Research Statement

I am a researcher specializing in **neuro-symbolic systems**, **sequential decision making** and **heuristic graph search**. They are the brains of the autonomous agents that operate in the real world as well as a general purpose high-performance solver for combinatorial tasks with a sequential nature. Given the current state of the surrounding environment, the condition for task completion (goal), and the possible actions the agent can take with their associated prerequisites, the agent constructs a plan: which action to execute in what order, and which object to interact with. For example, in a logistics task, a sequence load(truck, pkg), drive(truck, loc1, loc2), unload(truck, pkg) is one plan for delivering a package pkg from loc1 to loc2 using truck. Its applications span from robotics (Chitnis et al. 2022) to SQL query optimization (Robinson, McIlraith, and Toman 2014), quantum layout synthesis (Shaik and Van De Pol 2023), and efficient `matmul` algorithm discovery (Speck et al. 2023) and more.

1 Autonomous Agents

My most general research goal is to realize an **autonomous agent** capable of making effective logical decisions in the real world without human assistance, i.e., the holy grail of artificial intelligence research. Such agents require the ability to learn from the past and reason about the future in the real world to act intelligently, adapt to new situations quickly, and improve their behavior over time. The challenge lies in integrating the two, which require significantly different considerations.

Learning requires continuous sense-making due to the raw sensorimotor input. It is also less time-sensitive because the process can run off-line in a powerful compute environment and its cost can be amortized over a long time span. These aspects made deep neural networks a go-to solution for realizing an effective learning system. In contrast, reasoning requires agility for real-time acting and strict adherence to the rules for safety and correctness. Agility and compliance require a discrete, succinct, structured representation in symbolic expressions that enable precise verification and compilation optimized for modern hardware. **The lack of safety in learning-based systems has been under continuous scrutiny in recent years as they have become increasingly common in everyday life**, including autonomous driving and chatbot systems.

The continuous, ambiguous, and time-insensitive nature of the former is a stark contrast against the discrete, precise, time-sensitive nature of the latter, making them difficult to mix, like oil and water. However, neither of the learned instincts nor the deliberated reasoning — also called system-1 and system-2 thinking — is sufficient for effective autonomy by itself. To address this, **I have been studying *neuro-symbolic* methods that combine the best of both worlds**.

One aspect often overlooked in neuro-symbolic systems is that, just as humans rely on statistical tools to make important decisions, autonomous agents must use them effectively during both learning and reasoning. Moreover, just as humans

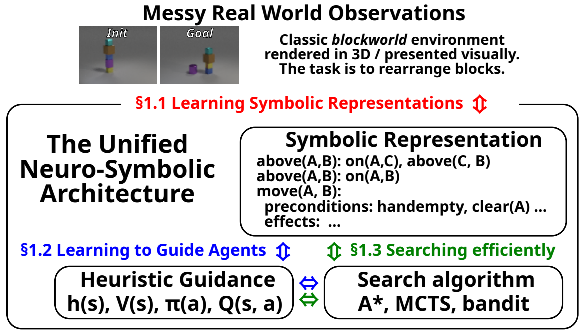

Figure 1: The unified neuro-symbolic architecture for autonomous agents in the real world.

can make misguided decisions due to an inappropriate use of statistical tools, it is difficult to select the correct statistical tools for learning and reasoning, as exemplified by the series of papers I published. An incorrect use easily leads to an ineffective system that is susceptible to the curse of dimensionality in both tasks. The challenge is exacerbated by the fact that autonomous decision making is an open-ended, broad task, and that **experts in learning and reasoning algorithms are not always experts in statistics.**

A task as abstract as my mission demands a viable approach, which is *divide and conquer*: I decompose it into three components, correct statistical flaws in existing work on each component, extend the correct statistical approach further, then finally achieve a **unified, cohesive system that enables fuller autonomy** (Fig. 1). The components include:

- Self/semi/supervised representation learning of symbolic representation from unstructured inputs.
- Offline learning of policy / value / heuristic functions for reasoning / search algorithms.
- Reasoning / search algorithms that act on the symbolic representation to make a decision, which immediately learns from the discovery made during the search. This requires search algorithms (e.g., A*, MCTS) combined with online learning / bandit algorithms.

