Determining Value in the Space of Ideas

Axel Cortes, Carlos A. Vera, Joshua Bate [DeSciWorld, CEL]

November 2025

1 Introduction

Beneath the layer of language lies a semantic surface, a multidimensional substrate that contains pure meaning independent of linguistic form. When we express ideas through language, we constrain their richness to a determined viewpoint, enabling communication but irreversibly losing information.

To approximate the original semantic surface more completely, repeated mappings of various orthogonal knowledge structures are required. Each projection contributes a unique perspective, and together these overlapping mappings can reconstruct a higher-fidelity image of the underlying semantic terrain. Within this surface, distinct regions of meaning, or functional units, can be identified through clustering of retrieval and usage patterns. These units can, in principle, be owned, exchanged, and studied as evolving entities, similar to species in an ecological taxonomy.

If such a framework holds, it enables the emergence of a global marketplace of ideas where concepts possess measurable lifecycles and leave stigmergic traces that guide collective attention and innovation. These dynamics form the foundation for a new class of socio-epistemic infrastructure: a distributed knowledge commons where collective intelligence emerges from the aggregation of networked insights. Such a system naturally creates the informational substrate for futarchy, enabling governance decisions to be informed by the real time evolution and validation of collective knowledge. Through semantic compression and self-referential feedback, this system could render the world legible to itself, creating the conditions for planetary scale coordination.

2 The Space of Ideas

2.1 Semantic Embedding Space

The key technical engine that underlies KNOW economics is the algorithm to compute semantic distance between two pieces of knowledge. Having this measure of distance between ideas allows us to think about "The Space of Ideas" as a geometrical concept.

In geometry, a *metric space* is broadly defined as a set of elements, which is endowed with a "distance" function between any two elements of the set.

In this section we will provide a short primer on relevant geometrical notions, as they apply to the space of ideas. Eventually, once we have this geometric interpretation, we can develop an analogy of "gravitational forces" between ideas (once we also assign a concept of "mass" to each idea)

2.2 Target Points and Output Points

We distinguish two types of elements that exist in the space of ideas:

1. **Target Points:** Semantic targets defined dynamically on the protocol level (protocol governance), node level (Bonfire governance) or the individual level (Requests for Knowledge / bounties).

   Protocol level targets are defined initially by the DeSciWorld team, later by users or institutions and $KNOW$ token governance.

   Semantic targets may be defined as areas of semantic space, taxonomy or classifier labels or specific natural language requests.

2. **Output Points:** The results produced by Bonfires — they represent actual units of knowledge being posted to the network.

Each Bonfire produces a set of outputs $\{k_i\}$, and the reward computation may depend on their proximity to one or more targets $\{t_j\}$.

Semantic Proximity and Match Quality

Let us define what is the "distance" metric, d_{ij} in this space of ideas, between two points i and j. Our distance metric must have the following key properties:

1. The distance metric d between any two objects must be a positive real number.

2. The distance between any point and itself must be $d_{ii} = 0$

3. The distance is symmetric under exchange of the two points, $d_{ij} = dji$.

4. The distance satisfies a "triangle inequality", $d_{ij} \leq d_{ik} + d_{kj}$.

We can define such a distance in terms of another computable metric between two pieces of knowledge, called **semantic similarity**, which we denote as $s_{ij} \in [0, 1]$.

(Semantic similarity is a measure of how closely related two pieces of text or concepts are in meaning, regardless of whether they use the same words.)

Semantic similarity is one, only in the case that the two pieces of knowledge are the same, $s_{ii} = 0$, and $s_{ij} = 0$ in the case that the points i and j are completely unrelated.

One can build a distance function out of the semantic similarity, which satisfies the properties above, as

$$d_{ij} = \frac{1 - s_{ij}}{s_{ij}}$$

Now that we have a measurable distance function, we can incorporate it into our gravitational model.

3 Gravitational Pull in the Space of Ideas

We now extend the space of ideas introduced in Section 3 with the notion of *mass* to define a gravitational pull model over the space of ideas.

