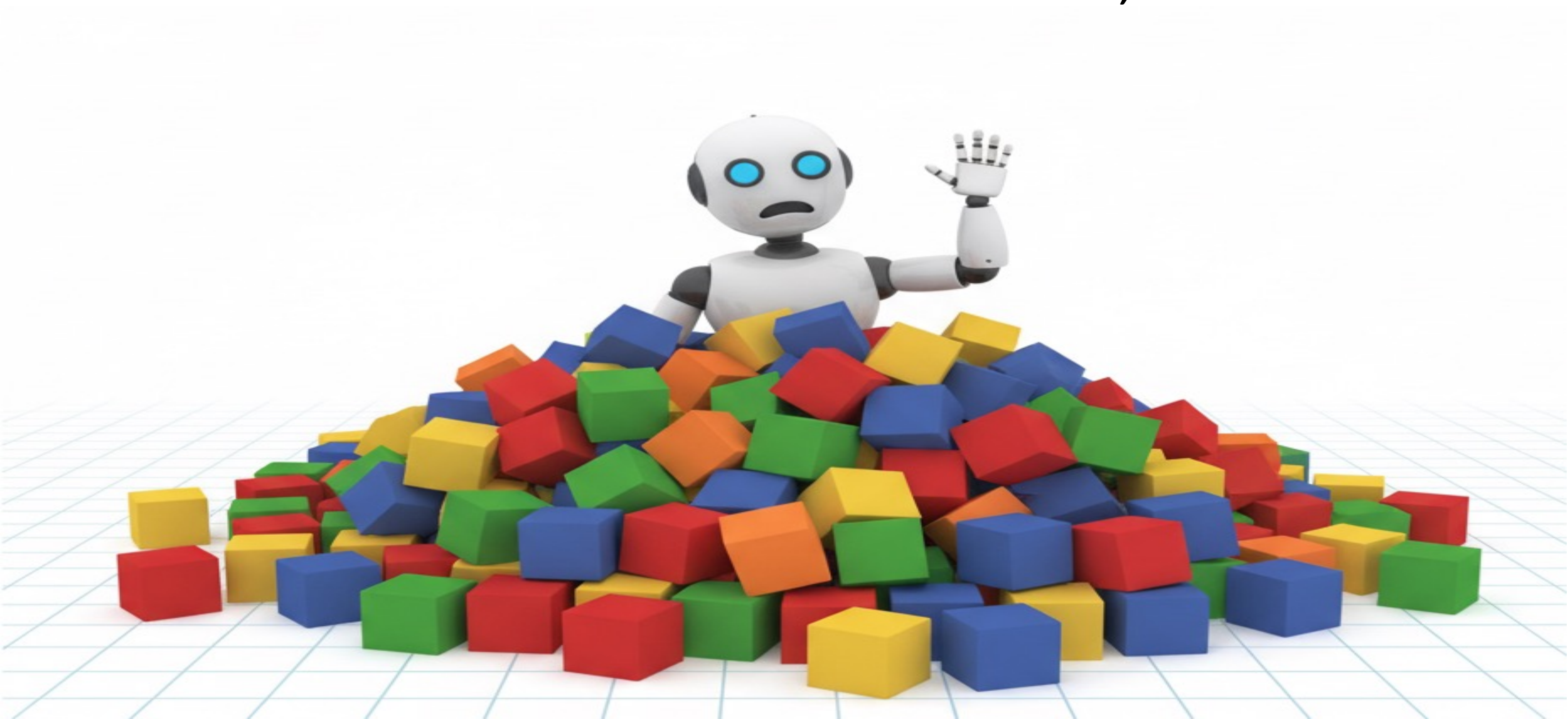
Planning in the Era of Language Models

ICAPS 2026 Summer School

Michael Katz, IBM



Planning problem

Mathematical Model:

- finite and discrete state space S
- a known initial state $s_0 \in S$
- a set $S_G \subseteq S$ of goal states
- actions $A(s) \subseteq A$ applicable in each $s \in S$
- a deterministic transition function
$$s' = f(a, s) \text{ for } a \in A(s)$$
- non-negative action costs $c(a, s)$

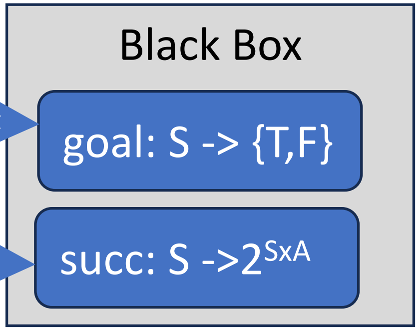
A solution is a sequence of applicable actions that maps s_0 into S_G, and it is optimal if it minimizes sum of action costs

Planning problem

Mathematical Model:

- finite and discrete state space S
- a known initial state $s_0 \in S$
- a set $S_G \subseteq S$ of goal states
- actions $A(s) \subseteq A$ applicable in each $s \in S$
- a deterministic transition function

  $$s' = f(a, s) \text{ for } a \in A(s)$$
- non-negative action costs $c(a, s)$



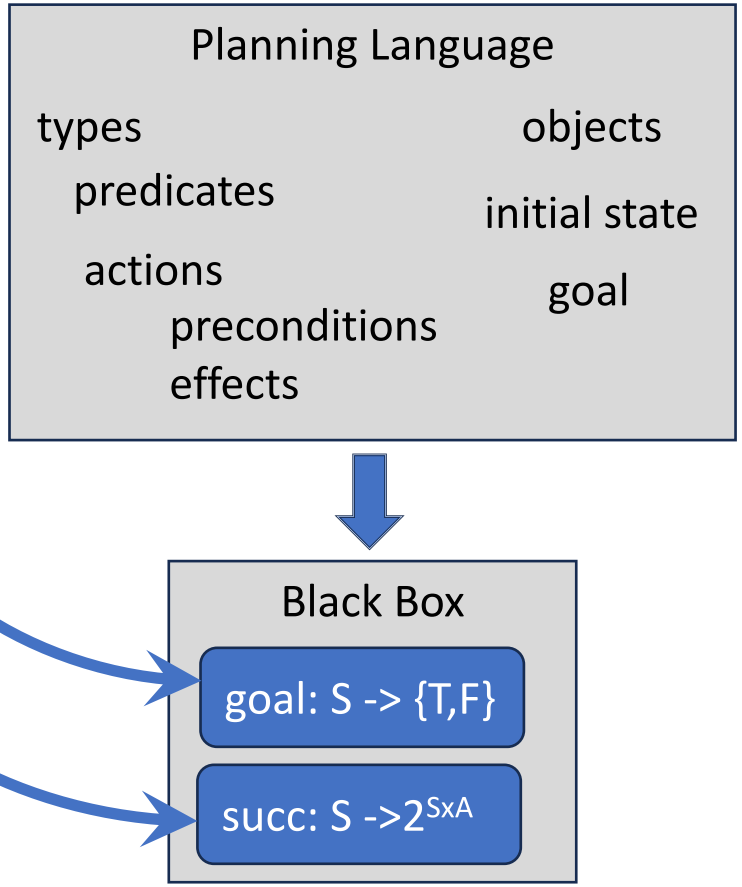
A solution is a sequence of applicable actions that maps s_0 into S_G, and it is optimal if it minimizes sum of action costs

Planning problem

Mathematical Model:

- finite and discrete state space S
- a known initial state $s_0 \in S$
- a set $S_G \subseteq S$ of goal states
- actions $A(s) \subseteq A$ applicable in each $s \in S$
- a deterministic transition function
  $$s' = f(a, s) \text{ for } a \in A(s)$$
- non-negative action costs $c(a, s)$

A solution is a sequence of applicable actions that maps s_0 into S_G, and it is optimal if it minimizes sum of action costs



Planning problem

The 24 Game is a mathematical card game in which the objective is to find a way to manipulate four integers so that the end result is 24. The game is played with a list of four numbers, and the player must use all four numbers exactly once, using any combination of addition, subtraction, multiplication, or division, to arrive at the number 24.

Mathematical Model:

- finite and discrete state space S
- a known initial state $s_0 \in S$
- a set $S_G \subseteq S$ of goal states
- actions $A(s) \subseteq A$ applicable in each $s \in S$
- a deterministic transition function

  $s' = f(a, s)$ for $a \in A(s)$
- non-negative action costs $c(a, s)$

A solution is a sequence of applicable actions that maps s_0 into S_G, and it is optimal if it minimizes sum of action costs

Planning Language

types
predicates
actions
preconditions
effects

objects
initial state
goal

Black Box

goal: S -> {T,F}

succ: S -> 2^{SxA}

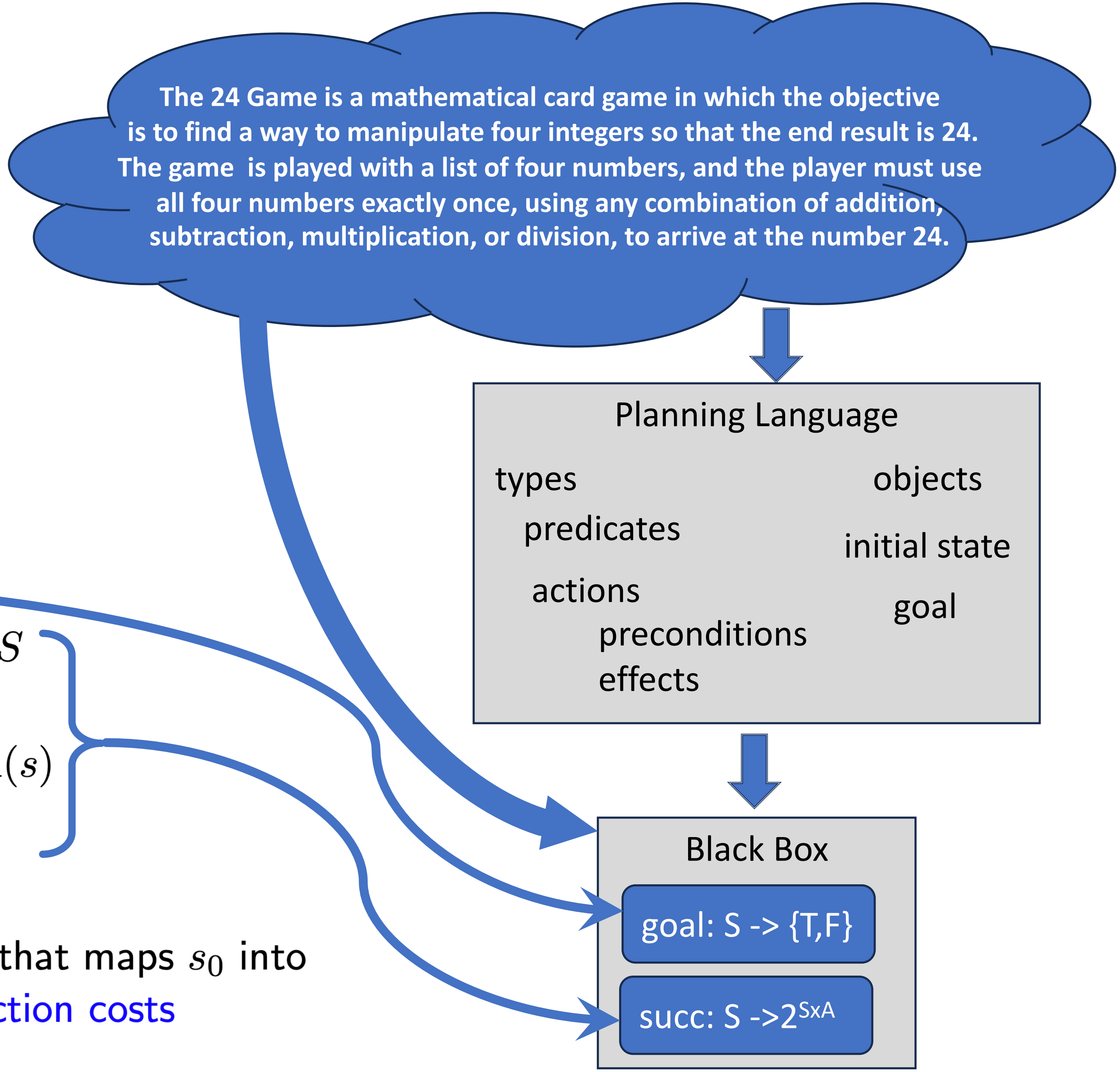
Planning problem

Mathematical Model:

- finite and discrete state space S
- a known initial state $s_0 \in S$
- a set $S_G \subseteq S$ of goal states
- actions $A(s) \subseteq A$ applicable in each $s \in S$
- a deterministic transition function

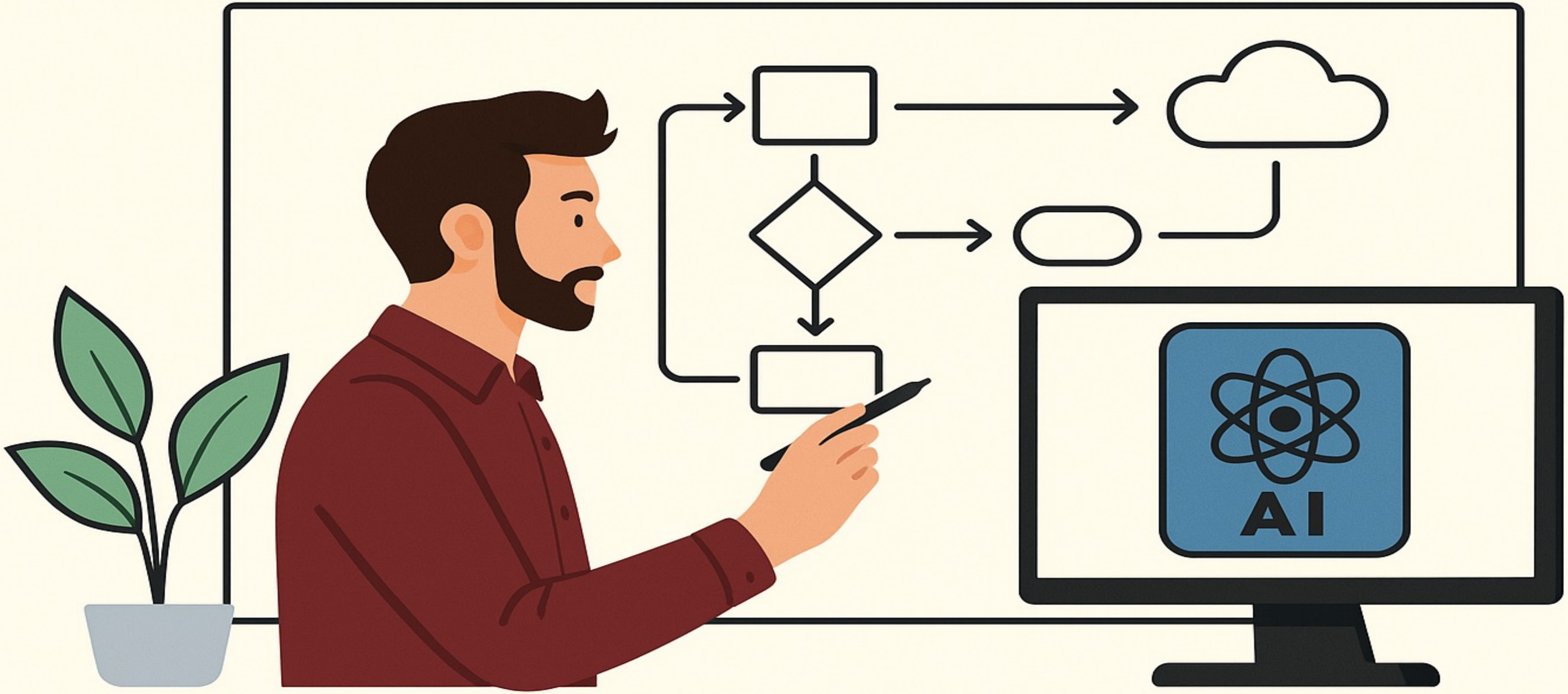
$$s' = f(a, s) \text{ for } a \in A(s)$$

- non-negative action costs $c(a, s)$

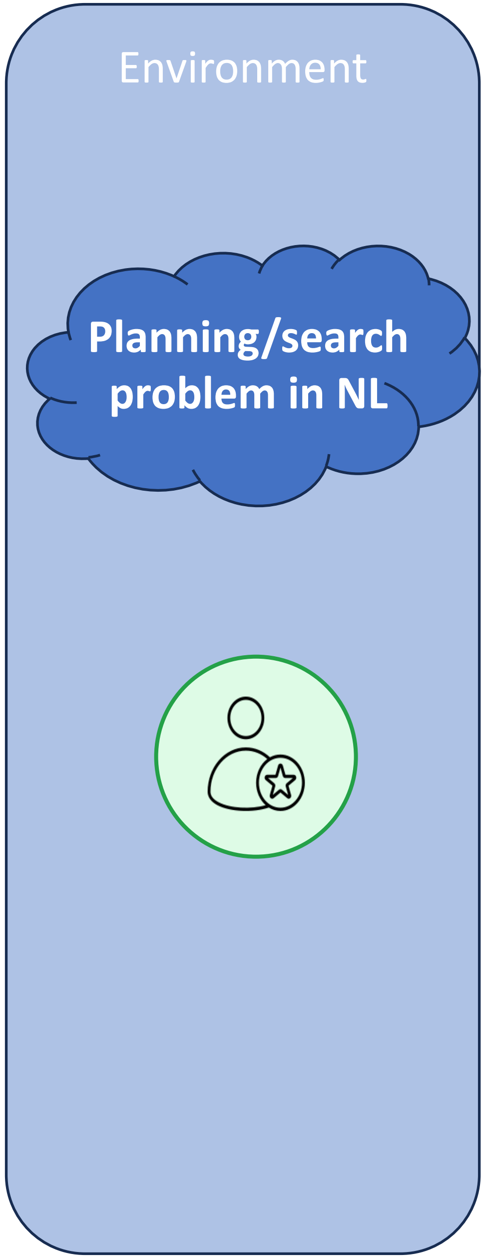


A solution is a sequence of applicable actions that maps s_0 into S_G, and it is optimal if it minimizes sum of action costs

PLANNING IN THE EARLY LLM ERA

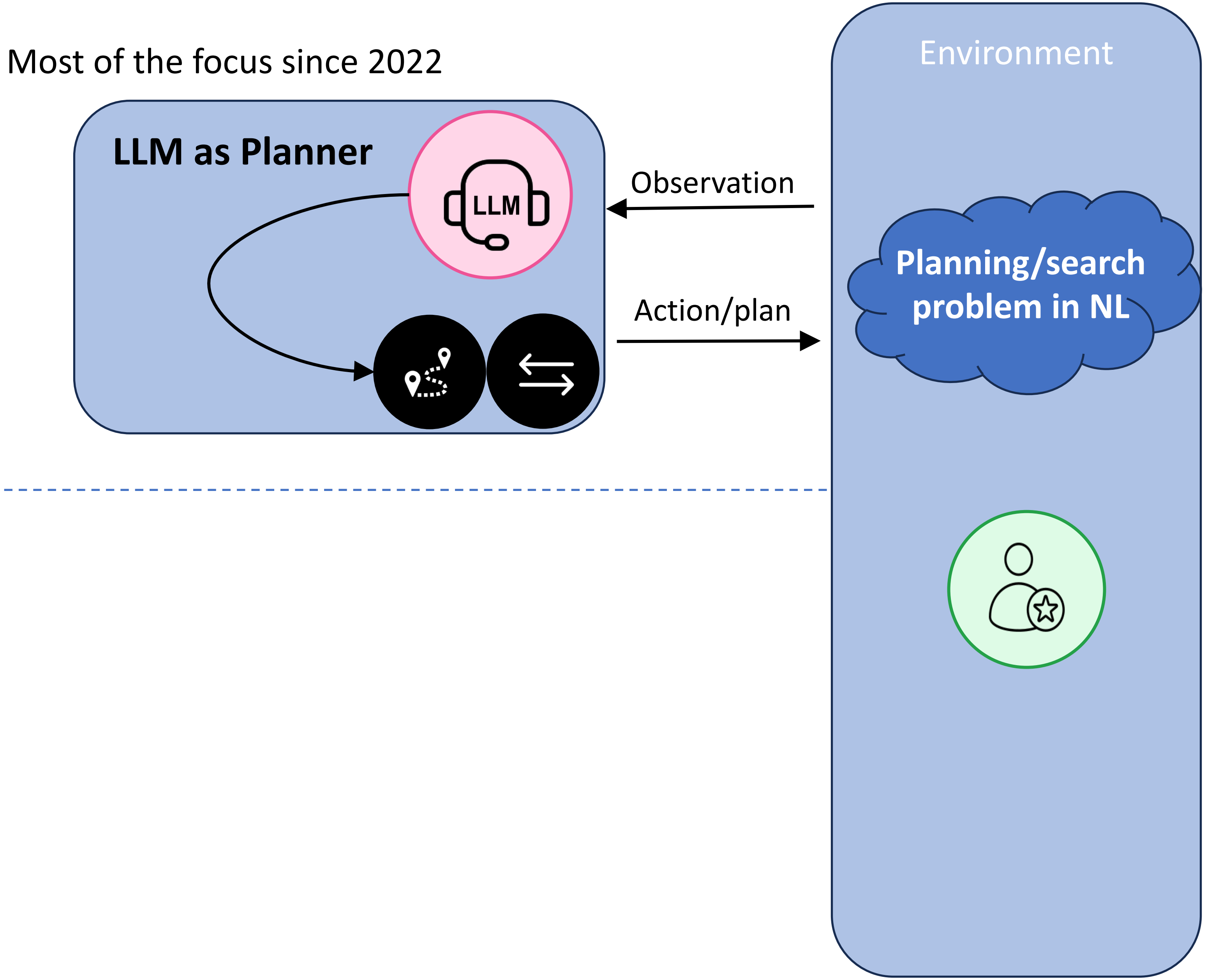


Solving NL/PDDL planning problems



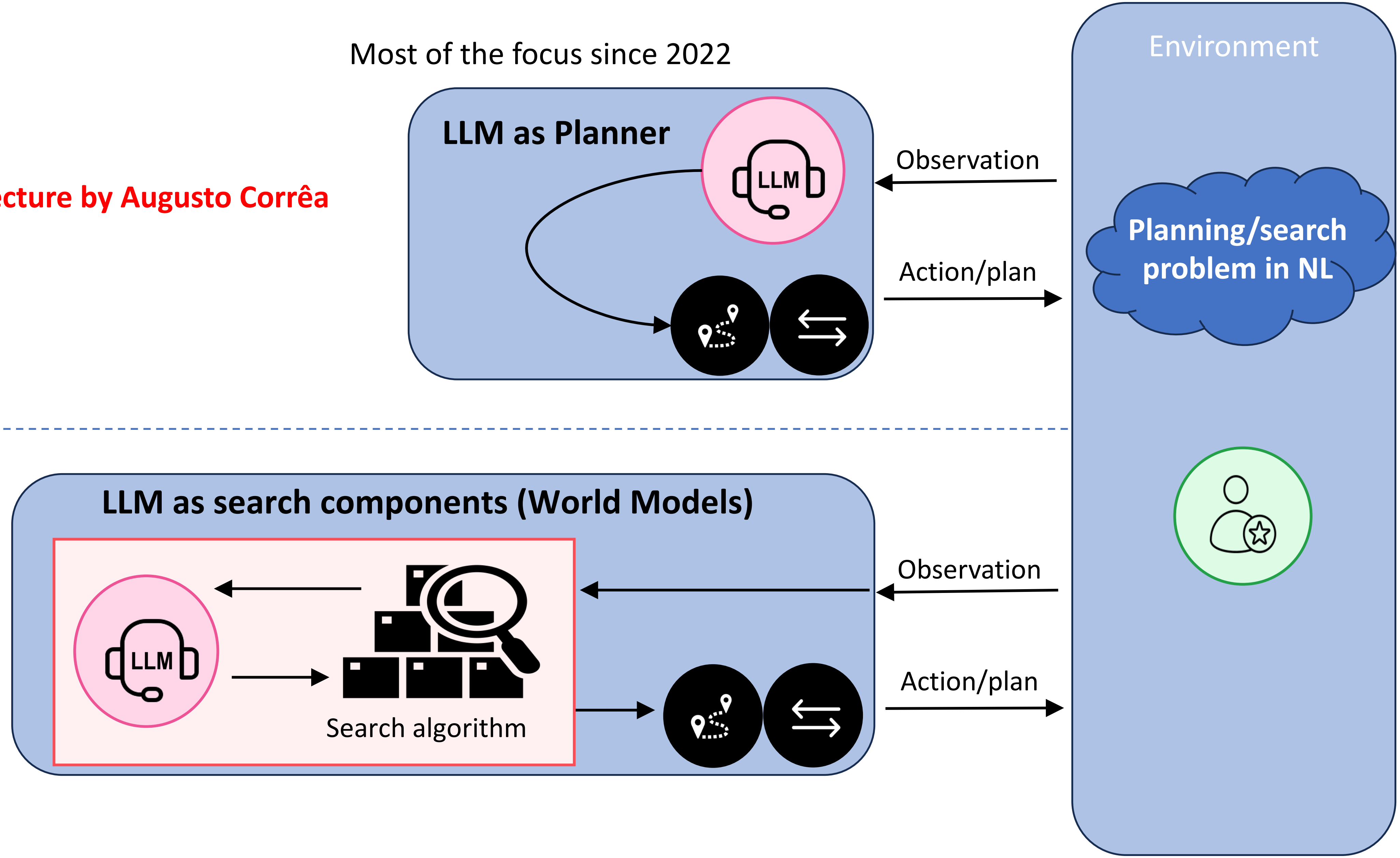
Solving NL/PDDL planning problems

Previous lecture by Augusto Corrêa



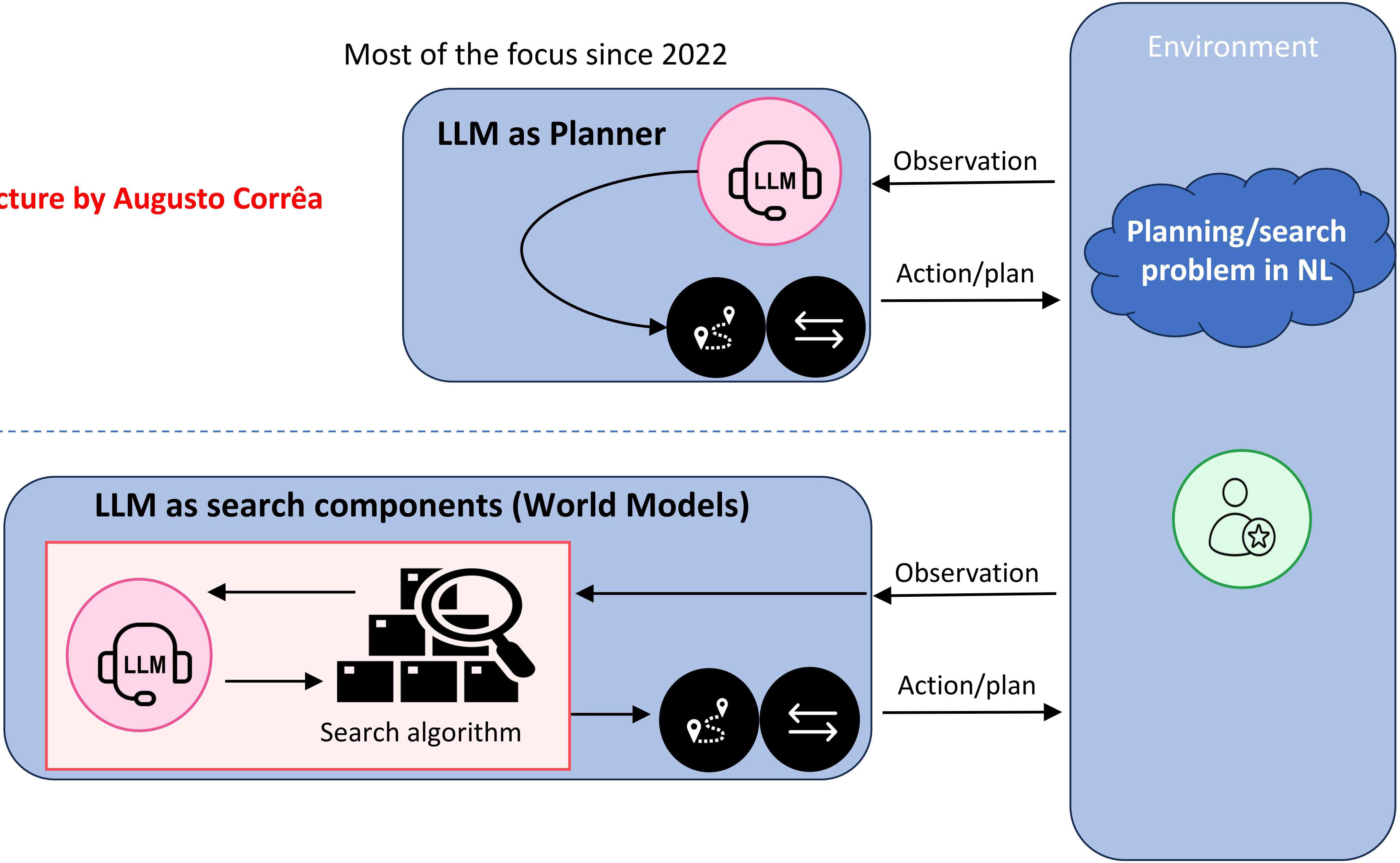
Solving NL/PDDL planning problems

Previous lecture by Augusto Corrêa



Solving NL/PDDL planning problems

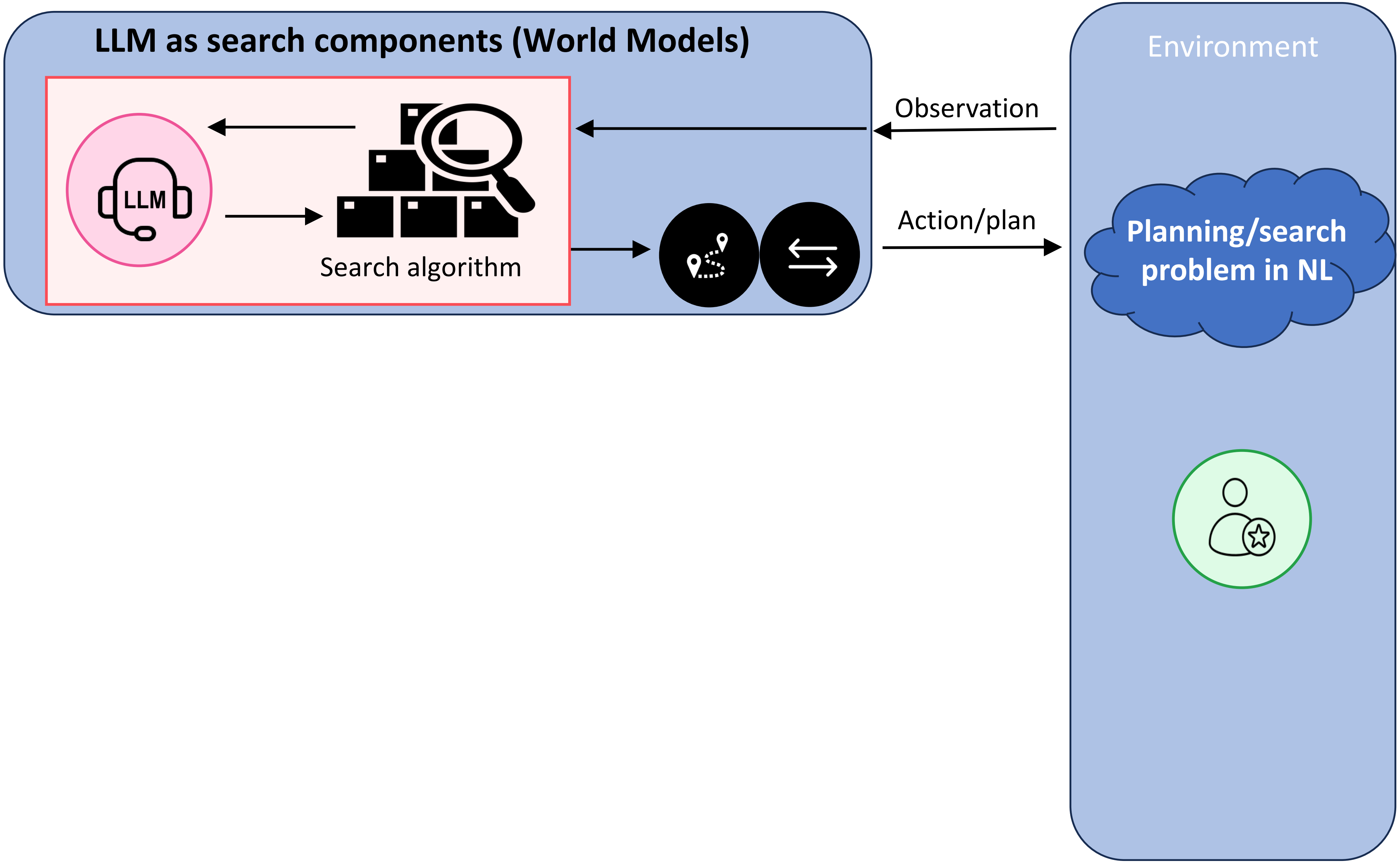
Previous lecture by Augusto Corrêa



Yao et al. ICLR 2023
Xu et al. Arxiv 2023
Hao et al. EMNLP 2023
Yao et al. NeurIPS 2023
Shinn et al. NeurIPS 2023
Zhou et al. ICML 2024
Sel et al. ICML 2024
Besta et al. AAAI 2024

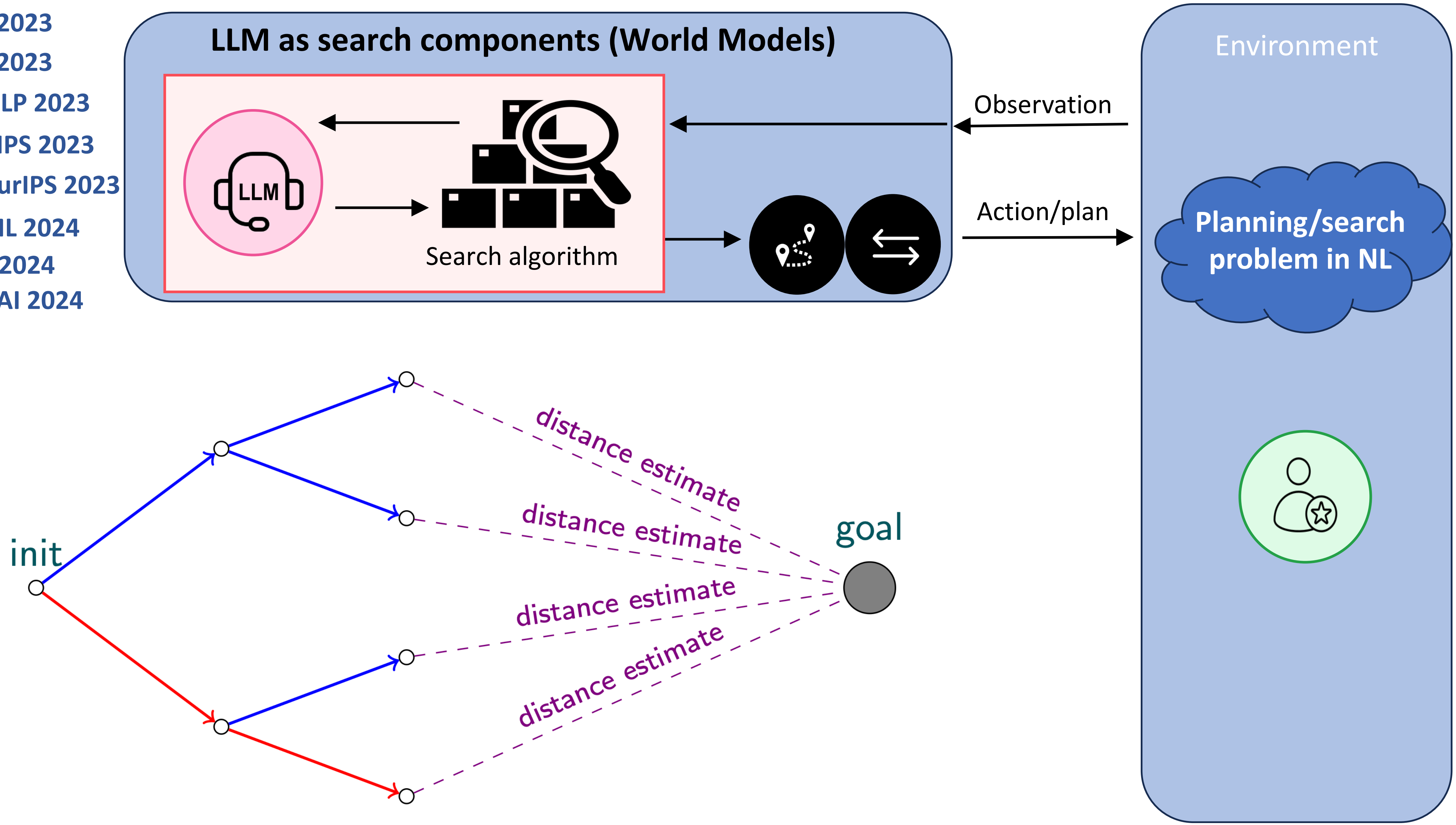
LLM as a World Model

Yao et al. ICLR 2023
Xu et al. Arxiv 2023
Hao et al. EMNLP 2023
Yao et al. NeurIPS 2023
Shinn et al. NeurIPS 2023
Zhou et al. ICML 2024
Sel et al. ICML 2024
Besta et al. AAAI 2024



