When Does Automating AI Research Produce Explosive Growth?

Tom Davidson [1] **Basil Halperin** [2] **Thomas Houlden** [3] **Anton Korinek** [4]

[1]Forethought, [2]University of Virginia, [3]Columbia University, [4]University of Virginia

December 2025

AI companies are aiming to automate AI research

> “ Advanced AI is interesting for many reasons, but perhaps **nothing is quite as significant as the fact that we can use it to do faster AI research.”**
>
> *Altman (2025)*

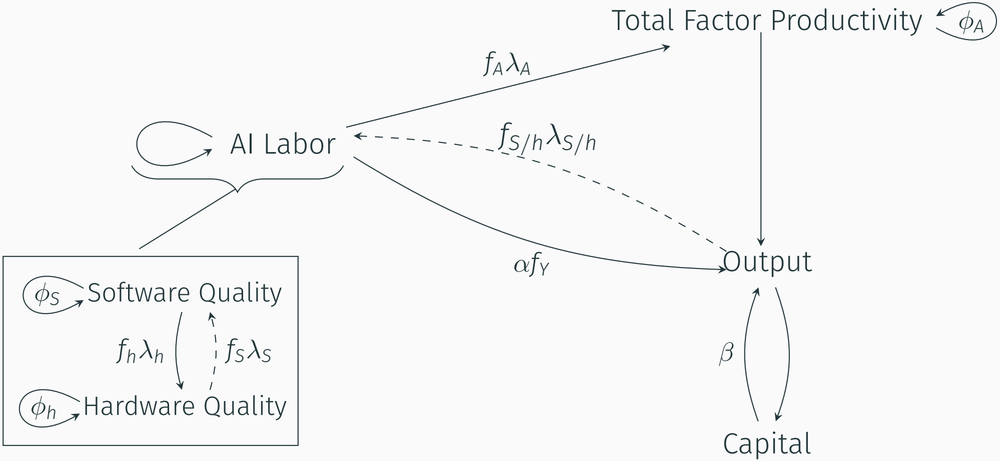
AI companies are aiming to automate AI research

> “ Advanced AI is interesting for many reasons, but perhaps **nothing is quite as significant as the fact that we can use it to do faster AI research.”**
>
> *Altman (2025)*

> “ We have set **internal goals** of having an automated AI research intern by September of 2026 … and a **true automated AI researcher by March of 2028**.”
>
> *Sam Altman*, Oct 2025

The software-hardware model of AI

The software-hardware model of AI



Roadmap ‣ Lit review

1. **Building blocks of the model**

   Lesson 1: diminishing returns prevent growth explosions

   Lesson 2: technological feedback loops create spillovers across sectors, offsetting diminishing returns

   Lesson 3: *economic* feedback loops create spillovers across sectors, offsetting diminishing returns

   Lesson 4: automation offsets diminishing returns

   Lesson 6 (empirical): diminishing returns are low in software/hardware

2. **The software-hardware model**
3. **Scope of claims**

1. Building blocks of the model

Lesson 1: diminishing returns prevent growth explosions

Lesson 2: technological feedback loops create spillovers across sectors, offsetting diminishing returns

Lesson 3: *economic* feedback loops create spillovers across sectors, offsetting diminishing returns

Lesson 4: automation offsets diminishing returns

Lesson 6 (empirical): diminishing returns are low in software/hardware

2. The software-hardware model

3. Scope of claims

1. Building blocks of the model

Lesson 1: diminishing returns prevent growth explosions

Lesson 2: technological feedback loops create spillovers across sectors, offsetting diminishing returns

Lesson 3: *economic* feedback loops create spillovers across sectors, offsetting diminishing returns

Lesson 4: automation offsets diminishing returns

Lesson 6 (empirical): diminishing returns are low in software/hardware

2. The software-hardware model

3. Scope of claims

The intelligence explosion

> **“Let an ultraintelligent machine be defined as a machine that can far surpass all the intellectual activities of any man.**
>
> — I.J. Good (1965)

The intelligence explosion

> “**Let an ultraintelligent machine be defined as a machine that can far surpass all the intellectual activities of any man.** Since the design of machines is one of these intellectual activities, an ultraintelligent machine could design even better machines;
>
> — I.J. Good (1965)

The intelligence explosion

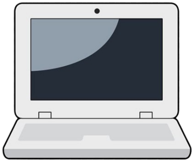
> "**Let an ultraintelligent machine be defined as a machine that can far surpass all the intellectual activities of any man.** Since the design of machines is one of these intellectual activities, an ultraintelligent machine could design even better machines; **there would then unquestionably be an 'intelligence explosion'**, and the intelligence of man would be left far behind."
>
> — I.J. Good (1965)

The intelligence explosion?

"**Let an ultraintelligent machine be defined as a machine that can far surpass all the intellectual activities of any man.** Since the design of machines is one of these intellectual activities, an ultraintelligent machine could design even better machines; **there would then unquestionably be an 'intelligence explosion'**, and the intelligence of man would be left far behind."

— I.J. Good (1965)

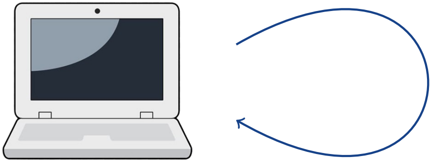
$\dot{S}_t$



The intelligence explosion?

> “**Let an ultraintelligent machine be defined as a machine that can far surpass all the intellectual activities of any man.** Since the design of machines is one of these intellectual activities, an ultraintelligent machine could design even better machines; **there would then unquestionably be an ‘intelligence explosion’**, and the intelligence of man would be left far behind.”
>
> — I.J. Good (1965)

$$\dot{S}_t = S_t$$

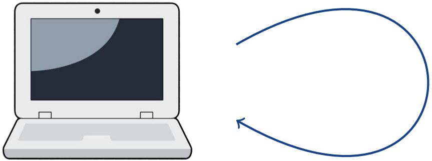


The intelligence explosion?

"**Let an ultraintelligent machine be defined as a machine that can far surpass all the intellectual activities of any man.** Since the design of machines is one of these intellectual activities, an ultraintelligent machine could design even better machines; **there would then unquestionably be an 'intelligence explosion'**, and the intelligence of man would be left far behind."

— I.J. Good (1965)

$$\dot{S}_t = S_t^{1+\phi}$$



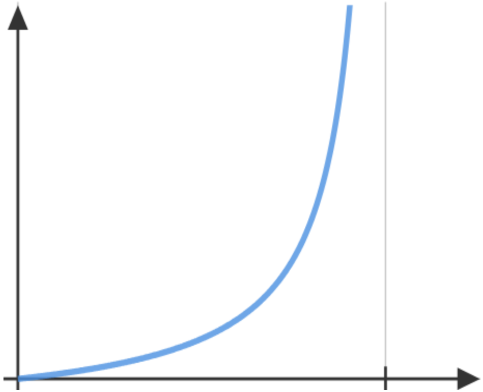
The intelligence explosion?

"**Let an ultraintelligent machine be defined as a machine that can far surpass all the intellectual activities of any man.** Since the design of machines is one of these intellectual activities, an ultraintelligent machine could design even better machines; **there would then unquestionably be an 'intelligence explosion'**, and the intelligence of man would be left far behind."

— I.J. Good (1965)

$$\dot{S}_t = S_t^{1+\phi}$$

$$\phi > 0$$



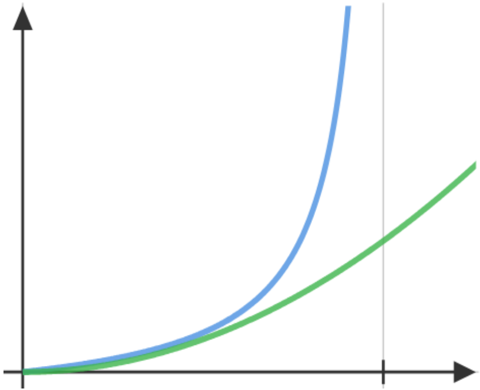
The intelligence explosion?

“**Let an ultraintelligent machine be defined as a machine that can far surpass all the intellectual activities of any man.** Since the design of machines is one of these intellectual activities, an ultraintelligent machine could design even better machines; **there would then unquestionably be an ‘intelligence explosion’**, and the intelligence of man would be left far behind.”

