WGU C951

Task 3

Machine Learning Proposal for Delivery Vehicle Routing Problem

Tyler Gautney

Student ID #001400532

24 October 2024

Table of Contents

[A. Project Overview 3](#_Toc180747877)

[1. Organizational Need 3](#_Toc180747878)

[2. Context and Background 3](#_Toc180747879)

[3. Outside Works Review 3](#_Toc180747880)

[4. Solution Summary 4](#_Toc180747881)

[5. Machine Learning Solution Benefits 5](#_Toc180747882)

[B. Machine Learning Project Design 5](#_Toc180747883)

[1. Scope 5](#_Toc180747884)

[2. Goals, Objectives, and Deliverables 6](#_Toc180747885)

[3. Standard Methodology 8](#_Toc180747886)

[4. Projected Timeline 9](#_Toc180747887)

[5. Resources and Costs 10](#_Toc180747888)

[6. Evaluation Criteria 10](#_Toc180747889)

[C. Machine Learning Solution Design 11](#_Toc180747890)

[1. Hypothesis 11](#_Toc180747891)

[2. Selected Algorithm 11](#_Toc180747892)

[a. Algorithm Justification 11](#_Toc180747893)

[b. Algorithm Advantage 12](#_Toc180747894)

[c. Algorithm Limitation 12](#_Toc180747895)

[3. Tools and Environment 13](#_Toc180747896)

[4. Performance Measurement 13](#_Toc180747897)

[D. Description of Data Sets 13](#_Toc180747898)

[1. Data Source 13](#_Toc180747899)

[2. Data Collection Method 13](#_Toc180747900)

[a. Data Collection Method Advantage 13](#_Toc180747901)

[b. Data Collection Method Limitation 13](#_Toc180747902)

[3. Quality and Completeness of Data 13](#_Toc180747903)

[4. Precautions for Sensitive Data 14](#_Toc180747904)

[E. References 14](#_Toc180747905)

# Project Overview

This proposal describes machine learning applications for solving a particular type of constrained optimization problem. Constrained optimization problems are challenging for computer systems because algorithms need to be able “… [to find] the best balance of competing factors while ensuring that the solution is actually possible”. Such algorithms must often maximize the reward function(s) that compete directly with minimize the cost function(s). Constrained optimization problems are the hidden architecture behind practically every human endeavor in the modern world.

## Organizational Need

A common constrained optimization problem is that of the delivery vehicle routing problem. Algorithms that attempt to solve this variation of the travelling salesman problem must find the most optimal set of routes for a fleet of vehicles to traverse in order to deliver a set of packages to a set of destinations. This problem is widely encountered in the logistics sector and businesses’ internal distribution and logistics operations. It can also be applied to fields such as entertainment, tourism, and production. Algorithms must be able to find the most optimal routes for multiple vehicles visiting a set of locations.

## Context and Background

WGUPS is a localized logistics supply company for organizations and consumers in Utah. In the past year, business has exploded with double digit growth and expansion outside of its Salt Lake City origin. As a result, its exponential business growth has outpaced its technological capabilities as its internal package delivery fleet, daily package load, and service area has grown. The current system relies on the greedy nearest neighbor algorithm for assigning packages to trucks and subsequently routing packages to their destination.

## Outside Works Review

### Simulated Annealing

The Simulated Annealing algorithm is a heuristic approach that analogizes the annealing of “… a material such as metal or glass by raising it to a temperature and then gradually reducing that temperature…” in that it makes random changes to the minimizing function until the temperature cools. (Walker, 2018) When the temperature is high, the algorithm is more willing to accept random changes that may be detrimental to the solution. As the temperature falls by order of an exponential decay function, the algorithm becomes less and less receptive to changes that do not improve the outcome. This algorithm is more accurate on larger datasets with a lower runtime complexity, but its heuristic nature means that it still doesn’t always find the most optimal solution.

