On the Effectiveness of Pretraining for Graph Combinatorial Optimization

Authors: David Aguado (david.aguado@alumnos.upm.es),
Daniel Fuertes, Carlos R. del-Blanco, Fernando Jaureguizar
Grupo de Tratamiento de Imágenes, Universidad Politécnica de Madrid

Acknowledgements: *This work was supported in part by the Comunidad de Madrid under project TEC-2024/COM-322 (IDEALCVCM), in part by MCIU/AEI/10.13039/501100011033 of the Spanish Government under project PID2023148922OA-I00 (EEVOCATIONS), and in part by "Ayudas a la Investigacion para el Personal Docente e Investigador de la ETSIT-UPM ´ (2026)" under project "SATURNO". The authors would also like to thank Airbus Defence and Space for their support.*

Neural Combinatorial Optimization and TSP

Exact Solvers
- Very slow
- Optimal solutions

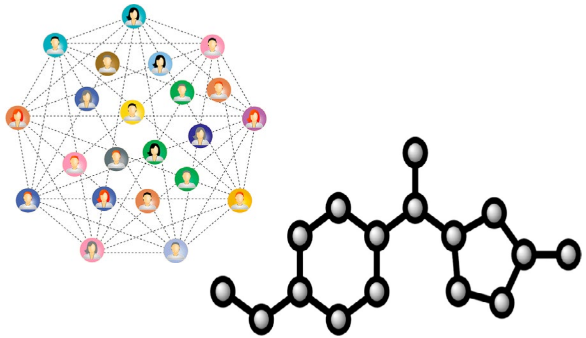
Heuristic Algorithms
- Fast
- Not optimal performance

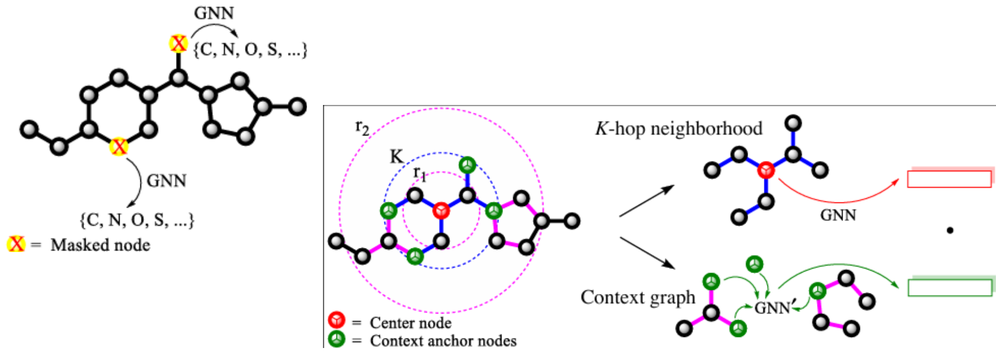
NCO
- Fast
- Better performance ¿?

Generate good embedddings

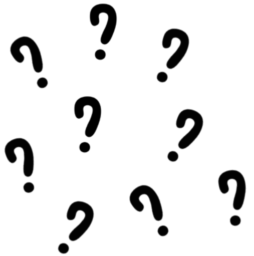
"Given a list of cities and the distances between each pair of them, what is the shortest possible route that visits each city exactly once and returns to the origin city?"

The problem of "attribute-poor" graphs

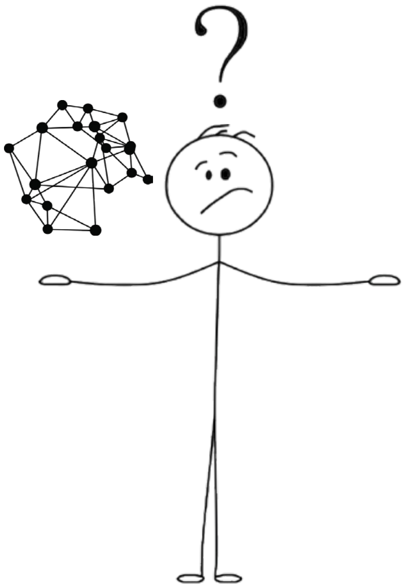




“STRATEGIES FOR PRE-TRAINING GRAPH NEURAL NETWORKS”