Another important aspect that helps achieve this goal is my effort in popularizing the neuro-symbolic approaches, including my active involvement in collaborations, public outreach to various communities through tutorials and talks (e.g., RSS 2022, SIG-FPAI 2024), or contributions of a book chapter (Asai et al. 2021). I wish to expand such talks especially to those that are still relatively unfamiliar with statistical techniques (e.g. ICAPS conference series).

1.1 Learning Symbolic Representations

I published a series of papers (Asai and Fukunaga 2018; Asai and Kajino 2019; Asai 2019; Asai and Muise 2020; Asai et al. 2022b; Asai and Ayton 2022) on *Latplan* system (Fig. 2), which learns a discrete symbolic world model (representation of states, actions, and transitions) from visual inputs without supervision, then generates a visual plan between two states.

The unique aspect of Latplan is that it does so by invoking a symbolic *STRIPS classical planner* (Fikes, Hart, and

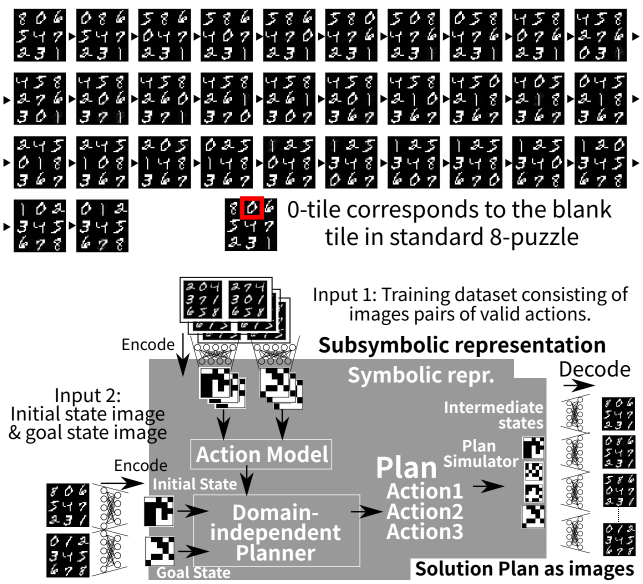

Figure 2: (Top) An example output of Latplan solving a sliding tile puzzle represented by segments of a picture. It solves the puzzle without initially knowing that this picture contains multiple tiles, or that the tile can move to an empty location next to it, or that there are maximum 4 directions. (Bottom) An overview of Latplan's pipeline.

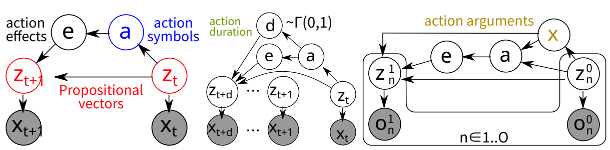

Figure 3: Graphical models in (Asai et al. 2022b) and its potential extensions.

Nilsson 1972; Helmert 2006) on the latent world model. Classical planners are optimized and scalable general solvers that enable fast solution generation but are incompatible with subsymbolic visual inputs or continuous representations. To address compatibility, its generative model (Fig. 3) extensively uses discrete latent variables reparameterized by the Gumbel-Softmax trick (Jang, Gu, and Poole 2017). The symbolic output of the planner is then passed to the generative distributions, which translate it back to a visualized plan for human comprehension. Unlike existing work on learning symbolic world models (Rodriguez et al. 2021; Ahmetoglu et al. 2022), our training loss is a variational lower bound (ELBO) of the likelihood (Asai et al. 2022b). **Such a comprehensive justification of the training loss is rarely seen in other neuro-symbolic approaches.**

