Design and Simulation-Based Testing of 5G-Connected Systems for Traffic Light Guidance

Alberto Gabriele, Francesca Mancini, Michael Khayyat, Stefano Arrigoni

Abstract

The establishment of fast and reliable communication technologies, such as 5G, is enabling the evolution of a new generation of connected ADAS. This work is part of Vodafone 5G Project and aims to develop curve safety and traffic light advisory systems utilizing an active map. This map is populated by real-time data coming from connected vehicles and infrastructures. An ADAS is developed and tested to improve road safety and reduce urban pollution: Multiple Traffic Light Advisor (MTLA) systems. The MTLA aims to improve intersection viability and reduce energy consumption by advising the vehicle on taking a green wave. The MTLA is further improved through a non-linear MPC approach. The system is tested in a virtual environment and results show good performances for both with a high potential for future developments.

Keywords: ADAS, Traffic Light Advisor, Connected Vehicles, 5G, V2X, MPC

1. Introduction

The World Health Organization estimates that 1.2 million people are killed and as many as 50 million are injured in road accidents every year [1]. Since crash-related injuries are the 8th leading cause of death [2], measures to reduce traffic accidents are needed; these relate to changes in human behavior and improvements in vehicle safety and road infrastructure [3]. Active safety
components try to prevent critical situations when possible or mitigate the effects when these cannot be avoided. Examples are systems like electronic stability control (ESC), anti-lock braking system (ABS) or more generally most Advanced Driver Assistance Systems (ADAS).

Along with road safety, environmental pollution is another topic of great interest to automotive research. One of the biggest cause of air pollution is road transportation, and for this reason, travel delays, fuel wastage and emissions are few of the challenges that mobility faces. To achieve the goals of sustainable transportation programs, several solutions have been implemented by vehicle manufacturers such as use of lighter and stronger materials, use of alternative fuels and increased efficiency of power train components. On the other hand, ADAS present promising methods to improve fuel savings as well as emissions [4, 5].

A system intended both for safety and sustainable transportation is GLOSA (Green Light Optimal Speed Advisory). This is representative of ECO-Driving strategies used to reduce emissions and fuel consumption by decreasing the number of stops at intersections. It computes a recommended speed profile and a warning (or control action) on how to modify the actual vehicle velocity is given to the driver (or to the vehicle). GLOSA systems can be divided in two main categories, according to the number of traffic lights analyzed by the system to give a real-time recommended speed: Single segment\(^1\) GLOSA (S-GLOSA) or Multiple segment GLOSA (M-GLOSA). S-GLOSA systems are based on the analysis of only the first traffic light ahead the vehicle while M-GLOSA systems take into account several traffic lights along the vehicle route.

Concerning S-GLOSA systems, different approaches are present in literature to compute the recommended velocity profile. In [6], Barth et al. interpret the eco-driving strategy as a minimization of the total tractive power demand and idling time. For instance, to avoid idling, the vehicle should get to the traffic light during its green phase. Thus, considering \(t_r\) and \(t_g\) as the time until phase

---

\(^1\)A segment is a portion of road delimited by two traffic lights
changes respectively to red and green, an admissible time interval to get the green is defined as in 1:

$$t \in \begin{cases} 
[0, t_r) \cup [t_g, t_{r,1}) & \text{if phase = green} \\
[t_g, t_r) & \text{if phase = red} 
\end{cases} \quad (1)$$

It is to be noticed that $t_r$ is the time to the first green to red shift while $t_{r,1}$ is the time to the second green to red sif.

Corresponding admissible velocities are defined considering a constant velocity profile ($[V_{lo}, V_{ho}]$). Consistency with the road limits is checked as in 2 and the minimum and maximum possible velocities (respectively $V_l$ and $V_h$) are defined. Then, the maximum one is suggested to the driver.

$$V_{possible} = [V_{lo}, V_{ho}] \cap [0, V_{limit}] = [V_l, V_h] \quad (2)$$

Admissible velocities to reach the traffic light during one of its green phases are defined considering a constant velocity profile. The highest one is then suggested as reference to the driver.

In [7], Cai and Ning propose a speed guidance system according to two different algorithms: “Best Feasibility” and “Best Efficiency”. “Best Feasibility” focuses on reducing driver annoyance from the suggested speed by having the smallest difference between actual and guidance speed. “Best Efficiency” algorithm aims at passing the traffic light as fast as possible. It is concluded that the optimized strategy is given by the combination of both algorithms: “Best Feasibility” should be used during braking phases while “Best Efficiency” in high speed and acceleration situations.

Katsaros et al. [8] study the impacts of GLOSA on fuel and traffic efficiency by analyzing average fuel consumption and average stop time behind a traffic light. In their system, target velocity is computed as following: the time needed for the vehicle to reach the traffic light is computed considering a uniformly accelerated motion profile, then if the vehicle reaches the traffic light when it is green, maximum road speed is suggested to the driver, otherwise the target speed is computed considering a uniformly accelerated motion so to reach the traffic light during the next green phase. Simulations show that, in a high
traffic density scenario, the higher the number of equipped vehicles the higher the benefits. On the other hand traffic efficiency increases if traffic density decreases.

When dealing with M-GLOSA systems, two main design approaches can be distinguished: Model Predictive Control (MPC) and Genetic Algorithms (GAs). For the application under analysis, traffic light phase changes have to be considered as constraints. Unfortunately, phases dynamic variability makes the feasible solution space non-convex, resulting in a computationally expensive optimization problems which may not converge to global optimum [9]. This issue is solved by managing the problem on two levels: a lower level takes into account for phase variations and defines a first target velocity, then the latter is given as reference to a higher MPC based level, that computes the final desired velocity. The resulting solution may be sub-optimal, but is real-time implementable.

