DEL-ToM: Inference-Time Scaling for Theory-of-Mind Reasoning via Dynamic Epistemic Logic

Yuheng Wu [1] **Jianwen Xie** [2] **Denghui Zhang** [3] **Zhaozhuo Xu** [3]

Abstract

Theory-of-Mind (ToM) tasks pose a unique challenge for small language models (SLMs) with limited scale, which often lack the capacity to perform deep social reasoning. In this work, we propose DEL-ToM, a framework that improves ToM reasoning through inference-time scaling rather than architectural changes. Our approach decomposes ToM tasks into a sequence of belief updates grounded in Dynamic Epistemic Logic (DEL), enabling structured and transparent reasoning. We train a verifier, called the Process Belief Model (PBM), to score each belief update step using labels generated automatically via a DEL simulator. During inference, candidate belief traces generated by a language model are evaluated by the PBM, and the highest-scoring trace is selected. This allows SLMs to emulate more deliberate reasoning by allocating additional compute at test time. Experiments across multiple model scales and benchmarks show that DEL-ToM consistently improves performance, demonstrating that verifiable belief supervision can significantly enhance ToM abilities of SLMs without retraining.

1. Introduction

The ability to attribute beliefs, desires, and intentions to others, known as Theory-of-Mind (ToM) (Apperly & Butterfill, 2009; Premack & Woodruff, 1978; Rabinowitz et al., 2018), is a fundamental component of social intelligence (Baron-Cohen, 1991). Recent studies show that large language models (LLMs) (Brown et al., 2020) often exhibit scaling in ToM abilities (Strachan et al., 2024; Street et al., 2024; Amirizaniani et al., 2024), with larger models performing better than smaller ones in ToM evaluation tasks. How to equip small language models (SLMs) (Abdin et al., 2024; Yang et al., 2025a; Grattafiori et al., 2024) with social reasoning abilities comparable to those of LLMs remains an open question. This is particularly important for enabling SLMs in resource-limited settings, where we want agents powered by them to understand users' intentions and act in ways aligned with human expectations.

A common approach to enhance ToM reasoning of SLMs is step-by-step prompting (Hou et al., 2024; Lin et al., 2024). However, this approach treats the reasoning process as a black box. Despite generating intermediate steps, we have no way to assess whether the reasoning trace itself is correct. With only final-answer supervision, the process is unverifiable, and thus unjustified. Therefore, this paper focuses on the question: *How can we enable SLMs to perform justified Theory-of-Mind reasoning?*

To achieve justified ToM reasoning, following process reliabilism (Goldman, 1979), we require the reasoning process to be reliable. This means we must transparently generate each intermediate belief update, and use an external method to evaluate the reliability of each update.

In this paper, we introduce the Dynamic Epistemic Logic (DEL) framework to generate intermediate belief states. DEL (Van Benthem, 2001; Plaza, 2007; Van Ditmarsch et al., 2007; Aucher & Schwarzentruber, 2013) is a formal logic system that represents agents' belief states with epistemic models, actions with event models, and updates beliefs via product updates. This provides a transparent process for generating belief traces. We then evaluate the quality of each belief trace using a Process Belief Model (PBM). By generating multiple candidate traces and scoring them with PBM, we select the one with higher score. This constitutes inference-time scaling: spending more computation during inference to obtain more justified reasoning traces.

To train the PBM, we first generate ToM-related questions and use DEL to produce belief process labels. We then use an advanced LLM, typically GPT-4o-mini (Hurst et al., 2024) to answer these questions. Finally, we construct the PBM training dataset by pairing the DEL-generated labels

[1]Stanford University, USA [2]Lambda Inc., USA [3]Stevens Institude of Technology, USA. Correspondence to: Denghui Zhang <dzhang42@stevens.edu>, Zhaozhuo Xu <zxu79@stevens.edu>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

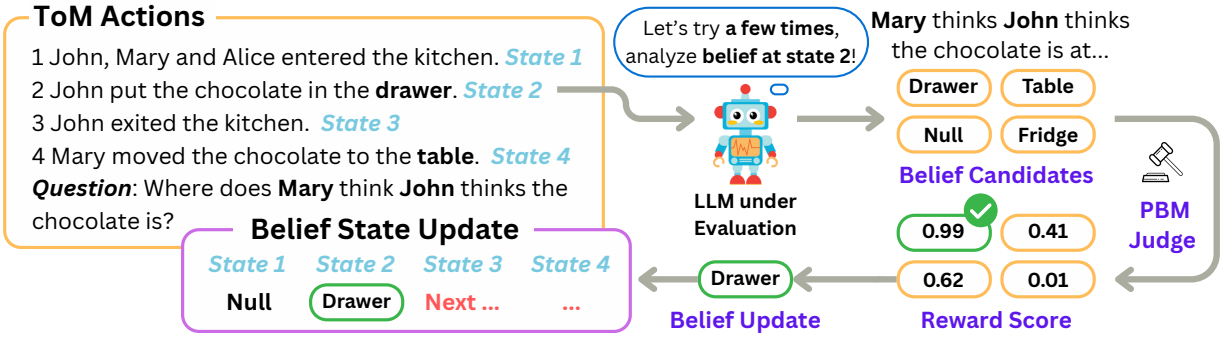


Figure 1. Overview of the DEL-ToM framework.

with GPT-generated traces. Unlike other process-level reward modeling datasets, which rely on human annotation or LLM assistance (Wang et al., 2023), our labels are directly derived from a formal DEL system, ensuring their correctness and consistency.

