Multi-objective Optimisation of Traffic Light Control for Fast and Safe Traffic Incident Recovery

Qihong Yang, Zhuowei Zhao, Dong Zhao, A. K. Qin*, and Yu Sun
Department of Computing Technologies
Swinburne University of Technology, Hawthorn, Victoria 3122, Australia
{qihongyang, alvinzhao, dorazhao, kqin, yusun}@swin.edu.au

***Abstract*—In post-incident traffic management, traffic light control is one of the most widely adopted approaches. Although many existing studies can optimise traffic signal control patterns using multi-objective evolutionary optimisation techniques, the commonly used objective functions are not specifically designed for post-incident management and thus cannot effectively facilitate efficient traffic recovery. In this work, we first formulate traffic recovery as traffic conditions returning to normal within a predefined road area, and then propose two recovery-oriented objectives: Relaxed Traffic Recovery Time, which aims to minimise the time required for traffic to return to normal, and Overall Recovery Count, which seeks to improve recovery stability at the network level. The proposed recovery-oriented objectives, together with a safety-related objective (speed variance), are incorporated into NSGA-II and evaluated via traffic simulation. This results in a bi-objective optimisation approach for deriving traffic light control patterns that enable fast and safe post-incident traffic recovery. Experiments on both synthetic and real-world road networks demonstrate that, compared to traditional congestion-based objectives, the proposed recovery-oriented objectives lead to more effective traffic light control for post-incident traffic recovery.**

***Index Terms*—Traffic incident recovery, Traffic light control, Traffic incident management, Multi-objective optimisation**

I. Introduction

Post-incident traffic management aims to restore disrupted networks to their normal operating state as quickly and safely as possible. Among the available operational levers, traffic light control (TLC) is particularly effective and widely used [1]–[3]. Signalised intersections are present across nearly all urban networks, and their timing plans can be adjusted remotely via electronic controllers without requiring additional physical infrastructure. Consequently, TLC represents both the most fundamental and the most readily actionable mechanism for incident response.

Recent research has explored using multi-objective genetic algorithms to optimise traffic light patterns. These studies typically aim not only to improve congestion-related network performance indicators, such as average travel time [4]–[6], average travel delay [7], [8], queue length [9], [10], and throughput [11], [12], but also account for potentially conflicting objectives, including safety [13]–[15] and emissions [11], [12]. Although improving congestion-based objectives can alleviate congestion and contribute to faster traffic recovery, these objective functions are not specifically designed for post-incident scenarios, and none explicitly optimise from the perspective of how quickly the network recovers to normal conditions. As a result, existing formulations cannot directly support recovery-oriented TLC, even when applied in post-incident traffic management.

To address the absence of a recovery-oriented formulation, we first formulate the concept of traffic recovery for an affected region, enabling recovery itself to become a directly optimisable criterion. Based on this formulation, we propose two recovery-oriented objective functions: Relaxed Traffic Recovery Time (RTRT), which aims to minimise the time required for traffic to return to normal, and Overall Recovery Count (ORC), which seeks to improve recovery stability at the network level. These objectives, when embedded into a multi-objective optimisation framework, provide a pathway to traffic light plans that explicitly pursue rapid and steady post-incident recovery, rather than achieving it as an incidental by-product of reducing delays.

In this work, we incorporate the proposed RTRT and ORC objectives into an NSGA-II optimisation framework. In addition, we use the Average Speed Variance (ASV) to reflect safety concerns during recovery. We test the framework using both synthetic and real-world networks. The results show that RTRT consistently leads to faster traffic recovery compared with conventional congestion-based objectives, while ORC demonstrates superior stability and comprehensiveness of the recovery process. These findings confirm the effectiveness of our recovery-oriented optimisation approach for post-incident TLC.

The main contributions of this paper are summarised below:

- We propose two recovery-oriented objective functions for post-incident TLC: RTRT and ORC. This is enabled by a clear formulation of network-level recovery, allowing optimisation to target recovery performance directly rather than relying on indirect congestion-based metrics.
- We develop an NSGA-II-based optimisation method that incorporates the two proposed objectives, evaluate it on synthetic and real-world road networks through a traffic simulation software, and compare it with conventional approaches. Our experiments demonstrate that recovery-oriented objectives outperform conventional congestion-based ones for post-incident traffic management.