Asadi and Vahidi [9] propose a so called “Predictive Cruise Control (PCC)” that minimizes the use of brakes based on traffic signal information and enforcing at the same time several physical constraints. A set of logical rules calculates a reference velocity for timely arrival at green lights considering a constant velocity profile. The obtained profile is fed as reference to the MPC that tracks this target velocity. The controller uses a vehicle model that is based on the linearization of the longitudinal dynamics and takes into account for vehicle mass and position, aerodynamic drag, rolling resistance and road grade forces. The authors define the cost function as to minimize brake force and deviation from target speed. Constraints bound speed, engine/brake forces and safe distance between follower and lead vehicle. Tests results show 59% reduction of fuel consumption, 39% less CO$_2$ emission and reduced travel time when the PCC controller is adopted. Further simulations are performed by Asadi and Vahidi in [10] to prove the effective reduction of fuel consumption and travel time of the developed system.

In [11], Jones et al. present an MPC approach to control an electric vehicle approaching a road segment with multiple traffic lights. Energy-optimal and time saving trajectories are computed considering a lower level “Fast MPC” to
compute a first attempt trajectory and a “Main MPC” which computes a more
detailed trajectory, taking into account for the vehicle dynamic behavior and
acceleration/deceleration limits. A linear kinematic model that includes also
the longitudinal inertial dynamics is used. As in the following:

\[
\begin{align*}
  v &= \frac{d(p)}{dt} \\
  a &= \frac{d(v)}{dt} \\
  a &= \frac{K}{T_s+1} a_{des}
\end{align*}
\]  

(3)

The cost function minimizes vehicle desired acceleration (input of the system)
and deviation of vehicle trajectory with respect to the reference one. Constraints
are adopted to limit vehicle position, speed and desired acceleration.

As previously mentioned, a different approach to deal with GLOSA systems
is through GAs. Genetic algorithms iteratively generate a population of candi-
date solutions until a termination condition is met. Each candidate solution has
a set of properties (its chromosomes or genotype) which can be modified. The
fitness of each candidate solution in the population is then calculated through
a fitness function. The fittest individuals are stochastically selected from the
current population and each individual’s genome is modified (recombined and
possibly randomly mutated) to form a new generation, which is then used in the
next iteration of the algorithm. Commonly, algorithms terminate when either
a maximum number of generations has been produced or a satisfactory fitness
level has been reached.

In [12, 13], an advisory speed is proposed to the driver according to selected
preferences like minimisation of total traveling time or fuel consumption. Test-
ing shows that in free-flow conditions such multi-segment GLOSA gives much
better results when compared with single-segment approach. Nguyen et al. [14]
propose an improved GLOSA method called R-GLOSA, which also takes into

\footnote{It is noteworthy that minimizing the desired acceleration is an equivalent way of mini-
mizing energy consumption of the vehicle, since the torque required to the electric motor is
directly connected to acceleration.}
account traffic density on the road to compute the optimal speed. Density information is obtained through the vehicle communication with Road Side Units (RSU) distributed along the road. Both single and multiple R-GLOSA are developed and compared with single/multiple GLOSA and no-GLOSA vehicle; results shows that the developed approach is better than non-RSU ones in terms of travel time and waiting time and that $CO_2$ emissions are reduced according to vehicle density.

It is important to highlight that even though there are many GLOSA system in literature, none of them addresses the following issues: comfort, variability of friction coefficient and minimization of setup variation with respect to a standard vehicle in order to apply the system. In this paper, a novel ADAS which tackles the issues above, is developed and discussed.

The developed guidance system, named Traffic Light Advisor (TLA), warns the driver in time on how to modify the vehicle velocity to get one (Single Traffic Light Advisor) or more (Multiple Traffic Light Advisor) green traffic lights. To this end, TLA uses 5G technology as mean of communication for obtaining necessary information, such as traffic light phases, road geometry and friction coefficient and speed limits. Two different versions of this ADAS are presented, a non-optimal MTLA and an optimal MTLA. The first one aims to keep a hardware configuration as coherent as possible with the one available in standard commercial vehicles. The second adopts Model Predictive Control techniques in order to improve comfort, which is crucial for this sort of applications, at the expenses of computation power required.

2. Setup Description

2.1. System Architecture

V2V and I2V communication technologies are used to transfer information that could be utilized to improve ADAS. Traditional V2V technology consists of wireless data transmissions between vehicles. Commonly, I2V communications
Figure 1: Vodafone Intelligent Speed Adaptation & Control - data transmission logic scheme

are wireless, bidirectional, and similarly to V2V, using Dedicated Short-Range Communication (DSRC) frequencies to transfer data [15].