3.1 Mass of Points in the Space of ideas

Each point in the space $\mathcal{X}$, whether a target or a knowledge unit, is assigned a **mass** $M \in \mathbb{R}_{\geq 0}$ that quantifies its importance or informational weight.

3.1.1 Mass of Target Points

Each target point $t_j \in \mathcal{X}$ has an associated mass M_j, which may be determined by one or more of the following:

- Priority of the Target (e.g., urgency, social relevance)
- Popularity or recurrence across Bonfires, determined by retrieval rate
- Stake weight placed by bounty issuers or Target allocators
- Manual or algorithmic reputation scores

This allows some targets to exert more influence over knowledge production than others.

3.1.2 Mass of Output Points

Each knowledge unit k_i produced by a Bonfire is assigned a mass m_i, which may depend on:

- Retrieval frequency
- Information density or content volume (e.g., length, complexity, novelty)
- Token stake placed on the output by Bonfire participants
- Social endorsements
- Bonfire-level reputation scores

We denote this as $m_i = m(k_i)$, and in general $m_i > 0$ for all non-trivial outputs.

3.2 Gravitational Energy Between Points

We define a gravitational Energy V_{ij} between an output point k_i and a target point t_j as follows:

$$V_{ij} = G \cdot \frac{m_i M_j}{f(d_{ij})}$$

Where:

- $d(k_i, t_j)$ is the distance in semantic space between k_i and t_j
- m_i is the mass of the output
- M_j is the mass of the target
- G is a constant scaling factor (can be normalized to $G = 1$)

This captures the intuition that outputs which are *closer* and *more massive* relative to important targets exert stronger intellectual "gravitational pull."

In actual Newtonian gravity, the gravitational energy decays as $f(d_{ij}) = d_{ij}$, but this doesn't need to be the case in our model. We want to keep the general intuition that the gravitational energy should fall off as the two points move further apart, but we are free to choose a different function $f(d_{ij})$ controlling how quickly the energy falls off with distance.

3.3 Interpretation

The gravitational model provides a mechanism for evaluating the relevance and value of outputs in a continuous and differentiable manner.

Outputs that are:

- semantically close to high-priority targets,
- rich in information,
- and well-supported or staked,

will generate higher gravitational pull and should be more strongly rewarded by the protocol.

3.4 Reward Function Based on Gravitational Pull

The gravitational pull model naturally leads to reward formulas. For example, a Bonfire's reward can be computed as the total gravitational "energy" its is stored between its outputs and all active targets. This will be formalized in Section 5.

This gravitational formalism also enables the integration of:

- Decentralized staking models for target creation and weighting

- Information-theoretic or content-aware estimation of knowledge mass
- Temporal smoothing to track influence over time
- Subnet-based or validator-enhanced reputation as additional mass multipliers

3.5 Fixing the function $f(d_{ij})$

The function that controls how quickly gravitational energy decays with distance should be fixed by deciding the ideal distribution of reward and wealth. This in turn depends on the properties of the underlying semantic similarity metric, s_{ij}.

Suppose we have a sample set of points, $\{I\}$ in the space of ideas. One can define a probability distribution $p(s_{ij})$ as the probability that the semantic similarity between two points $i, j \in I$ is s_{ij}. This probability distribution can be obtained empirically by analysing a representative sample of points I.

Now that we know how semantic similarity is distributed, we can ask: how do we want gravitational energy to be distributed? We can use $f(d_{ij})$ to control what probability distribution $1/f(d_{i,j})$ will follow, given the distribution $p(s_{ij})$. This distribution then gives a rough estimate of a "target reward inequality" in the network: to what extent are rewards concentrated only among the top performing Bonfires.

Suppose we define an ideal distribution

$$g\left(v\right)$$

which controls the distribution of rewards (concentration of reward to the top performers). Then we can choose the function $f(d_{ij})$ such that we obtain the desired distribution of rewards, given the distribution $p(s_{ij})$. The function $f(d_{ij})$ then must be chosen such that it satisfies:

$$g(v) = \frac{d\left[\int_0^{\frac{1}{1+f^{-1}(v)}} p(s)ds\right]}{dv}$$

3.5.1 Linear Decay in Semantic Similarity

We now propose a specific function, motivated by simplicity, and explore the effects that would have on network wealth distribution.

