

# Durable Goods Monopoly with Network Externalities with Application to the PC Operating Systems Market<sup>1</sup>

by

Nicholas Economides

## *Abstract*

We analyze a model of multi-period monopoly in durable goods. Taking into consideration the special conditions of software markets, we assume that there are no used software markets and that manufacturers stop selling older software when they introduce a replacement model. We show that nominal as well as discounted (real) prices decrease over time but are above cost, thereby violating the Coase conjecture. In contrast, when “new” durable goods are introduced by the monopolist which are only partially compatible with “old” durable goods, prices may increase over time. This occurs when the intensity of network externalities arising from weak partial forward compatibility (influencing the demand of the old good from sales of the new one) is low compared to the intensity of network externalities arising from partial backward compatibility (influencing the demand of the new good from sales of the old one). A new product introduced by an entrant is successful only when it has strong externalities arising from forward compatibility. In this case, the entrant and the incumbent have opposite incentives regarding the degree of forward compatibility of the new product (that defines the extent of network externalities of the old product on consumers of the new one).

Key words: durable goods, monopoly, network externalities

JEL classification numbers: L1, D4

Stern School of Business, New York, NY 10012, tel. (212) 998-0864, fax (212) 995-4218, e-mail [neconomi@stern.nyu.edu](mailto:neconomi@stern.nyu.edu), <http://www.stern.nyu.edu/networks/>

---

<sup>1</sup>

I thank Pino Lopomo and Andew Biehl for helpful comments on an earlier draft.

# Durable Goods Monopoly with Network Externalities with Application to the PC Operating Systems Market

## 1. Introduction

During the last decade, the network externalities literature has grown considerably, but has paid relatively little attention to durable goods markets. However, often durable goods exhibit network externalities. In fact, very significant current litigation (US v. Microsoft) centers on the degree of competition in the market for operating systems, which are durable goods. Surprisingly, although everyone recognizes that software is a durable good, almost all the arguments and discussion in the US v. MS case were made as if the software were a perishable good.

Traditionally, economists have analyzed infinitely-lived durable goods. The debate has centered on the validity of the Coase conjecture, which states that the infinitely-lived durable goods monopolist prices every period at marginal cost. Much of the debate has centered on various technical details of the models, and the results on the validity of the Coase conjecture are mixed.<sup>2</sup>

Robin Mason (1998) introduces network externalities in the standard model of durable goods that last forever and are sold in every instant up to infinity.<sup>3</sup> He finds that the perfectly competitive and the monopoly solutions coincide, since in both cases prices equal marginal cost. He also finds, as expected in his framework, that both the competitive market and the monopolist produce less than a planner who would fully internalize the externality.

For software markets, the model of durable goods that live and are traded forever and is not very useful. Of course, software is durable. But, if most of the value of an operating system comes from software applications that work with it (and the network externalities they imply), the durability of the operating system is not infinite, but rather depends on how many applications are written for it. The history of computer operating systems is littered with operating systems that were sometime successful and now are defunct. So, in modeling the operating systems market, it makes sense to consider durable goods that have a finite economic life. Even if software is assumed to have an infinite economic life, in modeling competitive interactions in the software market, it makes sense to allow the same product to be sold only during a few periods, to be replaced by another model later.

---

<sup>2</sup> See Morch von der Fehr and Kuehn (1997) for a recent re-examination of the Coase conjecture in models without network externalities.

<sup>3</sup> Because it utilizes the standard Bellman technique, Mason's (1998) model has the limitation that network externalities are not allowed to vary over time with the size of the evolving network.

The second interesting feature of the operating systems market, and more generally of software markets, is the lack of a used goods or secondary market. This arises for three reasons. First, often software is pre-installed and not portable to another computer. Second, even when software is portable, it is often hard to uninstall it, so that the used software can be legally sold. Third, often the license of software is not transferable to another user. Thus, in a model of competition in operating systems, it makes sense to assume that there is no used goods market.<sup>4</sup>

The third interesting feature of the operating systems market, and more generally of software markets, is that it is very common for a company to stop selling an old product when it introduces a new one.<sup>5</sup> So, given the absence of a market for used software, a new product competes with older products of the same company only across time periods.

