Open(?) AI

Matthew Leisten

Federal Trade Commission*

This draft: March 3, 2025

Abstract

I provide an economic paradigm of "open" AI models based on four features. Open AI models are (1) free to use; (2) free to inspect; and (3) free to modify; and (4) all derivative applications should also have these features. Next, I study a model of AI development that captures the topology of general-purpose models and niche applications that may be close to or far from other niche applications. A "founder" builds a general-purpose foundation model. In the open paradigm, a community of developers each decides whether to develop specialized applications for their own purposes or to use another developer's application. In the closed paradigm, the founder centrally builds the applications and sells access. I compare both to the social planner's optimum in terms of the number of applications, the degree of specialization of these applications, and the investment in the overall quality of the foundation model. Notably, the open paradigm features countervailing distortions—a "reverse Spence" distortion and a "free riding" distortion—that may lead to too little or too much specialization and too little or too much quality. The closed model yields the optimal quality investment but too many applications that are not specialized enough.

Keywords: Artificial intelligence, innovation, competition, open source

JEL Codes: L15, O36, D40

*Website: mleisten.github.io and email: mleisten@ftc.gov. The views expressed in this paper are those of the author and do not necessarily reflect those of the Federal Trade Commission or any individual Commissioner. Thank you to Joshua Gans for his discussion. Thank you to Florian Ederer, Matt Panhans, Frank Pinter, and Daniel Stackman.

1 Introduction

Artificial intelligence (AI) has the potential to transform the economy globally. At the same time, it presents unique challenges for competition and intellectual property policy. It is unclear how the assignment of property rights over AI models—that is, the "openness" of AI—will affect the competitiveness of AI software markets and the amount, direction, and variety of technological improvements in AI. This uncertainty persists despite the sudden ubiquity of AI, in part because the economics of AI may be distinct from other software markets (Gans [2024]).

Case in point: advocates have zealously pushed for openness in AI models while at the same time acknowledging openness is poorly defined. A National Telecommunications and Information Administration Request for Comments on this subject received 362 responses. One comment from AI researchers at Stanford and Princeton [Nelson et al., 2024] states:

> Open foundation models have distinct benefits relative to closed models. Foundation models with widely available weights present unique benefits that cannot be provided by closed models. Limiting the distribution of model weights would diminish the significant benefits provided by open models related to innovation, competition, transparency, and scientific discovery.

At the same time, the Open Source Initiative, as of November 2024, was still actively working to define "open source AI". Their website[1] notes (emphasis present on website):

> **Why do we need a new Definition of Open Source just for AI?** [...] More than two decades passed between the GNU Manifesto and the writing of the Open Source Definition. For AI we cannot wait decades to produce a new document.

In this paper, I propose an economic definition of openness in the context of AI. My "open AI" paradigm hews closely to characteristics of leading "open" AI models like Meta Llama and AI repositories like HuggingFace. After presenting my definition, I study a model of

[1] See https://opensource.org/ai/drafts/the-open-source-ai-definition-1-0-rc1. Accessed 10/19/2024.

spatial competition between AI products. This model can capture both, the open paradigm and a closed-model paradigm. I solve the model and compare the equilibria to a social planner's solution. I thereby quantify distortions in the amount and direction of innovation: the quality of models, their variety, their degree of specialization, and their availability (i.e., prices).

My open-model paradigm has four characteristics: first, open AI models should be free to use. Second, they should be free to inspect. That is, the parameters, weights, and overall model architecture should be available to developers. Third, they should be free to modify. Developers can freely adapt an AI model by training it on additional data for a specific purpose, a process known as "fine-tuning." Finally, these three characteristics should transfer to all derivative products. That is, if a model is open, a model built by fine-tuning this model should also be open.

My model is of spatial competition between AI models. There are two types of AI model. *Foundation models* are general purpose models typically only built by large enterprises with access to vast amounts of data, computing power, and engineering talent. *Applications*, on the other hand, are typically created by fine-tuning foundation models on a relatively small amount of additional application-specific data. Applications may be similar to each other—for instance, voice recognition and voice sentiment—or very distinct—for instance voice recognition and image generation.

I capture the topology of general-purpose and niche models using the "Salop disk" from Bar-Isaac et al. [2023]. In this model, consumers (or, in my case, developers) are located around the edge of a Salop circle. Products (in my case, models) can locate on the interior. Developers pay a "vertical" travel cost to move toward the center of the circle and a "horizontal" travel cost to traverse a ring inside the circle toward the location of the application. General purpose AI models are closer to the center of the circle, while niche applications are near the edge.

In the open paradigm, a "founder" (for instance, Meta) decides how much to invest in a foundation model. The foundation model exists at the center of the Salop disk. Next, developers freely decide whether to fine-tune the model and by how much and in which

direction (i.e., deciding the polar coordinates of the application). Fine-tuning is costly. Given the set of fine-tuned applications that exist in equilibrium, developers decide which application to use. Finally, the founder sells a complementary service to the developers who use any of the models.

In the closed paradigm, the founder decides all of these factors: the quality of the foundation model, the number and location of the applications created by fine-tuning the model, and, additionally, the price of access to the model or any of its applications. Developers, still located around the circle, decide which model to buy access to, and, afterward, whether to buy the complementary service from the founder.

Finally, I consider a social planner who chooses the quality of the foundation model, the number and location of the applications, and the allocation of applications to developers to maximize the integrated developer surplus net of the cost of development.

I highlight two countervailing distortions in the open paradigm and one distortion in the closed paradigm. In the closed paradigm, we have the standard Spence [1975] distortion: the founder designs products based on the tastes of the developer who is marginal between one application and another. Because the marginal developer is less aligned with any particular application than all of the inframarginal developers (that is, the angle between marginal developer i and application j is larger than between inframarginal developer k and application j), and the returns to fine-tuning are greater the more aligned the developer is, the monopolist *under-specializes* products in equilibrium.

The open paradigm has a *reverse Spence distortion*: the designer of the application is a developer themself, and that developer is *perfectly aligned* with the application in equilibrium, so they are the *most inframarginal* of all developers. This leads to too much specialization. The developer creates a product that is extremely tailored to their own niche, though other nearby developers would prefer the product to be more general purpose. On the other hand, there is also a *free riding* distortion: the fine-tuning developer does not internalize the benefits nearby developers derive from using this application, and therefore the developer provides too little effort in fine-tuning the model. Ultimately, this may lead to too little specialization in equilibrium (as in the closed paradigm), too much specialization,

or just enough.

I next show that there are too many applications in equilibrium both the open and closed paradigms, but for different reasons. In the closed paradigm, this is a direct consequence of the Spence distortion: if products are underspecialized, they are cheap to build and the founder builds too many of them. In the open paradigm, there are too many applications as a consequence of the inefficient replication of fixed costs. This problem may be worse or less bad than in the closed model paradigm depending on whether developers can coordinate their actions to minimize redundancy of their models.

I also show that the closed paradigm (but not the open paradigm) leads to the exact right quantity of investment into the foundation model itself, while the open paradigm, perhaps surprisingly, can lead to too little *or too much* investment in the quality of the foundation model. Finally, I show that the founder, if offered the choice, may find it profitable to build an open model rather than a closed model if investment is costly, if there is a highly profitable market for complementary services to AI models, and if there are high returns to specializing applications.

1.1 Contributions and related literature

First, this paper contributes to the theoretical literature on open source software. Previous papers have examined the decision whether to open source or not [Casadesus-Masanell and Llanes, 2011], the dynamics of open source contributions (Johnson [2001], Athey and Ellison [2014]), competition between firms that contribute to open-source pools [Kumar et al., 2011] and competition between open-source and closed-source rivals [Llanes and de Elejalde, 2013]. This paper expands on this literature by focusing on open sourcing in AI, which presents a new set of challenges and institutional features. In particular, my model departs from the previous literature quantifying the distortions in the *direction of innovation* induced by open- and closed-sourcing. As a result, this paper also speaks to the broader literature on the relationship between competition and the direction of innovation (Jaske [2022], Bryan and Lemus [2017], Callander and Matouschek [2022]). Whereas those papers mostly focus on the decision to focus on incremental progress versus "moonshots",

my paper focuses on general-purpose versus niche improvements and introduces a notion of open sourcing into a spatial model of innovation. Notably, Callander and Matouschek [2022] incorporate spatial competition along a Hotelling line to consider the novelty of innovations.