1.2 Learning to Guide Agents

Every reasoning / search algorithm requires effective guidance in order to overcome the curse of dimensionality in the state space. This guidance can take several forms, includ-

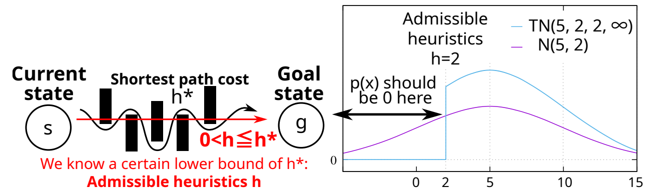

Figure 4: Traditional models for heuristic function learning ignore the lower bounds (admissible heuristics). Truncated Normal $\mathcal{TN}(\mu, \sigma, l, u)$ fixes it. (Núñez-Molina et al. 2024)

ing heuristic functions $h(s)$ that estimate the future undiscounted cumulative costs (negative rewards) from s, value $V(s)$ and action-value $Q(s, a)$ functions that estimate future discounted cumulative rewards, and policy functions $\pi(s)$, a distribution over the next possible actions. They have been tackled by separate communities: Reinforcement Learning (RL) community prefers to learn π, V, and Q, while the classical planning / heuristic search community prefers to learn h. In addition, classical planning tasks are always sparse reward problems, whereas rewards are usually dense in RL benchmarks. **The different norms across communities (reward sparsity, discounted rewards vs. non-discounted costs) have long prevented the successful application of RL to classical planning.**

In (Asai et al. 2022a), we made significant progress toward bridging this division by combining reward shaping, a common RL technique for sparse rewards, with existing symbolic heuristic functions in classical planning. The key innovation for shaping $V(s)$ with $h(s)$ is to convert $h(s)$, which returns undiscounted estimates, into one that returns discounted estimates. It not only avoids numerical errors when $h(s) = \infty$, but also matches the shaped value function with the optimal value function $V^*(s)$ when $h(s)$ equals the optimal cost $h^*(s)$.

My second work (Núñez-Molina et al. 2024) addresses the modeling flaw in existing supervised heuristic learning. Loss functions in existing work, such as Mean Squared Error or Mean Absolute Error, do not account for the domain knowledge provided by *admissible* heuristics, a class of heuristics that never overestimate the optimal cost h^*. We fixed it by defining a loss function based on a log likelihood of a Truncated Gaussian distribution, the maximum entropy distribution with a presence of lower bounds.

1.3 Searching Efficiently

Good guidance is necessary but not sufficient by itself because predictions are never perfect. The final piece of the puzzle toward effective autonomy is the search algorithm, which has been the core of artificial intelligence research throughout history (Newell, Shaw, and Simon 1959). One of my recent works (Wissow and Asai 2024; Asai 2026; Asai and Wissow 2026) made a significant breakthrough in combining distributional Monte-Carlo Tree Search (MCTS) (Kocsis and Szepesvári 2006) with traditional heuristic search and received an **outstanding paper award**. Despite the success in adversarial games, **MCTS has been outperformed by the traditional, priority-queue-based search algorithms** such

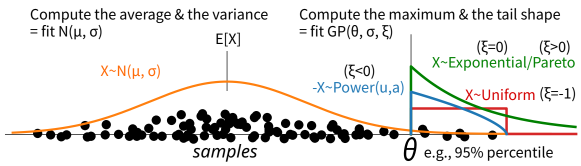

Figure 5: We fixed UCB1+MCTS with Gaussian and GP bandits (Wissow and Asai 2024; Asai and Wissow 2026).

as A* (Hart, Nilsson, and Raphael 1968) and GBFS (Bonet and Geffner 2001) in classical planning. I studied why this happens and how to fix it.

