The Economics of Frontier-AI Competition: Leader Profitability under Open-Weight Catch-Up*

Pavel Kocourek

July 2026 — preliminary draft

1 The question

Frontier AI labs spend tens of billions of dollars a year training models whose capabilities an open-weight fringe reproduces within months and then sells at marginal compute cost. It is thus an open question whether the frontier labs can ever turn a profit: whatever they build is soon offered by others at a price that leaves no margin. This note sets out a small dynamic model of that race to answer it.

The model treats the frontier labs as a single leader and the open-weight ecosystem as a competitive fringe that prices at cost. The leader can then charge only for the capability increment it holds over the fringe: its earnings are proportional to the value of the lead — the gap between what its model is worth and what the fringe's is. Under what conditions does the leader's flow profit turn positive within roughly five years — and stay positive? The answer turns on four rates and how they combine: how fast the frontier moves, how fast the follower closes the gap, how steeply value grows with capability, and how fast the model-building bill grows.

The note mirrors the companion widget exactly: Section 2 lays out the skeleton, and Sections 3–5 follow the widget's three levels — Basics, Dynamics, Catch-up channels — switching mechanisms on cumulatively, so nothing is ever replaced, only added.

2 Setup

Two players. The **leader** L is the set of frontier labs collapsed into one representative player training the single frontier model. The **follower** F is the open-weight / competitive fringe: it sells inference at marginal compute cost, so it earns nothing itself but disciplines the leader's price — customers can obtain the follower's capability essentially for free, and the leader can charge only for the increment above it.

---

*Preliminary and incomplete — an early draft, circulated for feedback. Written under a fellowship of the Center for AI Safety (CAIS). The note is the model companion to the interactive explorer at https://pkocourek.com/frontier_labs_profit, which implements exactly the model presented here, level by level, with the full calibration and a Monte Carlo forecasting mode. This note carries the symbolic model; calibration numbers, their sources, and sensitivity exploration live in the widget.

States. Capability is measured in orders of magnitude (OOMs) of *effective compute*: algorithmic level times physical training compute. With $a = \log_{10}$(algorithmic level) and $c = \log_{10}$(training compute),

$$x_t^i = a_t^i + c_t^i \quad (i \in \{L, F\}), \qquad \Delta_t = x_t^L - x_t^F, \tag{1}$$

normalized so that the early-2026 frontier sits at $x_0^L = 0$ — component-wise, $a_0^L = c_0^L = 0$; the follower starts a gap Δ_0 behind.

Payoff skeleton. Let $W(x)$ denote the *value index* of capability x: a dimensionless map with $W(0) = 1$, so W reads as value as a multiple of today's frontier value. The model carries no dollar level at all: earnings and the model-building bill are each normalised by their own level *today*, so every financial quantity is measured in multiples of today's model-building bill, and the leader's flow profit is simply

$$\Pi_t^L \;=\; E_t - B_t, \tag{2}$$

with earnings E_t riding the value gap $W(x_t^L) - W(x_t^F)$ and the bill starting at $B_0 = 1$. The model's whole content is in the dynamics that generate x_t^L and x_t^F, the shape of W, the cost path, and the one financial parameter the normalisation leaves behind.

3 Basics — the race, earnings, and costs

The base model holds every rate constant. The leader's capability grows at a constant speed: compute scales at rate g_c (OOM/yr) and algorithms improve at rate g_a, so $\dot{x}^L = g_c + g_a$. The follower has no engine of its own — it moves purely by catching up, closing the gap at rate δ (per year):

$$\dot{x}_t^F = \delta\,(x_t^L - x_t^F) = \delta\,\Delta_t, \qquad \text{so} \qquad \dot{\Delta}_t = \dot{x}^L - \delta\,\Delta_t. \tag{3}$$

The observed lag of the open-weight fringe behind the frontier pins δ: with $\delta\Delta_0$ equal to the leader's speed, the lag is constant and the gap holds at Δ_0.