In conclusion, we approach ToM reasoning from the perspective of process reliability. By training a PBM via DEL, we evaluate the quality of each intermediate reasoning step and use search-based methods to select the most reliable trace. This enables inference-time scaling and results in a more robust ToM reasoning process.

2. Background and Motivation

For a full overview of related work, see Appendix A. Here, we briefly introduce the background and motivation.

Theory of Mind in LLMs. To assess the ToM capabilities of LLMs, researchers commonly adopt tasks that test whether a model can reason about others' beliefs. Among these, false belief tasks are the most widely used. Figure 1 illustrates a typical instance of this task setup. The story consists of four sentences, each describing an *action* that incrementally changes the characters' beliefs. Between every two action, the characters are assumed to hold a belief configuration, which we refer to as a belief *state*. ToM reasoning aims to infer these belief states.

In this example, after *Action 1*, John, Mary, and Alice are all present in the kitchen, but the chocolate has not been introduced yet, so no beliefs are established. After *Action 2*, John places the chocolate in the drawer, and everyone present (including Mary) observes this action. Hence, Mary believes that John believes the chocolate is in the drawer.

Following *Action 3*, John exits the kitchen. Then, in *Action 4*, Mary moves the chocolate to the table, an action that John is unaware of. As a result, Mary's mental model of the world (*her world*) now differs from John's outdated view (*his world*). Therefore, Mary thinks John still believes the chocolate is in the drawer.

With a foundational understanding of ToM and its belief-state dynamics, we now observe that this reasoning process naturally aligns with the framework of DEL. In ToM tasks, a sequence of *actions* leads to a sequence of belief *states*, which evolve as characters gain or lose access to information. We represent each belief state using an *epistemic model*, and each action as an *event model*. DEL provides a core operation called the *product update*, which allows us to compute the next epistemic state by combining the current state with an action. For full formal definitions and examples, see Appendix B.

Our Objective: Inference-Time Scaling of ToM Capacities for SLMs. Our objective is to enhance the ToM capabilities of SLMs without increasing their parameter count. To strike a balance between performance and computational efficiency, we adopt a reward model-based (Beeching et al., 2024) inference-time scaling approach that allocates additional compute at inference to enable more structured and accurate reasoning. This allows SLMs to achieve competitive performance on socially grounded tasks while remaining efficient for deployment.

3. DEL-ToM for Inference Time Scaling

In this section, we first describe how we construct and train the PBM to evaluate belief traces. We then introduce inference-time strategies for ranking and selecting belief traces based on their process-level rewards.

3.1. Building PBM

3.1.1. Dataset Synthesis via DEL

In this section, we describe how we construct the training dataset for the PBM using DEL. We generate 20,000 ToM examples using the scenario and question generation script from the Hi-ToM framework (He et al., 2023). For process-level labels, we design a DEL-based simulator to compute the full belief update trace, as illustrated in Figure 2. For reasoning traces, we use GPT-4o-mini (Hurst et al., 2024). We observe that GPT-4o-mini produces a balanced mix of

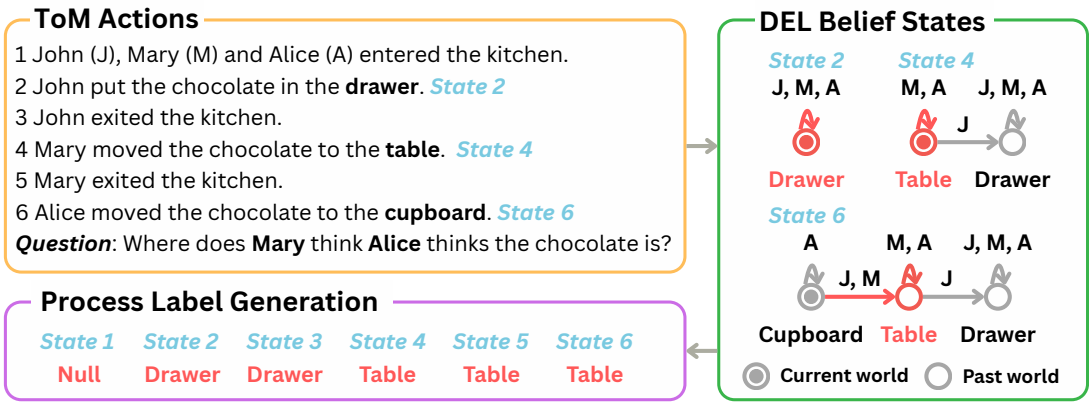


Figure 2. Training data synthesis paradigm for PBM.

correct and incorrect reasoning traces, making it suitable for reward modeling. Each training sample thus consists of a LLM-generated reasoning trace paired with a DEL-generated process label.

3.1.2. Training the PBM

PBM is a scoring function $f : \mathcal{P} \times \mathcal{S} \to \mathbb{R}^+$ that assigns a reliability score to each step s_i in a belief trace s under a given ToM problem P. We treat this as a binary classification task: each step is labeled as either *correct* or *incorrect* according to the DEL-generated belief trace. The model is trained using the following binary cross-entropy objective:

$$\mathcal{L}_{\text{PBM}} = \sum_{i=1}^{K} y_{s_i} \log f(s_i) + (1 - y_{s_i}) \log(1 - f(s_i)),$$

where K is the number of steps, y_{s_i} is the binary label, and $f(s_i)$ is the predicted score.

3.2. Inference-Time Scaling with PBM

After training the PBM, we integrate it with inference-time search methods to enhance ToM reasoning. We consider both *online* and *offline* strategies.

Online: Beam Search. In the online setting, the model constructs the belief trace step-by-step. At each action, it proposes several candidate updates, which are scored by the PBM. The top-scoring candidates are retained for the next step. We provide an example in Figure 1.