The critical flaw in existing papers that apply MCTS to classical planning is that they blindly use the popular UCB1 algorithm without fully understanding its theoretical requirements. UCB1 is a Multi-Armed Bandit (MAB) algorithm traditionally used with MCTS in games, and assumes an a priori known finite reward range $[0, c]$. While the assumption holds in games (win=1, loss=0, thus $c = 1$), it does not hold with heuristic functions that can take arbitrary non-negative values, which breaks the condition for UCB1's asymptotically optimal regret bound, leading to poor performance. Our first paper **(Outstanding Paper Award)** addressed the over-specification (fixed $[0, c]$) with Gaussian bandits whose support is $[-\infty, \infty]$. Our second paper further sharpened the model; $[-\infty, \infty]$ is an under-specification because heuristic values have an unknown but still finite support. We then proposed a bandit for the Generalized Pareto distribution based on Peaks-Over-Threshold Extreme Value Theory (Pickands III 1975; Balkema and De Haan 1974, POT EVT), a statistical framework for capturing extreme values (maximum / minimum). The discovery of POT EVT as the statistical theory behind combinatorial optimization can have a broad impact on any branch-and-bound optimization algorithm that operates on lower/upper bounds.

1.4 Neuro-Symbolic Systems for Language Models

The neuro-symbolic approach is also becoming a promising direction for supplementing the imprecise reasoning of Large Language Models (LLMs). In particular, so-called *tool-calling* ability (Schick et al. 2023) gives models access to precise symbolic algorithms. While LLMs are exciting technologies, they still massively lag behind traditional symbolic approaches in combinatorial and planning tasks, in terms of both the scalability and the reliability (Valmeekam et al. 2023; Goebel and Zips 2025).

Our STPR (Djuhera et al. 2025, Fig. 6) is a robotic pathfinding system that generates a Python program to define geometric constraints from prompts, then enforces them using traditional pathfinding algorithms. This results in a stricter rule-following behavior than recent zero-shot approaches that directly generate paths with LLMs.

I have also been working with a team leading the Generative Programming effort (Fulton et al. 2025) in IBM, where I lead a new algorithmic paradigm toward the responsible LLM use called Verbalized Algorithms (Lall et al. 2025, VA, Fig. 7). It limits the scope of LLMs by replacing the atomic operations of traditional discrete algorithms (e.g., comparison oracle in sorting algorithms) with LLM-based oracles that only answer to simple questions (e.g. yes/no). Applications of VA include sorting, top-k selection, clustering, submodular maximization, Pareto-front computation, combinatorial tasks, etc. **Unlike purely LLM-based reasoning approaches, VAs leverage existing results on correctness, time/space complexity, and parallelism**, and can sometimes quantify the theoretical error bound based on the error rate of the LLM-based oracle. Moreover, **the so-called reasoning ability ("think tags") is useful but not necessary in VAs**, enabling fast processing with a compact, quantized model.

VAs have advantages over another common approach to similar tasks that we call the "formalization-based approach" (Xie 2020; Guan et al. 2023; Li et al. 2024; Fine-Morris et al. 2024; Oswald et al. 2024; Olausson et al. 2023; Hao, Zhang, and Fan 2024). Leveraging the machine translation capability of LLMs, it converts an informal task description into a formal, symbolic representation, solves it with a specialized combinatorial solver, and then converts the solution back to the informal, natural language representation for the user, similar to Latplan. Among several issues, formalization suffers the most from the assumptions made in the target formal language that limit the scope of tasks it can solve. Verbalized algorithms, on the other hand, can handle them robustly through general and informal common sense reasoning on atomic operations.

Verbalization is an exciting new direction that could better tackle this challenge than formalization. However, there are multiple challenges ahead in solving planning tasks with "Verbalized Heuristic Search". For example, since heuristic search algorithms require a duplicate detection mechanism and a priority queue, verbalizing them requires verbalizing hash table operations, hash functions, and efficient priority queues as well.

2 Research Directions

My effort in the last several years established baselines for how to see each component of autonomous agent under the proper statistical lens. With these major groundwork, we could finally relax the current set of limiting assumptions in each of the component and tackle more general problems.

