# Multi-step reasoning: Synthetic benchmarks, expressivity, and stability

Hongyang R. Zhang

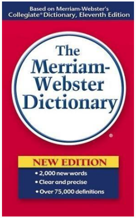
Assistant Professor, Northeastern University
Khoury College of Computer Sciences

August 16, 2026

## What is reasoning?

According to the Merriam-Webster dictionary, reasoning is defined as the drawing of inferences or conclusions through the use of reason. Two examples of reasoning used in a sentence:

- 1st example: Could you explain your reasoning?
- 2nd example: They told everyone the reasoning behind the decision.




For the scope of today's talk, we shall refer to reasoning as the ability to solve a complex problem by decomposing this problem into sub-problems

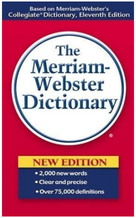
- Why are we interested in this question? The ability of LLMs to solve math problems is remarkable. For an example, see openai's recent article on ten advances in mathematics and theoretical computer science

**The focus of our work:** (multitask) **algorithmic reasoning** [VBB+22; MMI+24]: they provide a synthetic benchmark to evaluate reasoning, since we know the correct reasoning process at every single step.

## What is reasoning?

According to the Merriam-Webster dictionary, reasoning is defined as the drawing of inferences or conclusions through the use of reason. Two examples of reasoning used in a sentence:

- 1st example: Could you explain your reasoning?
- 2nd example: They told everyone the reasoning behind the decision.

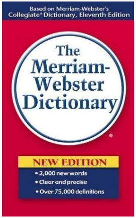



For the scope of today's talk, we shall refer to reasoning as the ability to solve a complex problem by decomposing this problem into sub-problems

- Why are we interested in this question? The ability of LLMs to solve math problems is remarkable. For an example, see openai's recent article on ten advances in mathematics and theoretical computer science

**The focus of our work:** (multitask) **algorithmic reasoning** [VBB+22; MMI+24]: they provide a synthetic benchmark to evaluate reasoning, since we know the correct reasoning process at every single step.

## What is reasoning?




According to the Merriam-Webster dictionary, reasoning is defined as the drawing of inferences or conclusions through the use of reason. Two examples of reasoning used in a sentence:

- 1st example: Could you explain your reasoning?
- 2nd example: They told everyone the reasoning behind the decision.

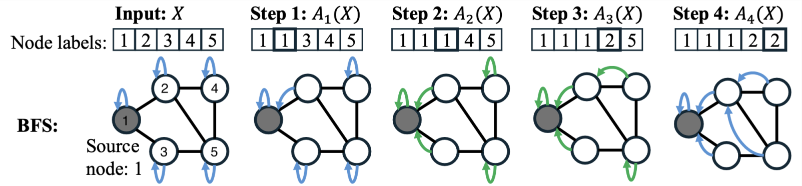
For the scope of today's talk, we shall refer to reasoning as the ability to solve a complex problem by decomposing this problem into sub-problems

- Why are we interested in this question? The ability of LLMs to solve math problems is remarkable. For an example, see openai's recent article on ten advances in mathematics and theoretical computer science

**The focus of our work:** (multitask) **algorithmic reasoning** [VBB+22; MMI+24]: they provide a synthetic benchmark to evaluate reasoning, since we know the correct reasoning process at every single step.

# Example

Here is an example corresponding to encoding the intermediate steps and the final answer of breadth-first search (BFS) on a simple graph

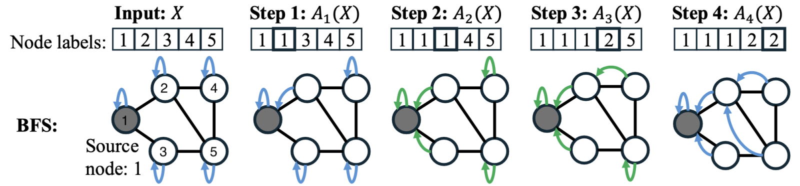



- Starting from node 1, the first step visits node 2. Hence, we update node 2's label to 1, its predecessor.
- Step 2 visits node 3, hence its node label is updated as 1.
- Step 3 visits node 4 (with predecessor 2) and Step 4 visits node 5 (with predecessor 2).