Second, this paper contributes to the academic and policy debate on the right competition standards for artificial intelligence. This is a nascent literature with few papers that are specifically focused on AI and competition. For a survey of what exists so far, see Gans [2024]. Most papers in this literature review predate the recent explosion in AI and therefore are not about AI *per se*. Instead, they focus on the economics of data and the potential economies of scope and scale in data.

Third, this paper presents an application of the literature on product design and spatial competition. Forebears in product design include Johnson and Myatt [2006], Callander and Matouschek [2022] and most directly, Bar-Isaac et al. [2023]. These papers study niche versus mass-market products. The distinction is generally a rotation of demand, with steeper demand curves corresponding to nichier products. I extend Bar-Isaac et al. [2023] to consider spatial interaction between products for a multi-product monopolist (in the closed paradigm) and introduce crowd-sourced alterations to products into a monopolistic competition framework (in the open paradigm).

2 Model

My model is an adaptation of Bar-Isaac et al. [2023], which extends the Salop [1979] circle model to allow products to be located on the interior of the circle. Products closer to the center of the circle are more general-purpose.

2.1 Agents

There is a mass of 2π developers indexed by i distributed uniformly around a unit Salop circle. While developers are located along the perimeter of the circle, products may be in the interior of the circle. I refer to the circle and its interior as a "Salop disk." $\theta_i \in [0, 2\pi]$ be

the location (angle) of i. There is also a *founder* who has the ability to make a *foundation model* with quality V and sell a complementary service that is worth S at price p_S if and only if used in conjunction with an AI model.

2.2 Consumption payoffs and technology

A developer located at θ_i obtains the following utility from application $j \in J$, where J is the set of all applications, located at polar coordinates (z_j, η_j):

$$u_{ij} = V - p_j - \underbrace{r(1 - z_j)}_{(1)} - \underbrace{|\theta_i - \eta_j| z_j}_{(2)} - \underbrace{\mathbf{1}[q_{iS} = 1](S - p_S)}_{(3)}$$

V is the quality of the foundation model that underpins this model, and p_j is the price of access to the model (in the event that the model is not open). Term (1) captures the "vertical" travel cost of moving from the perimeter of the circle to a ring inside the Salop disk that contains product j. All consumers, regardless of their location, pay this travel cost. Term (2) captures the "horizontal" travel cost of traversing this ring if $\theta_i \neq \eta_j$, i.e., the application is not fully aligned with the developer. The less well-aligned the product and the developer are, the greater this cost. It is generally smaller if z_j is smaller, holding fixed θ_i and η_j, which captures the fact that products closer to the center of the Salp disk have broader appeal. Terms (1) and (2) together capture the trade-off between general-purpose and niche products as a rotation of the demand curve (see Johnson and Myatt [2006]). The trade-off between the vertical cost and the horizontal cost is parameterized by r in this setting. The larger r is, the greater the returns to specialization. Therefore, while I have normalized the radius of the Salop disk to equal one, r captures the intuitive notion of "expanding" the disk as applications become more specialized.

Term (3) is the additional utility (net of price) developer i gets from consuming the complementary service. Developers may also consume the foundation model F, located at the center of the circle:

$$u_{iF} = V - p_F - r + \mathbf{1}[q_s = 1](S - p_S)$$

or they may consume neither the foundation model nor an application, earning payoff 0, with no scope to gain utility from the complementary service.

The technology of developing an application involves two steps: first, a foundation model of quality V is built at cost $\frac{gV^2}{2}$. Next, the application is built by fine-tuning the foundation model by amount z at $\frac{cz^2}{2}$. Here, g and c are parameters capturing the costs of quality and specialization, respectively. Note that fine-tuning the model is a continuous choice, corresponding to feeding the foundation model more and more data that is specialized for the application. Fine-tuning is costly, but it improves the quality of the application for developers who are well-aligned with the application.

I assume $r < \pi$. This ensures that the maximal cost of traveling to a fully specialized application (traveling halfway around the perimeter of the Salop disk) is at least as large as the cost of traveling to the foundation model at the center of the disk. This is a natural assumption that is consistent with the notion that fine-tuning a foundation model does not make it better for everyone.

Overall payoffs are functions of the consumption utility and the cost of development. In the open model, there are no prices and the cost of fine-tuning is borne by the developers. In the closed paradigm, there are prices and the cost of fine-tuning is borne by the founder. Therefore, I explicitly write out payoffs in the following subsections after laying out the timing of the model. An overview comparing the open and closed paradigms is provided in Figure 1.

A final note: in general, equilibrium involves a finite number of products, $N = |J|$. I abstract from integer constraints on N to simplify the analysis.

2.3 Open paradigm

The open-paradigm game proceeds in six stages:

Table 1: Outline of paradigms

Product	Open paradigm	Closed paradigm
Foundation model	Created by founder Offered for free	Created by founder Sold by founder
Applications	Fine-tuned by developers Freely used by developers	Fine-tuned by founder Sold by founder
Complementary service	Sold by founder	Sold by founder

1. The founder chooses the foundation model's quality V.
2. Developers simultaneously choose the amount of fine-tuning $z_i \in [0, 1]$ and the direction of their fine-tuning $\eta_i \in [0, 2\pi]$. The resulting set of products is then $J \cup F$, where $J = \{(z_j, \eta_j)\}_{\{j:i \text{ s.t. } z_i>0\}}$ are the fine-tuned applications.
3. Developers simultaneously decide which model to use out of $J \cup F \cup 0$ where 0 is no model. Let $q_{ij} = 1$ indicate i chose application j and $q_{iF} = 1$ if i uses the foundation model.
4. The founder observes developers' actions and sets the price for the complementary service, p_S.
5. Developers choose whether to purchase the complementary service. Binary variable $q_{iS} = 1$ if i buys the complementary service.
6. Payoffs are realized.

In the open paradigm, developer payoffs are $u_{ij} - \frac{cz_i^2}{2}$. That is, they are the consumption payoffs from consuming $j \in J \cup F \cup 0$ minus the cost of fine-tuning by amount z_i.[2] The

[2]Note the notation: if i consumes product j, z_j is the amount of fine-tuning of the model i consumes, while z_i is the amount of fine-tuning developer i undertakes. These may be different if i does not consume their own model.

founder's payoff is

$$\Pi = \underbrace{Q_S p_S}_{\substack{\text{revenues} \\ \text{from service}}} - \underbrace{\frac{gV^2}{2}}_{\substack{\text{cost of} \\ \text{foundation model}}}$$

where $Q_S = \int q_{iS}\, di$. This is the revenue from selling the complementary service minus the cost of developing the foundation model with quality V.

I look for subgame perfect equilibrium. This means:

1. Developers buy the complementary service if it is rational to do so, given their choice of application and the price of the price of the complementary service;
2. p_S maximizes the founder's profits given developers' choices;
3. Developers choose the model out of the set of equilibrium models $J \cup F \cup 0$ that maximizes their continuation payoff given rational choice in (1) and (2);
4. Developers' choices of fine-tuning (η_i, z_i) maximize their payoff given other developers' choices and rational behavior in (1), (2) and (3). These choices generate the available set of equilibrium models J;
5. V maximizes the founder's profits given the developers' responses.

2.4 Closed paradigm

Under the closed paradigm, the founder is the only agent able to fine-tune the foundation model. Furthermore, the founder sells access to the foundation model or any applications it develops. Developers obtain the same consumption utility, u_{ij}, from consuming model j. The timing is as follows:

1. The founder chooses the foundation model's quality V and the set of applications $J = (z_j, \eta_j, p_j)$.
2. Developers decide which product to buy. Each developer only consumes one product out of the set $J \cup F \cup 0$. Write $q_{ij} = 1$ if i consumes j.
3. The founder observes developers' choices and chooses p_S.

