Risk-Aware Submodular Optimization for Stochastic Travelling Salesperson Problem

Rishab Balasubramanian [1], Lifeng Zhou [2,3], Pratap Tokekar [3], P.B. Sujit [1]

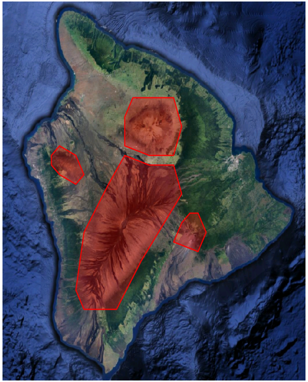
[1]IISER Bhopal

[2]University of Pennsylvania

[3]University of Maryland

2021 IEEE International Conference on Intelligent Robots and Systems

Motivation



- Determining an optimal tour to visit all the locations in a given set while minimizing/maximizing a metric is the classical Travelling Salesperson Problem (TSP)
- Several approaches have been previously proposed which consider deterministic rewards/costs.
- We argue that a different approach is necessary in many risk-sensitive applications, which are stochastic in the rewards and costs.
- We further extend this to present a multi-stage planner.

Introduction

- **Objective**: We introduce a risk-aware variant of the Traveling Salesperson Problem (TSP).
- In this formulation, the robot tour cost and reward have to be optimized simultaneously, while being subjected to uncertainty in both.
- We study the case where both rewards and costs are submodular, and propose a Risk-Aware Greedy Algorithm (RAGA) to find an approximate solution for this problem.

Measuring Risk

- The Conditional-Value-at-Risk (CVaR) is defined as

$$CVaR_{\alpha}(\mathcal{S}) = \mathbb{E}_y[f(\mathcal{S}, y) \mid f(\mathcal{S}, y) \leq VaR_{\alpha}(\mathcal{S})]. \tag{1}$$

  where $\mathcal{S}$ is the decision set, in our case the set of edges selected after each iteration of RAGA.
- By definition with respect to a specified probability level α, the Value-at-Risk (VaR) is the lowest amount τ such that, with probability α, the function f will not exceed τ.
- The CVaR is the conditional expectation of f above that amount τ.

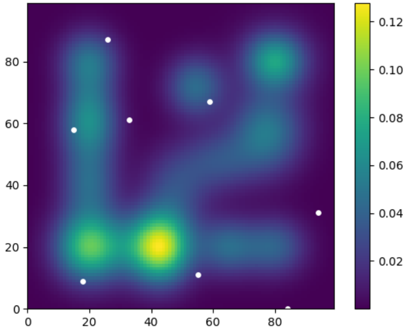
Measuring Risk

- Maximizing the value of $\text{CVaR}_\alpha(\mathcal{S})$ is equivalent to maximizing the auxiliary function $H(\mathcal{S},\tau)$

$$H(\mathcal{S},\tau) = \tau - \frac{1}{\alpha}\mathbb{E}_y[\,(\tau - f(\mathcal{S},y))^+\,], \tag{2}$$

where $[t]^+ = t,\ \forall t \geq 0$ and 0 when $t < 0$.

Problem Formulation



- Consider the map above which represents the information (reward) distribution in the exploration space.
- We represent the above as a graph $G(\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ represents the set of vertices and $\mathcal{E}$ the set of all edges between each pair of vertices.

Problem Formulation

- We define a utility function $f(\mathcal{S}, y)$ as :

$$f(\mathcal{S}, y) = (1-\beta)\, r(\mathcal{S}, y_r) + \beta\, (|\mathcal{S}|C - c(\mathcal{S}, y_c)), \tag{3}$$

  where $\beta \in [0, 1]$ is the weighting parameter representing the reward-cost trade-off.
- The utility function defined above is normalized, submodular, and monotone increasing in $\mathcal{S}$

Risk Aware Submodular Optimization

- Consider the set $\mathcal{I}$ as being formed from the union of power sets for each Hamiltonian tour $\mathcal{A}_i$ (i.e.) $\mathcal{I} := 2^{\mathcal{A}_1} \cup 2^{\mathcal{A}_2} \ldots \cup 2^{\mathcal{A}_n}$
- We define our problem of risk-aware TSP as maximizing $\text{CVaR}_\alpha(\mathcal{S})$, where $\mathcal{S} \in \mathcal{I}$.
- Which is equivalent to maximizing the auxiliary function

$$\max_{\mathcal{S}\in\mathcal{I},\ \tau\in[0,\Gamma]} \tau - \frac{1}{\alpha}\mathbb{E}_y[(\tau - f(\mathcal{S}, y))^+], \tag{4}$$

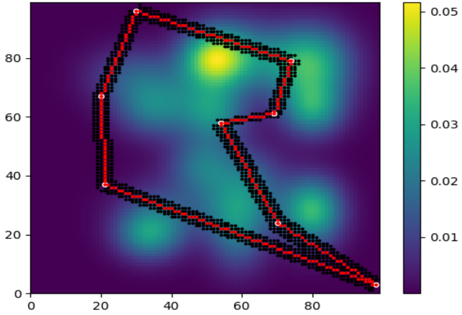
where Γ is the upper bound on the value of τ.

