
Why This Time in AI May Be Different

Odysseus Kaloidas
UC Berkeley

Abstract

AI could help build the tools and working practices needed to make AI itself useful. Could that change the pace and reach of technological progress? Earlier inventions required investment in implementation, training, and reorganization before their benefits appeared. Assistance across those activities might make more projects feasible. Evidence shows gains in some work settings and losses in others. The question is how broadly AI can improve the surrounding process, including its bottlenecks, rather than merely accelerate one visible task.

1 A Technology That Helps Build Its Complements

A factory does not gain everything electricity can offer merely by replacing its old power source with an electric motor. Reorganizing production around motors distributed through the factory can open further possibilities. David's history of electrification draws out this difference [9]. Could AI help carry out the comparable work needed to make AI useful? That would give it a role in building the conditions for its own success.

Economists call such surrounding changes complementary innovation: new ways of working that make the original technology more valuable [4]. AI could assist this work through implementation, interpretation, and tool construction. Writing code for a successor model is one instance. Selecting an informative experiment, understanding its result, or making verification cheaper would extend the effect further.

Would help across several stages change the pattern? Saving time on one activity is less valuable if the next activity consumes all of it. Helping with several stages could make entire projects feasible for more people, even where no single task becomes dramatically faster.

2 The Cost of Complementary Work

Brynjolfsson and Hitt connect information technology's business value to investment in the organization using it [5]. The productivity J-curve describes why gains may initially be hidden: training people and redesigning work consume resources before producing a return [6]. AI may help with that work, or add new supervision costs.

An experiment may still need scarce evidence or slow physical processes, while repeated attempts can reproduce the same mistake. Cheap production can also create congestion. More proposals, reports, and applications compete for limited attention. A gain for the producer can become additional work for recipients, particularly when verification remains expensive. Output growth and system-wide productivity can consequently diverge.

3 From Demonstration to Useful Work

Noy and Zhang provide experimental evidence of gains in bounded professional writing tasks [13]. A workplace study by Brynjolfsson, Li, and Raymond follows the staggered introduction of an assistant

across 5,172 customer-support agents and reports a 15 percent average increase in issues resolved per hour, with substantial differences across workers [7]. These findings establish more than fluent demonstrations: in the studied settings, assistance changed completed work.

Software development shows why dates matter. METR’s early-2025 randomized study found a 19 percent slowdown among 16 experienced open-source developers [2]. Its February 2026 follow-up reports suggestive speedups but also serious selection and time-measurement problems; the authors say those data cannot reliably determine the current gain [3]. The older result remains evidence about its original setting. Treating it as a permanent estimate would be as misleading as treating the noisier follow-up as a clean reversal.

Suppose drafting takes half a project’s time and everything else takes the other half. Doubling drafting speed does not halve the project time: the unchanged half remains. A calculation analogous to Amdahl’s law expresses this limit [1]. Set the original total time to one. If fraction f becomes s times faster and checking the new output adds time r , total speedup is

$$S = \frac{1}{(1 - f) + f/s + r}.$$

The denominator adds unchanged work, accelerated work, and new checking. In the example, $f = 1/2$ and $s = 2$. If checking adds one tenth of the original project time, $r = 0.1$, so the project is about 1.18 times faster. This comparison holds the work and its quality fixed; redesigning the project can change the calculation.

4 When Assistance Reaches the Bottleneck

Even within model training, more resources have to be spent on the right things. Hoffmann and colleagues study how to divide a fixed computation budget between model size and training data [10]. That allocation problem concerns the model itself. The calculation above asks a different question: how much does improving the model help a project whose remaining work still takes just as long?

The more ambitious prospect is that AI also helps with the parts left unchanged in this calculation. If assistance improves coding, experiment interpretation, and tool construction together, gains may propagate further than faster coding alone. A persistent bottleneck can itself become a target of assistance.

The arithmetic does not bound progress after a workflow changes, but the possibility of redesign supplies no measured acceleration rate. Faster implementation can still leave poor experiment choice untouched. The interesting possibility is that the next bottleneck becomes tractable too. An easier analysis, for example, might reveal which experiment is worth running, saving laboratory work as well as time at the computer.

A research team can improve its tools without handing its entire learning agenda to an agent. Liu and van der Schaar discuss a more demanding form of sustained self-improvement, in which agents assess and adapt their own learning processes, while allowing responsibilities to be shared [11]. The historical possibility considered here already arises when people and AI improve their joint research process. Full autonomy could extend it; full autonomy is not required for it to begin.

