Beyond the Heatmap

What really drives performance in Heatmap + MCTS TSP solvers?

Xuanhao Pan[1,†], Chenguang Wang[1,†], Chaolong Ying[1], Ye Xue[1,2,3], Tianshu Yu[1]

[1]School of Data Science, CUHK-Shenzhen [2]Shenzhen Research Institute of Big Data [3]SYSU Shenzhen Campus

Most prior attention **This paper re-evaluates here**

Heatmap → **MCTS** → **Final Tour**

Common belief

Better heatmap ⇒ better final result

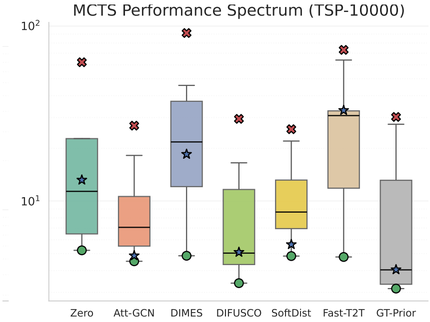
This paper asks

Is that still true when MCTS is tuned fairly?

The field may be over-crediting the heatmap and underestimating the search component.

MCTS tuning changes both results and conclusions

Search settings can flip the verdict on the same heatmap



Magnitude

Best vs. worst tuning can completely change performance.

Implication

Default-setting comparisons may misjudge heatmap quality.

DIMES on TSP-10000

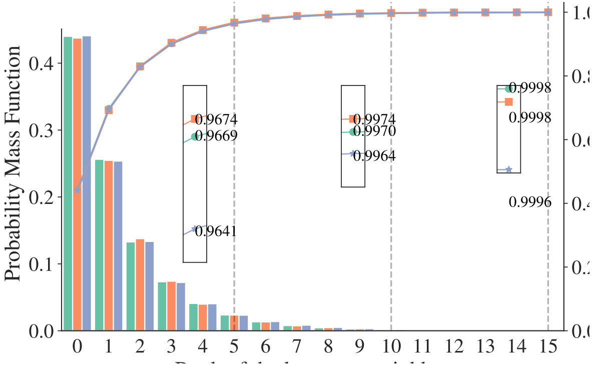
4.86% → 91.31% gap

Each heatmap tuned separately on TSP-10000; green = best, blue = default, red = worst.

SHAP takeaway: candidate set size is one of the most influential MCTS parameters.

With tuned search, simplicity is already strong

A simple prior, or even no prior, can already be competitive



Optimal tours are highly local: >94% of edges lie within top-5 nearest neighbors; >99% lie within top-10.

Tuned gap (%)

Method	500	1000	10000
Zero No heatmap cost	0.66	1.16	3.80
GT-Prior No heatmap cost	0.50	0.85	**2.14**
DIFUSCO	**0.33**	**0.53**	2.37

Core read

Even **Zero + tuned MCTS** still reaches **3.80%** on TSP-10000.

Cost cue: DIFUSCO heatmap generation still adds **28.51 min** on TSP-10000.

Robust across scale, because the prior matches geometry

GT-Prior stays strong across both larger sizes and shifted distributions

Scale transfer to TSP-10000

Method	Same-scale	Transfer
GT-Prior	2.14%	**2.13%**
DIFUSCO	2.36%	5.27%

Using a TSP-500-derived prior, GT-Prior changes by only **-0.01** points, while DIFUSCO degrades by **+2.91** points.

Distribution shift on TSP-10000

Method	Cluster	Expl.	Impl.
GT-Prior	**0.35**	**0.93**	**0.58**
DIFUSCO	1.96	2.50	2.42

A uniform-derived prior remains strong even when the point layout changes substantially.

Why this is robust: optimal Euclidean tours remain overwhelmingly local, so a k-nearest-neighbor prior captures a stable structural cue rather than a scale-specific learned pattern.

Takeaway: tune search before claiming heatmap gains

Not anti-learning: learning and search must be evaluated together fairly

1. MCTS tuning strongly affects solution quality

2. A simple k-NN prior can match or beat learned heatmaps

3. This paper provides a standardized MCTS tuning pipeline for fairer comparisons

Final takeaway

Before investing in a more complex heatmap, first make sure your search is properly tuned.

Future direction: co-optimize learning and search.