This work was supported by the Australian Research Council (Grant No. LP180100114). *Corresponding Author: A. K. Qin

II. METHODOLOGY

A. Problem Formulation

Given an intersection with n one-way roads, denoted as $r_1, r_2, \ldots, r_n$, a traffic light regulates vehicle movement at this intersection and operates in m phases, denoted as $P_1, P_2, \ldots, P_m$, with each representing a set of traffic rules. A phase P_k specifies which vehicle movements are permitted by defining binary variables $p_{i,j}$ for all pairs $i, j \in [1, n]$ with $i \neq j$. If $p_{i,j} = 1$, vehicles from road r_i are allowed to move to approach r_j; otherwise, that movement is prohibited in that phase, denoted as $p_{i,j} = 0$. Each phase is active for a duration x_{P_k}. Given a predefined sequence of traffic light phases, the traffic controller sequentially activates each phase for its designated duration and repeats the cycle from the first phase upon completion of the last.

Consider an area comprising multiple intersections, each controlled by a traffic light with predefined phases. When an incident occurs, traffic efficiency may be adversely affected. Our goal is to optimise the phase durations at all intersections to minimise the *traffic recovery time* (TRT) while ensuring safety, as formally defined below.

Traffic Recovery Time. To formulate the TRT, we first define the *weighted average speed* (WAS) for each road. Each road r_i consists of L_i lanes, with a traffic sensor on each lane that records both vehicle count and corresponding speeds. For a given time slot t, let $q_l(t)$ denote the number of vehicles detected on lane l, and $s_l(t)$ denote their average speed. Following [16], the WAS of road r_i at time slot t is defined as:

$$v_i(t) = \begin{cases} \frac{\sum_{l \in L_i} s_l(t) \cdot q_l(t)}{\sum_{l \in L_i} q_l(t)}, & \text{if } \sum_{l \in L_i} q_l(t) > 0 \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

To enable a fair comparison of WAS values across time slots, we employ the Z-score [17] to standardise the value. Specifically, we define an indicator $I_i(t_1, t_2)$ to represent whether the WAS at t_2 is statistically equivalent to or greater than the WAS at t_1 for road r_i:

$$I_i(t_1, t_2) = \begin{cases} 1, & \frac{v_i(t_2) - v_i(t_1)}{\sigma_i(t_1)} > c_1 \\ 0, & \text{otherwise,} \end{cases} \tag{2}$$

where $\sigma_i(t_1)$ is the standard deviation of vehicle speeds at time t_1, and c_1 is a positive threshold constant.

Let t_{pre} denote the time slot immediately before the incident occurs. Let t_{post} denote the first time slot that satisfies

$$\sum_{i=1}^{n} \sum_{k=0}^{c_2} I_i(t_{\text{pre}}, t_{\text{post}} - k) = c_2 n \ , \tag{3}$$

where c_2 is a positive integer. TRT is then measured by $t_{\text{post}} - t_{\text{pre}}$. Intuitively, t_{post} represents the earliest time when traffic conditions on all roads have recovered to a level statistically comparable to their pre-incident condition, as indicated by the WAS, for at least $c_2 + 1$ consecutive time slots.

Average Speed Variance. Large speed differences between vehicles can lead to more hazardous interactions. The ASV provides a concise measure of the heterogeneity of vehicle speeds within the system. For safety considerations, following [18], we introduce the ASV as an indicator, which is defined as follows:

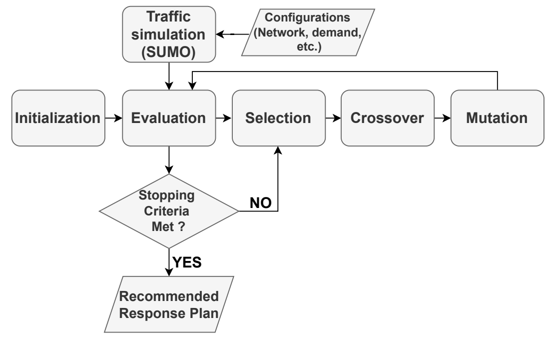


Fig. 1: Proposed post-incident decision-support framework for TLC.

$$\text{ASV}(t_1, t_2) = \frac{1}{V} \sum_{i=1}^{V} \sigma_i \ , \tag{4}$$

where V is the number of vehicles in the area between the time slots t_1 and t_2 (inclusive), and σ_i is the standard deviation of the speed for vehicle i during this interval. To ensure safety, our goal is to minimise $\text{ASV}(t_{\text{pre}} + 1, t_{\text{post}})$.