In this work, a new approach to these communication standard is used. Information is not transmitted directly between vehicles or between vehicle and infrastructure, but an intermediate and dynamic layer, that is here called Active Map. Data flow between vehicles, infrastructure and map is managed through Multi-access Edge Computing (MEC), which is a network architecture concept that enables cloud computing capabilities and an IT service environment at the edge of a network. Figure 1 demonstrates how the map is populated with information from the traffic light (timing and phases) and friction information. Road potential grip information is obtained through Smart Tyres; once measured, this is transmitted to the Active Map, along with the corresponding GPS location: in this way the map is populated with these data. Traffic light tim-
Table 1: Algorithm input parameter classification

| Parameter           | Variability | Source        | Type        |
|---------------------|-------------|---------------|-------------|
| V                   | Dynamic     | CAN BUS       | Input       |
| X                   | Dynamic     | GPS           | Input       |
| Y                   | Dynamic     | GPS           | Input       |
| Ψ                   | Dynamic     | GPS           | Input       |
| µ                   | Dynamic     | Active Map    | Input/Output|
| Road Geometry       | Static      | Active Map    | Input       |
| Road Limit Speed    | Static/Dynamic | Active Map  | Input       |
| Traffic Light Position | Static/Dynamic | Active Map  | Input       |
| Traffic Light Phases | Dynamic      | Active Map    | Input       |
| Traffic Light Timing | Dynamic      | Active Map    | Input       |

and phases, and the potential grip are part of the dynamic\(^3\) information stored in the map. Another piece of information of this class is maintenance works. Together with dynamic information, static one such as road geometry, road lanes, road limit speed and traffic lights position is present in the map too. This is labelled as static since it does not change in time. The TLA algorithms need further input quantities (other than the information retrieved from the Active Map). As seen in Figure 1, additional inputs are given by the vehicle CAN BUS, GPS and navigation system. Furthermore, the vehicle is equipped with a Computation and Display Unit, a tablet device which has the function to perform calculations required by the ADAS logic and display any warnings.

In Table 1 the input quantities needed for both TLA systems are reported. Speed, position and orientation (defined according to GPS reference system) of the vehicle, friction coefficient, limit speed and geometry of the road are obtained as previously described. It is noteworthy to mention that for these applications the road geometry is considered as the path to follow and is used only to relate the states \( (X, Y, \Psi) \) to the abscissa \( s \) of the path, while information about curves is disregarded.

\(^3\)refers to their variability in time
2.2. General Algorithm - Traffic Light Advisor Overview

The MTLA considers with the following scenarios:

- Stop&Go: the vehicle is approaching the traffic light when red, so it stops and starts again when the phase changes into green. The algorithm suggests a prior deceleration to avoid the stop.

- Last second braking: in this scenario the vehicle is approaching the traffic light when green, but the phase will change when the vehicle is close to the semaphore and an acceleration is unfeasible, hence a hard braking is necessary. Here the warning system informs the driver to reduce speed in advance so that a hard braking is avoided.

- Unnecessary stop: here the vehicle is approaching the traffic light when green, but the phase will change shortly. The STLA tells the driver to accelerate (respecting road speed limit and guaranteeing vehicle safety) so that the vehicle can pass when the light is still green.

Moreover, the MTLA is able to calculate a warning which enables the vehicle to take a green wave.

Three possible levels of warning can be issued to the driver:

1. Green Warning (no sound): it encourages the driver to increase the speed in order to take one (STLA) or more (MTLA) greens without stopping.
2. Red Warning (no sound): it suggest to the driver reducing speed.
3. Red + Sound Warning: it warns the driver that he/she is passing the traffic light when red, hence the request is to brake hard. Obviously this warning activates only when the driver has ignored the previous one.

It is noteworthy that if no modification of the speed is requested to the driver, no warning is issued.

In Figure 2, a scheme of the TLA system is presented. Inputs are reported on the left side and are classified as static, dynamic or both static and dynamic. First step is the abscissa computation (Localization block), which is then used in
the Activation Check block to trigger the computation of reference acceleration and warning (Reference Generation & Warning Definition block).

2.3. **Hypotheses**

The assumptions that are used in the development of the algorithms are the following:

1. A point mass model is adopted to model the vehicle, as only the longitudinal behavior is relevant. The speed $v$ is always tangent to the path and $v = \dot{s}$, and the acceleration is $a = \ddot{s}$. Note that the reference point for the point mass model is defined with respect to the front of the vehicle; this choice is motivated by the fact that when the vehicle needs to stop, the front should be before the stop line of the traffic light. It may be pointed out that in case of passing at the end of the green phase, it is the back that should be considered, but there is still the yellow phase that enables the vehicle to drive through the intersection safely.

2. The abscissa of the traffic light is the value of the abscissa of the stop line.

3. No yellow phase is considered, it is part of the red one.

4. The warning system needs to know when to trigger calculations, meaning that when the vehicle is at a certain distance from the TL the algorithm...
is run. This distance has been called horizon, because the vehicle looks ahead and checks if a TL is present within this. Calling $s_{TL}$ the abscissa of the first traffic light ahead and $l$ the distance between this and the vehicle ($l = s_{TL} - s$), the algorithm is activated if the condition in Equation 4 is verified:

$$l < h$$  \hspace{1cm} (4)

where $h$ is the horizon. One way of defining $h$ can be to consider a uniformly accelerated motion. The time needed to bring the actual velocity ($v$) to zero (to stop the vehicle) is in Equation 5:

$$t_{\text{comfort}} = \frac{v}{d_{\text{comfort}}}$$  \hspace{1cm} (5)

where $d_{\text{comfort}}$ is a comfort deceleration value set to $1 \text{m/s}^2$. According to [16], safe drivers tend to brake with a deceleration of approximately $3 \text{m/s}^2$, but, in order to be able to anticipate the braking, the deceleration value has been set to a $1 \text{m/s}^2$. So, the corresponding comfort distance is computed as in Equation 6

$$h = \frac{1}{2} d_{\text{comfort}} t_{\text{comfort}}^2 + vt_{\text{comfort}}$$  \hspace{1cm} (6)

The problem with the use of this formulation is that when the driver receives the red warning and reduces speed, the value of $h$ changes. This can lead to the deactivation of the warning (the horizon got smaller) while the braking is still required to stop the vehicle. For instance, if the TL is green but the vehicle is not able to pass the intersection before the change of phase, the warning is activated in advance so that a full stop is reached with a smaller deceleration. Assuming that the driver brakes and reduces its speed, as a consequence the horizon gets shorter. If the latter is smaller than the distance of TL, the warning deactivates and the driver maintains the speed constant while a deceleration is still needed to stop the vehicle. As the vehicle gets closer to the TL, the algorithm is triggered and the deceleration warning is issued again. This behavior leads to intermittent warning, that can cause discontinuous braking maneuvers which obviously
does not improve traffic flow. To overcome this problem, the horizon is fixed to $h = 100m$. This value has been chosen considering that with a velocity of 50km/h (that is the limit one in urban roads) the distance computed by means of Equation 5 and Equation 6 is 96m. Therefore, the value has been approximated to $h = 100m$.