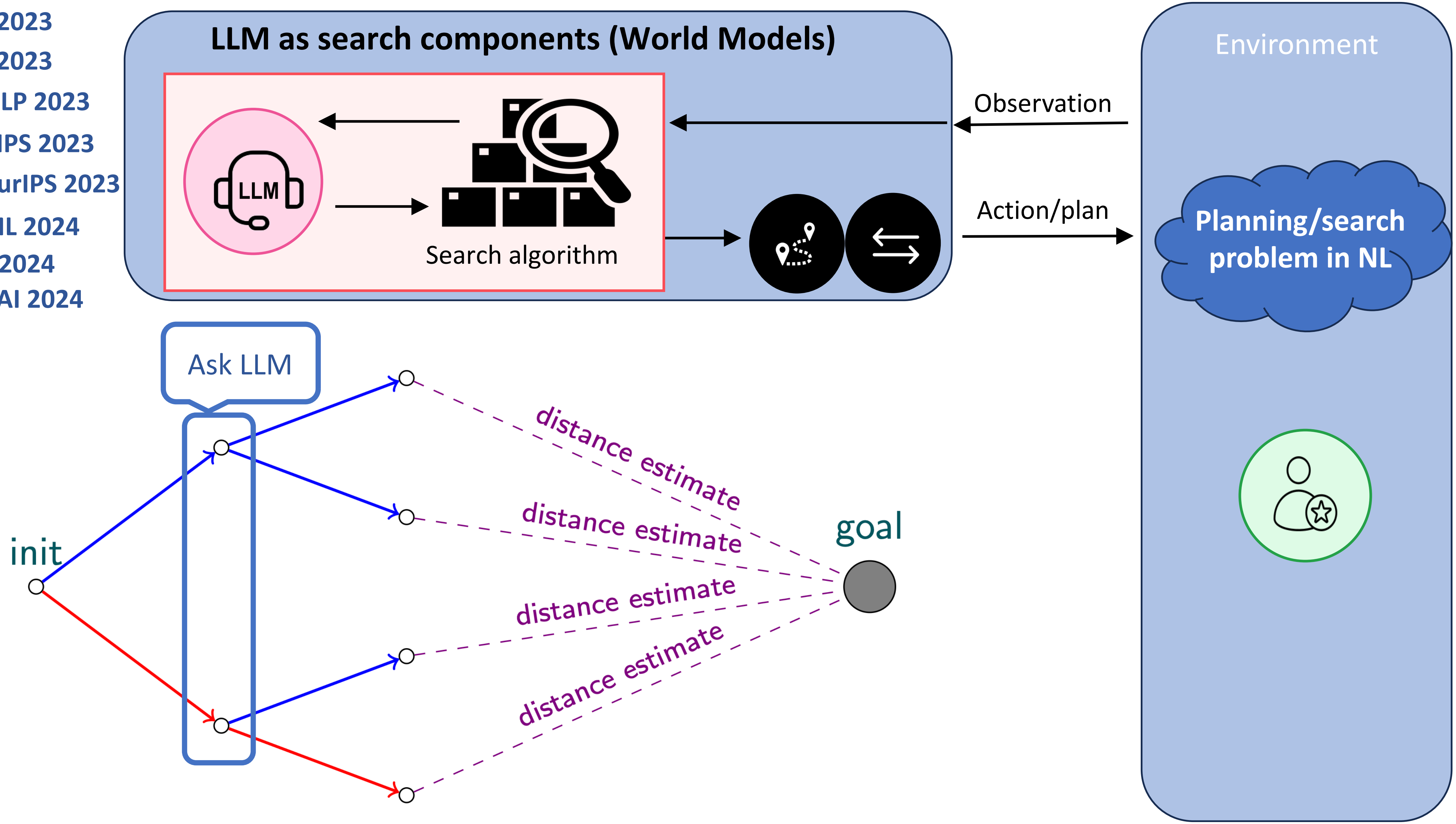
LLM as a World Model

Yao et al. ICLR 2023
Xu et al. Arxiv 2023
Hao et al. EMNLP 2023
Yao et al. NeurIPS 2023
Shinn et al. NeurIPS 2023
Zhou et al. ICML 2024
Sel et al. ICML 2024
Besta et al. AAAI 2024



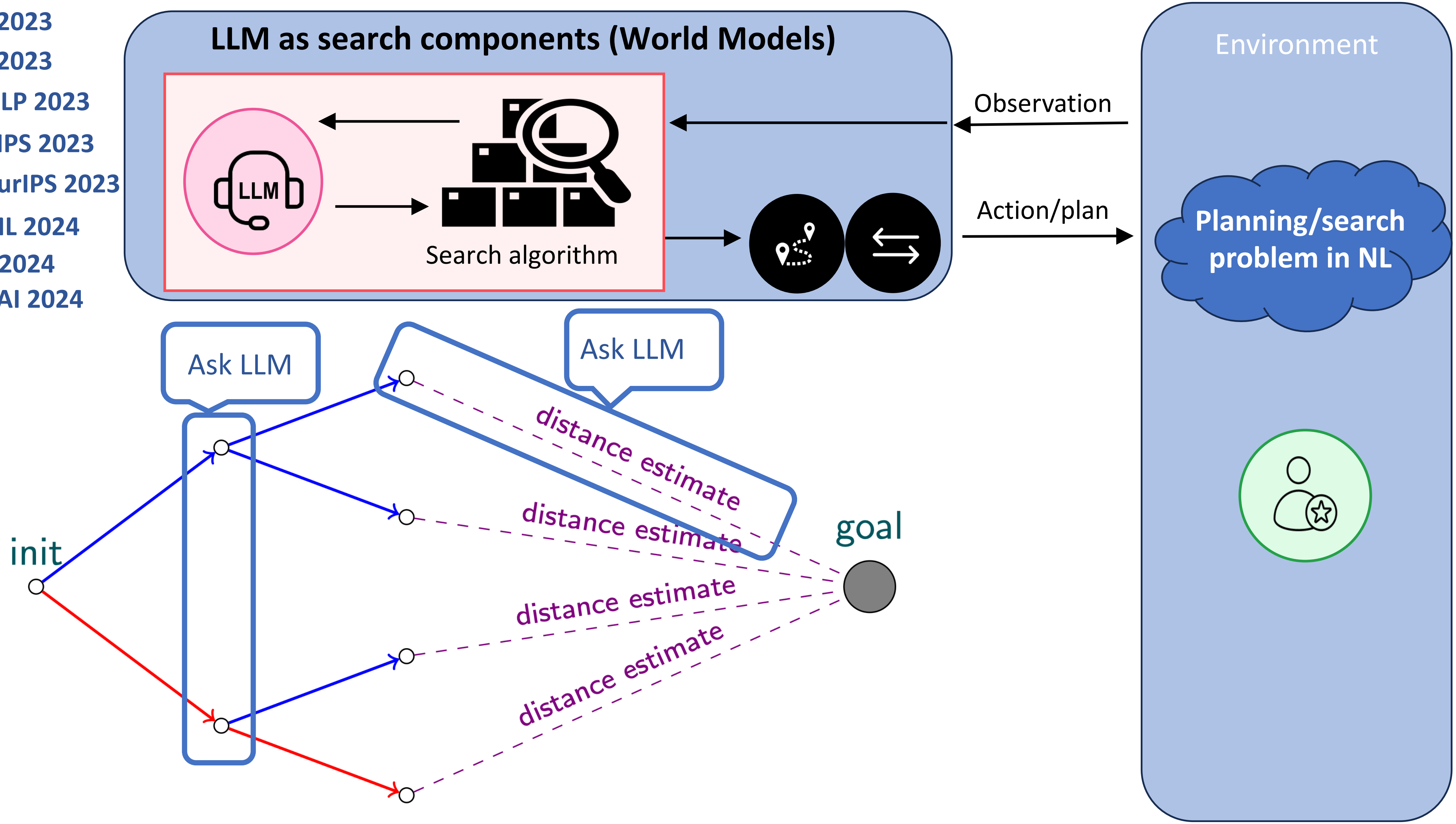
LLM as a World Model

Yao et al. ICLR 2023
Xu et al. Arxiv 2023
Hao et al. EMNLP 2023
Yao et al. NeurIPS 2023
Shinn et al. NeurIPS 2023
Zhou et al. ICML 2024
Sel et al. ICML 2024
Besta et al. AAAI 2024



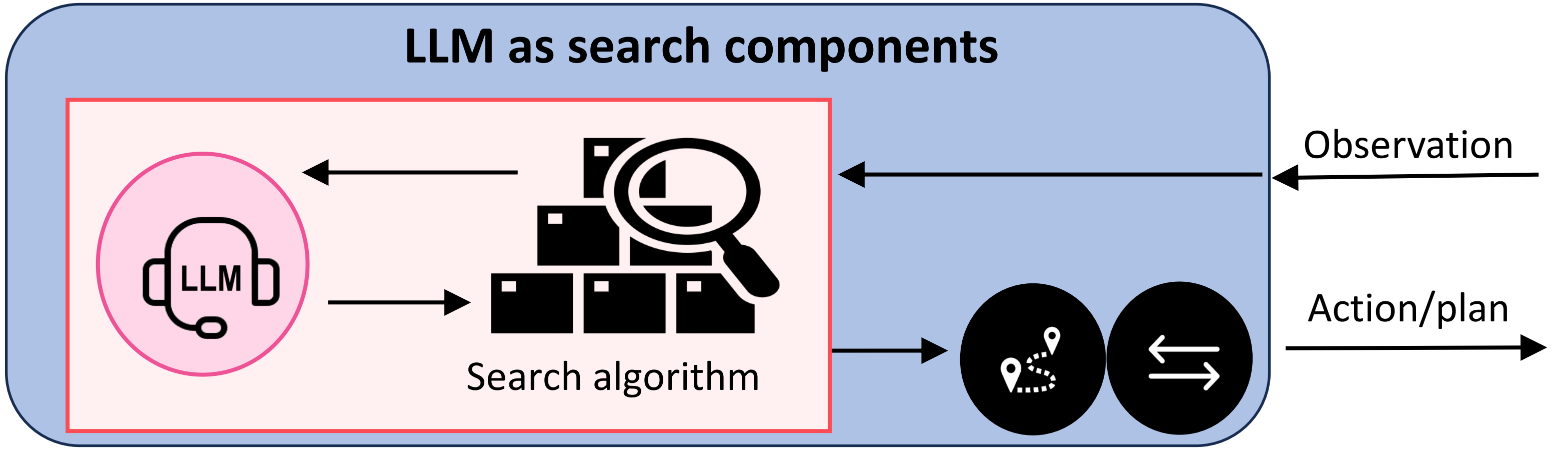
LLM as a World Model

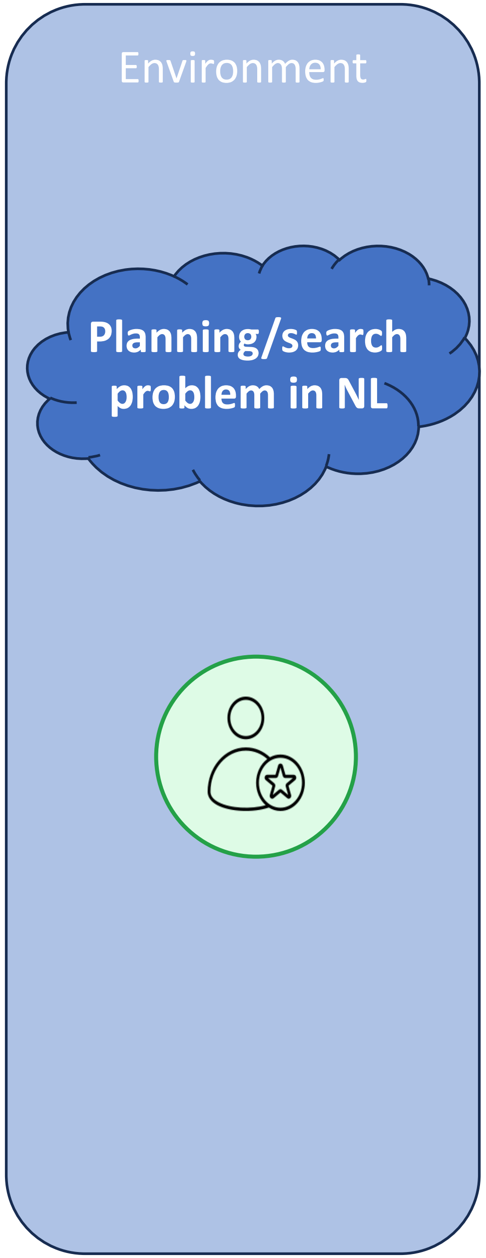
Yao et al. ICLR 2023
Xu et al. Arxiv 2023
Hao et al. EMNLP 2023
Yao et al. NeurIPS 2023
Shinn et al. NeurIPS 2023
Zhou et al. ICML 2024
Sel et al. ICML 2024
Besta et al. AAAI 2024

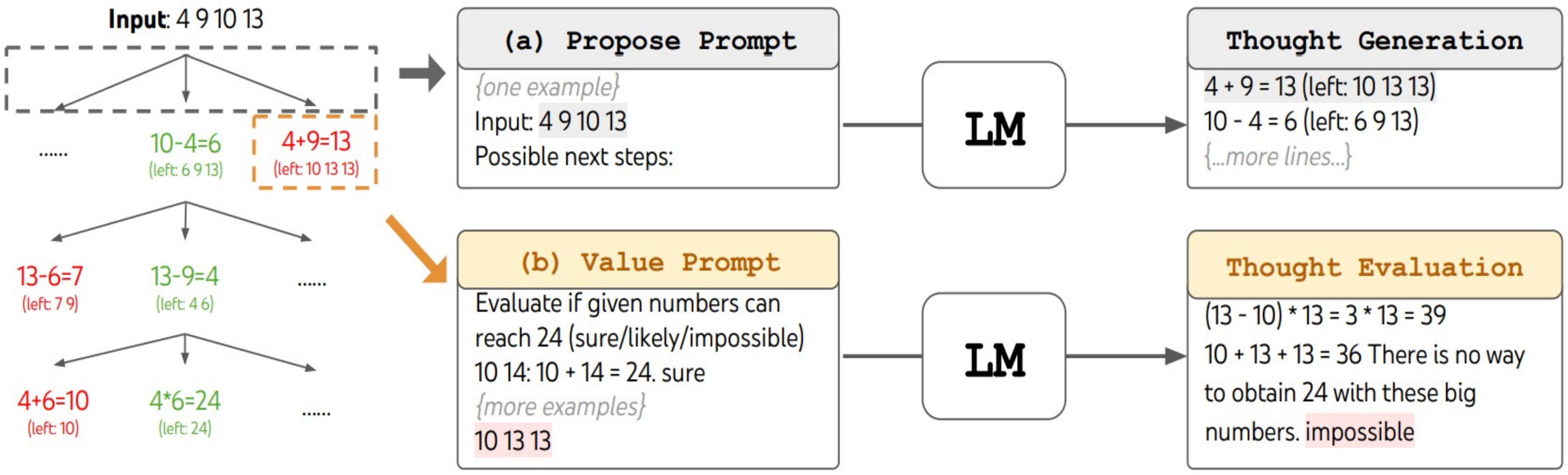


LLM as a World Model

Yao et al. ICLR 2023
Xu et al. Arxiv 2023
Hao et al. EMNLP 2023
Yao et al. NeurIPS 2023
Shinn et al. NeurIPS 2023
Zhou et al. ICML 2024
Sel et al. ICML 2024
Besta et al. AAAI 2024



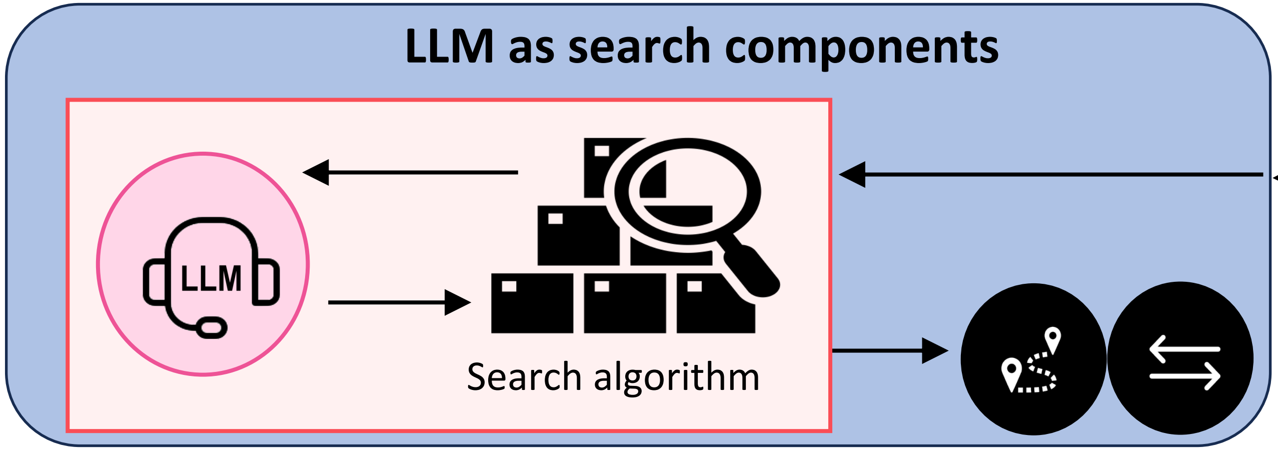


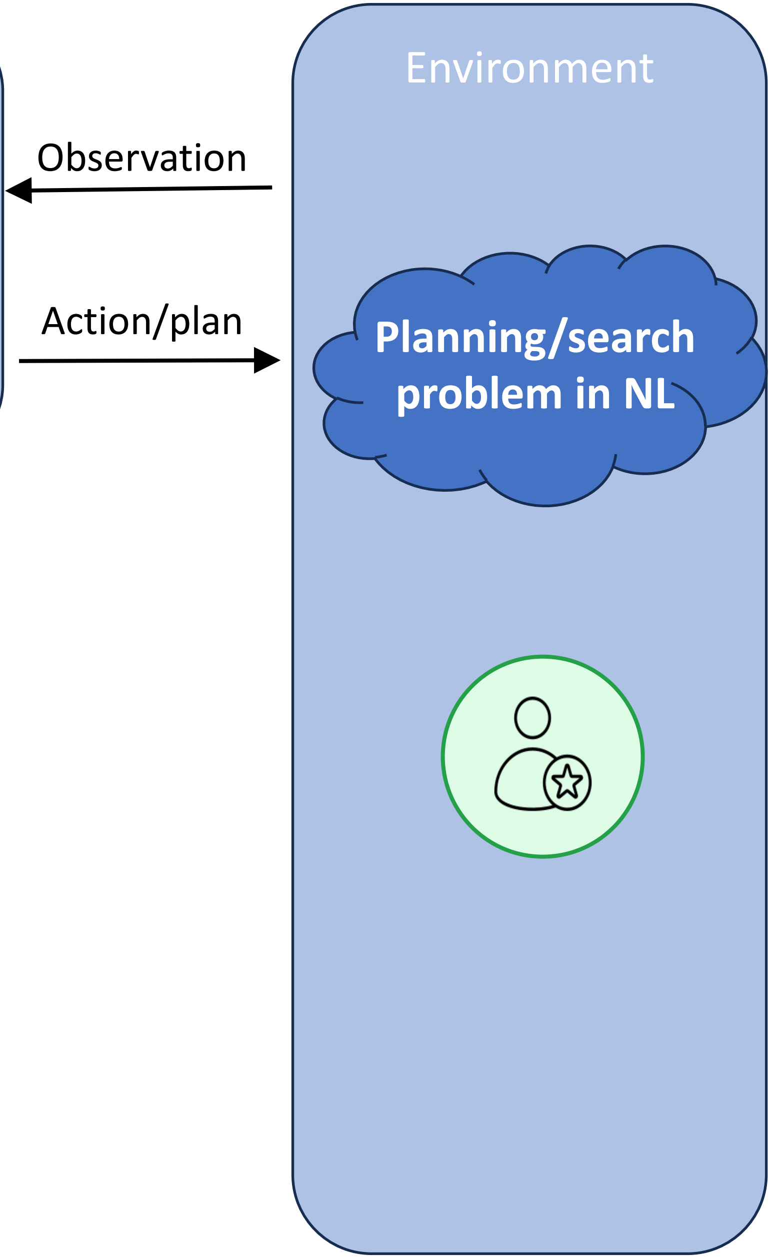


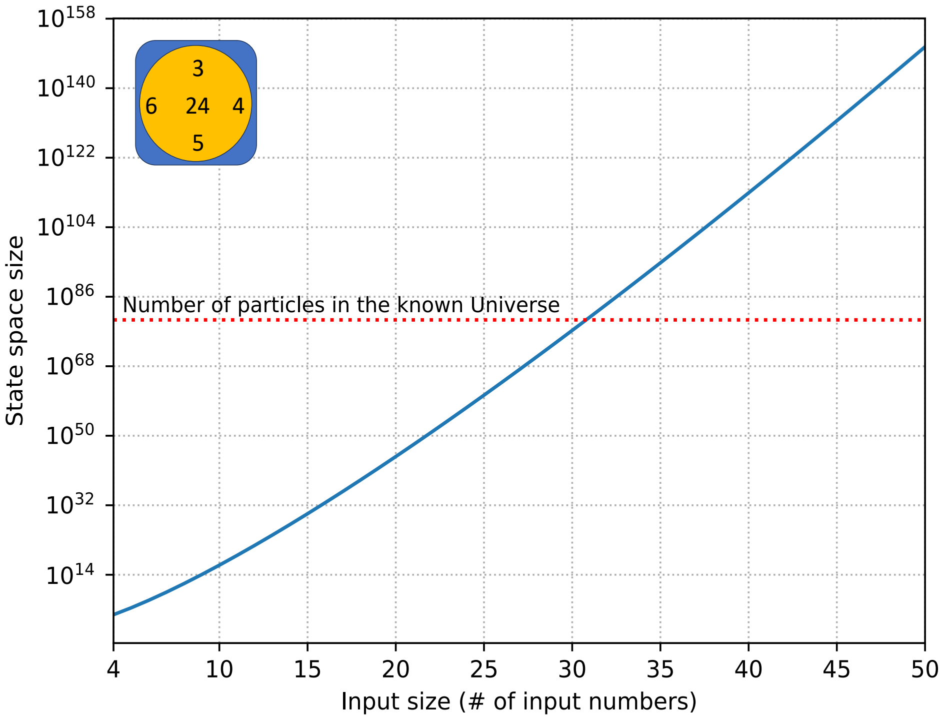
From: Yao et. al., NeurIPS 2023, Tree of Thoughts: Deliberate Problem Solving with Large Language Models

LLM as a World Model

Yao et al. ICLR 2023
Xu et al. Arxiv 2023
Hao et al. EMNLP 2023
Yao et al. NeurIPS 2023
Shinn et al. NeurIPS 2023
Zhou et al. ICML 2024
Sel et al. ICML 2024
Besta et al. AAAI 2024

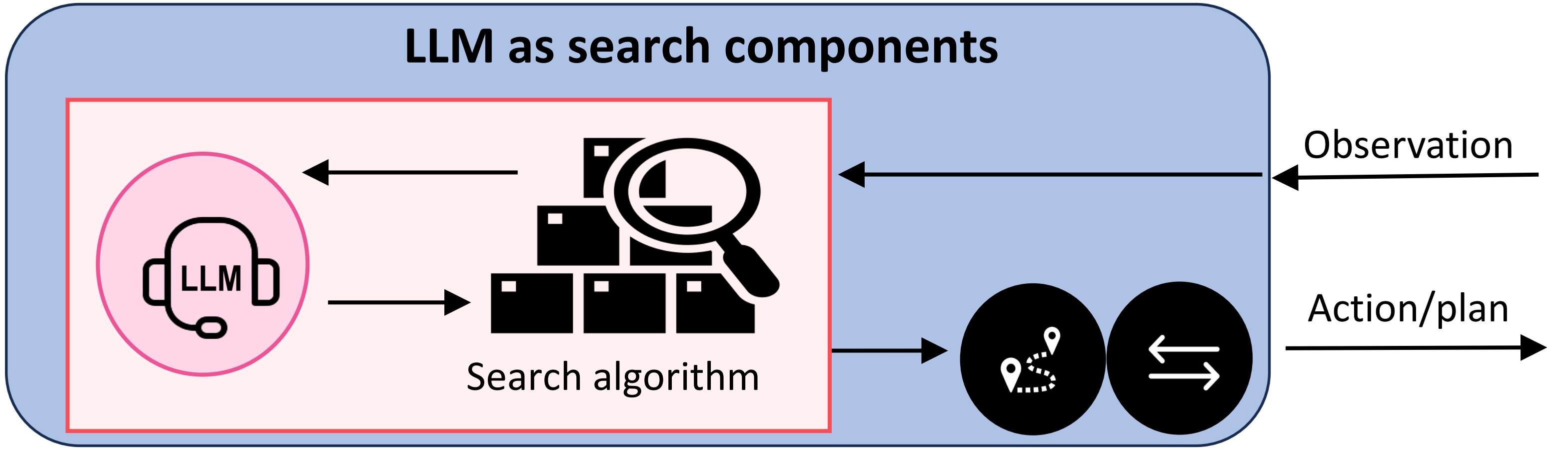


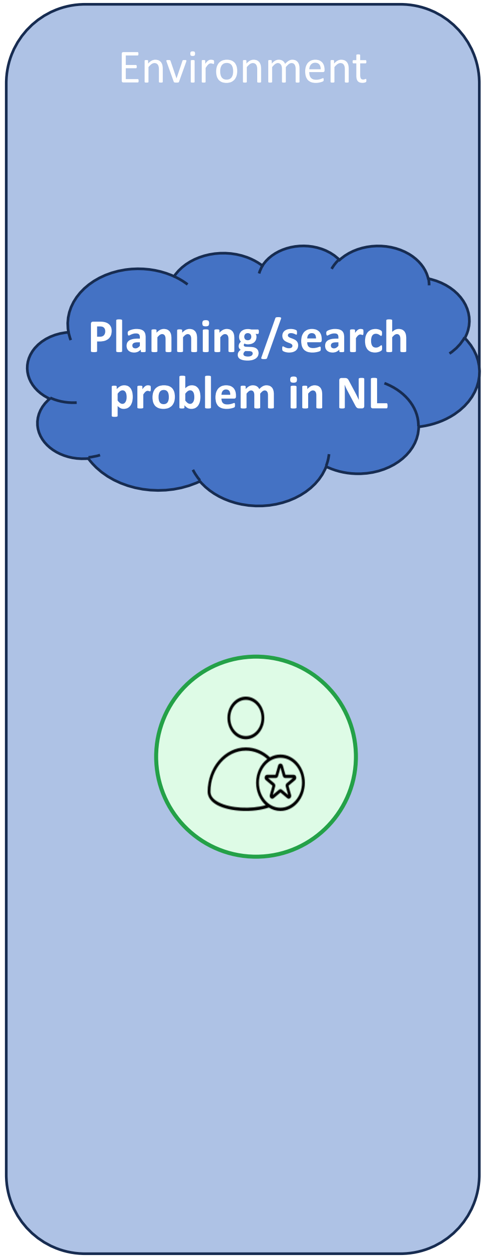




LLM as a World Model

Yao et al. ICLR 2023
Xu et al. Arxiv 2023
Hao et al. EMNLP 2023
Yao et al. NeurIPS 2023
Shinn et al. NeurIPS 2023
Zhou et al. ICML 2024
Sel et al. ICML 2024
Besta et al. AAAI 2024





Algorithm 1 ToT-BFS($x, p_\theta, G, k, V, T, b$)

Require: Input x, LM p_θ, thought generator $G()$ & size limit k, states evaluator $V()$, step limit T, breadth limit b.

$S_0 \leftarrow \{x\}$
for $t = 1, \cdots, T$ **do**
  $S'_t \leftarrow \{[s, z] \mid s \in S_{t-1}, z_t \in \mathrm{G}(p_\theta, s, k)\}$
  $V_t \leftarrow V(p_\theta, S'_t)$
  $S_t \leftarrow \arg\max_{S \subset S'_t, |S|=b} \sum_{s \in S} V_t(s)$
end for
return $G(p_\theta, \arg\max_{s \in S_T} V_T(s), 1)$

From: Yao et. al., NeurIPS 2023, Tree of Thoughts: Deliberate Problem Solving with Large Language Models

What are the properties of these algorithms?

Sound?

What are the properties of these algorithms?

Sound? No!

What are the properties of these algorithms?

Sound? No! If validator exists, we can make it sound

What are the properties of these algorithms?

Sound? No! If validator exists, we can make it sound

Complete?

What are the properties of these algorithms?

Sound?	No!	If validator exists, we can make it sound
Complete?	No!	

What are the properties of these algorithms?

Sound?	No!	If validator exists, we can make it sound
Complete?	No!	
Optimal?		

What are the properties of these algorithms?

Sound?	No!	If validator exists, we can make it sound
Complete?	No!	
Optimal?	No!	

What are the properties of these algorithms?

Sound?	No!	If validator exists, we can make it sound
Complete?	No!	
Optimal?	No!	

Computational complexity?

- Hard to measure. Most expensive operation is LLM call
- Computational effort vs. state space portion explored

What are the properties of these algorithms?

Sound? No! If validator exists, we can make it sound

Complete? No!

Optimal? No!

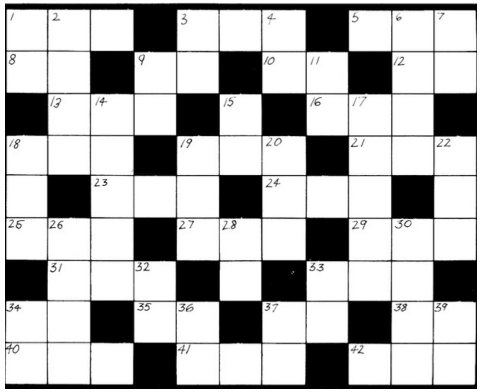
Computational complexity?

- Hard to measure. Most expensive operation is LLM call
- Computational effort vs. state space portion explored

		24Game		Crossword		BlocksWorld		PrOntoQA	
Approach	Complexity	States	Calls	States	Calls	States	Calls	States	Calls
IO	O(D)	0.02%	1362	4e-9%	20	0.5%	502	4%	4000
CoT	O(D)	0.02%	1362	4e-9%	20	0.5%	502	4%	4000
ReAct	O(LD)	0.07%	4086	4e-8%	200	7.8%	8032	24.6%	24K
ReWOO	O(LD)	0.07%	4086	4e-8%	200	7.8%	8032	24.6%	24K
RAP	O(TbLD)	3.3%	245K	2e-6%	12K	388%	482K	1229%	1.44M
ToT	O(bmLD)	1.6%	102K	1e-6%	5K	194%	201K	615%	600K
GoT	O(bLD)	0.3%	20K	2e-7%	1K	39%	40K	122%	120K
Reflection	O(LTD)	0.7%	68K	4e-7%	2.4K	77.6%	90K	245%	320K
LATS	O(TbLD)	3.3%	286K	2e-6%	14K	388%	562K	1229%	1.68M

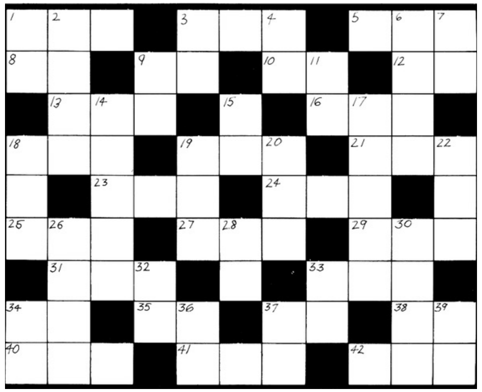
Katz et al, NeurIPS 2024, **Thought of Search: Planning with Language Models Through The Lens of Efficiency**

Why did it work in the first place?



Why did it work in the first place?

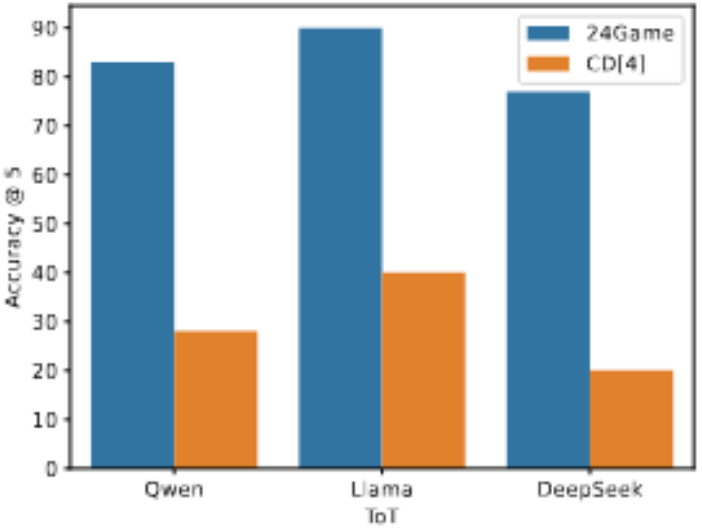
- Tested on data scraped from the internet
    - Data contamination concerns
- No external validator, lenient success criteria

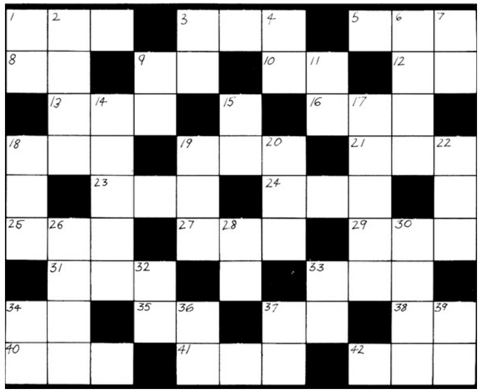


Why did it work in the first place?

