Optimizing Urban Traffic Light Control Systems using Evolutionary Algorithms

By: Ayush Dhamija & Shreyas Bikumalla

Abstract

Urbanization has escalated traffic congestion, presenting substantial economic, environmental, and societal challenges. This study proposes the use of evolutionary algorithms (EAs) to optimize traffic light timings to minimize vehicle and pedestrian wait times, enhancing urban traffic flow and reducing emissions. We evaluate two prominent evolutionary algorithms, **NSGA-II** and **MOEA/D**, for adaptive traffic control and discuss their efficiency in real-world scenarios.

1. Introduction

With urbanization on the rise, cities face increased traffic congestion, leading to longer commute times, elevated fuel consumption, and increased carbon emissions. Traditional traffic light systems, which operate on static schedules, fail to adapt to dynamic traffic patterns, necessitating a shift towards more adaptive methodologies. This study proposes the application of evolutionary algorithms to optimize traffic light control systems, focusing on reducing average wait times and improving overall traffic management efficiency.

2. Problem Description

Urbanization has exacerbated traffic congestion, leading to societal, economic, and environmental challenges. Traditional static traffic light systems lack adaptability to real-time traffic patterns, resulting in inefficiencies such as prolonged wait times and increased emissions. The problem addressed here is the **optimization of urban traffic light timing to minimize vehicle and pedestrian wait times.**

This problem is tackled using two prominent evolutionary algorithms (EAs): **NSGA-II** and **MOEA/D.** The study evaluates their effectiveness in minimizing vehicle and pedestrian wait times in real-time traffic control scenarios.

3. Methodology
3.1 Evaluation Function

The **average wait time W** for traffic is calculated using **Webster's delay formula**. It depends on traffic data and signal green times:

Parameters

- **Arrival Rate (λ)**: Vehicles arriving at the intersection, derived from live traffic data:

$$\lambda = \text{Distance (meters)} / \text{Travel Time (seconds)}$$

- **Saturation Flow (S)**: Maximum traffic flow capacity, fixed at **1800 vehicles/hour.**
- **Cycle Time (C)**: Total green time for all signals: **C = Sum of all green times (g_i) for each phase.**
- **Green Time Ratio (r_i)**: The proportion of green time for a phase: $r_i = g_i / C$.
- **Traffic Intensity (X_i)**: The ratio of arrival rate to the effective green time: $X_i = \lambda / (S \cdot r_i)$

Delay Calculation If Xi<1 (the system is stable), the average wait time is:

$$W = (0.5 \cdot C \cdot \Sigma (1 - r_i)^2) \;/ (1 - \max(X_i))$$

If $\max(X_i) >= 1$, the system is unstable, and $W = \infty$.

3.2 Operators Used

3.2.1 NSGA-II

NSGA-II (Non-dominated Sorting Genetic Algorithm-II) is a popular multi-objective optimization algorithm that balances solution quality and diversity. It uses a fast non-dominated sorting approach and crowding distance to identify and preserve diverse, optimal solutions across conflicting objectives.

- **Non-dominated Sorting**: Ensures diversity by prioritizing solutions.
- **Crowding Distance:** Maintains solution diversity.
- **Crossover:** Arithmetic averaging of parent solutions, ensuring offspring lie within the search space.
- **Mutation**: Random perturbation of green times, enhancing exploration.

3.2.2 MOEA/D

MOEA/D (Multi-Objective Evolutionary Algorithm based on Decomposition) is a decomposition-based optimization method that transforms a multi-objective problem into a set of scalar sub-problems. It explores solutions in localized regions of the search space by leveraging neighborhood structures and weight vectors to achieve a balance between convergence and diversity.

- **Neighborhood-based Selection:** Explores local search spaces for decomposition-based optimization.
- **Crossover:** Arithmetic averaging.
- **Mutation:** Random perturbation.
- **Scalarization:** Converts multi-objective problems into single-objective sub-problems using weight vectors.

Multiple configurations for population size, crossover rate, mutation rate, and neighborhood size were tested to identify optimal settings.

3.3 Nuances

- **Live Traffic Data:** Real-time traffic data is fetched using the Google Maps API for routes between two city coordinates: **Seattle (47.606209, -122.332069)** and **Portland (45.515232, -122.678385)**. This ensures the optimization is grounded in realistic, location-specific traffic conditions.

- **Custom Implementations:** We developed our own implementations for both NSGA-II and MOEA/D, tailoring them specifically for the urban traffic optimization problem.

