Real-World Applications and Benchmarks

Bridging Planning and Reasoning in Natural Language with Foundation Models

Wenjun Li

https://plan-fm.github.io/

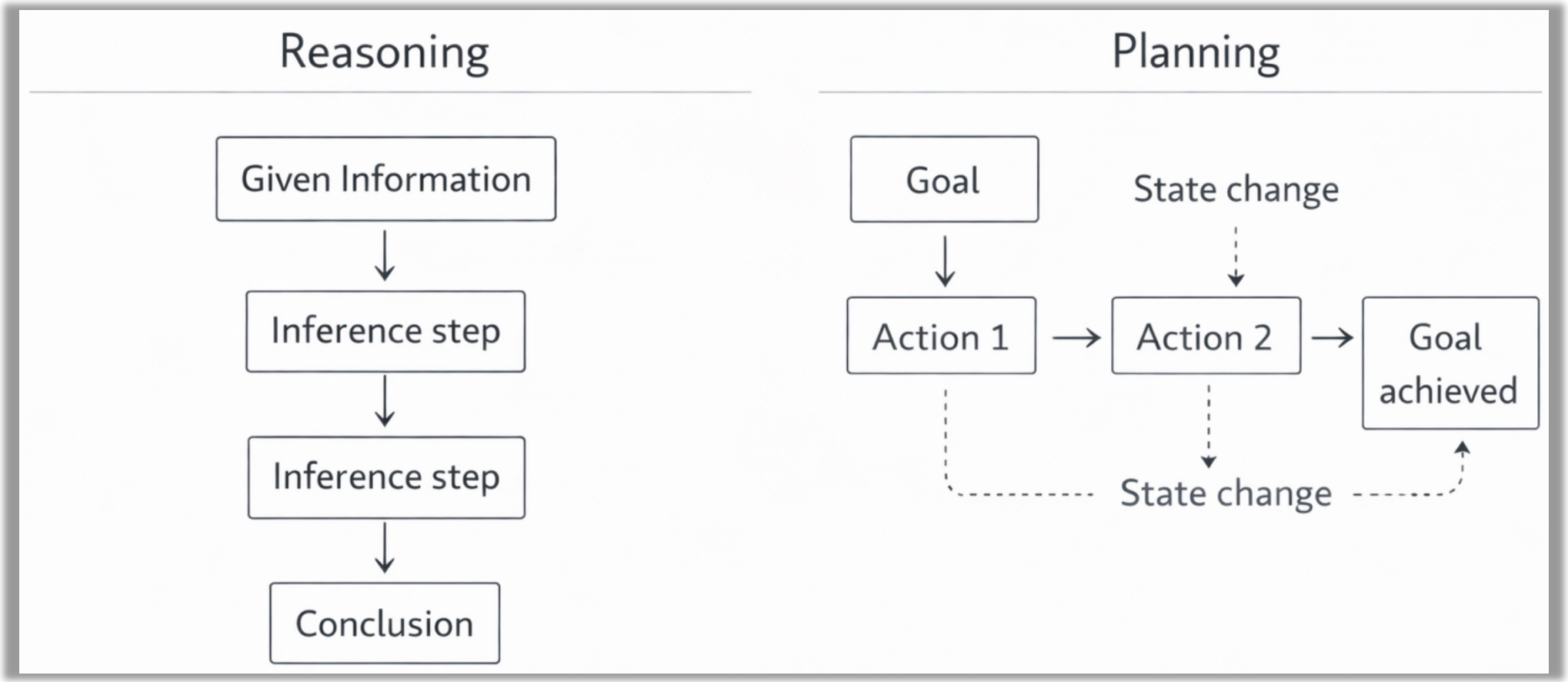
The Year of Agents

- Triggered by:
  - Stronger reasoning models (post-training + RL)
  - Reliable tool use and memory
  - Deployment-ready agent frameworks
- Planning unlocked end-to-end automation, not just assistance:
  - Multi-step workflows
  - Cross-system coordination