To approximately solve this problem we develop the Risk Aware Greedy Algorithm (RAGA), which has the following three steps:

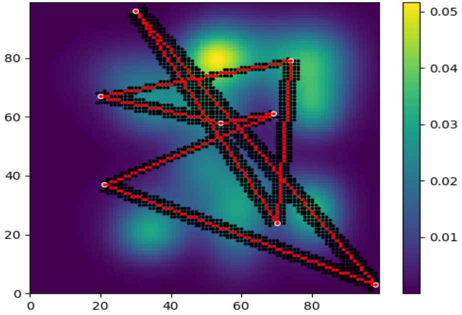
- **Initialization :** The decision set $\mathcal{S}$ (Hamiltonian cycle) is initialized as an empty set.
- **Search for valid tour for every τ :** For a particular value of τ, we iterate over the edge set $\mathcal{E}$ and greedily add the edge which gives us minimal marginal gain, while ensuring Hamiltonian tour properties (no sub-tours, with the degree of each vertex being at-most 2).
- **Selecting best tour set :** We finally select the tour which results in the minimal value of the function $H(\mathcal{S}, \tau)$ as the best tour and store the value of τ.

- Since the edges have stochastic rewards and costs, it is not possible to accurately estimate the function $H(\mathcal{S}, \tau)$.
- Hence for this reason, we use an oracle function $\mathbb{O}$, which takes several samples to approximate the auxiliary function.
- Finally, we prove RAGA has a polynomial running time of $O(\lceil \frac{\Gamma}{\gamma} \rceil (|\mathcal{V}|^3 (|\mathcal{V}| n_p + n_s + 2\log|\mathcal{V}| + 1)))$, with two constants n_p and n_s.

Change in tour selection with variation in β. A smaller value of β, places more emphasis on cost minimization than reward maximization.



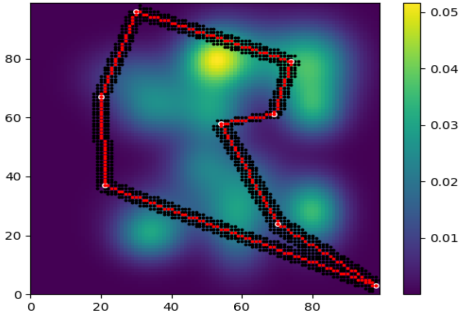
$\alpha = 0.1, \beta = 0$



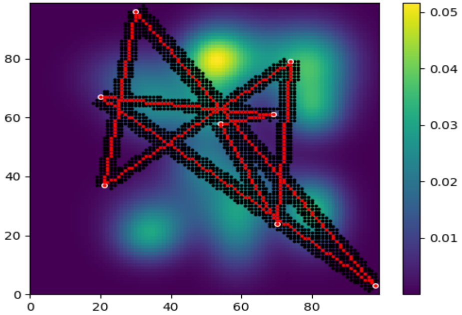
$\alpha = 0.1, \beta = 1$

Results

Change in tour selection with variation in α. A larger value of α results in a riskier tour (higher cost, higher reward).



$\alpha = 0.1, \beta = 0$



$\alpha = 1, \beta = 0$

Conclusion

- In this paper, we developed a risk-aware greedy algorithm for the TSP, while considering the risk-reward trade-off.
- We used a CVaR submodular maximization approach for selecting a solution set under matroidal constraints.
- We also analyzed the performance of RAGA and its running time. The results show RAGA's efficiency in optimizing both reward and cost simultaneously.

Future Work

- An ongoing work is to improve the running time of RAGA as a larger value of α has a higher running time.
- Another future avenue is to extend RAGA to address arbitrary distributions for the reward and cost, based on the real-world data.
- The algorithm can also be extended to other types of combinatorial optimization problems, and for the cases where the risk associated with the graph can be learned and the paths can be determined dynamically.
- Finally, RAGA can be generalized to the stochastic version of multi-robot TSP.

Thank You !