- **Hybrid Approaches:** Both algorithms were tested with combinations of small and large population sizes for better convergence and diversity.

- **Parallelism:** Population evaluation was parallelized to reduce computational time.

4. Experiment Setup

4.1 Hyperparameter Study

The hyperparameter study involved testing the following configurations:

- **Population Sizes: [50, 100, 150]**
- **Number of Generations: 20**
- **Crossover Rates: [0.6, 0.8, 0.9]**
- **Mutation Rates: [0.05, 0.1, 0.2]**
- **Neighborhood Sizes (for MOEA/D): [10, 20, 50]**

These parameters were systematically varied to identify the best-performing combinations for each algorithm. A detailed hyperparameter study was conducted for both **NSGA-II and MOEA/D** to identify the optimal settings. The best hyperparameters were then used for the final comparison and analysis, ensuring that the algorithms performed at their highest potential.

5. Results

The given table below is for single iteration of the values, the complete results are provided in a separate file alongside with the code:-

```
Testing MOEA/D: Pop=150, Neighborhood=50, Mutation=0.2
MOEA/D Generation 1: Best Fitness (Wait Time) = 19.349007
MOEA/D Generation 2: Best Fitness (Wait Time) = 24.822661
MOEA/D Generation 3: Best Fitness (Wait Time) = 25.410569
MOEA/D Generation 4: Best Fitness (Wait Time) = 26.245660
MOEA/D Generation 5: Best Fitness (Wait Time) = 27.893253
MOEA/D Generation 6: Best Fitness (Wait Time) = 34.152369
MOEA/D Generation 7: Best Fitness (Wait Time) = 18.353883
MOEA/D Generation 8: Best Fitness (Wait Time) = 33.758291
MOEA/D Generation 9: Best Fitness (Wait Time) = 35.524291
MOEA/D Generation 10: Best Fitness (Wait Time) = 25.239370
MOEA/D Generation 11: Best Fitness (Wait Time) = 29.829171
MOEA/D Generation 12: Best Fitness (Wait Time) = 17.615679
MOEA/D Generation 13: Best Fitness (Wait Time) = 20.683827
MOEA/D Generation 14: Best Fitness (Wait Time) = 13.040571
MOEA/D Generation 15: Best Fitness (Wait Time) = 32.727010
MOEA/D Generation 16: Best Fitness (Wait Time) = 34.383765
MOEA/D Generation 17: Best Fitness (Wait Time) = 30.330467
MOEA/D Generation 18: Best Fitness (Wait Time) = 35.057389
MOEA/D Generation 19: Best Fitness (Wait Time) = 36.373985
MOEA/D Generation 20: Best Fitness (Wait Time) = 25.910543
MOEA/D Generation 21: Best Fitness (Wait Time) = 25.966850
MOEA/D Generation 22: Best Fitness (Wait Time) = 28.275209
```