Let us choose the gravitational energy to grow linearly with semantic similarity, $V_{ij} \sim s_{ij}$, or in terms of the distance, we propose,

$$f(d) = 1 + d.$$

We can then explore the impact on wealth distribution under different assumptions.

As a simple example, let's assume the distribution of semantic similarity is linear, such that, in the language of the previous section,

$$p(s) = 2s.$$

With our choice of function $f(d)$, the wealth distribution will be linear as well,

$$g(v) = 2v.$$

The function $f(d)$ can be subject to change by protocol governance in the future, so we provide a general guideline here of how a specific wealth distribution could be targeted. This can also change depending on how the distribution $p(s)$ varies as the network collects more data.

Suppose we find the distribution semantic similarity can be approximated by some polynomial, $p(s) = (a+1)s^a$ for some positive real number, a. Then we can choose a more general function,

$$f(d) = (d+1)^b,$$

for some positive real number, b.

We can then calculate the wealth distribution will follow,

$$g(v) = \frac{a+1}{b} v^{\frac{a+1}{b}-1}.$$

We then provide a guideline such that the parameter b can be fine tuned in the future, given the empirical data characterized by a. Suppose we want the wealth distribution to target some specific polynomial distribution of the form $g(v) = (c+1)v^c$, where c is a positive real number characterizing our target distribution. This then dictates the parameter b that we must choose to reach this target, given an empirical distribution for semantic similarity,

$$b = \frac{a+1}{c+1}$$

4 Reputation and other Mass Modifiers

To compute gravitational pull as defined in the previous sections, we now introduce a formal model for assigning mass to both output points and target points. These masses reflect the economic value, social trust, and informational weight of each point, and are derived from the activity and reputation of contributors, Bonfires, and target creators.

4.1 Mass of Output Points

Let k_i be an output point (a knowledge unit) produced by Bonfire $\mathcal{B}_i$ and author a_i. We define its mass m_i as a product of three components:

$$m_i = \mu_{\text{info}}(k_i) \cdot \mu_{\text{rep}}(a_i) \cdot \mu_{\text{stake}}(\mathcal{B}_i)$$

- $\mu_{\text{info}}(k_i)$ — **Informational weight of the output** : Reflects how much meaningful content is in the knowledge unit. This can be proxied by:

  – Retrieval count: $\text{retr}(k_i)$

  Example (simple model):

  $$\mu_{\text{info}}(k_i) = 1 + \log(1 + \text{retr}(k_i))$$

- $\mu_{\text{rep}}(a_i)$ — **Reputation multiplier of the contributor** : This scalar reflects trust in the author (or agent). Reputation may be derived from:

  – Past contributions' impact

  – Retrieval history across outputs

  – Identity proofs (e.g., verified academic credentials via DID)

  – Peer feedback

  Reputation can be normalized to a scale, e.g., $[0.5, 2.0]$:

  $$\mu_{\text{rep}}(a_i) = R(a_i)$$

- $\mu_{\text{stake}}(\mathcal{B}_i)$ — **Stake multiplier** : Captures how much economic value has been committed to the Target, and can include:

  – *KNOW* stake

  – Bonfire community currency token stake

  – Economic throughput of the output itself

  Example:
  $$\mu_{\text{stake}}(\mathcal{B}_i) = \log(1 + S(\mathcal{B}_i))$$
  where $S(\mathcal{B}_i)$ is the total stake in Bonfire $\mathcal{B}_i$.

Putting it together:

$$m_i = (1 + \log(1 + \text{retr}(k_i))) \cdot R(a_i) \cdot \log(1 + S(\mathcal{B}_i))$$

This formulation ensures that both the *quality* of content and the *reputation and risk* of its contributors are reflected in its gravitational contribution.