— I.J. Good (1965)

$$\dot{S}_t = S_t^{1+\phi}$$

$$\phi \in (-1, 0)$$

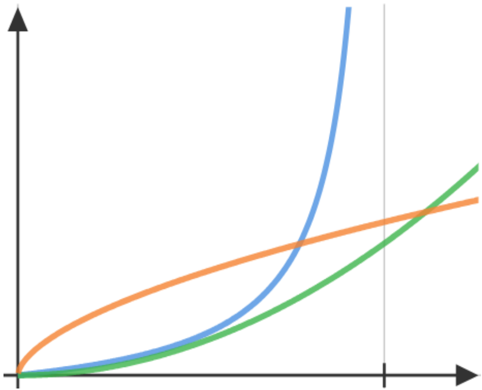


The intelligence explosion?

> “**Let an ultraintelligent machine be defined as a machine that can far surpass all the intellectual activities of any man.** Since the design of machines is one of these intellectual activities, an ultraintelligent machine could design even better machines; **there would then unquestionably be an ‘intelligence explosion’**, and the intelligence of man would be left far behind.”
>
> — I.J. Good (1965)

$$\dot{S}_t = S_t^{1+\phi}$$

$$\phi < -1$$



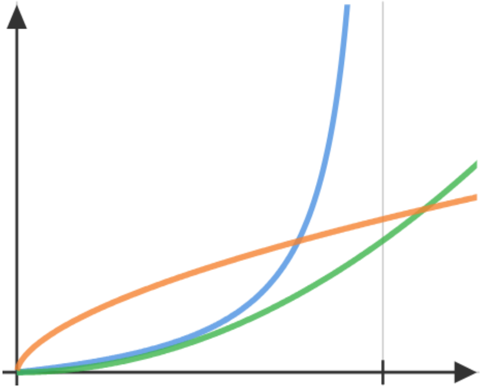
The intelligence explosion? The role of diminishing returns

“**Let an ultraintelligent machine be defined as a machine that can far surpass all the intellectual activities of any man.** Since the design of machines is one of these intellectual activities, an ultraintelligent machine could design even better machines; **there would then unquestionably be an ‘intelligence explosion’**, and the intelligence of man would be left far behind.”

— I.J. Good (1965)

$$\dot{S}_t = S_t^{1+\phi}$$

$$\phi < -1$$



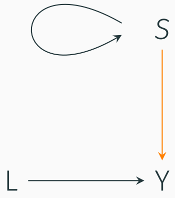
Will an *intelligence* explosion cause an *economic* explosion?

$$\dot{S} = S_t^{1+\phi}$$

Will an *intelligence* explosion cause an *economic* explosion?

$$\dot{S} = S_t^{1+\phi}$$

$$Y_t = S_t L_t^{\alpha}$$

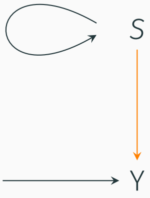


Will an *intelligence* explosion cause an *economic* explosion?

$$\dot{S} = S_t^{1+\phi}$$

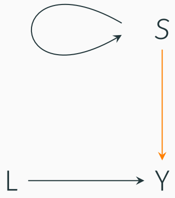
$$Y_t = S_t L_t^{\alpha}$$

Any form of 'intelligence explosion' causes the same form of economic explosion



Will an *intelligence* explosion cause an *economic* explosion?

$$\dot{S} = S_t^{1+\phi}$$
$$Y_t = S_t L_t^{\alpha}$$



Any form of 'intelligence explosion' causes the same form of economic explosion

- **Economic singularity** condition:

$$\phi > 0$$

1. Building blocks of the model

Lesson 1: diminishing returns prevent growth explosions

Lesson 2: technological feedback loops create spillovers across sectors, offsetting diminishing returns

Lesson 3: *economic* feedback loops create spillovers across sectors, offsetting diminishing returns

Lesson 4: automation offsets diminishing returns

Lesson 6 (empirical): diminishing returns are low in software/hardware

2. The software-hardware model

3. Scope of claims

Chip designers are using AI to design better chips

Article

A graph placement methodology for fast chip design

https://doi.org/10.1038/s41586-021-03544-w

Received: 3 November 2020

Accepted: 13 April 2021

Published online: 9 June 2021

Check for updates

Azalia Mirhoseini[1,4]✉, Anna Goldie[1,3,4]✉, Mustafa Yazgan[2], Joe Wenjie Jiang[1], Ebrahim Songhori[1], Shen Wang[1], Young-Joon Lee[2], Eric Johnson[1], Omkar Pathak[2], Azade Nova[1], Jiwoo Pak[2], Andy Tong[2], Kavya Srinivasa[2], William Hang[3], Emre Tuncer[2], Quoc V. Le[1], James Laudon[1], Richard Ho[2], Roger Carpenter[2] & Jeff Dean[1]

Chip floorplanning is the engineering task of designing the physical layout of a computer chip. Despite five decades of research[1], chip floorplanning has defied automation, requiring months of intense effort by physical design engineers to produce manufacturable layouts. Here we present a deep reinforcement learning approach to chip floorplanning. In under six hours, our method automatically

Chip designers are using AI to design better chips; AI labs use better chips to design better AI

Article

A graph placement methodology for fast chip design

https://doi.org/10.1038/s41586-021-03544-w

Received: 3 November 2020

Accepted: 13 April 2021

Published online: 9 June 2021

Check for updates

Azalia Mirhoseini[1,4]✉, Anna Goldie[1,3,4]✉, Mustafa Yazgan[2], Joe Wenjie Jiang[1], Ebrahim Songhori[1], Shen Wang[1], Young-Joon Lee[2], Eric Johnson[1], Omkar Pathak[2], Azade Nova[1], Jiwoo Pak[2], Andy Tong[2], Kavya Srinivasa[2], William Hang[3], Emre Tuncer[2], Quoc V. Le[1], James Laudon[1], Richard Ho[2], Roger Carpenter[2] & Jeff Dean[1]

Chip floorplanning is the engineering task of designing the physical layout of a computer chip. Despite five decades of research[1], chip floorplanning has defied automation, requiring months of intense effort by physical design engineers to produce manufacturable layouts. Here we present a deep reinforcement learning approach to chip floorplanning. In under six hours, our method automatically

Chip designers are using AI to design better chips;
AI labs use better chips to design better AI; ...

Article

A graph placement methodology for fast chip design

https://doi.org/10.1038/s41586-021-03544-w

Received: 3 November 2020

Accepted: 13 April 2021

Published online: 9 June 2021

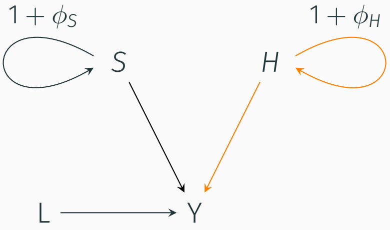
Check for updates

Azalia Mirhoseini[1,4]✉, Anna Goldie[1,3,4]✉, Mustafa Yazgan[2], Joe Wenjie Jiang[1], Ebrahim Songhori[1], Shen Wang[1], Young-Joon Lee[2], Eric Johnson[1], Omkar Pathak[2], Azade Nova[1], Jiwoo Pak[2], Andy Tong[2], Kavya Srinivasa[2], William Hang[3], Emre Tuncer[2], Quoc V. Le[1], James Laudon[1], Richard Ho[2], Roger Carpenter[2] & Jeff Dean[1]

Chip floorplanning is the engineering task of designing the physical layout of a computer chip. Despite five decades of research[1], chip floorplanning has defied automation, requiring months of intense effort by physical design engineers to produce manufacturable layouts. Here we present a deep reinforcement learning approach to chip floorplanning. In under six hours, our method automatically

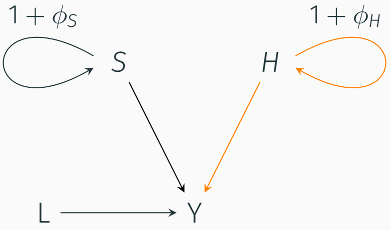
A networked semi-endogenous growth model

$$\dot{S}_t = S_t^{1+\phi_S}$$
$$\dot{H}_t = H_t^{1+\phi_H}$$
$$Y_t = (S_t H_t)^{1/2} L_t^{\alpha}$$