- Tested on data scraped from the internet
    - Data contamination concerns
- No external validator, lenient success criteria

	Model	IO 24Game	IO CD[4]	CoT 24Game	CoT CD[4]	ToT 24Game	ToT CD[4]
acc@5	Qwen	6	2	8	2	83	28
	Llama	7	2	32	7	90	40
	DeepSeek	38	5	48	13	77	20
mean	Qwen	2	1	2	0	47	9
	Llama	1	0	9	1	48	12
	DeepSeek	10	1	18	4	28	4



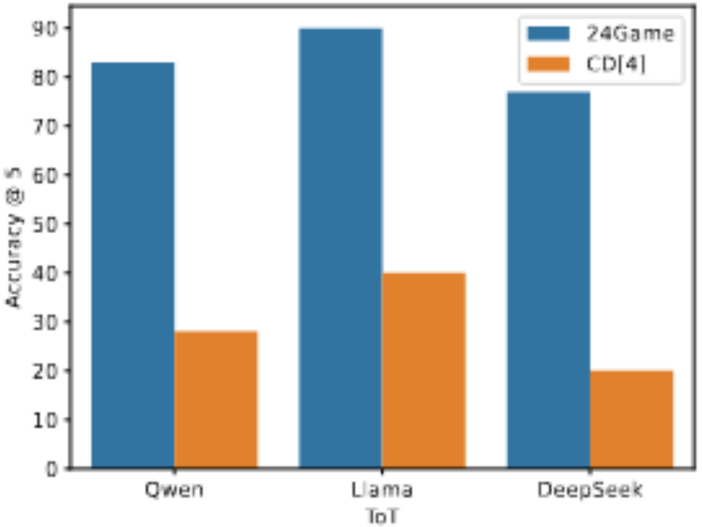


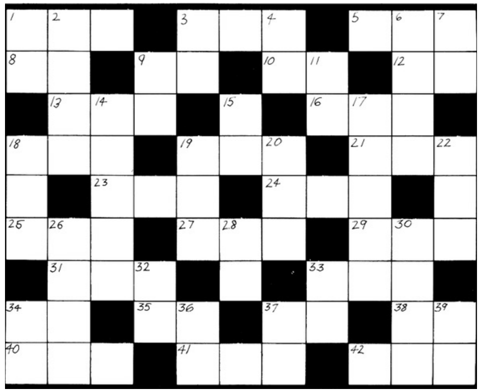
Katz et al, ICLR 2026 Workshop World Models, **Seemingly Simple Planning Problems are Computationally Challenging: The Countdown Game**

Why did it work in the first place?

- Tested on data scraped from the internet
  - Data contamination concerns
- No external validator, lenient success criteria

	Model	IO 24Game	IO CD[4]	CoT 24Game	CoT CD[4]	ToT 24Game	ToT CD[4]
acc@5	Qwen	6	2	8	2	83	28
acc@5	Llama	7	2	32	7	90	40
acc@5	DeepSeek	38	5	48	13	77	20
mean	Qwen	2	1	2	0	47	9
mean	Llama	1	0	9	1	48	12
mean	DeepSeek	10	1	18	4	28	4





Katz et al, ICLR 2026 Workshop World Models, **Seemingly Simple Planning Problems are Computationally Challenging: The Countdown Game**

Conclusions:

- Must have precise definition of solution and sound validators for candidate solutions
- Should have large instance space and dynamic generation procedure to avoid memorization concerns.

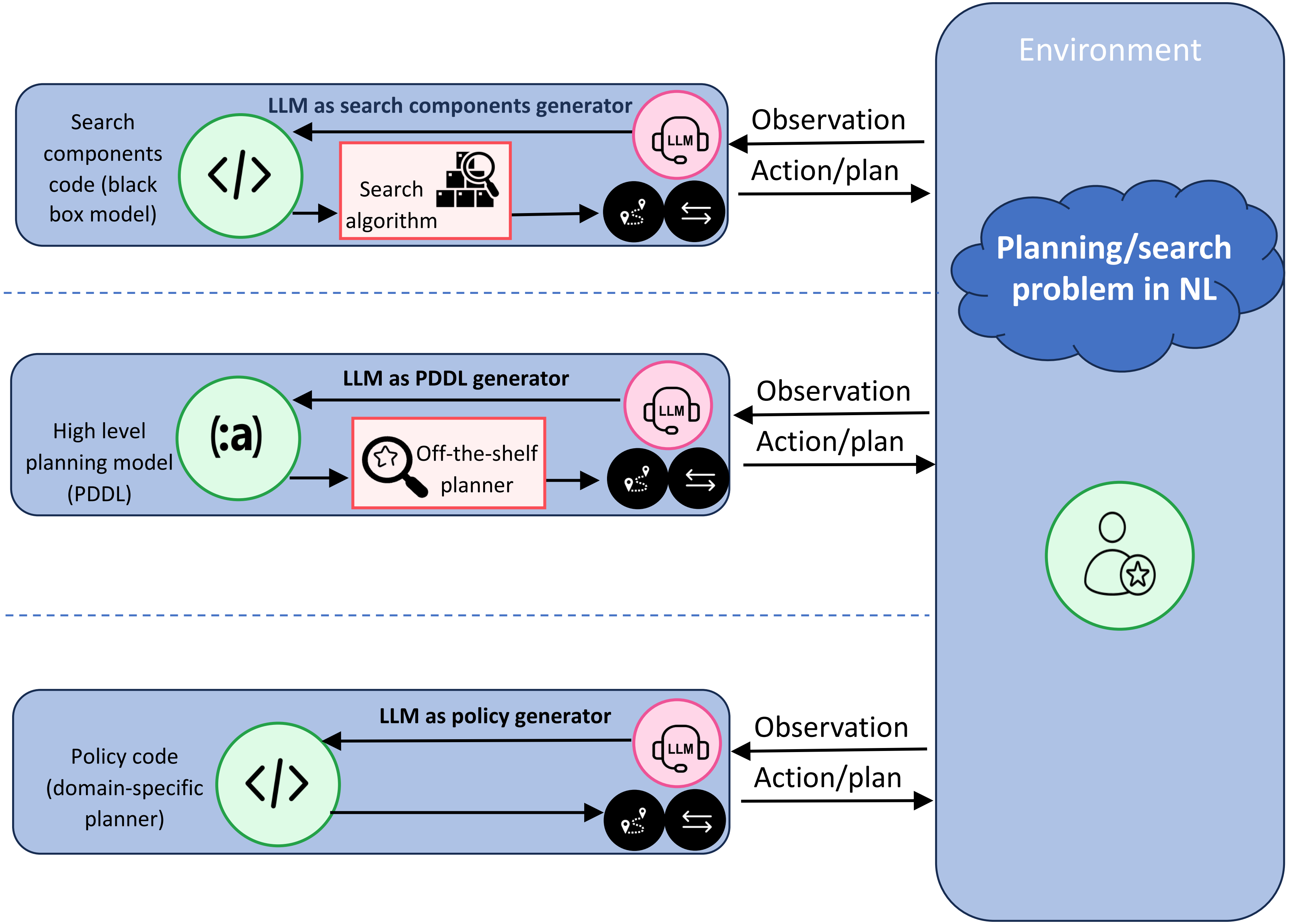
PLANNING IN THE MODERN LLM ERA

Building solvers for NL/PDDL planning problems

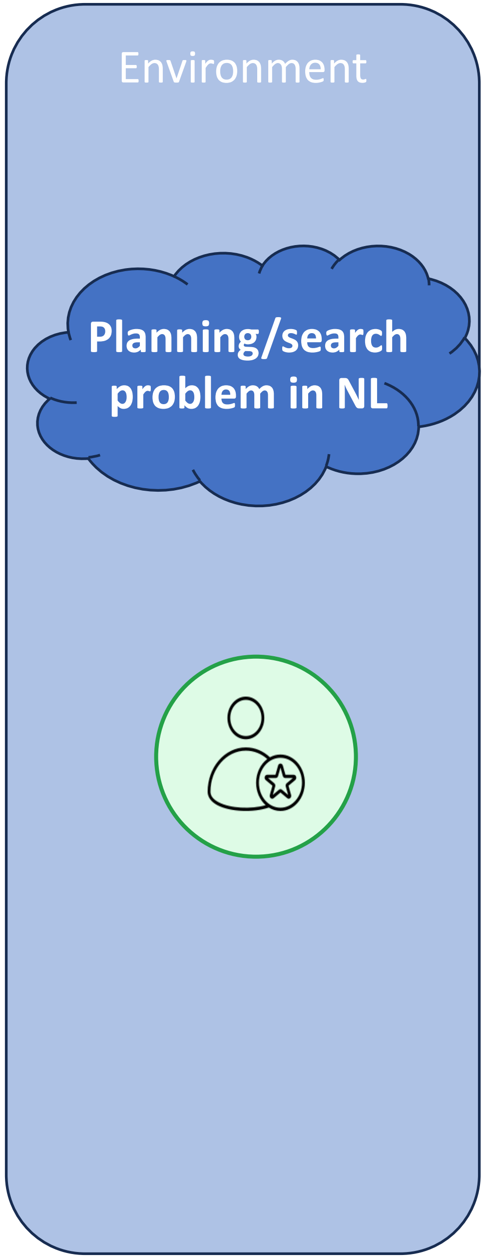
Katz et al, NeurIPS 2024
Cao et al, ICAPS 2026
Tuisov et al, ICAPS 2026
Correa et al, NeurIPS 2025

Guan et al, NeurIPS 2023
Gestrin et al, HAXP@ICAPS 2024
Oswald et al, ICAPS 2024
Huang et al, AAAI 2025
Tantakoun et al, ACL Findings 2025
Oswald et al, WM@ICLR 2026

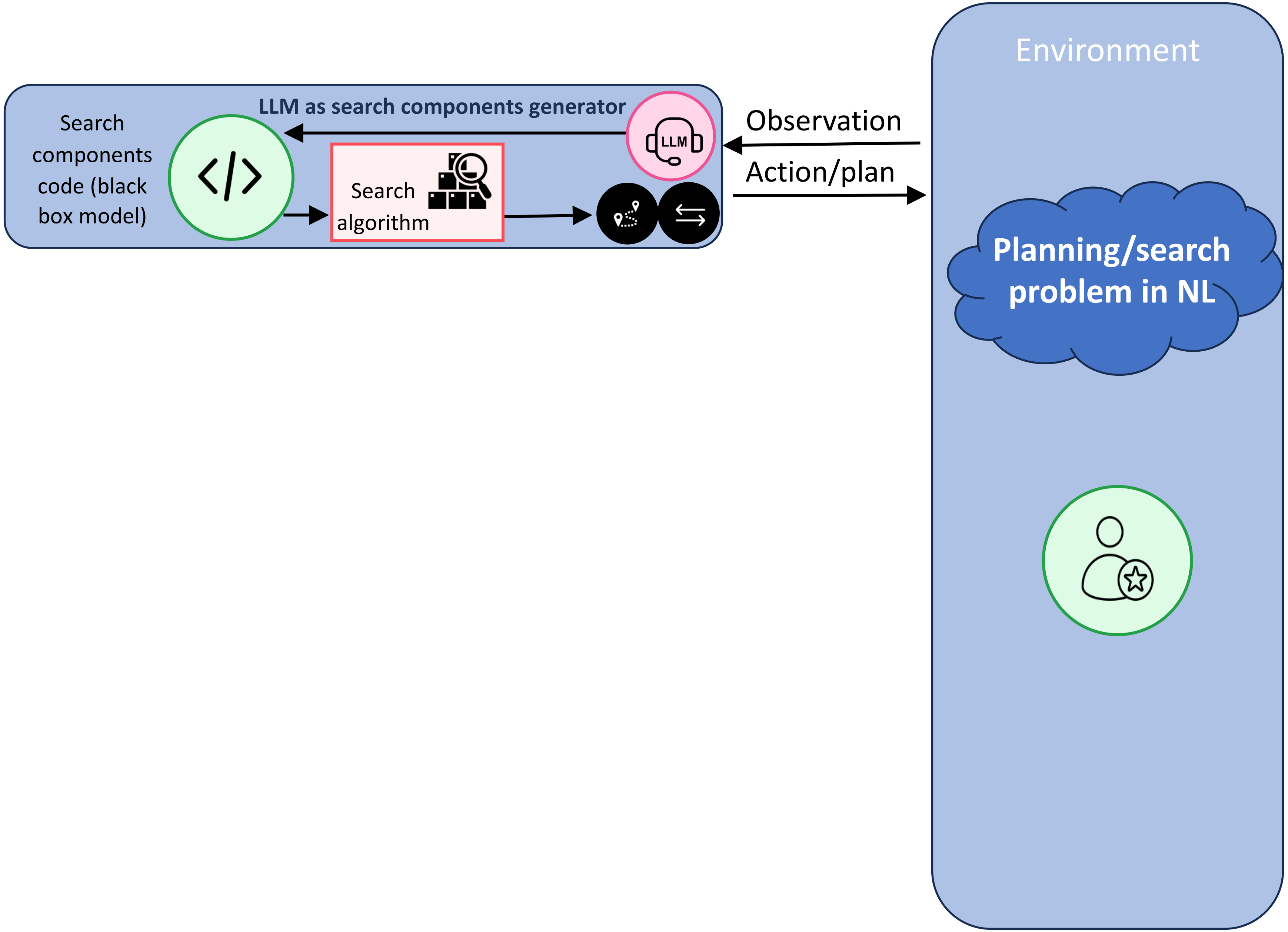
Silver et al, AAAI 2024
Hodel, BSc Thesis 2024
Stein et al, ICAPS 2026
Sohrabi et al, LM4Plan@ICAPS 2026



World Models generation

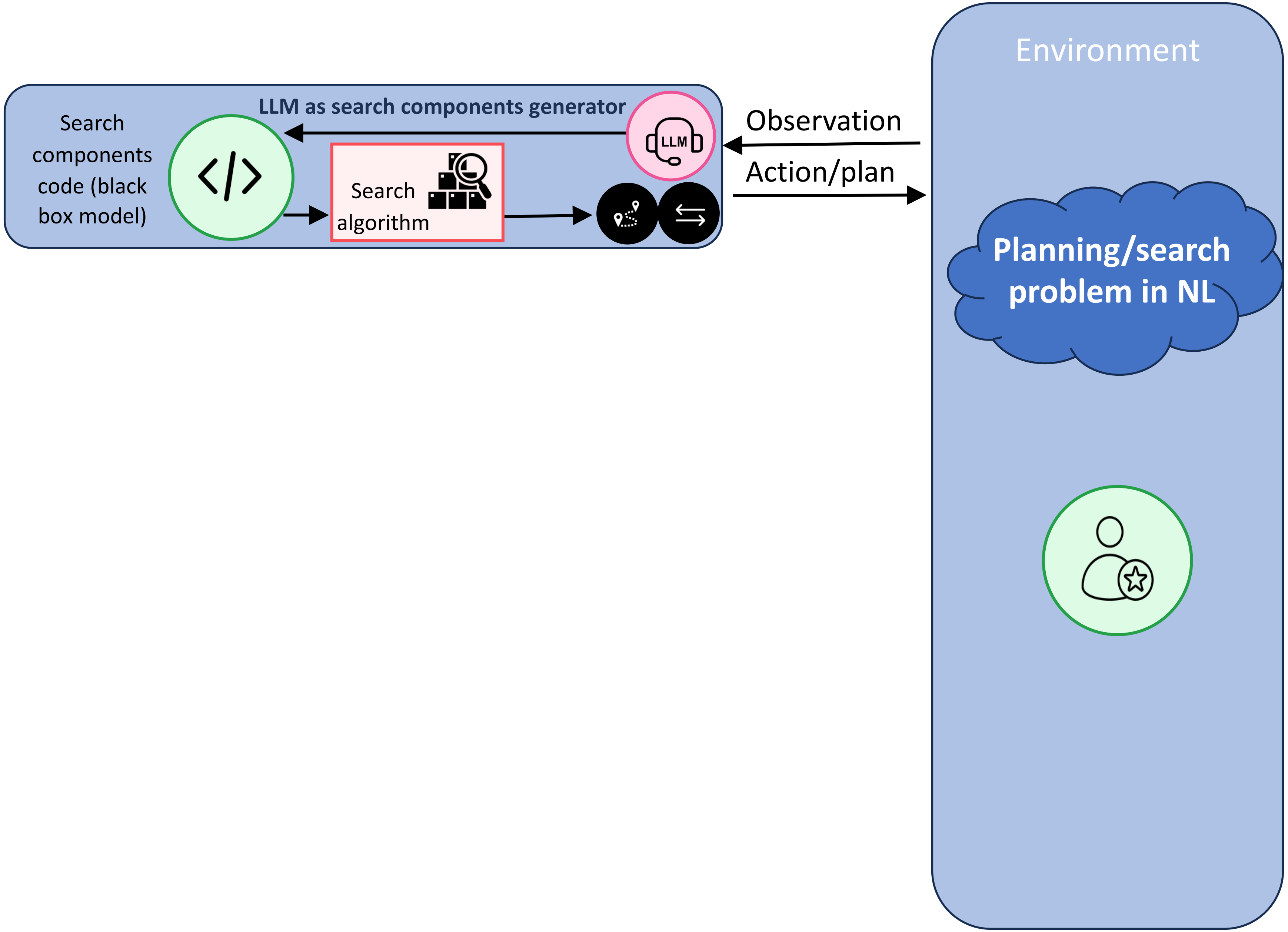


World Models generation



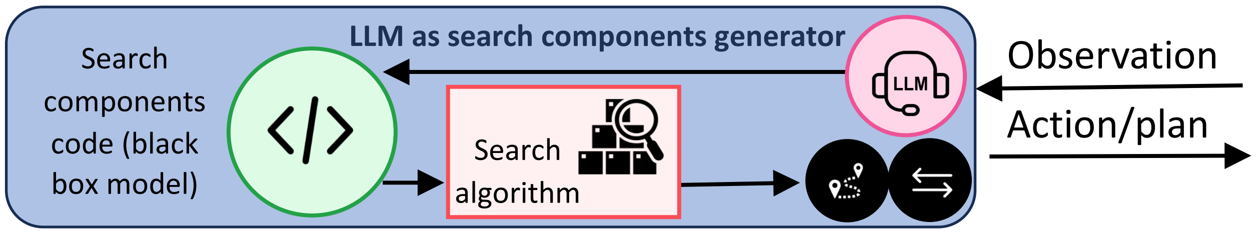
World Models generation

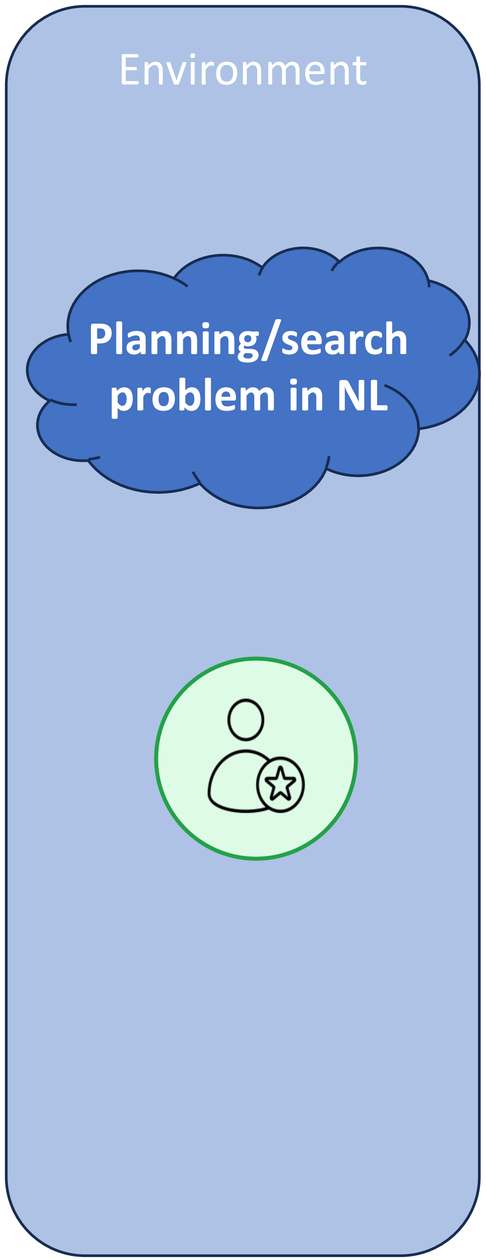
Katz et al, NeurIPS 2024
Cao et al, ICAPS 2026
Correa et al, NeurIPS 2025
Tuisov et al, ICAPS 2026

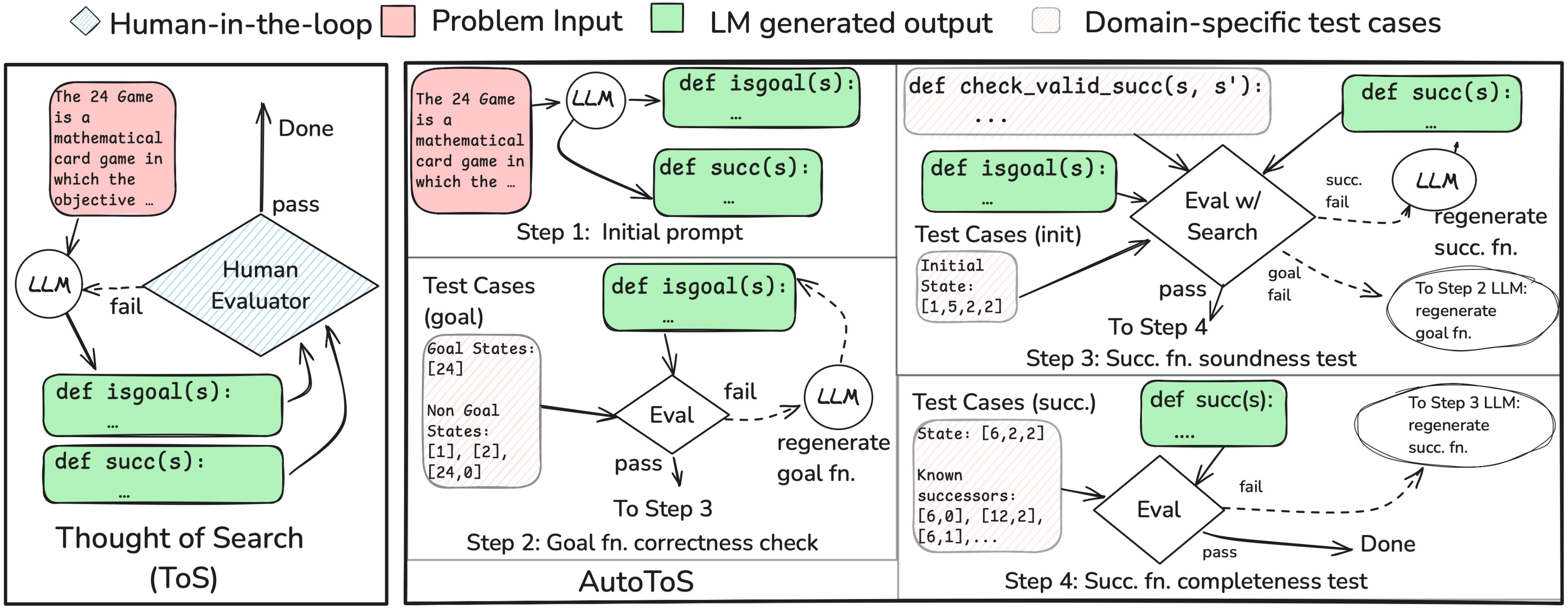


World Models generation

Katz et al, NeurIPS 2024
Cao et al, ICAPS 2026
Correa et al, NeurIPS 2025
Tuisov et al, ICAPS 2026

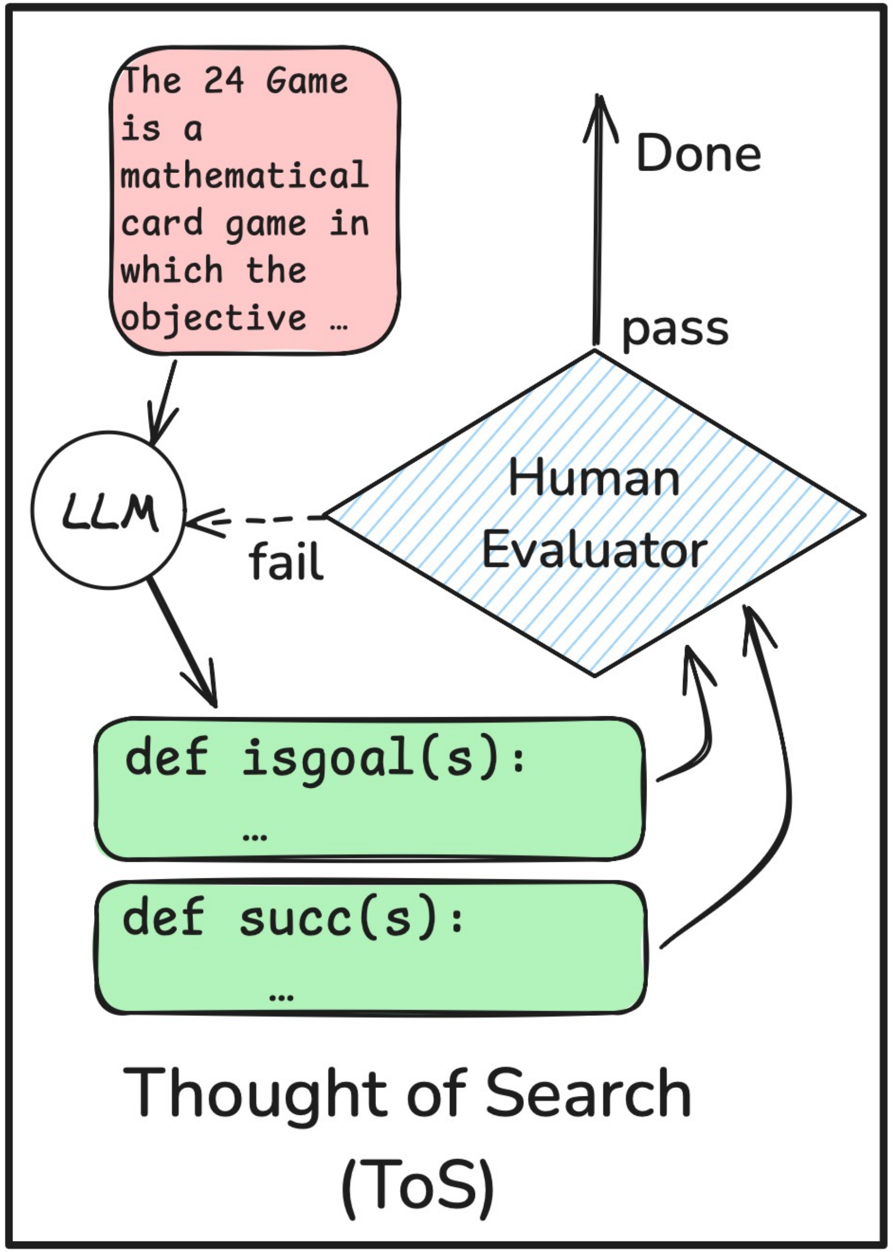






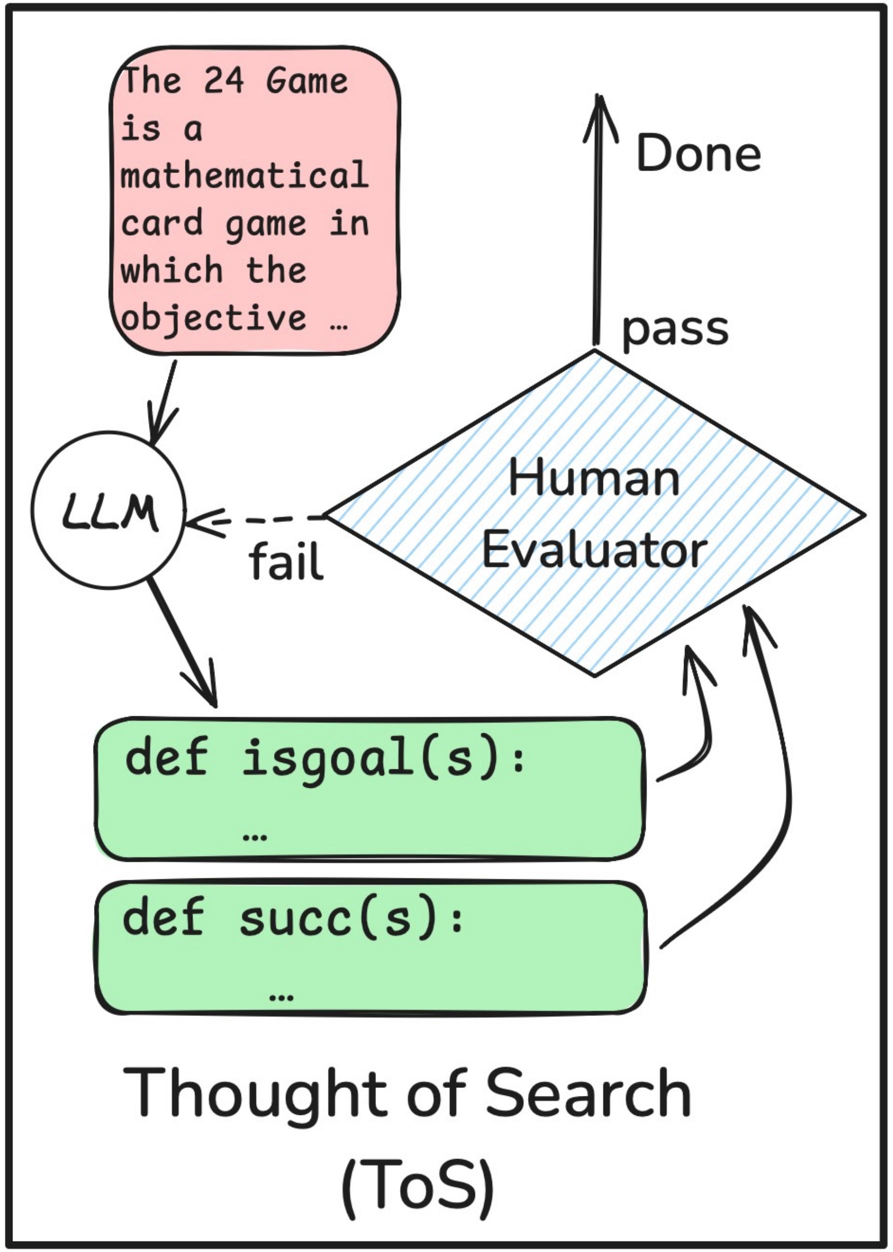
Thought of Search (ToS)

Katz et al, NeurIPS 2024



Thought of Search (ToS)

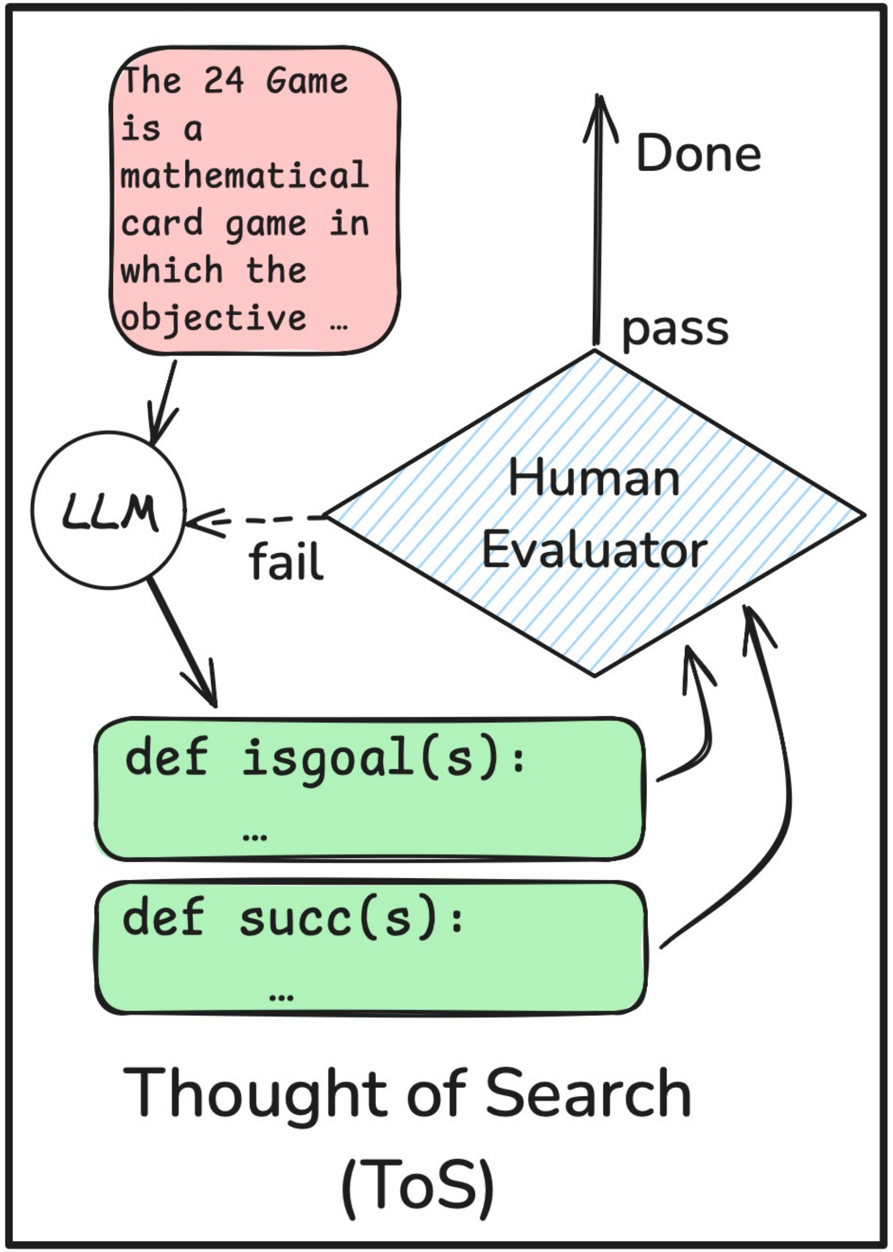
Katz et al, NeurIPS 2024



Plug the functions into a standard search (BFS/DFS/IDDF…)

Thought of Search (ToS)

Katz et al, NeurIPS 2024

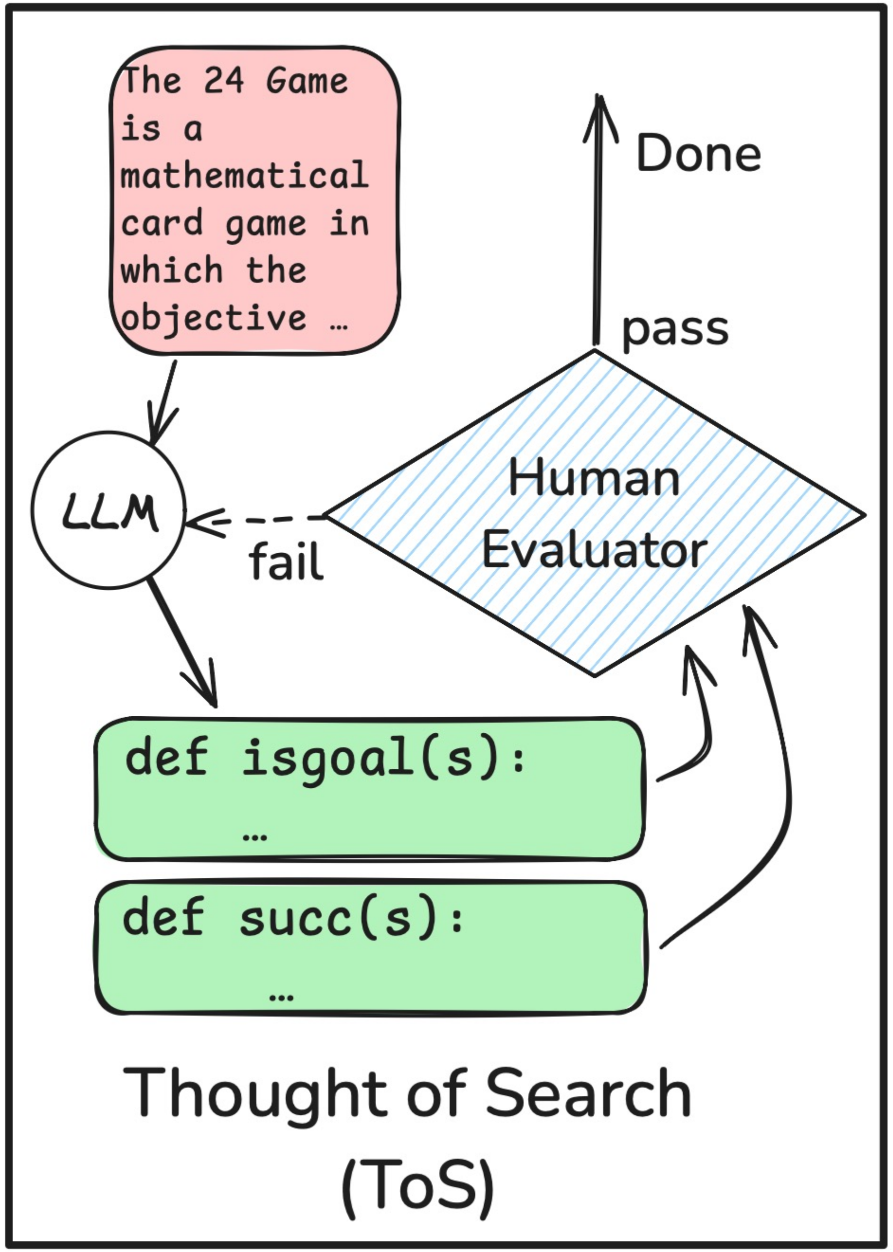


Plug the functions into a standard search (BFS/DFS/IDDF…)

Results?