The value of operating systems, even of very sophisticated operating systems, such as Windows or Unix is greatly enhanced by the existence of software applications that are compatible with the particular operating system. The higher the availability and sales of compatible applications, the higher the value of an operating system. Moreover, the higher the sales of an operating system, the more profitable are applications written for it. These positive feedback effects can be summarized by assuming that that every unit of sales of an operating system increases the willingness to pay for that system. This is commonly called the “network externality” or “network effect” of the sales of the operating system.

It is natural for competitors in software markets to try to capture and benefit from network externalities to the extent that it is possible. Thus, firms that design operating systems have strong incentives (i) to be compatible with existing applications, and run them better and faster if possible; and (ii) to promote the creation of applications compatible with their operating system. By now, it is common to have a release of hundreds of new applications simultaneously with the release of an operating system. Thus, it makes sense for network externalities associated with an operating system to be modeled as appearing immediately at the date of the operating system’s release, while, of course, externalities can grow as more applications are written.<sup>6</sup>

In a multi-good multi-period model, there are three types of compatibility, and each of these produces network externalities. First, there is full compatibility between

---

<sup>4</sup> Of course, there is a market for used personal computers which typically have some software installed. I only argue that there is a lack of a robust market for used operating system software that is disembodied from the hardware.

<sup>5</sup> Of course, software companies typically provide support for the installed base of earlier versions for some time.

<sup>6</sup> This is in contrast with the model of Bensaid and Lesne (1996), where externality effects are delayed.

the operating system and its applications. Second, when a new operating system is introduced in a future period, there is compatibility of the new operating system with existing (“old”) applications which we call “backward compatibility.” Clearly, new operating systems gain substantial value from being backwardly compatible. This may force the new product to be more similar to the old one, and therefore may intensify competition, if the older one is sold by a different company. Firms have to balance the benefits of extensive network externalities created by backward compatibility with the potential drawback of more intense competition. This is discussed in more detail in Economides and Flyer (1997) and in Jonard and Schenck (1998).

It is also worth noting that the cycles of introductions of new operating systems and applications software do not have to coincide. Moreover, depending on the technical differences between an old and a new operating system, new or upgraded applications software may be compatible under both the old and the new operating systems.

The third type of compatibility is the degree of compatibility with an older operating system with applications written primarily for a new operating system. We call this “forward compatibility.” To some extent, through design choices, the manufacturer of a new operating system can control the degree to which applications written primarily for the new operating system are compatible with applications written for the older one. If it is certain that an application will be written for the new operating system, the manufacturer of the new operating system has an incentive to make it less forwardly compatible, so that the old operating system is deprived of the positive externalities. On the other hand, if an application (or upgrade) compatible with the new operating system will be written only if some of its sales go to users of the old operating system, the manufacturer of the new operating system has an incentive to make it more forwardly compatible.

Taking into account these features of software markets, we model a durable goods monopolist selling goods with network externalities. We examine pricing and compatibility, as well as the incentive of a monopolist to introduce new goods. We also examine the possibility of entry in later periods, and its effect on the monopolist’s pricing.

## **2. The Benchmark Model**

We first consider the benchmark model of durable goods monopoly with network externalities. The primary example we have in mind is a computer operating system. Let a durable good be sold during two periods. The good is durable and lasts indefinitely after the first two periods. Potential consumers are distributed uniformly in  $[0, 1]$  according to their willingness to pay for the good. Each consumer can buy up to one unit of the good. Each consumer has the choice of buying in period 1, buying in period 2, or not buying the durable good. In the second period, the market (number of potential

customers) grows by  $g$  percent.<sup>7</sup> Consumers also benefit from network externalities that are proportional to the size of sales.

We assume that there is no resale (secondary) market for used operating system software. This is justified since often operating system software is pre-installed and not portable to another computer. Moreover, even when the operating system software is portable, it is often hard to uninstall from the original computer, so that the used software can be legally sold, and often there are legal restrictions to resale.<sup>8</sup>

Durable goods sold in every period create network externalities. In the back of our minds we have a model of complementary goods (software applications) that are more abundant and have higher sales as more of compatible durable goods are sold. For example, the software applications market may be modeled as monopolistic competition with free entry in a circular model of differentiated products as in Economides (1996b). For simplicity, we assume that network externalities are linear in sales. Moreover, when the same durable good is sold in every period, we assume that sales in either period contribute the same marginal network externality benefit.