5. No further vehicles, other than the ADAS equipped one, are present on the road.

2.4. Testing Scenario

The CarMaker-Simulink environment is used to perform the testing. The goal is to show how the MTLA system is able to help the driver in taking a green wave. To do so, a driver that receives no warning (driver 1) is compared with a driver that receives guidance from the GLOSA systems and follows it perfectly (driver 2). For both drivers, the initial speed is kept constant until an action is required to deal with traffic lights.

The test road scenario described in Figure 3 is a 1500 m long straight road, where four traffic lights with a 75 s cycle are placed. Traffic light location, phases and timing are based on real TLs located in the city of Milan [17]. Their position and relative phases are reported respectively in Figures 4 and 5.

3. Non-optimal MTLA

The MTLA aim is to calculate a reference acceleration profile and to issue a warning that enables the vehicle to pass as many green traffic light as possible, based on all the inputs mentioned before. The warning suggests modifications to the velocity according to a reference acceleration profile.
Figure 4: Traffic lights positions on map

Figure 5: Phase cycles of the four traffic lights
The four TLs ahead of the vehicle are iteratively analyzed every time the algorithm is triggered and the possibility to get one or more greens is subjected to the following conditions:

- a reference velocity that respects road limits and allows to pass at green phase has to be found for each TL;
- if an acceleration/deceleration maneuver is required to the driver, its safety, comfort and feasibility need to be checked;
- a common reference velocity among the considered traffic lights needs to be found.

The above mentioned velocity and acceleration are selected respectively from a velocity and an acceleration range. Indeed, for each traffic light a minimum and maximum velocity that permit the driver to get a green are defined, together with the corresponding accelerations. Through a comparison between the computed reference velocity with the actual vehicle velocity, the warning is issued.

In order to calculate the velocity and acceleration reference profiles, a certain type of motion needs to be chosen. For instance, in literature the velocity range is computed considering a constant speed profile, then the maximum velocity from this interval is considered as the reference one [9, 10]. The drawback of the constant velocity profile is that the time needed for the driver to reach the reference velocity is not taken into account; hence, it is not possible to use it directly as reference for the warning. In fact, in [9, 10], the reference speed is used in a cost function which is then optimized considering the constraint of vehicle dynamics to calculate the control action.

In this work, the driver acceleration phase is considered in the computation of the velocity range: it is represented through a uniformly accelerated motion. This profile does not represent perfectly the vehicle behavior, but it has been chosen as a trade of between accuracy and simplicity.
The computation of the vehicle velocity range required to reach a green traffic light is here analyzed. The following hypotheses are assumed:

1. if the phase of the traffic light under analysis is green, the feasibility to get either the actual or the following green phase is analyzed;
2. if the phase of the traffic light under analysis is red, the feasibility to get only the first green is analyzed;
3. if the first traffic light in front of the vehicle is analyzed, a uniformly accelerated motion (UAM) until the traffic light is supposed;
4. if a traffic light different from the first one is analyzed, the following motion profile is assumed: a uniformly accelerated motion up to the first semaphore is considered, then a constant speed motion (CSM) is adopted (UAM+CSM).

From here on, in order to differentiate among UAM and UAM+CSM, the first traffic light is referred to as first traffic light, while the next ones as $i^{th}$ traffic lights.

The motion profiles described in hypothesis 3 and 4 and the corresponding velocity range computation are analyzed in the next section.

3.1. First Traffic Light

The UAM in Equation 7 is adopted to generate the speed interval for the first traffic light:

\[
\begin{align*}
    d &= vt + \frac{1}{2}at^2 \\
    v_t &= v + at
\end{align*}
\]  

(7)

where; $v$ is the actual speed, $v_t$ is the target speed reached after the acceleration phase, $d$ is the acceleration distance which is set equal to the distance between the front of the vehicle and the traffic light ($d = l_1$).

It is important to note that Equation 7 is a set of two equations with three unknowns: $v_t$, $a$ and $t$, so a parameter still need to be fixed in order to obtain a unique solution. To do so, two cases can be distinguished according to the phase of the traffic light: green or red.
Green Phase. If the first green of the TL is taken into consideration, the minimum velocity is the one that allows crossing the traffic light when the phase is just starting to shift to red. The time at which the phase shift occurs is the remaining time of the first green phase ($t_{1,g1}$), where the first index refers to the number of the TL and the second to the number of the green phase. By substituting $t = t_{1,g1}$ in Equation 7, minimum velocity and acceleration can be computed. The maximum velocity is the one that allows getting the green state in the shortest time possible. So, in order to respect the rules of the road, the maximum velocity is set equal to the road maximum limit speed. By substituting $v_1 = v_{lim,road}$ in Equation 7, maximum velocity and acceleration can be computed too. An example of the two profiles is represented in Figure 6. $t_{lim}$ is the time needed to reach the traffic light with speed equal to $v_{lim,road}$, i.e. the minimum possible time assuming a UAM to the first traffic light. If the second green phase is under analysis the minimum time needed to reach the traffic light correspond to the time of the red to green phase shift $t_{1,r1}$. The maximum one coincides with the time of the second green to red phase shift $t_{1,g2}$. In this way the possibility to get either the first or the second red phase is avoided. By substituting $t_{1,g2}$ and $t_{1,r1}$ in Equation 7, the minimum and maximum velocity and acceleration profiles can be computed.
Red Phase. The minimum time needed for the driver to reach the first available green phase is the one of the red to green phase shift \( t_{1,r_1} \). The maximum one corresponds to the end of the green phase and so to the time of the green to red phase shift \( t_{1,g_1} \). The profiles are shown in Figure 7. By substituting the previously mentioned times \( t_{1,g_1} \) and \( t_{1,r_1} \) in Equation 7, the final formulation can be obtained.