Question: Can a neural network learn to perform an algorithmic reasoning procedure?

## Example

Here is an example corresponding to encoding the intermediate steps and the final answer of breadth-first search (BFS) on a simple graph

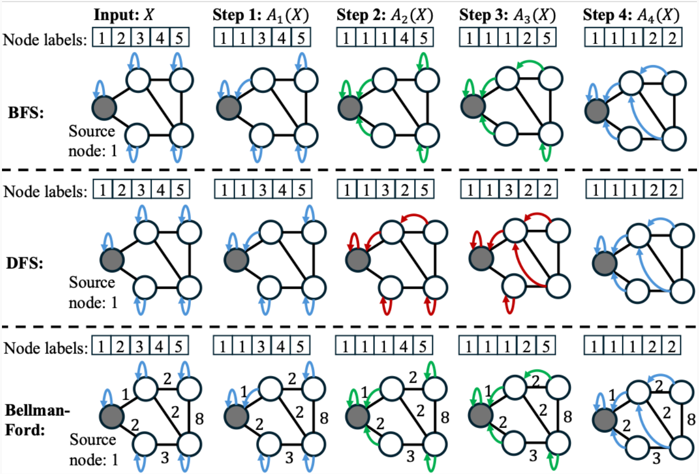



- Starting from node 1, the first step visits node 2. Hence, we update node 2's label to 1, its predecessor.
- Step 2 visits node 3, hence its node label is updated as 1.
- Step 3 visits node 4 (with predecessor 2) and Step 4 visits node 5 (with predecessor 2).

Question: Can a neural network learn to perform an algorithmic reasoning procedure?

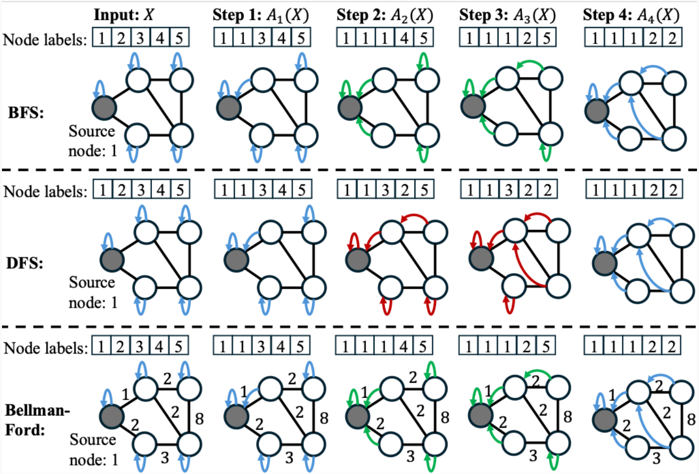
# Multitask algorithmic reasoning

Even more ambitiously, can a neural network learn to perform not just one, but multiple algorithmic reasoning tasks simultaneously?



# Labeling conflicts in multitask algorithmic reasoning

For example, how about breadth-first search (BFS), depth-first search (DFS), and Bellman-Ford? Note that these algorithms have different encodings in the intermediate steps (indicated by colors)



## Problem formulation of algorithmic reasoning

Next, introduce the formal problem setup corresponding to algorithmic reasoning.

**List of notations:**

- The input graph is given by $X$
- The $j$-th intermediate step label of an algorithm $A$ on input $X$ is given by $A_j(X)$
- We would like to learn a neural network with parameters $W$ such as a graph neural network (or a graph transformer) given by $f_W$
- The prediction of the network at step $j$ on input $X$ is given by $f_W(X; j)$
- Training objective minimizes the averaged loss over all steps in the execution:

$$\sum_X \frac{1}{S(X)} \sum_{j=1}^{S(X)} \ell(f_W(X; j), A_j(X))$$

$S(X)$ is the number of steps in the execution of algorithm $A$ on input $X$.