A consumer of willingness to pay (type)  $a$  who buys in period 1 receives utility

$$u_1 = a - p_1 + k(d_1 + d_2b),$$

where  $d_i$ ,  $i = 1, 2$ , are the sales in the  $i$ th period,  $k > 0$  is the degree of network externalities,  $1 > b \geq 0$  is the discounting rate, and  $p_i$  is the nominal price in period  $i$ . Similarly a consumer who buys in period 2 receives utility (discounted to the first period)

$$u_2 = b[a - p_2 + k(d_1 + d_2)].$$

Notice that a consumer in period 1 receives the extra network benefit  $k(1 - b)d_1$  compared to a consumer in period 2. This reflects consumption in period 1 of network benefits created in period 1. Period 2 buyers get network externalities benefits in period 2 from period 1 sales but can only receive them and consume them in period 2.

Consumers with  $u_1 > u_2$  and  $u_1 > 0$  buy in period 1, while consumers with  $u_2 > u_1$  and  $u_2 > 0$  buy in period 2. Defining as  $a_{12}$  (by  $u_1 = u_2$ ) the consumer who is indifferent between buying the good in period 1 and 2, and as  $a_{20}$  (defined by  $u_2 = 0$ ) the consumer who is indifferent between buying the good in period 2 and not buying any good, we have the implicit definition of demand functions:<sup>9</sup>

<sup>7</sup> The buyers who appear in period 2 can only buy in period 2.

<sup>8</sup> We also assume no delayed effects of externalities since software applications are introduced together with the operating system.

<sup>9</sup> The marginal consumers are:  $a_{12} = (p_1 - p_2b)/(1 - b) - k d_1$ ,  $a_{20} = p_2 - k(d_1 + d_2)$ .

$$d_1 = 1 - a_{12}(d_1, d_2),$$

$$d_2 = a_{12}(d_1, d_2) - a_{20}(d_1, d_2) + g(1 - a_{20}(d_1, d_2)),$$

where the last term in the second period demand arises from the growth of the market. Notice that sales of periods 1 and 2 appear on both sides of the demand system. This is natural for goods with network externalities. Expressing sales in each period as functions of prices only, we have:

$$d_1 = (1 - b - p_1 + bp_2)/[(1 - b)(1 - k)],$$

$$d_2 = [p_1(1 - k - gk) - p_2(1 + g - bg - k - gk) + g(1 - b)]/[(1 - b)(1 - k)(1 - k - gk)].$$

Without loss of generality, we assume that costs are zero. The monopolist's profits,

$$\Pi^M = p_1 d_1 + bp_2 d_2,$$

are maximized at (nominal) prices

$$p_1^* = p_2^* = 1/2.^{10}$$

Three observations are in order. First, the equilibrium prices do not depend on the extent of network externalities, or even on the existence of network externalities, since they are independent of  $k$ . This result generalizes to an  $n$ -period durable goods monopoly, as shown in the appendix. Second, price in period 1 is not lower than the discounted (real) price in period 2,

$$p_1^* \geq bp_2^* .$$

This standard result in durable goods markets is valid in the presence of network externalities. Indeed, if price were higher in period 2, all period 2 customers would switch to buying in period 1 since they would be paying no more (since then  $bp_2^* - p_1^* > 0$ ) as well as benefiting from consuming the good and the associated externalities in period 1 as well.<sup>11</sup>

Third, prices are above marginal cost. In fact, nominal prices are at the single-period monopoly level. Thus, the Coase conjecture, that predicts price equal to marginal cost, fails.

---

<sup>10</sup> Realized profits are :

$$\Pi^{M*} = [1 + bg - k(1 + g)]/[4(1 - k)(1 - k - gk)].$$

<sup>11</sup> For further discussion see Cabral, Salant, and Woroch (1999).

Also note that, in a market without growth, sales in the second period are zero, i.e., for  $g = 0$ ,

$$d_2^* = 0.$$

That is, the monopolist, prefers to sell all his output in the first period.<sup>12</sup> This result generalizes to n-periods. In an n-period model without market growth, the monopolist has positive sales only in the first period as shown in the appendix. The intuition of this result is as follows. Low prices in later periods interfere with the ability of the monopolist to extract surplus in earlier periods. Also, selling in an earlier period is preferable for the monopolist to selling in a later period because of discounting. Thus, when there is no market growth in later periods, the monopolist sells all output in the first period.