B. The Framework of Our Method

We now introduce our framework using a genetic algorithm, starting with a description of how to represent the problem using genetic sequences.

Solution Representation for Genetic Algorithm. A solution is represented as a chromosome that specifies the activation times of each phase at every intersection. Specifically, $x_{P_1}, \ldots, x_{P_m}$ denotes the activation times for one intersection, and concatenating these sequences across all intersections forms the complete solution representation (i.e., the TLC plan) for the entire network.

Framework Overview. To support traffic operators in selecting an effective post-incident TLC plan, we develop a decision-support framework that automatically explores and evaluates alternative response plans. This framework aims to balance two conflicting objectives: traffic recovery performance and safety. As shown in Fig. 1, the framework integrates a multi-objective optimisation algorithm and microscopic traffic simulation. In the optimisation loop, a set of candidate TLC plans is iteratively optimised using the NSGA-II algorithm [19]. Each candidate's plan is simulated to evaluate its performance through the objective functions. When the stopping criterion is reached, the algorithm outputs a Pareto front, meaning a set of solutions where no objective can be improved without worsening at least one other. This enables operators to consider response plans across multiple criteria and select one that best aligns with their operational priorities.

Optimisation Process of NSGA-II Algorithm. The optimisation module in the proposed framework is implemented based

Algorithm 1 NSGA-II with Local Search

Require: Population size N, maximum generations $G_{\max}$, local search start generation G_{ls}, crossover probabilities (Pr_{c1}, Pr_{c2}), mutation probabilities (Pr_{m1}, Pr_{m2})
Ensure: Final Pareto-optimal front
Initialise a population O_0 of N individuals randomly
Evaluate each individual in O_0 to obtain objective values (f_1, f_2)
Perform non-dominated sorting on O_0 and calculate crowding distance for all individuals in O_0
$t \leftarrow 0$
while $t < G_{\max}$ **do**
  Select N parents from O_t using the TournamentDCD method (selection based on dominance and crowding distance)
  Apply crossover with probabilities (Pr_{c1}, Pr_{c2}) and mutation with probabilities (Pr_{m1}, Pr_{m2}) to generate offspring Q_t of size N
  Evaluate all individuals in Q_t
  if $t \geq G_{ls}$ **then**
    Apply Local Search to Q_t ▷ See algorithm 2
  end if
  $R_t \leftarrow O_t \cup Q_t$
  Perform non-dominated sorting on R_t and calculate crowding distance for all individuals in R_t
  Select the best N individuals from R_t to form O_{t+1}
  $t \leftarrow t + 1$
end while
return The first Pareto front obtained from the final population O_t

on the NSGA-II algorithm. The overall process of NSGA-II is summarised in Algorithm 1.

Our Crossover Operator and Mutation Operator Implementation. The crossover operator exchanges genetic information between two parent individuals to produce diverse offspring. In this work, crossover is first applied at the individual level: for each selected parent pair, a predefined probability Pr_{c1} determines whether crossover occurs. The crossover operates at the segment level (each segment represents one intersection), where segments are swapped between parents with probability Pr_{c2} if triggered.

The mutation operator maintains diversity and encourages exploration of the solution space. Similarly, each chromosome undergoes mutation with probability Pr_{m1}, and, if triggered, each segment is mutated with probability Pr_{m2}. Mutated segments have all values replaced by random integers within the predefined range $[r_{\min}, r_{\max}]$, while others remain unchanged. This two-level design introduces randomness while preserving solution feasibility.

Our Local Search Operator Implementation. To further improve the offspring quality and explore the neighbourhood of Pareto-optimal individuals, a local search procedure is applied to the offspring population in each generation after certain loops. The detailed process is summarised in Algorithm 2.

C. Objective functions

Objective Functions 1: Traffic Recovery Performance. To better optimise traffic recovery performance, we further propose two objective functions, namely *relaxed traffic recovery time* and *overall recovery count*. Both functions are designed based on TRT but emphasise different aspects of recovery.

