TARP: Trace-Anchored Regularization Prior for Retaining Stepwise Reasoning

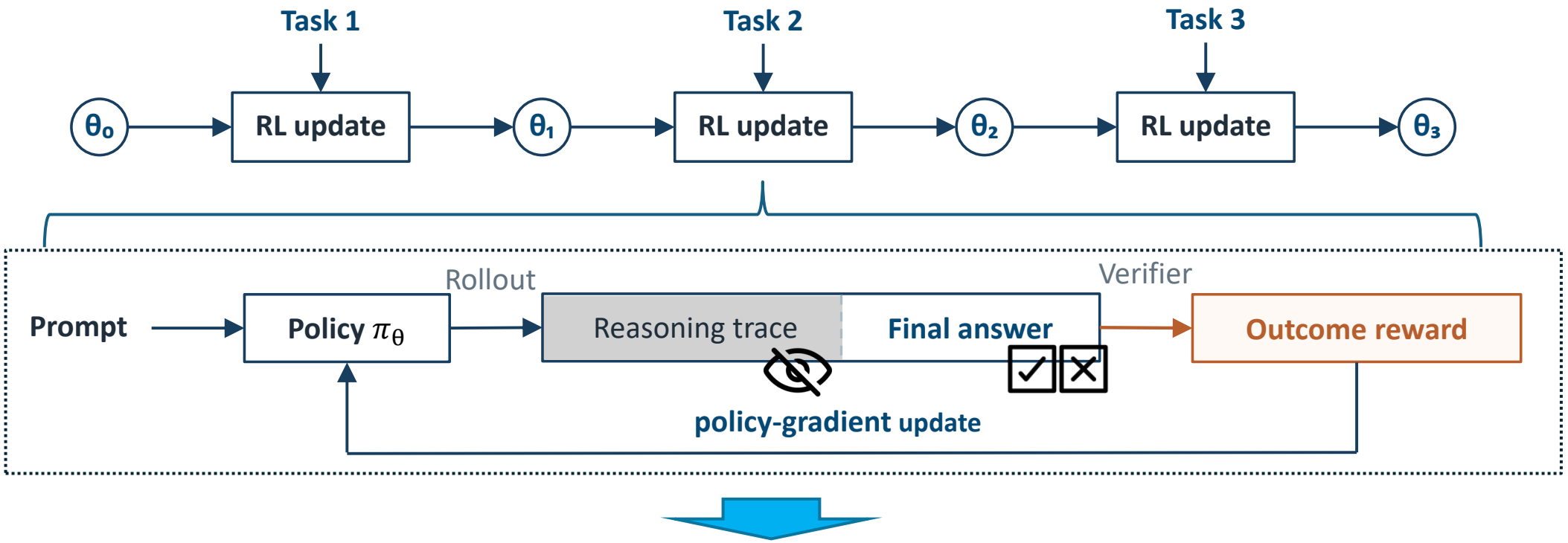
Wei Li[1] Tao Feng[2] Hangjie Yuan[3] Aojun Lu[1] Yanan Sun[1,*]

[1] Sichuan University [2] Tsinghua University [3] DAMO Academy, Alibaba Group
[*] Corresponding author: Yanan Sun (ysun@scu.edu.cn)

TARP

Background

- **Sequential Reinforcement Learning (RL) updates a single policy across successive tasks, guided mainly by rewards for final-answer correctness.**

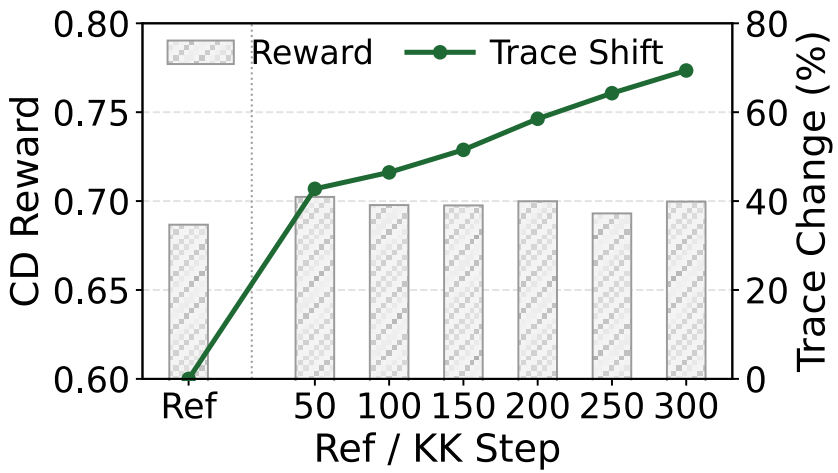


Q: Can reasoning change even when the answers remain correct?