### 3. New good introduction in durable goods monopoly

We now consider the case when the second period good differs from the first period good. This case models the introduction of new operating system software. As is customary in software markets, we assume that the first period good is no longer sold in the second period.

We assume that the second period good is only partially compatible with the older one. Thus, same period demand externalities differ in intensity from demand externalities across periods. This is a natural assumption since demand externalities arise out of applications that are compatible with the operating system. In every period, a consumer enjoys externalities from applications that are fully compatible with his operating system, as well as from applications that are only partially compatible with his operating system. We assume that externalities from fully compatible applications are proportional to same period sales. Positive externalities to a consumer of operating system  $i$  from partially compatible applications are proportional to sales of operating system  $j$  since these applications were written for operating system  $j$ . Moreover, we expect that the intensity of network externalities from partially compatible applications will be lower than the intensity network externalities from fully compatible applications.

We define as *backward compatibility* the degree of compatibility of the second period operating system with first period applications. We define as *forward compatibility* the degree of compatibility of the first period good with second period applications.

A consumer of willingness to pay (type)  $a$  who buys in period  $i$ ,  $i = 1, 2$ , receives utility

$$u_1 = a - p_1 + kd_1 + md_2b,$$

---

<sup>12</sup> As the appendix shows, then sales in the first period are  $d_1^* = 1/[2(1 - k)]$ .

$$u_2 = b(a - p_2 + nd_1 + kd_2),$$

where  $k$  is the degree of network externalities from sales in the same good,  $m$  is the intensity of network externalities to a good 1 consumer from sales good 2 (arising from partial forward compatibility), and  $n$  is the intensity of network externalities to a good 2 consumer from sales of good 1 (arising from partial backward compatibility). As before, given marginal consumers  $a_{12}$ ,  $a_{20}$ , we have the implicit definition of demand functions:

$$d_1 = 1 - a_{12}(d_1, d_2),$$

$$d_2 = a_{12}(d_1, d_2) - a_{20}(d_1, d_2) + g(1 - a_{20}(d_1, d_2)).$$

The fixed point of this mapping defines the demand functions in each period,<sup>13</sup>

$$d_1(p_1, p_2), d_2(p_1, p_2).$$

The monopolist's profits,  $\Pi^M = p_1 d_1 + p_2 d_2$ , are maximized at prices  $p_1^*$ ,  $p_2^*$ .<sup>14</sup>

Notice that, in contrast with the benchmark model where the good did not change across periods, in the present case of new good introduction, it is possible for the price to increase during the second period. This occurs when  $m$  is smaller than  $n$ ,

$$m < n,$$

*i.e.*, when the intensity of network externalities arising from weak partial forward compatibility is low compared to the intensity of network externalities arising from partial backward compatibility. This occurs when many of the applications written for the new operating system do not work (or work poorly) with the old operating system.<sup>15</sup>

---

<sup>13</sup> The demand functions are:

$$d_1 = [1 + bgm - b(1 - m) - k(1 + g) - p_1(1 - k - gk) + bp_2(1 - m - gm)]/[1 - b - k(2 + g - bg) + k^2(1 + g) + bm + bn - bmn(1 + g)],$$

$$d_2 = [(n - k + g(1 - b - k + n) + (1 - n - gn)p_1 - (1 - k + g - bg - gk)p_2)]/[1 - b - k(2 + g - bg) + k^2(1 + g) + bm + bn - bmn(1 + g)].$$

<sup>14</sup> Equilibrium prices are:

$$p_1^* = (2 - 2b - 4k - 2gk + 2bgk + 2k^2 + 2gk^2 + 2bm + bgm - b^2gm - bkm - bgkm + 2bn - bgn + b^2gn + bkn + bgkn - bmn - bgmn - bn^2 - bgn^2)/(4 - 4b - 8k - 4gk + 4bgk + 4k^2 + 4gk^2 + 4bm - bm^2 - bgm^2 + 4bn - 2bmn - 2bgmn - bn^2 - bgn^2),$$

$$p_2^* = (2 - 2b - 4k - 2gk + 2bgk + 2k^2 + 2gk^2 - m + 3bm + km + gkm + bm^2 + bgm^2 + n + bn - kn - gkn - bmn - bgmn)/(4 - 4b - 8k - 4gk + 4bgk + 4k^2 + 4gk^2 + 4bm - bm^2 - bgm^2 + 4bn - 2bmn - 2bgmn - bn^2 - bgn^2).$$

<sup>15</sup> It can be shown that the price difference,  $p_1^* - p_2^*$ , has the same sign as  $m - n$ , so that  $p_1^* < p_2^* \Leftrightarrow m < n$ . The proof is available from the author upon request.