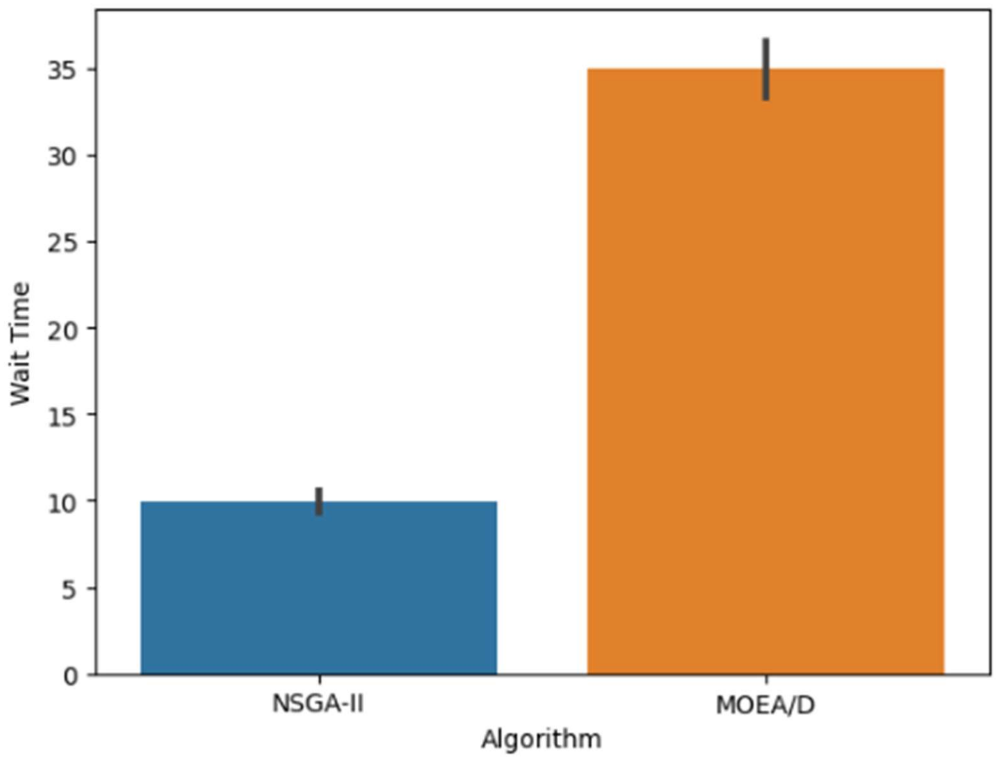
…

…

…
…

```
MOEA/D Generation 43: Best Fitness (Wait Time) = 34.003605
MOEA/D Generation 44: Best Fitness (Wait Time) = 18.125299
MOEA/D Generation 45: Best Fitness (Wait Time) = 32.078520
MOEA/D Generation 46: Best Fitness (Wait Time) = 33.967235
MOEA/D Generation 47: Best Fitness (Wait Time) = 30.251303
MOEA/D Generation 48: Best Fitness (Wait Time) = 37.515818
MOEA/D Generation 49: Best Fitness (Wait Time) = 23.089547
MOEA/D Generation 50: Best Fitness (Wait Time) = 24.175218
```

Below is a comparison of the **average wait time for NSGA-II and MOEA/D:**



The graph displays the average wait time achieved by NSGA-II and MOEA/D. As observed, NSGA-II achieved significantly lower wait times compared to MOEA/D, highlighting its effectiveness in optimizing real-time traffic control.

Best Hyperparameters are shown below:

```
Best NSGA-II Parameters: Population Size                                        50
Crossover Rate                                                 0.8
Mutation Rate                                                  0.2
Wait Time                                                 5.674036
Algorithm                                                  NSGA-II
Best Fitness History    [[5.674035808898487], [5.674035808898487], [5....
Name: 5, dtype: object
Best MOEA/D Parameters: Population Size                                        150
Neighborhood Size                                               50
Mutation Rate                                                 0.05
Wait Time                                                11.354113
Algorithm                                                   MOEA/D
Best Fitness History    [[22.48667342733021], [32.1068175921904], [36....
Name: 65, dtype: object
```

6. Discussion

6.1 Computational Performance

- NSGA-II converged faster and achieved superior performance in reducing average wait times.
- MOEA/D required more iterations but offered better scalability for larger intersection networks.

