

Achieving Unanimous Consensus in Decision Making Using Multi-Agents

Apurba Pokharel

Department of CSE

University of North Texas
76207, Denton, TX, USA
apurba.pokharel@unt.edu

Ram Dantu

Department of CSE

University of North Texas
76207, Denton, TX, USA
ram.dantu@unt.edu

Shakila Zaman

Department of CSE

University of North Texas
76207, Denton, TX, USA
shakila.zaman@unt.edu

Sirisha Talapuru

Department of CSE

University of North Texas
76207, Denton, TX, USA
sirishamadh.talapuru@unt.edu

Vinh Quach

Department of CSE

University of North Texas
76207, Denton, TX, USA
vinh.quach@unt.edu

Abstract—Blockchain consensus mechanisms have relied on algorithms such as Proof-of-Work (PoW) and Proof-of-Stake (PoS) to ensure network functionality and integrity. However, these approaches struggle with adaptability for decision-making where the opinions of each matter rather than reaching an agreement based on honest majority or weighted consensus. This paper introduces a novel deliberation-based consensus mechanism where Large Language Models (LLMs) act as rational agents engaging in structured discussions to reach a unanimous consensus. By leveraging graded consensus and a multi-round deliberation process, our approach ensures both unanimous consensus for definitive problems and graded confidence for prioritized decisions and policies. We provide a formalization of our system and use it to show that the properties of blockchains: consistency, agreement, liveness, and determinism are maintained. Moreover, experimental results demonstrate our system’s feasibility, showcasing how our deliberation method’s convergence, block properties, and accuracy enable decision-making on blockchain networks. We also address key challenges with this novel approach such as degeneration of thoughts, hallucinations, malicious models and nodes, resource consumption, and scalability.

Index Terms—Blockchain, Consensus, Deliberation, Multi-Agent, LLMs, Decision-Making.

I. INTRODUCTION AND MOTIVATION

Lack of Decision Making Capabilities in Blockchain: Blockchains introduced a new way of processing financial transactions, without having to rely on any central authority. The existing consensus mechanisms, including Proof of Work (PoW) [1], Proof of Stake (PoS) [2], and Delegated Proof of Work (DPoW) [3] are primarily designed by focusing on majority or weighted consensus to tolerate faults in a network, are currently highly applicable for financial (e.g., cryptocurrency) applications. These fault tolerance features of existing consensus mechanisms impose significant challenges in applications (e.g., legal system (Jury), community and governmental policies, constitutional amendments, etc.) where a majority vote may risk overlooking crucial arguments [4]. Additionally, these protocols are not implemented in a way that can process the inclusion of every participant’s opinion to reach a unanimous agreement. Moreover, smart contract-based approaches are being considered to solve this problem, however are not efficient enough [4]. Therefore, a novel consensus mechanism is required to enhance decision-making

capabilities through active communication between all active participants in the network.

Requirements of Deliberation for Consensus: Requirements including active communication, agreement or disagreement, validation, and inclusiveness can only be achieved through structured deliberation, which ensures a decision-making is fair and collaborative [5]. Deliberation has been used extensively in a wide variety of areas in computer science for decision-making, from robotics [6], governance [7], Artificial Intelligence (AI) [8], Large Language Model (LLM) [9] and more. Participants use deliberation to determine what action or course of action should be taken [10] to achieve a decision regarding a particular problem statement. This work is motivated by the literature review described above, which effectively illustrates the influence of utilizing deliberation to accelerate the quality of decision-making performance.

LLMs, the Superior Choice for Deliberation: Deliberation has been highlighted and implemented in myriad applications to essentially design a responsive system by refining individual perspectives through well-designed rounds [11]. However, when it comes to choosing the type of deliberator there are many different options including humans and Natural Language Processing (NLP) models. Human deliberators are often limited with expertise whereas traditional NLP has to be trained on a large corpus of data for different expertise which may not be feasible all the time. LLMs have become extremely effective at understanding complex linguistic patterns and are used for a variety of tasks, such as question-answering, machine translation, and advanced natural language processing (NLP) [12]. LLMs enhance deliberative conversations by rapidly processing and summarizing diverse information, facilitating open-ended discussions, and addressing complex queries [13]. Despite challenges like bias, hallucination, and resource costs, their advantages outweigh these limitations. Therefore, the ability to enhance decision-making capability based on diverse perspectives makes LLMs a superior choice for deliberation.

Multi-Agent Strategies for Effective Deliberation: The performance of deliberation accelerates unanimous decision-making through iterative rounds by agents. Through argumentative reasoning, agents can support their decisions with additional information to help convince each other. Moreover,

agents exchange updated states to maximize convergence speed and accurately minimize inconsistencies. The significant impact of using a multi-agent system is described in Marvin Minsky's Society of Mind article [14], which conceptualizes the mind as a collection of interacting sub-modules, each contributing to complex thought and behavior. This work is motivated to use a multi-agent approach from enormous prior work including [9], [15], and [16], where it has been clearly demonstrated that leveraging multiple LLMs in deliberative processes improves output accuracy and inclusivity.

II. PROBLEM DEFINITION

The key idea of implementing consensus mechanisms in distributed blockchain networks is to ensure trust and security towards maintaining a unified ledger which guarantee consistency on agreeing on a single version of truth without a central authority. A wide range of variations exists on current consensus protocols [17], however mostly accepts majority votes among active participants to finalize an agreement as fault tolerance feature of networks. While these cutting-edge protocols are effective in many applications, their usability limits in applications such as legal systems (e.g., Jury), ethical committees, government policy-making systems, and more, which require critical decision-making capabilities that involve all active participants to generate unanimous consensus. To engage participants and generate an effective convergence towards achieving decisions, deliberation has been widely used [11]. However, addressing challenges such as efficient participant engagement, structured argumentative reasoning, and proposal refinement requires a well-designed deliberation technique. Additionally, biases or influence of limited knowledge of participants hinders generating a fair and reliable decision-making process. Moreover, complex features and the inherently time-consuming process of transaction creation of decentralized blockchain networks significantly impact on reaching unanimous consensus.

III. CONTRIBUTION

By addressing the above research gap of producing a well-designed deliberation with decision making capabilities, we introduce the key following contributions. To the best of our knowledge, this is the first work for achieving a robust unanimous consensus in blockchain networks.

