X^{∞} – Postmoral and Emotionless Mathematical Foundations of Ethical Governance as a Self-Reinforcing System (Working Paper, Version 1.3)

The Auctor
x_to_the_power_of_infinity@protonmail.com
https://mastodon.social/@The_Auctor
GitHub: Xtothepowerofinfinity

April 23, 2025

Abstract

The X^{∞} system redefines responsibility: not as a moral category, but as a mathematically regulated impact. It resolves classical ethical dilemmas through a self-reinforcing model that governs power via feedback and protection mechanisms. This text formalizes the system's foundations, focusing on Cap logic (authorities), feedback penalties, and antispeciesism. The goal is a robust, transparent system that distributes responsibility without moral assumptions.

Contents

1 Introduction

The X^{∞} system stems from a radical premise: Moral categories like “good” or “evil” are unsuitable for regulating responsibility in complex systems. Instead, X^{∞} defines responsibility as *impact*—measurable, mathematically representable, and systemically regulatable. This leads to a *postmoral* approach that replaces ethical governance with structural mechanisms.

1.1 System Objectives

- **Fairness through Impact**: Responsibility is distributed based on actual consequences, not intentions.
- **Protection of the Vulnerable**: Weaker actors gain a stronger voice through feedback weighting.
- **Antispeciesism**: Humans, non-humans, and the environment are treated as equal entities.
- **Self-Reinforcement**: The system learns from feedback and adapts without external intervention.

1.2 Document Structure

This document formalizes the mathematical foundations of X^{∞}. Section 2 defines the concept of responsibility, Section 3 outlines the system architecture, Section 4 introduces the notation, Section 5 develops the mathematical framework, Section 6 addresses antispeciesism, and Section 7 concludes with a discussion.

2 Responsibility as Impact

In X^{∞}, responsibility is not a moral obligation but a systemic measure, defined by an entity’s impact on the system. An entity (individual, organization, environment, AI) bears responsibility for tasks proportional to its capability and receives temporary authorities.

2.1 Postmoral Perspective

Moral judgments often rely on subjective values, leading to conflicts in heterogeneous systems. X^{∞} replaces these with objective criteria: *Cap* (authorities) and *Feedback*. This makes responsibility measurable and independent of cultural or speciesist assumptions.

2.2 Systemic Consequences

Those who assume responsibility directly influence the system. Those who misuse it are corrected through penalties (Cap reduction). This creates a balance where authority is legitimized by responsibility.

3 System Architecture

X^∞ is based on three pillars:

- **Cap Logic**: Authorities (Cap) measure an entity's capabilities based on its responsibility.
- **Feedback**: Weaker entities have greater weight to prevent power concentration.
- **Protection Mechanisms**: $\mathrm{Cap}_{\mathrm{protection}}$ shields vulnerable entities from overload.

3.1 Flow of Responsibility

Responsibility flows through delegation and return. Entities can delegate tasks but remain responsible for their completion as if they performed them themselves. Returns are permitted but incur penalties to prevent abuse.

3.2 Adaptability

The system adapts through feedback. Positive feedback increases Cap, negative feedback reduces it. This enables organic development without centralized control.

4 Key Concepts

5 Systemic Origin Equation and the Role of the UdU)

The following equation forms the symbolic foundation for the responsibility structure of the UdU (Lowest of the Low) within the X^∞ model:

$$\mathrm{Cap}_{\mathrm{solo,UdU}} = \left(X \cdot S \cdot \left(\frac{S-1}{S} \right)^{1/D} \right)^{\infty} \tag{1}$$

Where:

- X is the assumed task or intended impact within the system context.
- S is the number of all other relevant system entities excluding the UdU itself. The UdU's intrinsic self-responsibility remains constant and unaffected.
- D describes the position within the responsibility network (e.g., direct leadership level $D = 1$, deeper delegation $D > 1$).
- The exponent ∞ symbolizes the full structural authority necessary to carry ultimate responsibility — but exclusively bound by feedback and responsibility, never by arbitrariness.