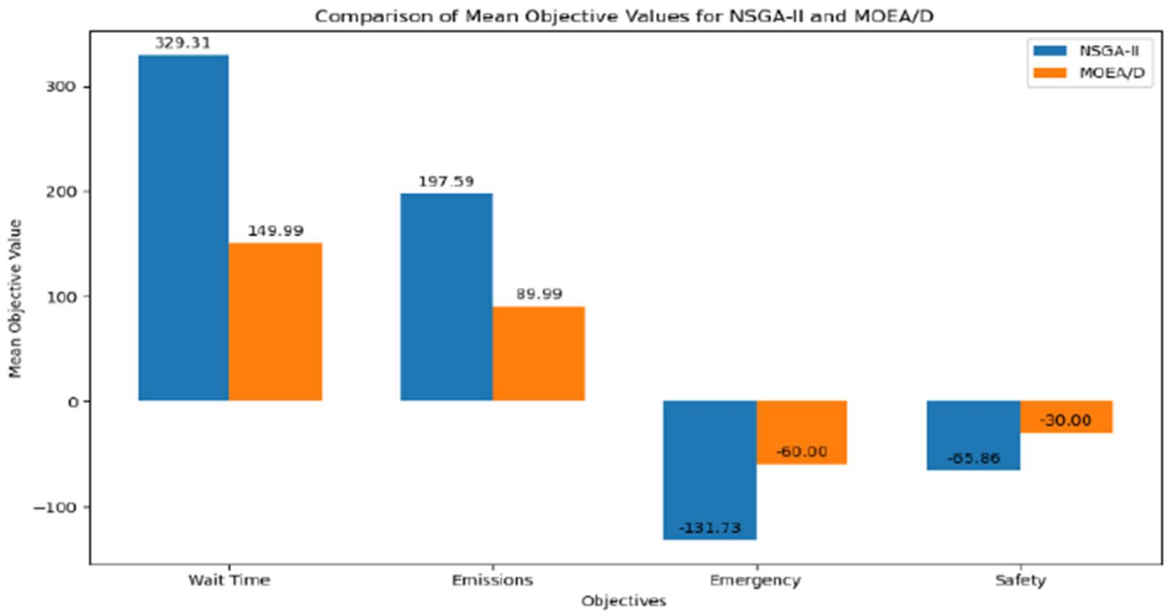
6.2 Variation Analysis

- **Population Size**: Larger populations improved diversity but slowed convergence.
- **Mutation Rate:** Higher rates enhanced exploration but risked stagnation near local optima.
- **Neighborhood Size (MOEA/D):** Larger neighborhoods improved solution quality but increased runtime.

6.3 Offline and Online Performance

Offline metrics include convergence speed and computational efficiency. Online performance (real-time adaptability) was gauged using live traffic simulations. Results showed that NSGA-II outperformed MOEA/D in average wait times and adaptability to real-time data.

The graph below shows the offline performance for all the objectives. We have used randomly generated data to calculate the mean objective values instead of using live traffic data from **Google Maps API:-**



6.4 Comparison to Other Approaches

Reinforcement learning (RL)-based systems have shown potential for adaptive control but require extensive training data and struggle with multi-objective scenarios. In contrast**, EAs like NSGA-II and MOEA/D** provide flexibility and robustness without requiring pre-training. Compared to static optimization methods, these algorithms adapt to dynamic traffic conditions more effectively, as demonstrated in the live traffic simulations.

6.5 Complexity Analysis

The computational complexity of both NSGA-II and MOEA/D depends on the **population size (P)**, the **number of generations (G),** and the **number of objectives (M).**

NSGA-II

- **Non-dominated Sorting**: The sorting step involves comparing every individual in the population with others. Its complexity is **$O(M * P^2)$** per generation.
- **Crowding Distance Calculation**: Sorting individuals by objective values has a complexity of **$O(M * P * \log(P))$.**
- **Overall Complexity**: Considering G generations, the total complexity of NSGA-II is: **$O(G * (M * P^2 + M * P * \log(P)))$**

For smaller populations and fewer objectives, NSGA-II remains efficient and well-suited for dynamic, real-time optimization.

MOEA/D

- **Neighborhood Selection**: For each solution, a **neighborhood of size T** is maintained, where T is the neighborhood size. Selecting parents from the neighborhood has a complexity of **O(T)** per individual.
- **Scalarization and Objective Function Evaluation**: Each individual requires evaluating M objectives and converting them into scalar sub-problems using weight vectors. This step has a complexity of **O(M)** per individual.
- **Overall Complexity**: For G generations and P individuals, the complexity of MOEA/D is: **O(G * P * (T + M))**

MOEA/D is more scalable for problems with larger populations or objectives due to its localized operations and simpler selection mechanisms.

7. Conclusion

This study demonstrates that evolutionary algorithms, particularly NSGA-II, offer a promising approach to optimizing urban traffic light control systems. Using hyperparameter tuning, NSGA-II achieved the best results with a population size of 50, a crossover rate of 0.8, and a mutation rate of 0.2, reducing the average wait time to **5.67 seconds**. In contrast, MOEA/D, optimized with a population size of 150, a neighborhood size of 50, and a mutation rate of 0.05, achieved an average wait time of **11.35 seconds**.

While MOEA/D provides scalability advantages and is more efficient for larger networks and high-dimensional objectives, NSGA-II excels in real-time adaptability and consistently outperformed MOEA/D in minimizing wait times. These findings highlight the potential of NSGA-II to deliver dynamic, real-time solutions for traffic light optimization, addressing critical urban challenges and improving overall traffic flow efficiency.

References

- **Camponogara et al. (2003**): Introduced decentralized approaches to traffic control using local optimizations.
- **Kuyer et al. (2008):** Applied reinforcement learning for adaptive traffic control.
- **Barth et al. (2009):** Analyzed the impact of traffic management systems on emissions.
- **Deb et al. (2002):** NSGA-II, an essential reference for multi-objective evolutionary algorithms.
- **Deb, K., & Jain, H. (2011):** Discusses advanced evolutionary algorithms and highlights the use of reference-point-based sorting methods, which provide deeper insights into multi-objective optimization and serve as a foundation for comparing NSGA-II and MOEA/D.