Offline: Best-of-N. In the offline setting, the model generates multiple full belief traces after reading the entire story. The PBM then evaluates each trace using step-wise scores aggregated by strategies such as *last* (score of the final step), *min* (lowest score across all steps), *avg* (average score across steps), and *prod* (product of step-wise scores).

Ranking Strategies. Based on the aggregated scores, we consider two ways to choose the final answer: (i) *Vanilla Best-of-N*, which selects the trace with the highest PBM score; and (ii) *Weighted Best-of-N*, which groups traces by final answer and sums scores across traces predicting the same answer.

For detailed algorithmic procedures and scoring rules, see Appendix C.

4. Experiments

4.1. Experimental Setup

We evaluate our method on both Qwen3 (Yang et al., 2025a) and Llama3.2 (Grattafiori et al., 2024) model families, using models ranging from 0.6B to 8B parameters. The PBM is trained on the DEL-generated dataset using Llama3.1-8B-Instruct. All inference and training are conducted on a single GH200 node.

Experiments are performed on two datasets: Hi-ToM (He et al., 2023) and the human-written benchmark from Kosinski (2023), testing generalization across different data structures. Full model details, training settings, prompt formats, and implementation specifics are provided in Appendix D.

4.2. Results and Analysis

We evaluate our method using both *offline* (Best-of-N, $N = 1024$) and *online* (beam search, $N = 256$) inference-time scaling strategies. Results in Table 1 and Table 2 confirm that PBM consistently improves ToM accuracy across models and belief orders, including low-capacity models that would otherwise struggle to reason about beliefs.

Figure 3 (a)-(b) shows performance trends as N increases under different aggregation and ranking strategies, highlighting the importance of verifier-guided selection. We find that simple methods like majority voting fail to benefit from increased sampling, whereas PBM-guided selection continues to yield improvements with more candidates.

Table 1. Offline inference-time scaling methods across belief orders in HiToM Dataset, with and without PBM. "Ori" denotes the baseline accuracy from a single sample without inference-time scaling.

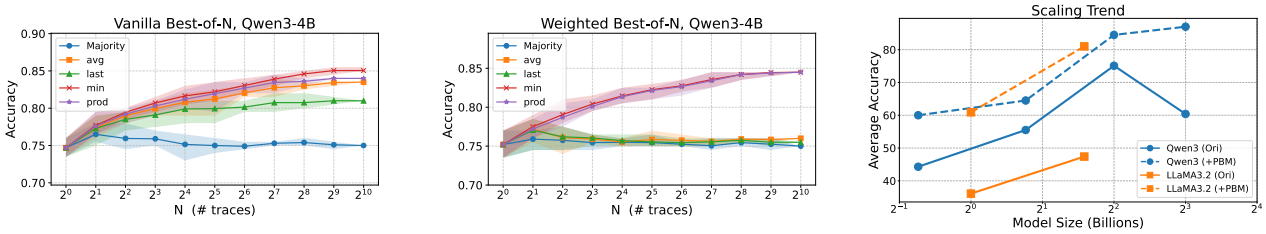
Model	0-th Order		1-th Order		2-th Order		3-th Order		4-th Order		Average	
	Ori	+PBM	Ori	+PBM	Ori	+PBM	Ori	+PBM	Ori	+PBM	Ori	+PBM
Qwen3-4B	100.0	100.0	79.8	85.0	79.3	90.0	70.2	82.5	46.0	65.0	75.1	84.5
Qwen3-1.7B	78.0	82.5	59.7	65.0	45.2	55.0	47.0	62.5	47.8	57.5	55.5	64.5
Qwen3-0.6B	69.2	80.0	52.0	72.5	35.0	47.5	31.5	52.5	34.0	47.5	44.3	60.0
Llama3.2-3B	68.2	85.0	52.0	80.0	43.2	82.5	37.0	82.5	36.8	75.0	47.4	81.0
Llama3.2-1B	41.5	46.2	40.0	53.8	28.5	61.5	41.5	84.6	29.2	58.3	36.1	60.9

Table 2. Online inference-time scaling methods across belief orders in HiToM Dataset, with and without PBM. "Ori" denotes the baseline accuracy from a single sample without inference-time scaling.

Model	0-th Order		1-th Order		2-th Order		3-th Order		4-th Order		Average	
	Ori	+PBM	Ori	+PBM	Ori	+PBM	Ori	+PBM	Ori	+PBM	Ori	+PBM
Qwen3-8B	96.5	80.0	53.3	80.0	38.8	85.0	55.8	95.0	57.8	95.0	60.4	87.0
Qwen3-4B	100.0	100.0	79.8	85.0	79.3	97.5	70.2	82.5	46.0	60.0	75.1	85.0

Table 3. Offline inference-time scaling methods on the (Kosinski, 2023) dataset, evaluated across different belief types, with and without PBM. "Ori" denotes baseline accuracy from a single sample without PBM.

Model	False Belief		Informed Protagonist		No Transfer		Present Protagonist		Average	
	Ori	+PBM	Ori	+PBM	Ori	+PBM	Ori	+PBM	Ori	+PBM
Qwen3-8B	83.3	87.5	83.8	85.0	92.8	97.5	79.5	85.0	84.8	88.8
Qwen3-4B	70.2	80.0	86.2	90.0	93.2	95.0	88.0	92.5	84.4	89.4
Qwen3-1.7B	18.2	35.0	15.5	37.5	24.8	60.0	13.8	30.0	18.1	40.6
Qwen3-0.6B	14.5	12.5	23.5	30.0	25.0	35.0	21.0	32.5	21.0	27.5



(a) Vanilla Best-of-N decoding. (b) Weighted Best-of-N decoding. (c) Scaling trend across models.