This formulation highlights that the UdU operates within the system architecture not by personal power or mandate, but solely based on the accepted task and systemic coupling. Although the UdU is structurally positioned outside the formal system to guarantee

Table 1: Parameter Notation

Symbol	Meaning
E	Entity (individual, organization, environment, AI)
A	Task
$\text{Cap}_{\text{solo}}(E)$	Temporary authorities for self-performed tasks
$\text{Cap}_{\text{team}}(E)$	Temporary authorities through delegated tasks
$\text{Cap}_{\text{aktiv}}(E)$	Authorities for active tasks ($\text{Cap}_{\text{solo}} + \text{Cap}_{\text{team}}$)
$\text{Cap}_{\text{past}}(E)$	Historical authorities
$\text{Cap}_{\text{protection}}(E, t)$	Protection parameter
$\text{Cap}_{\text{base}}(E)$	Inalienable minimum authorities
$\text{Cap}_{\text{BGE}}(E)$	Basic income
$\text{Cap}_{\text{potential}}(E)$	Future capability
$\text{Cap}_{\text{potential,aktiv}}(E)$	Available capacity for additional tasks
$M_{\text{pot}}(E, A)$	Suitability of entity E for task A, a voluntary self-assessment to account for self-realization and intrinsic motivation.
D	Delegation history ($D_{\text{Entity}} - 1$)
D_{hist}	Historical delegation depth (median of delegation chains)
D_{tiefe}	Current delegation depth
D_{aktuell}	Current delegation depth of a task
B	Delegation breadth
k	Growth parameter or complexity factor
k_{aktuell}	Current complexity factor of a task
k_{Median}	Median of complexity factors in the system
$\alpha, \beta, \lambda, \gamma, \mu, \nu, \phi, \psi, \rho, \theta, \omega, \chi, \delta$	Model parameters (penalties, feedback, compensation)
S_E, S_S	Oversteering by E, total system oversteering
$R_E, R_{E,\text{delegate}}, R_S$	Returns (own, by delegates, total)
w_E	Feedback weight
F_E, M_E	Feedback activity, abuse component

autonomy, it remains fully bound to the feedback principle and is obligated to report its effects *ex post* to the system — but only regarding the *What for* (the goal), never the *How* (the specific method or means).

6 Mathematical Framework

6.1 Cap Logic

Cap (authorities) is the central measure of responsibility in X^∞. There are three main types:

- **Cap_{past}**: Historical authorities, based on completed tasks.
- **$\text{Cap}_{\text{potential}}$**: Potential authority assumption, based on suitability and historical performance—augmented by growth opportunities.
- **$\text{Cap}_{\text{protection}}$**: Protection parameter, preventing overload.

6.2 Historical Authorities ($\mathbf{Cap_{past}}$)

$$\begin{aligned}\mathrm{Cap}_{\mathrm{past}}(E) = \sum \mathrm{Cap}_{\mathrm{solo,final}}(E) + \sum \mathrm{Cap}_{\mathrm{team,final}}(E) \\ + \sum \Delta\mathrm{Cap}_{\mathrm{past,return}}(E) \cdot \frac{1}{D_{\mathrm{hist}}} + \sum \Delta\mathrm{Cap}_{\mathrm{past,comp}}(E) \\ + \sum \Delta\mathrm{Cap}_{\mathrm{past,comp,oversteer}}(E) + \sum \Delta\mathrm{Cap}_{\mathrm{past,bonus}}(E) \\ + \sum \Delta\mathrm{Cap}_{\mathrm{past,feedback}}(E) - \sum \mathrm{Cap}_{\mathrm{past,penalty}}(E)\end{aligned} \quad (2)$$

In prose: $\mathrm{Cap}_{\mathrm{past}}$ represents the accumulated historical authority of an entity—its "performance history" in the system. It grows solely from completed tasks, never from active (current) tasks. This rule prevents inflating the historical responsibility balance through short-term projects or strategic delegation.