Algorithm 2 Local Search for NSGA-II

Require: Offspring population Q, range r_m, selection proportion q
Ensure: Updated offspring population Q after local search
$F \leftarrow$ Pareto front of Q
for each individual $ind \in F$ **do**
  $mutated\_individuals \leftarrow [\,]$
  Randomly select q fraction of gene cells in ind: $cells$
  for each $cell \in cells$ **do**
    $mut\_ind \leftarrow$ copy of ind
    $mut\_ind.genes[cell] \leftarrow \text{Clip}\big(mut\_ind.genes[cell] + \text{Uniform}(-r_m, r_m),\ r_{\min}, r_{\max}\big)$
    Append mut_ind to $mutated\_individuals$
  end for
  Evaluate all $mutated\_individuals$
  for each $mut\_ind \in mutated\_individuals$ **do**
    if mut_ind dominates ind **then**
      Replace ind with mut_ind
    end if
  end for
end for
return Updated Q

Relaxed Traffic Recovery Time. RTRT generalises TRT by: (i) relaxing the requirement that all roads must return to pre-incident conditions and instead only requiring a fraction αn of roads to recover, and (ii) providing a fallback measure based on the average speed difference when full recovery is not achieved within a given time.

Specifically, RTRT consists of two patterns. The first one is a relaxed version of the TRT where we apply a constant parameter $\alpha \in (0, 1]$ to Eq. (3), so that there exists a subset $S \subseteq \{1, 2, 3, \cdots, n\}$ and $|S| = \alpha n$ that satisfies:

$$\sum_{i \in S} \sum_{k=0}^{c_2} I_i(t_{\text{pre}}, t_{\text{post}} - k) = c_2 \alpha n \,. \tag{5}$$

Intuitively, we only require at least αn roads that have recovered to the pre-incident traffic level, and we denote the time as T_{rtr}. The reason behind this relaxation is that requiring 100% of roads to recover would often include the time taken to remove the incident vehicles, which is a factor that TLC cannot influence. By allowing a small proportion of roads to remain below the normal level, the measure better reflects the effectiveness of the TLC strategy in recovering traffic over the majority of the area, and hence leading to faster recovery overall.

However, it is possible that t_{post} cannot be found. In such cases, the second component of the objective function is introduced to capture the difference in WAS between the pre-incident state and time slot t.

Specifically, we use the area-level mean road speed difference as defined below:

$$\Delta \bar{v}(t_{\text{pre}}, t) = \frac{1}{n} \sum_{i=1}^{n} \begin{cases} 0 & \text{if } v_i(t_{\text{pre}}) - v_i(t) < 0 \\ v_i(t_{\text{pre}}) - v_i(t) & otherwise \end{cases} \tag{6}$$

This component accounts for the worst-case scenarios in which all TLC plans fail to restore normal traffic conditions in the area before the specified time slot t. In such cases, the algorithm selects the plan that minimises the speed difference.

The overall objective function can then be represented as:

$$f_1(t) = \begin{cases} w_1(T_{\text{rtr}} - c_3), & \text{if } t_{\text{post}} \text{ exists}, \\ \Delta\bar{v}(t_{\text{pre}}, t), & \text{otherwise}, \end{cases} \quad (7)$$

where w_1 are user-defined weight for visualisation scaling, and c_3 is a constant that ensures $f_1 < 0$ when t_{post} exists to flavour finding the (relaxed) TRT.

Overall Recovery Count. While RTRT rapidly restores traffic in the affected area to pre-incident levels, its traffic light plan is tailored to the incident scenario and may cause congestion if applied continuously. ORC is therefore introduced to guide the network toward a stable and comprehensive recovery within a given time window, still using the road recovery definition from Eq. (2). ORC represents the number of roads that have recovered ($I_i(t_{\text{pre}}, t) = 1$ as defined in Eq. (2)) at each time slot t for $\forall$ t between two time slots t_1 and t_2 (inclusive), which is defined as:

$$U(t_1, t_2) = \sum_{t=t_1}^{t_2} \sum_{i=1}^{n} I_i(t_{\text{pre}}, t). \quad (8)$$

By optimising

$$f_1 = U(t_{\text{pre}} + 1, t), \quad (9)$$

the objective function not only aims to accelerate the speed of recovery but also maintains a larger proportion of recovered roads over the entire recovery period. This formulation places more emphasis on the spatial extent and consistency of recovery across the network. Since NSGA-II is formulated as a minimisation problem, this objective is equivalently transformed by minimising its complement, i.e., the number of unrecovered road instances.

Both RTRT and ORC can effectively and efficiently guide the algorithms toward promising recovery plans. RTRT places greater emphasis on recovery time, whereas ORC prioritises the overall recovery rate, as will be demonstrated in the Experimental Section.