Value is exponential in capability: each additional OOM multiplies the index by 10^{ν},

$$W(x) = 10^{\nu x}, \tag{4}$$

with the slope ν measured in OOMs of value per OOM of capability.

Earnings. Earnings are the rent on the lead: proportional to the value gap, and normalised by today's gap,

$$E_t \;=\; \rho\,\frac{W(x_t^L) - W(x_t^F)}{W(x_0^L) - W(x_0^F)}, \tag{5}$$

where "earnings" means earnings *before* model-building costs and $\rho = E_0$ is the model's one financial parameter, discussed below. On the cost side, the leader rents compute and pays the bill B_t of the model it is currently running (the build lag arrives in Section 4), in units of today's bill:

$$B_t \;=\; 10^{\,c_t^L - g_p t}, \tag{6}$$

where g_p is the hardware price-performance decline (OOM/yr) — the bill tracks the compute path at falling prices, and algorithmic progress is already counted inside x, not here.

The coverage ratio. The headline object is the ratio of earnings to the bill,

$$\rho_t \;=\; \frac{E_t}{B_t}, \qquad \Pi_t^L > 0 \iff \rho_t > 1, \qquad \rho_0 = \rho. \tag{7}$$

The single financial parameter ρ is *coverage today*: operating earnings per unit of model-building outlay. Reported dollar figures are calibration evidence for it, not parameters of the model: with gross revenue R_0, operating margin before model-building costs m, and model-building outlay $k\,R_0$, coverage today is $\rho = m/k \approx 0.53$ — the industry earns roughly 53 cents per dollar of model-building spend. Dollar levels never enter: scaling every reported figure leaves the model unchanged, so only the ratio m/k was ever identified, and the model now carries just it. Break-even is $\rho_t = 1$; coverage starts below one, and the question is whether it climbs.

Will the leader ever be profitable? With the gap constant and value exponential, one inequality decides it. Earnings inherit W's growth rate along the leader's path, growing by the factor $10^{\nu(g_c+g_a)}$ per year, while the bill grows by $10^{\,g_c-g_p}$ per year; coverage rises, and break-even eventually arrives, if and only if

$$\nu\,(g_c + g_a) \;>\; g_c - g_p. \tag{8}$$

At the widget's calibration — compute scaling near 3.2× per year, the bill growing near 2.35× per year, value about 2.1× per OOM — earnings grow about 2.2× per year and (8) *fails*: coverage falls, and the base model's verdict is that break-even never comes. Because every rate is constant, that is one verdict for all time. It rests, however, on compute scaling at 3.2× per year forever — exactly the assumption the Dynamics level drops. Dropping it does not settle the question either way: a compute slowdown bends the bill, but it also slows the frontier and lets the follower close the gap — and earnings live on the gap — while accelerating algorithmic progress pulls in the opposite direction. Whether break-even ever arrives becomes a quantitative race between these forces.

4 Dynamics

The Dynamics level turns on two opposing forces — compute growth slows down while algorithmic progress speeds up — together with a build lag that starts to matter once compute growth bends, and a bend in the value curve itself. The follower, earnings, and profit carry over from Basics unchanged.

The transition curve Γ

Three of the model's mechanisms are *transitions*: something moves from one steady value to another, over time or over capability. Two appear in this section, and a third with the follower's compute in the next; all use the same primitive, stated once: the logistic S-curve, dialled in four *observables*,

$$\Gamma\big(u;\, y(0),\, y_\infty,\, u_{mid},\, p_0\big)\;: \tag{9}$$

today's value $y(0)$, the destination y_∞, the midtime u_{mid} at which the transition is halfway done, and the position p_0 — the fraction of the transition already completed today (necessarily $p_0 < \frac{1}{2}$: the midpoint is still ahead). Figure 1 shows the shape. The plateau the curve came from and its steepness are *not* free parameters: they are derived so that the curve passes