A networked semi-endogenous growth model

$$\dot{S}_t = S_t^{1+\phi_S}$$
$$\dot{H}_t = H_t^{1+\phi_H}$$
$$Y_t = (S_t H_t)^{1/2} L_t^{\alpha}$$

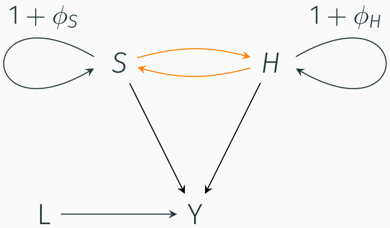


Economic singularity condition:

$$\phi_S \text{ or } \phi_H > 0$$

A networked semi-endogenous growth model: technological spillovers

$$\dot{S}_t = S_t^{1+\phi_S} H^{\lambda_S}$$
$$\dot{H}_t = H_t^{1+\phi_H} S^{\lambda_H}$$
$$Y_t = (S_t H_t)^{1/2} L_t^{\alpha}$$

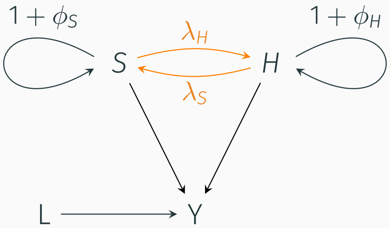


Economic singularity condition:

$$\phi_S \text{ or } \phi_H > 0$$

A networked semi-endogenous growth model: technological spillovers

$$\dot{S}_t = S_t^{1+\phi_S} H^{\lambda_S}$$
$$\dot{H}_t = H_t^{1+\phi_H} S^{\lambda_H}$$
$$Y_t = (S_t H_t)^{1/2} L_t^{\alpha}$$



Economic singularity condition:

$$\phi_S \text{ or } \phi_H > 0$$

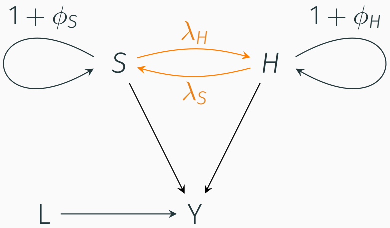
or

$$\underbrace{\qquad\qquad\qquad}_{\text{direct effects}} + \underbrace{\qquad}_{\text{indirect effects}} > 1$$

Proposition (explosive systems).
System explodes in finite time iff the interaction matrix, $\begin{bmatrix} 1+\phi_S & \lambda_S \\ \lambda_H & 1+\phi_H \end{bmatrix}$, has an eigenvalue > 1.

A networked semi-endogenous growth model: technological spillovers

$$\dot{S}_t = S_t^{1+\phi_S} H^{\lambda_S}$$
$$\dot{H}_t = H_t^{1+\phi_H} S^{\lambda_H}$$
$$Y_t = (S_t H_t)^{1/2} L_t^{\alpha}$$



Economic singularity condition:

$$\phi_S \text{ or } \phi_H > 0$$

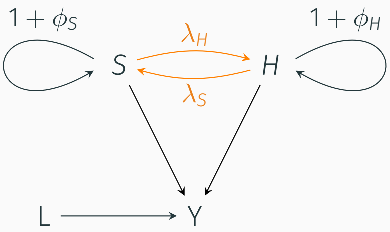
or

$$\underbrace{(1+\phi_S)+(1+\phi_H)}_{\text{direct effects}} + \underbrace{\quad}_{\text{indirect effects}} > 1$$

Proposition (explosive systems).
System explodes in finite time iff the interaction matrix, $\begin{bmatrix} 1+\phi_S & \lambda_S \\ \lambda_H & 1+\phi_H \end{bmatrix}$, has an eigenvalue > 1.

A networked semi-endogenous growth model: technological spillovers

$$\dot{S}_t = S_t^{1+\phi_S} H^{\lambda_S}$$
$$\dot{H}_t = H_t^{1+\phi_H} S^{\lambda_H}$$
$$Y_t = (S_t H_t)^{1/2} L_t^{\alpha}$$



Economic singularity condition:

$$\phi_S \text{ or } \phi_H > 0$$

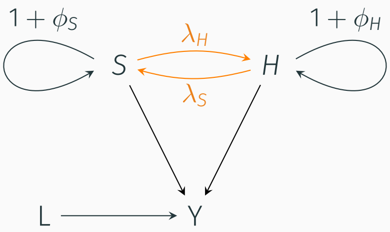
or

$$\underbrace{(1+\phi_S)+(1+\phi_H)-(1+\phi_S)(1+\phi_H)}_{\text{direct effects}} + \underbrace{\qquad}_{\text{indirect effects}} > 1$$

Proposition (explosive systems). System explodes in finite time iff the interaction matrix, $\begin{bmatrix} 1+\phi_S & \lambda_S \\ \lambda_H & 1+\phi_H \end{bmatrix}$, has an eigenvalue > 1.

A networked semi-endogenous growth model: technological spillovers

$$\dot{S}_t = S_t^{1+\phi_S} H^{\lambda_S}$$
$$\dot{H}_t = H_t^{1+\phi_H} S^{\lambda_H}$$
$$Y_t = (S_t H_t)^{1/2} L_t^{\alpha}$$



Economic singularity condition:

$$\phi_S \text{ or } \phi_H > 0$$

or

$$\underbrace{(1+\phi_S)+(1+\phi_H)-(1+\phi_S)(1+\phi_H)}_{\text{direct effects}} + \underbrace{\lambda_S \cdot \lambda_H}_{\text{indirect effects}} > 1$$

Proposition (explosive systems). System explodes in finite time iff the interaction matrix, $\begin{bmatrix} 1+\phi_S & \lambda_S \\ \lambda_H & 1+\phi_H \end{bmatrix}$, has an eigenvalue > 1.

1. Building blocks of the model

Lesson 1: diminishing returns prevent growth explosions

Lesson 2: technological feedback loops create spillovers across sectors, offsetting diminishing returns

Lesson 3: *economic* feedback loops create spillovers across sectors, offsetting diminishing returns

Lesson 4: automation offsets diminishing returns

Lesson 6 (empirical): diminishing returns are low in software/hardware

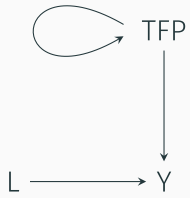
2. The software-hardware model

3. Scope of claims

Accumulable factors create economic feedback loops

$$\dot{A}_t = A_t^{1+\phi}$$

$$Y_t = A_t L_t^{\alpha} K_t^{\beta}$$

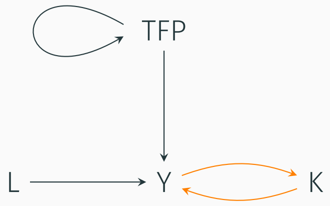


Accumulable factors create economic feedback loops

$$\dot{A}_t = A_t^{1+\phi}$$

$$Y_t = A_t L_t^{\alpha} K_t^{\beta}$$

$$\dot{K}_t = s_K Y_t - \delta K_t$$



Accumulable factors **create** economic feedback loops

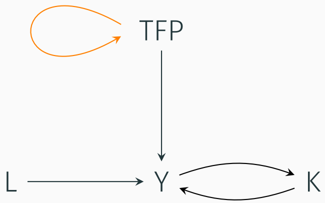
$$\dot{A}_t = A_t^{1+\phi}$$

$$Y_t = A_t L_t^{\alpha} K_t^{\beta}$$

$$\dot{K}_t = s_K Y_t - \delta K_t$$

Economic singularity condition:

$$\phi > 0$$



Accumulable factors create economic feedback loops

$$\dot{A}_t = A_t^{1+\phi}$$

$$Y_t = A_t L_t^{\alpha} K_t^{\beta}$$

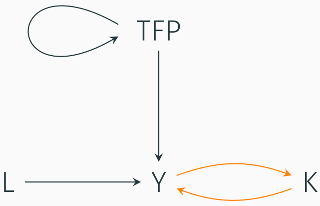
$$\dot{K}_t = s_K Y_t - \delta K_t$$

Economic singularity condition:

$$\phi > 0$$

or

$$\beta > 1$$

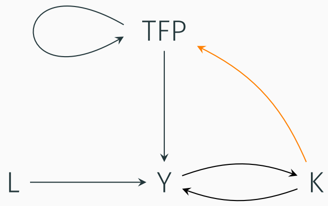


Accumulable factors create economic feedback loops

$$\dot{A}_t = A_t^{1+\phi}(\kappa K_t)^{\lambda}$$

$$Y_t = A_t L_t^{\alpha}((1-\kappa)K_t)^{\beta}$$

$$\dot{K}_t = s_K Y_t - \delta K_t$$

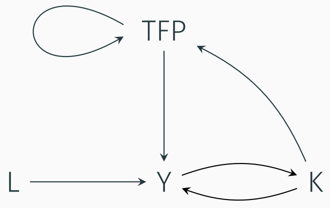


Accumulable factors create economic feedback loops

$$\dot{A}_t = A_t^{1+\phi}(\kappa K_t)^{\lambda}$$
$$Y_t = A_t L_t^{\alpha}\left((1-\kappa)K_t\right)^{\beta}$$
$$\dot{K}_t = s_K Y_t - \delta K_t$$