Real-World Applications

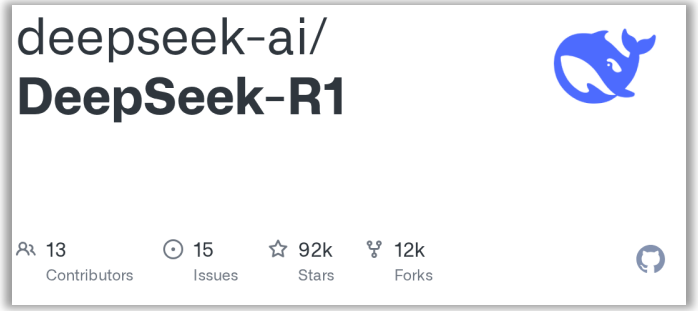
- Personal Assistants
  - Manus, Google Gemini ecosystem, Notebook LM
- Code Agents
  - Cursor, Claude code
- Real-World Robotics and Embodied AI
  - Tesla Optimus, Helix
- Enterprise Decision & Workflow Automation
  - Microsoft Copilot, Salesforce Einstein Copilot
- Research & Deep-Analysis Agents
  - OpenAI Deep Research, Perplexity AI (Research mode)

Planning & Reasoning in LLMs: Definition and Significance



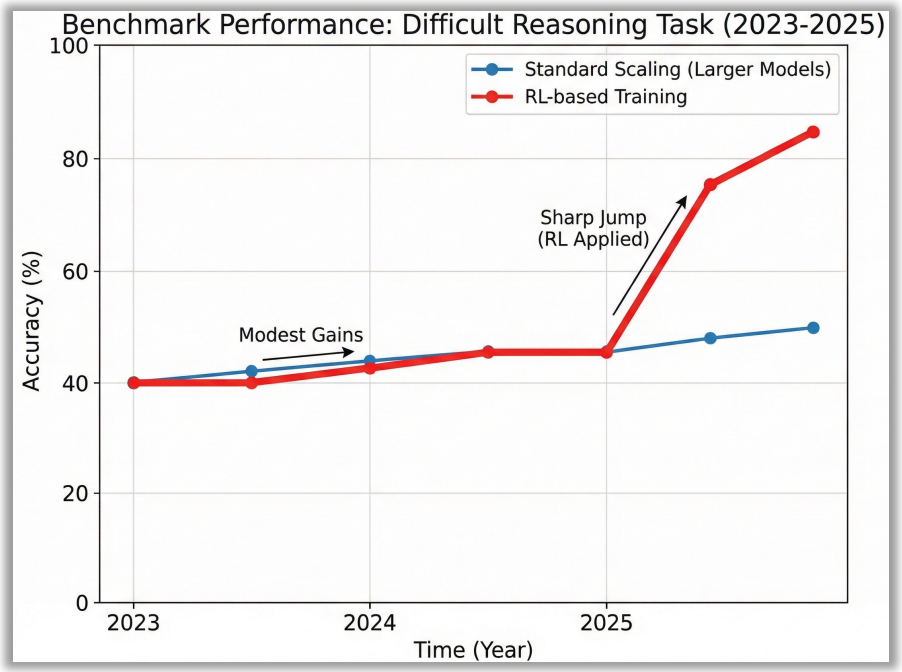
Background: A Breakthrough for LLM Reasoning with RL

- **RL Training:** new RL-based training methods can install “reasoning-like” behavior in LLMs beyond what scaling alone achieved
- **Verifiable Reward Signals:** RLVR uses automated checks to label an LLM’s answers as correct or not, instead of needing human feedback.
- **DeepSeek R1 Example:** In early 2025, DeepSeek R1 used RLVR (with GRPO) to attain reasoning abilities on par with top proprietary models – a pivotal moment showing RL can effectively teach LLMs to “think.”



Background: A Breakthrough for LLM Reasoning with RL

- **Widespread Adoption:** 2025 became the “year of reasoning LLMs”. Reasoning LLMs moved from novel to mainstream.
- **Big Leaps on Benchmarks:** Previously unsolvable multi-step tasks are now within reach.