## Evaluation in twelve algorithmic reasoning tasks

In this experiment, we consider twelve classical graph algorithms, using the input-output encoding scheme from the Example above.

We compare training one neural network on each algorithm, vs. branching neural networks, an architecture that fits a single neural network for all twelve algorithms [LZD+26]:

| Architecture | A single GNN for each | A branching neural net for all |
|---|---|---|
| BFS | $100.0 \pm 0.0$ | $99.6 \pm 0.3$ |
| DFS | $38.7 \pm 1.5$ | $38.7 \pm 1.5$ |
| Topological sort | $66.0 \pm 2.6$ | $74.2 \pm 1.7$ |
| Articulation points | $99.8 \pm 0.2$ | $93.0 \pm 0.2$ |
| Bridges | $82.6 \pm 2.0$ | $96.2 \pm 1.4$ |
| SCC Kosaraju | $93.1 \pm 3.1$ | $92.0 \pm 1.8$ |
| MST Kruskal | $69.9 \pm 1.5$ | $66.4 \pm 1.3$ |
| MST Prim | $53.8 \pm 1.3$ | $52.2 \pm 2.4$ |
| Dijkstra | $70.9 \pm 2.3$ | $68.8 \pm 1.7$ |
| Bellman Ford | $63.8 \pm 4.1$ | $61.8 \pm 2.4$ |
| DAG | $89.2 \pm 1.4$ | $89.1 \pm 1.5$ |
| Floyd Warshall | $69.4 \pm 0.5$ | $63.1 \pm 1.3$ |

## Summary of our main empirical findings

1. Depth-first search (DFS) is notably hard to "learn" as compared to breadth-first search (BFS): for inputs generated on Erdős-Rényi random graph inputs with 16 nodes, existing neural net architectures achieve less than 40% accuracy — this is also consistent with earlier findings [VBB+22].
2. Learning multiple algorithmic reasoning processes is as easy as learning a single process: for all twelve algorithms tested in this evaluation, training a single model with separate prediction heads for each algorithm plus a shared base turns out to perform as well as training separate models each.
3. For text description versions of the same problem, fine-tuning existing LLMs follows similar patterns, but the evaluation also becomes more challenging.

All of these findings point to natural open questions around the expressivity of neural networks in terms of algorithmic reasoning!

## Summary of our main empirical findings

1. Depth-first search (DFS) is notably hard to "learn" as compared to breadth-first search (BFS): for inputs generated on Erdős-Rényi random graph inputs with 16 nodes, existing neural net architectures achieve less than 40% accuracy — this is also consistent with earlier findings [VBB+22].
2. Learning multiple algorithmic reasoning processes is as easy as learning a single process: for all twelve algorithms tested in this evaluation, training a single model with separate prediction heads for each algorithm plus a shared base turns out to perform as well as training separate models each.
3. For text description versions of the same problem, fine-tuning existing LLMs follows similar patterns, but the evaluation also becomes more challenging.

All of these findings point to natural open questions around the expressivity of neural networks in terms of algorithmic reasoning!

## Summary of our main empirical findings

1. Depth-first search (DFS) is notably hard to "learn" as compared to breadth-first search (BFS): for inputs generated on Erdős-Rényi random graph inputs with 16 nodes, existing neural net architectures achieve less than 40% accuracy — this is also consistent with earlier findings [VBB+22].
2. Learning multiple algorithmic reasoning processes is as easy as learning a single process: for all twelve algorithms tested in this evaluation, training a single model with separate prediction heads for each algorithm plus a shared base turns out to perform as well as training separate models each.
3. For text description versions of the same problem, fine-tuning existing LLMs follows similar patterns, but the evaluation also becomes more challenging.

All of these findings point to natural open questions around the expressivity of neural networks in terms of algorithmic reasoning!