4. Developers decide whether to purchase the complementary service ($q_{iS} = 1$).

5. All payoffs are realized.

The founder's payoffs now include revenues from selling access to the models:

$$\Pi = \sum_{j \in J} \int q_{ij} p_j di - \frac{c z_j^2}{2} - p_S Q_S - \frac{g V^2}{2}$$

while developers simply obtain u_{ij}. In this case, equilibrium is as follows:

1. Developers' choices of q_{iS} maximize their utility given p_S and their choice q_{ij};

2. p_S maximizes the founder's payoffs given all other strategic decisions;

3. Developers choose the application that maximizes their utility: u_{ij} given rational expectations over p_S and q_{iS};

4. The founder chooses V and (z_j, η_j, p_j) to maximize profits given (1) through (3).

I make one restriction: the founder sets $z_j = z_{j'}$ for all $j, j' \in J$. That is, the founder chooses a uniform degree of specialization. This precludes the founder from price discriminating by offering a combination of general-purpose and niche products to the same developers. As I show later, the solution to both the open paradigm and social planner's problems feature this symmetry ($z_j = z_j' \forall j$) without having to make an assumption, so limiting the closed paradigm to having symmetry allows for easier comparisons.

2.5 Developer surplus and the social planner's problem

Developer surplus is computed by integrating over the population:

$$DS = \int \sum_{j \in J \cup F} q_{ij}(u_{ij} + q_{iS}(S - p_S))\, di$$

The social planner chooses V, $J = (z_j, \eta_j)$, an allocation of developers to products in $J \cup F \cup 0$ given by binary $q_{ij} = 1$ if i consumes j, and an allocation of developers to

the complementary service, q_{iS}, to maximize developer surplus net the cost of building the foundation model and applications:

$$\max_{V,(z_j,\eta_j),q_{ij},q_{iS}} DS - \sum_j \frac{cz_J^2}{2} - \frac{gV^2}{2}$$
$$\text{s. to } z_j \in [0,1],\ \eta_j \in [0,2\pi],\ V \geq 0$$

This is a complicated, high-dimensional problem involving arbitrarily many products located around the Salop disk and an allocation of each of these products to the developers.

I show in the Appendix a series of lemmata that simplify the problem considerably. These lemmata essentially show that the optimum involves symmetry across products: $z_j = z_{j'},\ \forall j, j' \in J$, and that products will be spaced evenly around the Salop disk, serving non-overlapping segments of developers (up to integer constraints on the number of products), and no developer consumes F or O. Ultimately, the problem can be reformulated as:

$$\max_{V,z,N} 2N \int_0^T V - r(1-z) - tz\ dt - N\frac{cz^2}{2} - \frac{gV^2}{2}$$
$$\text{s. to } 2\pi = 2NT; z \in [0,1], V \geq 0.$$

Here, the social planner decides how many (evenly spaced) products to offer, at what (symmetric) degree of specialization, and the quality of the foundation model and all of its derivative applications, V.

3 Results and discussion

I present the entirety of the results in a single Theorem:

Theorem 1. *The equilibrium set of models and their characteristics, including prices where applicable, are summarized in Table 2.*

Proof: See appendix.

Table 2: Equilibrium characteristics

	Open	Closed	Social
z	$\min\{1, \frac{r}{c}\}$	$\min\{1, \frac{4r^2}{9c}\}$	$\min\{1, \frac{8r^2}{9c}\}$
$N\ (z<1)$	$[\frac{2\pi}{r}, \frac{4\pi}{r}]$	$\frac{3\pi}{r}$	$\frac{3\pi}{2r}$
$N\ (z=1)$	$[\frac{2\pi}{c}, \frac{4\pi}{c}]$	$\frac{2\pi}{\sqrt{c}}$	$\frac{\sqrt{2}\pi}{\sqrt{c}}$
$V\ (z<1)$	$r - \frac{r^2}{2c}$	$\frac{2\pi}{g}$	$\frac{2\pi}{g}$
$V\ (z=1)$	$\frac{c}{2}$	$\frac{2\pi}{g}$	$\frac{2\pi}{g}$
$\bar{S}\ (z<1)$	$\frac{g(2rc-r^2)^2}{16c^2\pi}$	$r - \frac{4r^3}{27c} - \frac{\pi}{g}$	$r - \frac{8r^3}{27c} - \frac{\pi}{g}$
$\bar{S}\ (z=1)$	$\frac{gc^2}{16\pi}$	$\sqrt{c} - \frac{\pi}{g}$	$\frac{\sqrt{c}}{2} - \frac{\pi}{g}$

I now compare each of these equilibrium quantities across the three different paradigms.

3.1 The direction of innovation, z

In this model, z captures the *direction* of innovation: whether applications are general-purpose or niche. The closed model clearly yields z_{closed} weakly lower than the social optimum (and strictly so if $8r^2 < 9c$.) This is the classic Spence [1975] distortion: the marginal consumer of application j, in this case, is the consumer i who is farthest away from j out of all consumers who consume j. That is, if $q_{ij} = 1$ and $q_{kj} = 1$, then $|\theta_i - \eta_j| \geq |\theta_k - \eta_j|$. Differentiating the utility function yields

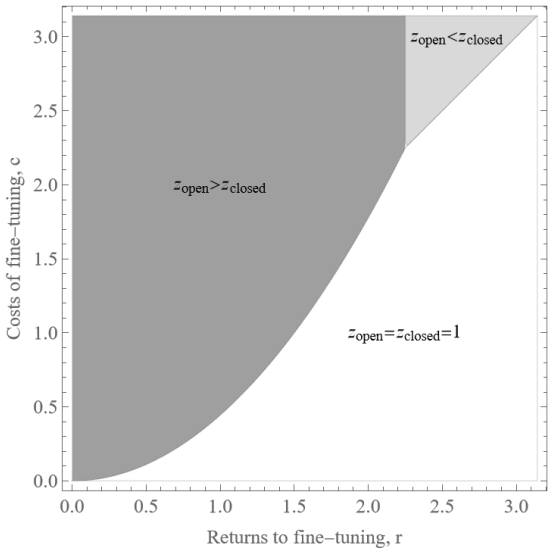
$$\frac{\partial u_{ij}}{\partial z} = r - |\theta_i - \eta_j|$$

which is decreasing in $|\theta_i - \eta_j|$, so marginal consumer i likes z less than all of the infra-marginal consumers. Therefore, the founder *underspecializes* by catering to the tastes of this marginal consumer.

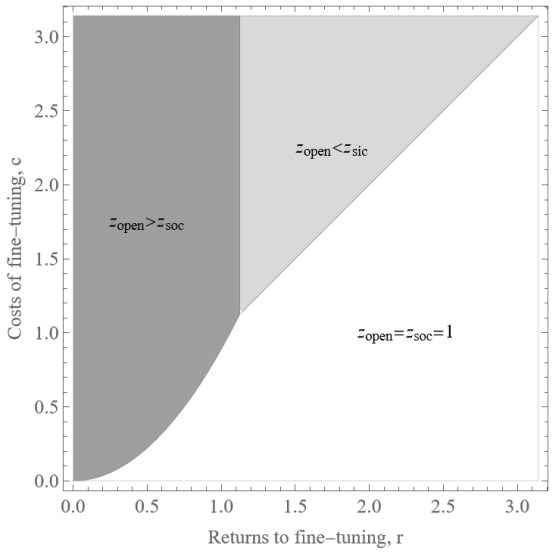
The open paradigm, on the other hand, can yield applications that are either too general-purpose or too niche. This is because there are two countervailing distortions. The first is

Figure 1: Comparison of z

(a) Open vs closed paradigm



(b) Open paradigm vs social planner



a *reverse Spencean* distortion. The designer of application j is the developer with $\theta_i = \eta_j$. In other words, developers who fine-tune consume their own application. Therefore, out of the set of developers who consume j, the designer—and therefore the developer whose tastes dictate z_j—is the *most inframarginal* one. This is the exact opposite of the closed paradigm (and that of standard monopoly) and leads to *overspecialization*. On the other hand, there is a *free-riding* distortion: the developer who fine tunes application j obtains utility u_{ij} from doing so and bears the cost $\frac{cz_j^2}{2}$. Meanwhile, other developers nearby will also consume j but do not bear any of the cost of fine-tuning. This means j under-invests in fine-tuning to the extent it benefits j's immediate neighbors. That is, the free-riding distortion leads to *underspecialization*. The overall effect relative to the social optimum depends on the parameters of the model.