### Combinatorial Bees Algorithm

The combinatorial bees algorithm mimics the nectar-seeking behavior of honey bees, and it competes well with other methods. It can be used to enhance the nearest neighbor algorithm – which is good for a codebase that already uses that algorithm to solve the delivery vehicle routing problem. The Nearest neighbor algorithm is the most basic heuristic solution to this problem that is easy to implement with a relatively straightforward processing load, but it doesn’t scale well with larger datasets. The combinatorial bees, using the nearest neighbor method to start the optimization for the Bees Algorithm population stage, solves the optimization problem iteratively. It has local and global search sections. In the local search section, more neighborhood searches are made on elite sites, while fewer searches are made on other sites. The algorithm starts with the determination of some parameters, and local search routes are changed with some operators for trying to find new routes with less cost. With each iteration, the algorithm selects a different operator – swap, insert, or revision – to use with the local search function. (Sahin, 2022)

### Branch and Bound Method

The Branch and Bound Method is a nonheuristic algorithm that breaks the problem into smaller subproblems and solves them one by one. It then uses a bounding function to determine whether the solution is optimal or not. If it is not optimal, it discards the solution and moves onto the next. If it is optimal, it updates the best solution found so far. This algorithm is the best algorithm of the three for finding the most optimal solution; however, it is also the slowest to execute with a worst-case runtime “…that grows exponentially with problem size.” (Boyd & Mattingley, 2018)

## Solution Summary

In summary, the delivery vehicle routing problem is a commonplace constrained optimization problem faced by businesses throughout nearly every industrial sector. Algorithms must attempt to minimize the distance travelled by a set of vehicles visiting a set of destinations in a computationally inexpensive manner. WGUPS confronts this problem within its core logistics and distribution business model. The current algorithm for solving this problem isn’t the most optimal while also being slow for the organization’s rapidly expanding business.

## Machine Learning Solution Benefits

Machine Learning algorithms; such as the simulated annealing algorithm, combinatorial bees algorithm, and branch and bound algorithm; can ingest the large amount of delivery destination, fleet, and package data in a computationally efficient manner while still getting close to the most optimal solution. There is some trade off between these two competing factors. Algorithms that get closer to the most optimal solution are most often the most computationally complex – thus requiring a longer runtime. Conversely, algorithms that are quicker often do not always achieve the most optimal routes for fleets to travel.

# Machine Learning Project Design

WGUPS will use the simulated annealing algorithm to facilitate efficient, effective delivery of packages to its customers with its fleet of delivery vehicles. An iterative approach is the best approach, so this project proposal outlines the first stage. The scope of the project is outlined as follows:

## Scope

### In-Scope Objectives

* Develop a simulated annealing algorithm:Create and implement a simulated annealing algorithm tailored to solve the vehicle routing problem. This includes defining the cooling schedule, neighborhood search, and acceptance criteria for solutions.
* Optimize Fleet Management:Utilize the simulated annealing algorithm to optimize the routing of a fleet of vehicles, aiming to minimize total travel distance, fuel consumption, and delivery time while considering constraints such as vehicle capacity, the number of available vehicles.
* Leverage existing codebase:Utilize the organization’s preexisting codebase as the foundation for the project. This involves integrating the simulated annealing algorithm with the existing software infrastructure: replacing its ageing use of the nearest neighbor algorithm with a leaner, cleaner design.

### Out-of-Scope Objectives

#### Integration of Real-World Data: The project will not include the incorporation of real-world data such as traffic conditions, road networks, and customer locations. This will be considered for future project cycles.

#### Delivery Time Window Constraints: The project will not address delivery time window constraints. This objective will be considered for the next project cycle.

#### Development of a Real-Time Tracking System: The project will not include the development of a real-time vehicle tracking system for fleet vehicles. While optimizing routes is within scope, the implementation of a system to monitor vehicle locations in real-time is considered out-of-scope for this effort.

## Goals, Objectives, and Deliverables

### Goals

#### Increase Route Optimization Efficiency: Achieve a 15% reduction in the time required to generate optimized vehicle routes using the simulated annealing algorithm, compared to the current heuristic-based Nearest Neighbor Method.

#### Reduce Operational Costs: Decrease overall fleet operational costs by 15% through optimized routing, leading to lower fuel consumption, reduced vehicle wear and tear, and minimized overtime expenses for drivers.