3.2. \( i^{th} \) Traffic Light

A generic traffic light after the first one is analyzed. The motion profile is based upon an acceleration up to the first traffic light, followed by a constant velocity motion up to the \( i^{th} \) traffic light. Such motion profile is described by Equation 8.

\[
\begin{align*}
\frac{d_1}{dt_1} &= vt_1 + \frac{1}{2}at_1^2 \\
v_t &= v + at_1 \\
d_2 &= vt_2 
\end{align*}
\]  \hspace{1cm} (8)

where: \( v \) is the actual speed of the vehicle, \( v_t \) is the target speed reached after the acceleration phase and \( d_1 \) is the distance traveled during the acceleration phase, i.e. the distance \( l_1 \) between the vehicle and the first traffic light. Then, \( d_2 \) is the space driven at constant speed, equal to the distance between the \( i^{th} \) traffic light and the first one, and \( t_2 \) is the difference between the duration of the overall maneuver \( t_{tot} \) and that of acceleration phase \( t_1 \).
Note that (Equation 8) is a set of three equations with four unknowns: $t_1$, $a$, $v_t$, $t_{tot}$. In order to find a solution a parameter still needs to be fixed. Again, two cases need to be distinguished according to the actual traffic light phase: green or red.

**Green Phase.** No constraints on the minimum time to reach the traffic light during its first green exists, thus the maximum velocity is set equal to the road limit speed. By imposing $v_t = v_{lim, road}$ in Equation 8, maximum velocity and acceleration are computed. The maximum time to get the first green of the $i^{th}$ TL is the time of the green to red phase change of that TL. Given the remaining time of the green phase $t_{i,g}$, the total maneuver time is equal to the latter. By imposing $t_{tot} = t_{i,g}$ in Equation 8 minimum velocity and acceleration can be computed. The solution is in Equation 9:

$$\begin{align*}
a_{min} &= \frac{2l}{l_1} - \frac{2v}{l_1} \\
v_{min} &= \frac{l + l_t - t_{i,g} v + \sqrt{(-l - l_1 + t_{i,g} v)^2 + 4l_{i,g} v(l - l_1)}}{2t_{i,g}} \\
t_{1,min} &= l_{i,g} - \frac{l - l_t}{v_{min}}
\end{align*}$$  

(9)

The profiles obtained with the minimum and maximum velocities are shown in Figure 8, respectively on the right and left.

**Red Phase.** The minimum time needed for the driver to reach the first green phase of the $i^{th}$ TL is the one of the red to green phase shift $t_{i,r}$. The maximum one is the time of the end of the green phase $t_{i,g}$. This scenario is represented in Figure 9, respectively on the left (minimum time) and right (maximum time).

3.3. MTLA Algorithm Steps

After defining the speed profile for each TL, the working principle of the algorithm which finds the final profile is shown in the flow chart of Figure 10, the Green Check and Red Check blocks are further illustrated in Figures 11 and 12 respectively. Index $i$ represents the number of the traffic light in analysis, while index $j$ refers to the green phase analyzed. The maximum value of $j$ has been set to 2 in order not to perform too many iterations each time the algorithm is
Figure 8: Motion profile to reach the $i^{th}$ traffic light during its first green phase when the green is on.

Figure 9: Motion profile to reach the $i^{th}$ traffic light during its first green phase when the red is on.
The first two steps from Figure 10 are the Localization and the Activation Check. They calculate the abscissa of the vehicle and check the presence of a TL within a certain distance, called horizon. Once the algorithm finds the first of the four traffic lights within the horizon, the system should be always active until the last semaphore is passed. If the distance between two subsequent TLs is greater
Figure 11: Non-Optimal Multiple Traffic Light Advisor algorithm flow chart - Green Check.
Figure 12: Non-Optimal Multiple Traffic Light Advisor algorithm flow chart - Red Check
than the horizon, the system is deactivated. To avoid so, the horizon for this application is set to 500 m. This value has been decided considering distances between semaphores in [17]. Once the algorithm is activated, an iterative cycle which analyzes the four traffic lights ahead of the vehicle starts.

The feasibility to get a green phase is examined and whenever a feasible maneuver to reach the $i^{th}$ traffic light is found, the following variables are stored inside the algorithm: reference velocity ($v_{ref}$), reference acceleration ($a_{ref}$), warning to be issued and interval of admissible velocities ($v_{adm,i} = [v_{min,i}, v_{max,i}]$).

If it is not possible to pass the traffic light under examination during a green phase, the algorithm terminates and the last stored warning is issued.