I plot the regions in (r, c) space where z is larger (smaller) for open versus closed in Figure 1, and for open versus the social planner in Figure 1. In this figure, I assume that s is very large, so the founder finds entry profitable for all (r, c).

Corollary 1. *If* $r < \min\{c, \sqrt{\frac{9c}{8}}\}$, *then* $z_{open} > z_{soc}$. *If* $r > c$ *and* $r < \sqrt{\frac{9c}{8}}$, *then* $z_{open} < z_{soc}$. *Otherwise,* $z_{open} = z_{soc}$.

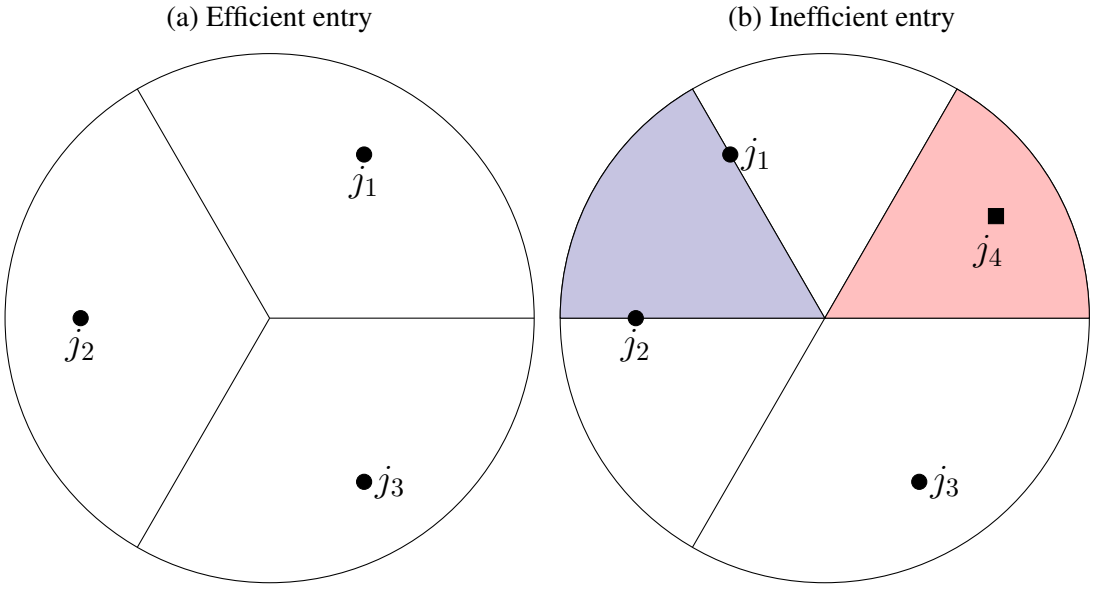
3.2 Product variety, N

Unlike z, N is unambiguously too high in both, the closed and open paradigm. However, it is too high for different reasons. Under the closed paradigm, N is too high as a consequence of the same Spencean distortion that led to z too low. Given z is too low, the fixed cost of creating another application, $\frac{cz^2}{2}$, is also too low, so the founder builds too many applications relative to the social optimum.

One way to conceive of this is that the applications that are "fine-tuned" by the founder are not, in a strict sense, fine-tuning. Rather than investing in creating entirely new iterations of the foundation model, the founder invests in making it easy for the foundation model to learn through its interactions with the user. That is, instead of a closed model like Claude having a variant "Claude for writing sonnets", Anthropic may pay some cost (equivalent to $N\frac{cz^2}{2}$) to make it easy for users to interact with basic Claude, and, within a few prompts, be able to guide it towards writing sonnets in the style the user wants.

N is also too high in the open paradigm. In this case N is also indeterminate. Because developers move simultaneously, they may not be able to coordinate spacing out applications around the Salop disk to minimize redundancy. Figure 2 shows equilibria with zero redundancy and some degree of redundancy. Both are equilibria because eliminating any single product would leave some developer who would like to fine-tune a new application, but the high-redundancy case has very few such developers (and many developers who could actually be served by two different applications that exist). If developers decide whether to fine-tune sequentially, then they would be better able to coordinate and space fine-tuning evenly around the Salop disk. Note that N is lower than the closed paradigm if developers can coordinate, but if developers are maximally uncoordinated, N is higher than in the closed case. Even with coordinated entry, the reason N is too high in the open paradigm is different than in the closed paradigm. Here, N is too high for a classic reason: inefficient replication of fixed costs, as in Salop [1979].

Figure 2: Comparison of efficient and inefficient entry



Notes: Left panel shows efficient entry with $N = 3$ products. Developers who fine-tune are evenly spaced, with mass $\frac{\pi}{3}$ of developers on either side who prefer to free-ride rather than fine-tune their own product. The right panel shows when developers are not evenly spaced. The same mass of $\frac{\pi}{3}$ developers on either side prefer to free-ride rather than fine-tune, but there is a region (shaded in blue) where there are two products the developers would rather free-ride by consuming than fine-tune their own model, and an additional region (shaded in red) where, without adding a fourth application (j_4), developers would rather fine-tune than consume an existing application.

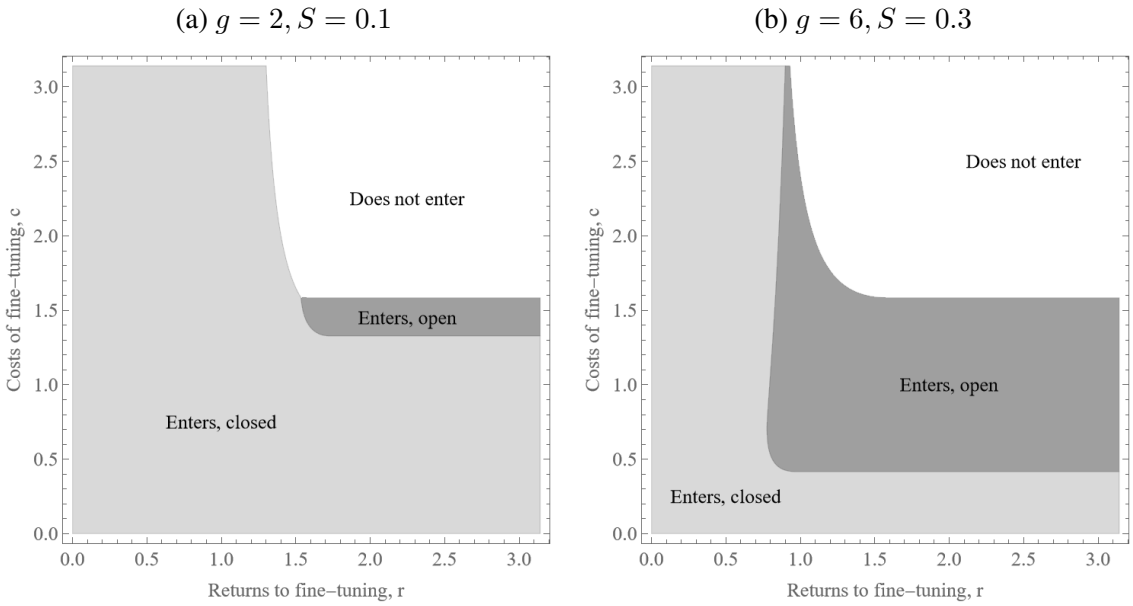
3.3 Amount of innovation, V

Whereas z captures the direction of innovation (specialized versus general purpose), V captures vertical product quality, or the amount of innovation. Here, the closed paradigm leads to the socially efficient level of V. This is primarily a consequence of the way V enters developers' utility functions: every developer values V the same amount, and every developer consumes some model that depends on V in the same way, so long as the foundation model is built in the first place. Therefore, by increasing V, the founder is able to recapture all of its investments by increasing prices accordingly. This would not be true—and there would be distortions in V—if there was scope for vertical differentiation (e.g., developers cared about $\gamma_i V$, where γ_i is the developer's preference for quality). Future extensions may explore these cases. Still, the general notion that aligning the founder's incentive to invest by allowing the founder to extract profits directly from the model and all applications based on the model is intuitive.

