A Machine Learning-Enhanced Column Generation for Vaccine Distribution

A. Kuangying Li[a], B. Hiruni Niwunhella[b], C. Ali Hajbabaie, Ph.D.[c] and D. Leila Hajibabai, Ph.D.[b,*]

[a]Operations Research Program, North Carolina State University, Raleigh, United States
a.kli32@ncsu.edu,

[b]Edwards P. Fitts Department of Industrial and Systems Engineering, North Carolina State University, Raleigh, United States
b.dhniwunh@ncsu.edu, d.lhajiba@ncsu.edu,

[c]Department of Civil, Construction, and Environmental Engineering, North Carolina State University, Raleigh, United States
c.ahajbab@ncsu.edu.

[*] Corresponding author

Extended abstract submitted for presentation at the Conference in Emerging Technologies in Transportation Systems (TRC-30)
September 02-03, 2024, Crete, Greece

April 29, 2024

Keywords: Machine Learning Based Column Generation, Linear SVM, Location-Allocation, Vaccine Distribution, Equity

1 INTRODUCTION

Our previous research introduces a modified Voronoi Diagram method to enhance the efficiency of COVID-19 vaccine distribution at the state level. This method shows a 5.92% reduction in transportation costs and a 28.15% increase in demand coverage, despite higher transportation expenses, when compared to the widely used column generation technique. While both methods effectively tackle the distribution problem, they experience large CPU times due to the complexity of decision variables and the large-scale nature of the data. Our paper focuses on improving computational efficiency while maintaining the quality of the solution.

The literature suggests various methods to enhance the efficiency of Voronoi diagram-based techniques. For instance, Lipin (2014) introduced a convex hull method and Chen & Merkel (2006) utilized such techniques to reduce selection overhead in random testing. Additionally, Li & Liu (2020), Ohya *et al.* (1984), Qin *et al.* (2017), and Karavelas (2004) each presented strategies to reduce computational redundancy and enhance efficiency. However, these methods do not directly apply to our modified Voronoi Diagram due to differences in reward functions and sub-region reshaping strategies. To address this, we propose developing a new algorithm that incorporates machine learning into an enhanced column generation (CG) method, aiming to improve the runtime.

2 METHODOLOGY & RESULTS

We develop a two-phase machine learning-based solution prediction framework: (i) offline training for specific optimization problems and (ii) implementation of the trained model on untrained datasets to predict optimal solutions, as illustrated in Figure 1.

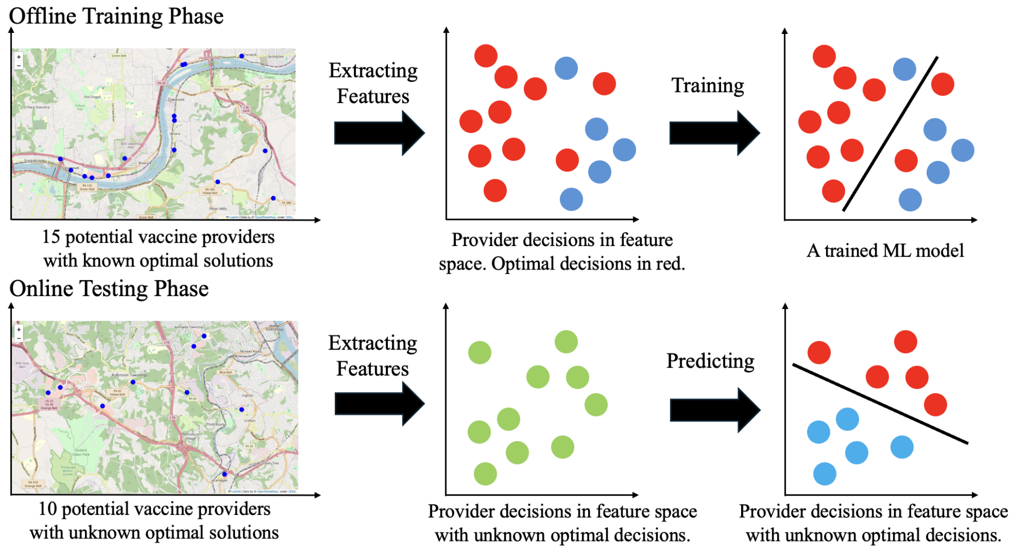

Figure 1 – *The machine learning-based solution prediction framework.*

During the offline training phase, the proposed methodology trains a model using an optimization problem that reflects the structure of our previously developed vaccine distribution location-allocation model with known optimal solutions. The graphical representation in the upper left corner of Figure 1 shows potential vaccine provider locations as blue dots. By leveraging the knowledge of the optimal solution within this training dataset, we effectively visualize decision variables within the feature space. In this, providers selected in the optimal solution are marked with red dots, while others are blue. We then use a linear support vector machine (SVM) model to create a hyperplane (i.e., a decision boundary) that effectively classifies the binary decision variable values of 0 and 1.