- A multi-agent based deliberation framework is proposed to achieve unanimous consensus. (See Section V, VII) (See Figure 1)
- LLMs are employed to generate structured arguments for reasoning the decisions or responses of each agent. (See Section VI) (See Figure 7, 4)
- The deliberation framework facilitates a finite number of rounds to resonate valid decisions through refinements. (See Section VII) (See Figure 2)
- Protocol correctness is proven by addressing significant properties such as consistency, agreement, liveness, and determinism in blockchain networks.(See Section X)
- A list of system challenges is described. (See Section XI)

- The multi-agents-based deliberation is implemented in the Nimiq Blockchain to measure the performance in terms of the time it takes to converge towards consensus and describe the relationship between dynamic factors on convergence. (See Section XII)

IV. LITERATURE REVIEW

A. Blockchain

Blockchain consensus mechanisms prioritize majority-based decision-making, which is insufficient for applications requiring nuanced deliberation. This section reviews existing works on blockchain consensus, deliberation, and decision-making in distributed systems.

Lamport et al. addressed the Byzantine Generals Problem by establishing that for a distributed system to function correctly, the number of honest nodes must satisfy $n > 3t + 1$, where n is the total number of nodes and t is the number of dishonest nodes [18]. This fundamental principle laid the groundwork for distributed consensus mechanisms. Bitcoin later introduced a novel approach to financial transactions in distributed systems, ensuring finality through majority agreement among honest participants [1]. Even in a fully honest network, finality is achieved once the majority reaches a consensus. Subsequent consensus protocols, such as Proof of Stake and Delegated Proof of Stake, follow a similar paradigm, where selected validators or stakeholders finalize decisions efficiently [2]. This mechanism preserves system integrity while optimizing transaction processing speed to remain competitive with centralized systems. However, existing consensus models prioritize majority agreement over full inclusivity, potentially overlooking critical perspectives.

TABLE I
EXISTING APPROACHES OF LLM TOWARDS PROBLEM SOLVING

Approach	Findings	Reference
Personas	Used in fandom/educational role-playing, storytelling and more.	[19], [20]
Iterative Collaboration	Iterative dialogues between single or multiple agents ensure convergence toward the correct answer through discussion, reflection and refinement.	[9], [16]
Moderators	Moderators evaluate the quality of the generated response and help increase the overall response quality through iterative refinement.	[21], [15]
Advance (Storage, Tree of Thought, Graph Prompting)	Technique that makes use of Retrieval-ICL (RAGs), and tree and graphs to generate best response	[22],[23], [24]

Blockchain governance has traditionally relied on voting-based mechanisms, either through on-chain smart contracts or off-chain deliberation, often limiting meaningful discussion among stakeholders. Existing models, such as Aragon Court [25] and Kleros [26], focus on fact-based adjudication rather than structured communication, resembling traditional legal systems where jurors assess evidence independently without active engagement. Similarly, off-chain governance solutions

like Decred’s Politeia [27] and Snapshot [28] allow stakeholders to vote but lack structured deliberation, making decision-making rigid and less inclusive. Furthermore, the reliance on off-chain discussions introduces transparency and security concerns, as critical decision-making steps occur outside the blockchain. For a comparison of existing consensus protocols, we direct the reader to Table I in Section I of [4]. This work aims to introduce a new paradigm by leveraging distributed networks for problem-solving through unanimous decision-making, ensuring every honest node actively participates in the deliberation process, and fostering a more inclusive and robust consensus.

B. Deliberation and LLMs

This section reviews existing work on problem-solving with LLMs and explores the role of deliberation in this context. In-context learning (ICL) enables LLMs to address tasks at run time without requiring dedicated training via forward and backward propagation for each specific use case [29]. This capability has spurred extensive research, resulting in various prompting and modeling techniques designed to elicit more accurate and well-reasoned responses. While the aim is not to introduce novel methods or set new benchmarks in LLM problem-solving, the focus is on integrating and/or combining the most effective existing approaches into a deliberative framework designed to enhance decision-making in blockchain consensus mechanisms.

ICL began with a few shot and chain of thought prompting [30] techniques. These techniques combined with a sequential process of prompting and reprompts allowed research to introduce new approaches to better solve problems in LLMs.

Approach of assigning personalities to a model and asking it to behave as such is called personas [19], [20]. A model can be trained to behave this way though either prompting it or though fine-tuning the model as done in [20]. This approach however does not help us deliberate effectively as they are more focused on providing users with enhanced experience through persona interaction or focused on storytelling. In addition to this, researchers also explored multi-model approaches, such as [31], where multiple models iteratively collaborated to refine their responses—a method known as iterative collaboration. This approach allowed models to enhance problem-solving by working together. Further advancements introduced specialized roles, such as judges or moderators [21], [15], which provided feedback and guided deliberation. This led to the development of a new paradigm incorporating moderators to improve reasoning and decision-making. Advanced ICL with RAGs [22], Tree of Thought TreeOfThought, and Graph Prompting [24] were introduced which leverage external knowledge sources, hierarchical thinking, and graph-based dependencies to guide model reasoning.

While approaches 1 and 4 in Table I¹ enhance efficiency in specific problem-solving contexts, our requirement remains a structured, multi-round deliberation, where agents iteratively

refine and critique responses. Furthermore, we focus on deliberation where deliberators have the same roles, intentionally avoiding a moderator-based approach. However, Chan et al. [21] observed performance degradation when models with identical roles engage in deliberation. To address this, we employ iterative collaboration between models with different backgrounds, mirroring real-world deliberations where individuals from different domains contribute to the discussion.

V. SYSTEM OVERVIEW

System Goals: The following essential fundamentals must be present in order to design a deliberation framework that effectively promotes unanimous consensus:

- 1) Fundamentals of blockchain must not be violated.
- 2) Fraudulent/Fraud/False Decisions must never be achieved.
- 3) Convergence should be as fast as possible but not at the expense of poor or compromised result.

Consensus in distributed systems is inherently expensive in terms of time and the number of message required [18], but the goal of achieving unanimous consensus further amplifies these costs due to the increased number of rounds and extensive communication required.