Focusing on accumulable factors:

$$\dot{A}_t = \text{stuff} \cdot A_t^{1+\phi} K_t^{\lambda}$$
$$\dot{K}_t = \text{stuff} \cdot A_t K_t^{\beta}$$

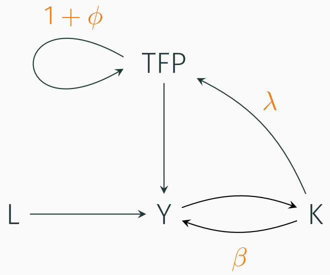


Accumulable factors **create** economic feedback loops

$$\dot{A}_t = A_t^{1+\phi}(\kappa K_t)^{\lambda}$$
$$Y_t = A_t L_t^{\alpha}\left((1-\kappa)K_t\right)^{\beta}$$
$$\dot{K}_t = s_K Y_t - \delta K_t$$

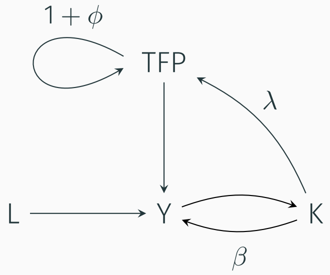
Focusing on accumulable factors:

$$\dot{A}_t = \text{stuff} \cdot A_t^{1+\phi} K_t^{\lambda}$$
$$\dot{K}_t = \text{stuff} \cdot A_t K_t^{\beta}$$



Accumulable factors create economic feedback loops

$$\dot{A}_t = A_t^{1+\phi}(\kappa K_t)^\lambda$$
$$Y_t = A_t L_t^\alpha ((1-\kappa)K_t)^\beta$$
$$\dot{K}_t = s_K Y_t - \delta K_t$$



Focusing on accumulable factors:

$$\dot{A}_t = \text{stuff} \cdot A_t^{1+\phi} K_t^\lambda$$
$$\dot{K}_t = \text{stuff} \cdot A_t K_t^\beta$$

Proposition (explosive systems).
System explodes in finite time if the exponent matrix, $\begin{bmatrix} 1+\phi & \lambda \\ 1 & \beta \end{bmatrix}$, has an eigenvalue > 1.

Explosion conditions:

$$\phi > 0 \text{ or } \beta > 1$$

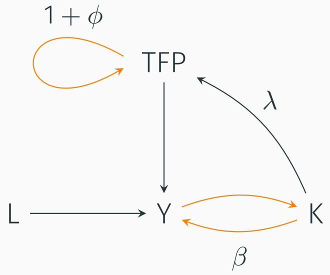
Accumulable factors create economic feedback loops

$$\dot{A}_t = A_t^{1+\phi}(\kappa K_t)^\lambda$$
$$Y_t = A_t L_t^\alpha \left((1-\kappa)K_t\right)^\beta$$
$$\dot{K}_t = s_K Y_t - \delta K_t$$

Focusing on accumulable factors:

$$\dot{A}_t = \text{stuff} \cdot A_t^{1+\phi} K_t^\lambda$$
$$\dot{K}_t = \text{stuff} \cdot A_t K_t^\beta$$

Proposition (explosive systems). System explodes in finite time if the exponent matrix, $\begin{bmatrix} 1+\phi & \lambda \\ 1 & \beta \end{bmatrix}$, has an eigenvalue > 1.



Explosion conditions:

$$\phi > 0 \text{ or } \beta > 1$$
$$\underbrace{(1+\phi)+\beta}_{\text{direct effects}} > 1$$

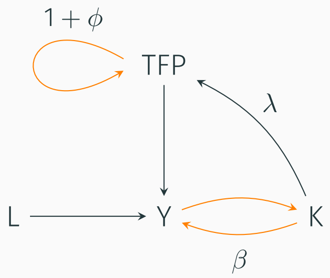
Accumulable factors create economic feedback loops

$$\dot{A}_t = A_t^{1+\phi}(\kappa K_t)^\lambda$$
$$Y_t = A_t L_t^\alpha \left((1-\kappa)K_t\right)^\beta$$
$$\dot{K}_t = s_K Y_t - \delta K_t$$

Focusing on accumulable factors:

$$\dot{A}_t = \text{stuff} \cdot A_t^{1+\phi} K_t^\lambda$$
$$\dot{K}_t = \text{stuff} \cdot A_t K_t^\beta$$

Proposition (explosive systems).
System explodes in finite time if the exponent matrix, $\begin{bmatrix} 1+\phi & \lambda \\ 1 & \beta \end{bmatrix}$, has an eigenvalue > 1.



Explosion conditions:

$$\phi > 0 \text{ or } \beta > 1$$
$$\underbrace{(1+\phi)+\beta}_{\text{direct effects}} - (1+\phi)\beta > 1$$

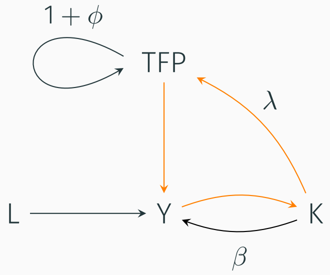
Accumulable factors **create** economic feedback loops

$$\dot{A}_t = A_t^{1+\phi}(\kappa K_t)^{\lambda}$$
$$Y_t = A_t L_t^{\alpha}\left((1-\kappa)K_t\right)^{\beta}$$
$$\dot{K}_t = s_K Y_t - \delta K_t$$

Focusing on accumulable factors:

$$\dot{A}_t = \text{stuff} \cdot A_t^{1+\phi} K_t^{\lambda}$$
$$\dot{K}_t = \text{stuff} \cdot A_t K_t^{\beta}$$

Proposition (explosive systems). System explodes in finite time if the exponent matrix, $\begin{bmatrix} 1+\phi & \lambda \\ 1 & \beta \end{bmatrix}$, has an eigenvalue > 1.



Explosion conditions:

$$\phi > 0 \text{ or } \beta > 1$$

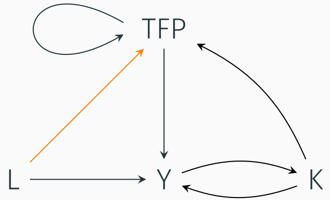
$$\underbrace{(1+\phi)+\beta}_{\text{direct effects}} - (1+\phi)\beta + \underbrace{\lambda \cdot 1}_{\text{indirect effects}} > 1$$

The canonical semi-endogenous growth model

$$\dot{A}_t = A_t^{1+\phi}(\ell L_t)^{\lambda}(\kappa K_t)^{\lambda}$$

$$Y_t = A_t\left((1-\ell)L_t\right)^{\alpha}\left((1-\kappa)K_t\right)^{\beta}$$

$$\dot{K}_t = s_K Y_t - \delta K_t$$

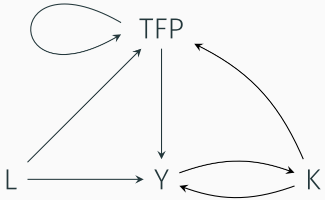


The canonical semi-endogenous growth model

$$\dot{A}_t = A_t^{1+\phi}(\ell L_t)^{\lambda}(\kappa K_t)^{\lambda}$$

$$Y_t = A_t\left((1-\ell)L_t\right)^{\alpha}\left((1-\kappa)K_t\right)^{\beta}$$

$$\dot{K}_t = s_K Y_t - \delta K_t$$



Best guess calibration:

- $\phi = -3.4$ (Bloom et al 2020) — $\phi > 0$ ✗
- $\beta = 0.4$ (capital share in production) — $\beta > 1$ ✗
- $\lambda = 0.1$ (capital share in R&D) — $(1+\phi)+\beta-(1+\phi)\beta+\lambda > 1$ ✗

1. Building blocks of the model

Lesson 1: diminishing returns prevent growth explosions

Lesson 2: technological feedback loops create spillovers across sectors, offsetting diminishing returns

Lesson 3: *economic* feedback loops create spillovers across sectors, offsetting diminishing returns

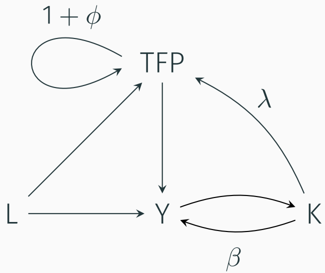
Lesson 4: automation offsets diminishing returns

Lesson 6 (empirical): diminishing returns are low in software/hardware

2. The software-hardware model

3. Scope of claims

$$\dot{A}_t = A_t^{1+\phi}(\ell_A L_t)^\lambda(\kappa_A K_t)^\lambda$$
$$Y_t = A_t\,(\ell_Y L_t)^\alpha\,(\kappa_Y K_t)^\beta$$
$$\dot{K}_t = s_K Y_t - \delta K_t$$



Explosion conditions:

$$\phi > 0$$
$$\beta > 1$$
$$(1+\phi) + \beta - (1+\phi)\beta + \lambda > 1$$

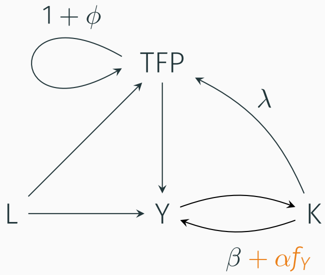
Introducing automation: offsets diminishing returns

[Aghion, Jones, and Jones (2019)]

$$\dot{A}_t = A_t^{1+\phi}(\ell_A L_t)^\lambda(\kappa_A K_t)^\lambda$$

$$Y_t = A_t\,(\ell_Y L_t)^{\alpha(1-f_Y)}\,(\kappa_Y K_t)^{\beta+f_Y\alpha}$$

$$\dot{K}_t = s_K Y_t - \delta K_t$$



Explosion conditions:

$$\phi > 0$$

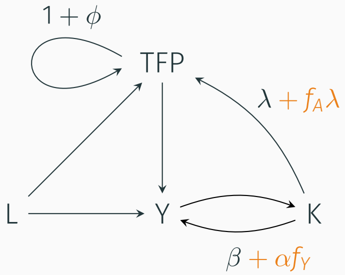
$$\beta + \alpha f_Y > 1$$

$$(1+\phi) + \beta + \alpha f_Y - (1+\phi)(\beta + \alpha f_Y) + \lambda > 1$$

Introducing automation: offsets diminishing returns

[Aghion, Jones, and Jones (2019)]

$$\dot{A}_t = A_t^{1+\phi}(\ell_A L_t)^{\lambda(1-f_A)}(\kappa_A K_t)^{\lambda+f_A\lambda}$$
$$Y_t = A_t\,(\ell_Y L_t)^{\alpha(1-f_Y)}\,(\kappa_Y K_t)^{\beta+f_Y\alpha}$$
$$\dot{K}_t = s_K Y_t - \delta K_t$$



Explosion conditions:

$$\phi > 0$$
$$\beta + \alpha f_Y > 1$$
$$(1+\phi) + \beta + \alpha f_Y - (1+\phi)(\beta + \alpha f_Y) + \lambda + f_A\lambda > 1$$

Takeaways

Takeaways:

0. Where are the feedback loops?

Takeaways:

0. Where are the feedback loops?
1. How strong are the diminishing returns?

Takeaways:

0. Where are the feedback loops?
1. How strong are the diminishing returns?
2. How strong are the technological spillovers?

CHM25 policy

Takeaways:

0. Where are the feedback loops?
1. How strong are the diminishing returns?
2. How strong are the technological spillovers?
3. What are the accumulable factors and how strong are the economic feedback loops?

Takeaways:

0. Where are the feedback loops?
1. How strong are the diminishing returns?
2. How strong are the technological spillovers?
3. What are the accumulable factors and how strong are the economic feedback loops?
4. Automation offsets diminishing returns

1. Building blocks of the model

Lesson 1: diminishing returns prevent growth explosions

Lesson 2: technological feedback loops create spillovers across sectors, offsetting diminishing returns

Lesson 3: *economic* feedback loops create spillovers across sectors, offsetting diminishing returns

Lesson 4: automation offsets diminishing returns

Lesson 6 (empirical): diminishing returns are low in software/hardware

2. The software-hardware model

3. Scope of claims

Lesson 6 (empirical): diminishing returns are much lower in software/hardware

1. Diminishing returns in general economy (Bloom et al 2020): $\phi = -3$

Lesson 6 (empirical): diminishing returns are much lower in software/hardware

1. Diminishing returns in general economy (Bloom et al 2020): $\phi = -3$
2. Diminishing returns in hardware (Bloom et al 2020): $\phi = -0.2$

Lesson 6 (empirical): diminishing returns are much lower in software/hardware

1. Diminishing returns in general economy (Bloom et al 2020): $\phi = -3$
2. Diminishing returns in hardware (Bloom et al 2020): $\phi = -0.2$
3. Diminishing returns in software (Erdil et al 2024; Ho and Whitfill 2025): $\phi = -1$

1. Building blocks of the model

Lesson 1: diminishing returns prevent growth explosions

Lesson 2: technological feedback loops create spillovers across sectors, offsetting diminishing returns

Lesson 3: *economic* feedback loops create spillovers across sectors, offsetting diminishing returns

Lesson 4: automation offsets diminishing returns

Lesson 6 (empirical): diminishing returns are low in software/hardware

2. The software-hardware model

3. Scope of claims

Software-hardware model: overview

Canonical semi-endogenous growth model, plus:

Software-hardware model: overview

Canonical semi-endogenous growth model, plus:

1. Automation of labor with “AI”

Software-hardware model: overview

Canonical semi-endogenous growth model, plus:

1. Automation of labor with “AI”
2. AI = software · hardware · hardware quality

The software-hardware model of AI

The software-hardware model of AI

AI substituting for labor:

$$\text{AI} \equiv Z = \underbrace{S}_{\text{software}} \cdot \underbrace{C}_{\text{hardware}}$$

The software-hardware model of AI

AI substituting for labor:

$$\text{AI} \equiv Z = \underbrace{S}_{\text{software}} \cdot \underbrace{C}_{\text{hardware}}$$

- **Software:** “algorithmic efficiency”

The software-hardware model of AI

AI substituting for labor:

$$\text{AI} \equiv Z = \underbrace{S}_{\text{software}} \cdot \underbrace{C}_{\text{hardware}}$$

- **Software:** "algorithmic efficiency"
- **Hardware:** computer hardware ("compute")

The software-hardware model of AI

AI substituting for labor:

$$\text{AI} \equiv Z = \underbrace{S}_{\text{software}} \cdot \underbrace{C}_{\text{hardware}} = \underbrace{S}_{\text{software}} \cdot \underbrace{c \cdot h}_{\text{hardware}}$$

- **Software:** “algorithmic efficiency”
- **Hardware:** computer hardware (“compute”)
  - Hardware quantity: c, “number of computer chips”
  - Hardware quality: h, “how many calculations (FLOPs) per chip”