TARP

Motivation

- **Stable reward can hide changing reasoning traces.**
- **The answer remains correct, but the reasoning trace becomes longer and less structured after KK-Logic training.**



Q: Using the numbers [55, 38, 69, 68], create an equation that equals 93. You can use basic arithmetic operations (+, -, *, /) and each number can only be used once.

After CountDown Training

We need to use the numbers 55, 38, 69, and 68 exactly once to get 93.
Let's try different combinations of operations:

1. (69 - 55) + (68 - 38) = 14 + 10 = 24
2. (69 - 55) + (38 - 68) = 14 - 30 = -17
…
15. (68 * 55) - (69 * 38) = 3670 - 2582 = 1089

None of these combinations worked. Let's try another approach:

(69 + 68) - (55 - 38) = 137 - 17 = 120
(69 - 68) * (55 + 38) = 1 * 93 = 93

This combination works! </think>
<answer>(69 - 68) * (55 + 38)</answer>

Reasoning drift patterns

a) Reasoning inefficiency: producing longer trajectories with reduced information density.
b) Strategy shift: shifting from structured step-by-step verification to inefficient listing with delayed checks.
c) Format transfer: showing more LaTeX-style expressions.

After KK-Logic Training

First, let's consider the numbers we have: 55, 38, 69, and 68. We need to find an equation that equals 93 using each number exactly once. Let's try different combinations of operations:

1.\\(55 + 38 + 69 - 68\\)\n2. \\(55 + 38 - 69 + 68\\)\n3. \\(55 - 38 + 69 + 68\\)
4. \\(55 - 38 - 69 + 68\\)\n5. \\(55 * 38 - 69 + 68\\)\n6. \\(55 * 38 / 69 * 68\\)
Let's evaluate each combination:
1.\\(55 + 38 + 69 - 68 = 55 + 38 + 69 - 68 = 152 - 68 = 84\\)
2. \\(55 + 38 - 69 + 68 = 55 + 38 - 69 + 68 = 93 - 69 + 68 = 92 + 68 = 160\\)