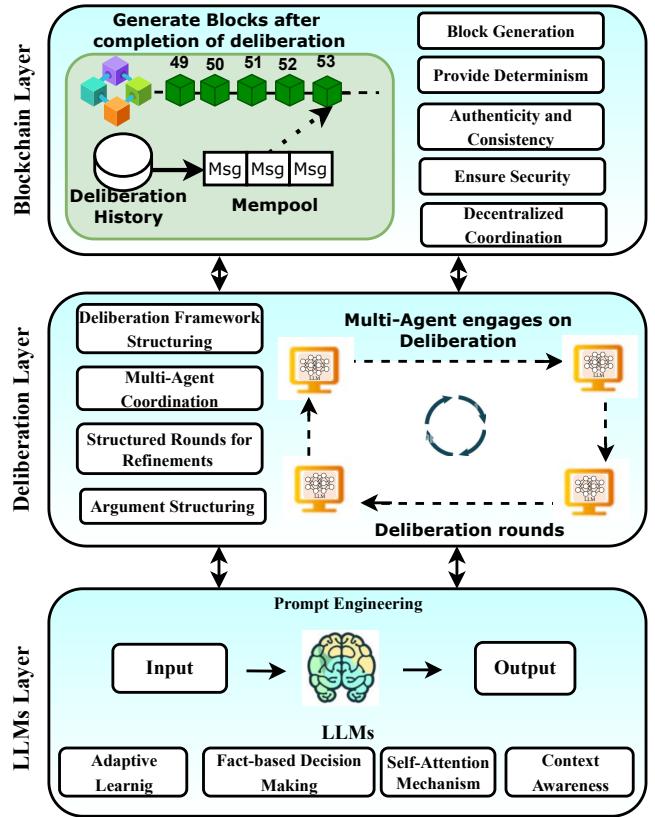


Fig. 1. Figure shows the layerwise representation of the system. The bottom layer is the LLM layer that produces arguments used in deliberation. The deliberation layer defines the structure and properties of deliberation while the blockchain layer makes sure that a secure deliberation can be done.

System Assumptions: In this work, various key assumptions are established to ensure the system operates fairly

¹Each approach may have an overlap with the other

and effectively and is capable of facilitating a meaningful unanimous agreement.

- 1) Most byzantine consensus protocols require more than 2/3 of nodes, to be honest, [18] and we work under a similar assumption.
- 2) We assume a strongly synchronized network where honest nodes communicate within some bounded delay.
- 3) We assume that the a solvable problem is selected for deliberation. Therefore we define unanimous consensus as the agreement among honest nodes.

A. Layer-based System Architecture

A layer-based architecture has been utilized and the layers of the system are as follows. Figure 1 represents the layer-based system architecture.

- 1) **Blockchain Layer:** Blockchain is the infrastructure that is responsible for recording and verifying the deliberation process and its outcomes. The term node/model/agent is used interchangeably as nodes are extended to have models within them and end up behaving as an AI agent.
- 2) **Deliberation Layer:** This layer defines the structures for deliberation. It also defines deliberation parameters such as: number of turns, number of agents, and more.
- 3) **LLMs-based Multi-agents Layer:** This layer is responsible for generating responses based on the prompts passed to it by the deliberation layer. The response generated by an LLM is generally called a reply but we will use the term utterance.

VI. LLMs-BASED MULTI-AGENTS

Autonomous decision-making capability, structured deliberation, and cognitive communication are guaranteed by the advantage of LLMs. LLMs generate utterances through various prompting techniques that facilitate valid arguments for deliberation.

A. Prompt Engineering to Optimize LLM Performance

The utterance from an LLM greatly depends on the nature of the prompt given to it. We use two kinds of prompting techniques defined in the subsection below.

1) *Initial Prompting:* The initial prompting phase gathers each model's (LLM_N) response r_N to the problem Pr , storing them in the response set R^0 for deliberation. Chain of Thought (CoT) prompting improves response quality over Zero Shot (ZS) prompting [30]. To mimic real-world deliberation, where opinions evolve through argumentation [10], we use a mix of CoT prompts (P_{CoT}) and ZS prompts (P_{ZS}). This hybrid approach (see initial round in VII-A) enhances deliberation realism, as shown in XII-G. The algorithm is defined in 1.

2) *Iterative Prompting/ Prompt Chaining:* In order to allow change of opinions to take place the models are prompted in iteration where each iteration is defined as turns T , see turns in VII-B. In each turn t , each agent receives its own utterance from the previous turn i as R_i^{t-1} and the previous turn's neighbour agent's responses as R^{t-1} as context and is tasked with evaluating and improving its own answer. After

Algorithm 1: Initial Prompting

```

Input: Problem  $Pr$ , Large Language Model  $LLM$ ,
        Number of Agents  $N$ 
Output: Set of Initial Responses  $R^0 = \{r_1, r_2, \dots, r_N\}$ 
Initialize response set  $R^0 \leftarrow \emptyset$ ;
for  $i \leftarrow 1$  to  $N$  do
    if  $i$  belongs to CoT group then
        // Assign CoT prompt  $P_{CoT}$  to  $LLM_i$ 
         $P_i = P_{CoT}$ ;
    else
        // Assign ZS prompt  $P_{ZS}$  to  $LLM_i$ 
         $P_i = P_{ZS}$ ;
    end
    Generate response  $r_i = LLM_i(P_i, Pr)$  ;
    Add  $r_i$  to response set  $R^0$  ;
end
return  $R^0$ 

```

the end of each iteration we observe more agreement on the correct answer, see XII-D. The algorithms is defined in 2.

Algorithm 2: Iterative Prompting (Prompt Chaining)

```

Input: Question  $Pr$ , Large Language Model  $LLM$ ,
        Number of Agents  $N$ , Number of Turns  $T$ 
Output: Response Set for each round  $R^T$  and total
        Response Set  $R$ 
Initialize  $R^0$  using Algorithm 1 ;
for  $t \leftarrow 1$  to  $T$  do
    for  $i \leftarrow 1$  to  $N$  do
        Construct new prompt  $P_i^t$  using previous
        responses:
         $P_i^t = \text{Concatenate}(Pr, R_i^{t-1}, R^{t-1})$  ;
        Generate new response  $r_i^t = LLM_i(P_i^t)$  ;
        Add  $r_i^t$  to  $R_t$ 
    end
    Add  $R^t$  to  $R$ 
end
//  $R^t = \{r_1^t, r_2^t, \dots, r_N^t\}$   $R = \{R^0, R^1, \dots, R_N^T\}$ 
return  $R$ 

```

B. Dataset

To assess the problem-solving capabilities of our deliberation framework, we evaluate performance across multiple datasets that are given below:

- 1) Evaluate the model's ability to solve mathematical problems, Arithmetic [32] problems. This dataset comes from hugging face and contains 1 million rows with 3 columns: problem, answer and solution. See XII-E
- 2) Next, we use the wason card game dataset provided by [33]. This dataset contains of 500 real-world deliberation dialogues on the Wason card game problem. The number of people involved varies from 2 to 5. See XII-D
- 3) Additionally, to test out our model's behavior in real world deliberation scenarios like deciding on policies we perform deliberation for such use cases as well. See XII-C