Software and hardware evolution

Hardware accumulates: just another form of capital

$$\dot{C}_t = s_C Y_t - \delta_C C_t$$

Software and hardware evolution

Hardware accumulates: just another form of capital

$$C_t = s_C Y_t - \delta_C C_t$$

Software is like ideas: better software allows for *faster software progress*

$$\dot{S}_t = (\ell_S L_t)^{\lambda_S} S_t^{1+\phi_S}$$

Software and hardware evolution

Hardware accumulates: just another form of capital

$$C_t = s_C Y_t - \delta_C C_t$$

Software is like ideas: better software allows for *faster software progress*

$$\dot{S}_t = (\ell_S L_t)^{\lambda_S} S_t^{1+\phi_S}$$

Remember ideas production function:

$$\dot{A}_t = (\ell_A L_t)^{\lambda_A} A_t^{1+\phi_A}$$

Software and hardware evolution

Hardware accumulates: just another form of capital

$$\dot{C}_t = h_t s_C Y_t - \delta_C C_t$$

Software is like ideas: better software allows for *faster software progress*

$$\dot{S}_t = (\ell_S L_t)^{\lambda_S} S_t^{1+\phi_S}$$

Hardware quality is like ideas and investment-specific technical change: better hardware quality allows for *faster accumulation of effective hardware*

[a la Greenwood-Hercowitz-Krusell]

$$\dot{h} = (\ell_h L_t)^{\lambda_h} h_t^{1+\phi_h}$$

Software and hardware evolution

Hardware accumulates: just another form of capital

$$\dot{C}_t = h_t s_C Y_t - \delta_C C_t$$

Software is like ideas: better software allows for *faster software progress*

$$\dot{S}_t = (\ell_S L_t)^{\lambda_S} S_t^{1+\phi_S}$$

Hardware quality is like ideas and investment-specific technical change: better hardware quality allows for *faster accumulation of effective hardware*

[a la Greenwood-Hercowitz-Krusell]

$$\dot{h} = (\ell_h L_t)^{\lambda_h} h_t^{1+\phi_h}$$

AI: $\mathrm{AI} = \underbrace{S}_{\text{software}} \cdot \underbrace{c \cdot h}_{\text{hardware}}$

AI replaces human labor in tasks

AI replaces human labor in tasks

Automation by AI: AI AI replaces human labor in some fraction of economic tasks, f_x, in sector x.

AI replaces human labor in tasks

Automation by AI: AI AI replaces human labor in some fraction of economic tasks, f_x, in sector x.

Labor in sector X: *(without automation)*

$$L_{x,t} = \ell_x L_t$$

Effective labor in sector X: *(with automation)*

$$\hat{L}_{x,t} = (\ell_x L_t)^{1-f_x} \cdot Z_{x,t}{}^{f_x}$$

AI replaces human labor in tasks

Automation by AI: AI AI replaces human labor in some fraction of economic tasks, f_x, in sector x.

Labor in sector X: *(without automation)*

$$L_{x,t} = \ell_x L_t$$

Effective labor in sector X: *(with automation)*

$$\hat{L}_{x,t} = (\ell_x L_t)^{1-f_x} \cdot Z_{x,t}{}^{f_x}$$

$$= (\ell_x L_t)^{1-f_x} \cdot \left(\underbrace{S_t}_{\text{software}} \cdot \underbrace{c_{x,t} \cdot h_t}_{\text{hardware}} \right)^{f_x}$$

Note: effective labor accumulates

The software-hardware model: equations

Output: $Y_t = A_t \hat{L}_{Y,t}^{\alpha} K_t^{\beta}$

The software-hardware model: equations

Output: $$Y_t = A_t \hat{L}_{Y,t}^{\alpha} K_t^{\beta}$$

Accumulable factors: $$\dot{K}_t = s_K Y_t - \delta_K K_t$$

$$\dot{c}_t = h_t s_c Y_t - \delta_c c_t$$

The software-hardware model: equations

Output: $$Y_t = A_t \hat{L}_{Y,t}^{\alpha} K_t^{\beta}$$

Accumulable factors: $$\dot{K}_t = s_K Y_t - \delta_K K_t$$

$$\dot{c}_t = h_t s_c Y_t - \delta_c c_t$$

Ideas: $$\dot{A}_t = \hat{L}_{A,t}^{\lambda_A} A_t^{1+\phi_A}$$

$$\dot{S}_t = \hat{L}_{S,t}^{\lambda_S} S_t^{1+\phi_S}$$

$$\dot{h}_t = \hat{L}_{h,t}^{\lambda_S} h_t^{1+\phi_h}$$

The software-hardware model: equations

Output: $$Y_t = A_t \hat{L}_{Y,t}^{\alpha} K_t^{\beta}$$

Accumulable factors: $$\dot{K}_t = s_K Y_t - \delta_K K_t$$

$$\dot{c}_t = h_t s_c Y_t - \delta_c c_t$$

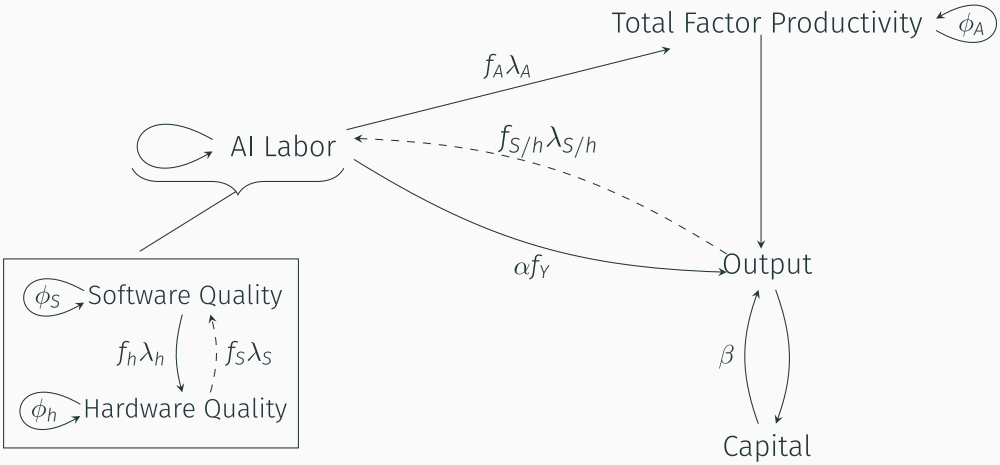
Ideas: $$\dot{A}_t = \hat{L}_{A,t}^{\lambda_A} A_t^{1+\phi_A}$$

$$\dot{S}_t = \hat{L}_{S,t}^{\lambda_S} S_t^{1+\phi_S}$$

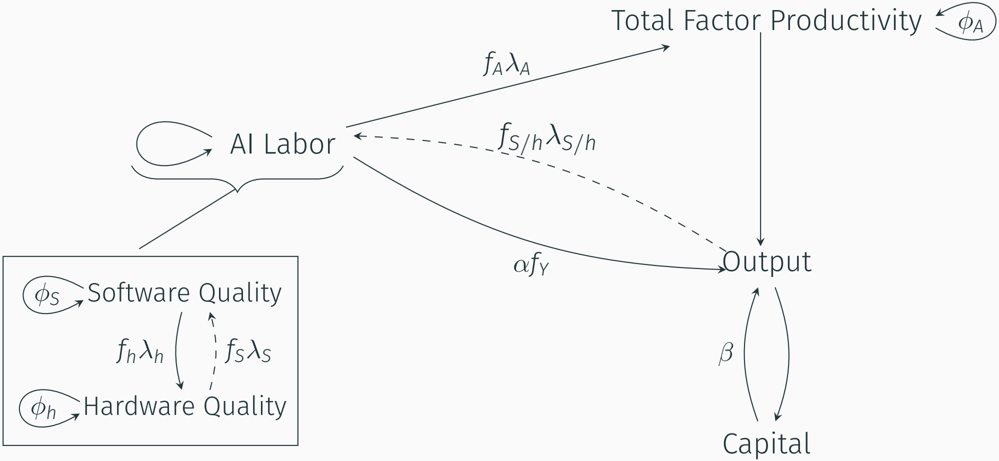
$$\dot{h}_t = \hat{L}_{h,t}^{\lambda_S} h_t^{1+\phi_h}$$

AI automation: $$\hat{L}_{x,t} = L_{x,t}^{1-f_x} \cdot (S_t \cdot c_{x,t} \cdot h_t)^{f_x}$$