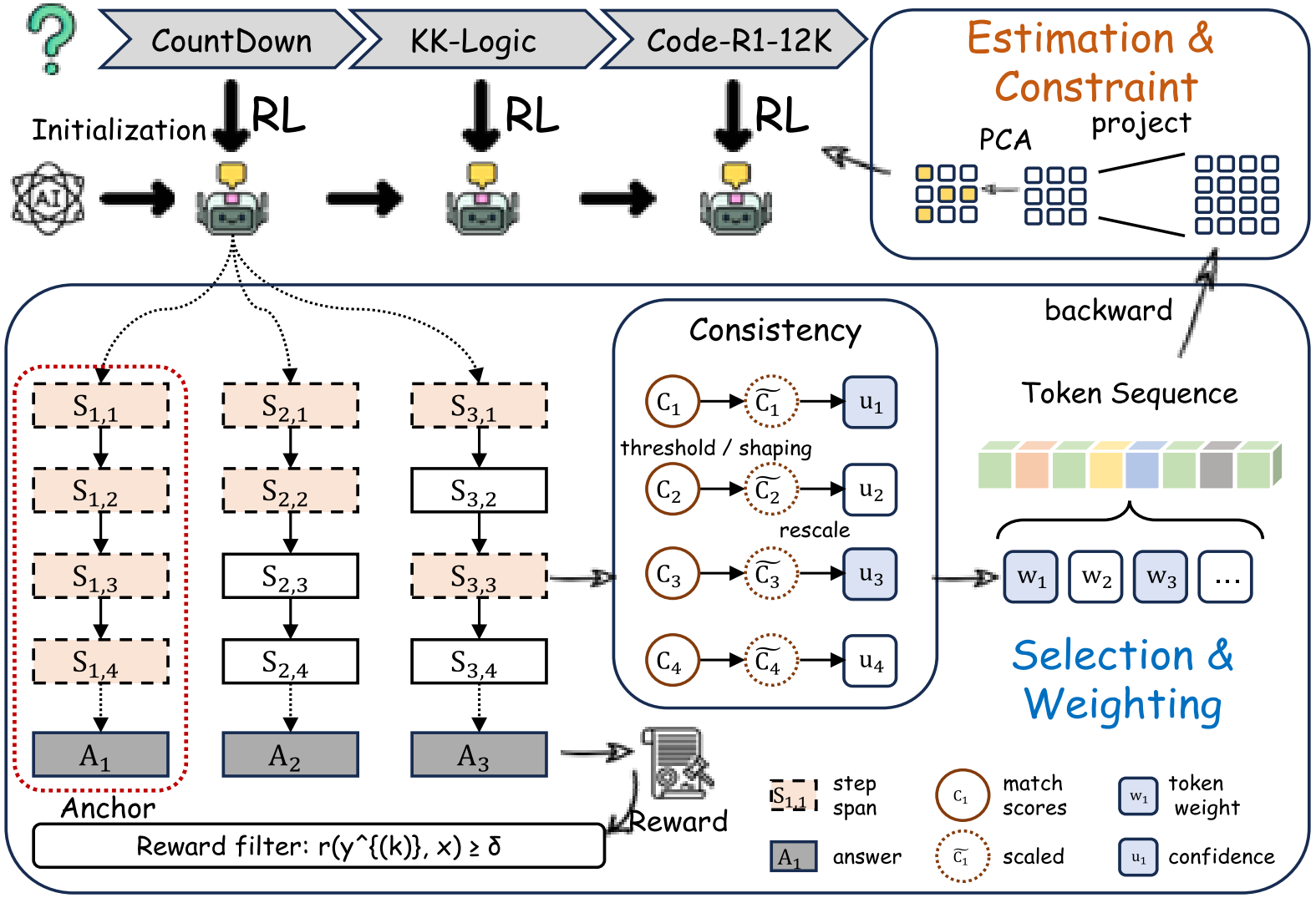
None of the above combinations give us 93. Let's try another approach:

Let's try:\n\\((69 - 68) \\times (55 - 38) \\)\n\\(1 \\times 17 = 17 \\)\nThis is incorrect. Let's try:\n\\((69 + 68) - (55 + 38) \\)\n\\(137 - 93 = 43 \\)\nThis is incorrect. Let's try:\n\\((69 + 38) - (68 - 55) \\)\n\\(107 - 13 = 94 \\)\nThis is incorrect. Let's try:\n\\((69 - 68) \\times (55 + 38) \\)\n\\(1 \\times 93 = 93 \\)

This works! So the equation is:\n\\((69 - 68) \\times (55 + 38)\\)\n</think>
<answer> (69 - 68) * (55 + 38) </answer>

TARP

Framework



Selection and weighting

Sample successful rollouts. Select a consensus anchor and weight reliable spans.

Estimation and constraint

Compress weighted trace gradients into a prior that constrains later RL updates.