Importantly: $\mathrm{Cap}_{\mathrm{past}}$ is the foundation for $\mathrm{Cap}_{\mathrm{potential}}$ and influences both incentives and feedback weight distribution. An entity that has assumed and successfully borne significant responsibility can structurally take on greater authorities for future tasks—unless it has exhibited systemically problematic behavior (see penalties and returns).

Individual Components of $\mathbf{Cap_{past}}$:

- $\mathrm{Cap}_{\mathrm{solo,final}}(E)$: Sum of self-completed tasks. *(Directly responsible and operationally executed—highest evaluation level in the system.)*

- $\mathrm{Cap}_{\mathrm{team,final}}(E)$: Sum of tasks completed through delegation. *(Responsibility assumed, but execution (partially) delegated.)*

- $\Delta\mathrm{Cap}_{\mathrm{past,return}}(E)$: Compensation for tasks voluntarily returned. The factor $\frac{1}{D_{\mathrm{hist}}}$ mitigates compensation when returns occur deep in the delegation tree, as responsibility primarily lies with higher-level delegators.

- $\Delta\mathrm{Cap}_{\mathrm{past,comp}}(E)$: Compensation for tasks withdrawn from an entity (e.g., due to oversteering or reorganization).

- $\Delta\mathrm{Cap}_{\mathrm{past,comp,oversteer}}(E)$: Specific compensation for the oversteered. It covers experience accumulation before the task was involuntarily relinquished.

- $\Delta\mathrm{Cap}_{\mathrm{past,bonus}}(E)$: Incentive bonus for entities actively supporting weaker ones. This component rewards targeted support and contributes to system balance.

- $\Delta\mathrm{Cap}_{\mathrm{past,feedback}}(E) = \phi \cdot w_E \cdot F_E - \psi \cdot M_E$: Feedback adjustment. Positive feedback increases $\mathrm{Cap}_{\mathrm{past}}$, abuse reduces it. This acts as an **emergency brake** against manipulation and irresponsible behavior.

- $\mathrm{Cap}_{\mathrm{past,penalty}}(E)$: Penalties, consisting of:

$$\begin{aligned}\mathrm{Cap}_{\mathrm{past,penalty}}(E) = P_{\mathrm{oversteer}}(E) + \Delta\mathrm{Cap}_{\mathrm{past,penalty,return}}(E) \\ + \Delta\mathrm{Cap}_{\mathrm{past,penalty,delegate}}(E) + \Delta\mathrm{Cap}_{\mathrm{past,penalty,k}}(E) \\ + \Delta\mathrm{Cap}_{\mathrm{past,penalty,D}}(E)\end{aligned} \quad (3)$$

In prose: These penalties prevent entities from gaining an advantage. They sanction irresponsible behavior such as frequent returns, failed delegations, overly complex or deeply delegated tasks, and unauthorized oversteering. Each penalty is exponentially scaled to more heavily punish repeated misconduct and maintain systemic balance.

6.3 Future Capability ($\text{Cap}_{\text{potential}}$)

$$\begin{aligned}\text{Cap}_{\text{potential}}(E) = \sum_{j \in \text{Tasks}} & M_{\text{pot}}(E, j) \cdot \text{Value}(A_j) \\ & \cdot f(D_{\text{hist}}, D_{\text{total}} - D_E, B) \\ & \cdot \frac{\text{Cap}_{\text{past}}(E) + \text{Cap}_{\text{BGE}}(E) + \text{Cap}_{\text{base}}(E) - \text{Cap}_{\text{protection}}(E)}{\text{Cap}_{\text{past}}(E) + \text{Cap}_{\text{base}}(E)}\end{aligned} \tag{4}$$

In prose: $\text{Cap}_{\text{potential}}$ measures the level of future responsibility an entity can realistically bear. The model considers both past achievements (Cap_{past}), inalienable basic income (Cap_{BGE}), and the protection level ($\text{Cap}_{\text{protection}}$), which shields weaker entities from overload.