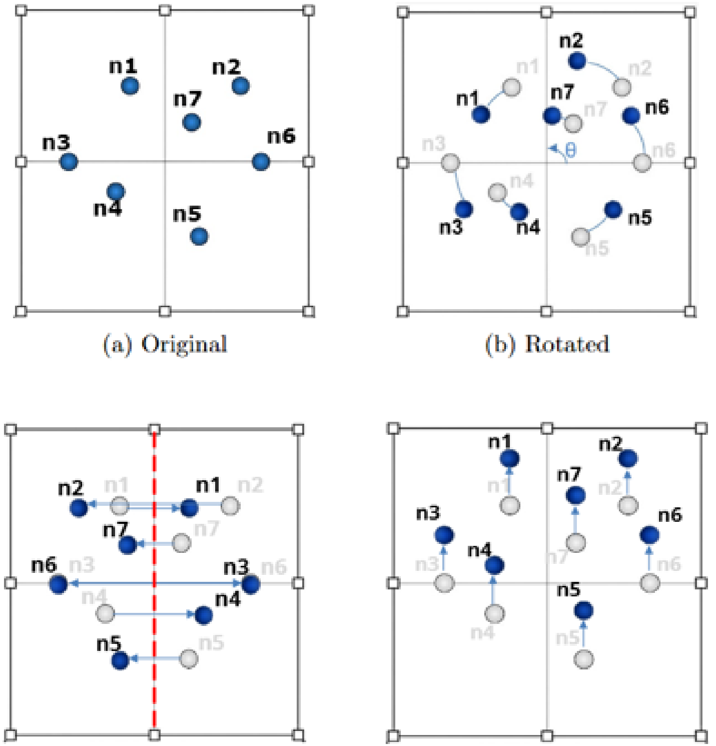


Isometric augmentations



Contrastive learning

Applied transformations



(a) Original (b) Rotated

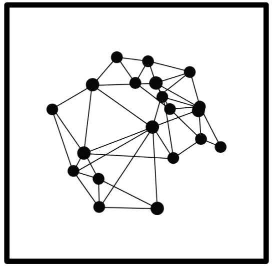
(c) Reflected (d) Displaced

Applied transformations

- Rotation:

$$f_\theta(\mathbf{x}_i) = \mathbf{R}_\theta(\mathbf{x}_i - \mathbf{c}) + \mathbf{c}, \quad \mathbf{R}_\theta = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

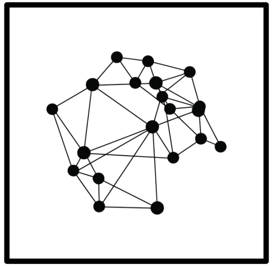
where R_θ is the 2D rotation Matrix



- Translation:

$$f_\Delta(x_i) = x_i + \Delta$$

where Δ is the displacement vector $\Delta = \begin{bmatrix} d\cos\alpha \\ d\sin\alpha \end{bmatrix}$

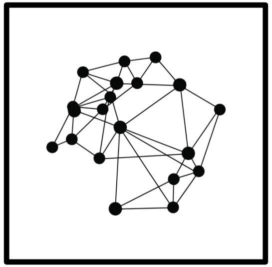


Applied transformations

- Reflection:

$$f_\phi(\mathbf{x}_i) = \mathbf{S}_\phi(\mathbf{x}_i - \mathbf{c}) + \mathbf{c}, \quad \mathbf{S}_\phi = 2\mathbf{v}\mathbf{v}^\top - \mathbf{I}$$

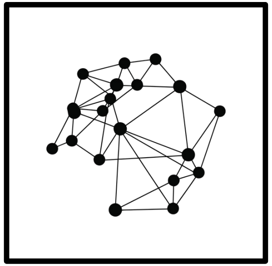
where S_ϕ the reflection matrix of Householder and v = [cos ϕ, sin ϕ]T the unitary vector that defines the reflection axis



- Hibrid:

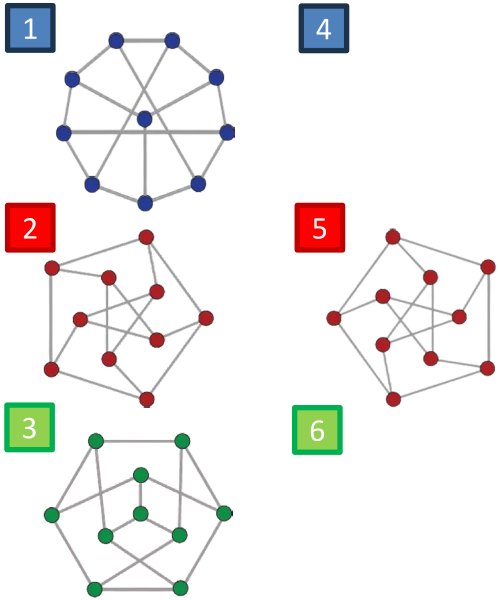
$$g(x_i) = f_\Delta(f_\theta(f_\phi(x_i)))$$

where f_θ is the rotation transformation and f_ϕ is the reflection transformation



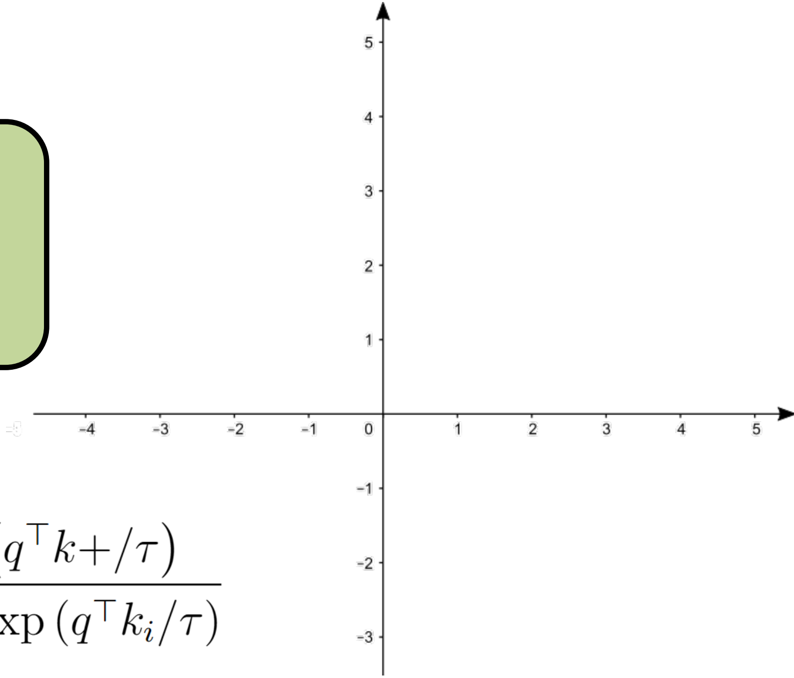
Training Flow – Pre-training

Generating
TSP20-50



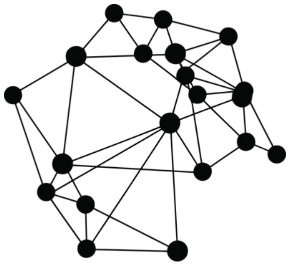
Encoder

Latent space



$$\mathcal{L}_{\text{InfoNCE}} = -\log \frac{\exp\left(q^{\top}k+/\tau\right)}{\sum_{i=0}^{K}\exp\left(q^{\top}k_i/\tau\right)}$$