Latplan is currently restricted to fully-observable deterministic environments with discrete time steps, and to propositional, factored latent representations. However, we plan to leverage Latplan's strict adherence to the rich and expressive paradigm of Bayesian generative modeling to propose multiple extensions. For example, adding a concept of *duration* to each action as a new Exponentially-distributed latent variable would enable reasoning over continuous actions (Fig. 3, middle). Adding a concept of *objects* that each action acts on is a major undertaking that would yield a structured first-order-logic latent representation that generalizes over the objects in the environment (Fig. 3, right). Such an extension allows Latplan to generalize to environments with dynamically changing number of various types of never-seen physical objects, which humans and animals can adapt to without issues. Having multiple action variables at a time would capture multi-agent settings and exogenous effects.

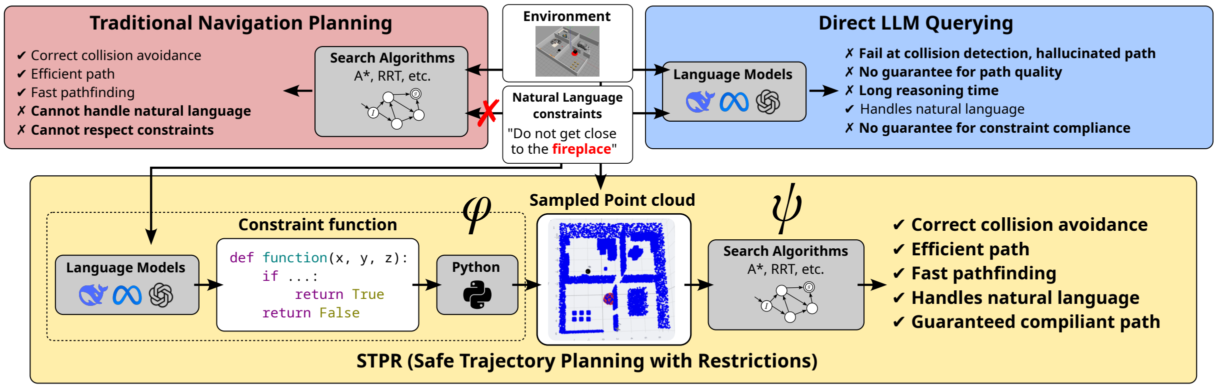

Figure 6: An overview of the STPR system.

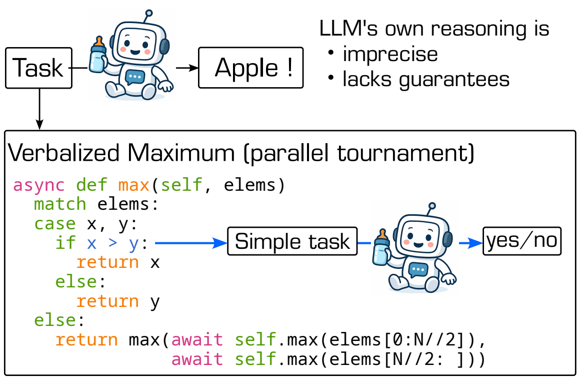

Figure 7: To select the sweetest element among ["water", "honey", "black coffee", "cucumber", "apple"], VA uses "is X sweeter than Y?" as a comparison query, and runs an $O(\log n)$ parallel tournament. By combining algorithmic framework with simple queries, VAs bypass the reasoning difficulty.

In a Task-and-Motion-Planning (Chitnis et al. 2022, TAMP) setting, actions must be further temporally decomposed into low-level motor controls. Finally, recent advancements in adversarial, contrastive, or diffusion-based generative models would provide significant boost in model accuracy.