4.2 Mass of Target Points

Let t_j be a target point (a research bounty or topic embedding). Its mass M_j reflects how important the network considers this target.

We define:
$$M_j = \nu_{\text{fund}}(t_j) \cdot \nu_{\text{rep}}(o_j)$$

- $\nu_{\text{fund}}(t_j)$ — **Funding or stake weight** : Represents how many tokens have been committed to the target by its creator or the Bonfire:

$$\nu_{\text{fund}}(t_j) = \log(1 + F(t_j))$$

where $F(t_j)$ is the total staked funding.

- $\nu_{\text{rep}}(o_j)$ — **Reputation of the originator** : Similar to author reputation, we allow reputation of target issuers (individuals or institutions) to modulate mass:

$$\nu_{\text{rep}}(o_j) = R(o_j)$$

Thus:

$$M_j = \log(1 + F(t_j)) \cdot R(o_j)$$

This creates an incentive for high-reputation actors to issue high-priority targets, while still allowing open staking-based target creation in the future.

Time-weighted Stake Mass

We may also want to modify the stake weight by adding time-discounting to account for knowledge decay or freshness. Such that creating one target point once does not mean that it will permanently be an important target. Instead funds staked into a target point would create an amount of "mass over time" that can be defined by modifying the original term,

$$\nu_{\text{fund}}(t_j) = \log(1 + F(t_j))$$

to become

$$\nu^t_{\text{fund}}(t_j) = log(1 + F^t(t_j))$$

where

$$F^t(t_j) = f_\alpha(t)(1 - \alpha)F(t_j)$$

where $f_\alpha(t)$ is a decaying function over the real positive half line (can be for instance the Half normal probability density function) where α is a parameter controlling how quickly the function decays (for instance the variance for the half normal distribution). The higher the variance chosen (the slower decaying the function) this weight is removed from the total mass, the total funds becoming effectively now

$$(1 - \alpha)F(t_j)$$

.

4.3 Formal Model of Reputation

We now define the reputation components used in the mass formulas for both contributors and target originators. Let $R : \mathcal{A} \cup \mathcal{O} \to [r_{\min}, r_{\max}]$ be a function mapping authors $a_i \in \mathcal{A}$ and originators $o_j \in \mathcal{O}$ to scalar reputation scores.

4.3.1 Contributor Reputation: $\mu_{\mathbf{rep}}(a_i)$

We define $\mu_{\text{rep}}(a_i)$ as:

$$\mu_{\text{rep}}(a_i) = R(a_i) = \alpha_1 \cdot R_{\text{impact}}(a_i) + \alpha_2 \cdot R_{\text{social}}(a_i) + \alpha_3 \cdot R_{\text{identity}}(a_i)$$

where:

- $R_{\text{impact}}(a_i)$: score based on the cumulative semantic impact of past outputs (retrievals, reuse, rewards earned)
- $R_{\text{social}}(a_i)$: score based on peer endorsements or delegations of trust (e.g., weighted attestations, positive feedback, DAO votes)
- $R_{\text{identity}}(a_i)$: trust score based on credentials, verified identity, or institutional affiliation (e.g., PhD, research group membership, DID proofs)

The weights $\alpha_1, \alpha_2, \alpha_3 \in [0, 1]$ are configurable and should satisfy $\alpha_1 + \alpha_2 + \alpha_3 = 1$.

Example: Impact-Based Reputation Let $\{k_1, ..., k_n\}$ be the outputs authored by a_i, and let $\phi(k)$ denote the reward or quality score assigned to output k over time. We define:

$$R_{\text{impact}}(a_i) = \frac{1}{n} \sum_{k \in \mathcal{K}(a_i)} \phi(k)$$

Example: Identity-Based Reputation

$$R_{\text{identity}}(a_i) = \begin{cases} 1.5 & \text{if verified credential (e.g. PhD)} \\ 1.2 & \text{if affiliated with institution} \\ 1.0 & \text{default (anonymous)} \end{cases}$$

Example: Social Reputation (Delegated Trust) If $\text{Endorse}(u \to a_i)$ denotes a trust endorsement by user u toward a_i, we define:

$$R_{\text{social}}(a_i) = \sum_u w_u \cdot \text{Endorse}(u \to a_i)$$

where w_u is the weight or reputation of the endorser u. This endorse function can be handled by e.g. KNOW staking.