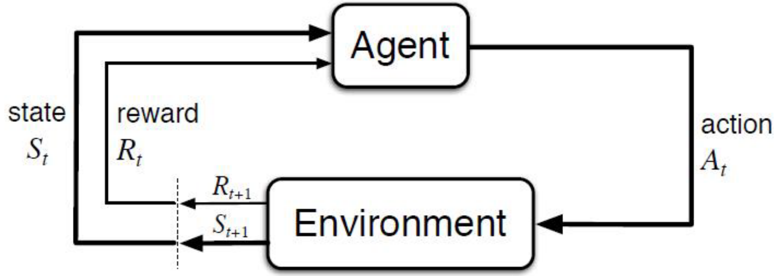
Training Flow - Training



TSP50

Encoder

Decoder

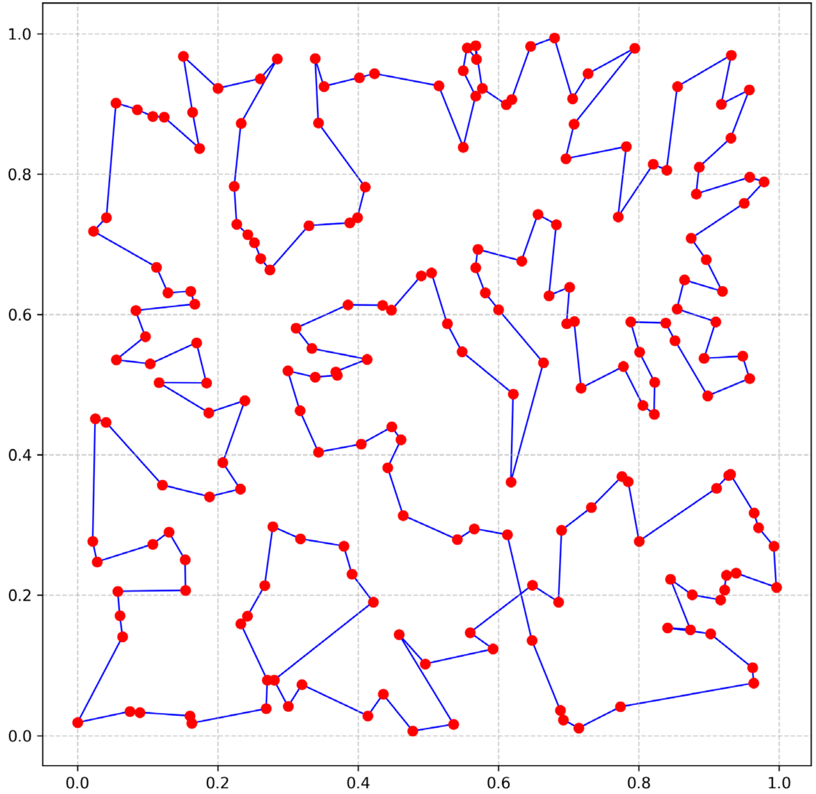


System Architecture

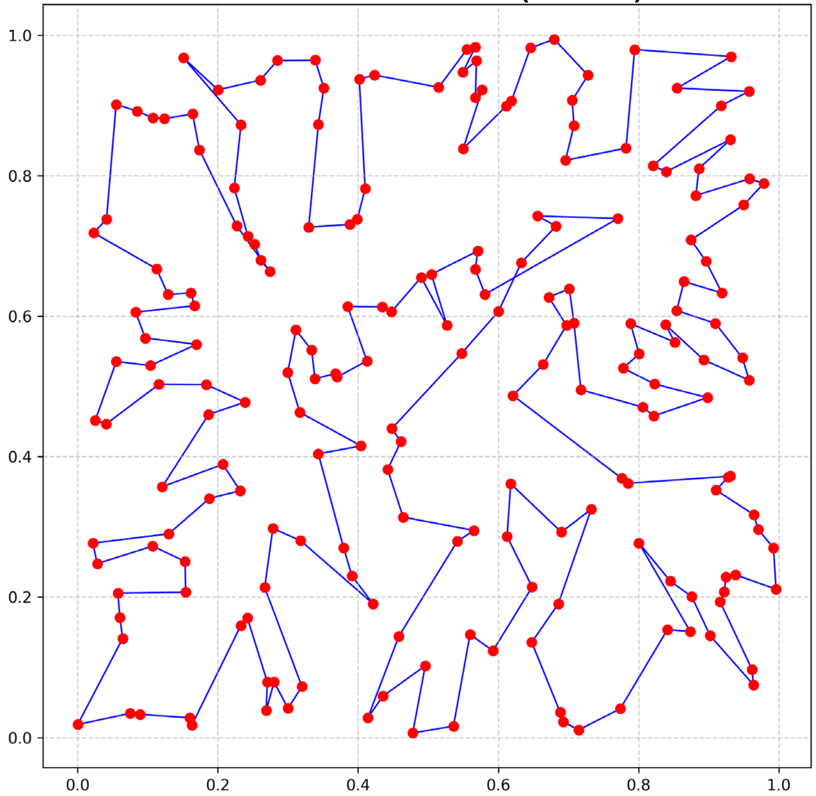
- **Encoder-Decoder Architecture:** Unified pipeline for sequential decision-making on graphs.
- **Encoder:** Gated Graph ConvNet (GatedGCN) based Graph Neural Network to transform coordinates into high-dimensional embeddings.
- **Decoder:** Autoregressive network with a Multi-Head Attention mechanism to iteratively construct the optimal route.
- **Universal Framework (Model-Agnostic):** The geometric pre-training strategy is architecture-independent and applicable to any graph network.