In heuristic learning, our sharpened theoretical understanding raises multiple research questions. Can an upper bound, rather than the lower bound, of the solution cost be used for training as well? Does it help if we tighten the bounds during the search as it finds more solutions? Does Truncated Gaussian improve Reinforcement Learning? How does the truncation idea apply to the policy learning / learning-to-rank setting?

In search algorithms, prior to our work, heuristic functions have not received proper statistical treatments that fully respect their natural properties. Our work pioneered the statistical interpretation of heuristic functions in search algorithm, and opens similar opportunities for more advanced search settings. For example, in anytime satisficing search, a search algorithm first finds a suboptimal solution, then continues searching the state space for a better one. This setting is captured by the Simple Regret bandit algorithms, in contrast to the cumulative regret bandit algorithms we proposed. As another example, *alternating open list* (Röger and Helmert 2010) is a popular search enhancement that uniformly alternates multiple heuristic functions during the search. The theoretical mechanism of why this simple uniform switching works well is unknown, but applying a bandit-theoretic explanation and proposing a bandit-based switching is a promising direction.

The conventional view that sees heuristic search as MAB, as popularized in games, also requires a reevaluation. Most heuristic search algorithms operate in finite state spaces with *duplicate detection* that prevents revisiting the same state, so they will eventually run out of states to visit if the solution does not exist. MAB's assumption is significantly different, as it allows pulling each arm indefinitely. A possible alternative that reflects this nature of heuristic search is the so-called Scratch Game (Féraud and Urvoy 2013), a variant of MAB where the rewards are random samples from finite urns.

Finally, while I continue to bridge the gap between communities for each individual component, I also unify the different components of autonomous agents organically under a well-founded theoretical principle, which is a prerequisite for safe and performant autonomy. **Such an organic integration results in a unified system that enables end-to-end training and deployment.** For example, we may train the policy function and the symbolic representation at once while using an MCTS-based search algorithm. With such an effort, I imagine a future where autonomous systems deployed in everyday life can flexibly learn from past experiences and human commands while also being able to make deliberate system-2 decisions with precise logical reasoning under strict rule compliance.

References

Ahmetoglu, A.; Seker, M. Y.; Piater, J.; Oztop, E.; and Ugur, E. 2022. Deepsym: Deep Symbol Generation and Rule Learning for Planning from Unsupervised Robot Interaction. *J. Artif. Intell. Res.(JAIR)*, 75: 709–745.

Asai, M. 2019. Unsupervised Grounding of Plannable First-Order Logic Representation from Images. In *Proc. International*

Conference of Automated Planning and Scheduling (ICAPS) (CORE Rank: A).*

Asai, M. 2026. Bilevel MCTS for Amortized O (1) Node Selection in Classical Planning. In *Proc. AAAI Conference on Artificial Intelligence (CORE Rank: A*).*

Asai, M.; and Ayton, B. 2022. Is Policy Learning Overrated?: Width-Based Planning and Active Learning for Atari. In *Proc. International Conference of Automated Planning and Scheduling (ICAPS) (CORE Rank: A*).*

Asai, M.; and Fukunaga, A. 2018. Classical Planning in Deep Latent Space: Bridging the Subsymbolic-Symbolic Boundary. In *Proc. AAAI Conference on Artificial Intelligence (CORE Rank: A*).*

Asai, M.; Gehring, C.; Chitnis, R.; Silver, T.; Kaelbling, L. P.; Sohrabi, S.; and Katz, M. 2022a. Reinforcement Learning for Classical Planning: Viewing Heuristics as Dense Reward Generators. In *Proc. International Conference of Automated Planning and Scheduling (ICAPS) (CORE Rank: A*).*

Asai, M.; and Kajino, H. 2019. Towards Stable Symbol Grounding with Zero-Suppressed State AutoEncoder. In *Proc. International Conference of Automated Planning and Scheduling (ICAPS) (CORE Rank: A*).*

Asai, M.; Kajino, H.; Fukunaga, A.; and Muise, C. 2021. Symbolic Reasoning in Latent Space: Classical Planning as an Example. In Hitzler, P.; and Sarker, M. K., eds., *Neuro-Symbolic Artificial Intelligence: The State of the Art*, volume 342 of *Frontiers in Artificial Intelligence and Applications*, 52–77. IOS Press.