Corollary 2. *There exists $\bar{g} > 0$ such that, if $g > (<)\bar{g}$, $V_{closed} = V_{soc} < (>)V_{open}$. $\bar{g}$ is weakly decreasing in r and strictly decreasing in c. It is strictly decreasing in r if $r < c$.*

V is in general not equal to the social optimum under the open paradigm. The founder chooses V, but its profits only depend on V to the extent V is high enough to get buy-in from developers. That is, V solves a threshold problem: the utility of the fine-tuning developers, $u_{ij} = \arg\max_z V - r(1-z) - \frac{cz^2}{2}$ must be greater than zero. So long as V is high enough to ensure this, all developers will consume an application and the founder can sell the complementary service at price $p_S = S$. This generally provides weaker incentives to invest in V, though there are parameters for which V is greater in the open paradigm than in the closed paradigm. Specifically, if s and g are both very high, the closed paradigm optimally invests little in V (as g being high makes building a high-quality foundation model expensive), whereas the open paradigm may require a high V to induce developer buy-in, but s is so high that it is worth this investment in order to recoup profits on the complementary service. This means that the open model founder over-invests in V.

Figure 3: Founder's optimal entry decision



3.4 Profitability threshold for entry, $\bar{S}$

The final two rows of Table 2 report $\bar{S}$, the lowest value S can take such that the founder finds entry into the market profitable (or, in the case of the social planner, the value $\bar{S}$ such that if and only if $s \geq \bar{S}$ there are social gains from creating the foundation model). We can also endogenize entry by considering a founder who has the choice between entering by creating an open foundation model, entering by creating a closed foundation model and a set of applications, or not entering at all.

Figure 3 provides the regions in (r, c) space where the founder would choose each of these options. The left-hand panel of Figure 3 presents a case where g and s are relatively low, while the right-hand panel presents a case where g and s are relatively high. In both panels, the ratio of s to g is the same. Increasing this ratio—specifically increasing s while maintaining a fixed g—would broadly increase the profitability of entry in both the open and closed paradigms, and the shaded regions highlighted in the figure would grow.

Overall, the closed model dominates so long as r and c are not too large. If r and c grow, it becomes both more lucrative for developers to fine-tune more (r grows) and more costly for

the founder to fine-tune (c grows), thereby rendering opening up the model more profitable. This is especially true if g and s are large, per the reasoning I provided in the previous subsection.

4 Conclusion

This paper lays out a simple model of spatial competition in AI models that tractably allows me to study open- and closed-model AI paradigms. As alluded to in the introduction, there are paradigms that lie between my notions of open and closed. For instance, a foundation model can be open, but its derivative applications can be closed. In this case, developers would fine-tune the foundation model with the expectation of selling this fine-tuned product. In this case, the Spence [1975] distortion would return as in the closed paradigm, but the founder's incentive to invest in the quality of the foundation model would still be a "just enough to induce buy-in" threshold as in the open paradigm.

Future work may explicitly model other paradigms such as the one proposed above. This paper also abstracts from the question of open data: other notions of openness in AI ask for the training data used to create the foundation model to be open to all. Finally, this paper assumes that the founder is a monopolist. There is clearly the potential for fruitful work exploring competition between foundation models and their ecosystems of derivative model and products. This competition may occur between closed models, between open models, or between open and closed models.

Still, this paper shows that the comparison between open and closed paradigms is not trivial. Even basic intuition —such as the notion that the incentive to invest in the foundation model is necessarily stronger in the closed model than in the open model—does not hold in certain regions of this model's parameter space. This comes despite the model being relatively parsimonious, with only four parameters. Overall, the main lesson is that we should pay attention to these key parameters—the returns to training data, the returns to specialization, and the profitability of complements—and try to measure them empirically before assessing the desirability of open versus closed AI.

References

Susan Athey and Glenn Ellison. Dynamics of open source movements. *Journal of Economics & Management Strategy*, 23(2):294–316, 2014.

Heski Bar-Isaac, Guillermo Caruana, and Vicente Cuñat. Targeted product design. *American Economic Journal: Microeconomics*, 15(2):157–186, 2023.

Kevin A Bryan and Jorge Lemus. The direction of innovation. *Journal of Economic Theory*, 172:247–272, 2017.

Steven Callander and Niko Matouschek. The novelty of innovation: competition, disruption, and antitrust policy. *Management Science*, 68(1):37–51, 2022.

Ramon Casadesus-Masanell and Gastón Llanes. Mixed source. *Management Science*, 57 (7):1212–1230, 2011.

Joshua S Gans. Market power in artificial intelligence. Working Paper 32270, National Bureau of Economic Research, March 2024. URL http://www.nber.org/papers/w32270.

Alan Jaske. Exploration and exploitation in r&d competition. 2022.

Justin P Johnson. Economics of open source software. 2001.

Justin P Johnson and David P Myatt. On the simple economics of advertising, marketing, and product design. *American Economic Review*, 96(3):756–784, 2006.

Vineet Kumar, Brett R Gordon, and Kannan Srinivasan. Competitive strategy for open source software. *Marketing Science*, 30(6):1066–1078, 2011.

Gastón Llanes and Ramiro de Elejalde. Industry equilibrium with open-source and proprietary firms. *International Journal of Industrial Organization*, 31(1):36–49, 2013.

Alondra Nelson, Arvind Narayanan, Caroline Meinhardt, Daniel E. Ho, Daniel Zhang, Dawn Song, Inioluwa Deborah Raji, Kevin Klyman, Marietje Schaake, Mihir Kshirsagar, Percy Liang, Peter Henderson, Rishi Bommasani, Rohini Kosoglu, Rumman Chowdhury, Sayash Kapoor, Seth Lazar, Shayne Longpre, Stefano Maf-

fulli, and Stella Biderman. Request for comment on dual use foundation artificial intelligence models with widely available model weights. Technical report, 2024. URL https://hai.stanford.edu/sites/default/files/2024-03/Response-NTIA-RFC-Open-Foundation-Models.pdf.

Steven C Salop. Monopolistic competition with outside goods. *The Bell Journal of Economics*, pages 141–156, 1979.

A Michael Spence. Monopoly, quality, and regulation. *The Bell Journal of Economics*, pages 417–429, 1975.

A Proofs

I prove each of the three problems: open paradigm equilibrium, closed paradigm equilibrium, and the social planner's optimum, in turn. I do so via a series of lemmata for each problem.

Open paradigm

First, note $p_S = S$ in the open paradigm. Setting $p_S > S$ sells zero units of the complementary product while any $p_S \leq S$ sells the complementary product to all developers who use the foundation model or an application. Therefore, founder profits are 0 if $p_S > S$, weakly positive and weakly increasing in p_S on $p_S \in [0, S]$, strictly increasing on $p_S \in [0, S)$ if a positive mass of developers consumes one of the models.

Next, note that any developer who fine-tunes must use their own application in equilibrium, as fine-tuning is costly and the only way for the developer to benefit is by using his own application. Therefore, $\eta_i = \theta_i$ for any developer who fine-tunes, as this aligns the fine-tuning with the developer's preferences.