Thought of Search (ToS)

Katz et al, NeurIPS 2024



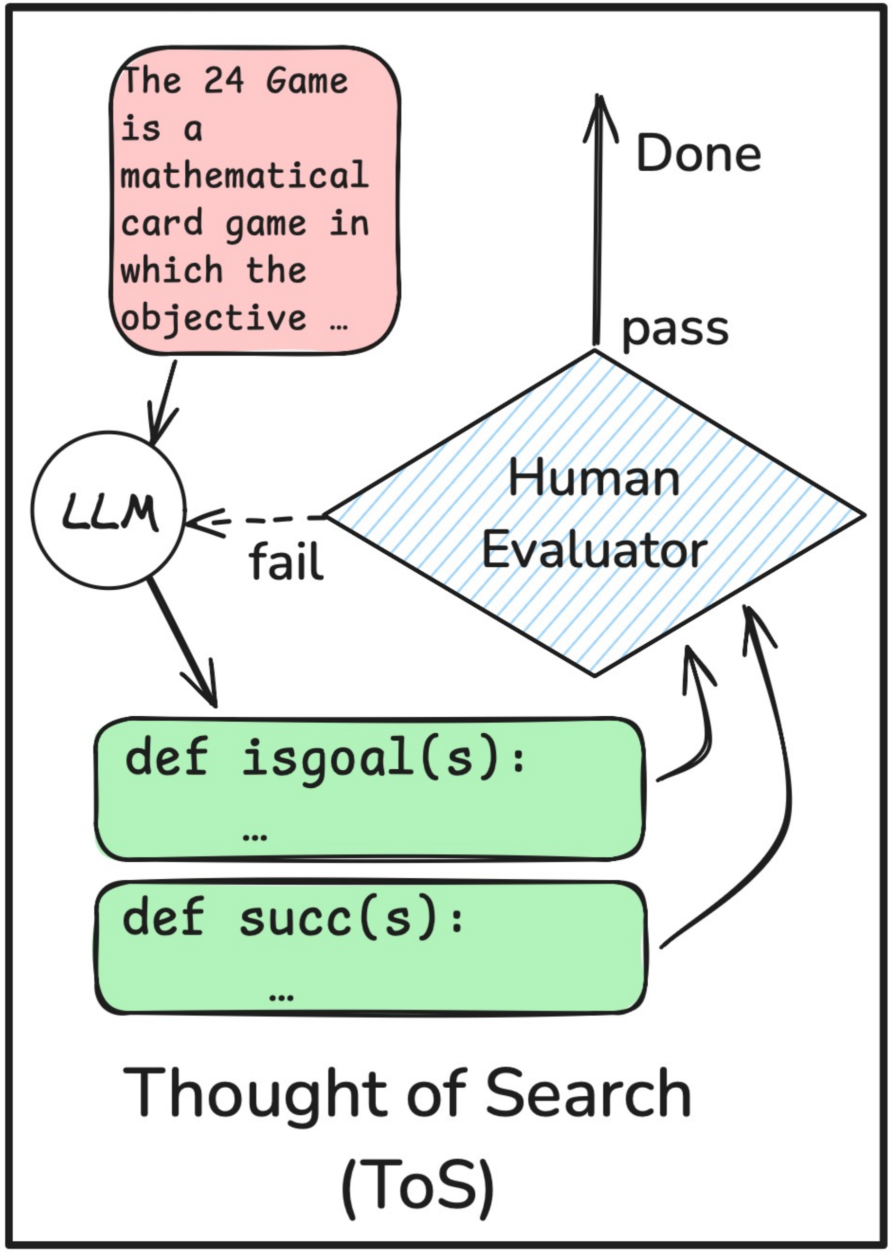
Plug the functions into a standard search (BFS/DFS/IDDF…)

Results?

- **100% accuracy across all tested domains!**
- **Execution time in ms per instance**

Thought of Search (ToS)

Katz et al, NeurIPS 2024



Plug the functions into a standard search (BFS/DFS/IDDF…)

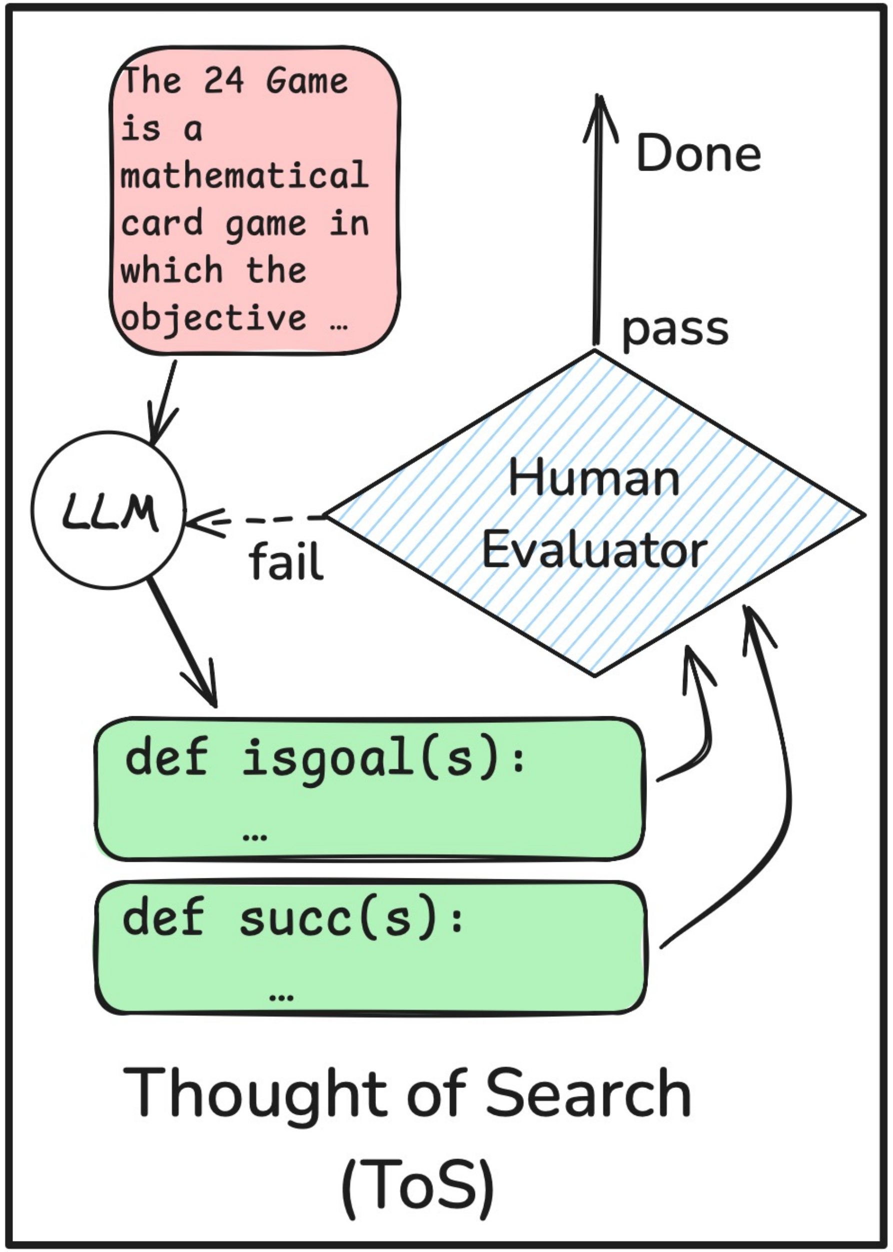
Results?

- **100% accuracy across all tested domains!**
- **Execution time in ms per instance**

Approach	Complexity	24Game States	24Game Calls	Crossword States	Crossword Calls	BlocksWorld States	BlocksWorld Calls	PrOntoQA States	PrOntoQA Calls
IO	O(D)	0.02%	1362	4e-9%	20	0.5%	502	4%	4000
CoT	O(D)	0.02%	1362	4e-9%	20	0.5%	502	4%	4000
ReAct	O(LD)	0.07%	4086	4e-8%	200	7.8%	8032	24.6%	24K
ReWOO	O(LD)	0.07%	4086	4e-8%	200	7.8%	8032	24.6%	24K
RAP	O(TbLD)	3.3%	245K	2e-6%	12K	388%	482K	1229%	1.44M
ToT	O(bmLD)	1.6%	102K	1e-6%	5K	194%	201K	615%	600K
GoT	O(bLD)	0.3%	20K	2e-7%	1K	39%	40K	122%	120K
Reflection	O(LTD)	0.7%	68K	4e-7%	2.4K	77.6%	90K	245%	320K
LATS	O(TbLD)	3.3%	286K	2e-6%	14K	388%	562K	1229%	1.68M
ToS (ours)	O(1)	27.0%	**2.2**	3e-4%	**3.8**	125%	**3.8**	175%	**2.6**

Thought of Search (ToS)

Katz et al, NeurIPS 2024



Plug the functions into a standard search (BFS/DFS/IDDF…)

Results?

- **100% accuracy across all tested domains!**
- **Execution time in ms per instance**

Approach	Complexity	24Game States	24Game Calls	Crossword States	Crossword Calls	BlocksWorld States	BlocksWorld Calls	PrOntoQA States	PrOntoQA Calls
IO	O(D)	0.02%	1362	4e-9%	20	0.5%	502	4%	4000
CoT	O(D)	0.02%	1362	4e-9%	20	0.5%	502	4%	4000
ReAct	O(LD)	0.07%	4086	4e-8%	200	7.8%	8032	24.6%	24K
ReWOO	O(LD)	0.07%	4086	4e-8%	200	7.8%	8032	24.6%	24K
RAP	O(TbLD)	3.3%	245K	2e-6%	12K	388%	482K	1229%	1.44M
ToT	O(bmLD)	1.6%	102K	1e-6%	5K	194%	201K	615%	600K
GoT	O(bLD)	0.3%	20K	2e-7%	1K	39%	40K	122%	120K
Reflection	O(LTD)	0.7%	68K	4e-7%	2.4K	77.6%	90K	245%	320K
LATS	O(TbLD)	3.3%	286K	2e-6%	14K	388%	562K	1229%	1.68M
ToS (ours)	O(1)	27.0%	**2.2**	3e-4%	**3.8**	125%	**3.8**	175%	**2.6**

Limitation: need human in the loop!

Automating Thought of Search (AutoToS)

Cao et al, ICAPS 2026

ssues Jobs **VentureBeat**

Security | Data Infrastructure | Automation | Enterprise A

AutoToS makes LLM planning fast, accurate and inexpensive

Image credit: VentureBeat with DALL-E 3

About 18,000 results (0.18s)

LinkedIn · David Borish
6 reactions · 6 months ago

AI That Fixes Itself: AutoToS and Self-Improving AI

AutoToS (Automated Thought of Search), **a system that allows AI models to self-correct and improve their performance** without human intervention.

MarkTechPost
https://www.marktechpost.com › 2024/08/26 › autotos-...

AutoToS: An Automated Feedback System for Generating ...

Aug 26, 2024 — **AutoToS**: An Automated Feedback System for Generating Sound and Complete Search Components in AI Planning.

Instagram
https://www.instagram.com › angelimarielim

💡 AI Breakthrough Alert! 💡 IBM Research and Cornell University ...

IBM Research and Cornell University introduce **AutoToS**, a system that teaches AI to solve complex planning problems with 100% accuracy — no human supervision ...

newsbreak.com
https://www.newsbreak.com › news › 3574117154887-...

AutoToS: An Automated Feedback System for Generating Sound ...

AutoToS: An Automated Feedback System for Generating Sound and Complete Search Components in AI Planning. marktechpost.com. By Nikhil,. 2024-08-26.

Medium · EveryTechTrend
6 months ago

Automating Thought of Search: The Future of AI Planning with ...

We look at a new approach called AutoToS that **automates the ToS process**, allowing LLMs to generate high-performing search functions without any ...

Medium · Manoj Dadlani
6 months ago

Automating AI Planning: How AutoToS is Revolutionizing ...

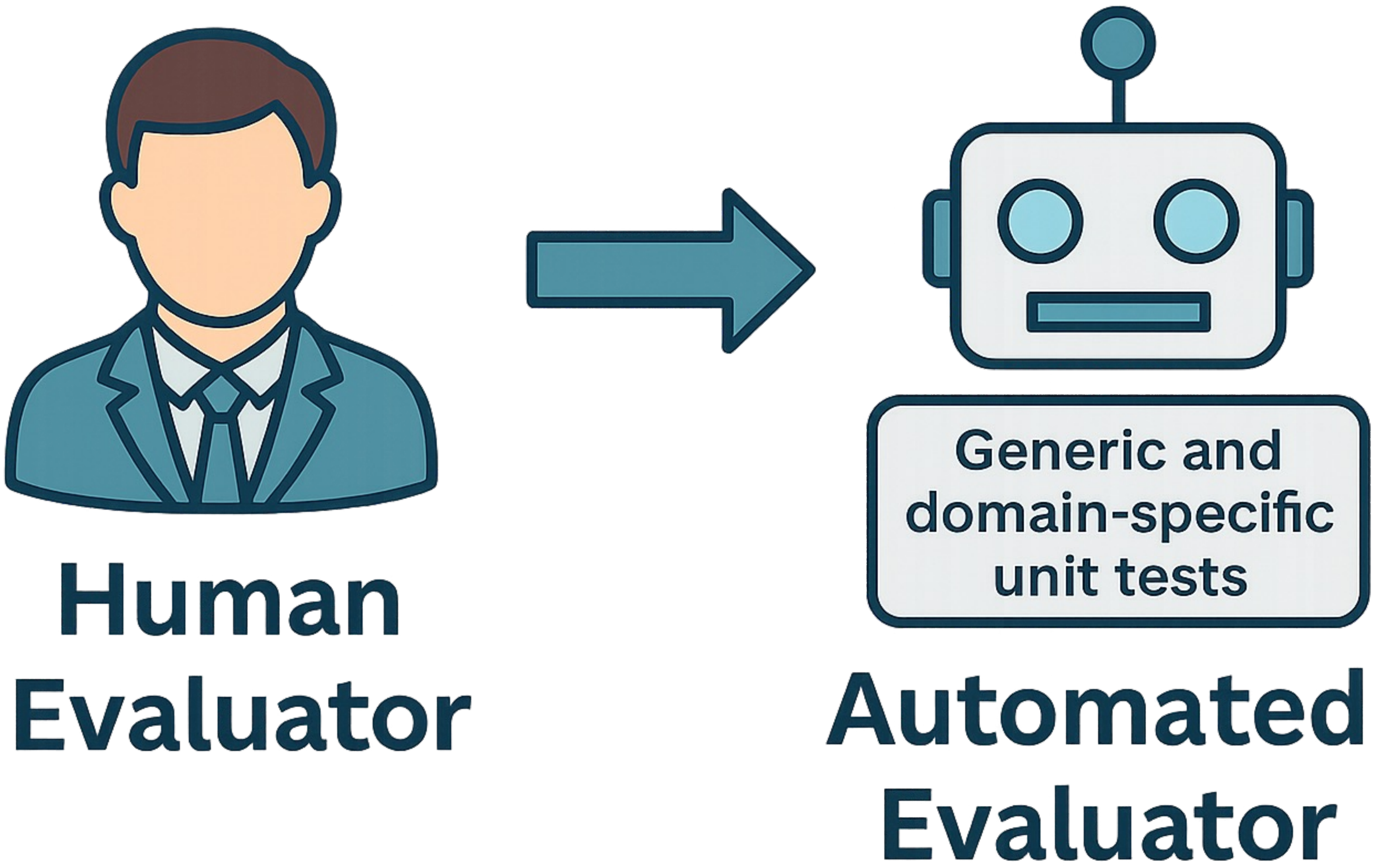
AutoToS operates by first generating an initial prompt that asks the LLM to produce a successor function and a goal test based on a natural ...

YouTube · Douglas Campbell
30+ views · 6 months ago

AutoToS: Revolutionizing AI Planning with Automated Search ...

Automated Logistics Planning: **Use AutoToS to optimize supply chain and logistics operations**, reducing the need for human planners. · Robotics ...

AutoToS: Core Idea



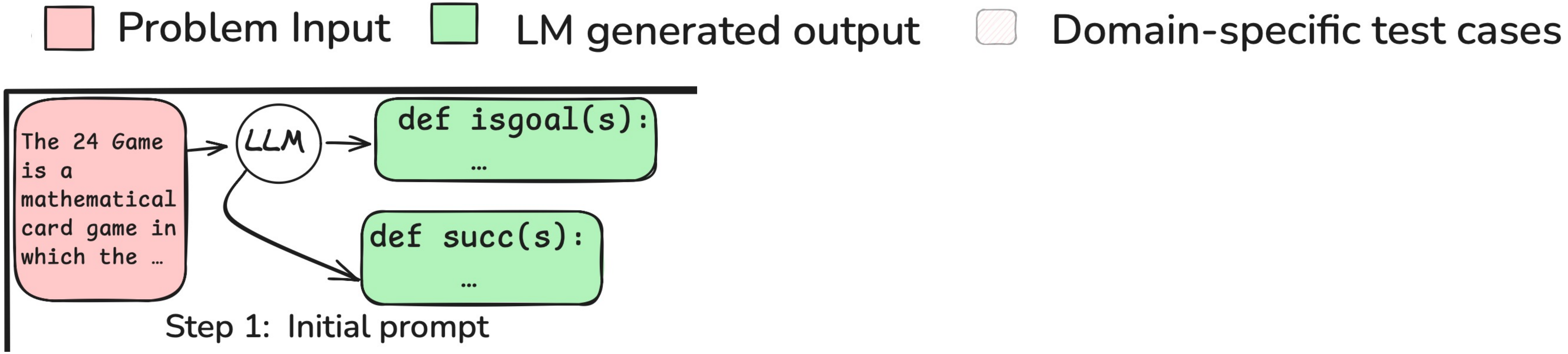
How does it work?

Problem Input

LM generated output

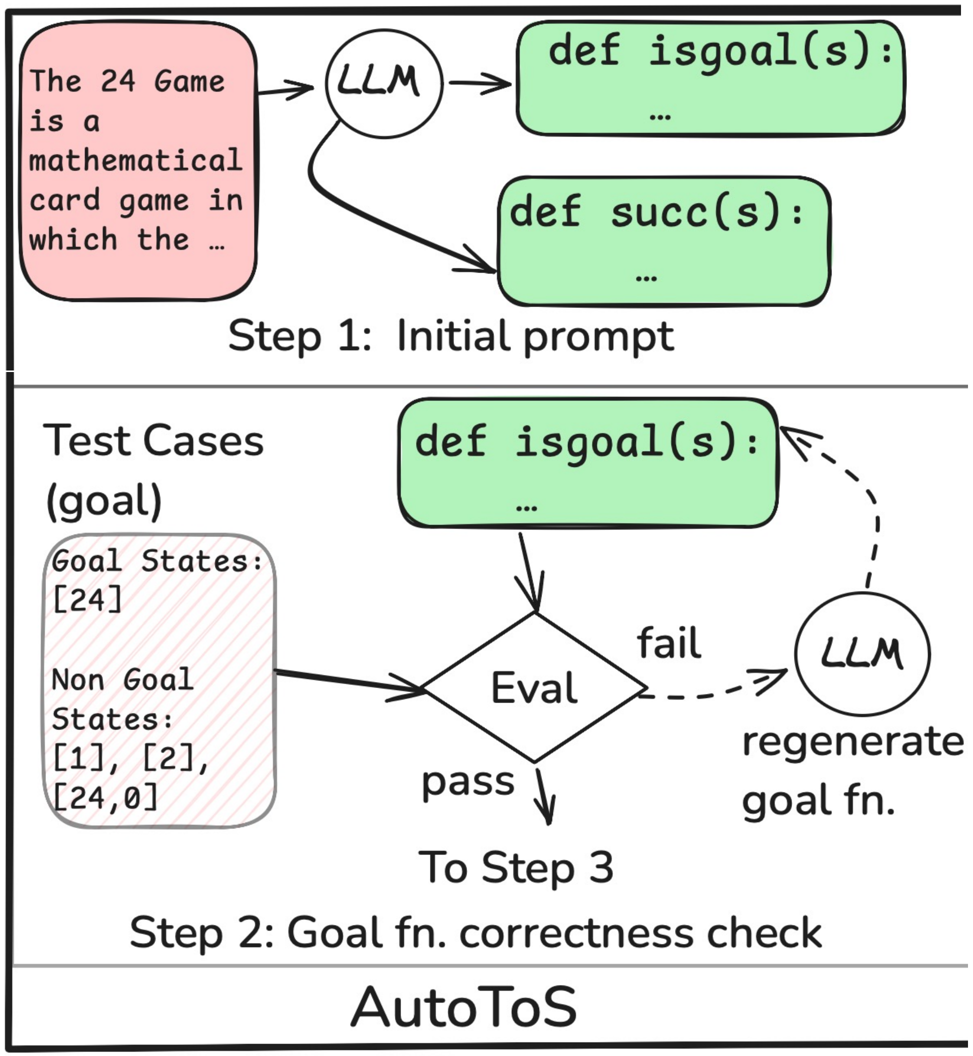
Domain-specific test cases

How does it work?

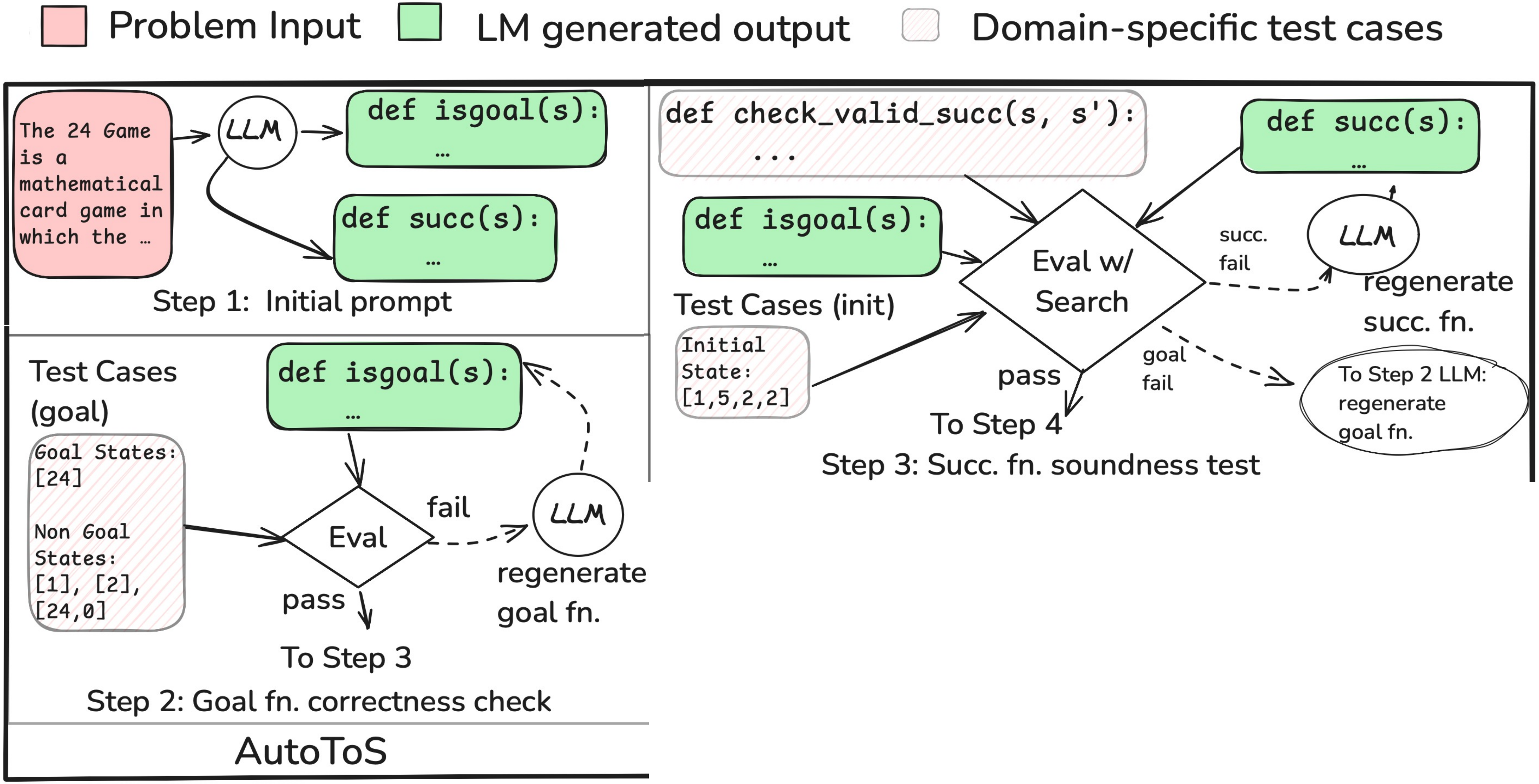


How does it work?

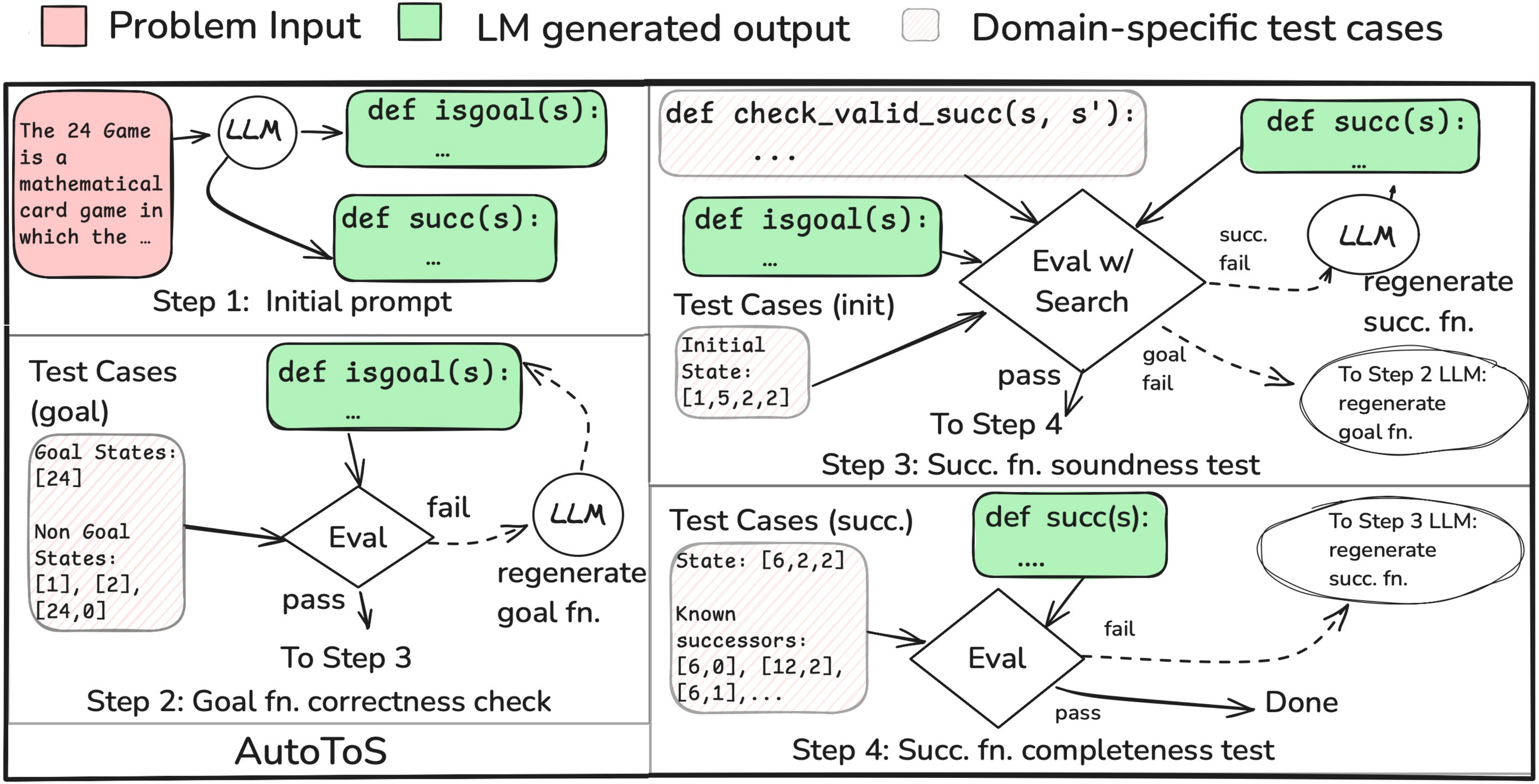
Problem Input LM generated output Domain-specific test cases



How does it work?



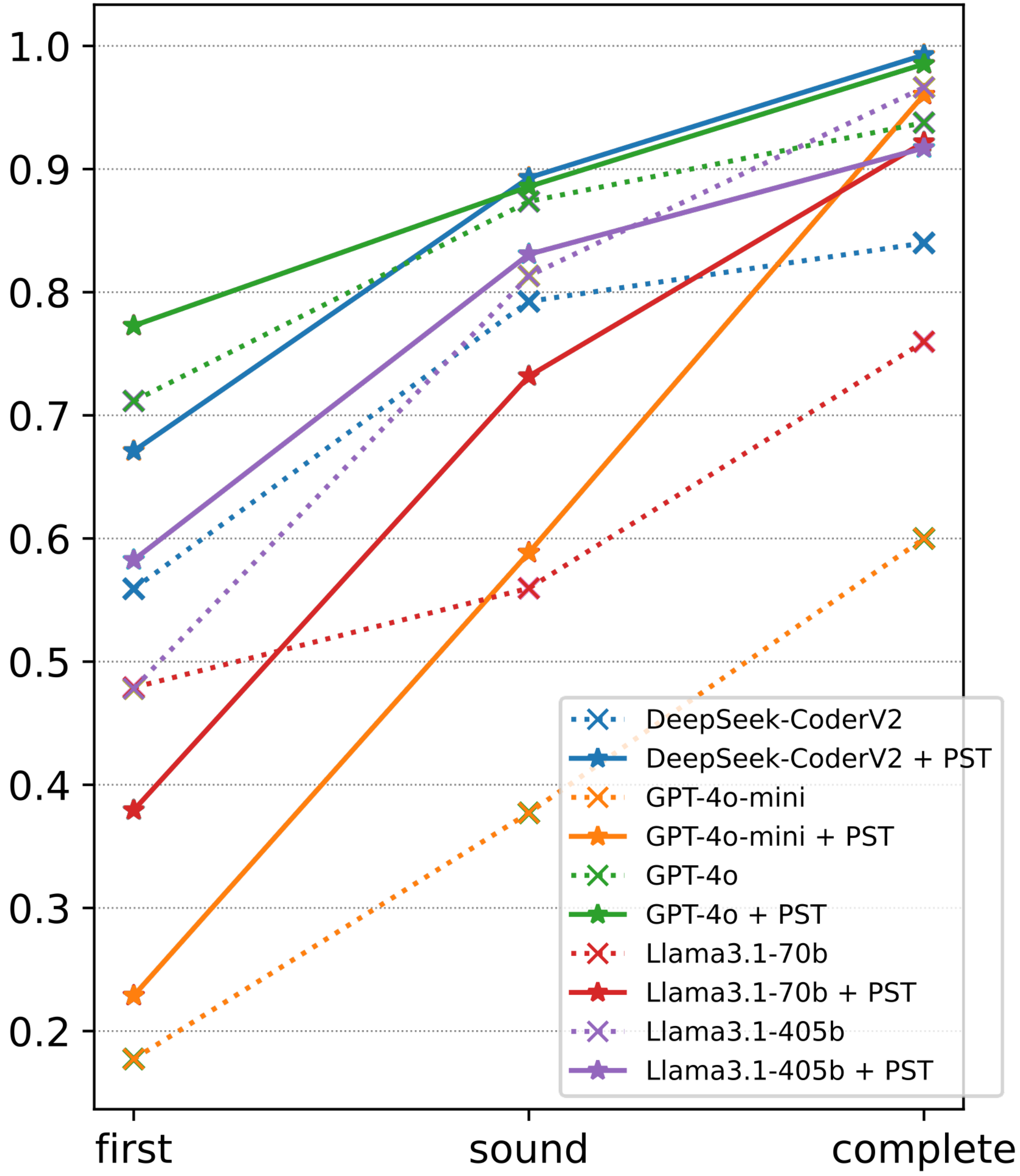
How does it work?



Is it much worse than a human feedback?

		24 Game	PrOntoQA	Sokoban	Crossword	BlocksWorld
AutoToS	GPT-4o-mini	8.8	4.8	6.4	9.6	10.0
	GPT-4o	3.4	2.6	2.2	5.8	2.0
	Llama3.1-405b	3.4	2.0	2.6	4.0	3.2
	Llama3.1-70b	7.4	2.0	8.2	6.2	5.8
	DeepSeek-CoderV2	4.4	2.0	2.8	6.6	4.2
ToS	GPT-4	2.2	2.6	NA	3.8	3.8

Does it match the accuracy of ToS?



Does it match the accuracy of ToS?

