Title of the project: Model-Based Gradient Search for Permutation Problems

Group - **G72**

Members

1. Yash Deepak Malve – S20230010141 (Section 1)
2. Purval Madhukar Bhude – S20230010193 (Section 2)
3. Piyush Kumar – S20230010186 (Section 3)

Project GUI Link - [aiproject.vercel]

Github repository link - [ai project]

Abstract:

This project focuses on solving hard optimization problems where the goal is to find the best order or arrangement (called permutations). These problems are tricky because we can't use normal gradient-based methods like in continuous optimization. To handle this, we use a smart approach: instead of directly optimizing solutions, we define a probability model that can generate good permutations. We then improve this model using gradients.

We use the **Plackett-Luce (PL)** model, which is good at handling permutations and allows us to calculate gradients easily. They implemented an algorithm based on this model and tested it on benchmark problems.They also explored improvements like adjusting the learning rate automatically, restarting the algorithm when it gets stuck, and using natural gradients to speed up learning.

In short, they studied, implemented, and analysed a gradient-based search method for solving permutation-based problems in a more efficient way.

Plane of Work:

Motivation:

- No Gradient in Combinatorial optimization problem (COPs)
- Model-based Gradient search for COPs
  - Identify a probability model over the discrete space of solution, whose pdf is differentiable with respect to its parameters
  - Optimize the expected objective value of model by applying gradient ascent/descent to the model's parameters
- There are proposals for binary problems, but not for permutations problem

Model - Based Gradient Search:

Algorithm 1 Gradient Search for Combinatorial Optimization

function GS($f : \Omega \rightarrow \mathbb{R}, \eta \in \mathbb{R}^{+}, \lambda \in \mathbb{N}^{+}$)
    initialize θ_0 in such a way that probabilities are uniformly distributed
    x^* maintains the best solution so far
    $t \leftarrow 0$
    while stopping criterion is not met **do**
        for $i \leftarrow 1$ **to** λ **do**
            sample $x_i \in \Omega$ by drawing it from p_{θ_t}
            evaluate $f(x_i)$ and update x^* if improvement found
            calculate $\nabla_{\theta_t} \log p_{\theta_t}(x_i)$
        end for
        calculate $\nabla_{\theta_t} J(\theta_t)$ according to Eq. (6)
        $\theta_{t+1} \leftarrow \theta_t + \eta \nabla J(\theta_t)$
        $t \leftarrow t + 1$
    end while
    return x^*
end function

Plackett-Luce as Probability Model:

- A Plackett-Luce model is parameterized by an n-length vector z
- The probability of an n-length permutation σ is

$$P(\sigma \mid \mathbf{z}) = \prod_{i=1}^{n-1} \frac{\exp(z_{\sigma(i)})}{\sum_{j=i}^{n} \exp(z_{\sigma(j)})}$$

- Permutations can be sampled in $\theta(n \log(n))$
- Gradient of the log-probability of a permutation is computed in $\theta(n)$

Other Algorithmic Components:

- **Cumulative Step-size Adaptation for the learning rate parameter**
- **Entropy-based adaptation for the sample size parameter**
- **Utility-based transformation of the objective values**
- **Soft restart mechanism**
- **Natural gradient (**to match the local curvature in the parameter space)

Experiments on the linear ordering problem:

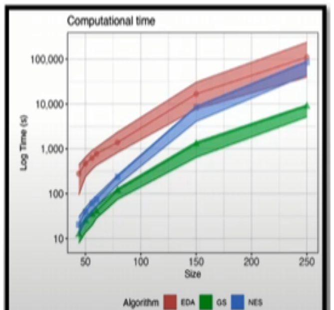

Figure 2: Efficiency analysis.

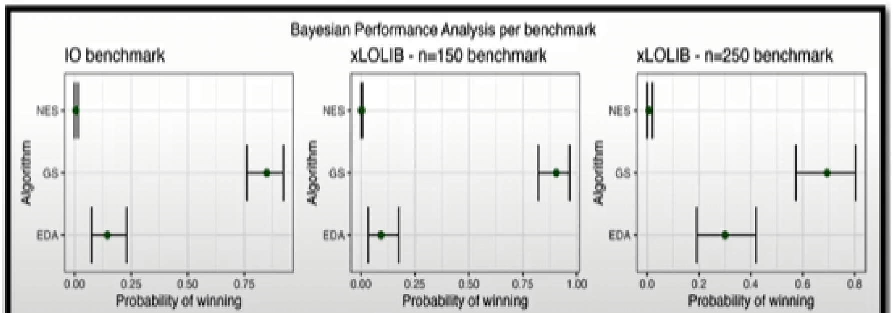

Figure 1: Effectiveness analysis.

It was observed that GS (gradient search without natural gradient) give highest probability of winning with more efficient time complexity

- GS - PL based gradient search without natural gradient
- NES - PL based Gradient search with natural gradient
- EDA - estimation of distribution algorithm using PL model

Reference paper:

- *J. Ceberio and V. Santucci, "Model-based Gradient Search for Permutation Problems," ACM Transactions on Evolutionary Learning and Optimization, vol. 3, no. 4, Article 15, December 2023. doi:10.1145/3628605.*

  https://doi.org/10.1145/3628605