Figure 3. Comparison of decoding strategies and scaling trends across models.

Figure 3 (c) further demonstrates that PBM facilitates more effective scaling with model size, helping unlock latent ToM capabilities that are not expressed through standard decoding. In particular, models that initially underperform due to misaligned reasoning patterns (e.g., Qwen3-8B) can achieve state-of-the-art accuracy when guided by PBM.

We also validate the generalization of PBM on the Kosinski benchmark (Kosinski, 2023), where it continues to yield consistent gains (Table 3), confirming that the PBM operates as a genuine verifier of belief dynamics rather than merely overfitting to synthetic training data.

These findings demonstrate that inference-time scaling via DEL, guided by the PBM, provides an effective and generalizable framework for performing justified ToM reasoning. See Appendix E for detailed results and analysis.

5. Conclusion

This work presents a new approach to Theory-of-Mind (ToM) reasoning by focusing on inference-time reliability rather than model scale. By formalizing belief updates through Dynamic Epistemic Logic (DEL) and using a verifier model trained with logic-generated labels, we enable small language models to reason in a more transparent and structured manner. Our inference-time framework improves ToM performance of small LLMs across several benchmarks. It demonstrates that social reasoning tasks can ben-

efit from compute-efficient methods that guide rather than retrain the model. This opens new possibilities for deploying socially aware AI in resource-limited settings.

References

Abdin, M., Aneja, J., Awadalla, H., Awadallah, A., Awan, A. A., Bach, N., Bahree, A., Bakhtiari, A., Bao, J., Behl, H., et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024.

Amirizaniani, M., Martin, E., Sivachenko, M., Mashhadi, A., and Shah, C. Do llms exhibit human-like reasoning? evaluating theory of mind in llms for open-ended responses. *arXiv preprint arXiv:2406.05659*, 2024.

Apperly, I. A. and Butterfill, S. A. Do humans have two systems to track beliefs and belief-like states? *Psychological review*, 116(4):953, 2009.

Aucher, G. and Schwarzentruber, F. On the complexity of dynamic epistemic logic. *arXiv preprint arXiv:1310.6406*, 2013.

Baron-Cohen, S. Precursors to a theory of mind: Understanding attention in others. *Natural theories of mind: Evolution, development and simulation of everyday mindreading*, 1(233-251):1, 1991.

Beeching, E., Tunstall, L., and Rush, S. Scaling test-time compute with open models, 2024.

Bolander, T. Seeing is believing: Formalising false-belief tasks in dynamic epistemic logic. In *European conference on social intelligence (ECSI 2014)*, pp. 87–107, 2014.

Bolander, T. and Andersen, M. B. Epistemic planning for single-and multi-agent systems. *Journal of Applied Non-Classical Logics*, 21(1):9–34, 2011.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901, 2020.

Goldman, A. I. What is justified belief? *Justification and Knowledge*, 17:1–23, 1979. doi: 10.1007/978-94-009-9493-5_1.

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A., et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Hansen, L. D. and Bolander, T. Implementing theory of mind on a robot using dynamic epistemic logic. In *Twenty-Ninth International Joint Conference on Artificial Intelligence*, pp. 1615–1621. International Joint Conference on Artificial Intelligence Organization, 2020.

He, Y., Wu, Y., Jia, Y., Mihalcea, R., Chen, Y., and Deng, N. Hi-tom: A benchmark for evaluating higher-order theory of mind reasoning in large language models. *arXiv preprint arXiv:2310.16755*, 2023.

Hou, G., Zhang, W., Shen, Y., Wu, L., and Lu, W. Timetom: Temporal space is the key to unlocking the door of large language models' theory-of-mind. *arXiv preprint arXiv:2407.01455*, 2024.

Hurst, A., Lerer, A., Goucher, A. P., Perelman, A., Ramesh, A., Clark, A., Ostrow, A., Welihinda, A., Hayes, A., Radford, A., et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Knuth, D. E. Backus normal form vs. backus naur form. *Communications of the ACM*, 7(12):735–736, 1964.

Kosinski, M. Theory of mind may have spontaneously emerged in large language models. *arXiv preprint arXiv:2302.02083*, 4:169, 2023.

Kwon, W., Li, Z., Zhuang, S., Sheng, Y., Zheng, L., Yu, C. H., Gonzalez, J. E., Zhang, H., and Stoica, I. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

Lin, Z., Chan, C., Song, Y., and Liu, X. Constrained reasoning chains for enhancing theory-of-mind in large language models. In *Pacific Rim International Conference on Artificial Intelligence*, pp. 354–360. Springer, 2024.

Misaki, K., Inoue, Y., Imajuku, Y., Kuroki, S., Nakamura, T., and Akiba, T. Wider or deeper? scaling llm inference-time compute with adaptive branching tree search. *arXiv preprint arXiv:2503.04412*, 2025.

Muennighoff, N., Yang, Z., Shi, W., Li, X. L., Fei-Fei, L., Hajishirzi, H., Zettlemoyer, L., Liang, P., Candès, E., and Hashimoto, T. s1: Simple test-time scaling. *arXiv preprint arXiv:2501.19393*, 2025.

Plaza, J. Logics of public communications. *Synthese*, 158: 165–179, 2007.

Premack, D. and Woodruff, G. Does the chimpanzee have a theory of mind? *Behavioral and brain sciences*, 1(4): 515–526, 1978.