4.3.2 Target Originator Reputation: $\nu_{\mathbf{rep}}(o_j)$

Similarly, we define the reputation of a target originator o_j as:

$$\nu_{\text{rep}}(o_j) = R(o_j) = \beta_1 \cdot R_{\text{history}}(o_j) + \beta_2 \cdot R_{\text{social}}(o_j) + \beta_3 \cdot R_{\text{identity}}(o_j)$$

- $R_{\text{history}}(o_j)$: reward-weighted impact of past bounties or RFKs issued
- $R_{\text{social}}(o_j)$: trust scores derived from endorsement or delegated reputation
- $R_{\text{identity}}(o_j)$: institutional or verified identity reputation

Again, $\beta_1 + \beta_2 + \beta_3 = 1$.

Example: Historical Accuracy or Alignment Let $\{t_1, ..., t_m\}$ be previous targets set by o_j, and let $\phi(t)$ be the quality-adjusted reward effectiveness of the target (e.g., how well it guided Bonfires). Then:

$$R_{\text{history}}(o_j) = \frac{1}{m} \sum_{t \in \mathcal{T}(o_j)} \phi(t)$$

4.3.3 Normalization and Bounded Range

To ensure consistent contribution to mass computation, we normalize all reputation values into a bounded range:

$$R(x) \in [r_{\min}, r_{\max}], \quad \text{e.g., } [0.5, 2.0]$$

This prevents reputation inflation while preserving a strong differential effect between trusted and untrusted agents.

4.4 Reputation Updates and Time Evolution

Reputation scores may evolve over time through an update rule. A general form:

$$R_{t+1}(x) = \gamma \cdot R_t(x) + (1 - \gamma) \cdot \hat{R}_t(x)$$

where:

- $R_t(x)$ is the reputation at time t
- $\hat{R}_t(x)$ is the newly computed score from activities in epoch t
- $\gamma \in [0, 1]$ is a memory decay factor

This balances historical trust with responsiveness to recent performance.

5 Gravitational Energy between a Bonfire and the set of Targets

Having defined the gravitational force between individual output points and targets, we now compute the total gravitational "energy" exerted by an entire Bonfire across the semantic space of targets.

Bonfire Output Set Let $\mathcal{B}$ denote a Bonfire, and let $\{k_i\}_{i=1}^{N_\mathcal{B}}$ be the set of knowledge units (outputs) produced by $\mathcal{B}$. Each $k_i \in \mathcal{X}$ has an associated mass $m_i = m(k_i)$.

Target Set Let $\{t_j\}_{j=1}^{T}$ be the set of active target points in the semantic space, each with associated mass M_j.

Total Gravitational Energy of a Bonfire We define the total gravitational Energy, (in analogy with Newtonian gravity), $\Phi(\mathcal{B})$ of Bonfire $\mathcal{B}$ as the sum over all forces exerted by its outputs on all targets:

$$\Phi(\mathcal{B}) = \sum_{i=1}^{N_\mathcal{B}} \sum_{j=1}^{T} \frac{m_i M_j}{(d(k_i, t_j) + \epsilon)}$$

where:

- $d(k_i, t_j)$ is the semantic distance between output k_i and target t_j
- m_i is the mass of output k_i
- M_j is the mass of target t_j
- $\epsilon > 0$ is a small constant for numerical stability

This function quantifies the total influence of a Bonfire's contributions in the direction of active knowledge priorities, as defined by the target set.

6 Retrieval score

In the previous sections we have built the notion of total Gravitational Energy between a given Bonfire and the set of targets. This total gravitational energy dictates how much reward the Bonfire will get for producing knowledge that was relevant to a requested target.