Solved TSPs

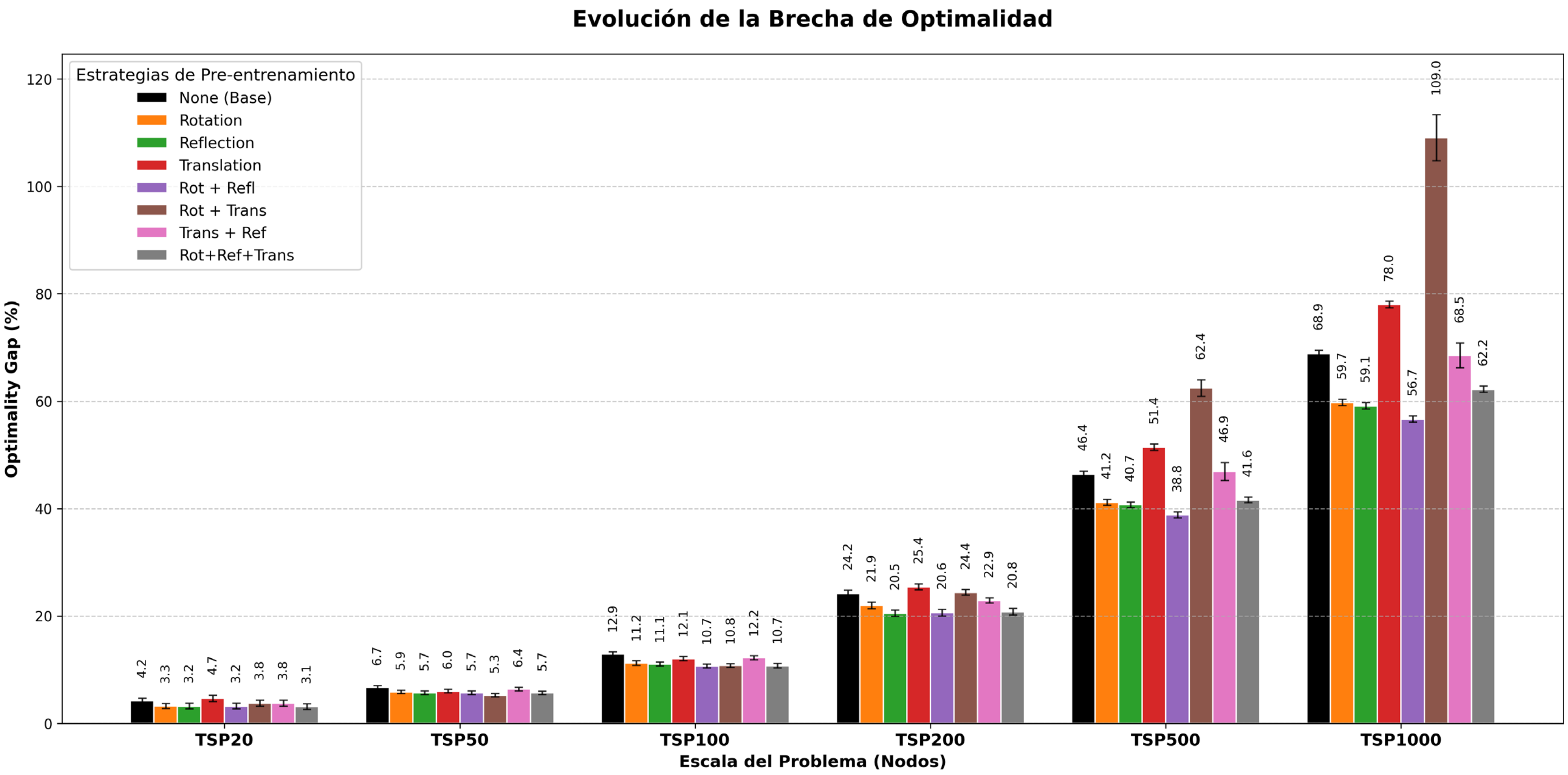
Pre-trained model | Cost: 11.914
Mean cost: 12.212 ± 0.059 (95% CI)