The software-hardware model: diagram



The software-hardware model: diagram



Strength of feedback increasing with all exponents

Explosion condition

Simplify the problem by assuming complete depreciation. Substituting in effective labor expressions and removing non-accumulable factors

$$\dot{S}_t \propto S_t^{f_S \lambda_S \frac{1-\beta}{1-f_Y\alpha-\beta}+1+\phi_S} h_t^{f_S \lambda_S \frac{1-\beta}{1-f_Y\alpha-\beta}} A_t^{\frac{f_S \lambda_S}{1-f_Y\alpha-\beta}}$$

$$\dot{h}_t \propto S_t^{f_h \lambda_h \frac{1-\beta}{1-f_Y\alpha-\beta}} h_t^{f_h \lambda_h \frac{1-\beta}{1-f_Y\alpha-\beta}+1+\phi_h} A_t^{\frac{f_h \lambda_h}{1-f_Y\alpha-\beta}}$$

$$\dot{A}_t \propto S_t^{f_A \lambda_A \frac{1-\beta}{1-f_Y\alpha-\beta}} h_t^{f_A \lambda_A \frac{1-\beta}{1-f_Y\alpha-\beta}} A_t^{\frac{f_A \lambda_A}{1-f_Y\alpha-\beta}+1+\phi_A}$$

Explosion condition

Simplify the problem by assuming complete depreciation. Substituting in effective labor expressions and removing non-accumulable factors

$$\dot{S}_t \propto S_t^{f_S \lambda_S \frac{1-\beta}{1-f_Y\alpha-\beta}+1+\phi_S} h_t^{f_S \lambda_S \frac{1-\beta}{1-f_Y\alpha-\beta}} A_t^{\frac{f_S \lambda_S}{1-f_Y\alpha-\beta}}$$

$$\dot{h}_t \propto S_t^{f_h \lambda_h \frac{1-\beta}{1-f_Y\alpha-\beta}} h_t^{f_h \lambda_h \frac{1-\beta}{1-f_Y\alpha-\beta}+1+\phi_h} A_t^{\frac{f_h \lambda_h}{1-f_Y\alpha-\beta}}$$

$$\dot{A}_t \propto S_t^{f_A \lambda_A \frac{1-\beta}{1-f_Y\alpha-\beta}} h_t^{f_A \lambda_A \frac{1-\beta}{1-f_Y\alpha-\beta}} A_t^{\frac{f_A \lambda_A}{1-f_Y\alpha-\beta}+1+\phi_A}$$

Applying explosion proposition yields **explosion threshold**:

$$\frac{1}{1-\beta} f_A r_A + \frac{\alpha}{1-\beta} f_Y + f_S r_S + f_h r_h > 1$$

Explosion condition

Applying explosion proposition yields **explosion threshold**:

$$\frac{1}{1-\beta} f_A r_A + \frac{\alpha}{1-\beta} f_Y + f_S r_S + f_h r_h > 1$$

Explosion condition

Applying explosion proposition yields **explosion threshold**:

$$\frac{1}{1-\beta} f_A r_A + \frac{\alpha}{1-\beta} f_Y + f_S r_S + f_h r_h > 1$$

r **factor:** for $x \in \{A, S, h\}$,

$$r_x \equiv \frac{\lambda_x}{-\phi_x}$$

- Intuition: in canonical model, $g_A = r_A \cdot$ population growth

Calibrating parameters

Explosion condition: $\frac{1}{1-\beta} f_A r_A + \frac{\alpha}{1-\beta} f_Y + f_S r_S + f_h r_h > 1$

Term	Parameter	Estimate	Source
Labor share	$\alpha \equiv 1-\beta$	0.6	–

Calibrating parameters

Explosion condition: $\frac{1}{1-\beta} f_A r_A + \frac{\alpha}{1-\beta} f_Y + f_S r_S + f_h r_h > 1$

Term	Parameter	Estimate	Source
Labor share	$\alpha \equiv 1 - \beta$	0.6	–
TFP r-factor	r_A	0.32	Bloom et al (2020)

Calibrating parameters

Explosion condition: $\frac{1}{1-\beta} f_A r_A + \frac{\alpha}{1-\beta} f_Y + f_S r_S + f_h r_h > 1$

Term	Parameter	Estimate	Source
Labor share	$\alpha \equiv 1 - \beta$	0.6	–
TFP r-factor	r_A	0.32	Bloom et al (2020)
Hardware r-factor	r_h (Moore's law)	5	Bloom et al (2020)

Calibrating parameters

Explosion condition: $\frac{1}{1-\beta} f_A r_A + \frac{\alpha}{1-\beta} f_Y + f_S r_S + f_h r_h > 1$

Term	Parameter	Estimate	Source
Labor share	$\alpha \equiv 1 - \beta$	0.6	–
TFP r-factor	r_A	0.32	Bloom et al (2020)
Hardware r-factor	r_h (Moore's law)	5	Bloom et al (2020)
Software r-factor	r_S	$\approx$1	Erdil et al (2024) & Ho and Whitfill (2025)

Calibrating parameters

Explosion condition: $0.5f_A + f_Y + f_S + 5f_h > 1$

Term	Parameter	Estimate	Source
Labor share	$\alpha \equiv 1 - \beta$	0.6	–
TFP *r*-factor	r_A	0.32	Bloom et al (2020)
Hardware *r*-factor	r_h (Moore's law)	5	Bloom et al (2020)
Software *r*-factor	r_S	≈1	Erdil et al (2024) & Ho and Whitfill (2025)

Calibrating parameters

Explosion condition: $0.5f_A + f_Y + f_S + 5f_h > 1$

Term	Parameter	Estimate	Source
Labor share	$\alpha \equiv 1 - \beta$	0.6	–
TFP r-factor	r_A	0.32	Bloom et al (2020)
Hardware r-factor	r_h (Moore's law)	5	Bloom et al (2020)
Software r-factor	r_S	$\approx$1	Erdil et al (2024) & Ho and Whitfill (2025)

Interpretation: Software and hardware have **much lower** diminishing returns to research than the rest of the economy

Calibrating parameters

Explosion condition: $0.5f_A + f_Y + f_S + 5f_h > 1$

Term	Parameter	Estimate	Source
Labor share	$\alpha \equiv 1 - \beta$	0.6	–
TFP r-factor	r_A	0.32	Bloom et al (2020)
Hardware r-factor	r_h (Moore's law)	5	Bloom et al (2020)
Software r-factor	r_S	$\approx$1	Erdil et al (2024) & Ho and Whitfill (2025)

Interpretation: Software and hardware have **much lower** diminishing returns to research than the rest of the economy $\implies$ if software/hardware grow as share of economy, large growth effects

1. **Building blocks of the model**

   Lesson 1: diminishing returns prevent growth explosions

   Lesson 2: technological feedback loops create spillovers across sectors, offsetting diminishing returns

   Lesson 3: *economic* feedback loops create spillovers across sectors, offsetting diminishing returns

   Lesson 4: automation offsets diminishing returns

   Lesson 6 (empirical): diminishing returns are low in software/hardware

2. **The software-hardware model**

3. **Scope of claims**

What we do *not* speak to

1. **No bottlenecks** (e.g. compute, data)
   - Cobb-Douglas technology $\implies$ one thing can always substitute for another

What we do *not* speak to

1. **No bottlenecks** (e.g. compute, data)
   - Cobb-Douglas technology $\implies$ one thing can always substitute for another
2. **What about "non-explosive" growth accelerations?**

What we do *not* speak to

1. **No bottlenecks** (e.g. compute, data)
    - Cobb-Douglas technology $\implies$ one thing can always substitute for another
2. **What about "non-explosive" growth accelerations?**
3. **Quality of parameter value estimates**

What we do *not* speak to

1. **No bottlenecks** (e.g. compute, data)
    - Cobb-Douglas technology $\implies$ one thing can always substitute for another
2. **What about "non-explosive" growth accelerations?**
3. **Quality of parameter value estimates**
4. **Endogenous automation**

What we do *not* speak to

1. **No bottlenecks** (e.g. compute, data)
   - Cobb-Douglas technology $\implies$ one thing can always substitute for another
2. **What about "non-explosive" growth accelerations?**
3. **Quality of parameter value estimates**
4. **Endogenous automation**
5. **More:**
   - Endogenous allocation rules
   - Decentralized allocation: roles of industrial organization + externalities
   - Learning by doing
   - Capital adjustment costs
   - Time to build

How much automation is necessary for a growth explosion?