It is possible that a Bonfire produces knowledge that is not particularly close to a given target, but is widely retrieved across the network, thus signalling it is a useful piece of knowledge even if it was not requested by the existing network Targets.

For this reason, there should be a second term to the reward function that relates to the total number of retrievals the outputs of the Bonfire get.

We can build a **Retrieval score** for each Bonfire.

Each time a certain Bonfire output is retrieved, there is an amount of retrieval fee associated with that retrieval.

The retrieval of a Bonfire over a period of time T is given by,

$$\mathcal{R}^T(\mathcal{B}) = M_j \sum_j \sum_i x_{i,j}$$

where i sums over all the Bonfire outputs k_i, and j sums over all the retrievals of output k_i that happened over the time period T. M_j is the mass of the Bonfire, and $x_{i,j}$ is the normalized retrieval fee associated with that retrieval.

Suppose retrievals work under a subscription service, the user l pays a monthly fee X_l for which allowed them to retrieve a total amount of data D_l over the course of that month. The total retrieval fees collected in that months correspond to summing over all users, $X = \sum_l X_l$, and $D = \sum_l D_l$ is the total amount of data to be retrieved by all users in that month.

Suppose a Bonfire output k_i consumes a total amount of retrieval data d_i out of the user's monthly budget D_l. We can then define the normalized fee as

$$x_{i,j} = \frac{X_l}{X} \frac{d_l}{D_l}$$

This retrieval score can now be put together with the total gravitational energy, to give a **Bonfire Score** of

$$\mathcal{S}(\mathcal{B}) = \Phi(\mathcal{B}) + f(X) R^T(\mathcal{B})$$

where $f(X)$ is a function that weighs how important each of the components of the score is. This weight could be a function of how much fee has been sinked into retrievals, compared to how much fee has been sinked into creating target, such that the weigh can vary depending on what users are valuing more: creating targets, or retrieving existing knowledge.

6.1 Reward Implications

This aggregate Bonfire score $\mathcal{S}(\mathcal{B})$ serves as the basis for computing token rewards in subsequent sections. It has several desirable properties:

- **Semantic Alignment**: Outputs closer to targets yield higher score
- **Contribution Quality**: Heavier outputs (more informative or higher reputation) yield higher score
- **Market Confidence Signal**: Targets with more stake-weighted mass (funding/tokens staked) pull more attention.
- **Usefulness of Knowledge**: Outputs from Bonfires that are highly retrieved are shown to be useful to network queries and drive revenue to the protocol, yielding a higher score

This formulation allows token incentives to be naturally driven by both content quality and alignment with research and protocol needs.

6.2 Total Supply and Epoch-based Reward Allocation

Let $S_{\max}$ denote the maximum token supply, fixed at network genesis. For example:

$$S_{\max} = 1{,}000{,}000{,}000 \quad \text{(1 billion tokens)}$$

We define a minting schedule $S(t)$ that determines how many tokens are allowed to enter circulation over time t. We define the **mining rate** as $R_t = \frac{dS(t)}{dt}$. We can impose the condition,

$$\lim_{t \to \infty} S(t) = S_{\max}$$

which implies that for large t, the rate R_t must decay at least as fast as $\sim 1/t$.

Suppose we now break the minting schedule into discrete epochs (e.g., monthly or weekly). The minting rate is now defined as the discrete difference

$$R_t = S(t+1) - S(t)$$

This budget R_t is then split among Bonfires using the Bonfire score:

$$\mathcal{S}(\mathcal{B}) = \Phi(\mathcal{B}) + f(X) \cdot \mathcal{R}^T(\mathcal{B})$$

as defined previously. The proportion of rewards for Bonfire $\mathcal{B}_i$ is then:

$$r(\mathcal{B}_i) = R_t \cdot \frac{\mathcal{S}(\mathcal{B}_i)}{\sum_{\mathcal{B}} \mathcal{S}(\mathcal{B})}$$

6.3 Alternative Design: KPI-Responsive Dynamic Minting

Here we propose, alternatively, that the minting rate R_t be not just a simple function of time, but one that can be adjusted based on network conditions and network KPI's.