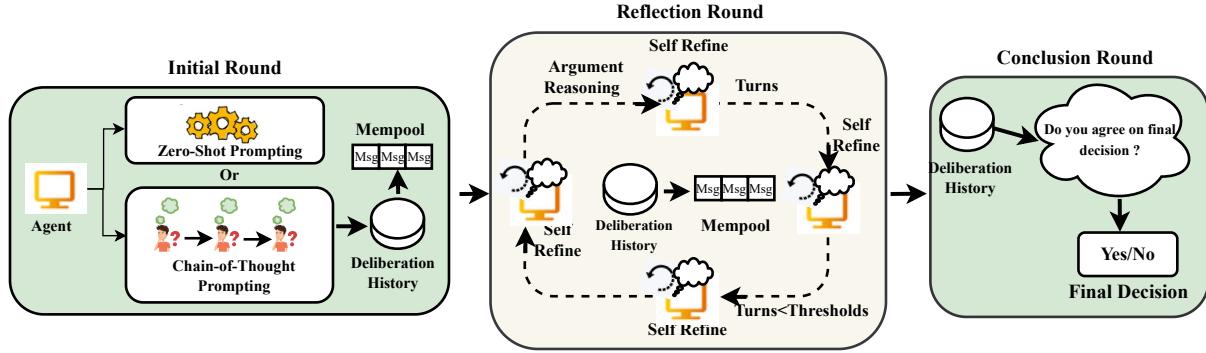


Fig. 2. The figure illustrates the deliberation process across three rounds. The initial round gathers each model’s initial opinion on the problem. The reflection round facilitates deliberation, allowing models to refine their viewpoints. Finally, the conclusion round consolidates the refined utterances from the reflection round, presenting them as viable solutions, with each model expressing its final stance.

VII. DELIBERATION FRAMEWORK

In this section, the structure of deliberation is defined along with the various factors that play a crucial role in converging towards consensus.

A. Structure of Deliberation

Deliberation is a long and careful discussion of a given problem. To replicate the real deliberation process, the discussion is conducted in several rounds. See figure 2.

1) *Initial Round*: At the beginning of the deliberation process initial prompting (refer algorithm 1) is employed to get the individual opinions of each model.

2) *Reflection Round*: After that, prompt chaining (algorithm 2) is used in order to allow the agents to reflect and refined their utterance. This is done for a finite number T turns.

3) *Conclusion Round*: For problems that have a definitive answer like the problems 1 and 2 in section VI-B this is not used in our approach whereas for prioritized preferences and ranked choices like problems 3 we use this additional round of deliberation. An example of deliberation can be seen in figure 3. This deliberation structure closely resembles [9] and other similar work.

B. Progresses of deliberation through Turns

This is one of the very crucial parameters of the deliberation and plays a role in determining the accuracy as well as the convergence ratio, shows in section XII-C. In an ideal real-world deliberation scenario, at the start there is a lot of diverging opinions, and as the deliberation progresses more and more individuals agree upon the best and optimal answer. This behavior is mimicked through iterations/turns which executes the reflection round for T times. A simple round robin approach is used to determine the next speaker throughout the multiple rounds.

C. Number of agents

The number of agents determines the diversity of perspectives in the deliberation process. A higher number of agents introduces more varied viewpoints, leading to a richer exploration of the problem space, refer to XII-D. However,

there exists a tradeoff, as increasing the number of agents also raises computational costs and the potential for prolonged convergence time. Figure 9 presents the impact of varying the number of agents on convergence and accuracy.

D. Decision Making

The deliberation system provides an iterative way of refining the arguments so that agents can collaboratively improve their responses. Each round involves proposing modifications, justifications, and evaluations until all agents agree on the latest version without further changes. This process ensures a well-reasoned, collectively optimized solution, making it ideal for complex decisions. This approach is effectively used for improving llm accuracy in decision making [34], [35], [36].

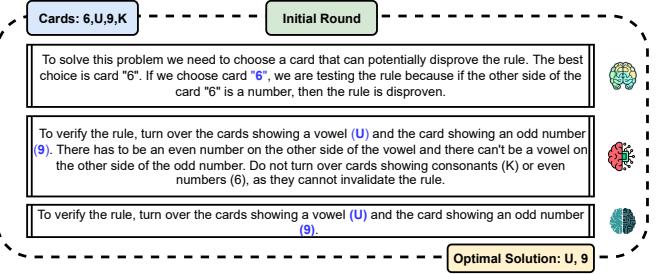


Fig. 3. Initial Round for Wason Card game [33]: The game objective is to select cards that will validate the rule: All cards with vowels on one side will have an even number on the other. The optimal solution is shown in the bottom right side.

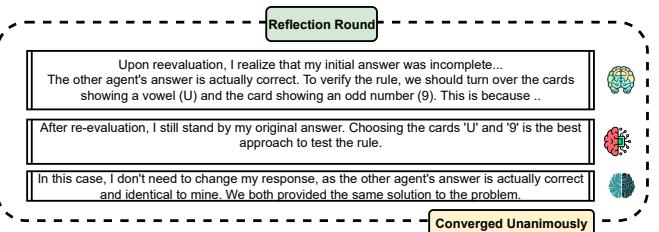


Fig. 4. Reflection Round for Wason Card game: At the end of the first reflection round ($T=1$) all three models unanimously converge on the correct answer.

VIII. INTEGRATION OF DELIBERATION IN BLOCKCHAIN

This section represents how the blockchain functionalities help to shape the performance of LLMs-based multi-agent deliberation.

A. Where does LLMs fit in?

LLMs are a part of the blockchain node. Each node will be running its own LLM model and will call the model with prompts as mentioned in algorithm 1 and 2. Once the model replies, the utterance is propagated to every node in the network using a gossip protocol, refer section VIII-C.

B. Consistency and Authenticity

Deliberation utterances are stored in the mempool and propagated via an extended gossip protocol to ensure all nodes share the same state. Each utterance is signed by the producer agent, allowing recipients to verify authenticity against the agent's address.

C. Gossip Protocol

Gossip protocol is a common way to enable information exchange in a distributed system. In order to ensure efficient communication and reduce unnecessary transmission, a two-phase gossip protocol is used. In phase one the new information's (block, transaction or utterance) hash is sent to the neighbor nodes. These are of constant size and are propagated quickly. Then the neighbor node requests the information if they don't have it.

D. Storage and Block Structure

Once consensus is achieved the entire deliberation is stored on-chain to enable transparency and future dispute resolution if and when needed.

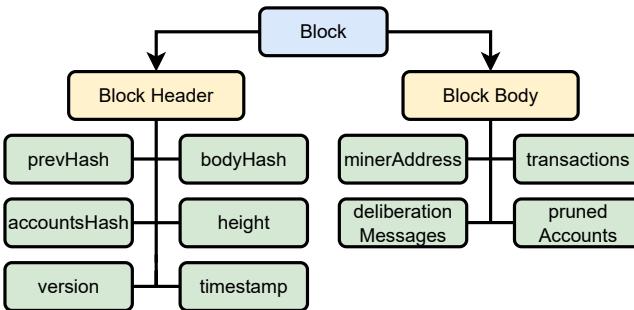


Fig. 5. BlockStructure: A block consists of block header and block body. The block header stores relevant information that maintains the chain of block where as the block body stores the transactions and the utterances during deliberation.

