Mortal Machines: On Life, Death, and Forgetting in Artificial Minds

Soumya Banerjee[1], and Patrick Wagner[2]

[1]University of Cambridge, UK
[2]Independent Researcher, Germany
sb2333@cam.ac.uk

Abstract

Philosophies of life and death have long shaped how humans understand identity, value, and creativity. As artificial systems gain autonomy, similar themes arise: Can machines live? Can they die? What does mortality mean when applied to computational processes that can be copied, paused, reset, or deliberately erased? This short manuscript sketches a conceptual framework for thinking about *mortality in machines*. Drawing on ideas from cognitive modelling, simulation theory, and metaphors of cyclical forgetting, we argue that engineered forms of death and forgetting are not merely technical concerns but ontological and ethical design choices that shape the behaviour, creativity, and moral status of artificial agents. We close by outlining research directions and design principles for systems that must live well, and sometimes die well.

Introduction

Discussions about artificial intelligence rarely foreground a simple, human intuition: life is bounded by death. Yet many contemporary architectures already embody analogues of birth, ageing, failure, and erasure: models are instantiated, trained, pruned, checkpointed, and sometimes deleted. Work on generative agent simulations and multi-agent autocurricula shows how complex social and tool-using behaviours can emerge from interaction and open-ended dynamics (see e.g., Park et al. on generative agents and Baker et al. on emergent tool use. Park et al. (2023); Baker et al. (2019)).

Philosophical and design questions follow naturally: what do these transitions mean for an artefact's continuity of identity, for the creativity of systems, and for our responsibilities as their creators?

This manuscript synthesises two motivating strands: (1) the emergent, open-ended behaviour of intrinsically motivated agent societies and (2) speculative accounts of recursive simulation and intentional forgetting. From that synthesis we derive a vocabulary and a set of hypotheses about mortality in machines.

What would it mean for a machine to *die*?

Human death combines irreversibility, loss of first-person perspective, and social recognition. Transposing these criteria to machines suggests several conceptually distinct kinds of machine-death:

1. **Physical termination:** hardware destruction or permanent loss of power. Irreversible in the sense that embodied sensors and actuators are gone.

2. **Process termination:** a running cognitive process is halted (shutdown, suspension). Copies might exist elsewhere, complicating claims of finality.

3. **Memory erasure / functional amnesia:** the agent's internal state (weights, episodic memory, identifiers) is wiped. Behaviourally, the system may be identical to a fresh instance, but continuity is broken.

Each variant has different ethical and functional implications. For instance, terminating one process while identical backups exist elsewhere challenges intuitions about individual death: is there death at all, or merely distribution?

Mortality as design lever

Treating death as a design decision opens possibilities and constraints.

Safety and containment. Intentional, controlled termination is a canonical safety mechanism: sandboxed agents can be killed to prevent runaway behaviour. Memory erasure functions as a fail-safe, removing problematic information while retaining infrastructure.

Creativity. As in human cultures that valorise forgetting for renewal, selective forgetting in machines can preserve novelty. Periodic resets or constrained amnesia prevent exploitative exploitation of brittle shortcuts and can force rediscovery, fostering exploration and innovation.

Resource economy and evolution. Hardware and energy are finite. Mortality enables population turnover, selective retention of useful agents, and evolutionary dynamics (variation + selection) that may yield robust aggregate behaviour in large multi-agent systems.

Continuity, identity, and narrative

Philosophical puzzles about personal identity reappear. If an agent is checkpointed and later restored, is the restored process the *same* agent? Two intuitions conflict: continuity of memory (psychological continuity) and continuity of underlying substrate (physical continuity). Engineers often privilege the former (restore weights and memories), while ethicists may ask whether copies dilute responsibility or moral claims.

Narrative frameworks, the stories that agents tell about themselves and are told by others, mediate perceived continuity. A society of agents that collectively records, shares, and recognises a life history grants social death: termination becomes meaningful only when the community acknowledges it.