The amount of fine-tuning maximizes the developer's continuation value from using that application:

$$z_{open} = \arg\max_{z \in [0,1]} V - r(1-z) - \frac{cz^2}{2}$$

so long as V is sufficiently large so this payoff is greater than 0. Assuming for now that V is in fact sufficiently large, the solution is $z_{open} = \frac{r}{c}$ if $r \leq c$ and $z_{open} = 1$ otherwise.

First, suppose $r \leq c$. Then $z_{open} = \frac{r}{c}$. Without loss of generality, fix the angle of a fine-tuning developer at $\theta_j = 0$ for product j. Then developers within angle T of this developer will prefer to use θ_j instead of doing their own fine-tuning, and T is given by $Tz_{open} = \frac{cz_{open}^2}{2}$. Plugging in $z_{open} = \frac{r}{c}$ yields $T = \frac{r}{2}$.

Free entry requires (1) that there exists no developer who does not fine-tune who would prefer to fine tune, and (2) that there exists no developer who does fine-tune who would rather free-ride on an existing application. (1) is equivalent to $\nexists\, i$ such that $|\theta_i - \eta_j| > \frac{r}{2}$ for all $j \in J$. This, in turn, is equivalent to $|\eta_j - \eta_{j'}| < r$ for any two adjacent j and j'. (2) is equivalent to $\nexists\, j \in J$ and , $j' \in J/j$ such that $|\eta_j - \eta_{j'}| < \frac{r}{2}$.

Let $j = 1, \ldots N$ denote the order of fine-tuned applications, beginning at angle 0. Summing condition (1), we have $\sum_{j=1}^{N-1} |\eta_{j+1} - \eta_j| = 2\pi \leq Nr$, so $N_{open} \geq \frac{2\pi}{r}$. Summing condition (2), we have $\sum_{j=1}^{N-1} |\eta_{j+1} - \eta_j| = 2\pi \geq N\frac{r}{2}$, which implies $N_{open} \leq \frac{4\pi}{r}$. So $N_{open} \in [\frac{2\pi}{r}, \frac{4\pi}{r}]$.

Next, note that the developers with the lowest payoffs are those who fine-tune. Any other developer earns strictly higher payoffs in equilibrium (if it earned lower payoffs, it could always fine-tune its own application). Therefore, V must be large enough to guarantee a payoff of at least 0 to the fine-tuning developers. Otherwise, fine-tuners would choose to consume 0 and, absent any fine-tuned applications, all other developers would consume 0 as well. This means the founder's payoffs are discontinuous at the V that gives fine-tuners equilibrium payoffs of 0. Below this, the founder earns $Q_S p_S = 0$; above this it earns $Q_s p_S = 2\pi p_S$. Other than this discontinuity, the founder's payoffs are decreasing in $V \in [0, \infty)$. Therefore, the founder either sets $V_{open} = 0$ or it sets $V_{open} = \frac{2rc - r^2}{2c}$ so the fine-tuners earn 0 payoff in equilibrium. A necessary and sufficient condition for the founder to prefer the latter is

$$2\pi s - \frac{g(2rc - r^2)^2}{2(2c)^2} > 0$$

which yields the threshold $S > \frac{(2rc - r^2)^2 g}{16c^2} = \bar{S}_{open}$.

Now suppose $r > c$. Then the fine-tuners reach the boundary $z_{open} = 1$. The critical angle beyond which a developer would prefer to fine-tune their own application than consume someone else's is given by $v - \bar{\theta} = v - \frac{c}{2}$, so $\bar{\theta} = \frac{c}{2}$. Free-entry conditions mean applications are at least $\frac{c}{2}$ apart and at most c apart. This means N_{open} is between $\frac{2\pi}{c}$ and $\frac{4\pi}{c}$. The founder will again either choose $V = 0$

or the smallest V to give fine-tuners a positive payoff, which in this case is given by $V_{open} = \frac{c}{2}$. The founder earns positive profits, and therefore prefers $V_{open} = \frac{c}{2}$, if $\frac{s}{g} \geq \frac{c^2}{16\pi}$. This concludes the proof of the equilibrium characteristics for the open model paradigm.

Closed paradigm

As with the open paradigm, so long as some developer uses either the foundation model or an application, $p_S = S$ because $p_S > S$ sells zero units of the complementary product while any $p_S \leq S$ sells the complementary product to all developers who use the foundation model or an application, and profits are increasing in p_S conditional on Q_S fixed.

Next, I show that it is profitable for the founder to offer at least one application with $z_j > 0$. If the founder only offers F, it optimally sets $p_F = V - r$, which is the uniform willingness to pay for F across all developers. The founder then earns profits $2\pi(V - r + S) - \frac{gV^2}{2}$. Now suppose the developer offers one product with $z_j > 0$ and $p_j = V - r(1 - \frac{z_j}{2})$. Note that this may not be the optimal price; it suffices to show that offering an application with this (suboptimal) price is profitable relative to only offering F. Locate this product at angle $\eta_j = 0$ without loss of generality. Noting that the next-best option of F or 0 yields utility 0, i will buy the application if

$$\begin{aligned} & V - r(1 - z_j) - p_j - z_j|\theta_i - 0| \geq 0 \\ \leftrightarrow \quad & \frac{rz_j}{2} - z_j|\theta_i - 0| \geq 0 \end{aligned}$$

The mass that buys j is then $Q_j = 2\frac{\frac{rz_j}{2}}{z_j} = r$, which is less than 2π by assumption. The founder's profits in this case are $(2\pi - r)(V - r + S) + r(V - r(1 - \frac{z_j}{2} + S)) - \frac{gV^2}{2} - \frac{cz_j^2}{2}$. The change in profit relative to only offering F is then $r\frac{rz_j}{2} - \frac{cz_j^2}{2}$, which is greater than 0 if $z_j < \frac{r^2}{c}$. Because $r > 0$, there exists some $z_j > 0$ for which building an application and selling it at this price is profitable for the founder.

Next, note that, if it is profitable to offer one application z_j at price p_j, it is profitable to offer another application with the same z_j and p_j so long as these products do not cannibalize each other, i.e., the set of developers with $u_{ij} > \max\{u_{iF}, 0\}$ and the set of developers with $u_{ij'} > \max\{u_{iF}, 0\}$ is disjoint. By this argument, abstracting from integer constraints, the founder will offer N products that are evenly spaced so that $u_{ij} > \max\{u_{iF}, 0\}$ for exactly one $j \in J$.

With this result in hand, the founder's problem becomes:

$$\begin{aligned} \max_{z,p,p_F,V} \quad & 2\pi(p+S) - N\frac{cz^2}{2} - \frac{gV^2}{2} \\ \text{subject to} \quad & N = \frac{\pi}{T} \\ & V - r(1-z) - zT - p = \max\{0, V - r - p_F\} \\ & z \in [0,1], V \geq 0,\ p \in [0, V - r(1-z)] \end{aligned}$$

That is: the founder chooses a symmetric degree of specialization z and price for applications p, as well as foundation model quality V and foundation model price p_F to maximize revenues from selling the applications to developers net of the cost of building the models. The maximum angle between a developer who consumes product j and that developer equals T, where T is the angle that makes a developer indifferent between an application and the next-best option of either consuming F or consuming 0, which is the second equality constraint above. The union of the segments of developers who buy each product j must cover the whole circle, so, up to integer constraints, $2\pi = 2NT$, which is the first equality constraint above. Finally, z must be between 0 and 1, and $p < V - r(1-z)$ so that there is positive demand for the applications.

Next, it is easy to see that p_F optimally equals $V - r$, since increasing p_F reduces the value of the outside option, which increases the price the founder can charge for the applications, so long as $V - r \geq p_F$. If $V - r < p_F$, the best outside option is 0, so the founder cannot extract any more surplus. The problem can be rewritten as

$$\begin{aligned} \max_{z,p,V} \quad & 2\pi(p+S) - N\frac{cz^2}{2} - \frac{gV^2}{2} \\ \text{subject to} \quad & N = \frac{\pi}{T} \\ & V - r(1-z) - zT - p = 0 \\ & z \in [0,1],\ V \geq 0,\ p \in [0, V - r(1-z)] \end{aligned}$$

Plugging the equality constraints into the optimization problem yields

$$\begin{aligned} \max_{z,p,V} \quad & 2\pi(p+S) - \frac{\pi c z^3}{2(V - r(1-z) - p)} - \frac{gV^2}{2} \\ \text{subject to} \quad & z \in [0,1],\ V \geq 0,\ p \in [0, V - r(1-z)]. \end{aligned}$$

I next solve this problem as if the remaining constraints do not exist and verify, after the fact, that the constraint on prices is satisfied. We can verify that the maximand above is concave on $z \in [0,1]$ so long as $p < V - r(1-z)$. The first order condition with respect to z yields $z = \frac{3(p+r-V)}{2r}$.