The monopolist has strong incentives to maximize both backward and forward compatibility. It can be shown that the incentive to increase compatibility is proportional to the equilibrium demands, and therefore is positive.

$$d\Pi^{M^*}/dn = d\Pi^{M^*}/dm = b(d_1^*)(d_2^*) > 0.$$

Note that the monopolist has no incentive to create incompatibilities to disadvantage the consumers of period 1, since he benefits from profits of both periods. This result is likely to be different when the market structure is not monopoly.

#### 4. Duopoly in the second period

We now consider the evolution of monopoly to duopoly. We keep the market as a monopoly in the first period. We assume that in the second period, in addition to the existing durable good provided by the monopolist, a new durable good is introduced by an entrant. We assume that the new good is of a different quality level than the original one, and has only partial network externalities with the “old” durable good.

In the first period, only the monopolized good is sold. In the second period, there are two goods: the “old” durable good (good #2) provided by the old monopolist, and the “new” good introduced by the competitor (good #3). The monopolist’s good in period 2 is identical with the one in period 1. As earlier, we denote by  $k$  the degree of network externalities under full compatibility. Let  $m$  be the degree of network externalities realized to good 1 from the partial forward compatibility between the “new” good and the “old” good 2. Let  $n$  be the intensity of network externalities to a good 3 consumer from sales of good 1 or 2 (arising from partial backward compatibility). Figure 1 shows the operating systems and the degrees of compatibility. We also assume that the new good has a quality difference of  $q$  compared to the old one.

A consumer of willingness to pay (type)  $a$  who buys in period 1 receives utility

$$u_1 = a - p_1 + k(d_1 + d_2b) + md_3b.$$

The respective utilities to a consumer of willingness to pay (type)  $a$  (discounted to period 1) who buys in period 2 from the monopolist (good 2) or the entrant (good 3) are

$$u_2 = b[a - p_2 + k(d_1 + d_2) + md_3],$$

$$u_3 = b[a(1 + q) - p_3 + n(d_1 + d_2) + kd_3],$$

respectively.

We define the marginal consumer between goods 1 and 2 as  $a_{12}$ , the marginal consumer between goods 2 and 3 as  $a_{23}$ , and the consumer who is marginal between good 3 and not buying as  $a_{30}$ . We define implicitly demand functions:

$$d_1 = 1 - a_{12}(d_1, d_2, d_3),$$

$$d_2 = a_{12}(d_1, d_2, d_3) - a_{23}(d_1, d_2, d_3),$$

$$d_3 = a_{23}(d_1, d_2, d_3) - a_{30}(d_1, d_2, d_3).$$

As before, we find the fixed point of the above mapping to define the demand functions in each period:

$$d_1 = d_1(p_1, p_2), \quad d_2 = d_2(p_1, p_2, p_3), \quad d_3 = d_3(p_2, p_3).$$

The monopolist maximizes profits,  $\Pi^M = p_1 d_1 + p_2 d_2$ , while the entrant maximizes  $\Pi^E = p_3 d_3$ . Because the expressions are very complicated, we present here only the results for certain values of the parameters, that is, for  $n = k$  and as  $q$  tends to zero. That is, we present results for the case when the partial backward compatibility externality is the same for goods 2 and 3, and goods 2 and 3 do not have a significant quality difference. Nash equilibrium prices  $p_1^*$ ,  $p_2^*$ ,  $p_3^*$  and sales  $d_1^*$ ,  $d_2^*$ ,  $d_3^*$  are positive only when  $m > k$ , i.e., when the new product has very strong externalities arising from forward compatibility.<sup>16</sup>