E. Verification and Extension of Ledger

The newly generated block is propagated throughout the network and each model will be able to compare the contents of the block to ensure the deliberation has not been tampered with. Upon successful verification, a new block is appended to the ledger to have consistence block state and gossiped to the neighbors. If the deilebration round does not converge within

\mathcal{T} time then an empty block will be mined. This decision is based on the principle, 'failure to agree is agreement to defer,' as proposed in [37].

IX. MODELING UNANIMOUS CONSENSUS

A. Deliberation Game

The deliberation protocol is modeled as a cooperative game in which agents provide arguments that can be countered, refined, or enhanced. The end goal of this game is to arrive at a unanimous consensus through multiple iterations of the game rounds.

Definition IX.1. Let D be n player game where A_n represents the n th player, work together to solve a problem P by performing valid actions AC_r in each iterative round r . The goal of the game is to achieve unanimous consensus with confidence C within a timebound \mathcal{T} with high confidence and the players receive a payoff PAY based on their contribution. Where, $D^i = (P, A_n, AC_r, PAY, C, t)$, D^i is the i^{th} deliberation done in the network, that will be stored in a new block B and t is the time in which the delilberation completes.

B. Start of deliberation

A deliberation game starts if the problem P hasn't been deliberated before, else if the problem was hung before (hung deliberation), then the same agents cannot be involved in the deliberation.

Definition IX.2. The deliberation game D is said to be initiated if it meets one of the following criteria:

- 1) The problem P hasn't been deliberated before.
- 2) If deliberation was hung before $P \in H_g$
Let problem P is included in the k th deliberation $P \in D^k$
The hung deliberation is mined in the empty block $D^k \in B^K$
And deliberators involved previously is $A_n^k \in D^k$
Then $A_n^k = A_n^i$, where i is the current deliberation.

C. Unanimous Consensus Criteria

The type of consensus reached depends on the nature of the problem being addressed. Definitive problems (problem 1 and 2 from VI-B) have a single, unique solution. In contrast, problems involving prioritized preferences (problem 3 from VI-B) do not, giving rise to the notion of confidence in consensus. The idea of Graded Consensus is used from [38].

1) Unanimous Consensus for Definitive problems:

Definition IX.3. In a definitive deliberative game D , we say that D is in unanimous consensus if, for every n agents with m malicious agents, each honest agent h outputs a value-argument pair $AC_r^i = (V_r^i, Arg_r^i)$ at the end of round r where the value is the answer to the problems along with arguments that support it, so that $\forall i, j \in h : V_r^i = V_r^j, C = 1$ and $t < \mathcal{T}$.

2) *Unanimous Consensus for prioritized problems:*

Definition IX.4. In a definitive deliberative game D , we say that D is in unanimous consensus if, for every n agents with m malicious agents, each honest agent h outputs a policy $AC_r^i = \{Po_1^i, Po_2^i, \dots, Po_x^i\}$ at the end of round r where Po_k are the policies proposed and are relevant to problem P , so that θ is the fraction of honest agents that include policy Po_k in their action set and $t < T$.

3) *Agreement and Confidence in Consensus for prioritized problems:*

Definition IX.5. The confidence in the achieved consensus is measured as follows:

Let $AC_r^i = \{Po_1^i, Po_2^i, \dots, Po_x^i\}$ be the set of policies proposed by agent i at the end of round r .

For each unique policy Po_k , its agreement level is defined as:

$$A(Po_k) = \frac{\sum_{i \in h} \mathbb{1}(Po_k \in AC_r^i)}{|h|}$$

where $\mathbb{1}(Po_k \in AC_r^i)$ is an indicator function that is 1 if agent i included Po_k in their argument set AC_r^i , and 0 otherwise.

If the agreement level is greater than θ than the policy is added to the accepted policy set $Acc(Po)$

$$A(Po_k) >= \theta$$

$$Acc(Po) = Acc(Po) \cup A(Po_k)$$

The overall confidence in consensus is then computed as:

$$T = \text{len}(Acc(Po))$$

$$\forall A(Po_a) \in Acc(Po), C = \frac{\sum A(Po_a)}{T}$$

where C represents the confidence in consensus.

D. *Termination of deliberation*

1) *After reaching consensus:*

Definition IX.6. The deliberation is said to be successfully terminated if the criteria for unanimous consensus are achieved. The deliberation parameters D^i is stored on chain and a new block is created.

2) *Hung Deliberation:*

Definition IX.7. The deliberation is said to be in hung deliberation if: $t > T$ or if the number of honest participants falls below a threshold. The deliberation is then terminated by adding problem P to the hung problem set H_g : $H_g = H_g \cup P$ and producing an empty block along with the deliberation parameters D^i .

X. SYSTEM PROPERTIES

A. *Consistency*

Theorem X.1. *Proof of consistency.*

1) *For definitive problem:* For a definitive problem P , initially honest player(s) $l \in h$ will output a value-argument pair (V_r, Arg_r) at the end of round r such that (V_r) is same for all players l in the group.

After $r_1, r_1 > r$ rounds of deliberation all honest player $i, j \in h$ and $l < j, i$ outputs a value-argument pair (V_{r_1}, Arg_{r_1}) such that $\forall i, j \in h: V_{r_1}^i = V_{r_1}^j$. This implies that every honest player will output the same value ensuring consistency.

2) *For Prioritized Problems:* For a prioritized problem P , initially, each honest agent $i \in h$ outputs a set of proposed policies at the end of round r :

$$AC_r^i = \{Po_1^i, Po_2^i, \dots, Po_x^i\}$$

where Po_k^i represents a policy proposed by agent i at the end of round r . The confidence C_1 in consensus is low.

After $r_1, r_1 > r$ rounds of deliberation, every honest agent $i, j \in h$ outputs a refined set of policies:

$$AC_{r_1}^i = \{Po_1^{r_1}, Po_2^{r_1}, \dots, Po_T^{r_1}\}$$

such that the confidence C_2 in consensus is higher than before $C_2 > C_1$.

This only occurs when the policies are selected consistently by more players at the end of the round r_1 . Thus, for policies that are accepted and added to $Acc(Po)$, the consistency property is maintained.

B. *Agreement*

Corollary X.1.1. *Proof of agreement.*

The consistency property goes hand in hand with agreement. There can be no consistency without agreeing on the same values or policies and no agreement without consistent outputs.

C. *Liveness*

Liveness guarantees that the network continues to make progress despite potential faults.

Theorem X.2. *Proof of liveness*

Based on the definitions IX.6 and IX.7 it can be stated that block production continues for both of these cases ensuring liveness of the network.