Objective Function 2: Average Speed Variance. For the second objective function, we use the previously defined $ASV(t_1, t_2)$ (Eq. (4)). The goal is to minimise $ASV(t_{\text{pre}}+1, t)$ to reduce abrupt speed variance and improve driving comfort and safety across the network.

III. Experiments

A. Scenario setup

We conduct traffic simulations in Eclipse SUMO (Simulation of Urban MObility) on both a synthetic road network and a real road network.

1) Scenario 1: synthetic network

The synthetic network consists of four standard four-leg intersections arranged in a grid layout, forming a 2×2 structure, as shown in Fig. 2a. Each road has three lanes and an average segment length of approximately 280 meters.

Each traffic light has 4 phases within a fixed 120 s cycle with $x_{P_1} + x_{P_2} = x_{P_3} + x_{P_4}$, and each phase is followed by a 3-second yellow light. For the phrase rules, we only show the rules with $p_{i,j} = 1$, and those not shown are with $p_{i,j} = 0$. Based on the road numbering shown in Figure 2a, the phases of Intersection 2 are summarised below:

- P_1: $p_{1,2} = 1$, $p_{1,4} = 1$, $p_{5,6} = 1$, $p_{5,8} = 1$;
- P_2: $p_{1,2} = 1$, $p_{1,6} = 1$, $p_{5,6} = 1$, $p_{5,2} = 1$;
- P_3: $p_{3,4} = 1$, $p_{3,6} = 1$, $p_{7,8} = 1$, $p_{7,2} = 1$;
- P_4: $p_{3,4} = 1$, $p_{3,8} = 1$, $p_{7,8} = 1$, $p_{7,4} = 1$.

The other three intersections have the same phrases, hence are omitted here.

2) Scenario 2: real-world network (Pasubio, Bologna)

For real-world validation, we further employ the *Pasubio* network from the SUMO scenario examples [20], located in Bologna, Italy, as shown in Fig. 2b. The network comprises 24 intersections in total, 12 of which are signalised. Among these, four key intersections are selected for TLC plan optimisation. The same cycle time is applied, with $x_{P_1} + x_{P_2} = x_{P_3} + x_{P_4}$, and each phase is followed by a 3-second yellow light. With the road numbering illustrated in Figure 2a, the phases of Intersection 2 are summarised as follows:

- P_1: $p_{1,2} = 1$, $p_{1,4} = 1$, $p_{1,6} = 1$, $p_{3,4} = 1$, $p_{5,6} = 1$, $p_{5,8} = 1$, $p_{5,2} = 1$, $p_{7,8} = 1$;
- P_2: $p_{1,2} = 1$, $p_{1,4} = 1$, $p_{1,6} = 1$, $p_{3,4} = 1$, $p_{5,6} = 1$, $p_{7,8} = 1$;
- P_3: $p_{1,2} = 1$, $p_{3,4} = 1$, $p_{3,6} = 1$, $p_{3,8} = 1$, $p_{5,6} = 1$, $p_{7,8} = 1$, $p_{7,2} = 1$;
- P_4: $p_{1,2} = 1$, $p_{3,4} = 1$, $p_{3,8} = 1$, $p_{5,6} = 1$, $p_{7,8} = 1$, $p_{7,4} = 1$.

The other three intersections have similar phrases, which are omitted here.

3) Synthetic incident

In both case studies, the incident occurs at 360 seconds and lasts until 1200 seconds. The locations of the incidents in the synthetic and real-world networks are illustrated in Figure 2a and Figure 2b (both on the rightmost lane), respectively. A 150-meter safety buffer zone is introduced around the incident vehicle, with different control strategies depending on the lane and position of the approaching vehicles:

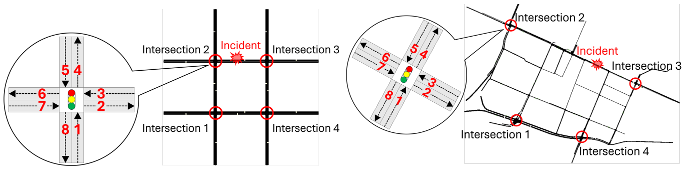
- Vehicles on the incident lane must merge to another lane at least 150 meters before the incident vehicle, with a 5-second merge delay and deceleration to 3 m/s. If merging fails, they queue at least 100 meters upstream and decelerate to a full stop without delay.
- Vehicles on other lanes decelerate more safely to avoid emergency braking: on the adjacent lane, a 3-second merge delay and slowdown to 8 m/s are applied; on lanes one step further, a 1-second delay and slowdown to 10 m/s are used.
- Vehicles that have passed the incident location gradually restored to normal speed over 5-seconds.