We find that the entrant has strong incentives to maximize both backward and forward compatibility,

$$d\Pi^{E^*}/dn > 0, \quad d\Pi^{E^*}/dm > 0.$$

In contrast, the monopolist has a divided view toward compatibility. We find that

$$d\Pi^{M^*}/dn < 0, \quad d\Pi^{M^*}/dm > 0.$$

---

<sup>16</sup> Equilibrium prices and sales are:

$$p_1^* = (3 - 3b - 4k - 4gk + m + 3bm + gm + 3bgm)/[2(3 - 4k - 4gk + m + gm)],$$

$$p_2^* = 2(1 + g)(m - k)/(3 - 4k - 4gk + m + gm),$$

$$p_3^* = (1 + g)(m - k)/(3 - 4k - 4gk + m + gm),$$

$$d_1^* = 1/[2(1 - k)],$$

$$d_2^* = (1 + 4g - m - gm)/[2(1 - k)(3 - 4k - 4gk + m + gm)],$$

$$d_3^* = (1 + g)/(3 - 4k - 4gk + m + gm).$$

Thus, the monopolist wants the competitors' good to be compatible with its own, so it receives the network externalities (forward compatibility externalities) of the competitor's good but does not want its own good to be compatible with that of the competitor, so that network externalities accruing to the competitor because of backward compatibility are minimized.

## 5. Concluding Remarks

We find that a monopolist selling a durable good with network externalities, such as a computer operating system, in a finite period setting, is likely to set its price significantly above cost. Thus, we expect that the Coase conjecture will not hold in the operating systems market.

As is well known, the durable goods monopolist does not increase price over time. However, we find that introduction of new goods which have variable degrees of network externalities to older ones could lead the monopolist to increase price over time.

A monopolist facing the threat of future entry that would drive prices to marginal cost in future periods, will charge significantly less than a monopolist that does not face such threat. The extent of the threat depends on the degree of compatibility of the entrants' products with the products of the incumbent monopolist.

The entry of one competitor is successful (in the sense of bring positive profits to the entrant) only if the entrant's product has very strong externalities arising from forward compatibility. The entrant wishes to maximize both forward and backward compatibility, while the incumbent wants to maximize forward compatibility but minimize backward compatibility of the entrant's product.

*References*

- Economides, Nicholas, (1996a), "The Economics of Networks," *International Journal of Industrial Organization*, vol. 14, no. 6, pp. 675-699.
- Economides, Nicholas, (1996b), "Network Externalities, Complementarities, and Invitations to Enter," *European Journal of Political Economy*, vol. 12, (1996), pp. 211-232.
- Cabral, Luis, David Salant and Glenn Woroch (1999), "Monopoly Pricing with Network Externalities," *International Journal of Industrial Organization*.
- Bensaid, Bernard and Jean-Philippe Lesne (1996), "Dynamic Monopoly Pricing with Network Externalities," *International Journal of Industrial Organization*, vol. 14, pp. 837-855.
- Jonard, Nicolas and Eric Schenck (1998), "Open Standard Compatibility: Network Externalities Versus Product Differentiation," mimeo.
- Katz, Michael and Carl Shapiro, (1986), "Technology Adoption in the Presence of Network Externalities," *Journal of Political Economy*, vol. 94, pp. 822-841.
- Mason, Robin (1998), "Network Externalities and the Coase Conjecture," mimeo.
- Morch von der Fehr, Nils-Henrik and Kai-Uwe Kuehn, (1996), "Coase versus Pacman: Who Eats Whom in the Durable-Goods Monopoly," *Journal of Political Economy*, vol. 103, no. 4, pp.785-812.
- Salant, Stephen, Mark Bagnoli and Joseph Swierzbinski, (1989), "Durable-Goods Monopoly with Discrete Demand," *Journal of Political Economy*, vol. 97, no. 6, pp. 1459-1478.

*Appendix*

**n-Period Durable Goods Monopoly With Network Externalities**

Let a durable good be sold in each of  $n > 2$  periods. A consumer who buys in period  $j, j = 1, \dots, n$ , receives utility

$$U_j = b^{j-1}(a - p_j + k(d_1 + d_2 + \dots + d_j) + k(b^j d_{j+1} + \dots + b^{n-1} d_n)).$$

The marginal consumer between goods  $j$  and  $j+1$ ,  $a_{j,j+1}$ , is defined by

$$U_j = U_{j+1}.$$

The solution of

$$U_n = 0,$$

$a_{n,0}$ , defines the marginal consumer who is indifferent between buying good  $n$  and not buying at all.

