

IDEA

- L2P (Predict -> dự đoán heatmap)

- **Deep Policy Dynamic Programming for Vehicle Routing Problems** (Sử dụng mạng GNN đơn giản để dự đoán heatmap của các cạnh tiềm năng) -> **Có thể dùng làm tăng xác suất chọn cạnh tốt hơn.**

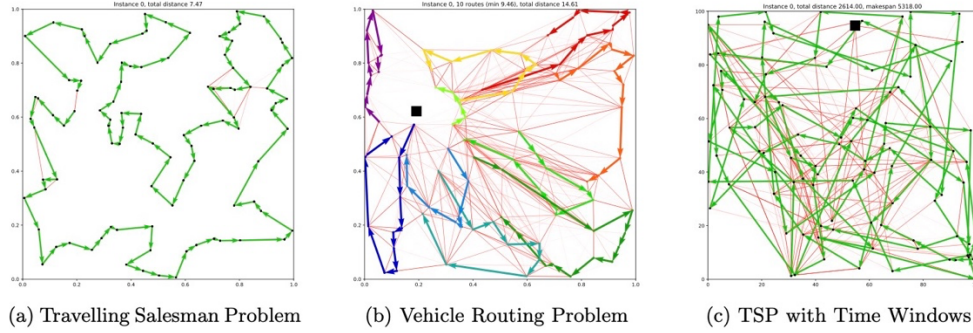
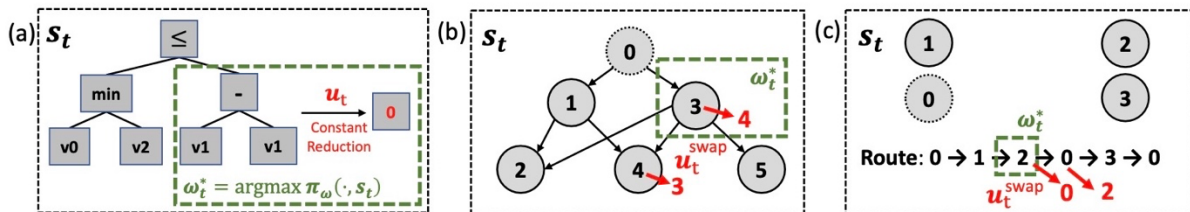


Figure 1: Heatmap predictions (red) and solutions (colored) by DPDP (VRP depot edges omitted).

- L2S (Search -> Dùng model để cải tiến 1 lời giải thành lời giải tốt hơn -> Giống Local Search).

- **Learning to Perform Local Rewriting for Combinatorial Optimization**

Optimization as a rewriting problem. Let \mathcal{U} be the rewriting ruleset. Suppose s_t is the current solution (or state) at iteration t . We first compute a state-dependent *region set* $\Omega(s_t)$, then pick a region $\omega_t \in \Omega(s_t)$ using the *region-picking policy* $\pi_\omega(\omega_t|s_t)$. We then pick a rewriting rule u_t applicable to that region ω_t using the *rule-picking policy* $\pi_u(u_t|s_t[\omega_t])$, where $s_t[\omega_t]$ is a subset of state s_t . We then apply this rewriting rule $u_t \in \mathcal{U}$ to $s_t[\omega_t]$, and obtain the next state $s_{t+1} = f(s_t, \omega_t, u_t)$. Given an initial solution (or state) s_0 , our goal is to find a sequence of rewriting steps $(s_0, (\omega_0, u_0)), (s_1, (\omega_1, u_1)), \dots, (s_{T-1}, (\omega_{T-1}, u_{T-1})), s_T$ so that the final cost $c(s_T)$ is minimized.



- **Learning to Search Feasible and Infeasible Regions of Routing Problems with Flexible Neural k-Opt** (Dùng model để cải tiến k-opt local_search sao cho thỏa mãn chọn ra được k vùng thích hợp)

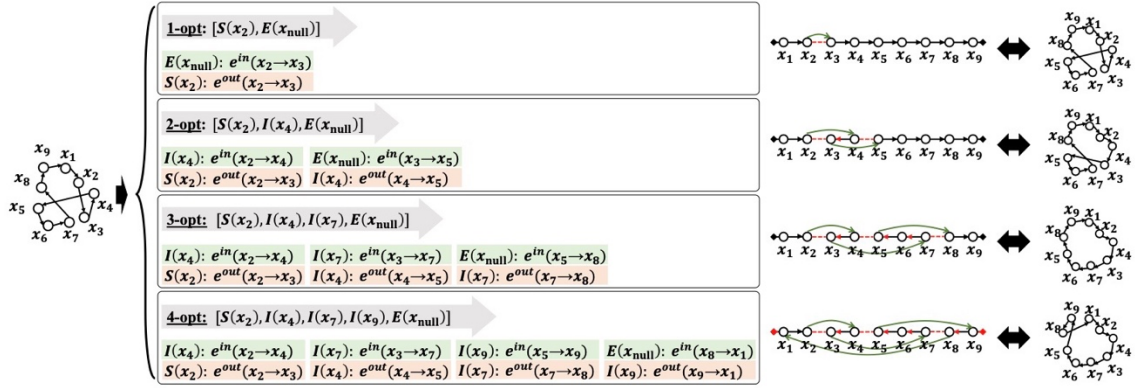


Figure 7: Examples of using the basis moves to factorize 1-opt (void action), 2-opt, 3-opt, and 4-opt.