Following the training of the linear SVM model with data from a known optimal solution, our next step is to incorporate this model into a location-allocation problem that shares the same structural attributes as the original. This ensures consistency across index sets, decision variables, the objective function, and constraints. By doing this, we leverage the predictive power of the linear SVM model to extrapolate the unknown optimal solutions for new problem instances.

The direct prediction of the optimal solution for the location-allocation problem has yielded undesirable results. Therefore, we have utilized the prediction solution to refine the column-selection strategy in the CG method Shen *et al.* (2022). The flowchart in Figure 2 illustrates how the algorithm integrates the prediction solution. We compare the results of the original CG method (Scenario 1) with those obtained by enhancing the column-selection strategy through the prediction solution (Scenario 2), as depicted in Figures 3 and 4.

To enhance both runtime efficiency and predictive accuracy, we have retrained our model to forecast outcomes for the restricted master problem (RMP) and dual master problem (DMP) within the CG method. Scenarios 3, 4, and 5 use the prediction model to forecast RMP or DMP solutions every 5 iterations, with the Gurobi solver handling the remaining iterations. Note that Gurobi has ensured high solution quality in the final iteration of each scenario. This machine learning-based approach significantly reduced runtime from 7.4 hours in Scenario 2 to 1.6 hours in Scenario 5, though with a 43% decrease in solution quality, representing an 80% runtime reduction. Subsequently, we have defined Scenario 6, which extended the logic of Scenario 5 but adjusted the prediction intervals to every 10 iterations. Scenario 6 further cut runtime to 1.2 hours but increased the optimal gap to 25.7%, positioning it as the most time-efficient scenario. This highlights the need to improve solution quality.

Our initial strategy aims to enhance the accuracy of the prediction model to improve the

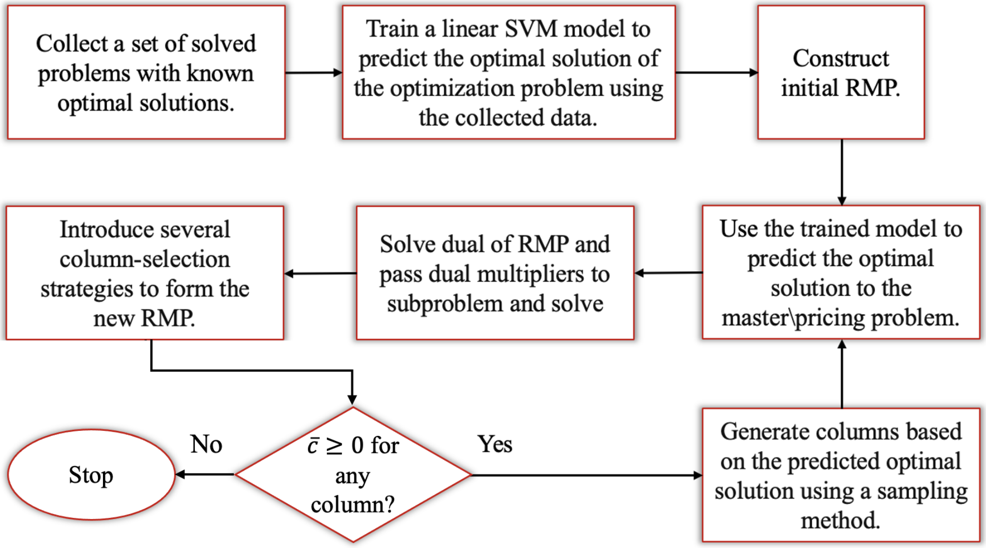

Figure 2 – *The flowchart of the integrated column generation and machine learning method.*

quality of the optimal solutions. We have expanded the training dataset for the machine learning model by including data from all 21 iterations, rather than just three, as in the modified Voronoi Diagram method. We have then defined Scenarios 7, 8, and 9 as counterparts to Scenarios 3, 4, and 5, using the improved prediction model. The new model demonstrated better runtime and solution quality. In Scenario 9, runtime was reduced to 1.4 hours—nearly matching our best runtime—and the optimal gap decreased by approximately 4%. However, the optimal gap in Scenario 9 still remained at 21.7%, indicating a need for further improvement to achieve near-optimal solutions.