B. Experimental setup and results

1) Parameter set up for NSGA-II

The NSGA-II algorithm is configured with the parameters as summarised in Table I.

The parameters related to the proposed objective functions are listed as follows. $c_1 = 2$ (Eq. 2), $c_2 = 4$, $\alpha = 0.9$ (Eq. 5),



(a) The synthetic network. (b) The real-world network.

Fig. 2: Networks used in experiments.

TABLE I: Parameter settings for the NSGA-II algorithm.

Parameter	Symbol / Value
Population size	$N = 40$
Number of generations	$G_{\max} = 150$
First-level crossover probability	$Pr_{c1} = 0.8$
First-level mutation probability	$Pr_{m1} = 0.125$
Second-level crossover probability	$Pr_{c2} = 0.5$
Second-level mutation probability	$Pr_{m2} = 0.5$
Mutation minimum range	$r_{\min} = 10$
Mutation maximum range	$r_{\max} = 44$
Local search start generation	$G_{ls} = 50$
Local mutation range	$r_m = 3$
Local search mutation rate	$Pr_s = 0.5$

$c_3 = 3600$ and $w_1 = \frac{1}{500}$ (Eq. 7). t is set to the last time slot of the simulation ($t_{\max}$) in Eq. (7) when t_{post} is not found, as well as Eq. (9).

2) Effectiveness of proposed objectives

We compare two objective functions proposed in this study: RTRT and ORC. RTRT focuses on fast recovery after an incident, while ORC aims to improve the overall traffic condition during the recovery process. We also compare both objectives with a recent study on incident management [21], which uses Average Travel Time (ATT) as its objective, which will be detailed below. For all objectives, the second objective in the multi-objective optimisation is ASV. We conduct these comparisons on the two networks as described above.

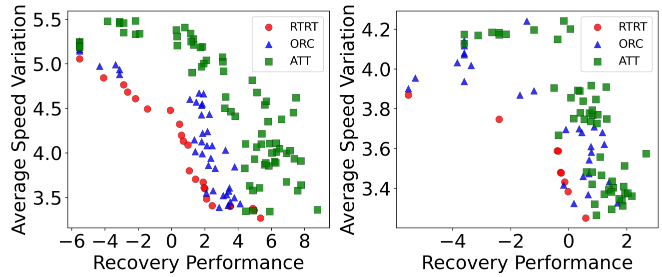
Average Travel Time. AAT is a conventional performance metric widely used in traffic optimisation studies. It represents the mean duration required by vehicles to complete their routes within the network. The ATT is calculated as the arithmetic mean of all such durations across all vehicles, as follows:

$$\text{ATT} = \frac{1}{V}\sum_{i=1}^{V} T_i \tag{10}$$

where V is the total number of vehicles, and T_i is the time of the i^{th} vehicle complete the route.

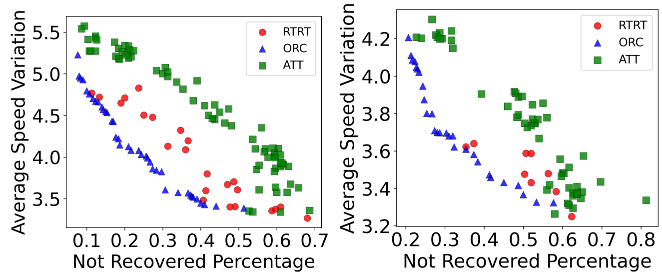
Comparison of Optimisation Objectives on Recovery Time. After optimisation, we collect the solutions from all three objective settings—(*RTRT*, *ASV*), (*ORC*, *ASV*), and (*ATT*, *ASV*)—across five independent runs. For each setting, we merge all solutions and compute a new Pareto front to represent the best trade-offs. To ensure a consistent evaluation, we evaluated the final Pareto front solutions derived from the (*ATT*, *ASV*) and (*ORC*, *ASV*) on the RTRT and plotted the results in Fig. 3 with (*RTRT*, *ASV*).