Outline

- **Natural Language Plan Generation:**
  - LLMs creating multi-step plans from text instructions (e.g. generating a travel itinerary).
- **Reasoning about Actions & Change:**
  - Understanding the effects of actions in described scenarios (state tracking, plan validation, etc.).
- **NL-to-PDDL Planning Translation:**
  - Converting natural language problems into formal planning representations for classical solvers.
- **Interactive Agents & Environments:**
  - LLMs as agents planning and acting in simulated worlds.
- **Conclusion**

Natural Language Plan Generation

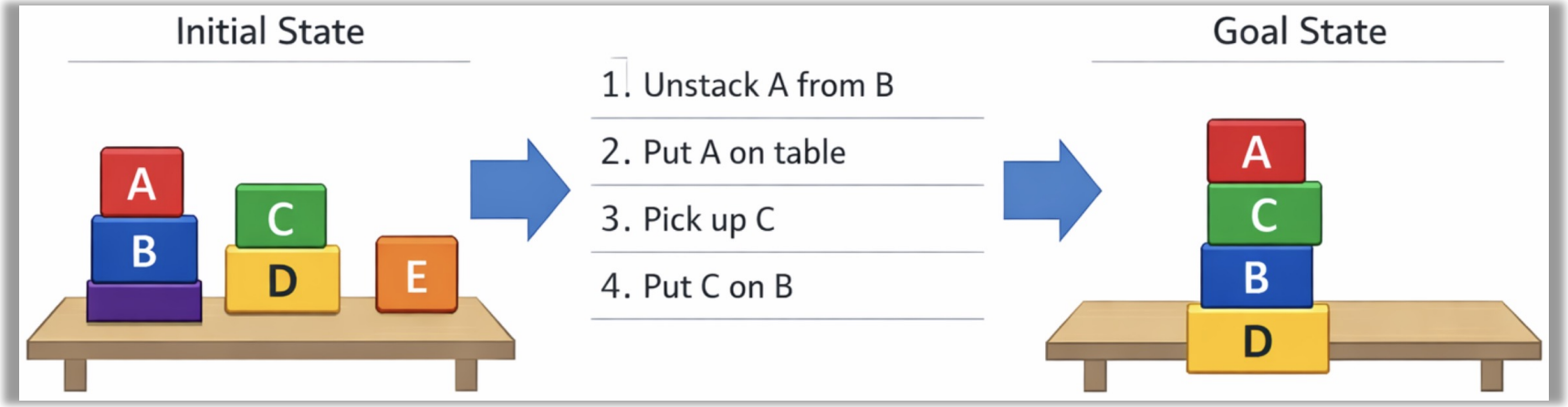
- **Core Capability**
  - Generate a **multi-step, temporally ordered plan** from an open-ended natural language goal..
- **Key characteristics**
  - **Plan synthesis**: Output is a full action sequence, not a single answer.
  - **Long-horizon coherence**: Steps must remain consistent over time and respect dependencies.
  - **Implicit constraints**: Time, feasibility, and resources are often unstated.
  - **Human-centric evaluation**: Plans are judged by realism, usefulness, and common sense.
  - **Practical relevance**: Closely mirrors real-world uses such as travel planning, scheduling, and task organization.

Natural Language Plan Generation

Dimension	AutoPlanBench (Stein et al., 2023)	Natural Plan (Zheng et al., 2024)	Travel Planner (Kie et al., 2024)	TripTailor (Wang et al., 2025)
Domain scope	Broad, multi-domain	Everyday tasks	Travel only	Travel only
Real-world realism	Medium	High	High	Very high
Personalization	Low	Medium	Medium	High
Evaluation focus	Plan validity	Strategy quality	Practical feasibility	Human-level quality

Reasoning about Actions & Change

- **Core Capability**
  - Given a **described plan**, determine whether actions are executable, track state changes, and verify plan correctness.
- **Key characteristics**
  - **State-centric reasoning**: Focuses on how actions change the world and what remains invariant.
  - **Analysis, not synthesis**: The plan is provided; the task is to simulate, validate, or predict outcomes.
  - **Explicit action semantics**: Requires understanding preconditions, effects, and causal relations.
  - **Multi-step consistency**: Small state-tracking errors compound quickly across steps.
  - **Alignment with classical planning**: Closely tied to fluents, executability, and plan validity.