#### Enhance Algorithm Performance: Improve the performance of the simulated annealing algorithm by 30% through iterative testing and refinement, ensuring more accurate and reliable routing solutions that outperform the current nearest neighbor method.

### Objectives

#### Increase Route Optimization Efficiency

1. Implement and fine-tune the simulated annealing algorithm to ensure it generates optimized routes within a 15% shorter timeframe compared to the nearest neighbor method.
2. Conduct performance benchmarking and analysis to identify bottlenecks and areas for improvement in the algorithm’s execution time.
3. Train team members on the new algorithm to ensure that they can effectively utilize it for route optimization tasks.

#### Reduce Operational Costs

1. Analyze and compare fuel consumption data before and after implementation of the simulated annealing algorithm to quantify cost savings.
2. Monitor vehicle maintenance records to track reductions in wear and tear due to optimized routing.
3. Evaluate driver overtime expenses to measure the impact of more efficient routing on labor costs.

#### Enhance Algorithm Performance

1. Conduct iterative testing and refinement of the simulated annealing algorithm to achieve a 30% improvement in solution quality over the nearest neighbor method.
2. Integrate feedback from real-world testing to continuously enhance the algorithm’s accuracy and reliability.
3. Document and implement best practices for algorithm development and optimization to ensure sustained performance improvements.

### Deliverables

In addition to the software and its source code, here are the objective deliverables for this project:

#### Project Plan and Timeline: A detailed project plan outlining the timeline, milestones, and deliverables for this project. This should include a Gantt chart.

#### Technical Documentation: Comprehensive documentation detailing the design, implementation, and usage of the simulated annealing algorithm. This should include code comments, API documentation, and user manuals.

#### Deployment Guide: A step-by-step guide for deploying the software in the organization’s existing infrastructure. This should cover installation, configuration, and troubleshooting.

#### Test Cases and Results: A suite of test cases used to validate the algorithm, along with the results of those tests. This should include unit tests, integration tests, and performance tests.

#### Maintenance Plan: A plan for ongoing maintenance and support of the software, including procedures for handling bug fixes, updates, and enhancements.

## Standard Methodology

Development will follow the CRISP-DM methodology:

### Business Understanding

* Schedule meetings with key stakeholders to gather requirements and define project objectives.
* Develop a project charter that outlines the scope, goals, success criteria, and high-level deliverables.
* Perform a risk assessment to identify potential challenges and mitigation strategies.

### Data Understanding:

* Facilitate collaboration among data owners to collect relevant datasets, such as vehicle routes and fleet information.
* Use data profiling tools to assess data quality, completeness, and consistency.
* And conduct exploratory data analysis sessions to visualize data patterns and gain insights.

### Data Preparation:

* Implement data cleaning processes to handle missing values, outliers, and inconsistencies.
* Develop new features that enhance the algorithm’s performance, such as distance metrics and vehicle capacities.
* Split the data into training and testing sets to validate the algorithm’s effectiveness.

### Modeling:

* Develop the simulated annealing algorithm, including defining the cooling schedule and neighborhood search function.
* Integrate the algorithm with the existing codebase, ensuring compatibility and efficiency.
* Train the algorithm using the prepared data and fine-tune hyperparameters for optimal performance.

### Evaluation:

* Define and calculate performance metrics to evaluate the algorithm’s effectiveness.
* Compare the algorithm’s performance against the nearest neighbor method.
* Collect feedback from stakeholders and drivers to assess the algorithm’s practical utility.

### Deployment:

* Develop a detailed deployment plan, including installation, configuration, and troubleshooting steps.
* Ensure seamless integration of the algorithm with the organization’s existing software infrastructure.
* Conduct training sessions for the deployment and development teams to ensure that they can effectively use and maintain the new algorithm.

## Projected Timeline

November 1, 2024 – The proposal is accepted

January 6, 2025 – A technical proof of concept is presented.