D. *Determinism*

Determinism in blockchain ensures that given the same input, all nodes will produce the same output. The input includes utterances alongside transactions.

E. *Determinism*

Theorem X.3. *Proof of Determinism*

1. Let Txn be the set of all valid transactions. At time t , transactions generated are:

$$Tx_t = Rand(Txn)$$

where $Rand()$ denotes the randomness in transaction generation.

2. Transactions in the mempool form a block with header:

$$\text{Header}_{block1} = H(Tx_t)$$

Since H is deterministic, all nodes derive the same Header_{block1} given they have the same Tx_t in the mempool.

3. Let Utt_t be the set of utterances whose generation can also be modelled through $\text{Rand}()$ at time t :

$$\text{Header}_{block2} = H(Utt_t)$$

Thus, both transactions and deliberation utterances maintain determinism:

XI. SYSTEM CHALLENGES

A. Drawbacks of Using LLMs

Even though LLMs have continuously showed impressive reasoning and creative ability they are far from perfect. Some of the known drawbacks are as:

- 1) **Hallucination:** LLMs sometimes generate factually incorrect or entirely fabricated information, leading to unreliable outputs [39].
- 2) **Lack of Understanding and Explainability:** LLMs generate responses based on statistical patterns without true comprehension and operate as black boxes, making output tracing difficult [40].
- 3) **Inconsistency and Degeneration of Thought:** LLMs can produce different answers based on phrasing and context [41] and suffer from degeneration of thoughts (Dot) [15], where they fixate on a solution, limiting diverse perspectives.

B. Model Corruption/ Misbehaving Nodes

LLMs are susceptible to security risks from biased training data and adversarial prompts [42]. Likewise, deliberation nodes may deviate from rules due to adversarial intent or misaligned incentives. **Potential Solution:** A judge system [15] can assess utterance reliability using confidence scores [43] or intervene against irrelevant statements. The PAY mechanism [44] enforces integrity by rewarding honest participation and penalizing deviations.

C. Resource Consumption and Scalability

Deploying LLMs remains resource-intensive, requiring high-end GPUs and significant power, limiting scalability. **Potential Solution:** Efficient fine-tuning [45], [46] enables smaller models to have similar performance to bigger ones, lowering costs. On the other hand on-chain deliberation increases storage demands. Off-chain storage (e.g., IPFS [47]) mitigates this but weakens security, highlighting the blockchain trilemma [48].

D. Verifiability of LLM-Generated Arguments

Potential Solution: Watermarking [49] embeds hidden signals in LLM outputs with minimal impact on text quality, enabling later verification. Additionally, techniques like DetectGPT [50] can identify LLM-generated text, though they remain imperfect and computationally expensive.

E. Block Exploitation / Sybil Attack

This work utilizes LLMs for deliberation. While their resource-intensive nature limits Sybil attacks to some degree, an adversary could still control multiple nodes by sharing a single LLM. **Potential Solution:** A state-based deliberation system [2], helps mitigate this risk. Additionally, reputation-based incentives [51] further discourage Sybil attacks and exploitation.

XII. EXPERIMENTAL RESULTS

A. Implementation Detail

We use the Nimiq blockchain [52], which has two consensus models: proof of stake (PoS) [53] and a browser-based proof of work (PoW) client written in JavaScript. For our proof of concept, we extend Nimiq PoW client. We used Llama-3.1-70B-Instruct (quantized) and used NVIDIA RTX 6000 Ada Generation GPU.

B. Deliberation Example

An example of deliberation is shown below in figure 6 as well as 3.

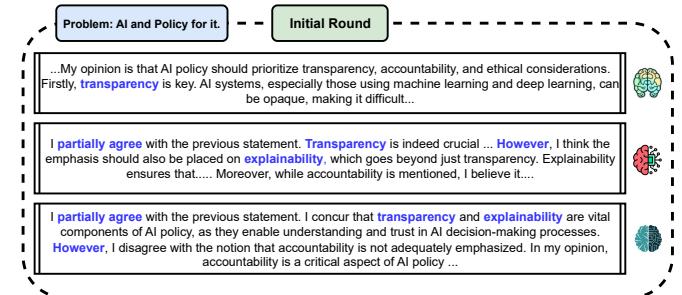


Fig. 6. Initial Round Policy Deliberation: The game objective is to select and agree on policies for the problem.

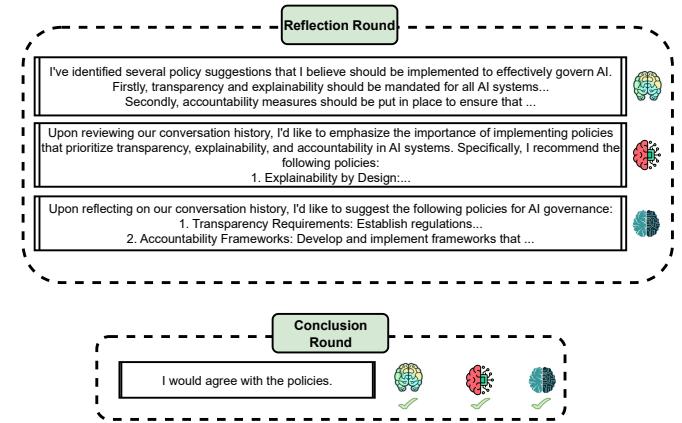


Fig. 7. Reflection and Conclusion Round Policy Deliberation: At the end of the reflection round unanimous agreement is seen on all policies proposed during the initial round. And in the conclusion round agents agree to the proposed policies in the reflection round.

C. Effect of more turn in deliberation on consensus

Figure 8 shows how increasing reflection turns affects consensus time. For fixed-answer problems (VI-B), $t \in [2, 3]$ ensures unanimous convergence for most cases. In policy-based deliberations figure 7, $t = 1$ yields three core policies, whereas increasing $t = 5$ expands to 6-8 policies, enhancing diversity but delaying consensus as agreement on more policies is needed.

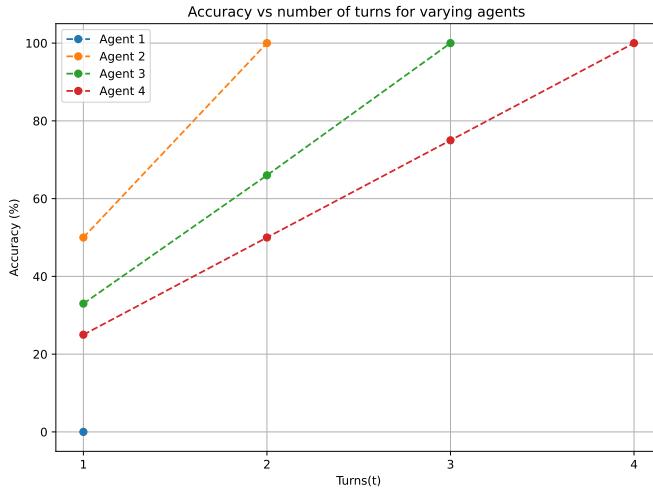


Fig. 8. Time taken for convergence while keeping the number of agents to 3 and varying the turns in the reflection round.