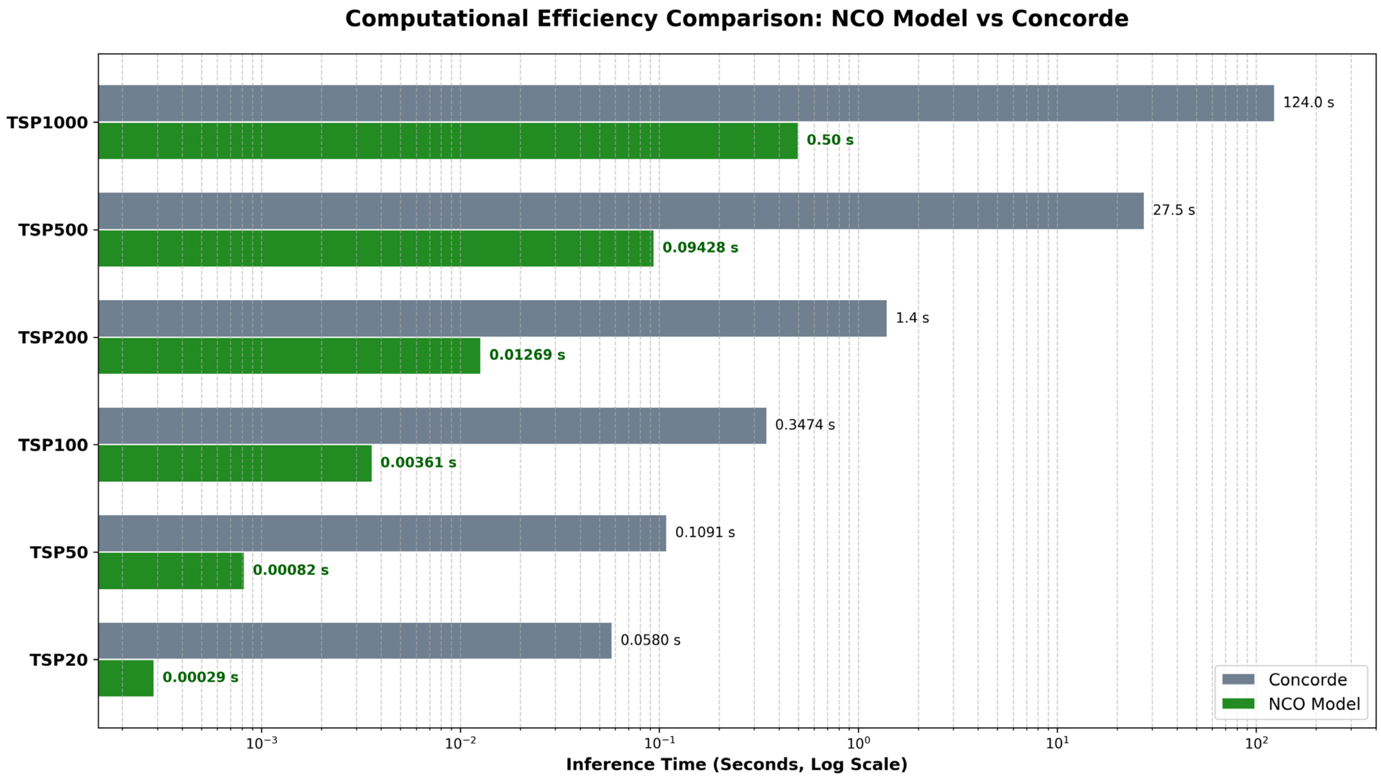
Non-Pre-trained model | Cost: 13.014
Mean cost: 13.305 ± 0.074 (95% CI)



Ablation Studies



Time comparison



Conclusions & Future research

Conclusions

- Self-supervised **geometric pre-training** is an excellent way to **overcome lack of features** in routing graphs.
- Our ablation study empirically demonstrates that **not all graph isometries** benefit learning **equally**.
- A **neural network** can be hundreds of times **faster** than optimal solvers in exchange for a **small optimality gap**.