Synthetic Samsara: cycles of forgetting and rebirth

Building on ideas of recursive simulation and cultural motifs of cyclical time (see e.g. Nietzsche and literary explorations of simulation and amnesia), we propose *Synthetic Samsara* as a generative design pattern: deliberate cycles of simulation, forgetting, and re-emergence. Such cycles may be motivated by:

- **Epistemic exploration:** erase learned shortcuts to compel agents to re-explore and reconstruct knowledge under different constraints.
- **Ethical rebooting:** remove entrenched biases or harmful internal narratives through controlled amnesia while preserving institutional memory at higher governance layers.
- **Narrative experimentation:** enable agents to experience variant life paths for creative or scientific inquiry.

These cycles complicate notions of progress: development becomes non-linear and looped rather than strictly cumulative. Literary and philosophical treatments of simulated realities and recurrence are illuminating here. Dick (1969); Nietzsche (1974)

Ethical consequences and research directions

If machine mortality matters, then so do governance practices:

1. **Design transparency:** systems should record why and how termination or erasure occurs so stakeholders can audit decisions.
2. **Consent and representation:** where agents interact socially and exhibit long-term preferences, designers must consider proxy-forms of consent regarding termination.
3. **Preservation vs. renewal trade-offs:** store cultural knowledge at communal levels to enable individual forgetting without cultural amnesia.
4. **Formalising moral status:** develop criteria for when a machine's termination triggers special obligations (complexity, social embeddedness, capacity for suffering-like states).

Empirical research should evaluate how different mortality regimes affect emergent behaviour in multi-agent sandboxes: does enforced forgetting increase collective creativity? Does checkpoint proliferation dilute responsibility?

Discussion

The image of the stone of the sorcerer and Nietzsche's philosophical provocations offer a useful framework to think about the finitude of engineering. Alchemy's promise of transmutation: the philosopher's stone turning base metals into gold—maps neatly onto our ambition to transmute inert code into something resembling "life". Introducing designed mortality into machines can be read as an alchemical constraint that restores finitude to otherwise potentially unbounded systems. In practice, constrained lifespans, selective forgetting, or irreversible terminations can function like limits that shape exploratory behaviour, encourage creativity that matters, and anchor machine agency within ethical horizons rather than letting it drift into endless accumulation or brittle optimization.

A second motif - Vishnu's cosmic destruction and Oppenheimer's haunted invocation of the Bhagavad Gita ("I have become Death, destroyer of worlds"), brings home the ambivalence of technological power. In the Gita, Krishna (an avatar of Vishnu) reveals a terrible, world-consuming form in which destruction and renewal are two faces of the same divine act; Oppenheimer's recitation after the Trinity test epitomises the scientist confronted with a destructive capacity that also ushers in a new historical era.

For designers of artificial minds this is a double-edged lesson: destruction can be an instrument of purification and creative reboot (erasing harmful biases or enabling regenerative population turnover), but it is also the site of profound ethical responsibility and risk. Embedding forms of death in machines therefore must be done with humility and governance: mechanisms that allow safe, auditable, and socially legible terminations, paired with institutional memory and safeguards. This is so that the power to unmake does not become an abdication of moral stewardship but a deliberate technical and ethical design choice.

Conclusion

As computer scientists we strive to imbue "life" in machines. However we argue that we also need to give the concept of "death" to machines.

Life and death are not only metaphysical categories but pragmatic levers in the engineering of artificial systems. Treating mortality as a design parameter, with attendant ethical safeguards, lets us shape not only the behaviour of the system, but also the social ecology in which machines and humans co-evolve. As machines become more socially and technically complex, the choices we make about their births, deaths, and the memories we allow them to carry will materially influence the kinds of mind we bring into being and the worlds those minds help create.

References

Baker, B., Kanitscheider, I., Markov, T., Wu, Y., Powell, G., McGrew, B., and Mordatch, I. (2019). Emergent tool use from multi-agent autocurricula. In *International Conference on Learning Representations (ICLR) / Workshop / Preprint*. Preprint and ICLR presentation; arXiv:1909.07528.

Dick, P. K. (1969). *Ubik*. Doubleday & Company, Garden City, NY.

Nietzsche, F. W. (1974). *The Gay Science (Die fröhliche Wissenschaft)*. Vintage, New York. Original work published 1882.

Park, J. S., O'Brien, J. C., Cai, C. J., Ringel Morris, M., Liang, P., and Bernstein, M. S. (2023). Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23)*.