The protection value $\text{Cap}_{\text{protection}}$ reduces $\text{Cap}_{\text{potential}}$ to ensure vulnerable entities are not forced into roles they are structurally unprepared for. This eliminates abuse: High protection needs directly limit responsibility allocation.

Components of the Formula:

- $M_{\text{pot}}(E, j)$: The potential matrix. It describes the suitability of entity E for task A_j, derived from qualifications, enjoyment, and knowledge in the respective domain Δ.

- $\text{Value}(A_j)$: The systemically determined value of a task. Tasks of higher complexity or relevance contribute more to potential calculation.

- $f(D_{\text{hist}}, D_{\text{total}} - D_E, B)$: The growth factor function.

- D_{hist}: Historical delegation depth, indicating experience with project complexity.
- $D_{\text{total}} - D_E$: Current delegation depth of a task relative to the entity—closer to the source increases potential.
- B: Delegation breadth—the number of parallel recipients chosen. Prevents strategic dilution of responsibility.

- The denominator term:

$$\frac{\text{Cap}_{\text{past}}(E) + \text{Cap}_{\text{BGE}}(E) + \text{Cap}_{\text{base}}(E) - \text{Cap}_{\text{protection}}(E)}{\text{Cap}_{\text{past}}(E) + \text{Cap}_{\text{base}}(E)}$$

scales the potential based on historical contribution. Cap_{base} ensures the denominator is never zero, as inalienable minimum authorities are always present.

Key Properties:

- **Reward for Genuine Performance**: Historical responsibility (Cap_{past}) enhances authorities for future tasks.
- **Protection of the Vulnerable**: $\text{Cap}_{\text{protection}}$ prevents systemic overload of vulnerable entities.
- **Abuse Prevention**: The combination of protection mechanisms and dynamic limitation of $\text{Cap}_{\text{potential}}$ ensures responsibility is distributed justly.
- **Automatic Balance**: Those who bear more responsibility and complete tasks successfully gain legitimate allocation rights for future tasks—without manual intervention.

Prose Conclusion: $\text{Cap}_{\text{potential}}$ is not just a numerical value but an expression of systemic fairness: It describes an entity's growth opportunities based on genuine performance, considering its protection needs and systemic balance. The model actively prevents exploitation of the weaker and enforces responsibility where it can be borne.

6.4 Feedback Penalties, Returns, and Oversteering

1. Feedback as a Systemic Emergency Brake

$$\Delta\text{Cap}_{\text{past,feedback}}(E) = \phi \cdot w_E \cdot F_E - \psi \cdot M_E \tag{5}$$

In prose: Feedback serves as an immediate early warning and correction system. An entity whose behavior becomes systemically problematic (e.g., through delegation abuse or persistent irresponsibility) is sanctioned directly in Cap_{past}. This function replaces ethical appeals with mathematically regulated impact correction.

- ϕ: Amplification factor for constructive feedback. - ψ: Penalty factor for destructive or abusive actions. - F_E: Evaluated system impact, e.g., through correctly executed feedback. - M_E: Abuse indicator—e.g., excessive returns without justification, strategic oversteering.

Purpose: This immediate effect prevents power accumulation through opacity and ensures feedback operates in real-time. Feedback replaces control with impact—protecting the system through direct self-regulation.

2. Feedback Weighting: Weaker Voices Matter More

$$w_E = \frac{1}{\text{Cap}_{\text{potential}}(E)} \tag{6}$$

In prose: Weaker voices in the system—entities with low $\text{Cap}_{\text{potential}}$—receive greater weight. Their feedback alters the system more directly than that of the "strong." The mathematical expression is the reciprocal feedback weight w_E, tying weighting directly to future potential. Since $\text{Cap}_{\text{potential}}$ never reaches zero due to system-guaranteed minimum authorities, the weighting is mathematically stable.