Suppose we have N different KPIs. Let $\Theta \subset \mathbb{R}^N$ be the set of all possible states of our KPIs, let $\theta_t \in \Theta$ represent a specific state of these indicators at any given time t, and let $\theta_t^* \in \Theta$ be the vector of these target values at time t. We consider that to each KPI i there corresponds a (relative) level of importance $w_i \geq 0$ with $\sum_i^N w_i = 1$. This gives us a way of favoring one KPI over the other (or weight them all equally as $w_i = 1/N,\ \forall i = 1, .., N$).

We can then define a minting rate function, R_t that depends on how close the network state, θ_t is to the desired target θ_t^*.

We can define some notion of "distance". $\delta(\theta_i, \theta_i^*)$, between the vectors θ_t and θ_t^*, that takes into account our weights, w_i, such as

$$\delta(\theta_t, \theta_t^*) := \sum_{i=1}^{N} w_i \delta_i(\theta_{i,t}, \theta_{i,t}^*)$$

The "distance" (where the term is being used loosely here) function between each component and the target, $\delta_i(\theta_{i,t}, \theta^*_{i,t})$, can be defined such that if it is large $\theta_{i,t}$ it is far below θ_{i,t^*}, but it becomes zero if $\theta_{i,t} \geq \theta^*_{i,t}$ (notice that a standard distance metric would only be zero if the two elements are identical). For instance we can define,

$$\delta_i(\theta_{i,t}, \theta^*_{i,t}) = \text{MAX}\left[\frac{(\theta^*_{i,t} - \theta_{i,t})}{\theta^*_{i,t}}, 0\right]$$

This distance function is defined such that it only takes values between 0 and 1, as long as the KPI's are positive valued.

Lastly, let $\rho_m, \rho_M \in \mathbb{R}_{\geq 0}$ denote the minimum (possibly 0) and maximum minting rates at any moment in time. These upper and lower bounds provide some safety rails so that (i) at least some tokens are minted when things are not going well or (ii) we don't over mint when things are going well. For any fixed t, we can then define the instantaneous minting rate adjusment factor as:

$$\rho(\theta_t, \theta^*_t) = \rho_m + [\rho_M - \rho_m] \cdot (1 - \delta(\theta_t, \theta^*_t))$$

Now, we want to combine this with our previously defined function R_t, to define a new KPI-adjusted minting rate,

$$R^\theta_t = R_t \rho(\theta_t, \theta^*_t)$$

Now, suppose tokens are being minted at a rate R^θ_t instead of R_t. Since $R^\theta_t \leq R_t$, this would mean that some tokens may never be minted at all, since

$$\int_0^\infty R^\theta_t dt \leq S_{\max}$$

In this case, we can create a new feedback mechanism that ensures these un-minted tokens are not "lost" but instead are saved to be used in the future. This can be done via the following steps:

1. Start with the pre-defined function R_t.
2. For a given time epoch, t, find the minting rate adjustment factor, $\rho(\theta_t, \theta^*_t)$, and mint token at the actual rate, R^θ_t.
3. Re-adjust the future minting function R_t, such that the maximum supply $S_{\max}$ will still be minted at infinity. This can be done by adjusting R_t at time t.
4. This can be done by calculating what has been minted so far, $S^\theta(t) = \int_0^t R^\theta_{t'} dt'$, which allows us to compute the difference between what has been minted and the maximum that could have been minted, $\Delta S(t) = S(t) - S^\theta(t)$.

5. The pre-defined minting rate is re-calibrated to,

$$R_t \to \frac{S_{\max} + \Delta S(t)}{S_{\max}} R_t.$$

These steps ensure that all tokens that have not been minted will be used to increase future rewards.

6.4 Governance KPI Levers

The minting mechanism we have outlined leaves open the question of which KPIs are deemed important by the network and precisely how much weight to assign each of them. The precise configuration of network priorities is given by the choice of weights $\{w_i\}$, as well as the targets θ_i^*.