As shown in Fig. 3a, in the synthetic network, the solutions obtained using the proposed RTRT objective function show a clear advantage over the other two objectives. Given that $c_3 = 3600$ and $w_1 = \frac{1}{500}$, the value of -1 on the x-axis indicates a recovery time of 3100 seconds (Eq. (7)). A single unit on the x-axis corresponds to an average speed difference of 1 m/s relative to the pre-incident average speed.



(a) The synthetic network. (b) The real-world network.

Fig. 3: Comparison of solutions using different objective functions evaluated by RTRT.



(a) The synthetic network. (b) The real-world network.

Fig. 4: Comparison of solutions using different objective functions evaluated by NRP.

In the RTRT setting, solutions almost always dominate those from ORC and ATT. In the fully recovered region (values < 0), RTRT also produces a more balanced and continuous spread of solutions, indicating more stable trade-offs. ORC performs worse than RTRT but still outperforms ATT. In contrast, ATT yields relatively weaker results, as it does not explicitly target incident recovery and only improves recovery indirectly via travel time reduction. Notably, ATT and ORC outputs more unrecovered plans (with x-axis values > 0) than RTRT, but in these cases, RTRT can still output safer and smaller speed difference plans.

For the real network shown in Fig. 3b, similar trends are observed across RTRT, ORC, and ATT. However, only RP and TS achieve the best recovery performance values, as the real network is more complex and inherently more challenging to recover.

Comparison of Optimisation Objectives on Overall Recovered Percentage. To evaluate the overall recovery performance for each individual sensor, we introduce the *not recovered percentage*, defined as follows:

$$\text{Not Recovered Percentage} = 1 - \frac{U(t_{\text{pre}}+1, t_{\max})}{n(t_{\max} - t_{\text{pre}} - 1)}.$$

Intuitively, it reflects the proportion of unrecovered sensors accumulated over all time slots. As shown in Fig. 4a, solutions optimised with the ORC achieve the best performance, almost dominating those from RTRT and ATT. RTRT-based solutions tend to flavour plans with higher unrecovered percentages and lower ASV. This is because RTRT focuses on identifying faster recovery plans and does not account for the level of congestion before recovery. ATT performs relatively worse among the three objectives, further demonstrating the robustness of our proposed objective functions for post-incident management.

A similar pattern is observed in the real-world network as shown in Fig. 4b. ORC again delivers the best overall performance and dominates most solutions. Although the performance gap between RTRT and ORC narrows compared to the synthetic network, both remain clearly superior to ATT. This shows that ORC provides the most stable optimisation across networks, while RTRT continues to offer significant advantages for recovery time.

IV. Conclusions and Future Work

In this work, we investigated post-incident traffic management via TLC and formulated traffic recovery as restoring normal traffic conditions within a predefined road area. To achieve fast and stable recovery, we propose two recovery-oriented objective functions: Relaxed Traffic Recovery Time and Overall Recovery Count, which are used in conjunction with a safety-related objective (speed variance) in NSGA-II for bi-objective optimisation, where the objective function evaluation is based on a traffic simulation software SUMO. Experimental results on both synthetic and real-world scenarios show that, compared to traditional congestion-based objectives, the proposed recovery-oriented ones lead to the improved TLC plan for post-incident recovery. In particular, RTRT achieves faster recovery, while ORC enhances stability and overall traffic conditions during recovery.

Effective traffic incident management typically requires rapid decision-making. Therefore, TLC strategies are expected to be generated and deployed promptly after the incident is detected to gain practical benefits. However, the real-time capability of this work remains limited due to the high computational cost of traffic simulations. Our future work plan to investigate hardware parallelisation and algorithmic surrogate modelling to address the limitation. In addition, while speed variance is relevant, it may not closely characterise traffic safety. Our future work will explore safety metrics more suitable as optimisation objectives.

Acknowledgement

We used Grammarly and ChatGPT to improve language quality during the preparation of this paper. All content was reviewed and edited by the authors, who take full responsibility for the final publication.

References

[1] S. A. Arhin, A. Gatiba, M. Anderson, M. Ribbisso, and B. Manandhar, "Improving Mobility on Segments with Closely Spaced Intersections in Urban Areas," *Open Journal of Civil Engineering*, vol. 09, no. 01, pp. 46–56, 2019.

[2] S. S. S. M. Qadri, M. A. Gökçe, and E. Öner, "State-of-art review of traffic signal control methods: challenges and opportunities," *European Transport Research Review*, vol. 12, no. 1, pp. 1–23, 2020.