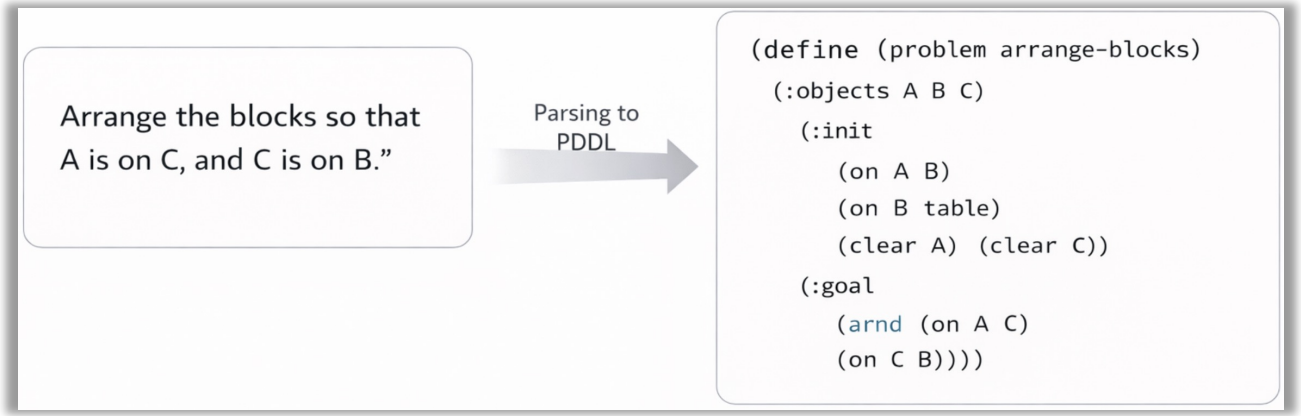


Reasoning about Actions & Change

Dimension	TRAC (He et al., ACL 2023)	ActionReasoningBench (Handa et al., ICLR 2025)	ACPBench (Kokel et al., AAAI 2025)	MAP-THOR (Nayak et al., 2025)
Domain	Blocksworld	Multiple domains	Classical planning domains	3D household environment
Reasoning focus	Step-level state tracking	Step & rule-level reasoning	Structural & causal reasoning	Multi-agent, contextual reasoning
Planning depth	Medium	Medium	High	Medium
Observability	Full	Full	Full	Partial
Agents & dynamics	Single, static	Single, static	Single, static	Multiple, dynamic & embodied

NL-to-PDDL Planning Translation

- **Core Capability**
  - Translate **informal natural language tasks** into **formal PDDL planning problems** that classical planners can solve..
- **Key characteristics**
  - **Language-to-symbol grounding**: Map entities, relations, and goals into predicates and objects.
  - **Formal correctness**: Small errors (missing predicates, wrong arity) cause planner failure.
  - **Explicit planning structure**: Identify state variables, preconditions, and goals.
  - **Planner-in-the-loop evaluation**: Success is measured by whether a classical planner can solve the output.
  - **Neural–symbolic bridge**: Connects LLM reasoning with decades of symbolic planning.