Demand is defined by

$$d_1 = 1 - a_{1,2}$$

$$d_j = a_{j-1,j} - a_{j,j+1}, \quad j = 2, \dots, n-1$$

$$d_n = a_{n-1,n} - a_{n,0}.$$

Using the definition of  $a_{j,j+1}$ , we have:

$$(1 - b) a_{j,j+1} = p_j - b p_{j+1} - (1 - b)k (d_1 + d_2 + \dots + d_j).$$

Since

$$d_1 + d_2 + \dots + d_j = 1 - a_{j,j+1},$$

it follows that

$$a_{j,j+1} = [p_j - b p_{j+1} - (1 - b)k] / [(1 - k)(1 - b)], \quad j = 1, \dots, n-1$$

and

$$d_1 = 1 - [(p_1 - b p_2) - k(1 - b)] / [(1 - k)(1 - b)] = 1 / (1 - k) - (p_1 - b p_2) / [(1 - k)(1 - b)],$$

$$d_j = (p_{j-1} - (1 + b)p_j + b p_{j+1}) / [(1 - k)(1 - b)], \quad j = 2, \dots, n-1,$$

$$d_n = (p_{n-1} - p_n) / [(1 - k)(1 - b)].$$

Maximizing profits

$$\Pi^M = \sum_j b^{j-1} p_j d_j$$

with respect to  $p_j$  requires

$$d\Pi^M/dp_1 = d_1 + (bp_2 - p_1)/[(1 - k)(1 - b)] = 0,$$

$$d\Pi^M/dp_j = b^{j-1} \{d_j + [p_{j-1} - p_j(1 + b) + bp_{j+1}]/[(1 - k)(1 - b)]\} = 0, \quad j = 2, \dots, n-1,$$

$$d\Pi^M/dp_n = b^{n-1} \{d_n + (p_{n-1} - p_n)/[(1 - k)(1 - b)]\} = 0,$$

which is solved by

$$p_j = 1/2, \text{ all } j, \quad d_1 = 1/[2(1 - k)], \quad d_j = 0, j \neq 1.$$

It is easy to show that this solution is unique.

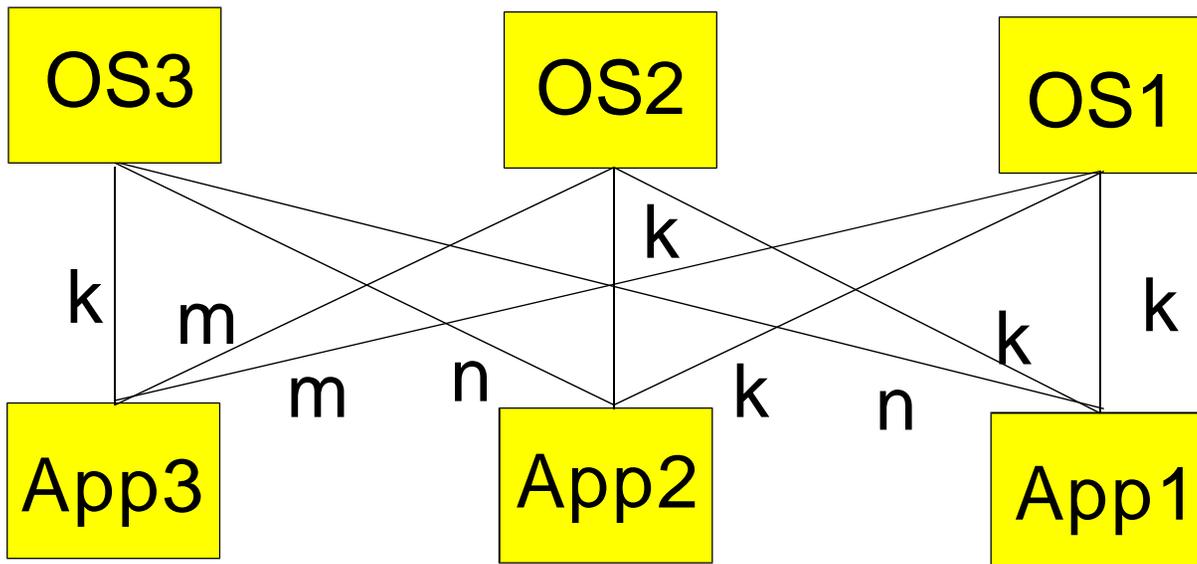


Figure 1: We denote the monopolist's operating system in period  $i$ ,  $i=1, 2$ , as  $OS_i$ . The entrant sells  $OS_3$  in period 2. The letters  $k$ ,  $m$  and  $n$  denote full, forward, and backward compatibility with the various applications.