Asai, M.; Kajino, H.; Fukunaga, A.; and Muise, C. 2022b. Classical Planning in Deep Latent Space. *J. Artif. Intell. Res. (JAIR) (CORE Rank: A).*

Asai, M.; and Muise, C. 2020. Learning Neural-Symbolic Descriptive Planning Models via Cube-Space Priors: The Voyage Home (to STRIPS). In *Proc. International Joint Conference on Artificial Intelligence (IJCAI) (CORE Rank: A*).*

Asai, M.; and Wissow, S. 2026. Extreme Value Monte Carlo Tree Search for Classical Planning. In *Proc. AAAI Conference on Artificial Intelligence (CORE Rank: A*).*

Balkema, A. A.; and De Haan, L. 1974. Residual Life Time at Great Age. *Annals of Probability*, 2(5): 792–804.

Bonet, B.; and Geffner, H. 2001. Planning as Heuristic Search. *Artificial Intelligence*, 129(1): 5–33.

Chitnis, R.; Silver, T.; Tenenbaum, J. B.; Lozano-Perez, T.; and Kaelbling, L. P. 2022. Learning Neuro-Symbolic Relational Transition Models for Bilevel Planning. In *Proc. of IEEE International Workshop on Intelligent Robots and Systems (IROS)*, 4166–4173. IEEE.

Djuhera, A.; Seffo, A.; Asai, M.; and Boche, H. 2025. “Don’t Do That!”: Guiding Embodied Systems through Large Language Model-based Constraint Generation. *arXiv:2506.04500 (under submission @ ICRA 2026).*

Féraud, R.; and Urvoy, T. 2013. Exploration and Exploitation of Scratch Games. *Machine Learning*, 92: 377–401.

Fikes, R. E.; Hart, P. E.; and Nilsson, N. J. 1972. Learning and Executing Generalized Robot Plans. *Artificial Intelligence*, 3(1-3): 251–288.

Fine-Morris, M.; Hsiao, V.; Smith, L. N.; Hiatt, L. M.; and Roberts, M. 2024. Leveraging LLMs for Generating Document-Informed Hierarchical Planning Models: A Proposal. In *AAAI 2025 Workshop LM4Plan*.

Fulton, N.; Strobelt, H.; LoRocco, J.; Balakrishnan, A.; and Asai, M. 2025. Mellea: Build enterprise AI without guesswork.

Goebel, K.; and Zips, P. 2025. Can LLM-Reasoning Models Replace Classical Planning? A Benchmark Study. *arXiv preprint arXiv:2507.23589*.

Guan, L.; Valmeekam, K.; Sreedharan, S.; and Kambhampati, S. 2023. Leveraging Pre-Trained Large Language Models to Construct and Utilize World Models for Model-Based Task Planning. *Advances in Neural Information Processing Systems*, 36: 79081–79094.

Hao, Y.; Zhang, Y.; and Fan, C. 2024. Planning Anything with Rigor: General-Purpose Zero-Shot Planning with LLM-based Formalized Programming. In *Proc. of the International Conference on Learning Representations*.

Hart, P. E.; Nilsson, N. J.; and Raphael, B. 1968. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *Systems Science and Cybernetics, IEEE Transactions on*, 4(2): 100–107.

Helmert, M. 2006. The Fast Downward Planning System. *J. Artif. Intell. Res.(JAIR)*, 26: 191–246.

Jang, E.; Gu, S.; and Poole, B. 2017. Categorical Reparameterization with Gumbel-Softmax. In *Proc. of the International Conference on Learning Representations*.

Kocsis, L.; and Szepesvári, C. 2006. Bandit Based Monte-Carlo Planning. In *Proc. of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, 282–293. Springer.