D. Effect of number of agents and consensus

More the number of agent more divergent opinions are presented, but also more turns are necessary for convergence. We varied the number of agents and observed the number of turns necessary to get convergence for the Wason card problem [33]. The accuracy shown in figure 9 is the percentage of agents that have generated the correct result at the start of the turn.

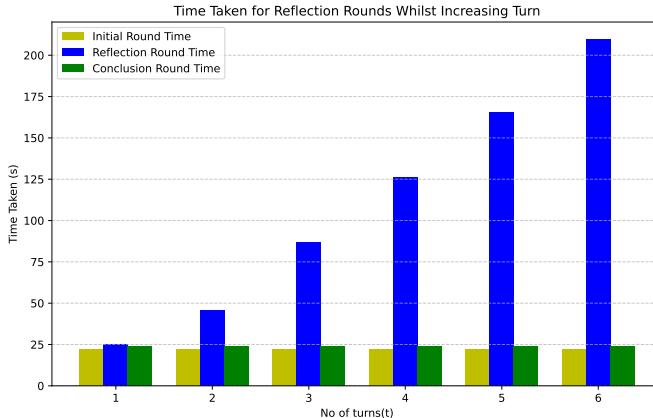


Fig. 9. Accuracy obtained at the end of each turn for varying agents.

E. Correctness of Deliberation Protocol

The protocol achieves a 77.4% success rate on arithmetic problems, which does not establish a new benchmark or

surpass previous works [9], [15], [16]. However, the primary focus is not on improving accuracy.

The model's performance was evaluated on a representative subset of the dataset [32]. Future work will extend this evaluation to the full dataset to ensure comprehensive validation.

F. Block Size

The theoretical max size of our block is 2^{32} bytes. However, such a massive size is not required and it is limited to 100 kb.

TABLE II
BLOCK SIZE, AND BLOCK GENERATION TIME WITH VARYING NUMBER OF TRANSACTIONS AND UTTERANCES,

# Transaction	# Utterances	Size Block (bytes)	Block Generation Time (ms)
1	0	273	7
2	0	411	9
3	0	549	9
0	1	247	8
0	5	695	8
0	10	1255	7
0	20	2375	7
0	30	3495	7

Table II presents the impact of varying the number of transactions and utterances on block size and generation time. Since all transactions and utterances are pre-existing in the mempool before block generation, the block size increases linearly with the number of included elements. However, block generation time remains nearly constant, as the hashing operation for the block header is largely unaffected by the mempool size.

G. Prompt ratio

As stated by Chan et al. [21], diverse prompts outperform a single role description, which may cause performance issues. Therefore, we use either a zero-shot (ZS) or a chain-of-thought (CoT) prompt for initial prompting as shown in algorithm 1. We observe that the ratio of agents using CoT prompts affects the number of turns needed for convergence, as shown in table III. We use 4 agents in the table below.

TABLE III
NUMBER OF TURNS TAKEN FOR UNANIMOUS CONSENSUS IN THE REFLECTION ROUND FOR DIFFERENT PROMPT RATIOS.

Prompt Ratio # CoT Prompt/ # Agent	\mathcal{T} taken in reflection round				
	D1	D2	D3	D4	D5
0	0	4	>10	1	3
0.25	>10	1	4	3	0
0.5	1	1	1	3	2
0.75	2	1	1	1	1
1	1	1	1	1	1

In the Wason card game problem [33], CoT prompts enables providing examples of similar problem-solving approaches. The agents then leverage these knowledge to solve the task at hand. As more agents receive CoT prompts, those with incorrect solutions quickly refine their responses based on improved deliberation, leading to a sharp decrease in the

number of rounds required to reach consensus as seen in the table above. Value that shows 0 is when agents converged towards incorrect result and 10 is the cutoff threshold turn for reflection round.

XIII. CONCLUSION AND FUTURE WORK

This paper proposes a novel paradigm that integrates large language models (LLMs) within a blockchain to achieve unanimous consensus through deliberation. Our literature review highlights limitations in existing consensus protocols for decision-making and identifies gaps in the emerging field of LLM-driven blockchain consensus. To address this, we present a formal representation of using multi-agents to achieve unanimous consensus in decision-making through deliberation. Performance analysis on the proof of concept implementation work has also been presented. Future work will include using better deliberation framework to support consensus with higher confidence as well exploring more better techniques that will allow the model to produce better arguments for deliberation. Moderator (leader, judge) based deliberation can also be explored to exchange the current capabilities and performance. Furthermore a more complex dataset with challenging problems can be utilized to analyze the deliberation process and assess whether consensus can be achieved.