The Green Check block, (Figure 11) iterates according to the index $j$ and analyzes the possibility to get either the first ($j=1$) or second ($j=2$) green phase of the $i^{th}$ traffic light. If neither of them can be get, the algorithm terminates and a warning is issued to the driver. The first step of the Green Check is the Velocity Range Definition, this step computes the required velocity range to get the actual green phase $v_{req,i}$ according to the indexes $i$ and $j$ as previously explained. The second step is the Intersection Check. In this block the required velocity range to get the $i^{th}$ green with the one needed to get the $i-1^{th}$ green are intersected as in Equation 10. The aim is to find a velocity range that allows to pass both the $i^{th}$ and $i-1^{th}$ traffic lights.

$$v_{adm,i} = v_{req,i} \cap v_{adm,i-1}$$ (10)

If the first traffic light is studied ($i = 1$), the admissible velocity range is defined by the road minimum (20 km/h) and maximum (50 km/h) admissible velocities. On the other hand, when $i > 1$ the admissible interval is already defined from the $i-1^{th}$ iteration. Now, if Equation 10 results in an empty interval, it is not possible to reach the $i^{th}$ traffic light when its $j^{th}$ green phase is on, so the possibility to get the next one is then analyzed ($j + 1$). On the contrary if the interval is non-empty, $v_{adm,i}$ is defined.

Then, the Actual Velocity Check (AVC) block analyzes the possibility for the driver to keep their current speed rather than performing an acceleration
or deceleration maneuver. If the last check is satisfied, the next traffic light is analyzed \((i + 1)\), if not, the Acceleration/Deceleration Maneuver Check (AMC) is performed. Given that the diver cannot keep its actual speed constant, the safeness of the acceleration or deceleration maneuver is checked. If the AMC block result is positive the driver can get the \(j^{th}\) green of the \(i^{th}\) traffic light. The index \(i\) is increased by one unit and the next traffic light is studied. If the result is negative, the next green phase is analyzed.

The steps performed by the Red Check block are the same as for the green case. The only difference is that, if the actual phase is red, only the possibility to get the first green is evaluated Figure 12.

4. Optimal MTLA

The reference acceleration profile calculated by the non-optimal MTLA is optimized by means of MPC and then the new optimal acceleration profile is used to trigger the warning. An overview of the system is presented in Figure 13, the profile generated by the non-optimal MTLA algorithm (Non-Optimal MTLA Algorithm block) is fed as reference to the model predictive controller (MPC block) along with state constraints (computed in the State Constraints block).

![Figure 13: Optimal Traffic Light Advisor algorithm scheme](image)

With respect to similar works in literature, some noteworthy differences include the following:
1. The cost function here proposed not only includes the error with respect to a reference profile but also penalizes the jerk of the vehicle: in this way it is possible to increase driving comfort [18];

2. a UAM+CSM reference profile is adopted rather than only a CSM one [9];

3. the dynamic variability of traffic lights phases makes the solution space of the optimal problem non-convex and a constant speed reference motion profile is used in literature to tackle this problem. Here, through the use of a more accurate profile (UAM+CSM), a solution can be found faster and it is more likely to be a global-optimum[9];

Since literature lacks detailed explanation of how to consider TL phases into state constraints, an algorithm for generating the position constraints according to TL color is proposed and discussed.

4.1. Motivation

The algorithm in Figure 10 generates a UAM+CSM profile, which could be use as reference for defining the warning; however, it has some weaknesses. The UAM+CSM reference profile is not continuous in acceleration and it does not consider vehicle dynamics, for these reasons it may be difficult for the driver to follow the suggested maneuver or it may lead to discomfort due to high jerk values. In order to overcome the above mentioned problems, the reference profile is optimized as explained later through Equation 12 and the motivations for which a Model Predictive Control approach has been chosen are the following:

- It allows to consider vehicle dynamics as constraint to the Optimal Control Problem (OCP).
- Different goal can be optimized (e.g: following a reference and minimizing jerk).
- It allows to consider online state constraints to reassure that the optimized reference still allows the vehicle to pass through traffic lights correctly (i.e. to pass only during green phases of TLs).
• It allows to introduce indexes that evaluate comfort (acceleration and jerk constraints).

• The predicted acceleration is always more continuous than the reference one, as the jerk is minimized.

4.2. Problem Formulation

The OCP in Equation 12 is defined by the cost function, dynamic model and constraints. In order to obtain a smoother acceleration profile the problem has been defined so to minimize the predicted jerk while still following the reference trajectory.

To take into account vehicle dynamics yet still have a simple modelling, a one degree of freedom plane model is used, meaning that road inclination and lateral dynamics are not considered. The input constraints limit the net maximum braking and net traction force.

Moreover, state constraints are used to avoid passage of the vehicle with red traffic light. To do so the vehicle position is limited to some region of the time-space plane as shown in Figure 14.

It is noteworthy that only two traffic lights and one phase shift for each of them are considered. This is due to the fact that, having chosen the length of the prediction horizon \( t_f \) equal to 6 s, more than two TLs cannot be encountered in this interval and more than one phase shift cannot occur. The length of the prediction horizon is chosen as a trade off between small enough discretization and computational effort while still guaranteeing good optimization results.

In order to calculate the admissible region in ??, the algorithm represented by the flow charts in Figures 15 and 16 is used. The flow charts of the Red Check block is shown in Figure 17. If the time of the phase shift is larger or equal than the prediction horizon, the vehicle should be ahead of the TL for the whole horizon. Otherwise it can be beyond the TL only after the phase shift. The flow chart of the Green Check block is illustrated in Figure 18. If the time of the phase shift is larger than the prediction horizon, the vehicle can
Figure 14: Graphical explanation of the algorithm for state constraints formulation. $s_0$ is the abscissa of the vehicle on the path, $s_1$ and $s_2$ are the abscissa of the first and second traffic light respectively, the lines color represents the phase (green or red), while $t_1$ and $t_2$ are the times of the phase change for the first and second TLs.