[3] C. Gheorghe and A. Soica, "Revolutionizing Urban Mobility: A Systematic Review of AI, IoT, and Predictive Analytics in Adaptive Traffic Control Systems for Road Networks," *Electronics (Switzerland)*, vol. 14, no. 4, pp. 1–25, 2025.

[4] Z. Li, M. Shahidehpour, S. Bahramirad, and A. Khodaei, "Optimizing Traffic Signal Settings in Smart Cities," *IEEE Transactions on Smart Grid*, vol. 8, no. 5, pp. 2382–2393, 2017.

[5] T. Mao, A.-S. Mihaita, and C. Cai, "Traffic signal control optimization under severe incident conditions using Genetic Algorithm," no. October, pp. 21–25, 2019. [Online]. Available: http://arxiv.org/abs/1906.05356

[6] T. Mao, A. S. Mihaita, F. Chen, and H. L. Vu, "Boosted Genetic Algorithm Using Machine Learning for Traffic Control Optimization," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 7112–7141, may 2022.

[7] N. R. Sabar, L. M. Kieu, E. Chung, T. Tsubota, and P. E. Maciel de Almeida, "A memetic algorithm for real world multi-intersection traffic signal optimisation problems," *Engineering Applications of Artificial Intelligence*, vol. 63, pp. 45–53, 2017.

[8] M. K. Tan, H. S. E. Chuo, R. K. Y. Chin, K. B. Yeo, and K. T. K. Teo, "Optimization of urban traffic network signalization using genetic algorithm," *ICOS 2016 - 2016 IEEE Conference on Open Systems*, pp. 87–92, 2017.

[9] F. Shen and X. Yang, "Optimization algorithm of urban road traffic signal plan based on NSGAII," *Proceedings - International Conference on Intelligent Computation Technology and Automation, ICICTA 2008*, vol. 2, pp. 398–401, 2008.

[10] J. Ferrer, J. García-Nieto, E. Alba, and F. Chicano, "Intelligent testing of traffic light programs: Validation in smart mobility scenarios," *Mathematical Problems in Engineering*, vol. 2016, pp. 1–19, 2016.

[11] W. Kou, X. Chen, L. Yu, and H. Gong, "Multiobjective optimization model of intersection signal timing considering emissions based on field data: A case study of Beijing," *Journal of the Air and Waste Management Association*, vol. 68, no. 8, pp. 836–848, 2018.

[12] Z. Lin, K. Gao, N. Wu, and P. Nagaratnam Suganthan, "Problem-Specific Knowledge Based Multi-Objective Meta-Heuristics Combined Q-Learning for Scheduling Urban Traffic Lights with Carbon Emissions," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 10, pp. 15 053–15 064, 2024.

[13] X. Li and J. Q. Sun, "Multi-objective optimal predictive control of signals in urban traffic network," *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, vol. 23, no. 4, pp. 370–388, 2019.

[14] A. Cacco and G. Iacca, "Simulation-Driven Multi-objective Evolution for Traffic Light Optimization," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12104 LNCS, pp. 100–116, 2020.

[15] S. S. Leal and P. E. M. de Almeida, "Traffic light optimization using non-dominated sorting genetic algorithm (NSGA2)," *Scientific Reports*, vol. 13, no. 1, p. null, 2023.

[16] E. Bulteau, R. Leblanc, S. Blandin, and A. Bayen, "Traffic flow estimation using higher-order speed statistics," in *Transportation Research Board 92nd Annual Meeting*, 2013.

[17] C. Andrade, "Z scores, standard scores, and composite test scores explained," *Indian journal of psychological medicine*, vol. 43, no. 6, pp. 555–557, 2021.

[18] M. Quddus, "Exploring the relationship between average speed, speed variation, and accident rates using spatial statistical models and gis," *Journal of Transportation Safety & Security*, vol. 5, no. 1, pp. 27–45, 2013.

[19] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182–197, 2002.

[20] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner, "Microscopic traffic simulation using sumo," in *The 21st IEEE International Conference on Intelligent Transportation Systems*. IEEE, 2018. [Online]. Available: https://elib.dlr.de/124092/

[21] T. Mao, A.-S. Mihăiţă, F. Chen, and H. L. Vu, "Boosted genetic algorithm using machine learning for traffic control optimization," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 7112–7141, 2021.