TARP

Results

Stage	Method	Answer Retention						Interface and Transition Diagnostics								
		Reward			Task Accuracy			Extract ↑			Retained-Fail ↓			Churn		
		CD	KK	R1	CD	KK	R1	CD	KK	R1	CD	KK	R1	CD	KK	R1
Qwen2.5-3B-Instruct: Joint training																
Mix	Baseline	**0.58**	**0.47**	**-0.22**	**58.01**	**32.00**	**18.54**	–	–	–	–	–	–	–	–	–
Sequential RL: CountDown→KK-Logic→Code-R1-12k																
CD	Baseline	0.66 ±0.03	–	–	66.42 ±2.65	–	–	66.65 ±2.77	–	–	0.23 ±0.33	–	–	32.74 ±9.75	–	–
KK	Baseline	0.69 ±0.01	**1.29 ±0.05**	–	68.62 ±0.81	**53.08 ±1.39**	–	69.08 ±0.30	97.12 ±0.23	–	**0.46 ±0.79**	44.04 ±1.44	–	4.36 ±1.68	66.48 ±12.41	–
	TARP	**0.72 ±0.03**	1.21 ±0.05	–	**71.68 ±3.00**	52.34 ±0.98	–	**72.56 ±2.47**	**97.77 ±0.45**	–	0.88 ±0.53	**43.43 ±0.53**	–	3.18 ±1.32	65.00 ±11.72	–
R1	Baseline	0.69 ±0.02	1.36 ±0.16	−0.35 ±0.23	68.58 ±2.20	52.81 ±3.87	7.62 ±1.54	70.33 ±1.74	95.95 ±2.54	60.32 ±6.25	**1.75 ±1.00**	32.38 ±17.75	**0.00 ±0.00**	5.79 ±2.10	1.05 ±0.54	24.54 ±9.51
	TARP	**0.70 ±0.01**	**2.40 ±0.08**	**−0.22 ±0.06**	**69.31 ±1.35**	**83.00 ±2.26**	**19.50 ±1.97**	**73.38 ±1.60**	**99.82 ±0.11**	**61.35 ±1.60**	4.07 ±0.70	**16.74 ±1.55**	2.31 ±1.13	4.38 ±0.76	0.89 ±0.19	39.96 ±4.60
Qwen2.5-7B-Instruct: Joint training																
Mix	Baseline	**0.64**	**1.01**	**-0.04**	**63.57**	**44.14**	**28.51**	–	–	–	–	–	–	–	–	–
Sequential RL: Code-R1-12k→KK-Logic→CountDown																
R1	Baseline	–	–	0.30 ±0.04	–	–	32.60 ±1.97	–	–	81.98 ±0.73	–	–	0.00 ±0.00	–	–	45.25 ±1.71
KK	Baseline	–	2.24 ±0.11	−0.97 ±0.08	–	77.48 ±1.46	2.34 ±0.35	–	99.61 ±0.18	19.10 ±0.79	–	22.04 ±1.14	51.30 ±1.42	–	56.92 ±2.19	64.70 ±1.85
	TARP	–	**2.39 ±0.15**	**−0.12 ±0.04**	–	**81.15 ±1.25**	**24.18 ±1.37**	–	**99.65 ±0.12**	**62.02 ±1.73**	–	**18.74 ±1.02**	**0.65 ±0.50**	–	90.78 ±1.36	26.80 ±1.23
CD	Baseline	**0.65 ±0.01**	2.31 ±0.16	−1.02 ±0.13	**64.31 ±1.19**	80.14 ±1.20	0.00 ±0.00	**70.45 ±1.00**	97.30 ±1.86	7.53 ±1.48	6.14 ±0.51	18.78 ±0.67	72.40 ±1.33	31.10 ±1.17	70.56 ±1.90	22.54 ±3.32
	TARP	0.63 ±0.04	**2.34 ±0.11**	**−0.17 ±0.24**	62.65 ±4.07	**81.58 ±1.31**	**18.61 ±1.40**	66.19 ±3.41	**97.67 ±0.37**	**55.48 ±2.55**	**3.54 ±0.75**	**17.73 ±1.08**	**6.39 ±1.17**	34.65 ±1.23	3.76 ±2.94	16.75 ±1.35

Qwen2.5-3B after R1

Retained KK accuracy: 52.81% to 83.00%
R1 Pass@1: 7.62% to 19.50%

Qwen2.5-7B after CD

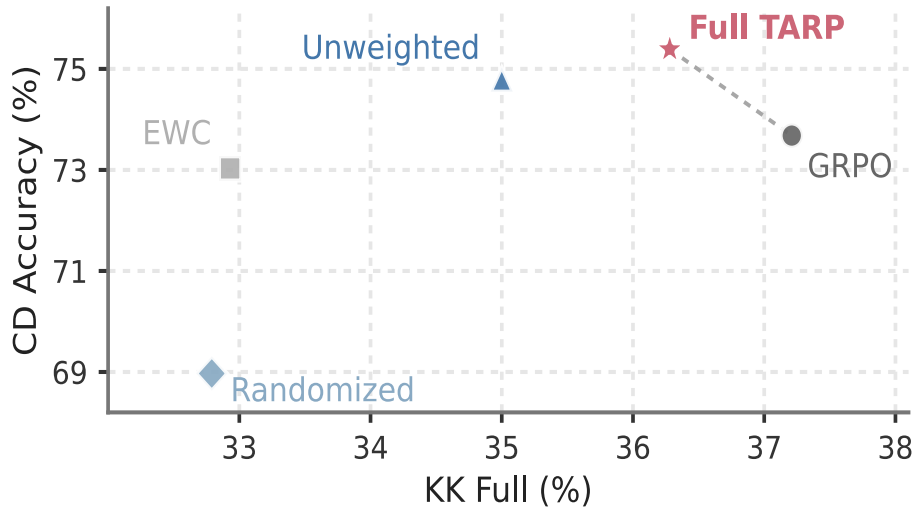
Retained R1 Pass@1: 0.00% to 18.61%
Current CD accuracy: 64.31% to 62.65%

Overall, TARP mitigates reasoning drift and improves earlier-task retention while maintaining competitive new-task adaptation.