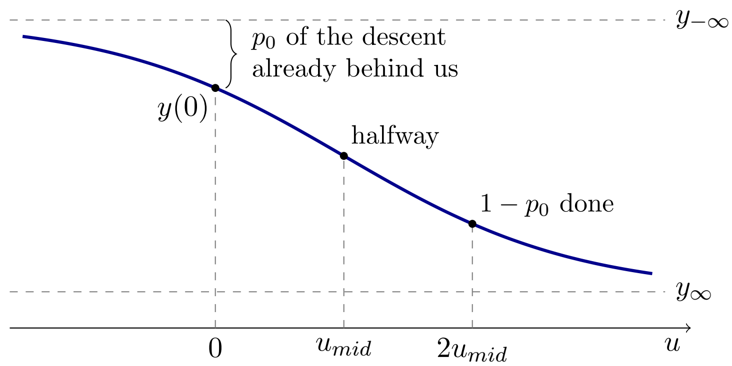

Figure 1: The transition curve $\Gamma(u;\, y(0),\, y_\infty,\, u_{mid},\, p_0)$, drawn for a falling transition (the compute slowdown's shape) with $p_0 = 0.25$. The four arguments are observables; the pre-transition plateau $y_{-\infty}$ and the steepness are derived (Appendix A), so the curve passes through today's value $y(0)$ at $u = 0$ for any choice of u_{mid} and p_0.

through today's value, already p_0 of the way along, at $u = 0$ (Appendix A has the closed form). So $\Gamma(0) = y(0)$ identically — an observation never depends on a scenario parameter, and moving u_{mid} or p_0 says where the transition *started*, never what it is now. By the logistic's symmetry the curve is $1 - p_0$ done at $2u_{mid}$, so a transition occupies roughly the window $[0,\, 2u_{mid}]$; and whenever $y_\infty = y(0)$ it is exactly constant.

Force 1 — the compute slowdown

Today's compute scaling cannot persist: power, fabrication capacity, and capital all bind. Compute growth now rides the transition curve down to a long-run floor $g_{c\infty}$, half-done at t_{mid}:

$$\dot{c}_t^L \;=\; g_{c,t} \;=\; \Gamma\big(t;\, g_c,\, g_{c\infty},\, t_{mid},\, p_0^c\big). \tag{10}$$

The anchor is *today's* realised growth g_c — the same g_c as in Basics, held fixed by construction — so positing that the slowdown is further along (p_0^c larger) raises the pre-slowdown plateau it came down from, never today's growth. Whenever $g_{c\infty} = g_c$ the curve is exactly constant and the base model never sees the shape.

Force 2 — algorithmic progress speeds up

The base model's constant algorithmic rate g_a becomes a *produced* quantity: a CES aggregate of researcher effort, accelerated by AI assistance, and experiment throughput, which moves with compute growth:

$$\dot{a}_t^L \;=\; g_a\left[(1-\alpha)\left(\frac{\psi(x_t^L)}{\psi(0)}\right)^{\eta} + \alpha\left(\frac{g_{c,t}}{g_c}\right)^{\eta}\right]^{1/\eta}, \qquad \psi(x) = 1 + \beta_0\, 10^{\,\gamma x}. \tag{11}$$

Both inputs are normalized to one at $t = 0$, so $\dot{a}_0^L = g_a$ remains today's rate. The function ψ is the AI-assistance multiplier: β_0 is the speedup AI delivers to its own R&D today, and γ is how strongly that assistance compounds with capability — $\gamma = 0$ freezes AI assistance at today's level. The exponent η sets how researchers and experiment compute substitute ($\eta = 1$, the base case, is a weighted average; $\eta < 0$ makes them complements, so the scarcer input rules); α is the experiment-compute weight.

With $\gamma > 0$ the ψ-feedback makes the frontier law formally super-exponential: for large enough γ the path reaches a finite-time singularity inside the horizon, and the widget flags when the feedback's share of algorithmic progress stops being small. More consequential for the race, the α-weighted term couples algorithmic progress to compute growth, so the slowdown drags the algorithmic engine down with it rather than leaving it untouched.