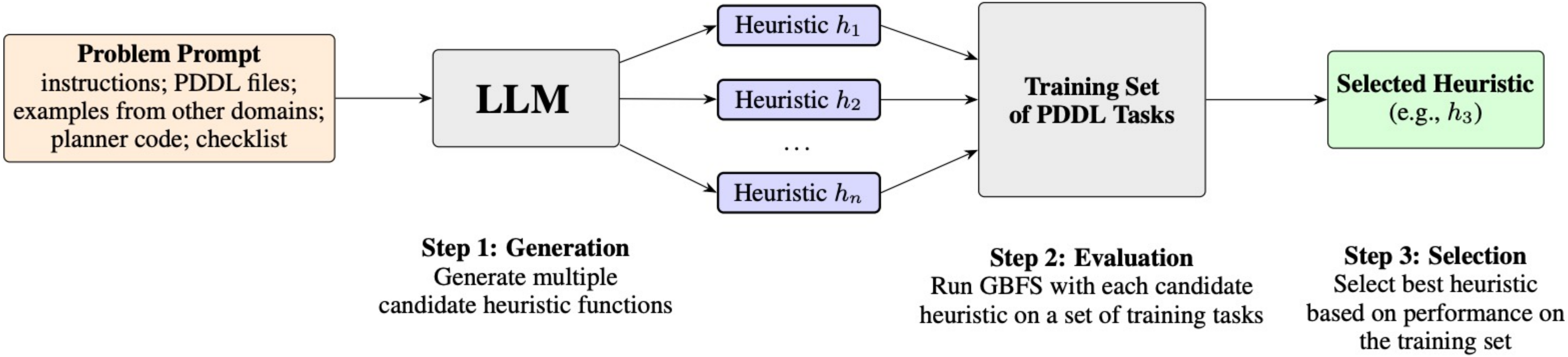
	BlocksWorld			Sokoban		
model	F	S	C	F	S	C
DS-CoderV2+PST	1 (.53)	1 (1)	1 (1)	1 (.56)	1 (.76)	1 (.96)
GPT-4o-mini+PST	0 (0)	.83 (.21)	.83 (.83)	0 (0)	1 (.73)	1 (.97)
GPT-4o+PST	1 (1)	1 (1)	1 (1)	1 (.76)	1 (.93)	1 (.93)
Llama3.1-70b+PST	0 (0)	1 (.64)	1 (.79)	0 (0)	.82 (.41)	.82 (.82)
Llama3.1-405b+PST	0 (0)	1 (.86)	1 (.86)	1 (.36)	1 (.36)	1 (.73)

Heuristics generation

Correa et al, NeurIPS 2025

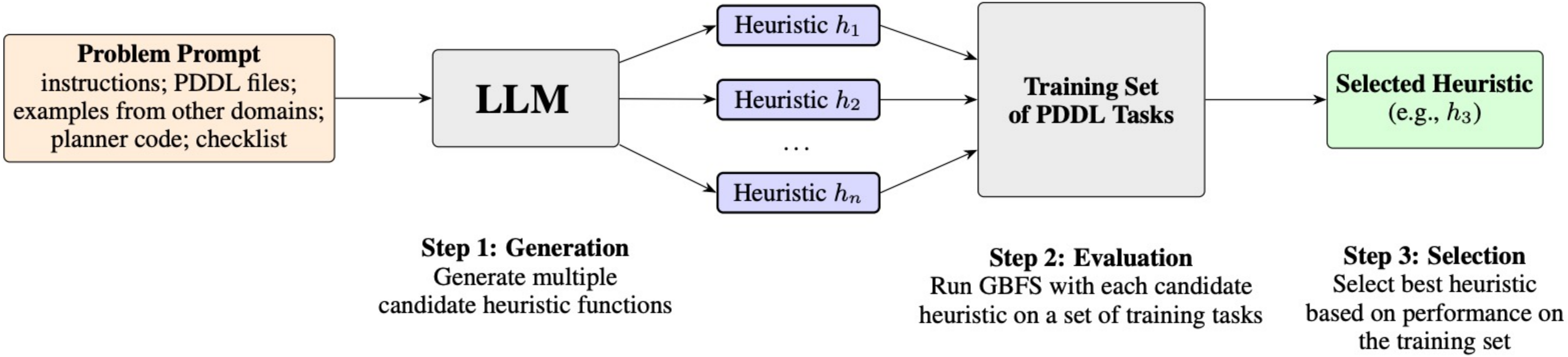
Heuristics generation

Correa et al, NeurIPS 2025



Heuristics generation

Correa et al, NeurIPS 2025



A domain-dependent heuristic for the Blocksworld domain.

 # Summary
This heuristic estimates the number of actions required to achieve the goal by considering the number of blocks that need to be moved and the blocks above them in the current state. For each block not in its goal position, the cost is 2 times the number of blocks above it plus 2. Held blocks not in the goal position add 1 to the cost.

Heuristics Generation

Correa et al, NeurIPS 2025

	End-to-End				Pyperplan							Fast Downward (C++)						
	Gemini 2.0		DeepSeek				**Gemini 2.0**		**DeepSeek**									
Domain	–	Think.	V3	R1	h^0	h^{FF}	**–**	**Think.**	**V3**	**R1 D.**	**R1**	h^{GC}	h^{lmc}	h^{FF}	h^{cea}	h^{cg}	h^{add}	h^{WLF}_{GPR}
Blocksworld (90)	2	40	2	17	6	24	35	37	45	34	66	32	39	27	40	34	44	72
Childsnack (90)	3	59	12	40	9	17	32	14	55	16	22	23	13	25	29	29	29	31
Floortile (90)	0	0	0	0	1	10	4	8	3	3	4	3	3	12	10	7	14	2
Miconic (90)	6	21	10	24	30	74	90	88	64	30	90	90	90	90	79	90	90	90
Rovers (90)	0	5	0	10	12	28	32	39	34	32	32	38	41	34	36	39	33	37
Sokoban (90)	0	14	0	8	24	31	31	32	31	24	30	42	43	36	33	35	33	38
Spanner (90)	6	39	21	47	30	30	30	30	69	30	70	30	30	30	30	30	30	73
Transport (90)	2	24	3	28	8	29	42	57	42	45	59	36	36	41	49	54	51	28
Sum (720)	19	202	48	174	120	243	296	305	343	214	373	294	295	295	306	318	324	371

Heuristics Generation

Correa et al, NeurIPS 2025

	End-to-End				Pyperplan							Fast Downward (C++)						
	Gemini 2.0		DeepSeek				Gemini 2.0		DeepSeek									
Domain	–	Think.	V3	R1	h^0	h^{FF}	–	Think.	V3	R1 D.	R1	h^{GC}	h^{lmc}	h^{FF}	h^{cea}	h^{cg}	h^{add}	h^{WLF}_{GPR}
Blocksworld (90)	2	40	2	17	6	24	35	37	45	34	66	32	39	27	40	34	44	72
Childsnack (90)	3	59	12	40	9	17	32	14	55	16	22	23	13	25	29	29	29	31
Floortile (90)	0	0	0	0	1	10	4	8	3	3	4	3	3	12	10	7	14	2
Miconic (90)	6	21	10	24	30	74	90	88	64	30	90	90	90	90	79	90	90	90
Rovers (90)	0	5	0	10	12	28	32	39	34	32	32	38	41	34	36	39	33	37
Sokoban (90)	0	14	0	8	24	31	31	32	31	24	30	42	43	36	33	35	33	38
Spanner (90)	6	39	21	47	30	30	30	30	69	30	70	30	30	30	30	30	30	73
Transport (90)	2	24	3	28	8	29	42	57	42	45	59	36	36	41	49	54	51	28
Sum (720)	19	202	48	174	120	243	296	305	343	214	373	294	295	295	306	318	324	371

Heuristics Generation

Correa et al, NeurIPS 2025

	End-to-End				Pyperplan							Fast Downward (C++)						
	Gemini 2.0		DeepSeek				Gemini 2.0		DeepSeek									
Domain	–	Think.	V3	R1	h^0	h^{FF}	–	Think.	V3	R1 D.	R1	h^{GC}	h^{lmc}	h^{FF}	h^{cea}	h^{cg}	h^{add}	h^{WLF}_{GPR}
Blocksworld (90)	2	40	2	17	6	24	35	37	45	34	66	32	39	27	40	34	44	72
Childsnack (90)	3	59	12	40	9	17	32	14	55	16	22	23	13	25	29	29	29	31
Floortile (90)	0	0	0	0	1	10	4	8	3	3	4	3	3	12	10	7	14	2
Miconic (90)	6	21	10	24	30	74	90	88	64	30	90	90	90	90	79	90	90	90
Rovers (90)	0	5	0	10	12	28	32	39	34	32	32	38	41	34	36	39	33	37
Sokoban (90)	0	14	0	8	24	31	31	32	31	24	30	42	43	36	33	35	33	38
Spanner (90)	6	39	21	47	30	30	30	30	69	30	70	30	30	30	30	30	30	73
Transport (90)	2	24	3	28	8	29	42	57	42	45	59	36	36	41	49	54	51	28
Sum (720)	19	202	48	174	120	243	296	305	343	214	373	294	295	295	306	318	324	371

Heuristics Generation

Correa et al, NeurIPS 2025

	End-to-End				Pyperplan							Fast Downward (C++)						
	Gemini 2.0		DeepSeek				Gemini 2.0		DeepSeek									
Domain	–	Think.	V3	R1	h^0	h^{FF}	–	Think.	V3	R1 D.	R1	h^{GC}	h^{lmc}	h^{FF}	h^{cea}	h^{cg}	h^{add}	h^{WLF}_{GPR}
Blocksworld (90)	2	40	2	17	6	24	35	37	45	34	66	32	39	27	40	34	44	72
Childsnack (90)	3	59	12	40	9	17	32	14	55	16	22	23	13	25	29	29	29	31
Floortile (90)	0	0	0	0	1	10	4	8	3	3	4	3	3	12	10	7	14	2
Miconic (90)	6	21	10	24	30	74	90	88	64	30	90	90	90	90	79	90	90	90
Rovers (90)	0	5	0	10	12	28	32	39	34	32	32	38	41	34	36	39	33	37
Sokoban (90)	0	14	0	8	24	31	31	32	31	24	30	42	43	36	33	35	33	38
Spanner (90)	6	39	21	47	30	30	30	30	69	30	70	30	30	30	30	30	30	73
Transport (90)	2	24	3	28	8	29	42	57	42	45	59	36	36	41	49	54	51	28
Sum (720)	19	202	48	174	120	243	296	305	343	214	373	294	295	295	306	318	324	371

Heuristics Generation

Correa et al, NeurIPS 2025

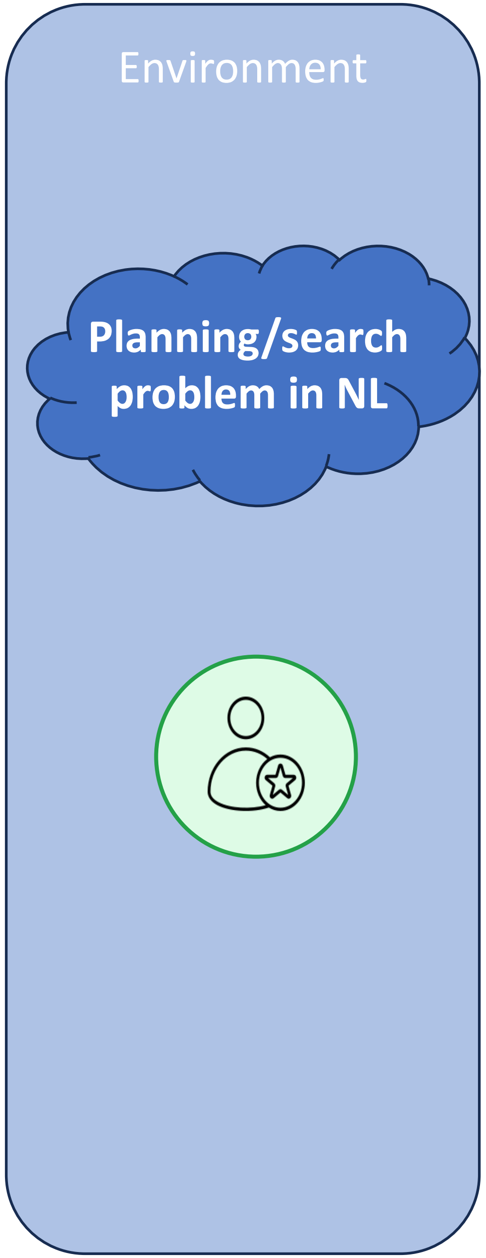
	End-to-End				Pyperplan														
	Gemini 2.0		DeepSeek				Gemini 2.0		DeepSeek			Fast Downward (C++)						
Domain	–	Think.	V3	R1	h^0	h^{FF}	–	Think.	V3	R1 D.	R1	h^{GC}	h^{lmc}	h^{FF}	h^{cea}	h^{cg}	h^{add}	h^{WLF}_{GPR}
Blocksworld (90)	2	40	2	17	6	24	35	37	45	34	66	32	39	27	40	34	44	72
Childsnack (90)	3	59	12	40	9	17	32	14	55	16	22	23	13	25	29	29	29	31
Floortile (90)	0	0	0	0	1	10	4	8	3	3	4	3	3	12	10	7	14	2
Miconic (90)	6	21	10	24	30	74	90	88	64	30	90	90	90	90	79	90	90	90
Rovers (90)	0	5	0	10	12	28	32	39	34	32	32	38	41	34	36	39	33	37
Sokoban (90)	0	14	0	8	24	31	31	32	31	24	30	42	43	36	33	35	33	38
Spanner (90)	6	39	21	47	30	30	30	30	69	30	70	30	30	30	30	30	30	73
Transport (90)	2	24	3	28	8	29	42	57	42	45	59	36	36	41	49	54	51	28
Sum (720)	19	202	48	174	120	243	296	305	343	214	373	294	295	295	306	318	324	371

Heuristics Generation

Correa et al, NeurIPS 2025

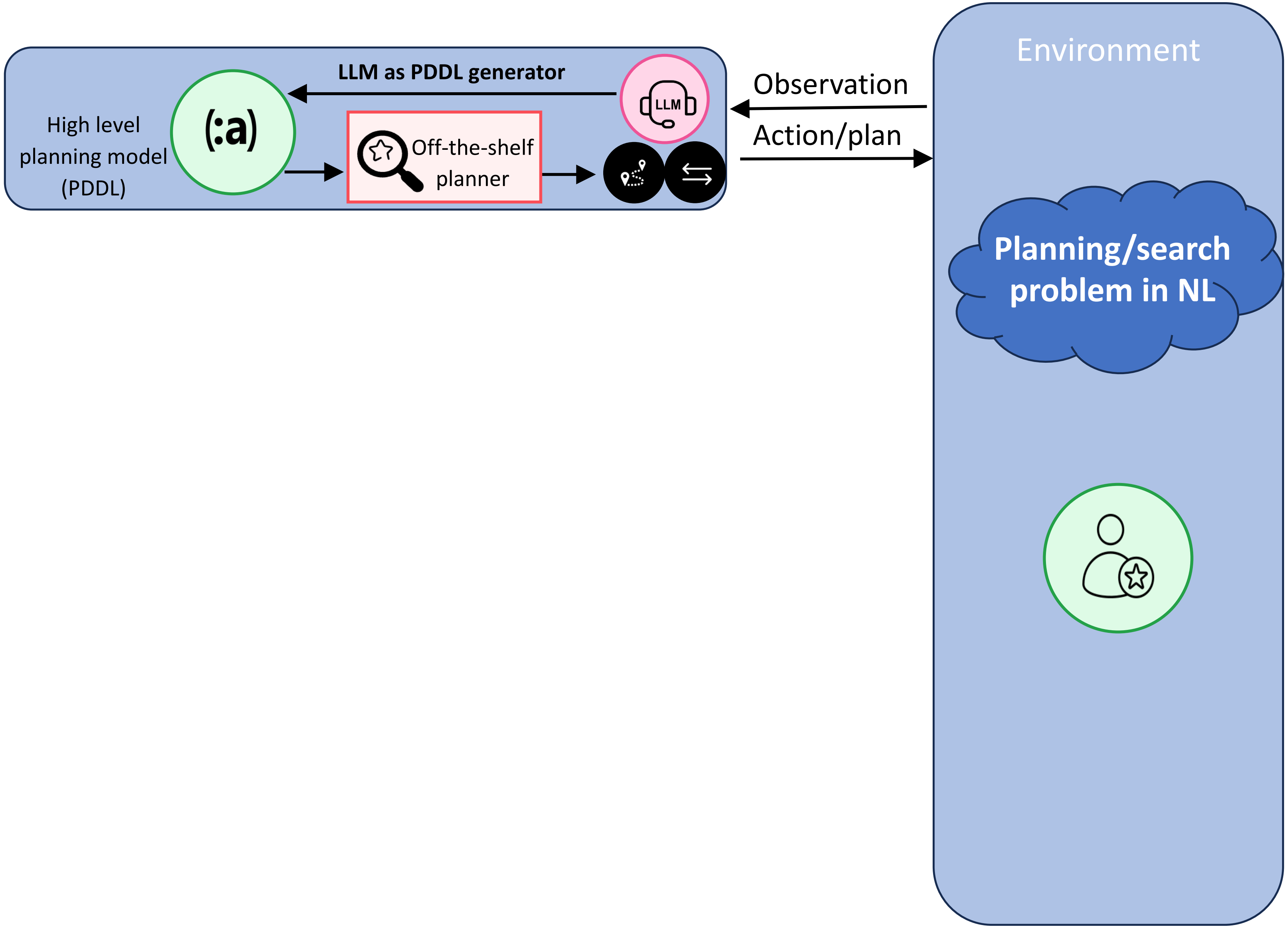
	End-to-End				Pyperplan													
	Gemini 2.0		DeepSeek				**Gemini 2.0**		**DeepSeek**			Fast Downward (C++)						
Domain	–	Think.	V3	R1	h^0	h^{FF}	**–**	**Think.**	**V3**	**R1 D.**	**R1**	h^{GC}	h^{lmc}	h^{FF}	h^{cea}	h^{cg}	h^{add}	h^{WLF}_{GPR}
Blocksworld (90)	2	40	2	17	6	24	35	37	45	34	66	32	39	27	40	34	44	72
Childsnack (90)	3	59	12	40	9	17	32	14	55	16	22	23	13	25	29	29	29	31
Floortile (90)	0	0	0	0	1	10	4	8	3	3	4	3	3	12	10	7	14	2
Miconic (90)	6	21	10	24	30	74	90	88	64	30	90	90	90	90	79	90	90	90
Rovers (90)	0	5	0	10	12	28	32	39	34	32	32	38	41	34	36	39	33	37
Sokoban (90)	0	14	0	8	24	31	31	32	31	24	30	42	43	36	33	35	33	38
Spanner (90)	6	39	21	47	30	30	30	30	69	30	70	30	30	30	30	30	30	73
Transport (90)	2	24	3	28	8	29	42	57	42	45	59	36	36	41	49	54	51	28
Sum (720)	19	202	48	174	120	243	296	305	343	214	373	294	295	295	306	318	324	371

PDDL Generation



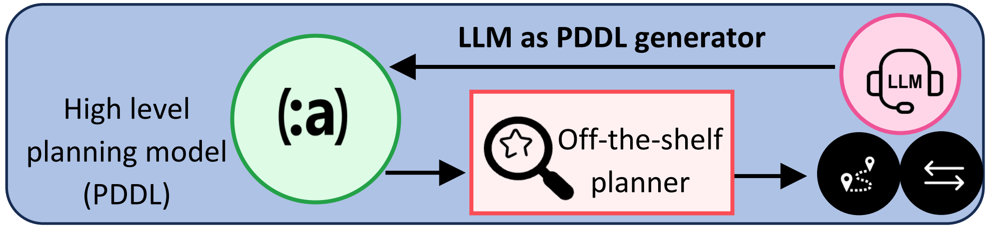
PDDL Generation

Guan et al, NeurIPS 2023
Gestrin et al, HAXP@ICAPS 2024
Oswald et al, ICAPS 2024
Huang et al, AAAI 2025
Tantakoun et al, ACL Findings 2025
Oswald et al, WM@ICLR 2026