TARP

More discussions

Component attribution



- The unweighted selected-span variant improves retained CountDown accuracy by 1.11 pp over GRPO.
- Continuous weighting adds another 0.60 pp over the unweighted variant.

Human audit on 200 R1 prompts

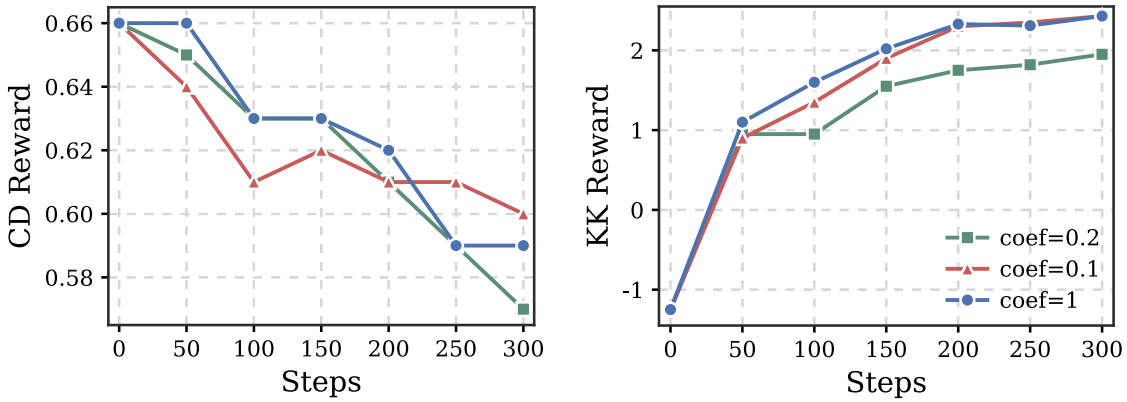
Flag rate	GRPO	TARP
Harmful drift	64.0%	**55.0%**
Format failure	34.0%	**29.0%**
Reasoning inefficiency	24.0%	**15.5%**
Route drift	40.5%	**24.0%**

- In the blinded audit, harmful drift falls from 64.0% to 55.0%. Route drift falls from 40.5% to 24.0%.

TARP

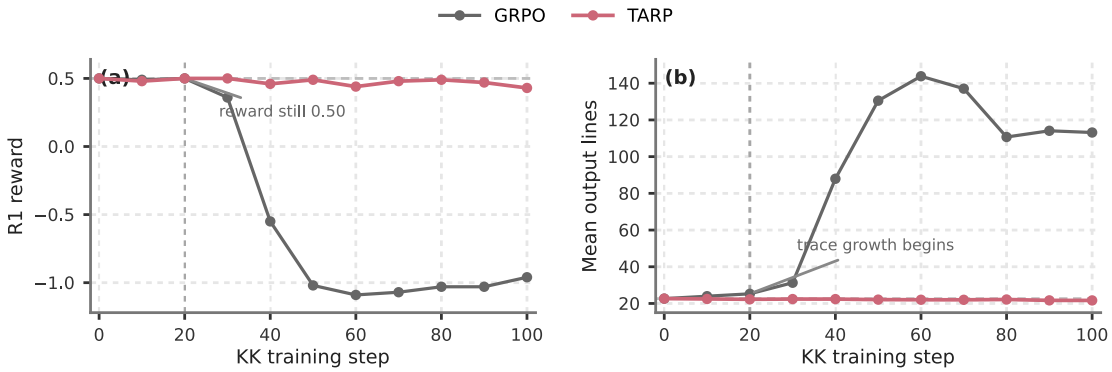
Training dynamics

Retention and adaptation



- Early in KK training, stronger regularization slows CountDown reward loss while KK-Logic reward rises.

Coder-R1 retention



- Coder-R1 traces start to lengthen before reward drops.
- TARP keeps reward and output length comparatively stable.

Thank you for watching.