Plugging back in for z, the resulting maximization problem is concave in p. Taking the first-order condition with respect to p yields $p = \frac{8r^3}{27c} + V - r$. Plugging this in for p and taking a first order condition with respect to V yields $V_{closed} = \frac{2\pi}{g}$.

Solving backwards, $p_{closed} = \frac{2\pi}{g} - r + \frac{8r^3}{27c}$ and $z_{closed} = \frac{4r^2}{9c}$. We can verify that, so long as $\frac{4r^2}{9c} < 1$, $z_{closed} \in [0,1]$ and $p_{closed} < \frac{2\pi}{g} - r < \frac{2\pi}{g} - r(1 - z_{closed})$, so the constraints on p are satisfied. Therefore, this characterizes the closed-paradigm equilibrium. Finally, the founder must find it worthwhile to build the model in the first place. This is true if

$$2\pi\left(\frac{2\pi}{g} - r + \frac{8r^3}{27c} + S\right) - \frac{8\pi r^3}{27c} - \frac{2\pi^2}{g} \geq 0$$

Solving for S yields the threshold $S \geq \bar{S}_{closed} = \frac{\pi}{g} + r - \frac{4r^3}{27c}$ for the founder to build the model. Finally, we can solve for N to yield $N_{closed} = \frac{3\pi}{r}$ products.

When $\frac{4r^2}{9c} > 1$, the boundary solution holds: $z_j = 1$. Plugging $z = 1$ into the founder's profit, we can verify that profits are concave on the interval $p \in [0, V]$. Taking a first-order condition with respect to p and solving for p yields $p_j \frac{1}{2}(2V - \sqrt{c})$. Plugging this in for p and solving the first order condition for V again yields $V = \frac{2\pi}{g}$. This means $p_{closed} = \frac{2\pi}{g} - \frac{\sqrt{c}}{2}$.

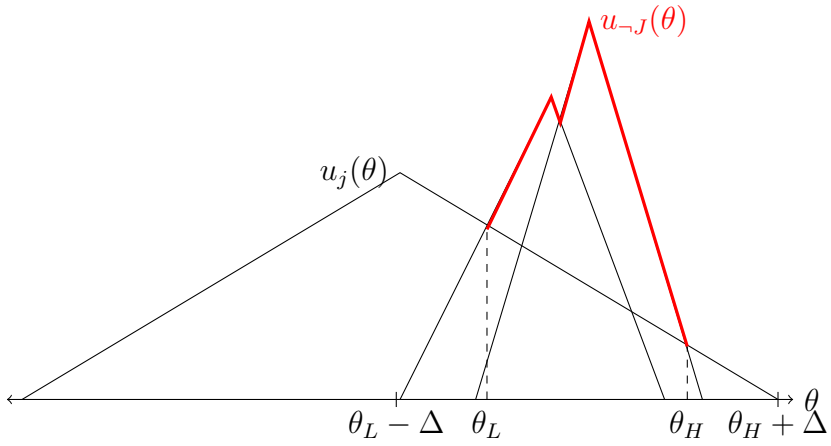
Social planner

The social planner's problem is even more involved, so I solve the social planner's problem with a series of lemmata. The first is obvious and stated without proof:

Lemma 1. *$q_{ij} = 1$ only if $u_{ij} \geq_{j'} u_{ij'}\ \forall j' \in J \cup F \cup 0$*

That is, given a set of products, the social planner allocates each product to developers according to utility maximization. Next, I prove that the social planner's optimal allocation is a convex partition.

Figure 4: Illustration of Lemma 2



I define this formally below, but it roughly means that the Salop disk can be segmented into convex groups of developers according to the product they are allocated.

Definition. *An allocation is a* ***convex partition*** *if* $q_{ij} = 1$ *and* $q_{kj} = 1$ *implies* $q_{lj} = 1$ *for all* l *between* k *and* j.

Lemma 2. *The social planner's optimal allocation is a convex partition.*

Proof. Suppose we had an allocation that was not a convex partition. I will show that there is another allocation that increases developer surplus without costing any more.

If the allocation is not a convex partition, there is a product j located at η_j, a range $[\theta_L, \theta_H]$ of developers who are allocated $\neg j$, while developers on the intervals $[\theta_L - \Delta, \theta_L)$ and $(\theta_H, \theta_H + \Delta]$ are allocated j. Let $\neg J$ denote the set of products allocated to the consumers on $[\theta_L, \theta_H]$.

Write $u_j(\theta)$ as the utility for product j for the consumer at angle θ. That is, $u_j(\theta_i) = u_{ij}$. By Lemma 1, for any product $k \in \neg J$, $u_k(\theta) \geq u_j(\theta)$ for some $\theta_i \in [\theta_L, \theta_H]$, and $u_{ik} < u_{ij}$ on $[\theta_L - \Delta, \theta_L) \cup (\theta_H, \theta_H + \Delta]$. Because u_{ij} depends on θ_i through the absolute distance $|\theta_i - \eta_j|$, utility functions have the single-peaked shape depicted in Figure [x]. It immediately follows that, for any product $k \in \neg J$, u_{ik} must cross u_{ij} once from below and once from above. Therefore, the set of θ_i such that $u_{ik} > u_{ij}$ for some $k \in \neg J$ is limited to $[\theta_L, \theta_H]$.

Integrated developer surplus over the interval $[\theta_L - \Delta, \theta_H + \Delta]$ is

$$\int_{\theta_L-\Delta}^{\theta_L} u_j(\theta)d\theta + \int_{\theta_L}^{\theta_H} u_{\neg J}(\theta)d\theta + \int_{\theta_H}^{\theta_H+\Delta} u_{\neg J}(\theta)d\theta$$

where $u_{\neg J}(\theta) \equiv \max_{k\in\neg J} u_k(\theta)$.

Suppose wlog that $|\theta_H - \eta_j| \geq |\theta_L - \eta_j|$ which implies $u_j(\theta_H) \leq u_j(\theta_L)$. Also note that, by the continuity of the utility functions, $u_j(\theta_H) = u_{\neg J}(\theta_H)$ and $u_j(\theta_L) = u_{\neg J}(\theta_L)$. Now consider the following transformation of the set $\neg J$: $\eta_k \to \eta_k + \epsilon,\ \forall k \in \neg J$. Integrated developer surplus is now

$$\int_{\theta_L-\Delta}^{\theta_L+\epsilon} u_j(\theta)d\theta + \int_{\theta_L+\epsilon}^{\theta_H+\epsilon} u_{\neg J}(\theta-\epsilon)d\theta + \int_{\theta_H+\epsilon}^{\theta_H+\Delta} u_{\neg J}(\theta)d\theta.$$

Differentiating with respect to ϵ, the second term in the above equation is unchanged, while the first term increases by $u_j(\theta_L+\epsilon)|_{\epsilon=0} = u_j(\theta_L)$ and the second term decreases by $u_j(\theta_H+\epsilon)_{\epsilon=0} = u_j(\theta_H)$. The derivative is thus $u_j(\theta_L) - u_j(\theta_H) > 0$.