Future research

- Explore **new geometric transformations** based on homotheties (spatial dilation and contraction).
- Investigate the **adaptation** of **masking strategies** within the TSP framework.
- Transfer this architecture to **more complex logistic problems**, such as VRPs with capacity or time windows.

Any Questions?

GSP 2026
Madrid, Spain

On the Effectiveness of Pretraining for Graph Combinatorial Optimization

Authors: David Aguado (david.aguado@alumnos.upm.es),
Daniel Fuertes, Carlos R. del-Blanco, Fernando Jaureguizar
Grupo de Tratamiento de Imágenes, Universidad Politécnica de Madrid

Acknowledgements: *This work was supported in part by the Comunidad de Madrid under project TEC-2024/COM-322 (IDEALCVCM), in part by MCIU/AEI/10.13039/501100011033 of the Spanish Government under project PID2023148922OA-I00 (EEVOCATIONS), and in part by "Ayudas a la Investigacion para el Personal Docente e Investigador de la ETSIT-UPM ´ (2026)" under project "SATURNO". The authors would also like to thank Airbus Defence and Space for their support.*

Temporal difference (NCO Model vs Concorde)

	TSP20	TSP50	TSP100	TSP200	TSP500	TSP1000
Híbrido (performance)	3.909 ± 0.020	5.963 ± 0.019	8.425 ± 0.033	12.334 ± 0.075	21.686 ± 0.087	34.192 ± 0.137
(Time)	0.000289s	0.000822s	0.003606s	0.012690s	0.094282s	0.500099s
Concorde (performance)	3.829 ± 0.020	5.704 ± 0.015	7.790 ± 0.032	10.726 ± 0.064	16.489 ± 0.090	23.308 ± 0.132
(Time)	0.058013s	0.109062s	0.347416s	1.401078s	27.494105s	124.008415s
Cociente de tiempo	x200.8	X132.7	X96.35	x110.4	x291.6	X248.0

Performance of different pretrainings (pretrain 20-50, train50)

Pretraining	TSP20	TSP50	TSP100	TSP200	TSP500	TSP1000
None	3.989 ± 0.020	6.087 ± 0.019	8.796 ± 0.031	13.321 ± 0.064	24.139 ± 0.092	38.940 ± 0.137
Rotation	3.954 ± 0.020	6.039 ± 0.018	8.666 ± 0.033	13.080 ± 0.066	23.278 ± 0.094	36.835 ± 0.134
Reflection	3.953 ± 0.020	6.030 ± 0.019	8.651 ± 0.028	**12.927 ± 0.064**	23.201 ± 0.089	36.694 ± 0.132
Translation	4.008 ± 0.022	6.047 ± 0.019	8.729 ± 0.031	13.453 ± 0.062	24.970 ± 0.094	41.042 ± 0.148
Rot + Refl	3.953 ± 0.020	6.030 ± 0.018	**8.623 ± 0.028**	**12.936 ± 0.066**	**22.887 ± 0.091**	**36.123 ± 0.139**
Rot + Trans	3.973 ± 0.021	**6.005 ± 0.018**	**8.630 ± 0.028**	13.344 ± 0.058	26.782 ± 0.255	48.199 ± 0.986
Trans + Ref	3.973 ± 0.021	6.070 ± 0.019	8.743 ± 0.028	13.183 ± 0.055	24.219 ± 0.276	38.855 ± 0.529
Rot+Ref+Trans	**3.949 ± 0.020**	**6.028 ± 0.019**	**8.626 ± 0.033**	**12.959 ± 0.065**	23.350 ± 0.089	37.405 ± 0.137
Concorde	3.829 ± 0.020	5.704 ± 0.015	7.790 ± 0.032	10.726 ± 0.064	16.489 ± 0.090	23.058 ± 0.132