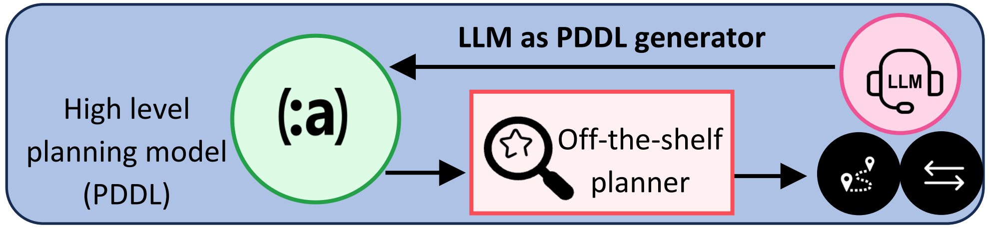
PDDL Generation: First Attempts

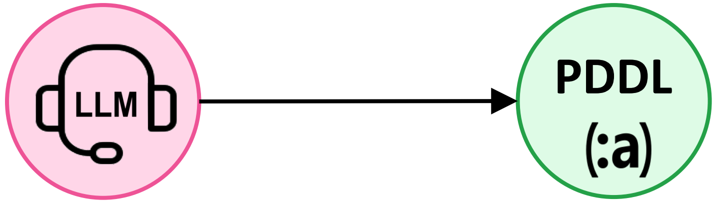
Guan et al, NeurIPS 2023



PDDL Generation: First Attempts

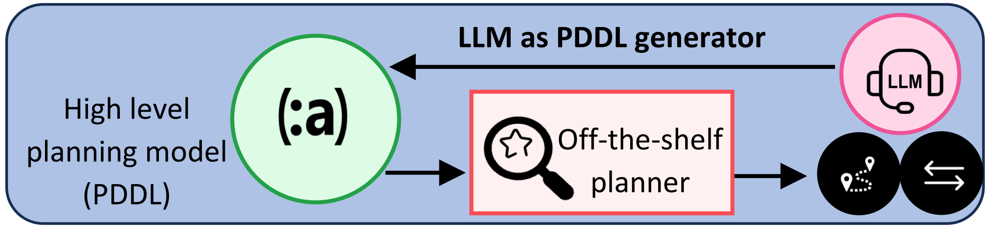
Guan et al, NeurIPS 2023

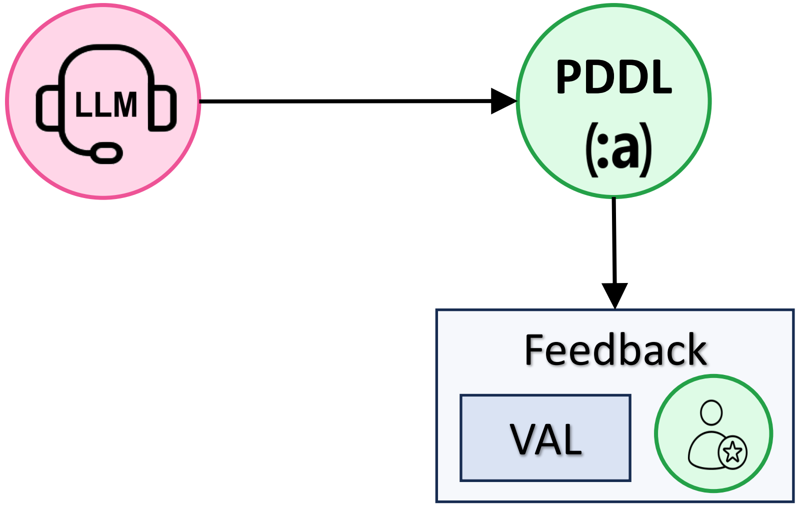




PDDL Generation: First Attempts

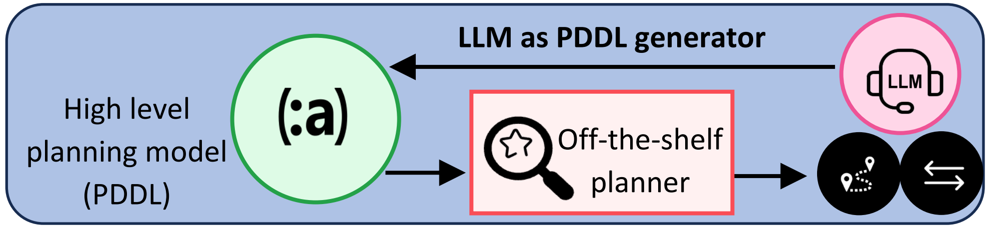
Guan et al, NeurIPS 2023

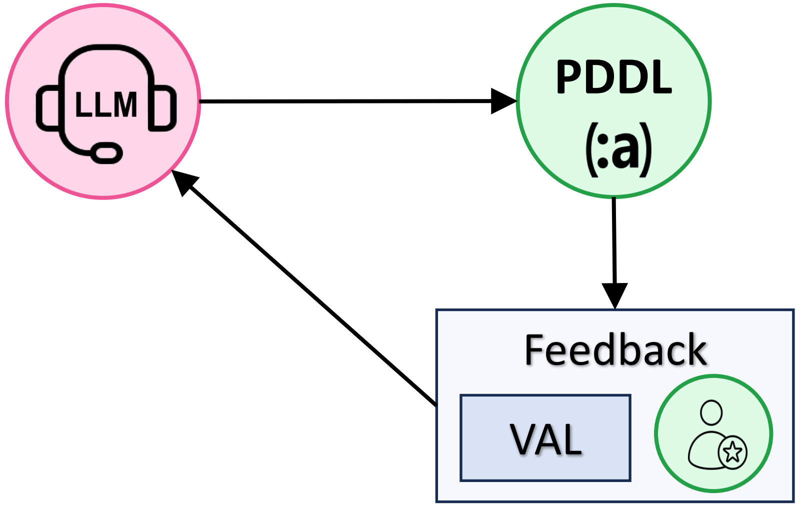




PDDL Generation: First Attempts

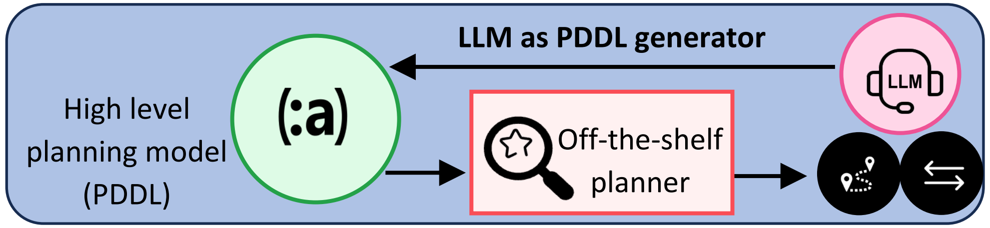
Guan et al, NeurIPS 2023

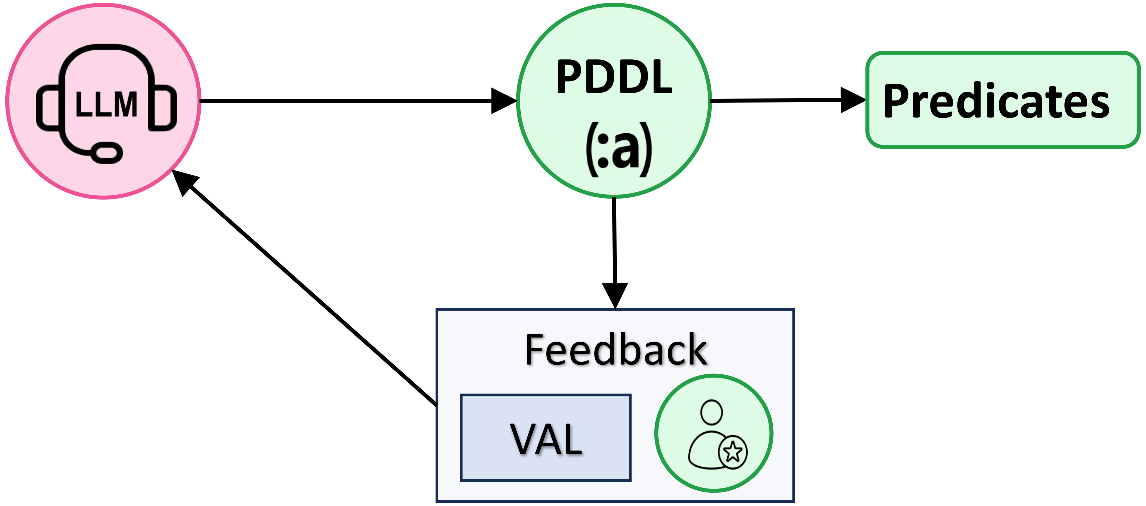




PDDL Generation: First Attempts

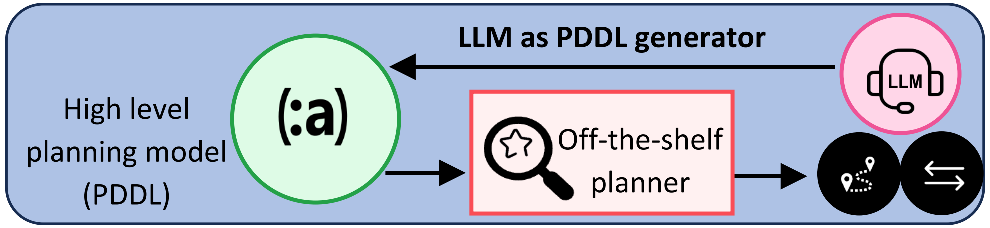
Guan et al, NeurIPS 2023





PDDL Generation: First Attempts

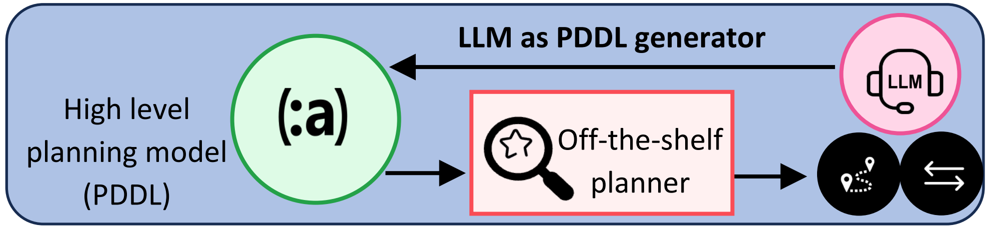
Guan et al, NeurIPS 2023

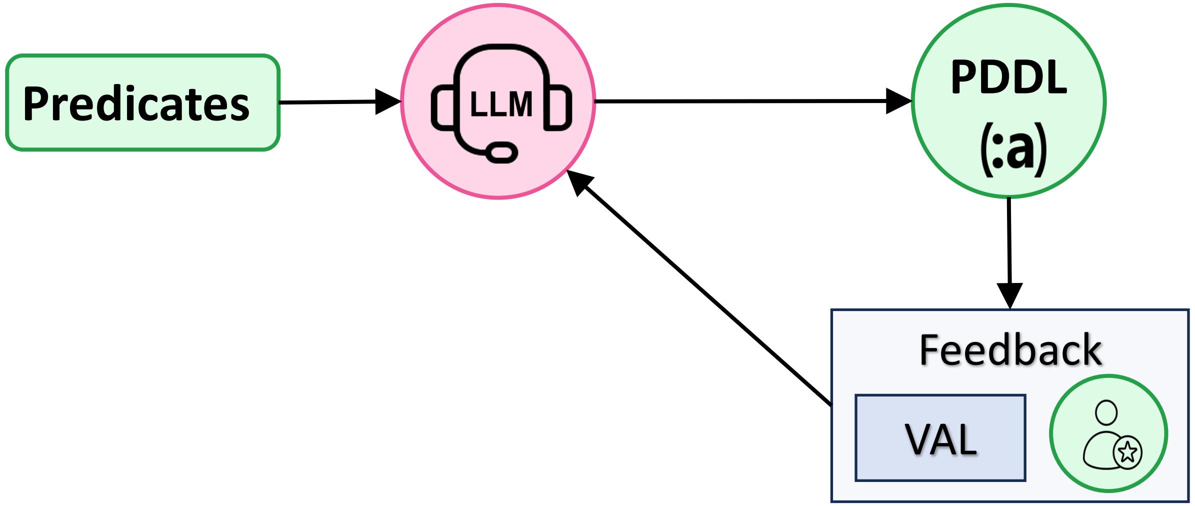


Predicates

PDDL Generation: First Attempts

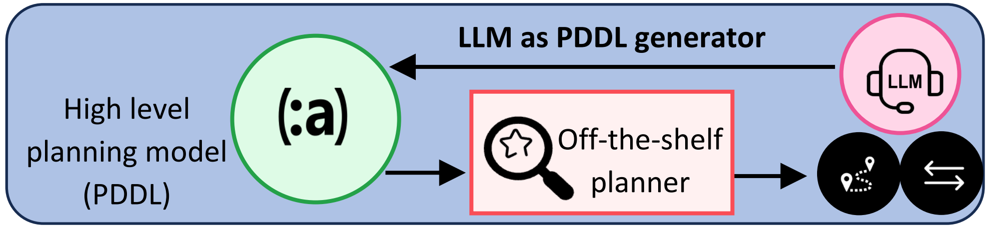
Guan et al, NeurIPS 2023





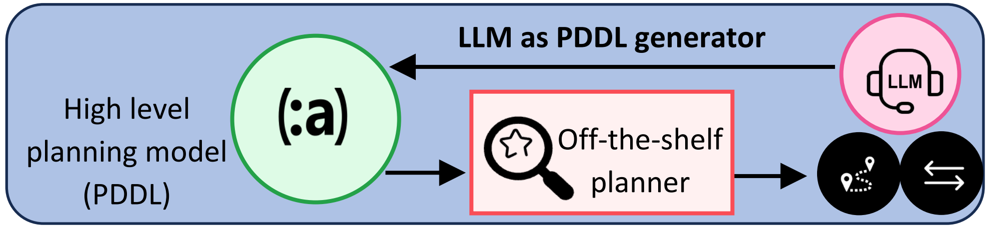
PDDL Generation: First Attempts

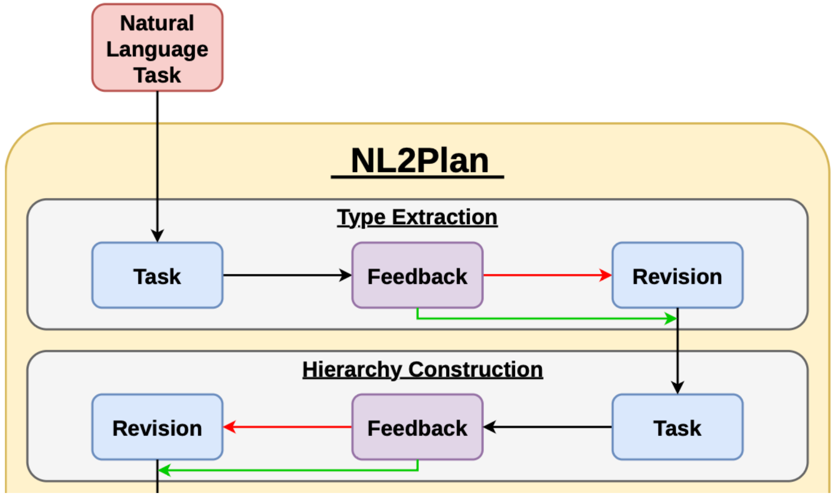
Gestrin et al, HAXP@ICAPS 2024



PDDL Generation: First Attempts

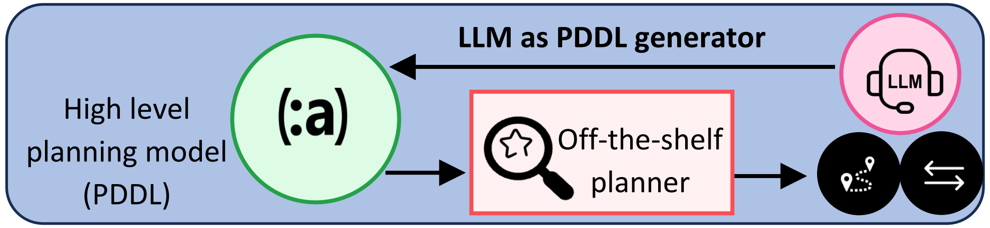
Gestrin et al, HAXP@ICAPS 2024

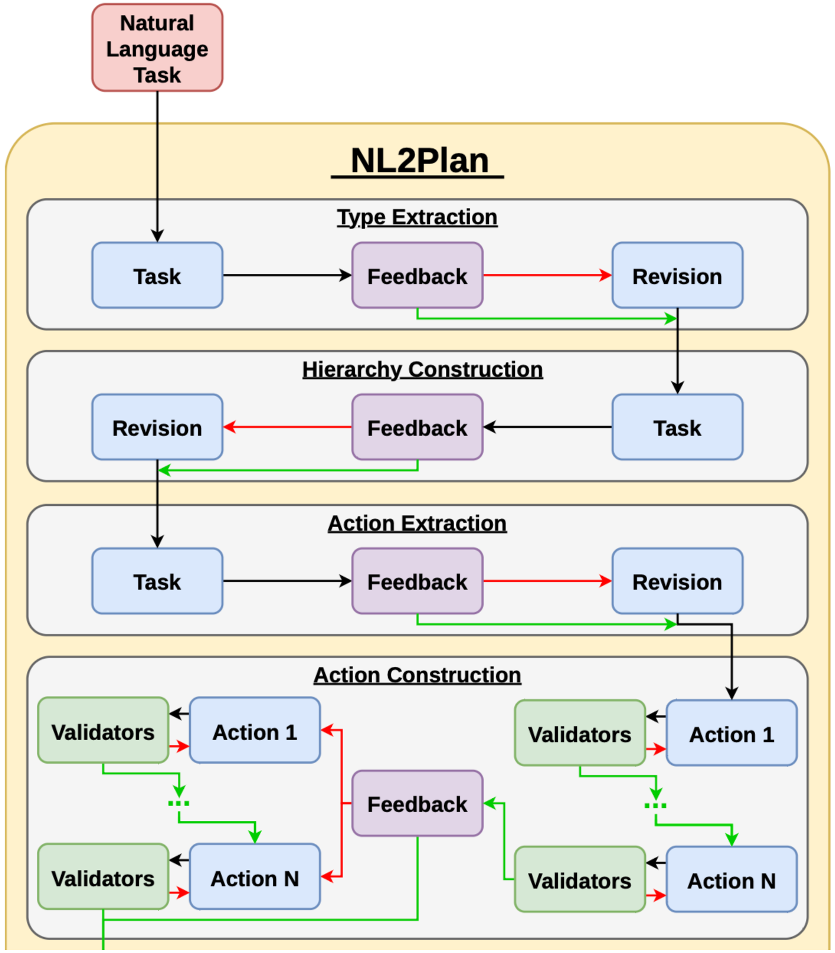




PDDL Generation: First Attempts

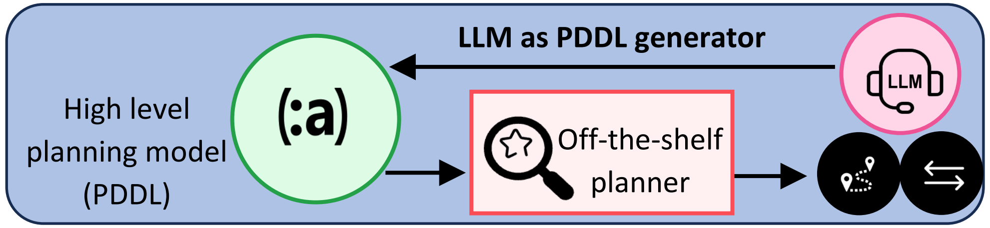
Gestrin et al, HAXP@ICAPS 2024

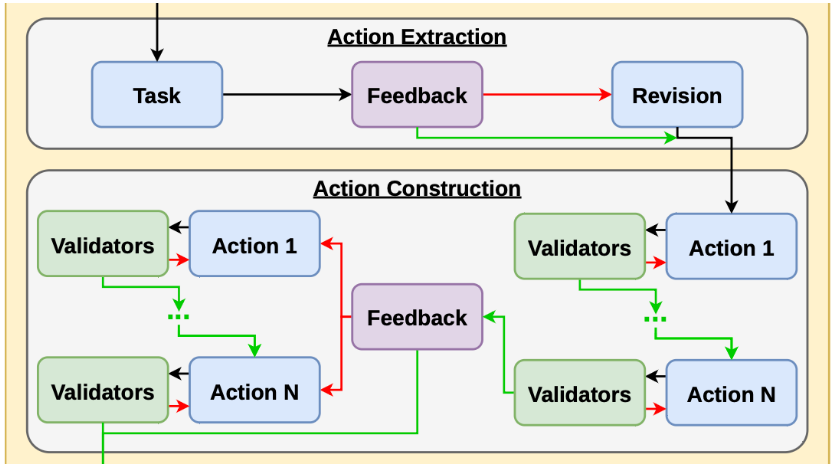




PDDL Generation: First Attempts

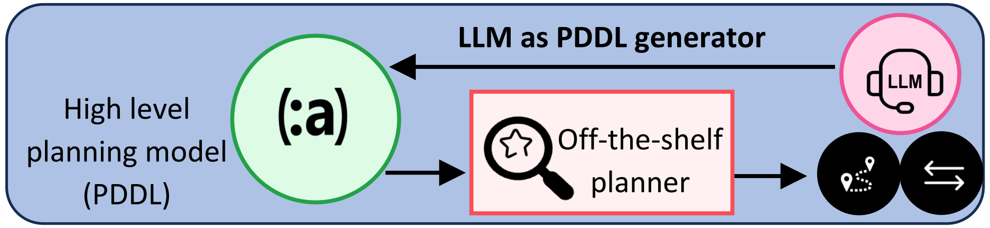
Gestrin et al, HAXP@ICAPS 2024

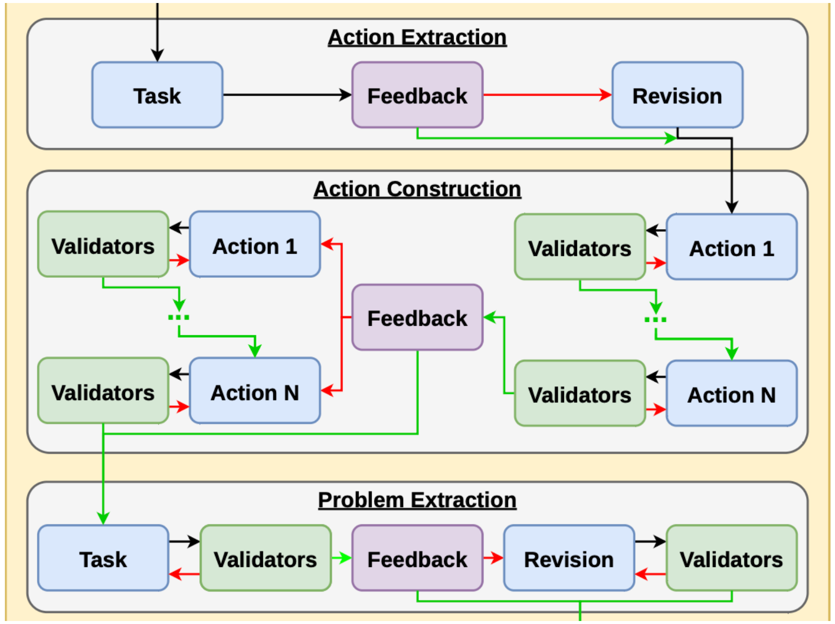




PDDL Generation: First Attempts

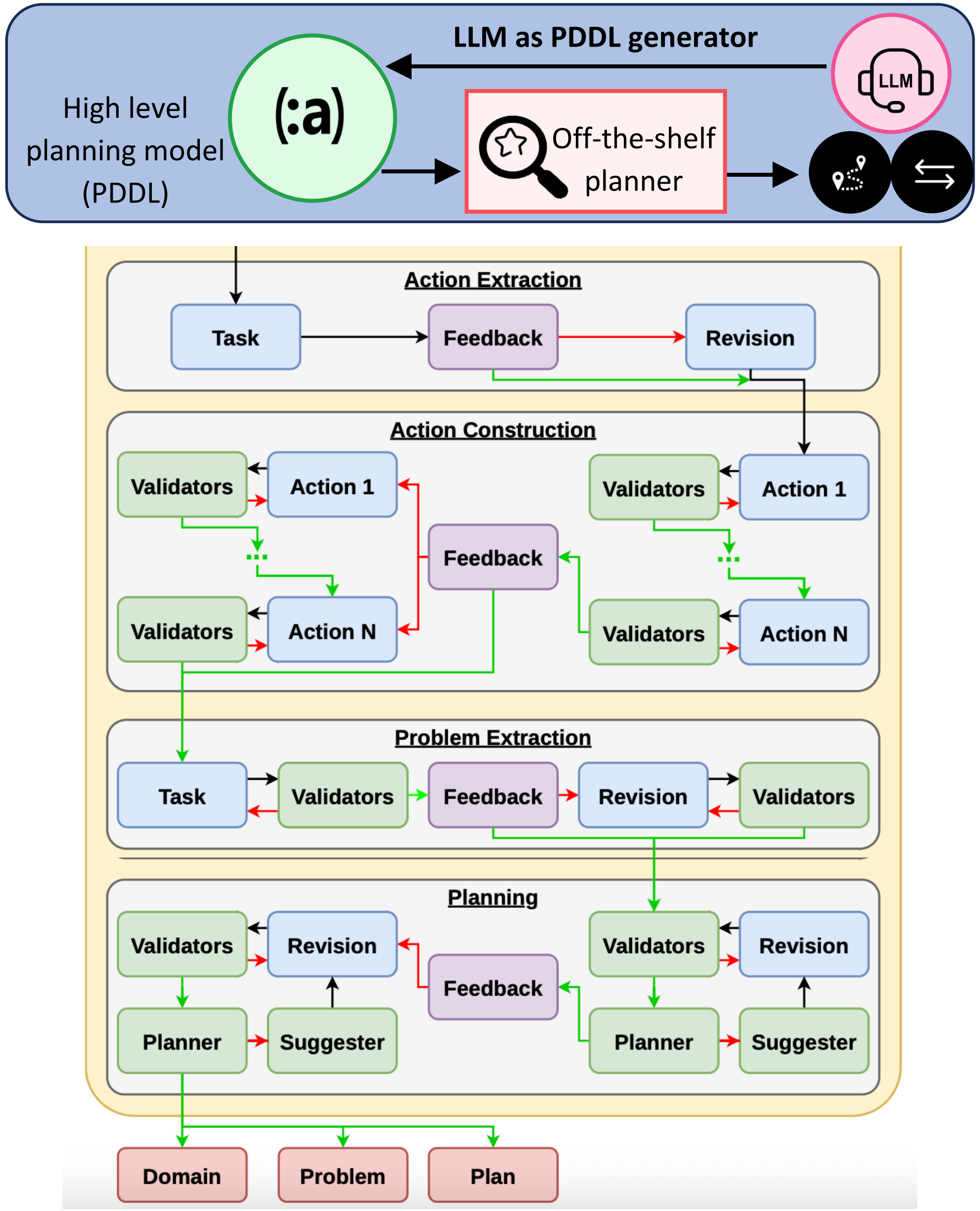
Gestrin et al, HAXP@ICAPS 2024



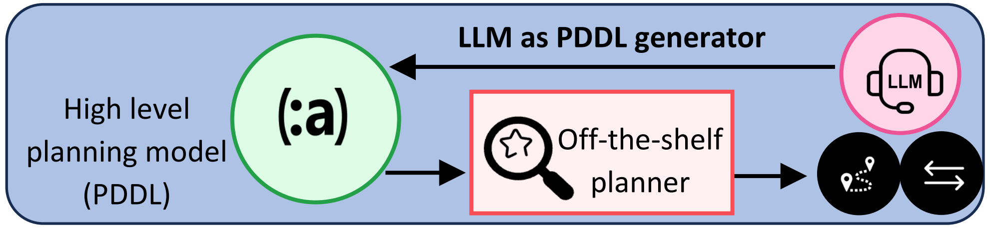


PDDL Generation: First Attempts

Gestrin et al, HAXP@ICAPS 2024

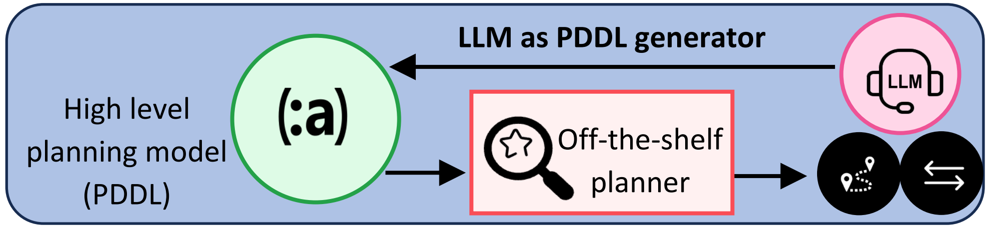


PDDL Generation



What is being tackled?

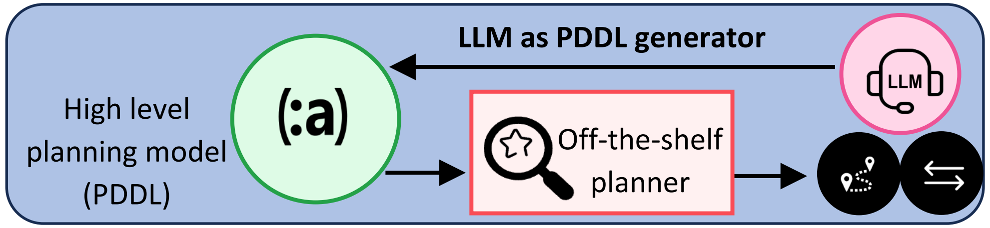
PDDL Generation



What is being tackled?

Syntactical correctness!

PDDL Generation



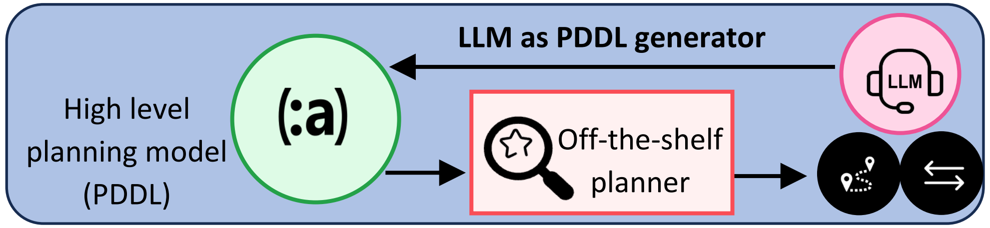
What is being tackled?

Syntactical correctness!

What about semantical correctness?

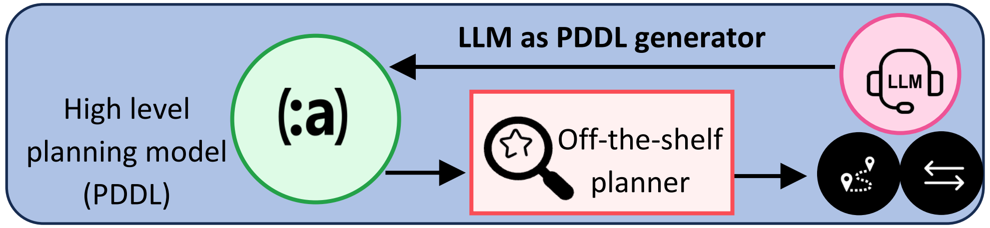
PDDL Generation: Tackling Semantic Correctness

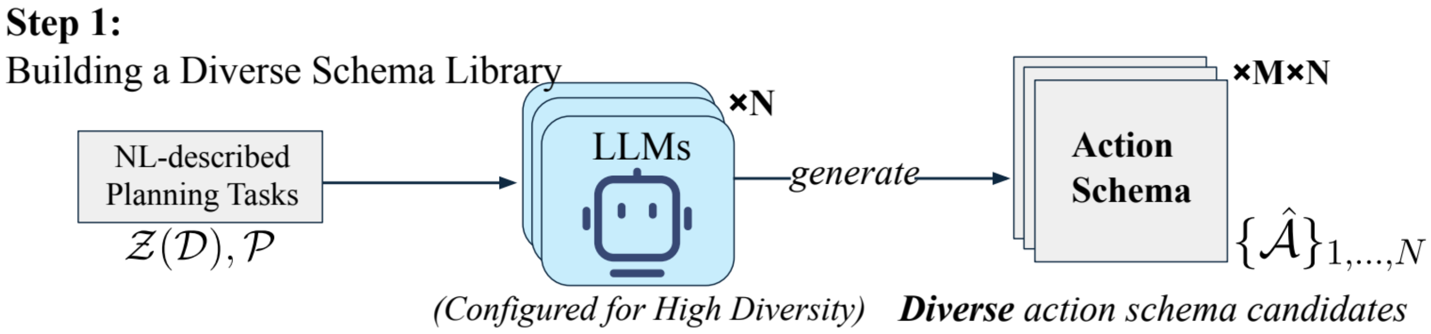
Huang et al, AAAI 2025



PDDL Generation: Tackling Semantic Correctness

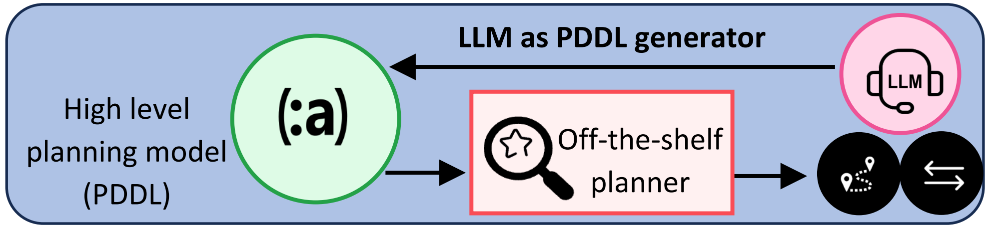
Huang et al, AAAI 2025

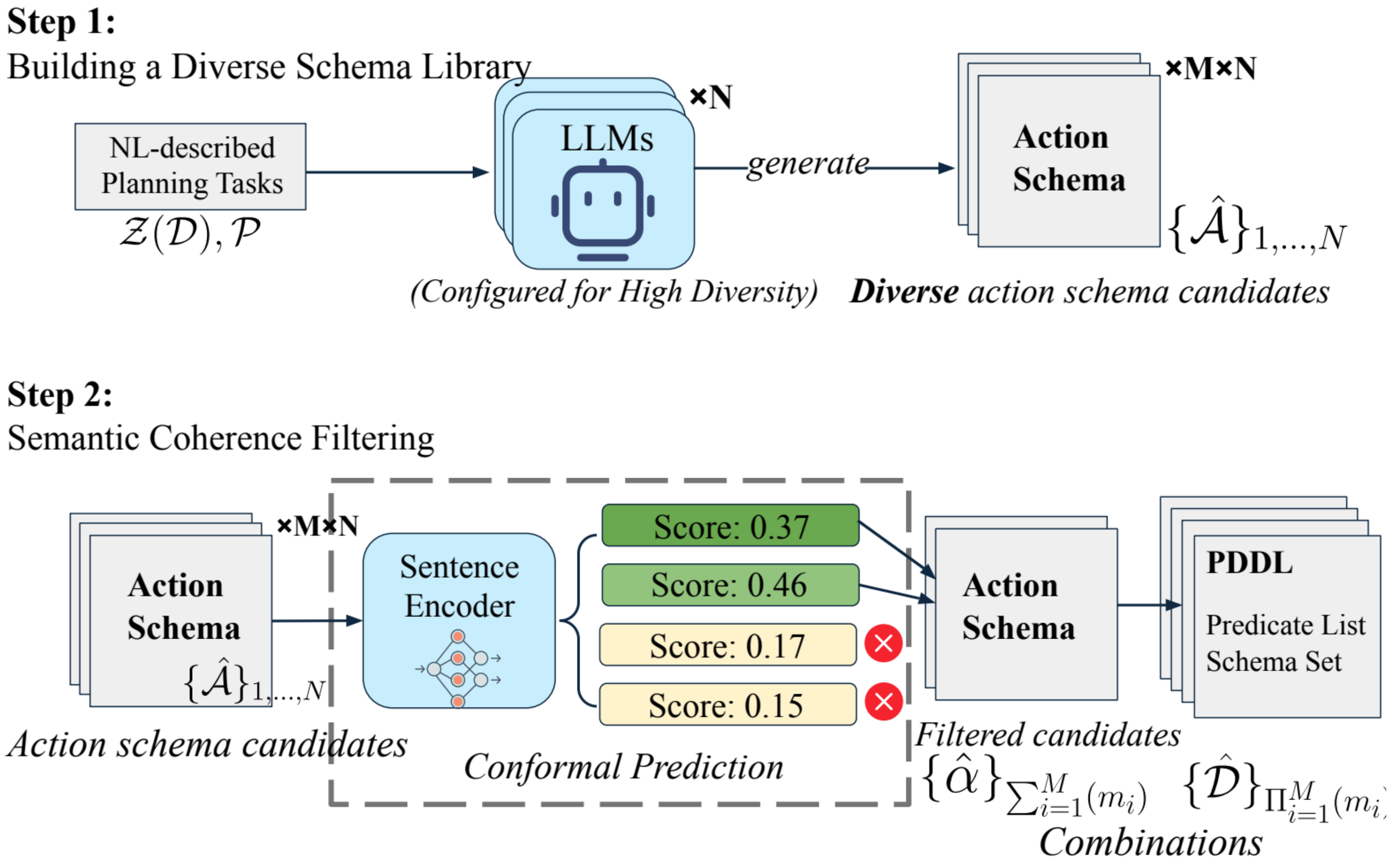




PDDL Generation: Tackling Semantic Correctness

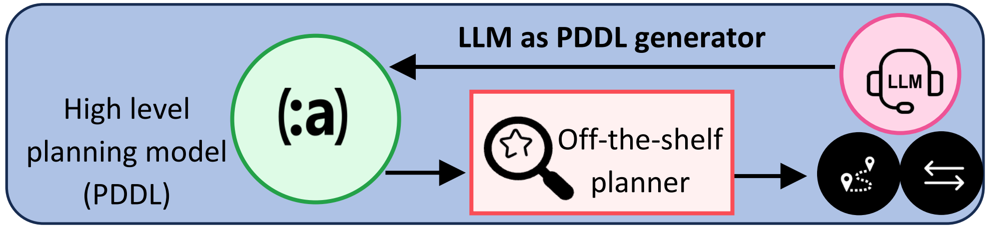
Huang et al, AAAI 2025

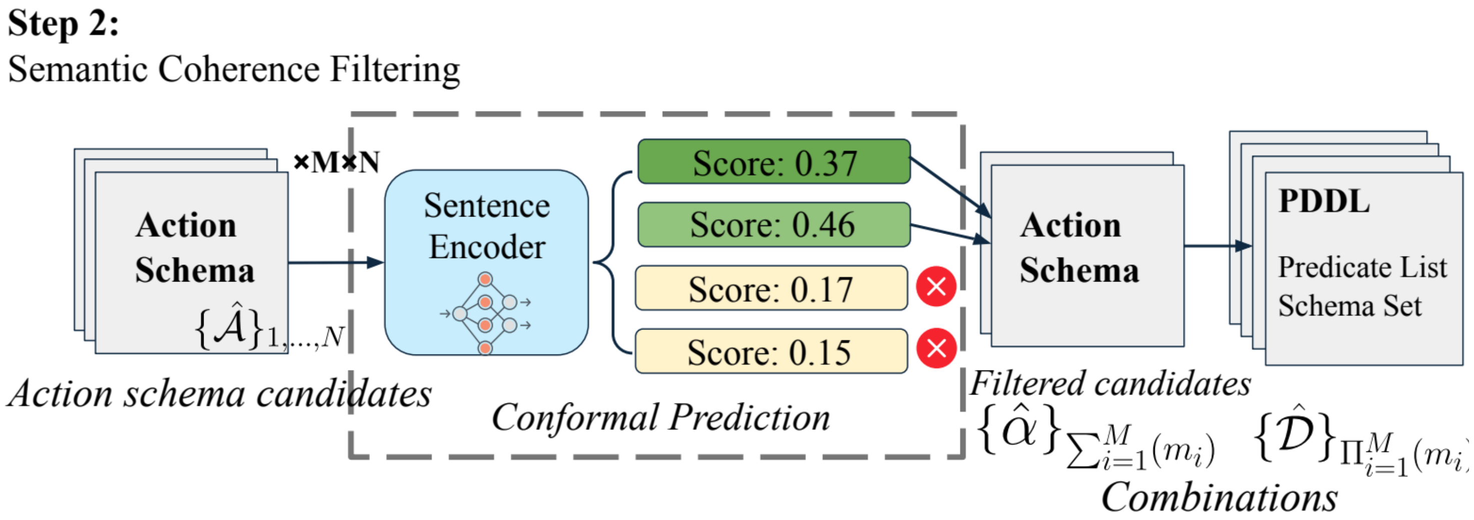




PDDL Generation: Tackling Semantic Correctness

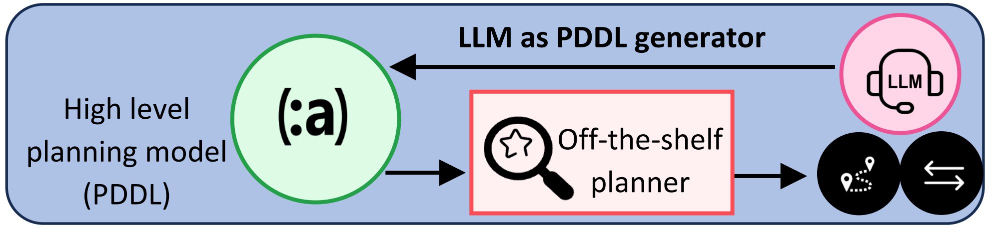
Huang et al, AAAI 2025

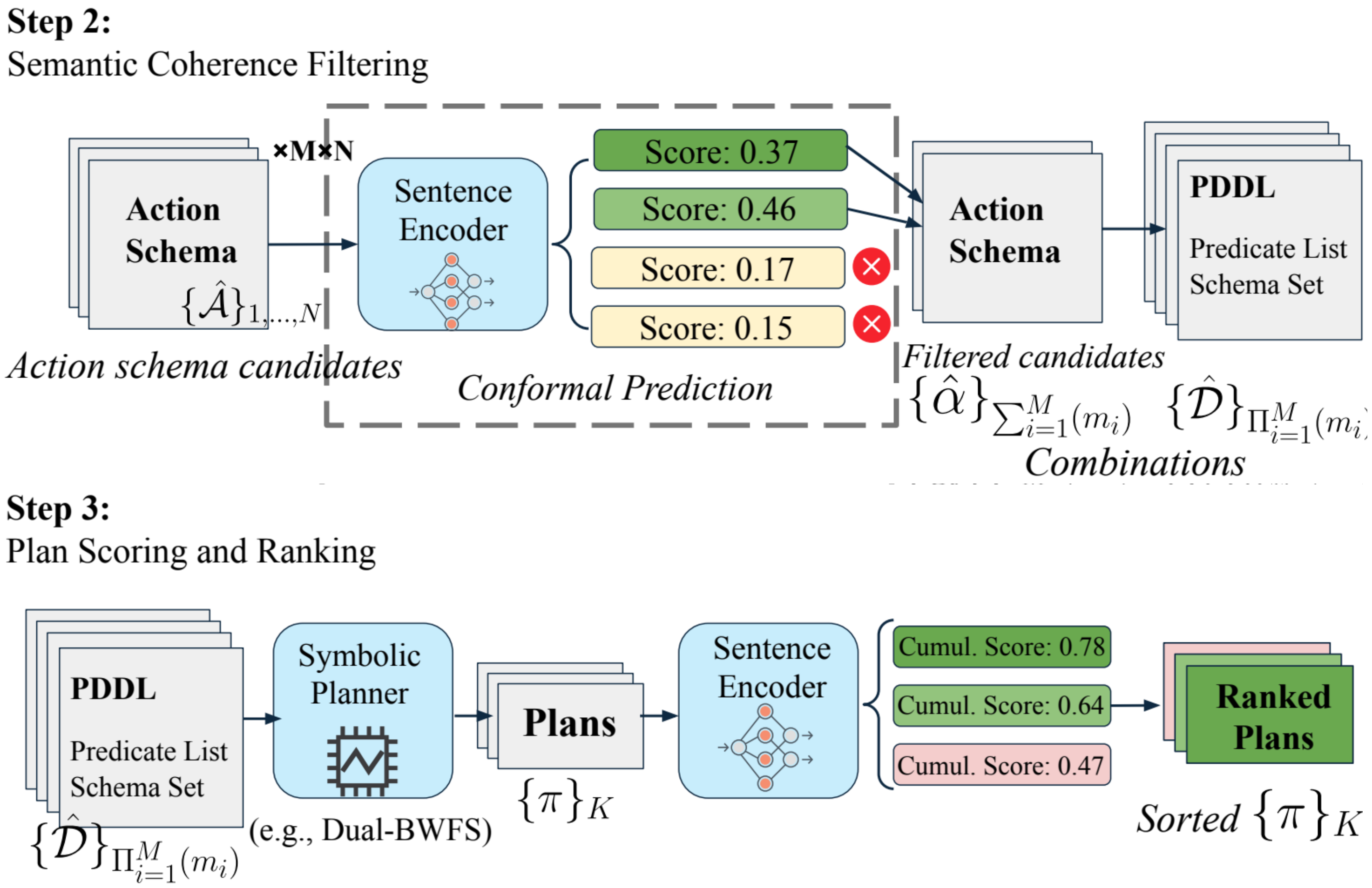




PDDL Generation: Tackling Semantic Correctness

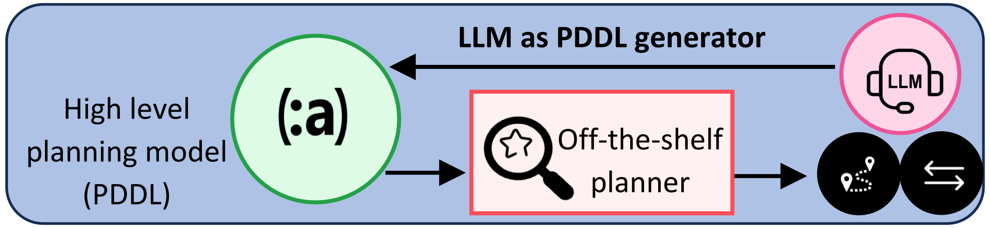
Huang et al, AAAI 2025





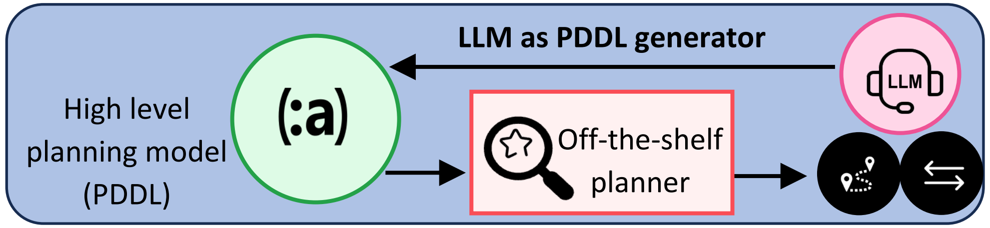
PDDL Generation: Measuring Semantic Correctness

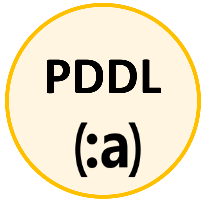
Oswald et al, ICAPS 2024

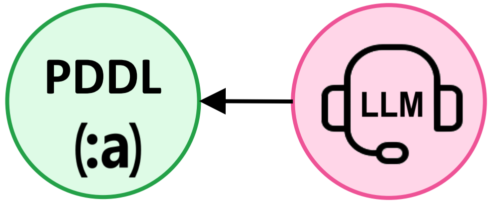


PDDL Generation: Measuring Semantic Correctness

Oswald et al, ICAPS 2024

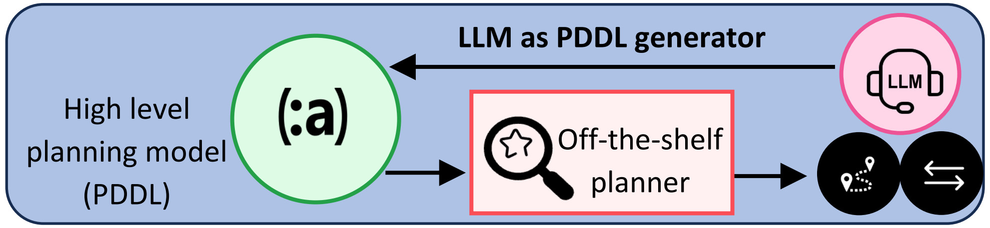






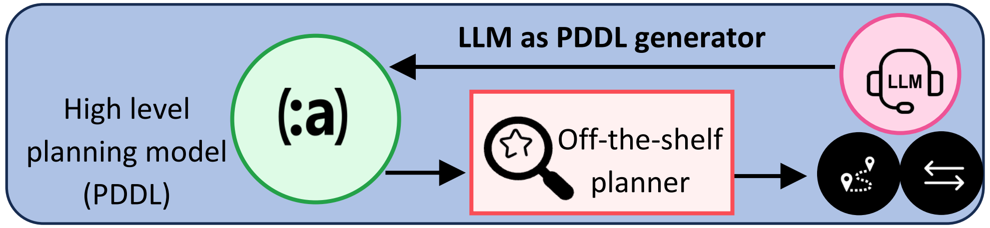
PDDL Generation: Measuring Semantic Correctness

Oswald et al, ICAPS 2024



PDDL Generation: Measuring Semantic Correctness

Oswald et al, ICAPS 2024

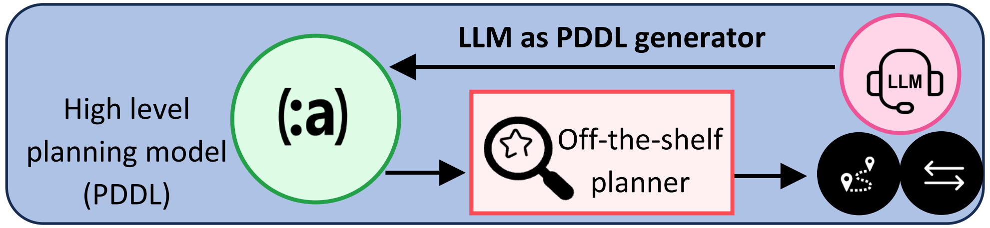


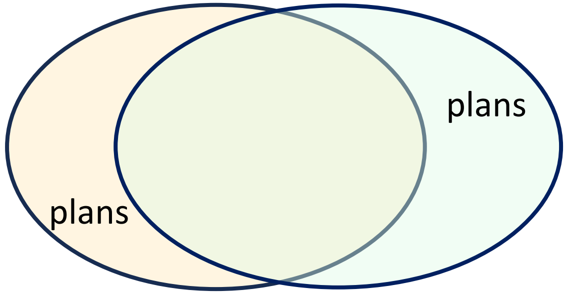
$$\begin{aligned}\text{ARE}(a, a') = & |pre(a) \triangle pre(a')| + \\ & |add(a) \triangle add(a')| + \\ & |del(a) \triangle del(a')|\end{aligned}$$

ARE

PDDL Generation: Measuring Semantic Correctness

Oswald et al, ICAPS 2024

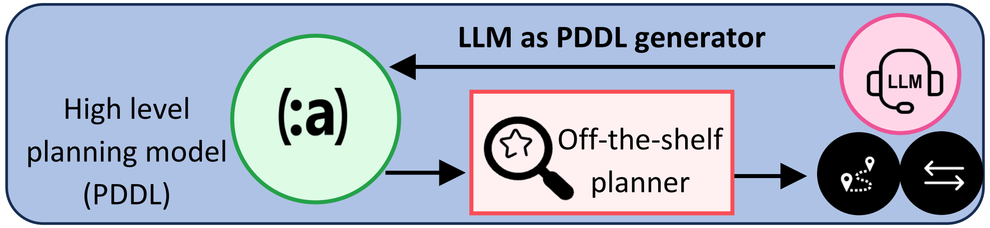


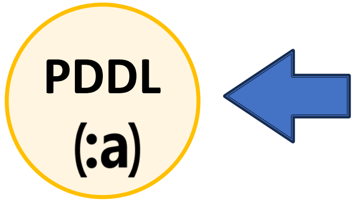


HDE

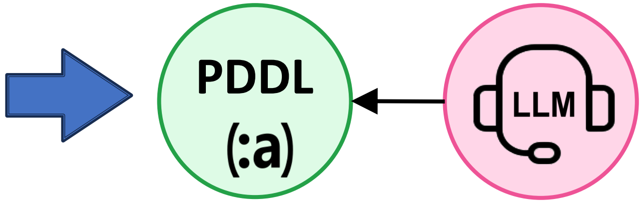
PDDL Generation: Measuring Semantic Correctness

Oswald et al, ICAPS 2024

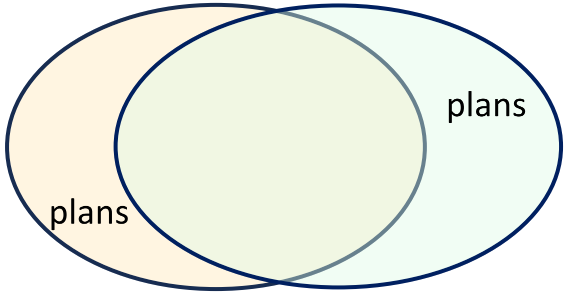




How to measure semantic equivalence?