Training in advance — the build lag ℓ

A frontier model is not paid for when it ships: the firm pays now for the compute of the model shipping a lead time ℓ (years) ahead,

$$B_t \;=\; 10^{\,c^L_{t+\ell} - c^L_\ell - g_p t}. \tag{12}$$

Because today's bill is the numéraire, the path stays anchored at $B_0 = 1$ whatever ℓ is — the exponent is referenced to c^L_ℓ, the compute of the model being paid for *today*: the lead time tilts the cost path, it does not lift its level. Under the base model's constant compute growth the tilt cancels exactly; it is the slowdown, arriving at this same level, that makes ℓ bite. The mechanism captures a contrast visible in the industry: economics that are roughly break-even *per model* can coexist with a flow loss, because the flow is always funding the next, bigger model — the contrast the press draws between Anthropic and OpenAI.

Value in logs — the slope transition

Write value in logs, $w = \log_{10} W$, so $W(x) = 10^{\,w(x)}$ with $w(0) = 0$ (the index anchor $W(0) = 1$). What the Dynamics add is that the *slope* of w rides the transition curve — its second use, this time over capability rather than time:

$$w'(x) \;=\; \Gamma\big(x;\, \nu,\, \nu_\infty,\, x_{mid},\, p^w_0\big). \tag{13}$$

Each OOM of capability is worth $10^{\nu}\times$ *today* ($w'(0) = \nu$, the same ν as in Basics), easing to a floor of $10^{\nu_\infty}\times$ once the transition is done, halfway at x_{mid} (OOMs above today's frontier). There is no hard ceiling: value grows without bound at the floor slope, so each further OOM is worth at least $10^{\nu_\infty}\times$.

The shape of the value curve is the model's pivotal assumption. On the steep stretch a defended, constant gap earns *growing* rents — the leader harvests its lead. Once the slope eases, the same gap's rents grow ever more slowly: commoditization in all but name, if ν_∞ is low. Where the transition sits (x_{mid}) and where the slope lands (ν_∞) are therefore the decisive explored parameters.

The frontier speed race

The frontier's speed is the sum of the two forces,

$$\dot{x}^L_t \;=\; \dot{c}^L_t + \dot{a}^L_t, \tag{14}$$

and they pull in opposite directions: the compute slowdown brakes, the ψ-feedback accelerates — and through the α-weighted term the brake also drags on the accelerator. Whether the frontier speeds up or slows down is a calibration question, not a modelling one. One consequence carries forward: with δ still pinned by the observed lag, a decelerating leader means a *closing* gap. The next level gives the follower an engine of its own and makes the gap stationary again.

5 Catch-up channels

The follower now gets an engine of its own — its own algorithmic rate g_a^F and its own compute path, the same transition-curve family with its own parameters,

$$\dot{c}_t^F \;=\; \Gamma\big(t;\, g_c^F,\, g_{c\infty}^F,\, t_{mid}^F,\, p_0^F\big), \tag{15}$$

and the single catch-up rate δ unpacks into two channels. What catches up is the follower's *algorithmic* level, because compute cannot be copied:

$$\dot{a}_t^F \;=\; g_a^F \;+\; \underbrace{\delta_{dev}\left(a_t^L - a_t^F\right)}_{\text{developed-model channel}} \;+\; \underbrace{\delta_{rel}\left(x_t^L - x_t^F\right)}_{\text{released-model channel}} \;. \tag{16}$$

The *developed-model* channel δ_{dev} is methods diffusion — espionage, papers, talent mobility — so it runs on the algorithmic gap and operates even if the leader releases nothing. The *released-model* channel δ_{rel} is distillation of what the leader's deployed model can do, so it runs on the capability gap. Both gaps are assumed positive in (16) — the leader is ahead; in the implementation each gap is floored at zero, so imitation dies at parity and the follower never un-learns. Both cross terms measure imitation only; self-improvement gains run on each player's own engine.