Initially, the set of weights will be calibrated by the DeSciWorld team and adjusted as the network evolves. Equally, the targets can be adjusted if the original targets no longer seem accurate. These choices can eventually be opened for calibration by network governance.

We finally propose a few possible KPIs that could be considered for inclusion in this mechanism (though this is not exhaustive, as more relevant KPIs are bound to arise as the network evolves):

1. **Total Gravitational Energy**: defined as $\sum_{\mathcal{B}} \Phi(\mathcal{B})$. This gives some measure of how much total relevant knowledge has been created in the network.

2. **Number of Targets**: this indicates how much knowledge is being requested from the network, and the demand for distinct pieces of knowledge

3. **Total Target Mass**: this metric includes how much total monetary stake (or TVL) there is in the network, as well as accounting for other factors, such as reputation.

4. **Total Bonfires Scores**: this indicates the total network value of existing Bonfires, including the amount of revenue they bring from retrieval fees and aggregate reputation of output.

7 Token Utility and Ecosystem Functions

The *KNOW* token plays multiple roles within the Bonfires.ai and DeSciWorld ecosystem:

1. **Deploy a Bonfire**

    (a) **Deployment** Tokens are required to instantiate a Bonfire.

(b) **Customisation** Additional tokens are required to add features that augment the core Bonfire product; third party feature providers may set their own fees structures.

2. **Maintain a Bonfire**

    (a) **Compute** Tokens are required to pay the costs of distributed compute for a Bonfire, various pricing models are supported: per query; credits; time-based with rate limits, across multiple providers.

    (b) **Storage** Tokens are required to purchase distributed storage plans from aggregated providers.

3. **Join a Bonfire** $KNOW$ staked in to a Bonfire becomes $wKNOW$ and will mint a internal, unique community currency at an issuance rate determined by each Bonfire. Unstaking will return the underlying $KNOW$ tokens.

4. **Network Issuance** Tokens are emitted as a baseline incentive for the contribution of knowledge to the open network, as discussed in this paper.

5. **Knowledge Target Bounty Funding** Tokens can be staked to fund bounties for new knowledge targets, adjust for qualities such as gravitational pull, mass, semantic value or reputation requirements, as determined by the bounty issuer.

6. **Custom Policy Requirements** Bonfires may set pricing for external retrievals of their knowledge, denominated in KNOW or other currencies. Retrievals from paid users do not count towards $KNOW$ emissions.

7. **Governance Participation** Token holders may vote on network upgrades, inflation parameters, target creation policies, and funding proposals.

Key Considerations

- **Price of KNOW**: the amount of KNOW required for 1), 2) and 3) are market driven and self adjusting as the price of KNOW changes.
- **Cost of Compute**: As compute costs change and alternative offerings for provision become available, the cost of hosting and maintaining a Bonfire, and thus the cost of retrievals, may change.
- **Price of Posting a Target**: The amount of KNOW that should be used to create a target point depends on how much demand there is to create other target points in general. This, if too little KNOW is spent, will not be an attractive target for Bonfires to provide knowledge outputs to.
- **Amount of KNOW staked**: Similarly, the amount of KNOW a Bonfire should stake in order to augment a high Bonfire score (to receive more KNOW rewards) is dictated by the total amount of KNOW staked across the network.

8 Bonfires.ai Genesis NFT sales and token allocation

KNOW is the currency of the Knowledge Economy. It is used to deploy and join Bonfires, to govern policy across Knowledge Network and to reward users for their knowledge contributions, alongside further utility.

The Genesis NFT sale is allocated 40 percent of the initial KNOW token supply, across 20,000 units, available at mint.bonfires.ai.

KNOW token contracts are upgradable, leaving room for economic policy changes including further minting requirements. It will be clear in 20 years whether the world has accepted centrally owned and governed collective intelligence systems or demanded and adopted a decentralised and cooperatively owned collective consciousness.