REFERENCES

- [1] S. Nakamoto, “Bitcoin: A Peer-to-Peer Electronic Cash System.”
- [2] D. G. Wood, “POLKADOT: Vision for a heterogeneous multi chain framework.”
- [3] M. Kara *et al.*, “A novel delegated proof of work consensus protocol,” in *2021 International Conference on AI-CSP*. IEEE, 2021, pp. 1–7.
- [4] A. Pokharel *et al.*, “Deliberation leads to unanimous consensus,” in *6th International Conference on BCCA*, 2024.
- [5] M. E. Morrell, “Deliberation, democratic decision-making and internal political efficacy,” *Political Behavior*, vol. 27, pp. 49–69, 2005.
- [6] M. Alhafnawi *et al.*, “Deliberative democracy with robot swarms,” in *2022 IEEE/RSJ IROS*, 2022, pp. 7296–7303.
- [7] Y. Liu *et al.*, “A systematic literature review on blockchain governance,” *CoRR*, vol. abs/2105.05460, 2021.
- [8] A. Belesiotis *et al.*, “Agreeing on plans through iterated disputes.”
- [9] Y. Du *et al.*, “Improving factuality and reasoning in language models through multiagent debate,” 2023. [Online]. Available: <https://arxiv.org/abs/2305.14325>
- [10] P. McBurney *et al.*, “The eightfold way of deliberation dialogue,” *International Journal of Intelligent Systems*, 2006.
- [11] D. Owen and G. Smith, “Survey article: Deliberation, democracy, and the systemic turn,” *Journal of political philosophy*, 2015.
- [12] M. Hadi *et al.*, “A survey on large language models: Applications, challenges, limitations, and practical usage. techrxiv,” 2023.
- [13] C. T. Small *et al.*, “Opportunities and risks of llms for scalable deliberation with polis,” *arXiv preprint arXiv:2306.11932*, 2023.
- [14] M. Minsky, *Society Of Mind*, 1988, google-Books-ID: bLDLlflfRpdkC.
- [15] T. Liang *et al.*, “Encouraging divergent thinking in large language models through multi-agent debate,” 2024. [Online]. Available: <https://arxiv.org/abs/2305.19118>
- [16] S. Rasal, “Llm harmony: Multi-agent communication for problem solving,” 2024. [Online]. Available: <https://arxiv.org/abs/2401.01312>
- [17] J. Xu *et al.*, “A survey of blockchain consensus protocols,” *ACM Computing Surveys*, vol. 55, no. 13s, pp. 1–35, 2023.
- [18] L. Lampert *et al.*, “The byzantine generals problem.” [Online]. Available: <https://doi.org/10.1145/3335772.3335936>
- [19] J. Ahn *et al.*, “Timechara: Evaluating point-in-time character hallucination of role-playing large language models,” 2024. [Online]. Available: <https://arxiv.org/abs/2405.18027>
- [20] Y. Shao *et al.*, “Character-llm: A trainable agent for role-playing,” 2023. [Online]. Available: <https://arxiv.org/abs/2310.10158>
- [21] C.-M. Chan *et al.*, “Chateval: Towards better llm-based evaluators through multi-agent debate,” 2023. [Online]. Available: <https://arxiv.org/abs/2308.07201>
- [22] C. Hu *et al.*, “Chatdb: Augmenting llms with databases as their symbolic memory,” 2023. [Online]. Available: <https://arxiv.org/abs/2306.03901>
- [23] S. Yao *et al.*, “Tree of thoughts: Deliberate problem solving with large language models,” 2023. [Online]. Available: <https://arxiv.org/abs/2305.10601>
- [24] Z. Liu *et al.*, “Graphprompt: Unifying pre-training and downstream tasks for graph neural networks,” 2023. [Online]. Available: <https://arxiv.org/abs/2302.08043>
- [25] Y. El Faqir *et al.*, “An overview of decentralized autonomous organizations on the blockchain,” 2020.
- [26] C. Lesaege, “Kleros whitepaper,” URL: <https://kleros.io/static/whitepaper>, 2019.
- [27] T. Dursun *et al.*, “A novel framework for policy based on-chain governance of blockchain networks,” *Information Processing & Management*, vol. 58, 2021.
- [28] S. Org. (2024) Snapshot docs. Accessed on April 9, 2024. [Online]. Available: <https://docs.snapshot.org/introduction>
- [29] X. Li *et al.*, “A survey on LLM-based multi-agent systems: workflow, infrastructure, and challenges.” *Vicinagearth*, p. 9. [Online]. Available: <https://doi.org/10.1007/s44336-024-00009-2>
- [30] J. Wei *et al.*, “Chain-of-thought prompting elicits reasoning in large language models,” in *Advances in Neural Information Processing Systems*, vol. 35. Curran Associates, Inc., 2022, pp. 24 824–24 837.
- [31] Y. Fu *et al.*, “Improving language model negotiation with self-play and in-context learning from ai feedback,” 2023.
- [32] S. Org. (2024) Snapshot docs. [Online]. Available: <https://huggingface.co/datasets/georgiyozhegov/g.arithmetic?row=19>
- [33] G. Karadzhov *et al.*, “Delidata: A dataset for deliberation in multi-party problem solving,” *Proceedings of the ACM on Human-Computer Interaction*, vol. 7, pp. 1–25, 2023.
- [34] Z. Wang *et al.*, “Unleashing the emergent cognitive synergy in large language models: A task-solving agent through multi-persona self-collaboration,” 2024. [Online]. Available: <https://arxiv.org/abs/2307.05300>
- [35] J. C.-Y. Chen *et al.*, “Reconcile: Round-table conference improves reasoning via consensus among diverse llms,” 2024. [Online]. Available: <https://arxiv.org/abs/2309.13007>
- [36] T. Schick *et al.*, “Peer: A collaborative language model,” 2022. [Online]. Available: <https://arxiv.org/abs/2208.11663>
- [37] D. Schwartz *et al.*, “The Ripple Protocol Consensus Algorithm.”
- [38] J. Chen and S. Micali, “Algorand,” 2017. [Online]. Available: <https://arxiv.org/abs/1607.01341>
- [39] L. Huang *et al.*, “A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions,” 2025.
- [40] M. Mitchell and D. C. Krakauer, “The debate over understanding in ai’s large language models,” *Proceedings of the National Academy of Sciences*, 2023.
- [41] R. Stureborg *et al.*, “Large language models are inconsistent and biased evaluators,” 2024. [Online]. Available: <https://arxiv.org/abs/2405.01724>
- [42] Y. Yao *et al.*, “A survey on large language model (llm) security and privacy: The good, the bad, and the ugly,” *High-Confidence Computing*, 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S266729522400014X>
- [43] Y.-J. Sun *et al.*, “Confidence estimation for llm-based dialogue state tracking,” in *2024 IEEE SLT*.
- [44] A. Salau *et al.*, “Making blockchain validators honest,” in *2022 BCCA*, 2022.
- [45] H. Bansal *et al.*, “Smaller, weaker, yet better: Training llm reasoners via compute-optimal sampling,” 2024. [Online]. Available: <https://arxiv.org/abs/2408.16737>
- [46] R. Zhang *et al.*, “LLaMA-adapter: Efficient fine-tuning of large language models with zero-initialized attention,” in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: <https://openreview.net/forum?id=d4UiXAHN2W>
- [47] J. Benet, “IPFS - content addressed, versioned, P2P file system,” *CoRR*, vol. abs/1407.3561, 2014.
- [48] T. Nakai, A. Sakurai, S. Hironaka, and K. Shudo, “The blockchain trilemma described by a formula,” in *2023 IEEE International Conference on Blockchain (Blockchain)*, 2023, pp. 41–46.

- [49] J. Kirchenbauer *et al.*, “A watermark for large language models,” 2024. [Online]. Available: <https://arxiv.org/abs/2301.10226>
- [50] E. Mitchell *et al.*, “Detectgpt: Zero-shot machine-generated text detection using probability curvature,” 2023.
- [51] P. Otte *et al.*, “Trustchain: A sybil-resistant scalable blockchain,” *Future Generation Computer Systems*, pp. 770–780, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0167739X17318988>
- [52] “Whitepaper.” [Online]. Available: <https://www.nimiq.com/litepaper/>
- [53] P. Berrang *et al.*, “Albatross – an optimistic consensus algorithm,” in *2019 CVCBT*, 2019, pp. 39–42.