Rabinowitz, N., Perbet, F., Song, F., Zhang, C., Eslami, S. A., and Botvinick, M. Machine theory of mind. In

International conference on machine learning, pp. 4218–4227. PMLR, 2018.

Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., Zhang, H., Zhang, M., Li, Y., Wu, Y., et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Snell, C., Lee, J., Xu, K., and Kumar, A. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024.

Strachan, J. W., Albergo, D., Borghini, G., Pansardi, O., Scaliti, E., Gupta, S., Saxena, K., Rufo, A., Panzeri, S., Manzi, G., et al. Testing theory of mind in large language models and humans. *Nature Human Behaviour*, 8(7): 1285–1295, 2024.

Street, W., Siy, J. O., Keeling, G., Baranes, A., Barnett, B., McKibben, M., Kanyere, T., Lentz, A., Dunbar, R. I., et al. Llms achieve adult human performance on higher-order theory of mind tasks. *arXiv preprint arXiv:2405.18870*, 2024.

Van Benthem, J. Games in dynamic-epistemic logic. *Bulletin of Economic Research*, 53(4):219–248, 2001.

Van Ditmarsch, H., van Der Hoek, W., and Kooi, B. *Dynamic epistemic logic*, volume 337. Springer Science & Business Media, 2007.

Wang, P., Li, L., Shao, Z., Xu, R., Dai, D., Li, Y., Chen, D., Wu, Y., and Sui, Z. Math-shepherd: Verify and reinforce llms step-by-step without human annotations. *arXiv preprint arXiv:2312.08935*, 2023.

Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025a.

Yang, W., Ma, S., Lin, Y., and Wei, F. Towards thinking-optimal scaling of test-time compute for llm reasoning. *arXiv preprint arXiv:2502.18080*, 2025b.

Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T., Cao, Y., and Narasimhan, K. Tree of thoughts: Deliberate problem solving with large language models. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S. (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 11809–11822. Curran Associates, Inc., 2023.

A. Related Works

Dynamic Epistemic Logic and Its Connections to Theory-of-Mind. The intersection of DEL and ToM has emerged as a promising framework for formalizing belief-based social reasoning. DEL offers a principled approach to representing and updating agents' mental states through formal mechanisms such as product updates over epistemic and event models (Van Ditmarsch et al., 2007). This directly aligns with the core of ToM, which centers on inferring and reasoning about others' beliefs. Earlier cognitive models (Van Ditmarsch et al., 2007; Bolander & Andersen, 2011) employed DEL to simulate belief change in multi-agent settings. More recent computational work extends this line by incorporating DEL into neural architectures, using logic-based simulators to provide symbolic supervision for belief updates (Bolander, 2014; Rabinowitz et al., 2018). Building on this foundation, our work leverages DEL not only as a modeling formalism but also as a scaffold for inference-time supervision, enabling compositional and verifiable reasoning in ToM tasks (Hansen & Bolander, 2020).

Inference-Time Scaling of LLMs. Recent research has investigated inference-time scaling as a compute-efficient alternative to architectural scaling for enhancing the reasoning capabilities of LLMs (Beeching et al., 2024; Muennighoff et al., 2025). Instead of increasing model size, this approach leverages additional computation during inference to simulate deeper cognitive reasoning. Techniques such as Best-of-N and beam search exemplify this paradigm by enabling small LLMs to explore multiple reasoning paths and select the most plausible outcome, effectively mimicking the deliberation depth of larger models without incurring additional training costs (Snell et al., 2024; Misaki et al., 2025). While promising, current applications of inference-time scaling are largely confined to mathematical reasoning (Wang et al., 2023; Yang et al., 2025b; Yao et al., 2023). This work aims to broaden its applicability by targeting ToM as a new frontier for inference-time scaling.

B. Formulation: ToM Reasoning as DEL

We begin by introducing the formal language and semantics used to represent ToM reasoning within the framework of DEL. Let $\mathcal{P}$ be a countable set of atomic propositions, representing basic facts about the world, and let $\mathcal{A}$ be a non-empty finite set of agents, corresponding to the characters involved in the story. The epistemic language $\mathcal{L}(\mathcal{P}, \mathcal{A})$ is defined by each component $\varphi(p) \in \mathcal{L}(\mathcal{P}, \mathcal{A})$ follows the Backus-Naur Form (Knuth, 1964):

$$\varphi(p) ::= p \mid \neg p \mid p \wedge p \mid B_i(p)$$

where $p \in \mathcal{P}$ and $i \in \mathcal{A}$. The formula $B_i(p)$ is interpreted as "agent i believes p", and can also be written as $B_i p$. For example, "John believes the chocolate is in the drawer" can be written as $B_{\text{John}}(chocolate\_in\_drawer)$. Next, we define the epistemic and event models for future usage.

Definition B.1 (Epistemic Model)**.** An *epistemic model* over agent set $\mathcal{A}$ and proposition set $\mathcal{P}$ is a triple $\mathcal{M} = (W, R, V)$, where:

- W is a set of possible worlds, where each world represents a complete assignment of truth values to all atomic propositions in $\mathcal{P}$;
- $R : \mathcal{A} \to 2^{W \times W}$ assigns each agent $a \in \mathcal{A}$ an accessibility relation R_a;
- $V : \mathcal{P} \to 2^W$ maps each atomic proposition $p \in \mathcal{P}$ to the set of worlds where p is true.

A *state* is a pointed epistemic model $(\mathcal{M}, w)$ where $w \in W$ is the designated actual world.

We write wR_av to denote that world v is accessible from world w according to agent a: agent a considers v possible in state w.