January 10, 2025 – Submitted for review

April 7, 2025 – Model Training begins

July 7, 2025 – MVP Software Delivery

August 7, 2025 – Technical Documentation, Deployment Guide, Test Cases and Results, and Maintenance Plan Complete

|  |  |  |  |
| --- | --- | --- | --- |
| Proof of Concept Sprint Schedule | | | |
| **Sprint** | **Start** | **End** | **Tasks** |
| 1 | November 4, 2024 | November 15, 2024 | Project Team starts development |
| 2 | November 18, 2024 | November 26, 2024 | Lead Developer designs architecture and delegates coding tasks |
| 3 | December 2, 2024 | December 13, 2024 | Project management completes plan for next phase of development |
| 4 | December 16, 2024 | January 3, 2025 | Complete Proof of concept |

## Resources and Costs

|  |  |  |
| --- | --- | --- |
| **Resource** | **Description** | **Cost** |
| Salaries | This includes the salaries for the Project Manager, Lead developer, Scrum Master, development team, test team, and deployment team | $300,000 |
| Software Licenses and Subscriptions | Fees and costs for development tools, data analysis tools, and monitoring tools. | $60,000 |
| Hardware Upgrades | This covers any additional development workstations that need to be acquired or upgraded. | $30,000 |
| AI Training Server | This covers the cost of a Supermicro AI Training SuperServer SYS-822GA-NGR3 Complete System | $105,000 |
|  | **Total** | $495,000.00 |

## Evaluation Criteria

|  |  |
| --- | --- |
| **Objective** | **Success Criteria** |
| Increase Route Optimization Efficiency | Track and compare the time taken to generate optimized routes using the simulated annealing algorithm versus the nearest neighbor method. This can be done by running both algorithms on the same set of data and recording the time taken for each. |
| Reduce Operational Costs | Monitor and compare the overall fleet operational costs before and after implementing the simulated annealing algorithm. This includes tracking fuel consumption, vehicle maintenance costs, and driver overtime expenses. The organization can use financial reports and expense tracking tools to gather this data. |
| Enhance Algorithm Performance | Evaluate the quality of the solutions generated by the simulated annealing algorithm by comparing key performance indicators such as total travel distance, number of vehicles used, and adherence to capacity constraints. Conduct iterative testing and refinement to measure improvements in these indicators over time. |

# Machine Learning Solution Design

## Hypothesis

Replacing the nearest neighbor method currently in use by the organization’s software delivery system will result in reduced computational load, increased route efficiency, and reduced operational costs.

## Selected Algorithm

The simulated annealing algorithm will use the reinforcement machine learning model to automatically adjust the algorithm’s temperature schedule, random step selection, and the probability that the algorithm will accept a route at a given temperature based upon previous iterations of the algorithm.

### Algorithm Justification

With simulated annealing, the solution is more likely to explore parts of the search space that other local-search algorithms alone, such as 2-opt and 3-opt algorithms, would have never even considered. Nonetheless, however, the simulated annealing algorithm is, after all, also a local-search heuristic, which means that it is still possible to get trapped in a local minima. Furthermore, with practical usage of the simulated annealing algorithm, it’s often tricky to tune the appropriate scheduling of the temperature and the algorithm’s probability of accepting certain outcomes at any given temperature. Thus, finding a way to balance the time it takes to converge, and also sufficiently exploring the search space is no easy task. Using reinforcement learning attempts to alleviate, or even eliminate, these problems with traditional simulated annealing. In a research paper proposed in the journal, Mathematics, researchers proposed a novel approach to solving the traveling salesman problem with the JAYA metaheuristic algorithm based upon reinforcement learning and simulated annealing. In it, the researchers used the basic Q-learning algorithm (a type of reinforcement learning algorithm) to update the solution in the current state, the Metropolis acceptance criterion of the simulated annealing algorithm to determine whether to accept candidate solutions, and the 3-opt selection process applied to the best solution of the current iteration at a certain frequency. Their experimental findings hold that this algorithm achieves significantly better results in most instances compared to the traditional JAYA algorithm. (Xu et al., 2023) Ultimately, this proposal does not contend that our solution should (or should not) use the exact same methodology as these researchers; however, their solution presents interesting results that can at least be partially incorporated into our internal effort.