5 Projects That Become Possible

Even finite improvements across several tasks could have a large historical impact. Coordinating the work still matters [12]: ten affordable experiments can reveal something useful or merely enlarge the review queue.

METR’s May 2026 analysis distinguishes gains on an old task mix, gains on the new mix chosen with AI, and gains in value [8]. Cheap production can make elaborate work attractive that nobody previously considered worth doing. Measuring how long that new work would have taken manually can exaggerate its added value; measuring only old work can miss valuable new possibilities. Fixed-work speedup captures neither effect completely.

This also gives the optimistic case more substance. A useful project can become possible before any familiar workflow gets much faster. A small team can acquire a range of implementation and

analytical abilities previously requiring more specialists. Such gains could widen participation in invention even if physical experiments, judgment, and coordination continue to impose limits.

One reason this time may be different is that AI could help build the conditions for its own usefulness. Faster drafting alone leaves much of the work untouched. Help with implementation, interpretation, and tool building could change what a team can attempt in the first place. How far this assistance spreads remains the historical question.

References

- [1] Gene M. Amdahl. Validity of the single processor approach to achieving large scale computing capabilities. In *Proceedings of the Spring Joint Computer Conference*, pages 483–485, 1967. doi: 10.1145/1465482.1465560.
- [2] Joel Becker, Nate Rush, Elizabeth Barnes, and David Rein. Measuring the impact of early-2025 AI on experienced open-source developer productivity, 2025. URL <https://arxiv.org/abs/2507.09089>. arXiv:2507.09089.
- [3] Joel Becker, Nate Rush, Tom Cunningham, David Rein, and Khalid Mahamud. We are changing our developer productivity experiment design. METR, February 24, 2026. URL <https://metr.org/blog/2026-02-24-uplift-update/>.
- [4] Timothy F. Bresnahan and Manuel Trajtenberg. General purpose technologies ‘Engines of growth’? *Journal of Econometrics*, 65(1):83–108, 1995. doi: 10.1016/0304-4076(94)01598-T.
- [5] Erik Brynjolfsson and Lorin M. Hitt. Beyond computation: Information technology, organizational transformation and business performance. *Journal of Economic Perspectives*, 14(4): 23–48, 2000. doi: 10.1257/jep.14.4.23.
- [6] Erik Brynjolfsson, Daniel Rock, and Chad Syverson. The productivity J-curve: How intangibles complement general purpose technologies. *American Economic Journal: Macroeconomics*, 13(1):333–372, 2021. doi: 10.1257/mac.20180386. URL <https://www.aeaweb.org/articles?id=10.1257/mac.20180386>.
- [7] Erik Brynjolfsson, Danielle Li, and Lindsey Raymond. Generative AI at work. *The Quarterly Journal of Economics*, 140(2):889–942, 2025. doi: 10.1093/qje/qjae044. URL <https://academic.oup.com/qje/article/140/2/889/7990658>.
- [8] Tom Cunningham and Parker Whitfill. Task substitution and uplift. METR, May 8, 2026. URL <https://metr.org/blog/2026-05-08-task-substitution-and-uplift/>.
- [9] Paul A. David. The dynamo and the computer: An historical perspective on the modern productivity paradox. *American Economic Review*, 80(2):355–361, 1990. URL <https://www.jstor.org/stable/2006600>.
- [10] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, et al. Training compute-optimal large language models, 2022. URL <https://arxiv.org/abs/2203.15556>. arXiv:2203.15556.
- [11] Tennison Liu and Mihaela Van Der Schaar. Position: Truly self-improving agents require intrinsic metacognitive learning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 81714–81727. PMLR, 2025. URL <https://proceedings.mlr.press/v267/liu25cw.html>.
- [12] Elliot Meyerson and Xin Qiu. Position: Scaling LLM agents requires asymptotic analysis with LLM primitives. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 81797–81809. PMLR, 2025. URL <https://proceedings.mlr.press/v267/meyerson25a.html>.
- [13] Shakked Noy and Whitney Zhang. Experimental evidence on the productivity effects of generative artificial intelligence. *Science*, 381(6654):187–192, 2023. doi: 10.1126/science.adh2586. URL <https://doi.org/10.1126/science.adh2586>.