The initial gap splits between the two states,

$$a_0^F = -\,\text{split}\cdot\Delta_0, \qquad c_0^F = -(1-\text{split})\,\Delta_0. \tag{17}$$

The calibration treats the total and the split differently: the observed open–closed lag pins the *total* $\delta_{dev} + \delta_{rel}$ — with the follower supplying most of its own speed, the channels together only need to close the wedge between the leader's speed and the follower's own engine — while the *split* between them has no clean observable. It is a free, explored parameter; because the two channels run on different gaps — the algorithmic gap and the full capability gap — how the total divides shapes the follower's path.

6 Outputs

The model is now complete: the leader's path from (10)–(11), the follower's from (15)–(16), value from (13), and profit from the skeleton (2) with cost (12).

Headline. The headline output is the path of the leader's flow profit Π_t^L: does it turn positive within roughly five years, and does it *stay* positive? Equivalently, does the coverage ratio $\rho_t = E_t/B_t$ cross one — starting from its handicap $\rho_0 = \rho$?

The long-run benchmark. When the system settles — bounded growth regime, constant gap — earnings grow at $\nu_\infty\,\dot{x}^L_\infty$ and the bill at $g_{c\infty} - g_p$ (both in OOM/yr), so in the long run the leader harvests if and only if

$$\nu_\infty\,\dot{x}^L_\infty \;>\; g_{c\infty} - g_p. \tag{18}$$

The normalisations and the build lag are constant factors and drop out of growth rates. This limit favors harvest at strikingly low floor slope ν_∞: once compute growth has fallen to its floor, even modest value scaling outruns the bill. The battleground is therefore the *transition* — today the bill's growth outruns earnings growth for moderate ν, so Π^L_t dips before recovering, and the five-year sign question is about levels and the slowdown timing t_{mid}, not the limit.

What this note does not do. Every parameter above is calibrated to a stated observable or explicitly flagged as free, and the contested ones carry ranges rather than points; the companion widget[1] exposes the same three levels — Basics, Dynamics, Catch-up channels — documents every calibration target, and draws jointly from all ranges in a Monte Carlo mode to turn the headline question into a forecast distribution.

Appendix A The transition curve in closed form

Let σ be the sigmoid function, $\sigma(z) = 1/(1+e^{-z})$, rising from 0 to 1 with $\sigma(0) = \frac{1}{2}$, $\sigma(-z) = 1 - \sigma(z)$, and inverse the log-odds $\sigma^{-1}(q) = \ln\frac{q}{1-q}$. The transition curve is the sigmoid run between two plateaus,

$$\Gamma\big(u;\, y(0),\, y_\infty,\, u_{mid},\, p_0\big) \;=\; y_{-\infty} + \big(y_\infty - y_{-\infty}\big)\,\sigma\big(s\,(u - u_{mid})\big), \tag{19}$$

with the steepness s and the pre-transition plateau $y_{-\infty}$ derived from the four dialled observables by two identities.

The steepness, from the position. The fraction of the transition completed at $u = 0$ is $\sigma(-s\,u_{mid}) = p_0$, so

$$s\,u_{mid} \;=\; \ln\frac{1-p_0}{p_0}, \tag{20}$$

the log-odds of the journey remaining. The curve then reads p_0 done at $u = 0$, one half at u_{mid}, and — by the sigmoid's symmetry — $1 - p_0$ at $2u_{mid}$, for any p_0.

The plateau, from today's value. Today's value is the position-weighted average of the two plateaus, $y(0) = (1-p_0)\,y_{-\infty} + p_0\,y_\infty$; un-mixing it gives

$$y_{-\infty} \;=\; \frac{y(0) - p_0\,y_\infty}{1-p_0}. \tag{21}$$

As $p_0 \to 0$ the plateau is today's value — nothing has happened yet. For a slowdown ($y_\infty < y(0)$) it returns $y_{-\infty} > y(0)$, correctly: we have already come part of the way down.

Substituting both identities into (19) confirms $\Gamma(0) = y(0)$ for every u_{mid} and p_0.

[1] https://pkocourek.com/frontier_labs_profit