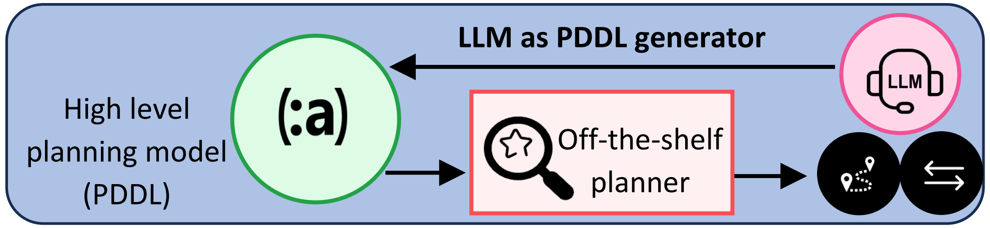


$$\mathrm{HDE}(\Pi, \Pi') = \frac{1}{2}\left(\frac{|P \cap \mathcal{P}_{\Pi'}|}{|P|} + \frac{|P' \cap \mathcal{P}_{\Pi}|}{|P'|}\right)$$

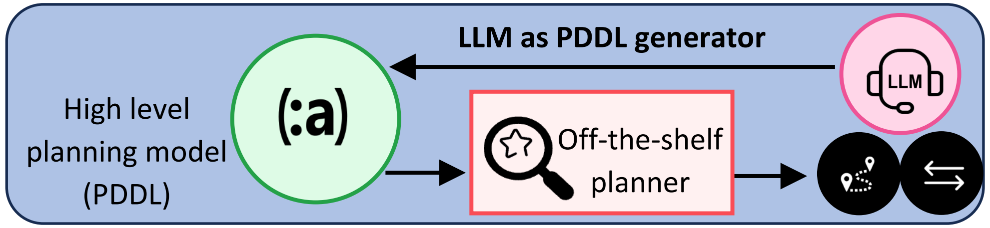


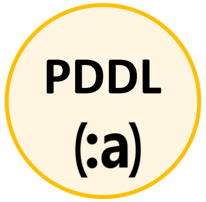
HDE

PDDL Generation: Beyond Single Loop

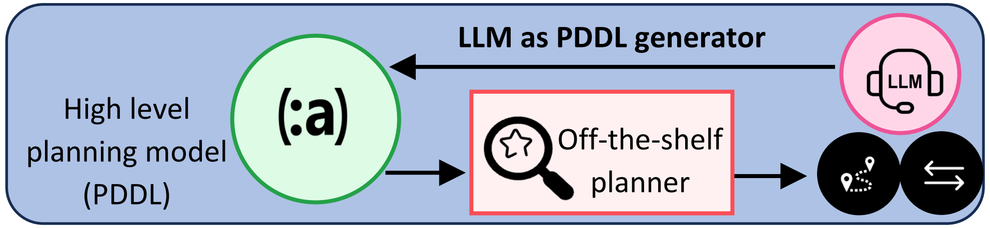


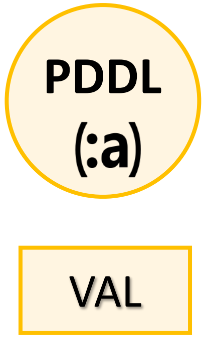
PDDL Generation: Beyond Single Loop





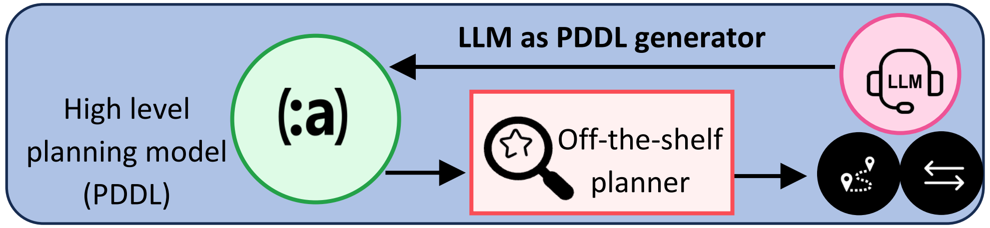
PDDL Generation: Beyond Single Loop





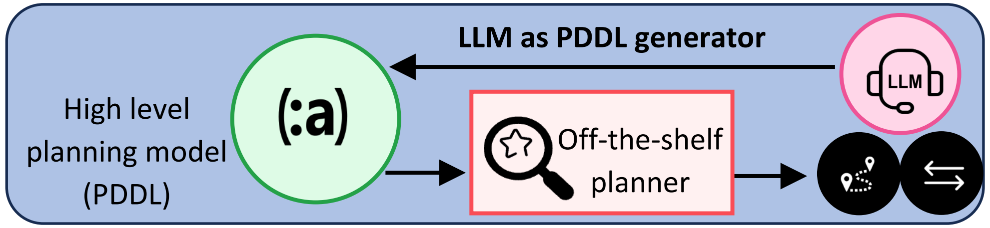
Oswald et al, WM@ICLR 2026

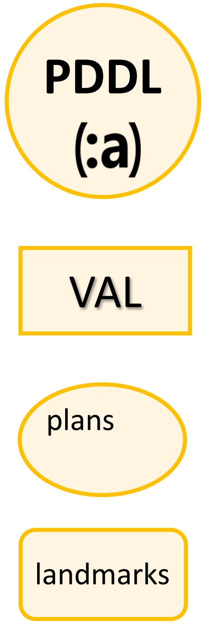
PDDL Generation: Beyond Single Loop



Oswald et al, WM@ICLR 2026

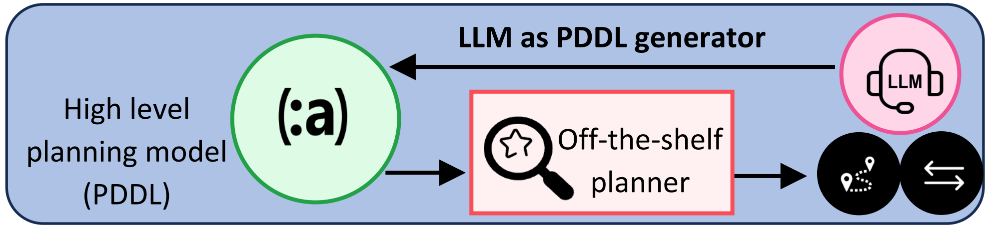
PDDL Generation: Beyond Single Loop





Oswald et al, WM@ICLR 2026

PDDL Generation: Beyond Single Loop



Semantic feedback – many forms

PDDL (:a)

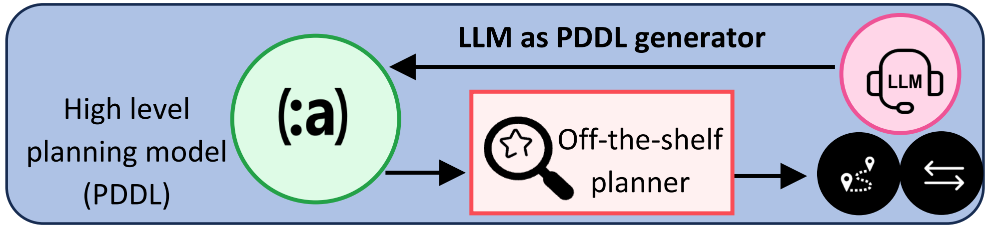
VAL

plans

landmarks

Oswald et al, WM@ICLR 2026

PDDL Generation: Beyond Single Loop



Semantic feedback – many forms

This sequence of actions should have been a plan

PDDL (:a)

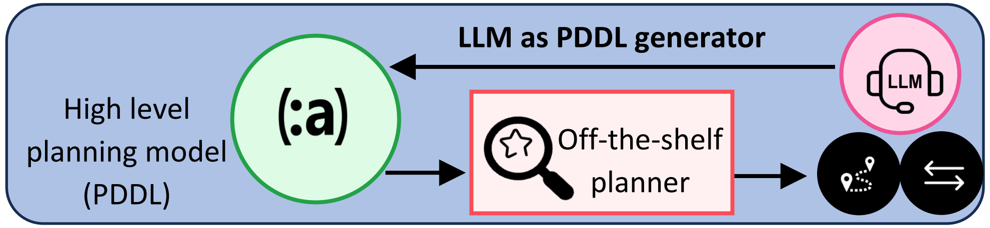
VAL

plans

landmarks

Oswald et al, WM@ICLR 2026

PDDL Generation: Beyond Single Loop



Semantic feedback – many forms

This sequence of actions should have been a plan

This sequence of actions should have not been a plan

PDDL (:a)

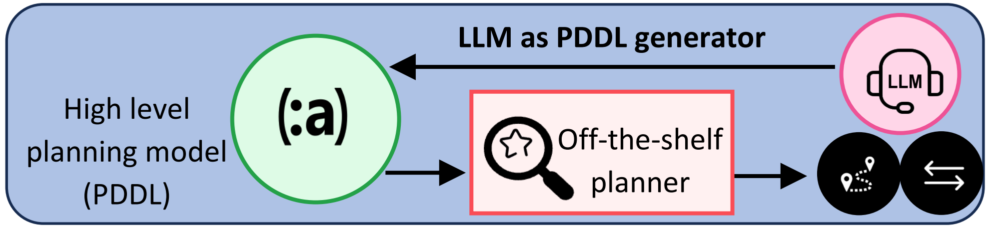
VAL

plans

landmarks

Oswald et al, WM@ICLR 2026

PDDL Generation: Beyond Single Loop

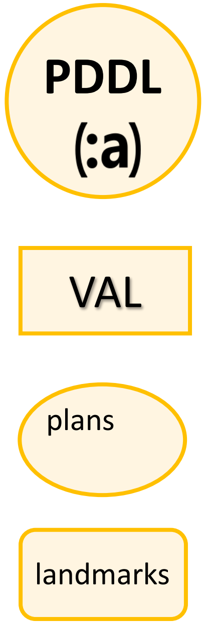


Semantic feedback – many forms

This sequence of actions should have been a plan

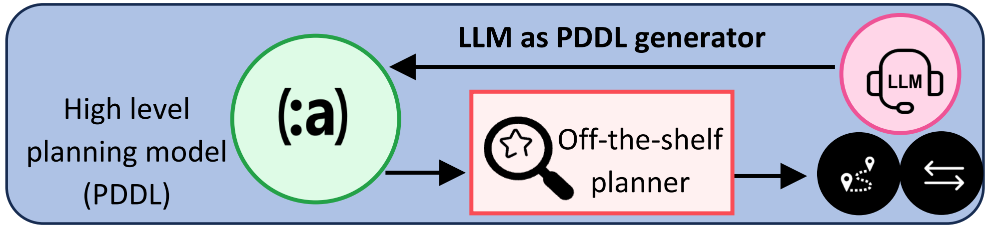
This sequence of actions should have not been a plan

This set of actions should have been a landmark



Oswald et al, WM@ICLR 2026

PDDL Generation: Beyond Single Loop



Semantic feedback – many forms

This sequence of actions should have been a plan

This sequence of actions should have not been a plan

This set of actions should have been a landmark

We can branch on the feedback given to the model!

PDDL (:a)

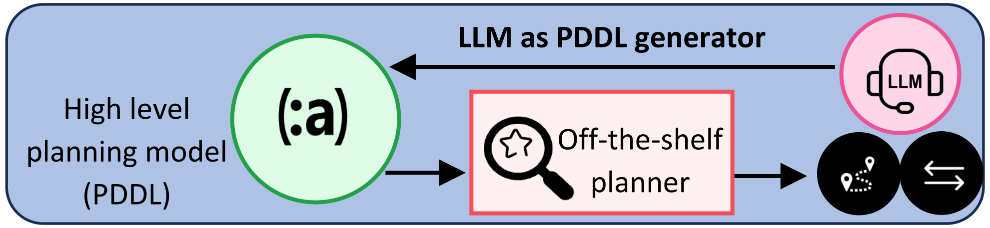
VAL

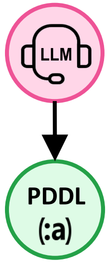
plans

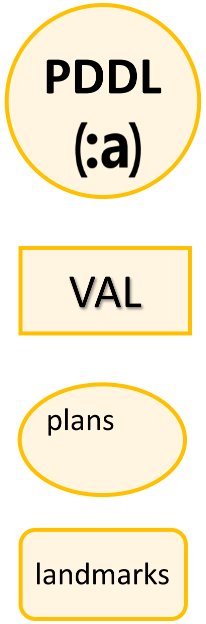
landmarks

Oswald et al, WM@ICLR 2026

PDDL Generation: Beyond Single Loop







Semantic feedback – many forms

This sequence of actions should have been a plan

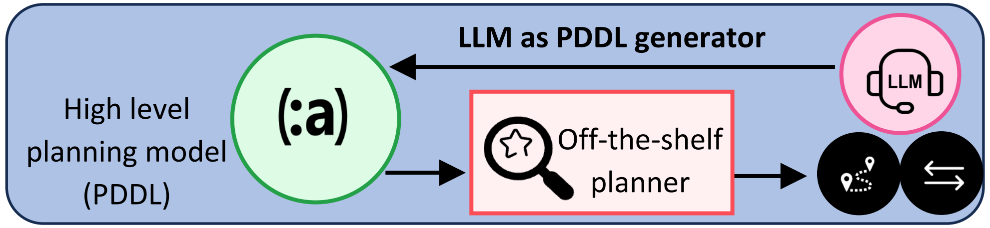
This sequence of actions should have not been a plan

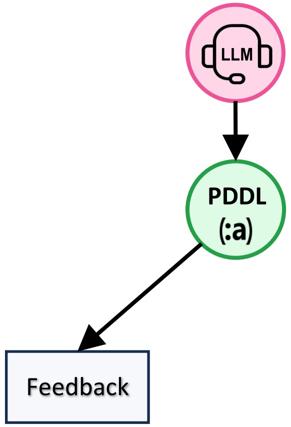
This set of actions should have been a landmark

We can branch on the feedback given to the model!

Oswald et al, WM@ICLR 2026

PDDL Generation: Beyond Single Loop





Semantic feedback – many forms

This sequence of actions should have been a plan

This sequence of actions should have not been a plan

This set of actions should have been a landmark

We can branch on the feedback given to the model!

PDDL (:a)

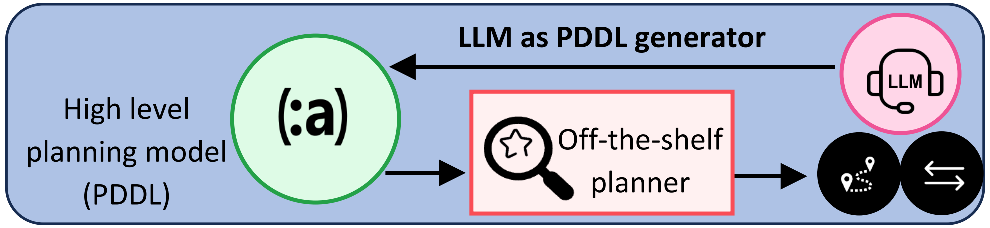
VAL

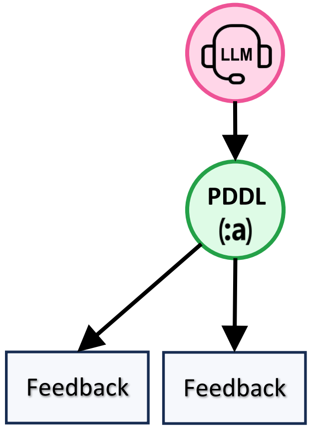
plans

landmarks

Oswald et al, WM@ICLR 2026

PDDL Generation: Beyond Single Loop



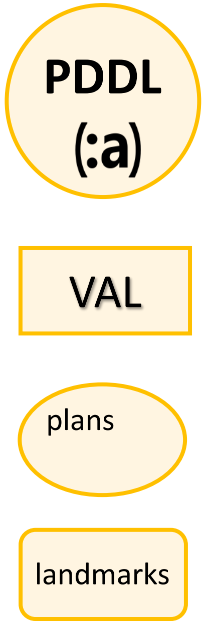


Semantic feedback – many forms

This sequence of actions should have been a plan

This sequence of actions should have not been a plan

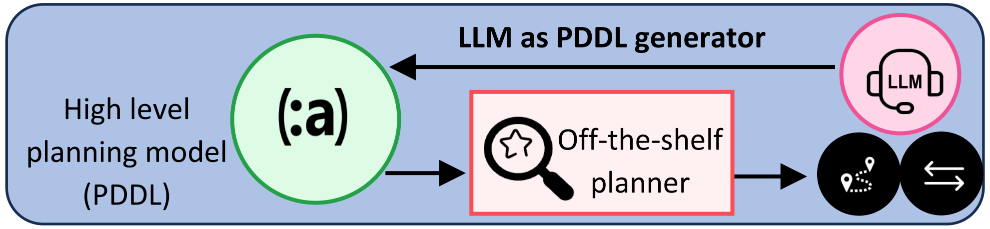
This set of actions should have been a landmark

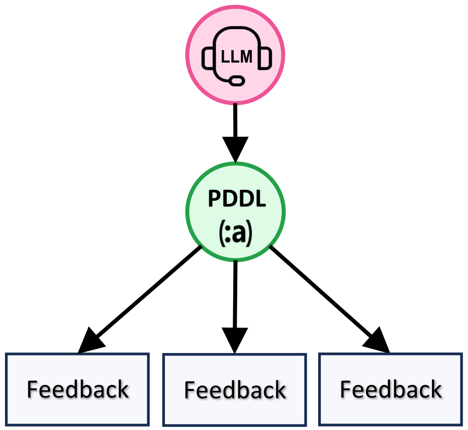


We can branch on the feedback given to the model!

Oswald et al, WM@ICLR 2026

PDDL Generation: Beyond Single Loop



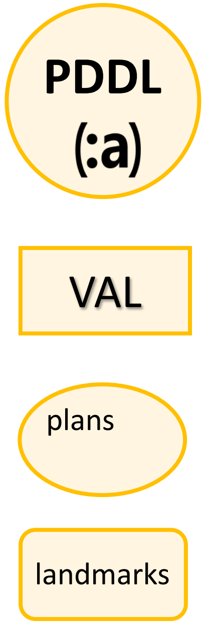


Semantic feedback – many forms

This sequence of actions should have been a plan

This sequence of actions should have not been a plan

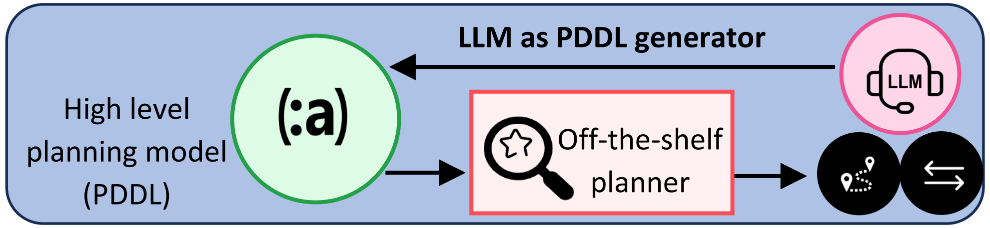
This set of actions should have been a landmark

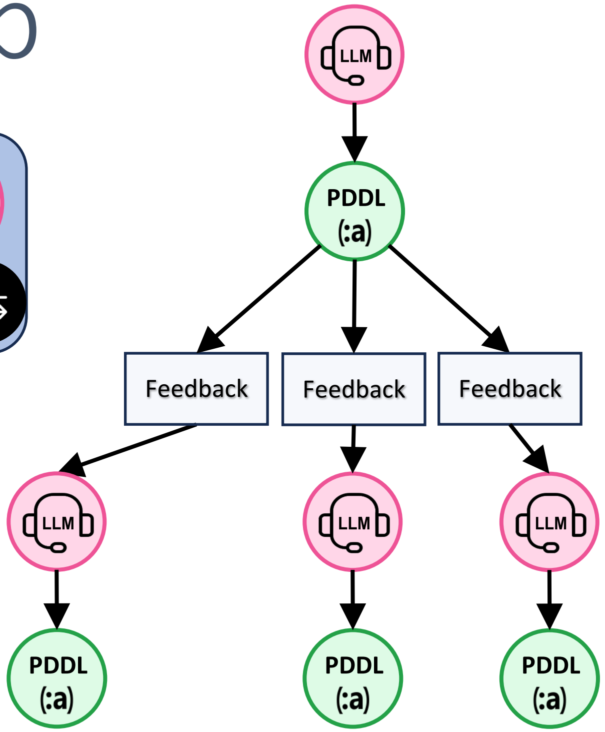


We can branch on the feedback given to the model!

Oswald et al, WM@ICLR 2026

PDDL Generation: Beyond Single Loop



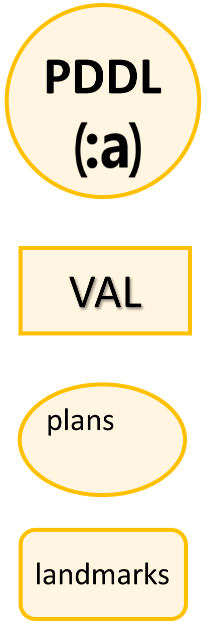


Semantic feedback – many forms

This sequence of actions should have been a plan

This sequence of actions should have not been a plan

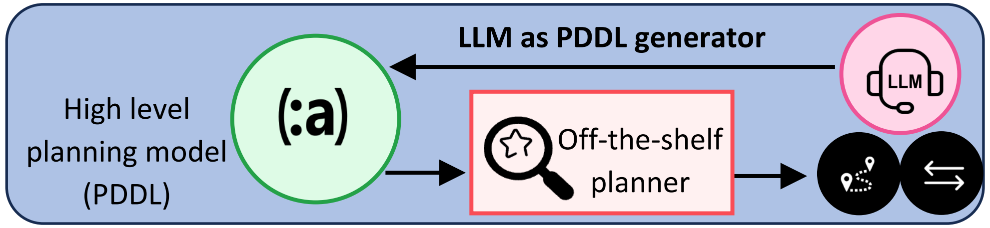
This set of actions should have been a landmark

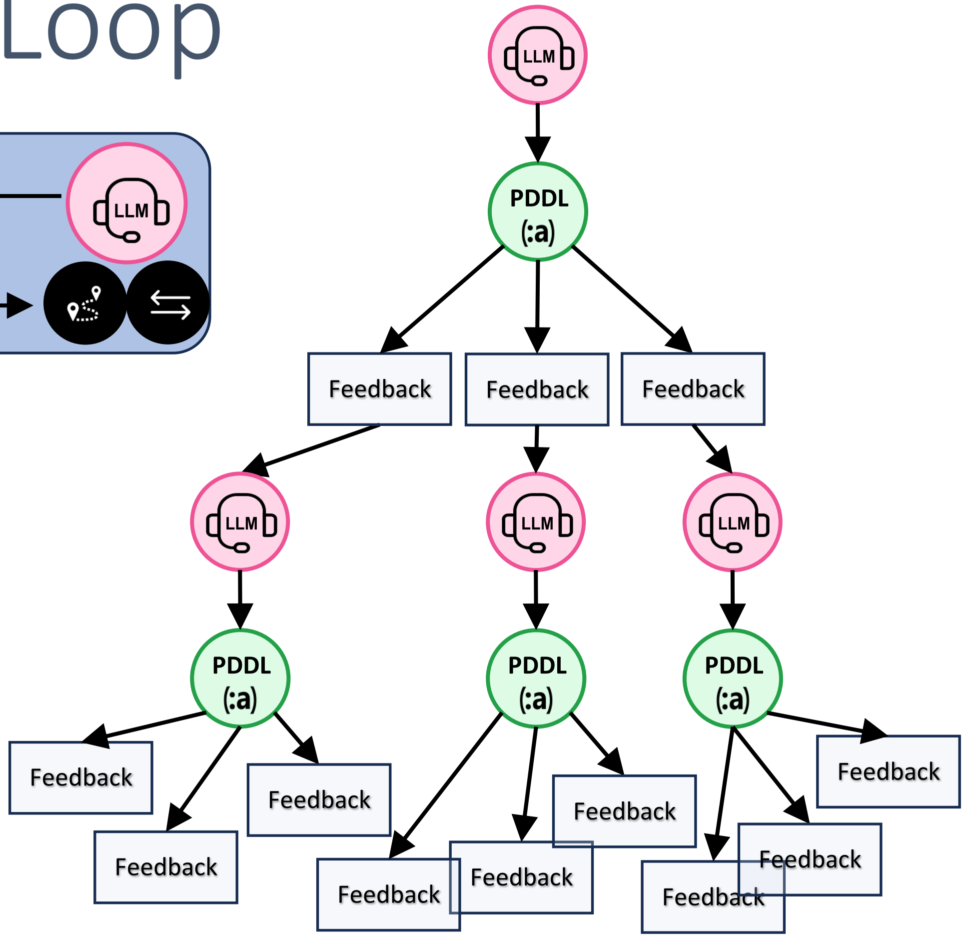


We can branch on the feedback given to the model!

Oswald et al, WM@ICLR 2026

PDDL Generation: Beyond Single Loop





PDDL (:a)

VAL

plans

landmarks

Semantic feedback – many forms

This sequence of actions should have been a plan

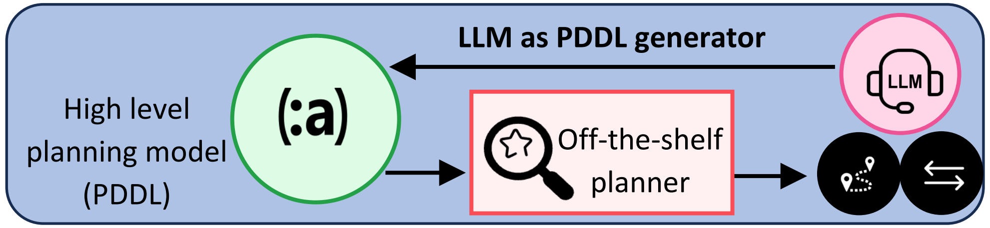
This sequence of actions should have not been a plan

This set of actions should have been a landmark

We can branch on the feedback given to the model!

Oswald et al, WM@ICLR 2026

PDDL Generation: Beyond Single Loop

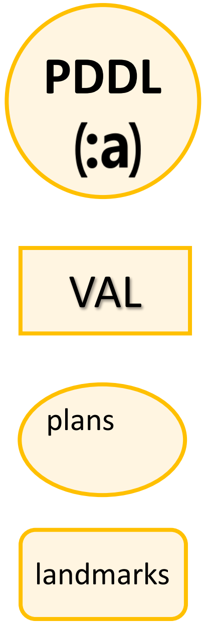


Semantic feedback – many forms

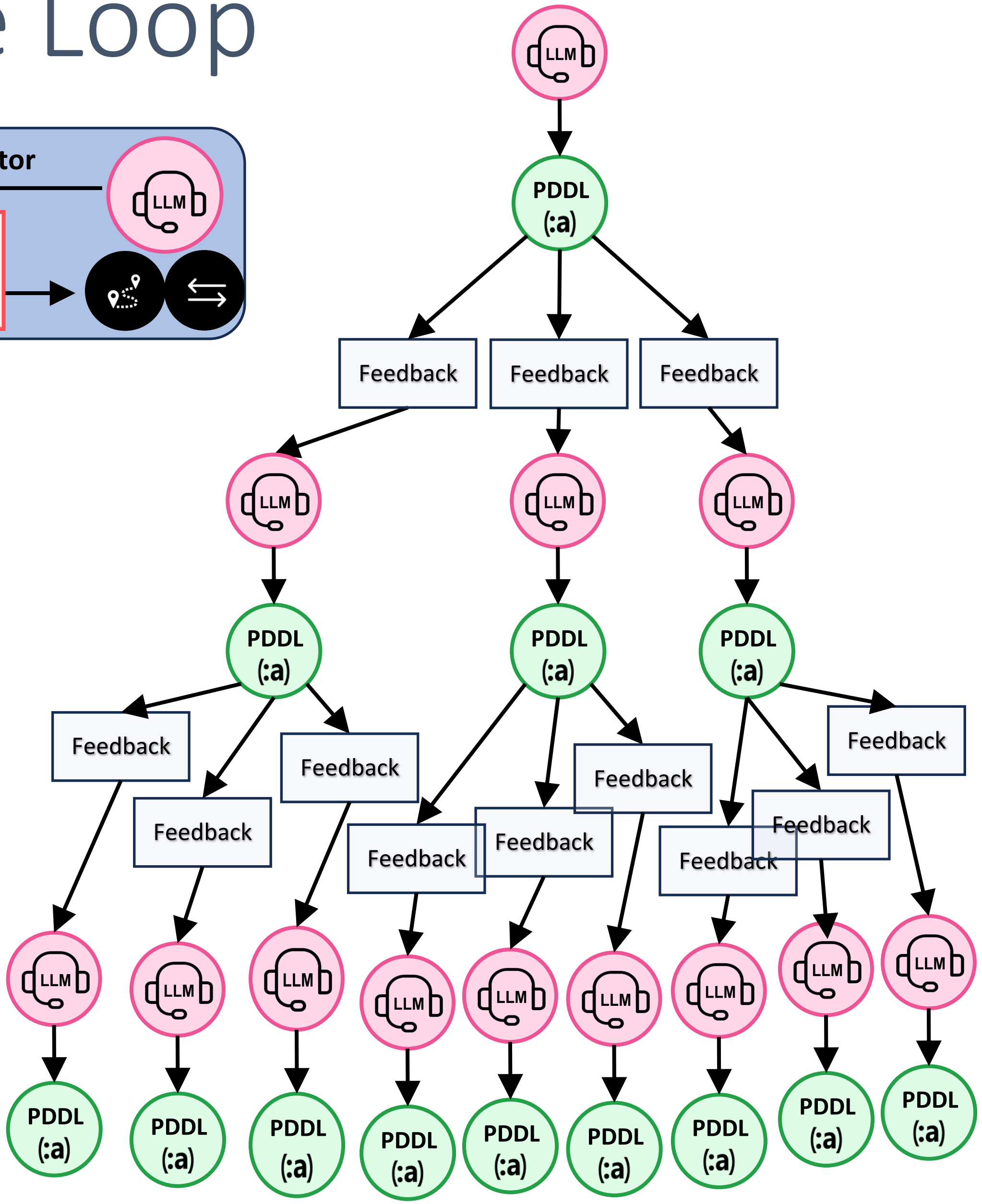
This sequence of actions should have been a plan

This sequence of actions should have not been a plan

This set of actions should have been a landmark



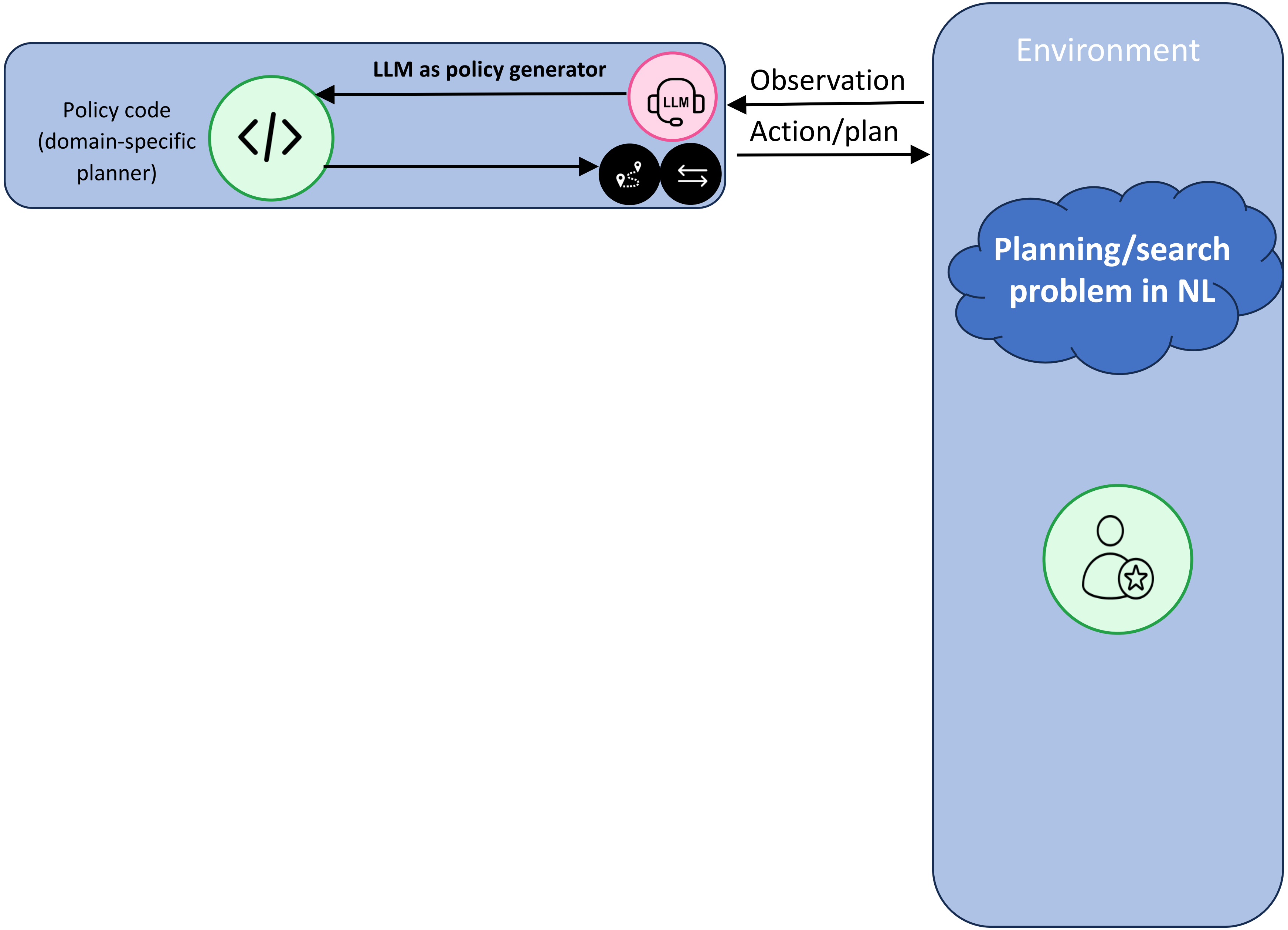
We can branch on the feedback given to the model!