Lall, S.; Farrell, C.; Pathanjaly, H.; Pavic, M.; Chezhian, S.; and Asai, M. 2025. Verbalized Algorithms. In *NeurIPS 2025 Workshop on Efficient Reasoning*.

Li, R.; Cui, L.; Lin, S.; and Haslum, P. 2024. Naruto: Automatically Acquiring Planning Models from Narrative Texts. In *Proc. of AAAI Conference on Artificial Intelligence*, volume 38, 20194–20202.

Newell, A.; Shaw, J. C.; and Simon, H. A. 1959. Report on a General Problem Solving Program. In *IFIP congress*, volume 256, 1959. Pittsburgh, PA.

Núñez-Molina, C.; Asai, M.; Mesejo, P.; and Fernández-Olivares, J. 2024. On Using Admissible Bounds for Learning Forward Search Heuristics. In *Proc. International Joint Conference on Artificial Intelligence (IJCAI) (CORE Rank: A*).*

Olausson, T. X.; Gu, A.; Lipkin, B.; Zhang, C. E.; Solar-Lezama, A.; Tenenbaum, J. B.; and Levy, R. P. 2023. LINC: A Neurosymbolic Approach for Logical Reasoning by Combining Language Models with First-Order Logic Provers.

Oswald, J.; Srinivas, K.; Kokel, H.; Lee, J.; Katz, M.; and Sohrabi, S. 2024. Large Language Models as Planning Domain Generators. In *Proc. of the International Conference on Automated Planning and Scheduling (ICAPS)*, volume 34, 423–431.

Pickands III, J. 1975. Statistical Inference using Extreme Order Statistics. *Annals of Statistics*, 119–131.

Robinson, N.; McIlraith, S.; and Toman, D. 2014. Cost-based query optimization via AI planning. In *Proc. of AAAI Conference on Artificial Intelligence*, volume 28.

Rodriguez, I. D.; Bonet, B.; Romero, J.; and Geffner, H. 2021. Learning First-Order Representations for Planning from Black Box States: New Results. In *Proc. of the International Conference on Principles of Knowledge Representation and Reasoning (KR)*, volume 18, 539–548.

Röger, G.; and Helmert, M. 2010. The More, the Merrier: Combining Heuristic Estimators for Satisficing Planning. In *Proc. of the International Conference on Automated Planning and Scheduling (ICAPS)*.

Schick, T.; Dwivedi-Yu, J.; Dessì, R.; Raileanu, R.; Lomeli, M.; Hambro, E.; Zettlemoyer, L.; Cancedda, N.; and Scialom, T. 2023. Toolformer: Language models can teach themselves to use tools. *Advances in Neural Information Processing Systems*, 36: 68539–68551.

Shaik, I.; and Van De Pol, J. 2023. Optimal Layout Synthesis for Quantum Circuits as Classical Planning. In *2023 IEEE/ACM International Conference on Computer Aided Design (ICCAD)*, 1–9. IEEE.

Speck, D.; Höft, P.; Gnad, D.; and Seipp, J. 2023. Finding Matrix Multiplication Algorithms with Classical Planning. In *Proc. of the International Conference on Automated Planning and Scheduling (ICAPS)*, volume 33, 411–416.

Valmeekam, K.; Marquez, M.; Sreedharan, S.; and Kambhampati, S. 2023. On the Planning Abilities of Large Language Models – A Critical Investigation. *Advances in Neural Information Processing Systems*, 36: 75993–76005.

Wissow, S.; and Asai, M. 2024. Scale-Adaptive Balancing of Exploration and Exploitation in Classical Planning. In *European Conference on Artificial Intelligence (ECAI) (CORE Rank: A)*. **Outstanding Paper Award**.

Xie, Y. 2020. Translating Natural Language to Planning Goals with Large-Language Models. *The International Journal of Robotics Research*, 2019: 1.