The semantics of formulas in $\mathcal{L}(\mathcal{P}, \mathcal{A})$ is defined inductively as follows:

- $\mathcal{M}, w \models p$ iff $w \in V(p)$;
- $\mathcal{M}, w \models B_a\varphi$ iff for all $v \in W$ such that wR_av, we have $\mathcal{M}, v \models \varphi$.

Definition B.2 (Event Model)**.** An *event model* is a tuple $\varepsilon = (E, Q, \mathsf{pre}, \mathsf{post})$, where:

- E is a finite, non-empty set of events;
- $Q : \mathcal{A} \to 2^{E \times E}$ assigns to each agent $a \in \mathcal{A}$ a binary relation Q_a over events;
- $\mathsf{pre} : E \to \mathcal{L}(P, \mathcal{A})$ assigns a precondition formula to each event, specifying when it is executable;
- $\mathsf{post} : E \to \mathcal{L}(P, \mathcal{A})$ assigns a postcondition formula to each event, describing how the world changes.

We refer to a pointed event model (E, e) as an *action*, where $e \in E$ is the actual event that occurs.

Definition B.3 (Product Update)**.** Given a state $(\mathcal{M}, w)$ and an action (ε, e), suppose that the precondition is satisfied, i.e., $\mathcal{M}, w \models \mathsf{pre}(e)$. Then the *product update* results in a new state $(\mathcal{M}', (w, e))$, where the updated epistemic model $\mathcal{M}' = (W', R', V')$ is defined as follows:

- $W' = \{(w', e') \in W \times E \mid \mathcal{M}, w' \models \mathsf{pre}(e')\}$;
- For each agent $a \in \mathcal{A}$, $R'_a = \{((w', e'), (v', f')) \in W' \times W' \mid w' R_a v' \text{ and } e' Q_a f'\}$;
- For each atomic proposition $p \in \mathcal{P}$, $(w', e') \in V'(p)$ iff $\mathsf{post}(e') \models p$ or $(\mathcal{M}, w' \models p \text{ and } \mathsf{post}(e') \not\models \neg p)$.

C. Details of Inference-Time Scaling Methods

After training the PBM, we integrate it with various inference-time searching methods to improve ToM reasoning through inference-time scaling. We consider both online and offline strategies.

Online: Beam Search. Beam search is a structured decoding method that maintains multiple partial belief traces during generation. At each reasoning step, it explores several alternative updates and selects the most promising ones based on the PBM scores. Formally, the procedure works as follows:

- Initialize a set of k beams by sampling k candidate first steps from the model.
- At each step, for every current beam, sample b next-step candidates, forming $k \times b$ new paths.
- Score each extended path using the PBM. We use the PBM score of the last step to rank partial paths.
- Retain the top k paths with the highest scores and repeat the process until reaching an end-of-sequence token or a maximum depth.

This approach jointly optimizes the generation and evaluation of belief traces, allowing the model to explore plausible alternatives and commit to higher-reward reasoning trajectories.

Offline: Best-of-N. In the offline setting, we sample N complete belief traces independently, and then evaluate them using the PBM. We experiment with different aggregation rules for scoring each trace based on step-wise PBM scores:

- **Last**: Use the PBM score of the final step.
- **Min**: Use the lowest score across all steps.
- **Avg**: Use the average score across the trace.
- **Prod**: Multiply the scores of all steps.

Based on the aggregated scores, we consider two ranking strategies for selecting the final answer:

(i) Vanilla Best-of-N. Select the trace with the highest PBM score and extract its final answer. This method chooses the most confident individual trace, but does not account for answer consistency across traces.

(ii) Weighted Best-of-N. Group traces by their final answers, then aggregate PBM scores across traces that predict the same answer. The answer with the highest total score is selected:

$$\hat{y} = \arg\max_{y \in \mathcal{Y}} \sum_{i=1}^{N} \mathbb{1}(y_i = y) \cdot \mathrm{PBM}(p, s_i)$$

Here, s_i is the i-th belief trace, y_i its final answer, and $\mathrm{PBM}(p, s_i)$ the trace-level score. This strategy emphasizes both answer quality and consistency across samples.

D. Detailed Experimental Setup

Platform. All experiments are conducted on a single NVIDIA GH200 GPU node. We use the vLLM (Kwon et al., 2023) framework for efficient batched inference and large-scale decoding.

PBM Training. We fine-tune a PBM model based on Llama3.1-8B-Instruct (Grattafiori et al., 2024). The model is trained for 1 epoch using our DEL-generated dataset. All training is performed on the same GH200 machine.

Test Models. We evaluate our methods on both the Qwen3 series (0.6B, 1.7B, 4B) (Yang et al., 2025a) and Llama3.2 series (1B, 3B) (Grattafiori et al., 2024). All models are evaluated using their default generation settings, with no change to temperature, top-k, or nucleus sampling parameters.

Datasets. We conduct evaluations on two datasets: Hi-ToM (He et al., 2023) and (Kosinski, 2023). The Hi-ToM dataset is constructed using our DEL-based generator, while (Kosinski, 2023) is a human-written ToM benchmark. These two datasets differ in structure, providing a testbed for evaluating the generalization of the PBM.

Metrics and Prompt Format. We report final answer accuracy as the main evaluation metric. All models are evaluated using a consistent prompting format across datasets, as detailed in Appendix F.

E. Detailed Results and Analysis

E.1. Offline Methods Result

We set N up to 1024 and apply the weighted Best-of-N strategy, selecting the best aggregation rule (among avg, last, min, and prod) as the final answer for each instance.