Our primary model is trained to predict two key variables: the location decision for vaccine providers and the demand coverage decision. These predictions show higher accuracy compared to other variables. Therefore, we have used the outcomes of these two predictions as inputs to rerun the CG method for the remaining decision variables. This strategy, referred to as Scenario 10, effectively maintains input accuracy while reducing the number of decision variables, improving the solution quality significantly. According to Figure 3, Scenario 10 has achieved the lowest optimal gap among all scenarios at 5.7%. However, this improvement require increased computation time, extending the runtime to 6 hours.

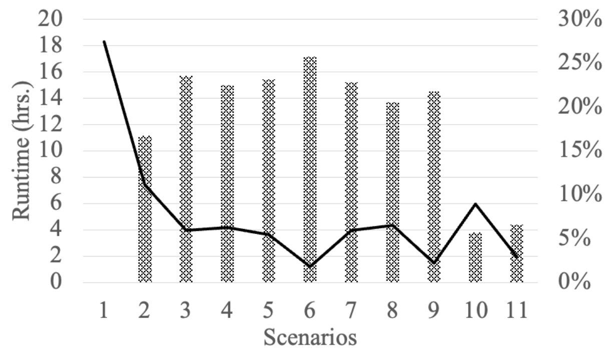

Figure 3 – *The runtime & optimality gap among different scenarios.*

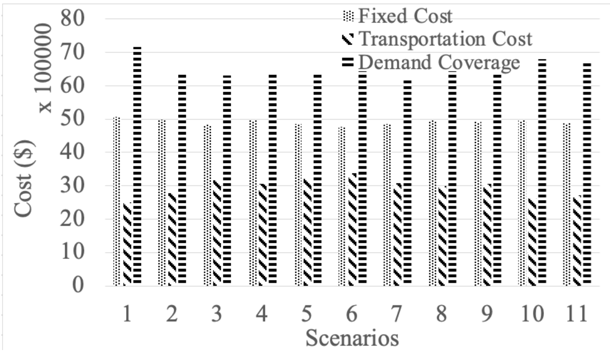

Figure 4 – *The objective values among different scenarios.*

References

Chen, Tsong Yueh, & Merkel, Robert. 2006. Efficient and effective random testing using the Voronoi diagram. *Pages 6–pp of: Australian Software Engineering Conference (ASWEC'06)*. IEEE.

Karavelas, Menelaos I. 2004. A robust and efficient implementation for the segment Voronoi diagram. *Pages 51–62 of: International symposium on Voronoi diagrams in science and engineering*, vol. 2004.

Li, Yang, & Liu, Gang. 2020. Area queries based on voronoi diagrams. *Pages 2064–2068 of: 2020 IEEE 36th International Conference on Data Engineering (ICDE)*. IEEE.

Lipin, Zhang. 2014. Research on Methods of Construction of Voronoi Diagram and Nearest Neighbor Query in Constrained Regions. *Computer Science.*

Ohya, Takao, Iri, Masao, & Murota, Kazuo. 1984. IMPROVEMENTS OF THE INCREMENTAL METHOD FOR THE VORONOI DIAGRAM WITH COMPUTATIONAL COMPARISON OF VARIOUS ALGORITHMS. *Journal of The Operations Research Society of Japan*, **27**, 306–337.

Qin, Yipeng, Yu, Hongchuan, & Zhang, Jianjun. 2017. Fast and Memory-Efficient Voronoi Diagram Construction on Triangle Meshes. *Pages 93–104 of: Computer Graphics Forum*, vol. 36. Wiley Online Library.

Shen, Yunzhuang, Sun, Yuan, Li, Xiaodong, Eberhard, Andrew, & Ernst, Andreas. 2022. Enhancing column generation by a machine-learning-based pricing heuristic for graph coloring. *Pages 9926–9934 of: Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36.