Figure 15: Flow chart of the algorithm for the definition of state constraints.
be ahead or beyond the TL. Contrary, if the time is lower or equal, a further analysis is performed on the possibility to pass the TL during the actual green phase. This is done by checking if the MTLS algorithm calculates that it is not possible to pass the $i^{th}$ traffic light ($N_{green} < i$) or to pass it during the second green phase ($N_{pass} > 1$). If one of these two conditions is verified the vehicle should be ahead of the semaphore after the phase shift, otherwise it should be beyond it.

Once the admissible regions are obtained for the two TLs, the Range Intersection block (Figure 15) is executed. It consists of making the intersection of the state constraints of the single TLs as in Equation 11:

$$[s_{min}(t), s_{max}(t)] = [s_{min,1}(t), s_{max,1}(t)] \cap [s_{min,2}(t), s_{max,2}(t)]$$

In addition to the position state constraint, also velocity, acceleration and
Figure 17: Red Check block - flow chart of the algorithm for the definition of the $i^{th}$ traffic light state constraints

Figure 18: Green Check block - flow chart of the algorithm for the definition of the $i^{th}$ traffic light state constraints
Where \( j(t) \) is the jerk, \( a_{\text{ref}}(t) \) and \( v_{\text{ref}}(t) \) are the reference acceleration and velocity from the non-optimal MTLA algorithm and \( w_a, w_v \) and \( w_j \) are the weights of acceleration, velocity and jerk cost terms respectively. \( A_f \) is the frontal area of the vehicle, \( \rho \) is the air density, \( C_d \) is the aerodynamic drag coefficient, \( m \) is the vehicle mass, \( C_r \) is the rolling resistance coefficient and \( F \) is the generalized longitudinal tire force. \( F_{\text{min}} < 0 \) represents the maximum braking force and \( F_{\text{max}} > 0 \) represents the maximum traction force.

5. Simulation

In this section, results of test cases are shown. Position, speed, acceleration and energy consumption are considered. Moreover, to prove that the developed MTLA is able to effectively reduce vehicle consumption, a simple energetic analysis is proposed considering instantaneous and average energy consumptions.

Instantaneous energy consumption is computed starting from the instantaneous power \( P \) (\( P = T \omega \) with \( T \) being the motor torque and \( \omega \) the motor angular velocity) as in Equation 13:

\[
I.E.C. = P \frac{\text{travelled time}}{\text{travelled distance}}
\]
Average energy consumption is computed starting from the instantaneous one, as in Equation 14:

\[
A.E.C_n = \frac{I.E.C_1 + \ldots + I.E.C_n}{n}
\]

\[
A.E.C_{n+1} = \frac{I.E.C_{n+1} + \ldots + nA.E.C_n}{n+1}
\]  

(14)

5.1. Non optimal MTLA

This test case highlights advantages of MTLA in terms of stops, travel time and vehicle consumption. From Figure 19 it is visible how driver 1 is subject to three stops, thus increasing the total travel time to cross all the traffic lights. With the initial velocity of 40 km/h driver 1 is able to reach the first traffic light in its green phase, while it is not able to get the second one. On the contrary, driver 2 given an acceleration warning is able to avoid such stop. From Figures 20 and 21 it can be seen that driver 1 increases its velocity as soon as the first traffic light is in the horizon of activation of the algorithm. This acceleration allows also to get the third semaphore, but not the fourth. It is to be noticed that the vehicle stops accelerating before reaching the first traffic light (Figures 20 and 21) because a velocity that allows reaching all the previously analyzed three green light is reached.

When the second traffic light phase shifts to green again driver 1 accelerates and reaches the speed of 40 km/h. While approaching the third traffic light, driver 1 starts decelerating to stop the vehicle again as it is red. The same deceleration/acceleration maneuver occurs while reaching the fourth traffic light. Given a deceleration warning, driver 2 is able to avoid these two stops. As soon as driver 2 overcomes the second traffic light, a new profile can be calculated which allows to pass also the fourth semaphore without stopping.

It is noteworthy to note that when the algorithm is triggered at first, no feasible solution could be found to reach the fourth traffic light; however, as the vehicle passes across the TLs, new profiles are calculated and a feasible UAM+CVM profile to cross the fourth semaphore is found. This is due to the algorithm’s continuous update.
Moreover, it is visible from Figure 21 that acceleration values of driver 2 are lower than driver 1, meaning higher maneuver comfort.

Figure 22 shows the average energy consumption. It can be noticed how the final average consumption (after passing the four traffic lights) of driver 2 is lower than driver 1: 9.4 kWh/100km for driver 2 versus 15 kWh/100km for driver 1. Hence, considering these last values, MTLA shows a 37.3% reduction in energy consumption.
Figure 21: MTLA Test Case - acceleration profiles of driver 1 and driver 2

Figure 22: MTLA Test Case - Average Energy Consumption of driver 1 and driver 2
5.2. Optimal MTLA

First, the comparison between the vehicle equipped with optimal MTLA system (driver 2) and the one without any kind of warning system is reported (driver 1). Looking at Figure 23 it can be noticed how driver 1 has to stop at three traffic lights, while driver 2, having followed the warning, is able to take a green wave and thus reduce the total travel time. The comments done for the Non-Optimal MTLA are also valid for this case since driver 1 is the same and driver 2 has the same behavior with the only difference that the acceleration profile is smoother. Acceleration and velocity of driver 1 and 2 are plotted with respect to abscissa in Figures 24 and 25. Figure 26 shows the average consumption (computed as in Equation 14) for driver 1 and driver 2. As for the previous plots, comments done previously are valid also for the energetic analysis. It can be appreciated how driver 1 final average consumption (15 kWh/100km) is higher than driver 2 (9.3 kWh/100km), resulting in a 38% reduction.