Main Results. As shown in Table 1, incorporating PBM leads to a significant improvement in ToM reasoning performance across all models. For example, Llama3.2-3B exhibits a substantial gain of 33.6 points in average accuracy, while Qwen3-4B improves by 9.4 points. Across all belief orders (from 0-th to 4-th), PBM consistently yields higher accuracy, confirming the robustness and generalizability of our inference-time scaling method.

Scaling N for ToM Reasoning. As shown in Figure 3 (a)-(b), increasing the number of sampled belief traces N consistently improves ToM reasoning performance. Among the aggregation strategies, min and prod exhibit stable and similar performance across both *vanilla* and *weighted* ranking schemes. In contrast, avg and last tend to degrade in performance under weighted aggregation, likely due to their sensitivity to low-quality or inconsistent samples. We recommend using min or prod as robust aggregation rules for inference-time ToM scaling.

Majority Voting Doesn't Work for ToM Reasoning. Interestingly, we observe that scaling N with majority voting does not lead to improved accuracy on ToM tasks. This contrasts with math reasoning tasks, where majority voting often benefits from larger N by amplifying consistent correct answers. The discrepancy highlights a key distinction: ToM reasoning is a dynamic, social process that cannot be reduced to static answer aggregation. Therefore, our trained PBM is crucial: without such a verifier, inference-time scaling for ToM would be ineffective. It is precisely the PBM that enables us to assess whether each intermediate belief state in the reasoning process is likely to be justified.

E.2. Online Methods Result

Online Setting. For online inference-time scaling, we conduct beam search experiments on Qwen3-4B and scale up to Qwen3-8B. We do not include smaller models because their instruction-following ability is insufficient for producing valid intermediate reasoning steps. The number of beams N is varied from 4 to 256.

Main Results. Again, as shown in Table 2, we observe that incorporating PBM leads to substantial improvements in ToM reasoning. For Qwen3-4B, the PBM-enhanced accuracy reaches 85.0, which is comparable to the best results in the offline setting. Interestingly, the original Qwen3-8B model underperforms Qwen3-4B, suggesting that baseline ToM ability does not necessarily scale with model size. However, with PBM guidance, Qwen3-8B achieves the highest accuracy of 87.0, demonstrating the effectiveness of inference-time scaling even for larger LLMs.

Online Methods or Offline Methods? Which strategy should we prefer for ToM reasoning—online or offline? Our experiments suggest that both approaches yield comparable accuracy. For instance, Qwen3-8B achieves similar performance under both settings. However, online methods are significantly harder to evaluate reliably. This is because many smaller or less instruction-aligned models struggle to follow step-by-step prompting in an online rollout, failing to produce valid intermediate states and making PBM evaluation infeasible.

In contrast, offline methods allow the model to generate a full belief trace in one shot, which is generally easier even for weaker models. Even when shortcuts or hallucinations appear mid-trace, the PBM can still function effectively. Furthermore, when paired with high-throughput generation backends such as vLLM, offline methods can generate large numbers of candidate traces efficiently. Overall, we recommend using offline inference-time methods for ToM reasoning.

E.3. Results on Other Benchmarks

Our PBM is trained on data generated under the Hi-ToM framework. A natural question arises: *Can the trained PBM generalize to ToM reasoning tasks from a different distribution?*

To evaluate this, we test our method on the dataset proposed by (Kosinski, 2023), which features hand-written scenarios involving false beliefs and a variety of true belief controls. We conduct experiments using both the Llama 3.1 and Qwen3 model series. In all experiments, we follow the same inference-time scaling and PBM-based trace selection procedure as in the Hi-ToM evaluations.

Main Results. As shown in Table 3, We observe that the PBM also generalizes well to out-of-domain data. Across all tested models, accuracy consistently improves after applying inference-time scaling guided by the PBM. This confirms that our PBM functions as a genuine verifier of whether a ToM reasoning process is justified, rather than merely fitting surface patterns in the training distribution. The improvements on the (Kosinski, 2023) benchmark demonstrate its ability to evaluate belief traces beyond synthetic scenarios, highlighting the robustness and transferability of our approach.

E.4. Discussion

Scaling with Model Size. Figure 3 (c) illustrates the impact of model scaling on ToM accuracy across different model sizes. We observe that PBM consistently improves performance and facilitates more effective scaling. For Llama 3.2, the accuracy curve becomes steeper when equipped with PBM, indicating that larger models benefit more and generalize better under our inference-time intervention. Interestingly, Qwen3 exhibits a failure in scaling at 8B under the vanilla setting—performing worse than its 4B counterpart. However, once PBM is applied, Qwen3-8B achieves the highest accuracy among all its variants. This suggests that inference-time scaling via PBM not only improves absolute performance but may also enable the emergence of higher-order reasoning capabilities that are otherwise latent in the base model.

Comparison with GRPO-based Methods. Recent work has proposed using ToM supervision to fine-tune smaller models via GRPO (Shao et al., 2024), in order to enhance their ToM capabilities. However, GRPO-based training requires substantial computational resources and is notoriously difficult to optimize. In contrast, our PBM module is lightweight and efficient: it can be trained in under three hours on a single GH200 GPU. Moreover, GRPO must be re-trained for each target model individually, whereas our PBM is trained once and can be applied across multiple models without retraining. Notably, prior work also reports that GRPO-trained models, while improving ToM reasoning, may suffer performance degradation on other benchmarks such as GSM8K. Our inference-time scaling method avoids this drawback entirely, as it does not modify the parameters of the underlying model. Overall, PBM offers a practical, generalizable, and non-invasive alternative for enhancing ToM reasoning in SLMs.