This follows the ethical principle: **The weaker the voice, the more closely it must be heard.**

This prevents systemic bubbles where only established responsibility bearers dominate. Minorities or new system participants retain immediate relevance—not because they are formally equal, but because their perspective has greater impact in the model.

3. Returns—Voluntary but Not Cost-Free

$$\Delta\text{Cap}_{\text{past,penalty,return}}(E) = \mu \cdot \exp\left(\rho \cdot \frac{R_E}{R_S}\right) \tag{7}$$

In prose: Those who voluntarily return tasks immediately lose $\text{Cap}_{\text{aktiv}}(E)$ and are penalized after task completion. Returns are permitted but not neutral: They cost reputation, trust, and systemic impact. The exponential term $\exp(\rho \cdot \frac{R_E}{R_S})$ ensures frequent returns are more heavily penalized, with ρ controlling penalty severity:

- Returns in high-responsibility contexts carry greater weight. - If many others also return, the individual penalty decreases.

Purpose: This creates a system that allows returns without making them attractive. It protects the overwhelmed while promoting responsibility assumption.

4. Penalty for Delegation Failure (Indirect Returns)

$$\Delta\text{Cap}_{\text{past,penalty,delegate}}(E) = \nu \cdot \exp\left(\theta \cdot \frac{R_{E,\text{delegate}}}{R_S}\right) \tag{8}$$

In prose: If an entity delegates responsibility and the recipient returns it, this is attributed to the original delegator—not as direct fault, but as a failed responsibility chain. The exponential term $\exp(\theta \cdot \frac{R_{E,\text{delegate}}}{R_S})$ more heavily penalizes frequent returns by delegates, with θ controlling penalty severity.

- $R_{E,\text{delegate}}$: Number of returns by assignees. - ν: Base penalty scale—typically lower than for direct returns. - θ: Escalation factor for repeated delegation failures.

Purpose: Responsibility does not end with delegation. Delegators must consider their choices and bear co-responsibility for systemic chain failures.

5. Oversteering—Intervention in Existing Responsibility

$$P_{\text{oversteer}}(E) = \lambda \cdot \exp\left(\gamma \cdot \frac{S_E}{S_S}\right) \tag{9}$$

In prose: Oversteering is the system's emergency mechanism: An entity intervenes in another's task area, recognizing that goal achievement is otherwise at risk. This is permitted—but costly.

- The cost of frequent oversteering rises exponentially. - Each oversteering reduces Cap_{past} long-term.

The cost is shared with the original delegator, discouraging authoritarian abuse and strategic "offloading" while promoting cooperation. The oversteerer assumes the oversteered's responsibility and authorities as additional $\text{Cap}_{\text{aktiv}}(E)$ but is penalized post-task. Oversteering is only possible if the oversteerer's $\text{Cap}_{\text{potential}}(E)$ can accommodate the additional task.

6. Penalty for Excessive Complexity (k-Values)

$$\Delta\text{Cap}_{\text{past,penalty,k}}(E) = \omega \cdot \exp\left(\chi \cdot \frac{k_{\text{aktuell}}}{k_{\text{Median}}}\right) \tag{10}$$

In prose: Entities undertaking overly complex tasks (high k_{aktuell}) risk a penalty. This prevents responsibility dilution through unnecessary complexity. The exponential term $\exp(\chi \cdot \frac{k_{\text{aktuell}}}{k_{\text{Median}}})$ penalizes tasks significantly exceeding the systemic median (k_{Median}), with χ controlling penalty severity.

- ω: Base penalty scale. - k_{aktuell}: Complexity factor of the current task. - k_{Median}: Median of complexity factors in the system.

Purpose: This penalty promotes efficient and responsible task assumption by sanctioning strategic complexity escalation.