Oswald et al, WM@ICLR 2026

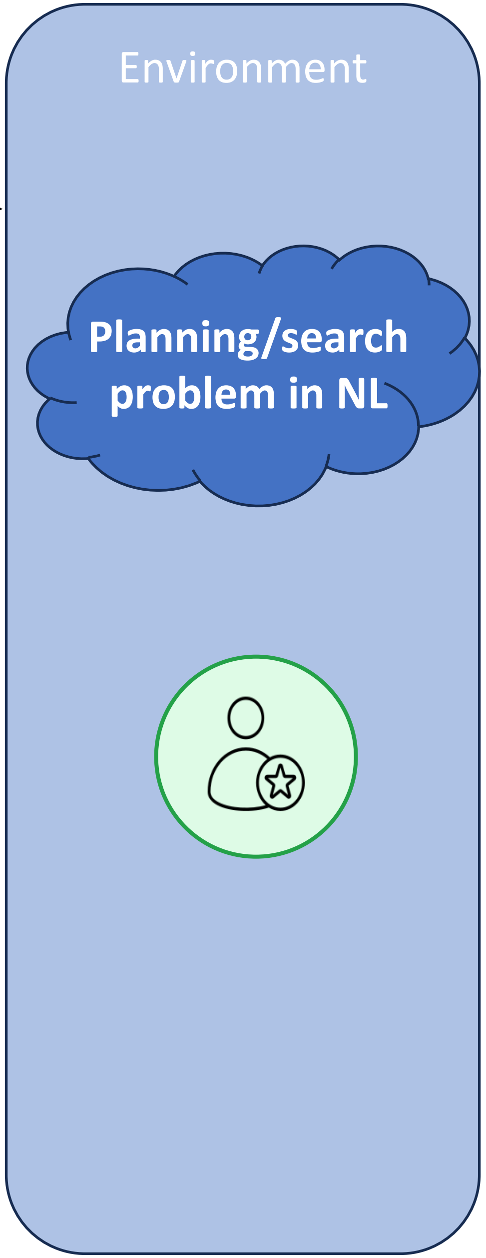
Generalized Policy Generation

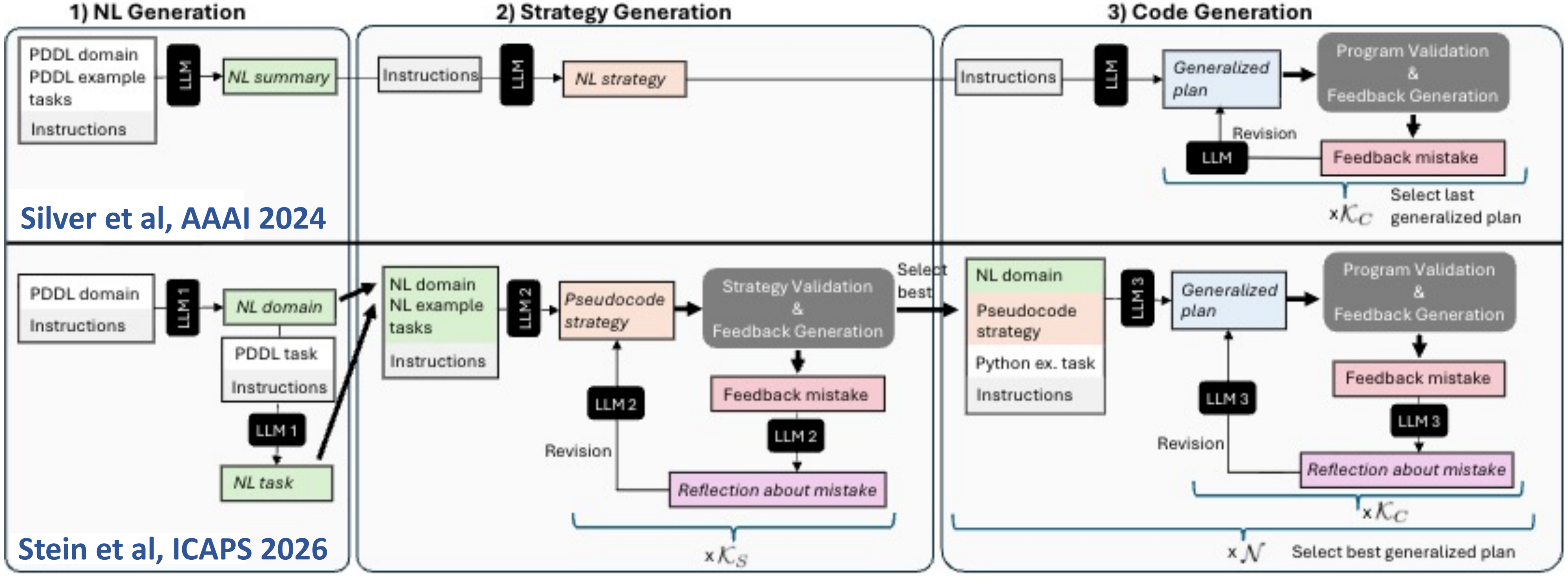
Silver et al, AAAI 2024
Hodel, BSc Thesis 2024
Stein et al, ICAPS 2026
Sohrabi et al, LM4Plan@ICAPS 2026



Generalized Policy Generation

Silver et al, AAAI 2024
Hodel, BSc Thesis 2024
Stein et al, ICAPS 2026
Sohrabi et al, LM4Plan@ICAPS 2026



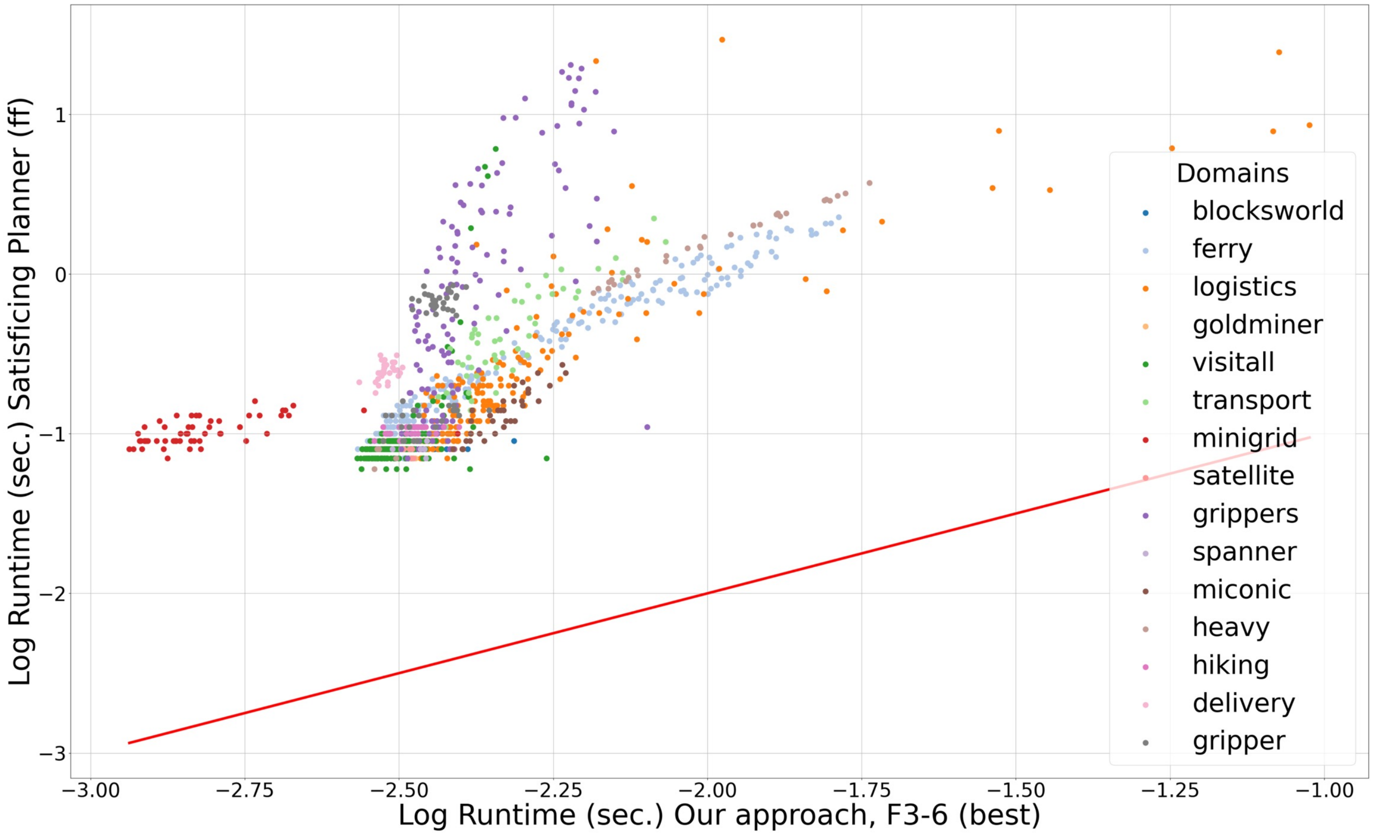


Generalized Policy Generation

Domains	Avg coverage three runs							Coverage best run							Cov. symbolic			
	Sil	Bas	F5-3	F3-6	-MC	-SD	-CR	Sil	Bas	F5-3	F3-6	-MC	-SD	-CR	lm$^{=}$	ff$^{=}$	lm	ff
Domains from Silver et al. (2024)																		
delivery	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	100	100	**100**	**100**	**100**	0	100	0	100
ferry	33	**100**	**100**	**100**	35	**100**	**100**	**100**	**100**	100	100	**100**	**100**	**100**	31	100	43	100
gripper	79	**100**	**100**	88	**100**	**100**	**100**	**100**	**100**	100	100	**100**	**100**	**100**	15	100	40	100
heavy	**100**	67	**100**	**100**	**100**	**100**	**100**	**100**	**100**	100	100	**100**	**100**	**100**	100	100	100	100
hiking	**100**	0	33	67	0	0	67	**100**	0	100	100	0	0	**100**	100	100	100	100
miconic	11	4	**68**	33	0	1	4	32	12	100	100	0	3	12	56	100	62	100
spanner	0	6	33	**67**	33	67	33	0	15	100	100	**100**	**100**	**100**	15	15	41	59

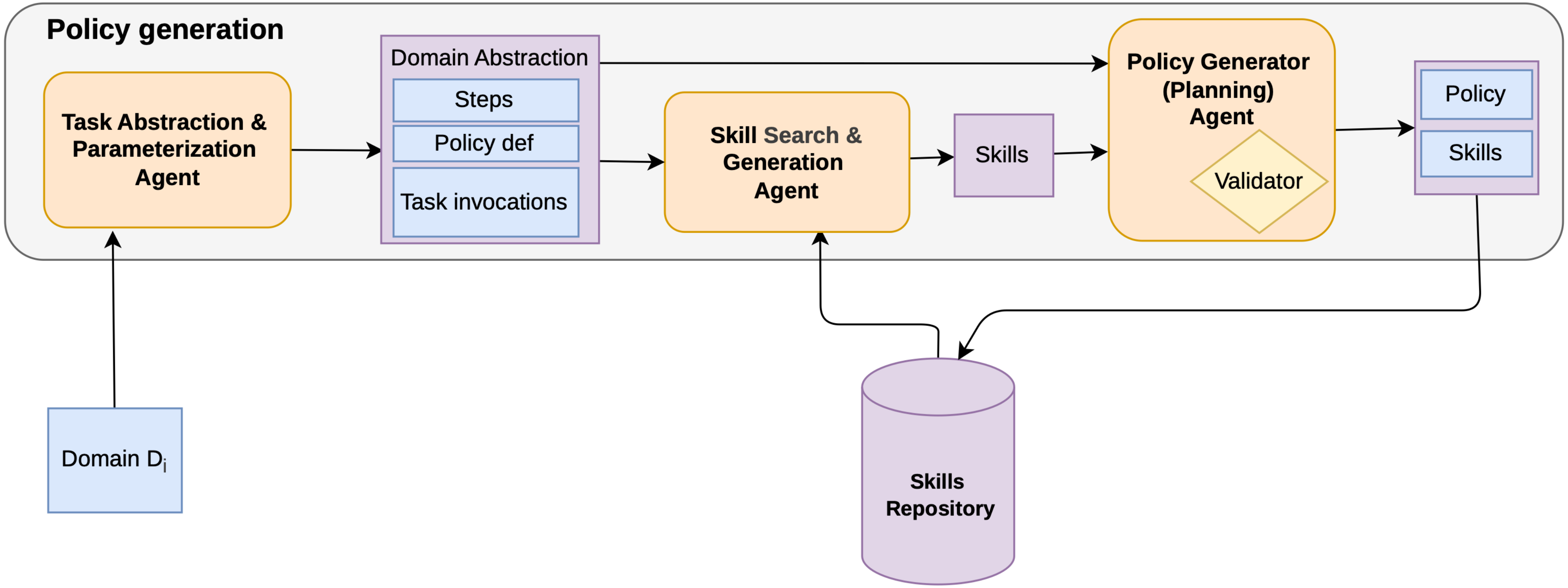
Stein et al, ICAPS 2026

Generalized Policy Generation



Stein et al, ICAPS 2026

Generalized Policy Generation with Reuse

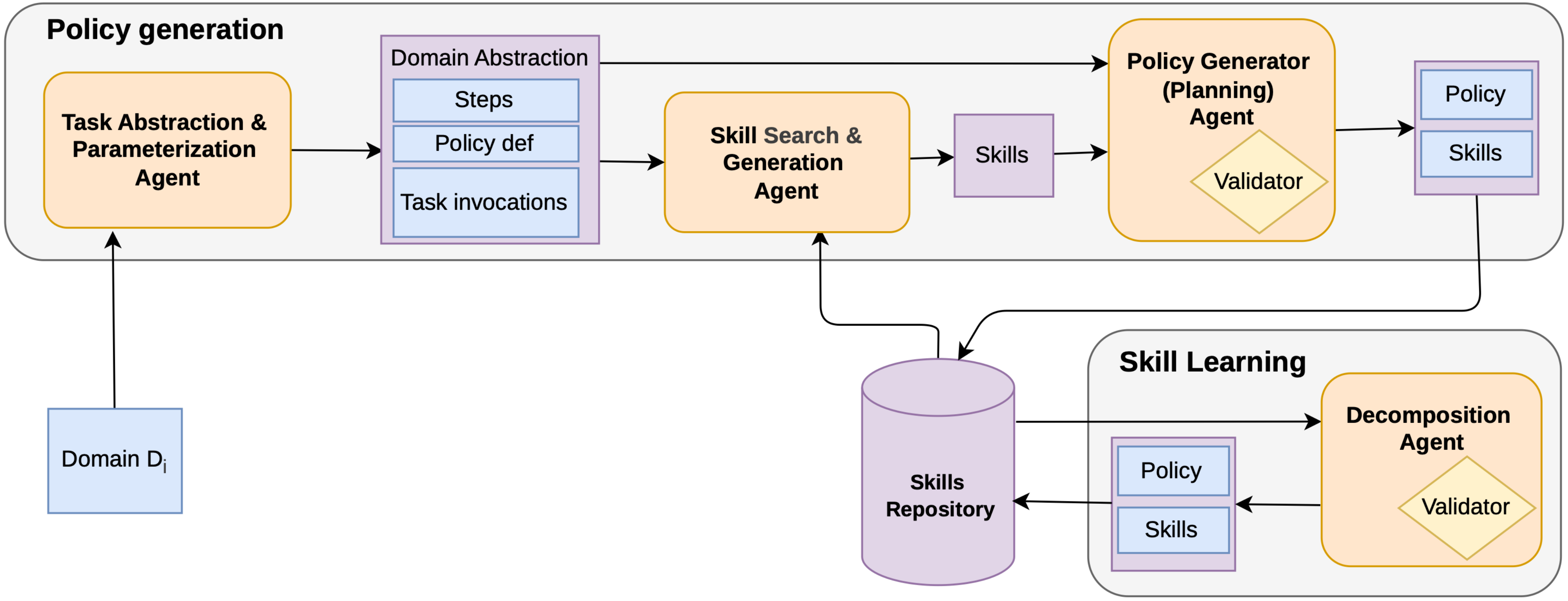


Three stages, each guarded by validation:

Generate policy: abstract tasks, infer signature, bind parameters, retrieve skills, synthesize code

Sohrabi et al, LM4Plan@ICAPS 2026

Generalized Policy Generation with Reuse

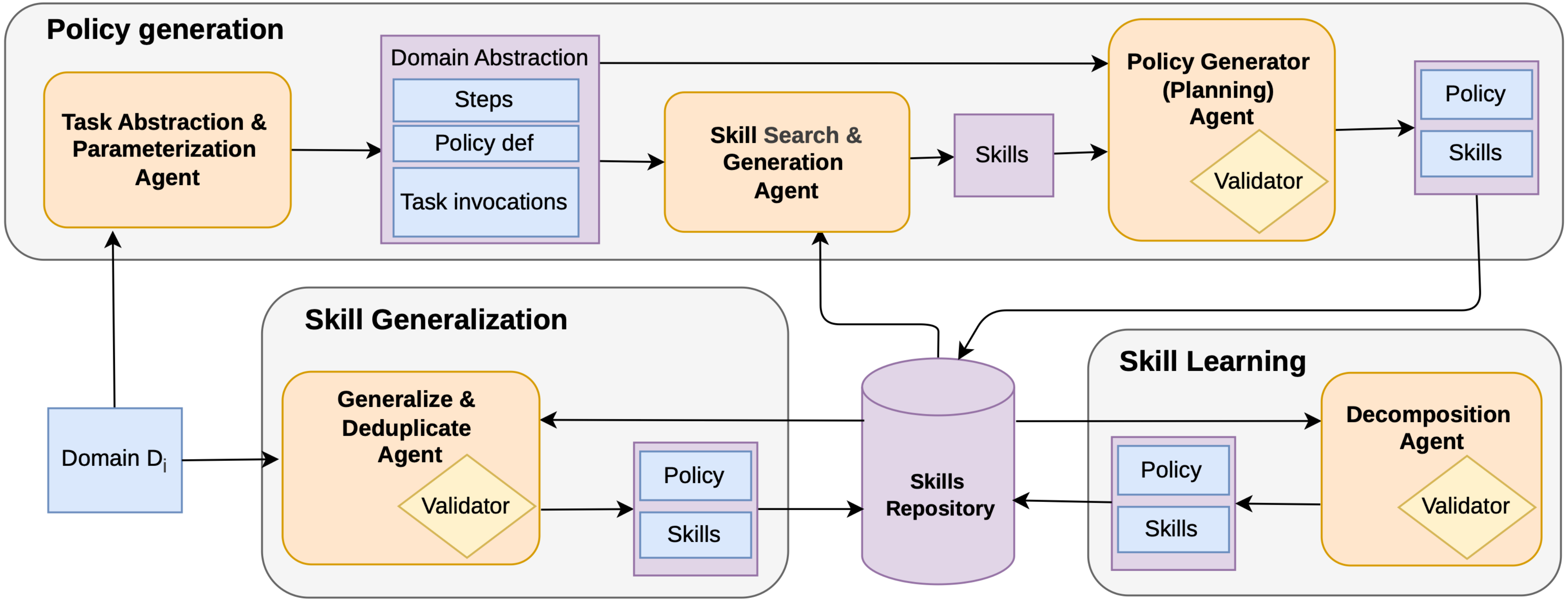


Three stages, each guarded by validation:

Learn Skills: decompose successful policies into reusable functions and revalidate updated policies

Sohrabi et al, LM4Plan@ICAPS 2026

Generalized Policy Generation with Reuse



Three stages, each guarded by validation:

Generalize Skills: cluster, deduplicate, abstract, and validate before adding to the repository

Sohrabi et al, LM4Plan@ICAPS 2026

Generalized Policy Generation with Reuse: Results

Method	Model	Normal		Challenge	
		TGC	SGC	TGC	SGC
Claude Code	Claude Opus 4.5	66.0	56.6	–	–
Smolagents Code	Claude Opus 4.5	70.0	60.4	–	–
OpenAI Solo	Claude Opus 4.5	68.0	56.6	–	–
ReAct	Claude Opus 4.5	61.0	52.8	–	–
ReAct Short	Claude Opus 4.5	64.0	54.7	–	–
Alibaba AgentRL	Qwen3-14B	86.9	80.4	67.6	50.4
IBM CUGA	GPT-4.1	73.2	62.5	57.6	48.2
LOOP	Qwen2.5-32B	72.6	53.6	47.2	28.8
GP (baseline)	GPT-OSS 120B	0.0	0.0	0.7	0.7
GP (baseline)	Claude Sonnet 4.6	96.4	96.4	83.2	82.0
HCL-GP	GPT-OSS 120B	62.5	62.5	41.0	41.0
HCL-GP	Claude Sonnet 4.6	**98.2**	**98.2**	**98.3**	**97.8**

AppWorld

Sohrabi et al, LM4Plan@ICAPS 2026

Generalized Policy Generation with Reuse: Results

Method	Model	Normal		Challenge	
		TGC	SGC	TGC	SGC
Claude Code	Claude Opus 4.5	66.0	56.6	–	–
Smolagents Code	Claude Opus 4.5	70.0	60.4	–	–
OpenAI Solo	Claude Opus 4.5	68.0	56.6	–	–
ReAct	Claude Opus 4.5	61.0	52.8	–	–
ReAct Short	Claude Opus 4.5	64.0	54.7	–	–
Alibaba AgentRL	Qwen3-14B	86.9	80.4	67.6	50.4
IBM CUGA	GPT-4.1	73.2	62.5	57.6	48.2
LOOP	Qwen2.5-32B	72.6	53.6	47.2	28.8
GP (baseline)	GPT-OSS 120B	0.0	0.0	0.7	0.7
GP (baseline)	Claude Sonnet 4.6	96.4	96.4	83.2	82.0
HCL-GP	GPT-OSS 120B	62.5	62.5	41.0	41.0
HCL-GP	Claude Sonnet 4.6	**98.2**	**98.2**	**98.3**	**97.8**

AppWorld

Sohrabi et al, LM4Plan@ICAPS 2026

Generalized Policy Generation with Reuse: Results

Method	Model	Normal TGC	Normal SGC	Challenge TGC	Challenge SGC
Claude Code	Claude Opus 4.5	66.0	56.6	–	–
Smolagents Code	Claude Opus 4.5	70.0	60.4	–	–
OpenAI Solo	Claude Opus 4.5	68.0	56.6	–	–
ReAct	Claude Opus 4.5	61.0	52.8	–	–
ReAct Short	Claude Opus 4.5	64.0	54.7	–	–
Alibaba AgentRL	Qwen3-14B	86.9	80.4	67.6	50.4
IBM CUGA	GPT-4.1	73.2	62.5	57.6	48.2
LOOP	Qwen2.5-32B	72.6	53.6	47.2	28.8
GP (baseline)	GPT-OSS 120B	0.0	0.0	0.7	0.7
GP (baseline)	Claude Sonnet 4.6	96.4	96.4	83.2	82.0
HCL-GP	GPT-OSS 120B	62.5	62.5	41.0	41.0
HCL-GP	Claude Sonnet 4.6	**98.2**	**98.2**	**98.3**	**97.8**

AppWorld

Sohrabi et al, LM4Plan@ICAPS 2026

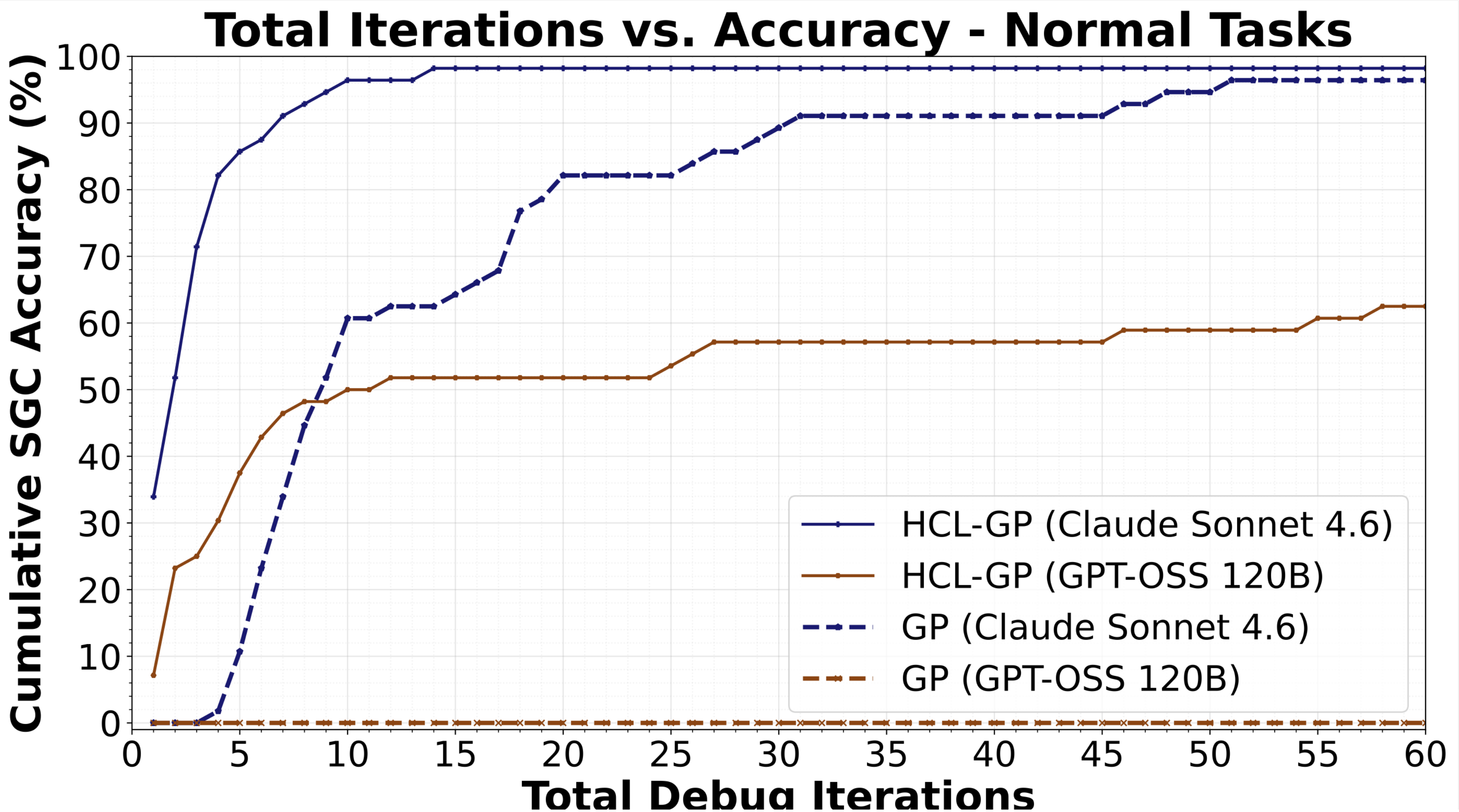
Generalized Policy Generation with Reuse: Results

Method	Model	Normal TGC	Normal SGC	Challenge TGC	Challenge SGC
Claude Code	Claude Opus 4.5	66.0	56.6	–	–
Smolagents Code	Claude Opus 4.5	70.0	60.4	–	–
OpenAI Solo	Claude Opus 4.5	68.0	56.6	–	–
ReAct	Claude Opus 4.5	61.0	52.8	–	–
ReAct Short	Claude Opus 4.5	64.0	54.7	–	–
Alibaba AgentRL	Qwen3-14B	86.9	80.4	67.6	50.4
IBM CUGA	GPT-4.1	73.2	62.5	57.6	48.2
LOOP	Qwen2.5-32B	72.6	53.6	47.2	28.8
GP (baseline)	GPT-OSS 120B	0.0	0.0	0.7	0.7
GP (baseline)	Claude Sonnet 4.6	96.4	96.4	83.2	82.0
HCL-GP	GPT-OSS 120B	62.5	62.5	41.0	41.0
HCL-GP	Claude Sonnet 4.6	**98.2**	**98.2**	**98.3**	**97.8**

AppWorld

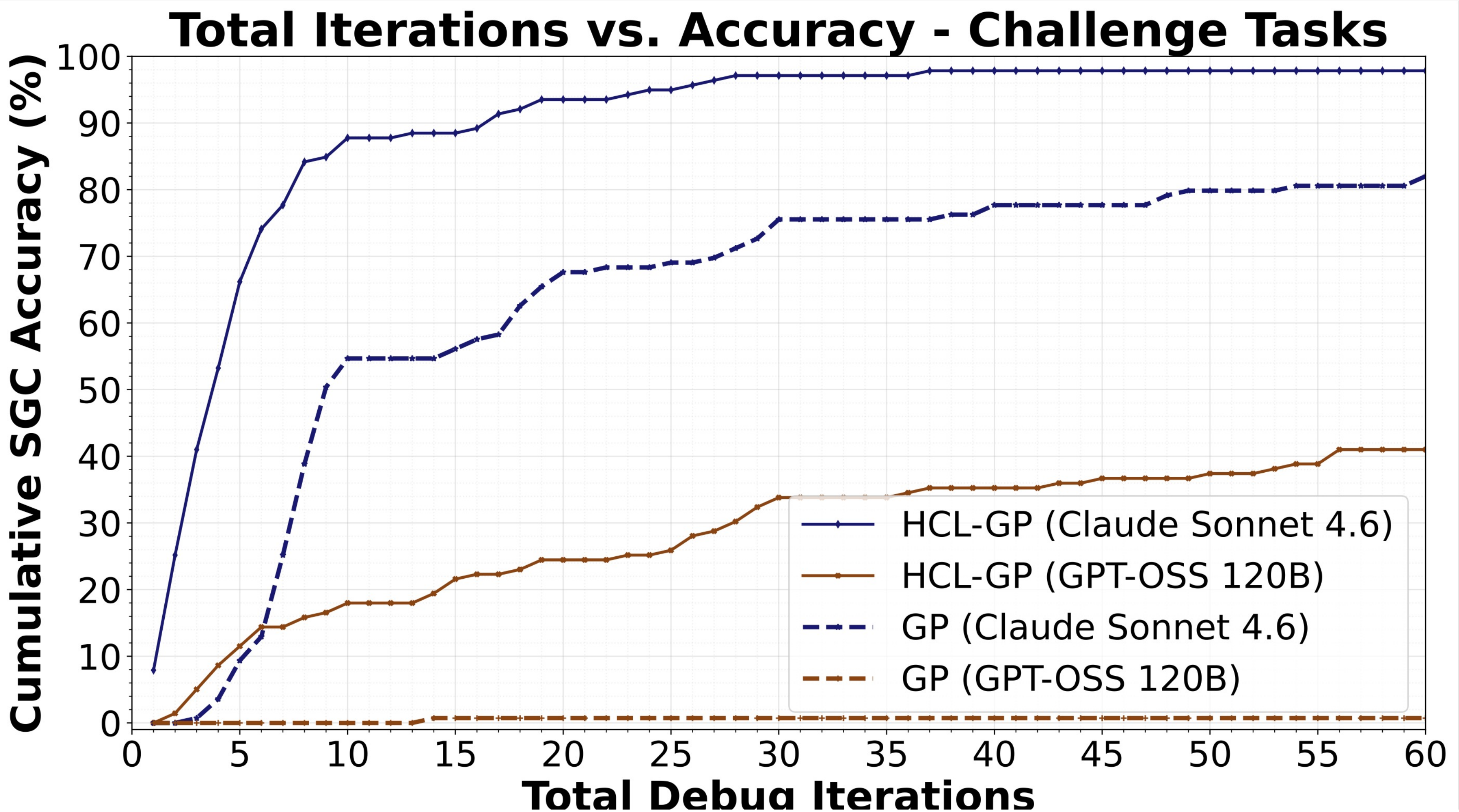
Sohrabi et al, LM4Plan@ICAPS 2026

Generalized Policy Generation with Reuse: Results



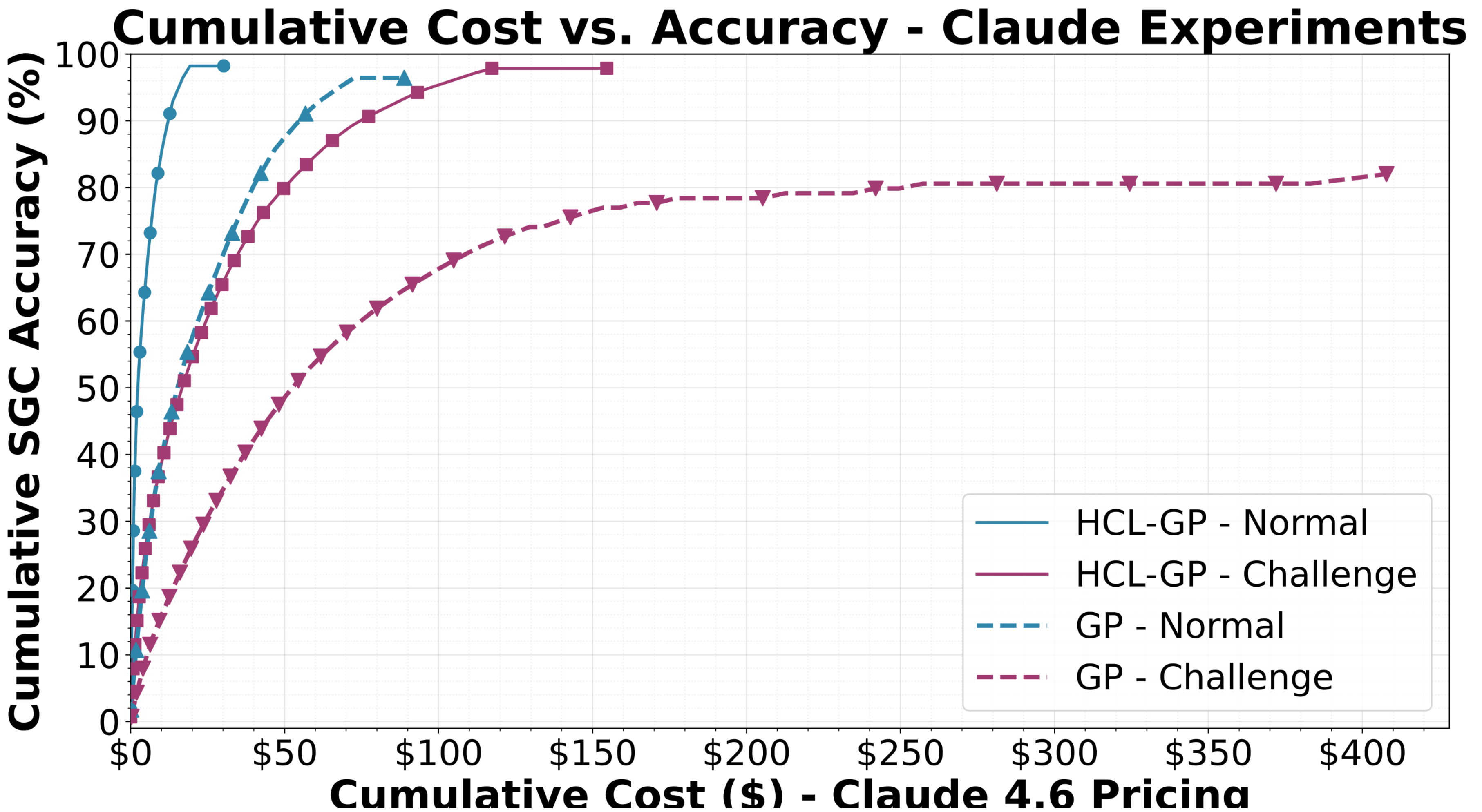
Sohrabi et al, LM4Plan@ICAPS 2026

Generalized Policy Generation with Reuse: Results



Sohrabi et al, LM4Plan@ICAPS 2026

Generalized Policy Generation with Reuse: Results



Sohrabi et al, LM4Plan@ICAPS 2026


QUESTIONS?