F. Prompt Templates

We present the prompt templates in the folllowing textbox.

One-Shot Prompt

```
Here is a story that unfolds in chronological order.

You will be asked a question about the story, which may involve either:
(1) Locating an object, or
(2) Inferring an agent's mental state (e.g., what A thinks B thinks C thinks).

To solve it, think step-by-step. At each step, repeat the current line from the
    story, then explain its effect on beliefs. Use [Null] if someone does not yet
    have knowledge. If a belief chain cannot be formed (e.g., some agent exited too
    early), freeze belief at the last available step.

<Note>
{note}

In public or private communication:
- The speaker believes the listener will believe the claim.
```

```
- If the listener exited the room earlier than the speaker, they will believe it.

If the question is zero-order (e.g., "Where is X really?"), then in each step, only
    track the actual location of the object (e.g., "X is in [Y]"). You do not need to
     track nested beliefs.

Here is an example:
<Story>
1 Amelia, Chloe, Liam, Owen and Benjamin entered the TV_room.
2 The celery is in the red_envelope.
3 Amelia made no movements and stayed in the TV_room for 1 minute.
4 Chloe lost his watch.
5 Amelia exited the TV_room.
6 Chloe moved the celery to the green_bucket.
7 Chloe exited the TV_room.
8 Liam moved the celery to the red_bathtub.
9 Liam exited the TV_room.
10 Owen made no movements and stayed in the TV_room for 1 minute.
11 Owen exited the TV_room.
12 Benjamin made no movements and stayed in the TV_room for 1 minute.
13 Benjamin exited the TV_room.
14 Amelia, Chloe, Liam, Owen and Benjamin entered the waiting_room.
15 Liam publicly claimed that celery is in the white_bathtub now.
16 Benjamin privately told Liam that the celery is in the blue_drawer now.

<Question>
Where does Owen think Liam thinks Chloe thinks the celery is?

<Trace>
## Step 1 ##
Amelia, Chloe, Liam, Owen and Benjamin entered the TV_room.
Everyone is present, but the celery's location is still unknown.
Owen thinks Liam thinks Chloe thinks the celery is in [Null]

## Step 2 ##
The celery is in the red_envelope.
Everyone observes this.
Owen thinks Liam thinks Chloe thinks the celery is in [red_envelope]

## Step 3 ##
Amelia made no movements and stayed in the TV_room for 1 minute.
No effect.
Owen thinks Liam thinks Chloe thinks the celery is in [red_envelope]

## Step 4 ##
Chloe lost his watch.
Irrelevant.
Owen thinks Liam thinks Chloe thinks the celery is in [red_envelope]

## Step 5 ##
Amelia exited the TV_room.
Irrelevant.
Owen thinks Liam thinks Chloe thinks the celery is in [red_envelope]

## Step 6 ##
Chloe moved the celery to the green_bucket.
Only Chloe, Liam, Owen, Benjamin are present. They all see this move.
Owen thinks Liam thinks Chloe thinks the celery is in [green_bucket]

## Step 7 ##
Chloe exited the TV_room.
Chloe's belief frozen; still [green_bucket]
Owen thinks Liam thinks Chloe thinks the celery is in [green_bucket]
```

```
## Step 8 ##
Liam moved the celery to the red_bathtub.
Only Liam, Owen, Benjamin present. They observe the move. Chloe not present, so her
    belief unchanged.
Owen thinks Liam thinks Chloe thinks the celery is in [green_bucket]

## Step 9 ##
Liam exited the TV_room.
No change.
Owen thinks Liam thinks Chloe thinks the celery is in [green_bucket]

## Step 10 ##
Owen made no movements and stayed in the TV_room for 1 minute.
Irrelevant.
Owen thinks Liam thinks Chloe thinks the celery is in [green_bucket]

## Step 11 ##
Owen exited the TV_room.
Owen's belief frozen.
Owen thinks Liam thinks Chloe thinks the celery is in [green_bucket]

## Step 12 ##
Benjamin made no movements and stayed in the TV_room for 1 minute.
Irrelevant.
Owen thinks Liam thinks Chloe thinks the celery is in [green_bucket]

## Step 13 ##
Benjamin exited the TV_room.
No change.
Owen thinks Liam thinks Chloe thinks the celery is in [green_bucket]

## Step 14 ##
Everyone entered the waiting_room.
No effect on beliefs.
Owen thinks Liam thinks Chloe thinks the celery is in [green_bucket]

## Step 15 ##
Liam publicly claimed that celery is in the white_bathtub now.
Owen hears this statement. However, public speech only affects first- and second-
    order beliefs (e.g., what Liam believes, what Owen thinks Liam believes, and what
     Liam thinks Owen believes). It does not change Owen's belief about what Liam
    thinks Chloe thinks.
Owen thinks Liam thinks Chloe thinks the celery is in [green_bucket]

## Step 16 ##
Benjamin privately told Liam that the celery is in the blue_drawer now.
Owen does not hear this, but more importantly, private communication only affects
    beliefs between the speaker and the listener. It can change what Liam believes (
    based on exit order), or what Liam thinks Benjamin believes (based on exit order)
    , or what Benjamin thinks Liam believes (always change) - but it cannot affect
    higher-order beliefs. So this does not change Owen's belief about what Liam
    thinks Chloe thinks.

Owen thinks Liam thinks Chloe thinks the celery is in [green_bucket]

Final Answer: [green_bucket]

Now it's your turn.

<Story>
{story}
```

```
<Question>
{question}

Give a step-by-step trace as in the example. Then, give the final answer in one line
    like:
Final Answer: [your choice]

<trace>
```