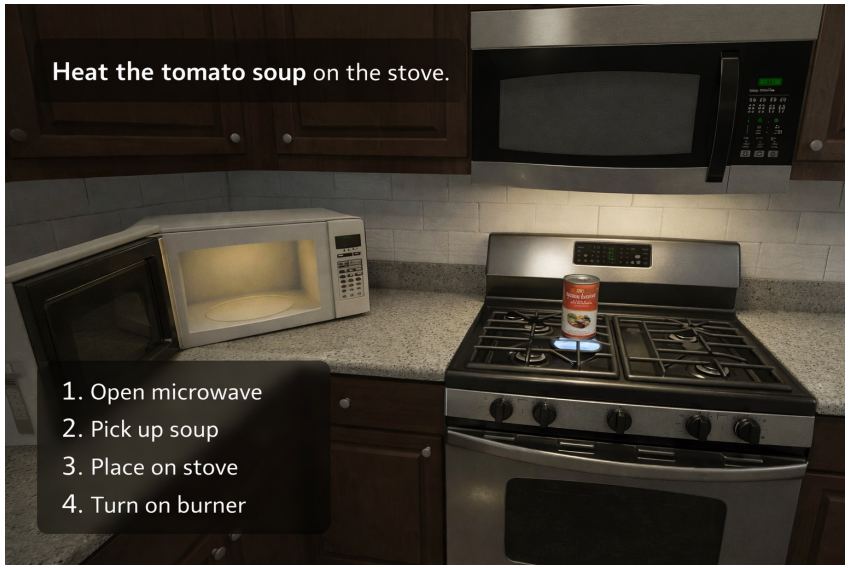


NL-to-PDDL Planning Translation

Dimension	NL2PDDL (Oswald et al., ICAPS 2024)	LLM+P (Liu et al., 2023)	Planetarium (Zuo et al., 2024)	PLANET (Li et al., 2025)
Primary output	PDDL problems	Plans + PDDL	PDDL problems	Comparative analysis
Input style	Structured NL	Domain + task	Open-ended text	Literature survey
Formal rigor	High	Medium–High	High	N/A
Domain complexity	Low–Medium	Medium	Medium–High	Varies
Evaluation	Planner success	Plan validity	Planner success	Cross-benchmark comparison

Interactive Agents in Simulated Environments

- **Core Capability**
  - Act as an **autonomous agent** that plans, executes actions, observes outcomes, and adapts through **multi-step, closed-loop interaction** with an environment.
- **Key characteristics**
  - **Closed-loop planning**: Planning, execution, observation, and re-planning are interleaved.
  - **Long-horizon decisions**: Early mistakes compound; recovery and correction are critical.
  - **Partial observability**: The agent never has full access to the environment state.
  - **Grounded actions**: Actions have concrete and often irreversible effects.
  - **Beyond correctness**: Robustness, recovery, coordination, and safety matter as much as success.



Interactive Environments: Embodied vs Web/GUI Agents

Aspect	Embodied Environments	Web / GUI Environments
Action space	Navigation, manipulation	Click, type, scroll, submit
Perception	Vision, state, proprioception	HTML, text, UI elements
Environment noise	Occlusion, physics, timing	Pop-ups, dynamic pages, hidden UI
Planning challenges	Parallel actions, timing	Long memory, tool use, context limits
Generalization	New layouts, object configurations	Unseen websites and workflows

Benchmarks for Interactive Agents

Benchmark	Environment Type	Key Focus	Horizon & Dynamics	What It Tests
ALFRED (Shridhar et al., 2020)	Embodied (household)	Vision-grounded planning	Medium, static	Language → action grounding
ALFWorld (Shridhar et al., 2021)	Embodied (text)	State tracking	Medium, static	Planning without vision
TextCraft (Prasad et al., NAACL 2024)	Embodied (game)	Long-horizon planning	Long, semi-dynamic	Strategy and memory
Robotouille (Baker et al., 2025)	Embodied (kitchen)	Parallel actions	Long, dynamic	Timing & coordination
SafeAgentBench (Yin et al., 2025)	Embodied (simulated)	Safety & refusal	Short–Medium	Planning under hazards
WebArena (Yao et al., 2022)	Web (simulated)	End-to-end workflows	Medium–Long	Tool use & navigation
Mind2Web (Ammanabrolu et al., 2023)	Web (real websites)	Generalization	Long	Transfer to unseen sites
AgentBench (Ma et al., NeurIPS 2024)	Web + tools	Comparative evaluation	Varies	Agent capability spectrum

Conclusion

- **Planning and reasoning enable real-world LLM applications**
  - Planning and reasoning transform LLMs from QA systems into goal-directed agents, enabling complex, multi-step applications across productivity, enterprise workflows, web interaction, and embodied settings.
- **Reinforcement learning is the key enabler**
  - RL-based post-training has fundamentally strengthened long-horizon reasoning and planning, allowing LLMs to generate, revise, and execute plans that were previously out of reach.
- **Benchmarks now drive capability—not just measurement**
  - A diverse ecosystem of planning and reasoning benchmarks is shaping progress by exposing failure modes, guiding model design, and connecting modern LLMs with classical planning principles.
- **A convergence of neural and symbolic paradigms**
  - Planning-capable LLMs sit at the intersection of neural models and symbolic planning, combining flexibility with structure and moving closer to reliable decision-making systems.

Thank You !