CHM25 policy

Automated Factors	Automation Threshold $2 \times g_Y^{BGP}$
S	–
H	–
S, H	–
$H,\ Y$	12%
$S,\ Y$	16%
$S,\ H,\ Y$	11%
$S,\ H,\ A,\ Y$	8%
$S,\ H,\ A,\ Y$ (10% fixed factor)	9%

How much automation is necessary for a growth explosion?

▸ CHM25 ▸ policy

Automated Factors	Automation Threshold	
	$2 \times g_Y^{BGP}$	$\infty \times g^{BGP}$ (Hyperbolic)
S	–	
H	–	
S, H	–	
$H,\ Y$	12%	
$S,\ Y$	16%	
$S,\ H,\ Y$	11%	
$S,\ H,\ A,\ Y$	8%	
$S,\ H,\ A,\ Y$ (10% fixed factor)	9%	

How much automation is necessary for a growth explosion?

▸ CHM25 ▸ policy

Automated Factors	Automation Threshold	
	$2 \times g_Y^{BGP}$	$\infty \times g^{BGP}$ (Hyperbolic)
S	–	$\sim 100\%$
H	–	20%
S, H	–	17%
$H,\ Y$	12%	17%
$S,\ Y$	16%	50%
$S,\ H,\ Y$	11%	14%
$S,\ H,\ A,\ Y$	8%	13%
$S,\ H,\ A,\ Y$ (10% fixed factor)	9%	14%

Conclusion: "Why wasn't automating agriculture enough for a growth explosion?"

Tractors $\rightarrow$ more food $\rightarrow$ more people $\rightarrow$ better tractors $\rightarrow \cdots$

Conclusion: "Why wasn't automating agriculture enough for a growth explosion?"

Tractors $\rightarrow$ more food $\rightarrow$ more people $\rightarrow$ better tractors $\rightarrow \cdots$

1. Maybe it was? Our condition speaks to 'are we *on track*' for a growth explosion

Conclusion: "Why wasn't automating agriculture enough for a growth explosion?"

Tractors → more food → more people → better tractors → · · ·

1. Maybe it was? Our condition speaks to 'are we *on track*' for a growth explosion

2. 'Diminishing returns' is one reason; diminishing returns are *less* strong in hardware and software

Conclusion: "Why wasn't automating agriculture enough for a growth explosion?"

Tractors $\rightarrow$ more food $\rightarrow$ more people $\rightarrow$ better tractors $\rightarrow \cdots$

1. Maybe it was? Our condition speaks to 'are we *on track*' for a growth explosion

2. 'Diminishing returns' is one reason; diminishing returns are *less* strong in hardware and software

3. Bottlenecks or other limits: we do not speak to *all* limits

Thank you!

Appendix

On bottlenecks

Cobb-Douglas: with $\alpha > 0$

$$Y = L^{\alpha} K^{1-\alpha}$$

Fix L, send $K \to \infty \implies Y \to \infty$.

Potential bottlenecks:

- **Compute** bottlenecking algorithmic progress
- **Algorithmic progress** bottlenecking compute
- **Energy** bottlenecking everything
- **Data** bottlenecking everything

CES with complements: with $\phi < 0$

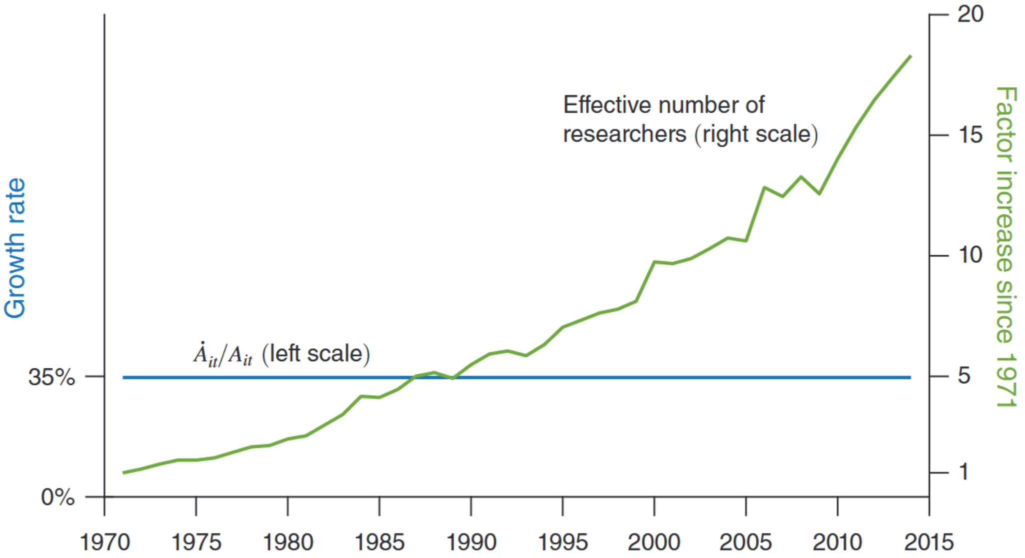
$$Y = \left[L^{\phi} + K^{\phi}\right]^{1/\phi}$$

Fix L, send $K \to \infty \implies Y = L$

Potential reasons to think bottlenecks will be less of an issue:

- 2x efficient algorithims $\implies$ 2x as many experiments
- Aum and Shin (2024): software and labor are substitutes not complements

Could ϕ be falling over time? Doesn't appear to be for Moore's Law



Multisector semi-endogenous growth model

Standard one-sector model:

- Idea production functions:

$$\dot{A}_t = L_t^{\lambda} A_t^{1+\phi}$$

- BGP:

$$\frac{\dot{A}}{A} = \frac{\lambda}{-\phi} n$$

Multisector semi-endogenous growth model

Standard one-sector model:

- Idea production functions:

$$\dot{A}_t = L_t^{\lambda} A_t^{1+\phi}$$

- BGP:

$$\frac{\dot{A}}{A} = \frac{\lambda}{-\phi} n$$

Two-sector model:

- Aggregate TFP: $A_t = A_{1t}^{\sigma_1} A_{2t}^{\sigma_2}$
- Idea production functions:*

$$\dot{A}_{it} = (s_i L_t)^{\lambda_i} A_{it}^{1+\phi_i}$$

* s_i exogenous and constant ("Solow-style"). It can be shown, though, that optimally s_1/s_2 is constant under Cobb-Douglas aggregation.

Multisector semi-endogenous growth model

Standard one-sector model:

- Idea production functions:

$$\dot{A}_t = L_t^{\lambda} A_t^{1+\phi}$$

- BGP:

$$\frac{\dot{A}}{A} = \frac{\lambda}{-\phi} n$$

Two-sector model:

- Aggregate TFP: $A_t = A_{1t}^{\sigma_1} A_{2t}^{\sigma_2}$
- Idea production functions:*

$$\dot{A}_{it} = (s_i L_t)^{\lambda_i} A_{it}^{1+\phi_i}$$

- BGP:

$$\frac{\dot{A}}{A} = \sum_i \left[\sigma_i \frac{\lambda_i}{-\phi_i} n \right]$$

* s_i exogenous and constant ("Solow-style"). It can be shown, though, that optimally s_1/s_2 is constant under Cobb-Douglas aggregation.

Multisector semi-endogenous growth model

Standard one-sector model:

- Idea production functions:

$$\dot{A}_t = L_t^{\lambda} A_t^{1+\phi}$$

- BGP:

$$\frac{\dot{A}}{A} = \frac{\lambda}{-\phi} n$$

Two-sector model:

- Aggregate TFP: $A_t = A_{1t}^{\sigma_1} A_{2t}^{\sigma_2}$
- Idea production functions:*

$$\dot{A}_{it} = (s_i L_t)^{\lambda_i} A_{it}^{1+\phi_i}$$

- BGP:

$$\frac{\dot{A}}{A} = \sum_i \left[\sigma_i \frac{\lambda_i}{-\phi_i} n \right]$$

Comparative static: Suppose $-\phi_1 > -\phi_2$. Increase σ_2. Obviously $g_A \uparrow$

---

* s_i exogenous and constant ("Solow-style"). It can be shown, though, that optimally s_1/s_2 is constant under Cobb-Douglas aggregation.