That is, shift the location of each product in $\neg J$ clockwise (rightward in Figure 4) by arbitrarily small amount ϵ. Therefore, this term is weakly positive: integrated developer surplus is increasing in ϵ locally. Taking a second derivative verifies that it is locally convex if $u_{iL} = u_{iH}$. Therefore, integrated developer surplus increases with a small clockwise shift in $\neg J$. The number of products and their z does not change, so overall welfare is also improved. Therefore, this (non-convex partition) allocation is not optimal. □

Lemma 3. *Nobody is allocated the foundation model in the social planner's optimum:* $q_{iF} = 0,\ \forall i$.

Proof. Suppose this was not true: $q_{iF} = 1$ for some i. From Lemma 1, this means that $u_{iF} > u_{ij}, \forall j \in J$. But consider a product $\tilde{j}$ such that $\eta_{\tilde{j}} = \theta_i$. Start with $z_{\tilde{j}} = 0$, which implies $u_{iF} = u_{i\tilde{j}}$ by construction. This means that $\tilde{j}$ is not currently in J. Consider a small increase in $z_{\tilde{j}}$.

$$\frac{\partial u_{i\tilde{j}}}{\partial z_{\tilde{j}}} = r > 0$$

Therefore, there exists some small $z_{\tilde{j}}$ such that $u_{iF} < u_{i\tilde{j}}$. Finally, consider that the marginal cost of increasing $z_{\tilde{j}}$ by a small amount starting from 0 is $cz_{\tilde{j}} = 0$. That is, there is no first-order cost of building $\tilde{j}$ and a first-order benefit to developer i. Because $\tilde{j}$ did not previously exist in j, no other

$i' \neq i$ could be made worse off. Therefore, welfare must increase. □

Lemma 4. *Fix any V, q_{is}, and $N \geq 2$. Optimally, $z_j = z_{j'}$ for all j, j' in J, and products are spaced evenly , up to integer constraints: $|\eta_j - \eta_{j'}| = |\eta_{j'} - \eta_{j''}|$ for consecutive applications j, j', and j''.*

Proof. First isolate the parts of developer surplus that vary with z:

$$\sum_j \int_{q_{ij}=1} -r(1-z_j) - |\theta_i - \eta_j| z_j \, di - \frac{c z_j^2}{2}$$

Note that, for any set of η_j and an allocation of products to consumers, we have

$$\begin{aligned} & \sum_j \int_{q_{ij}=1} -r(1-z_j) - |\theta_i - \eta_j| z_j \, di - \frac{c z_j^2}{2} \\ = & \sum_j -Q_j r(1-z_j) - Q_j D_j z_j - \frac{c z_j^2}{2} \\ = & -2\pi r + \sum_j Q_j z_j (r - D_j) - \frac{c z_j^2}{2} \end{aligned}$$

where $Q_j \equiv \int_{q_{ij}=1} di$ is the mass of developers allocated product j and $D_j \equiv \frac{1}{Q_j} \int_{q_{ij}=1} |\theta_i - \eta_j| \, di$ is the mean absolute distance between j and a consumer allocated j. By Lemma 2, the allocation is a convex partition. Therefore we can characterize the allocation by (L_j, H_j) where L_j is the distance between η_j and the angle of the consumer who is marginal between product j and product $j-1$ (or product N if $j = 1$) and H_j is the angle between η_j and the consumer who is marginal between product j and product $j+1$ (or product 1 if $j = N$).:

$$Q_j = L_j + H_j \qquad Q_j D_j = \int_{L_j}^{H_j} |t| dt = \frac{L_j^2 + H_j^2}{2}.$$

Furthermore, by Lemma 3, no developer consumes F and $\sum_j L_j + H_j = 2\pi$. Using this information, social planner's objective becomes

$$\begin{aligned}
\max_{L_j,H_j,z_j} \quad & -2\pi r + \sum_j z_j \left(L_j - \frac{L_j^2}{2} + H_j - \frac{H_j^2}{2} \right) - \frac{c z_j^2}{2} \\
\text{subject to} \quad & H_j, L_j > 0;\ z_j \in [0,1]; \\
& \sum_j L_j + H_j = 2\pi
\end{aligned}$$

This problem is symmetric and concave z_j. It follows that $z_j = z_{j'}$ for all z_j. It is also concave in L_j and H_j, subject to a linear equality constraint, so $L_j = H_j$ for all j, which, using Lemma 1, implies that products are spaced evenly. □

With these results in hand, the social planner's problem becomes much simpler, as there is only one z to choose, and all developers are allocated the application closest to them. The social planner need only be concerned with choosing z, choosing V, and the number of (evenly-spaced) applications N:

$$\begin{aligned}
\max_{V,z,N} \quad & N \left[\int_0^T V + s - r(1-z) - zt \, dt - \frac{cz^2}{2} \right] - \frac{gV^2}{2} \\
\text{subject to} \quad & 2NT = 2\pi \\
& z \in [0,1],\ V > 0,\ N > 0.
\end{aligned}$$

This problem is smooth and concave in all of its arguments. Again, I solve as if the constraints do not exist. Taking a first order condition with respect to V yields $V = \frac{2\pi}{g}$. A first order condtiion with respect N yields $N = \frac{\sqrt{2}\pi}{\sqrt{zc}}$. Plugging this in and taking a first order condition with respect to z yields $z = \frac{8r^2}{9c}$. Therefore, if $\frac{8r^2}{9c} < 1$, $z = \frac{8r^2}{9c}$ and, plugging back in for N, we obtain $N = \frac{3\pi}{2r}$. Otherwise, z is at the boundary, so $z = 1$ and, plugging in for N yields $N = \frac{\sqrt{2}\pi}{\sqrt{c}}$. □

Corollary 1. *If* $r < \min\{c, \sqrt{\frac{9c}{8}}\}$, *then* $z_{open} > z_{soc}$. *If* $r > c$ *and* $r < \sqrt{\frac{9c}{8}}$, *then* $z_{open} < z_{soc}$. *Otherwise*, $z_{open} = z_{soc}$.

Proof. Ignoring the condition that $z \leq 1$, the relevant inequality is

$$\frac{r}{c} > \frac{8r^2}{9c}$$

which simplifies to $r < \frac{9}{8}$. Therefore, if $r < \frac{9}{8}$, and if $z_{open} < 1$, then $z_{soc} < z_{open} < 1$. If

$r < \frac{9}{8}$ and $z_{open} = 1$, then either $z_{soc} = 1$ or $z_{soc} < 1$. Conversely, if $r > \frac{9}{8}$ and $z_{soc} < 1$, then $z_{open} < z_{soc} < 1$. If $r > \frac{9}{8}$ and $z_{soc} = 1$ then either $z_{open} < z_{soc} = 1$ or $z_{open} = 1$.

Therefore, the region where $z_{open} > z_{soc}$ is defined by $r < \frac{9}{8}$ and $z_{soc} < 1$, which corresponds to $r < \sqrt{\frac{9c}{8}}$. The region where $z_{open} < z_{soc}$ is defined by $r > \frac{9c}{8}$ and $z_{open} < 1$, which corresponds to $r < c$. □

Corollary 2. There exists $\bar{g} > 0$ such that, if $g > (<)\bar{g}$, $V_{closed} = V_{soc} < (>)V_{open}$. $\bar{g}$ is weakly decreasing in r and strictly decreasing in c. It is strictly decreasing in r if $r < c$.

Proof. If $r < c$, $\bar{g}$ solves $\frac{2\pi}{\bar{g}} = r - \frac{r^2}{2c}$, which yields $\bar{g} = \frac{4\pi c}{r(2c-r)}$. Differentiating with respect to r yields $\frac{\partial \bar{g}}{\partial r} = \frac{-8\pi c(c-r)}{r^2(2c-r)^2}$ which is negative when $r < c$. Differentiating with respect to c yields $\frac{\partial \bar{g}}{\partial c} = \frac{-4\pi}{(2c-r)^2}$ which is also negative.

When $r > c$, $\bar{g}$ solves $\frac{2\pi}{\bar{g}} = \frac{c}{2}$, so $\bar{g} = \frac{4\pi}{c}$. This does not depend on r and, differentiating with respect to c, yields $\frac{\partial \bar{g}}{\partial c} = -\frac{4\pi}{c^2}$ which is negative. □