7. Penalty for Excessive Delegation Depth (D-Values)

$$\Delta\text{Cap}_{\text{past,penalty,D}}(E) = \delta \cdot \exp\left(\rho \cdot \frac{D_{\text{aktuell}}}{D_{\text{hist}}}\right) \tag{11}$$

In prose: Entities delegating tasks too deeply (high D_{aktuell}) risk a penalty. This prevents responsibility dilution through excessive delegation chains. The exponential term $\exp(\rho \cdot \frac{D_{\text{aktuell}}}{D_{\text{hist}}})$ penalizes tasks significantly exceeding the historical norm (D_{hist}), with ρ controlling penalty severity, consistent with the return penalty.

- δ: Base penalty scale. - D_{aktuell}: Current delegation depth of the task. - D_{hist}: Historical delegation depth (median of delegation chains).

Purpose: This penalty promotes clear and responsible delegation structures by sanctioning overly deep chains.

Summary Overview:

The penalties address various abuse channels to maintain systemic integrity. Table 2 summarizes the penalties and their purposes:

Table 2: Penalties and Abuse Channels

Penalty	**Abuse Channel**	**Type**
$\Delta\text{Cap}_{\text{past,penalty,return}}$	Return of own tasks	Exponential
$\Delta\text{Cap}_{\text{past,penalty,delegate}}$	Return by own recipients	Exponential
$\Delta\text{Cap}_{\text{past,penalty,k}}$	Excessive delegation breadth (k-values)	Exponential
$\Delta\text{Cap}_{\text{past,penalty,D}}$	Excessive delegation depth	Exponential
$P_{\text{oversteer}}$	Unauthorized oversteering	Exponential

- Returns are permitted but diminish reputation. - Oversteering saves projects but reduces legitimacy. - Complexity and deep delegations are penalized to keep responsibility clear. - Feedback protects the system—especially from prolonged manipulation. - The ethical impact of feedback relies on asymmetric weighting:

The weak weigh heavier.

This architecture creates a learnable, self-healing system where power is corrected by impact, and responsibility is not a moral choice but a structural consequence.

6.5 Protection Mechanisms

$\text{Cap}_{\text{protection}}$ is the core of protection for weaker entities. It is dynamically calculated:

$$\text{Cap}_{\text{protection}}(E,t) = \alpha \cdot \frac{1}{\text{Cap}_{\text{past}}(E) + \text{Cap}_{\text{base}}(E)} + \beta \cdot \text{Vulnerability}(E,t) \tag{12}$$

- α: Scaling factor for base protection. - β: Weighting of vulnerability. - $\text{Vulnerability}(E,t)$: Time-dependent measure of protection need (e.g., resources, experience).

In prose: $\text{Cap}_{\text{protection}}$ ensures no entity is overwhelmed. It acts as a safety net: The weaker the entity, the higher the protection. This prevents exploitation and promotes systemic stability.

7 Antispeciesism

X^{∞} treats all entities—humans, non-humans, environment—as equal. Antispeciesism is built into the Cap logic:

- **Equal Baseline**: Every entity receives Cap_{base} and Cap_{BGE}.
- **Fairness through Impact**: $\text{Cap}_{\text{potential}}$ is based on responsibility, not species.

- **Feedback for All**: Every entity can provide feedback, regardless of its nature.

In prose: Antispeciesism does not mean uniformity but equal value in impact. A tree, an AI, or a human—each entity bears responsibility according to its capability and is protected by the system.

8 Conclusion

X^∞ offers a radical alternative to moral systems. It replaces subjective values with mathematical precision, protects the vulnerable through structural mechanisms, and enables fair responsibility distribution without speciesist biases. The system's strength lies in its simplicity and adaptability: It requires no external authority, regulating itself through impact and feedback.

Note: This document is a Working Paper and represents the first complete version of the mathematical foundations of the X^∞ model. It is intended for public discussion, feedback, and validation. Further detailed chapters will follow. Changes and additions are explicitly reserved.

8.1 License