Now, the non-optimal and optimal MTLA are compared for this test case. In Figure 27 the vehicle acceleration is plotted with respect to abscissa. Also from this plot it can be seen how the optimal strategy allows to have a more continuous acceleration profile. In Figure 28 an example of the non-optimal and
Figure 24: Optimal MTLA Test Case 1 - acceleration profiles of driver 1 and driver 2

Figure 25: Optimal MTLA Test Case 1 - velocity profiles of driver 1 and driver 2
optimal references along the prediction horizon is shown, the instant to which it refers is the beginning of the first acceleration phase (100 m from the starting point of the simulation). It can be noticed how the predicted acceleration (optimal reference) gradually reaches the non-optimal reference acceleration. Also the velocity profiles are reported in Figure 29. Non-optimal and optimal average consumption profiles are depicted in Figure 30 it can be seen how, from an energy point of view, the optimal approach is not introducing significant advantages. However, a final reduction of 2% of the average energy consumption is visible at the end of the simulation.
Figure 27: Optimal MTLA Test Case 1 - non-optimal and optimal acceleration profiles

Figure 28: Optimal MTLA Test case 1 - non-optimal and optimal acceleration profiles along prediction horizon at given instant of simulation
Figure 29: Optimal MTLA Test case 1 - non-optimal and optimal velocity profiles

Figure 30: Optimal MTLA Test Case 1 - non-optimal and optimal Average Energy Consumption
References

[1] A. ISTAT, Incidenti stradali-anno 2019.

[2] W. H. Organization, et al., Global status report on road safety 2018: summary, Tech. rep., World Health Organization (2018).

[3] M. Lu, K. Wevers, R. Van Der Heijden, Technical feasibility of advanced driver assistance systems (adas) for road traffic safety, Transportation Planning and Technology 28 (3) (2005) 167–187.

[4] R. Schmied, H. Waschl, R. Quirynen, M. Diehl, L. del Re, Nonlinear mpc for emission efficient cooperative adaptive cruise control, IFAC-paperonline 48 (23) (2015) 160–165.

[5] M. Li, K. Boriboonsomsin, G. Wu, W.-B. Zhang, M. Barth, Traffic energy and emission reductions at signalized intersections: a study of the benefits of advanced driver information, International Journal of Intelligent Transportation Systems Research 7 (1) (2009) 49–58.

[6] M. Barth, S. Mandava, K. Boriboonsomsin, H. Xia, Dynamic eco-driving for arterial corridors, in: 2011 IEEE forum on integrated and sustainable transportation systems, IEEE, 2011, pp. 182–188.

[7] L. S. Cai, G. B. Ning, Adaptive driving speed guiding to avoid red traffic lights, in: Applied Mechanics and Materials, Vol. 347, Trans Tech Publ, 2013, pp. 3832–3836.

[8] K. Katsaros, R. Kernchen, M. Dianati, D. Rieck, Performance study of a green light optimized speed advisory (glosa) application using an integrated cooperative its simulation platform, in: 2011 7th International Wireless Communications and Mobile Computing Conference, IEEE, 2011, pp. 918–923.
[9] B. Asadi, A. Vahidi, Predictive use of traffic signal state for fuel saving, IFAC Proceedings Volumes 42 (15) (2009) 484–489.

[10] B. Asadi, A. Vahidi, Predictive cruise control: Utilizing upcoming traffic signal information for improving fuel economy and reducing trip time, IEEE transactions on control systems technology 19 (3) (2010) 707–714.

[11] E. Kural, S. Jones, A. F. Parrilla, A. Grauers, Traffic light assistant system for optimized energy consumption in an electric vehicle, in: 2014 International Conference on Connected Vehicles and Expo (ICCVE), IEEE, 2014, pp. 604–611.

[12] M. Seredynski, W. Mazurczyk, D. Khadraoui, Multi-segment green light optimal speed advisory, in: 2013 IEEE International Symposium on Parallel & Distributed Processing, Workshops and PhD Forum, IEEE, 2013, pp. 459–465.

[13] M. Seredynski, B. Dorronsoro, D. Khadraoui, Comparison of green light optimal speed advisory approaches, in: 16th international IEEE conference on intelligent transportation systems (ITSC 2013), IEEE, 2013, pp. 2187–2192.

[14] V. Nguyen, O. T. T. Kim, T. N. Dang, S. I. Moon, C. S. Hong, An efficient and reliable green light optimal speed advisory system for autonomous cars, in: 2016 18th Asia-Pacific Network Operations and Management Symposium (APNOMS), IEEE, 2016, pp. 1–4.

[15] F. Arena, G. Pau, An overview of vehicular communications, Future Internet 11 (2) (2019) 27.

[16] S. G. Klauer, T. A. Dingus, V. L. Neale, J. D. Sudweeks, D. J. Ramsey, Comparing real-world behaviors of drivers with high versus low rates of crashes and near crashes, Tech. rep. (2009).
[17] A. M. A. Territorio, Centrale di controllo del traffico, https://www.amat-mi.it/it/servizi/centrale-controllo-traffico/, [Online; accessed 19-March-2022] (2020).

[18] I. Bae, J. Moon, J. Seo, Toward a comfortable driving experience for a self-driving shuttle bus, Electronics 8 (9) (2019) 943.