We introduce a new factorization method that constructs a k -opt exchange using a combination of three *basis moves*, namely the *starting move*, the *intermediate move*, and the *ending move*. Concretely, the sequential k -opt can be simplified as performing one S-move, several (possibly none) I-moves, and finally one E-move, where the choice of the k corresponds to determining the number of I-moves.

- LARGE-SCALE PROBLEM

- **GLOP: Learning Global Partition and Local Construction for Solving Large-scale Routing Problems in Real-time.**

Chia bài toán CVRP thành 2 phase, phase 1 sẽ sử dụng heatmap để phân chia đồ thị thành các subTSP (predict, no autoregressive), phase 2 là thực hiện auto_regressive trên từng subTSP đó.

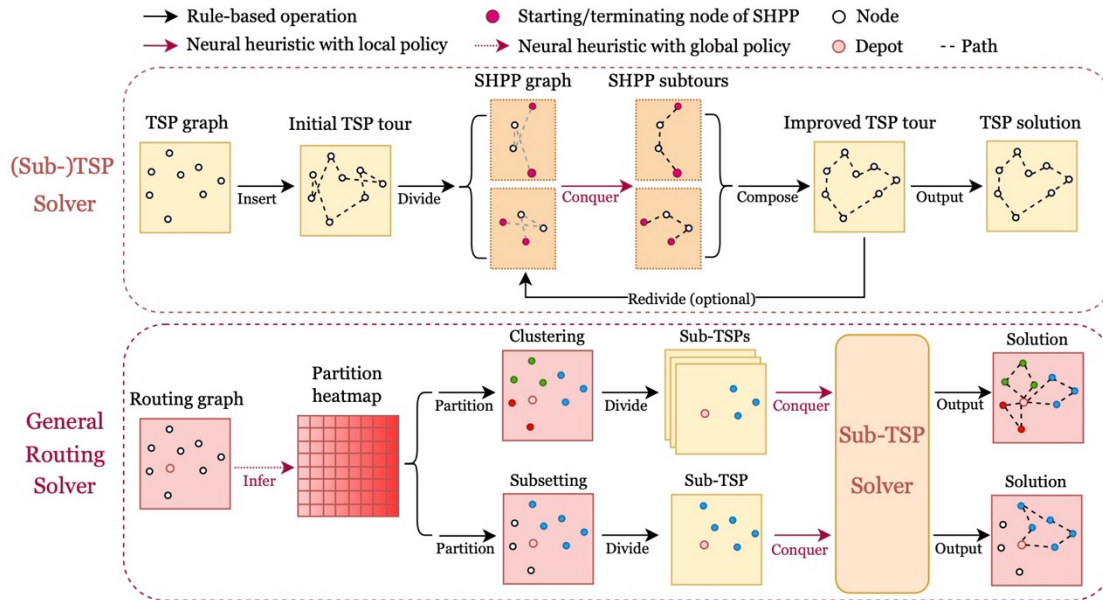


Figure 1: The pipeline of GLOP.

- Other

- **POMO: Policy Optimization with Multiple Optima for Reinforcement Learning**

Thay vì phụ thuộc hoàn toàn vào node đầu tiên để dự đoán thì sẽ tăng tính đa dạng bằng cách cho nhiều node vào làm node đầu tiên.

Ví dụ: Với bài toán TSP, nếu (1,2,3,4,5,6) là 1 lời giải thì (3,4,5,6,1,2) cũng là 1 lời giải. 2 node xuất phát là 1 và 3 đều cho cùng 1 lời giải, tuy nhiên phụ thuộc vào model thì có thể tăng tính đa dạng.

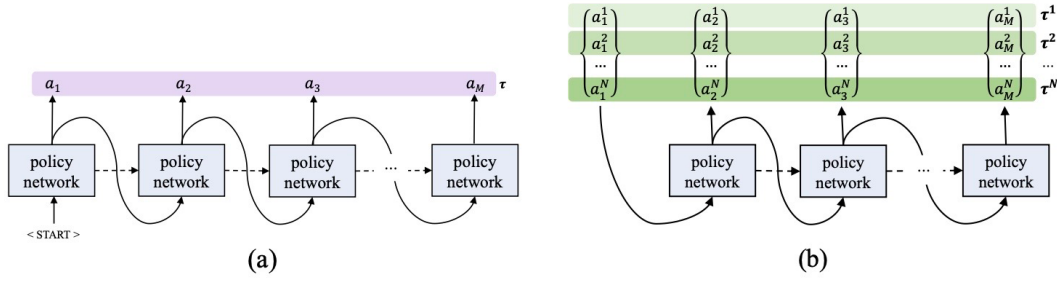


Figure 2: (a) Common method for generating a single solution trajectory (τ) based on START token scheme. (b) POMO method for multiple trajectory $\{\tau^1, \tau^2, \dots, \tau^N\}$ generation in parallel with a different starting node for each trajectory.

- **DeepACO: Neural-enhanced Ant Systems for Combinatorial Optimization**

Sử dụng ACO làm base. Model sẽ tập trung vào dự đoán heuristic measure đồ thị.

$$P(s_t | s_{<t}, \rho) = \begin{cases} \frac{\tau_{ij}^\alpha \cdot \eta_{ij}^\beta}{\sum_{c_{il} \in N(s_{<t})} \tau_{il}^\alpha \cdot \eta_{il}^\beta} & \text{if } c_{ij} \in N(s_{<t}), \\ 0 & \text{otherwise.} \end{cases}$$