## Summary of our main empirical findings

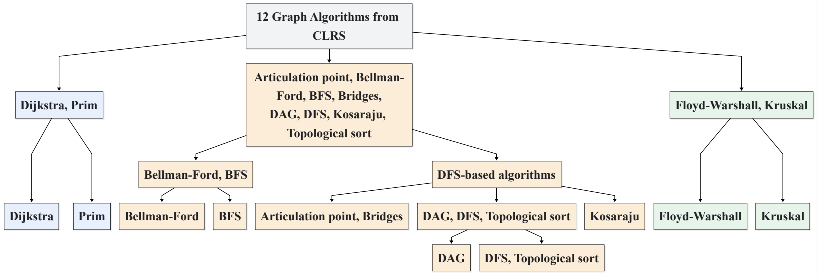
1. Depth-first search (DFS) is notably hard to "learn" as compared to breadth-first search (BFS): for inputs generated on Erdős-Rényi random graph inputs with 16 nodes, existing neural net architectures achieve less than 40% accuracy — this is also consistent with earlier findings [VBB+22].
2. Learning multiple algorithmic reasoning processes is as easy as learning a single process: for all twelve algorithms tested in this evaluation, training a single model with separate prediction heads for each algorithm plus a shared base turns out to perform as well as training separate models each.
3. For text description versions of the same problem, fine-tuning existing LLMs follows similar patterns, but the evaluation also becomes more challenging.

All of these findings point to natural open questions around the expressivity of neural networks in terms of algorithmic reasoning!

## A new architecture for multitask algorithmic reasoning

In the paper [LZD+26], we also propose a new algorithm to estimate branching neural networks. We formulate the search problem as a recursive partitioning of $n$ tasks into a $k$-ary tree.

- Searching over these partitions naively costs $O(k^{nL})$. Instead, we solve a convex relaxation at each layer to approximate the optimal partition, which brings the cost down to $O(nL)$.
- An illustration of the tree partitioning corresponding to 12 algorithms



# Outline

Evaluation of multitask algorithmic reasoning

A theoretical analysis of multi-step reasoning from the perspective of stability

## Stability of multi-step reasoning in long sequences

During training, the model always sees the *correct* preceding steps. At test time, it has to build on *its own* earlier outputs — so a predicted error made early can accumulate over the rest of the reasoning process.

Key observation: How fast the error grows is governed by how strongly each step *amplifies* the errors it inherits from earlier steps.

We formalize this intuition through measuring the Jacobian norms of each step:

- For transformers trained on simple in-context prediction tasks such as quadratic functions, we show that gradient descent converges to the stable regime where the error dies out.
- Long reasoning is not inherently unstable — to improve stability, one could instead reduce each step's sensitivity.

## Conclusion

- We conducted an evaluation of algorithmic reasoning — the input is generated based on the algorithmic execution processes of an algorithm (such as depth/breadth-first search), and we ask whether existing neural network models (e.g., graph neural nets, transformers) are able to memorize/predict the execution processes [LZD+26]:
  - Expressivity between BFS vs. DFS: Our explanation is that for the network to predict DFS, it must first identify the node IDs; if these IDs are randomly assigned at the input, then it's hard to learn. Whereas for BFS, at each step, given one node, the search process pushes through the entire neighboring set of that node.
- We propose a measure of accumulated errors in multi-step reasoning based on the product of step-by-step Jacobian norms.

Paper

Code

Li, D., Zhang, Z., Duan, M., Dobriban, E., and Zhang, H. R. (2026). **"Efficiently Learning Branching Networks for Multitask Algorithmic Reasoning".** In: *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)* (10, 15, 18).

Markeeva, L., McLeish, S., Ibarz, B., Bounsi, W., Kozlova, O., Vitvitskyi, A., Blundell, C., Goldstein, T., Schwarzschild, A., and Veličković, P. (2024). **"The CLRS-Text Algorithmic Reasoning Language Benchmark".** In: *arXiv preprint arXiv:2406.04229* (2–4).

Veličković, P., Badia, A. P., Budden, D., Pascanu, R., Banino, A., Dashevskiy, M., Hadsell, R., and Blundell, C. (2022). **"The CLRS Algorithmic Reasoning Benchmark".** In: *International Conference on Machine Learning (ICML)* (2–4, 11–14).