$.
This implies that if $\beta > 1 - \frac{2}{3\alpha}$, then $s_{\mathrm{w}}^* < s_{\mathrm{g}}^*$.
This completes the proof. □

References

Alm, J., G. McCelland, and W. Schulze. 1999. "Changing the Social Norm of Tax Compliance by Voting." *Kyklos* 52: 141–71.

Besen, S., and S. Kirby. 1989. "Private Copying, Appropriability, and Optimal Copying Royalties." *The Journal of Law and Economics* 32: 255–80.

Conner, K., and R. Rulmet. 1991. "Software Piracy: An Analysis of Protection Strategies." *Management Science* 37: 125–39.

de Juan, A., M. Lasheras, and R. Mayo. 1994. "Voluntary Tax Compliant Behavior of Spanish Income Tax Payers." *Public Finance* 49: 90–105.

Economides, N. 1996. “Network Externalities, Complementarities, and Invitations to Enter.” *European Journal of Political Economy* 12: 211–33.
Inceoglu, F. 2015. “Copyright Protection and Entry Deterrence.” *Information Economics and Policy* 32: 38–45.
Jain, S. 2008. “Digital Piracy: A Competitive Analysis.” *Management Science* 27: 610–26.
Johnson, W. 1985. “The Economics of Copying.” *Journal of Political Economy* 93: 158–74.
Katz, M., and C. Shapiro. 1985. “Network Externalities, Competition, and Compatibility.” *The American Economic Review* 75: 424–40.
Liebowitz, S. 1985. “Copyright and Indirect Appropriability: Photocopying of Journals.” *Journal of Political Economy* 93: 945–57.
Novos, I., and M. Waldman. 1984. “The Effects of Increased Copyright Protection: An Analytical Approach.” *Journal of Political Economy* 92: 236–46.
Rasch, A., and T. Wenzel. 2013. “Piracy in a Two-Sided Software Market.” *Journal of Economic Behavior & Organization* 88: 78–89.
Shy, O., and J. Thisse. 1999. “A Strategic Approach to Software Protection.” *Journal of Economics and Management Strategy* 8: 163–90.
Slive, J., and D. Bernhardt. 1998. “Pirated for Profit.” *Canadian Journal of Economics* 31: 886–99.
Takayama, L. 1994. “The Welfare Implications of Unauthorized Reproduction of Intellectual Property in the Presence of Network Externalities.” *The Journal of Industrial Economics* 42: 155–66.
Wenzel, M. 2005. “Motivation or Rationalization? Causal Relations between Ethics, Norms and Tax Compliance.” *Journal of Economic Psychology* 26: 491–508.
Yoon, K. 2002. “The Optimal Level of Copyright Protection.” *Information Economics and Policy* 14: 327–48.