TransPath: Learning Heuristic for Grid-Based Pathfinding via Transformers

Federal Research Center Computer Science and Control of Russian Academy of Sciences

Daniil Kirilenko[1], Anton Andreychuk[2], Aleksandr Panov[1,2], Konstantin Yakovlev[1,2]

anedanman@gmail.com, andreychuk@airi.net, panov@airi.net, yakovlev@isa.ru

[1]FRC CSC RAS, Moscow, Russia
[2] AIRI, Moscow, Russia

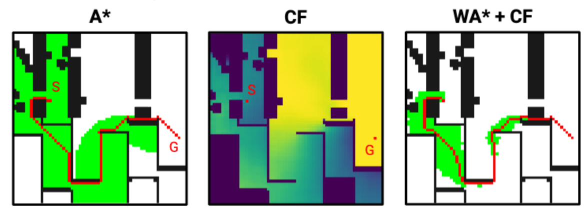
1. Problem Statement

Heuristic search performance strongly depends on a *given* heuristic function. Typically, heuristic functions are **not instance-dependent**, which results in **excessive expansions**. We utilize moder **transformer**-based neural networks to learn **informed heuristic** function that are instance-dependent and take obstacles into account.

2. Correction Factor

The first heuristic we propose to learn is the **correction factor (cf)**, which is defined as the ratio of the value of the instance-independent heuristic to the value of the perfect heuristic

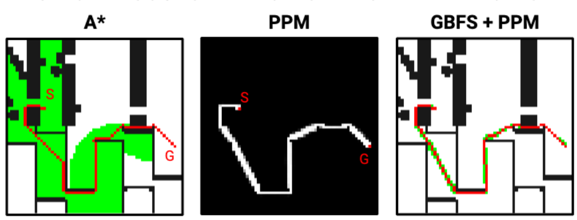
$$cf(n) = h(n)/h^*(n)$$



3. Path Probability Map

The second type of heuristic is the **path probability map (ppm)**. This heuristic is designed to demonstrate, how likely the node is lying on the shortest path from the start to the goal. We use Theta* to compute ppm values that tell us how close the cost of the path though the cell *n*.

$$cost(\pi(s,g))/(cost(\pi(s,n)+cost(\pi(n,g))$$

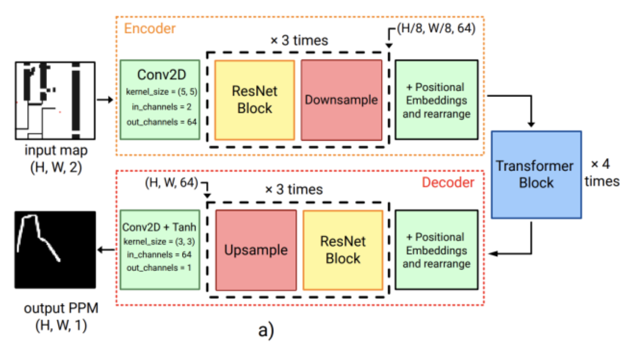


4. Neural Network

Our model includes 3 main blocks: convolutional encoder, spatial transformer and convolutional decoder. Convolutional blocks are aimed at extracting local features, while transformer blocks establish the global relations between these features.

The network is quite small and includes 1 million parameters.
We train each model using direct supervision. Mean squared error is used as a training signal.

$$\text{Loss} = \text{MSE}(h_{gt}, h_{pred})$$



5. Dataset

- 64k TMP (Yonetani et al.) maps
- 10 problem instances per map
- Map size is 64x64
- Train-val-test split is 8-1-1

6. Experiments

Test dataset

	Optimal Found Ratio (%) ↑	Cost Ratio (%) ↓	Expansions Ratio (%) ↓
A*	100	100	100
WA*	40.66	103.52 ±4.85	44.43 ±25.92
Neural A*	29.82	104.90 ±6.56	52.30 ±30.47
A*+HL	79.11	100.27 ±0.62	80.50 ±74.40
WA*+CF	**85.40**	100.25 ±1.13	36.98 ±21.18
FS+PPM	82.97	**100.24** ±**0.74**	26.36 ±21.08
GBFS+PPM	83.02	100.25 ±0.90	**23.60** ±**18.34**

Out-of-Distribution dataset

	Optimal Found Ratio (%) ↑	Cost Ratio (%) ↓	Expansions Ratio (%) ↓
A*	100	100	100
WA*	8.13	104.31 ±4.76	57.52 ±30.72
Neural A*	3.24	107.10 ±6.77	63.08 ±34.63
A*+HL	**29.02**	**101.90** ±2.72	148.94 ±136.95
WA*+CF	10.61	106.10 ±5.59	63.64 ±36.31
FS+PPM	18.66	105.62 ±5.61	55.06 ±39.57
GBFS+PPM	18.59	106.12 ±6.54	**54.33** ±**47.24**

Code repository

Yonetani, R.; Taniai, T.; Barekatain, M.; Nishimura, M.; and Kanezaki, A. 2021. Path planning using neural A* search. In Proceedings of the 38th International Conference on Machine Learning (ICML 2021), 12029–12039.