### Algorithm Advantage

The Vehicle Routing Problem, which is merely a variation on the Travelling Salesman Problem, is fundamentally an NP-hard problem; that is, there is no polynomial time algorithm that can find the most optimal route. This means that this proposal’s objective goal of finding a more optimal solution is in direct opposition to the objective goal of reducing the time required to generate optimal routes. That is, algorithms that produce more optimal routes take longer to complete with their worst-case runtime complexity; and algorithms that take less time to complete are less consistently accurate at finding more optimal routes. Therefore, this approach means that the algorithm can train itself to be more efficient and more accurate over time as it learns from its past approaches to an optimal solution.

### Algorithm Limitation

Reinforcement learning, in general, has three key limitations applicable to this solution. First, reinforcement learning can exhibit high variance and instability during the learning process. Thus, it may be challenging to train the model in such a way that it achieves consistent performance. Second, it requires a huge amount of data. Therefore, training the model can be time consuming and computationally expensive. Thirdly, and perhaps most obviously, enhancing the simulated annealing algorithm with reinforcement learning increases the overall complexity of the algorithm; thereby making it more challenging to implement and understand.

## Tools and Environment

The model will be trained with PyTorch on Red Hat Enterprise Linux with Intel’s new Gaudi AI Platform.

## Performance Measurement

The existing nearest neighbor algorithm’s performance will be used as a baseline for this model’s performance. The model will receive rewards if the cooling schedule results in a more optimized result faster than that of the nearest neighbor algorithm for the given data.

# Description of Data Sets

## Data Source

We will use a dataset containing information on customer locations, delivery windows, vehicle capacities, and distances between locations.

## Data Collection Method

The data will be collected from our existing logistics database and supplemented with GPS data from our fleet.

### Data Collection Method Advantage

By combining data from the logistics database and GPS, we get a more comprehensive view of the operations. The logistics database provides historical and structured data that can be used alongside our nearest neighbor algorithm for data training purposes, while GPS data offers real-time updates on vehicle locations. This combination allows for more accurate and dynamic decision-making, which is crucial for solving the Vehicle Routing Problem effectively.

### Data Collection Method Limitation

Since the data is collected from multiple sources, there might be inconsistencies or inaccuracies. GPS data can sometimes be imprecise due to signal loss or interference – leading to incorrect location information. Additionally, integrating data from different sources can introduce discrepancies that need to be resolved during preprocessing. GPS data can also sometimes include outliers caused by drivers deviating from their assigned routes for personal reasons, such as taking lunch breaks or making unscheduled stops. These deviations can introduce noise into the dataset, making it more challenging to accurately model and optimize the routes. To address this, additional filtering and validation steps may be necessary to identify and exclude such outliers from the analysis.

## Quality and Completeness of Data

To ensure data quality, we will preprocess the data to handle missing values and outliers. The dataset is expected to contain approximately 15,000 records and will be updated daily.

## Precautions for Sensitive Data

Data will be stored in a secure cloud database, with access restricted to authorized personnel to maintain data privacy and security.

# References

Walker, J. (2018, June). *Simulated annealing*. Simulated Annealing: The Travelling Salesman Problem. [https://www.fourmilab.ch/documents/travelling/anneal/](https://www.fourmilab.ch/documents/travelling/anneal/%20)

Boyd, S., & Mattingley, J. (2018, April 1). Branch and bound methods - Stanford University. [https://stanford.edu/class/ee364b/lectures/bb\_notes.pdf](https://stanford.edu/class/ee364b/lectures/bb_notes.pdf%20)

Sahin, M. (2022). Solving TSP by using combinatorial bees algorithm with nearest neighbor method. Neural Computing and Applications, 35(2), 1863–1879. <https://doi.org/10.1007/s00521-022-07816-y>

Xu, J., Hu, W., Gu, W., & Yu, Y. (2023). A discrete Jaya algorithm based on reinforcement learning and simulated annealing for the traveling salesman problem. Mathematics, 11(14), 3221. <https://doi.org/10.3390/math11143221>

Google. (2024, August 28). Vehicle routing problem  |  or-tools  |  google for developers. Google. <https://developers.google.com/optimization/routing/vrp>