Forecast Based Traffic Signal Coordination Using Congestion Modelling and Real-Time Data

Pietro Meschini

December 13, 2018

Abstract

This dissertation focusses on the implementation of a Genetic Algorithm based Real-Time Signal Coordination module for arterial traffic, as a proof of concept for the potential of integrating signal optimisation tools into a Real-Time Traffic Management System. The endeavour represents an attempt to address a number of shortcomings observed in most currently marketed on-line signal setting solutions. Within a fully functional traffic modelling and management framework, the optimiser can be developed independently, leaving ample space for future adaptations and extensions, while relying on the best available technology for its inputs, solution evaluation and visualisation, and command implementation.
The optimiser can then operate on a high quality network model that is well calibrated and continuously up-to-date with real-world road conditions; rely on robust, multi-source network wide traffic data, rather than being attached to single detectors; manage area coordination using an external simulation engine, rather than a naïve flow propagation model that overlooks crucial traffic dynamics; and even incorporate real-time traffic forecast to account for transient phenomena in the near *future* to act as a feedback controller. finale

Contents

Introduction

The fundamental role of *traffic signals* is to equitably and efficiently administer the right of way amongst conflicting streams of road users.

Since the first sporadic appearances around the turn of the 20th century, traffic lights have become a ubiquitous feature in the everyday life of all road users, regardless of their preferred mode of transportation: whether they sit behind the wheel of their own car, walk, or let the public service carry them about their business, traffic lights will be regulating their movements and those of others around them (most noticeably, of those in front).

It is therefore natural that traffic signals should garner so much attention: they are perceived (only *sometimes* unfairly) as a major source of delay and frustration to drivers, and the tantalising idea of an intelligent traffic control system often comes to identify, in the general public fantasy, with the very notion of an *Intelligent Transport System*.

In fact, a history of case studies shows that wherever public money has been invested into the development and maintenance of a signalisation system tailored to the transportation needs of a community, the returns have invariably surpassed expenditures by far Koonce, Rodegerdts, Lee, Quayle, Beaird, Braud, Bonneson, Tarnoff, and Urbanik [2008]. comment on WHAT parameters were considered to determine "returns"

Carefully planned signalisation allows a more efficient use of the existing road infrastructure, minimising the stress suffered by drivers as well as the risk of accidents, favouring public transport and improving air quality, with a positive impact on virtually every aspect of life in a modern city.

About Notation

A quick glossary of the relevant variables is provided below, alongside the units of each dimensional quantity.

For a leaner presentation of the model, subscripts referring to topological elements may be dropped to simplify notation.

Network Topology

$i, j \in \mathrm{N}$		nodes (junctions)
$a, b \in \mathrm{A} \subseteq \mathrm{N} \times \mathrm{N}$		arcs (lane groups)
ℓ_a	[m]	length of arc a
$(\mathrm{N}_a^- \, , \, \mathrm{N}_a^+) = a$		tail and head nodes of arc a
$\mathrm{A}_i^+ = \{a \in \mathrm{A} \,\vert\, \mathrm{N}_a^- = i\}$		forward star of node i (outgoing arcs)
$\mathrm{A}_i^- = \{a \in \mathrm{A} \,\vert\, \mathrm{N}_a^+ = i\}$		backward star of node i (incoming arcs)
$y, z \in \mathrm{Y}$		manoeuvres

Signal Phases

$p, q \in \mathrm{P}_j$	signal phases at junction j
$\mathrm{A}_p \subseteq \mathrm{A}_j^-$	lane groups open during phase p

Signal Timing

t_j^C	[s]	cycle time at junction j
t_p	[s]	nominal duration of phase p
g_a	[s]	effective green duration for arc a
$\gamma_a = \dfrac{g_a}{t_j^C}$	[%]	effective green share of arc a
t_j^L	[s]	time lost per cycle at junction j
t_j^O	[s]	offset of junction j

Demand and Supply

q_a	[veh/s]	demand flow on arc a
$\hat{q}_a$	[veh/s]	saturation flow of arc a
$\phi_a = \dfrac{q_a}{\hat{q}_a}$		flow ratio on arc a
$\chi_a = \dfrac{\phi_a}{\gamma_a}$		saturation on arc a

Simulation Parameters

$\tau \in \mathrm{T}$		intervals of the simulation window
Δt^{τ}	s	duration of interval τ

Simulation Results

$n_{a,t}$	[veh]	vehicles on arc a at time t
$t^{t}_{a,t}$	[s]	travel time for arc a entering at time t

Performance Indicators

t^{Q}_{a}	s	queue clearance time on arc a (per cycle)
ω^{Q}_{a}	[%]	queue length relative to total length of arc a
ω^{D}_{a}	s	average delay of arc a
ω^{S}_{a}		share of q_a stopping at or before N^{+}_{a}
ω^{n}_{a}, ω^{n}_{C}	[veh]	total inflow to arc a or all sections of corridor C
ω^{t}_{a}, ω^{t}_{C}	[s]	user time spent on arc a or corridor C

Chapter 1

Signalisation of Urban Networks

The present work concerns the regulation of urban traffic by means of traffic signals.

The *lights*, which are nowadays a ubiquitous feature of the urban landscape, first appeared in 1868 outside the British House of Commons in Victorian London, where the horse drawn carriage traffic was becoming an insurmountable barrier posing a serious threat to pedestrians. Since then, and especially as motor cars were introduced, traffic regulation proved indispensable to administer the right of way among competing traffic flows and safeguard the more vulnerable users of the urban road environment.

This chapter introduces the formal representation of the signalised road network used for all practical purposes in this dissertation. It builds upon the definition of the network itself to describe the way it interacts with its users, modelling the problems that traffic signals need to tackle and the ways in which they might do so. Finally, the most relevant signal planning approaches based on the paradigm just outlined are illustrated, as they form the basis for the adaptive signalisation strategies to which the present work aims to contribute.

1.1 The Urban Network

In the context of transport modelling and planning, a transportation network is represented as a *directed graph* in the mathematical sense, with the *vertices* representing locations and the *edges* connections that a user may travel between them. The term *connection* is used loosely on purpose here, since in general these need not be *roads* but may be transit lines, footpaths, railways etc. each with complex properties which determine its *cost*, or even accessibility, to a given class of users.

In its extremely simplified acceptation of *road network* which will serve the purposes of the present work, a transportation network may be reduced to an ordered pair (N, A) where

- N is the set of vertices of the graph, called *nodes*, representing junctions and road ends;
- A is the set of directed edges between them, called *arcs*, along which the users move.

This allows to encapsulate both the network topology and the properties of individual roads, which determine the way in which the users will interact with them: the choice of a path between two nodes depends on the perceived cost of each alternative as determined by a combination of its properties, e.g. length, toll, number of lanes, pleasantness; the same properties, albeit through conceptually different mechanisms, determines how the users will be able to move along the chosen path.

1.2 Anatomy of a Signal Plan

The following section briefly illustrates the main features of a signal plan devised for urban traffic regulation. This term encompasses all timings and schedules behind the delicate clockwork of traffic signals, from the elements that constitute a single signal program at one of the many junctions of the network, to the succession of network-wide program changes designed to meet the daily evolution of traffic demand and the propagation of vehicle flows. The features presented in this section fully define what is commonly called a pre-timed plan, and as such do not describe any real-time actuation or decision making logic. They are themselves, however, the decision variables of most optimisation methods and adaptive strategies, and it is crucial to understand their significance in order to appreciate the diversity of setting and control approaches illustrated in more detail throughout this chapter.

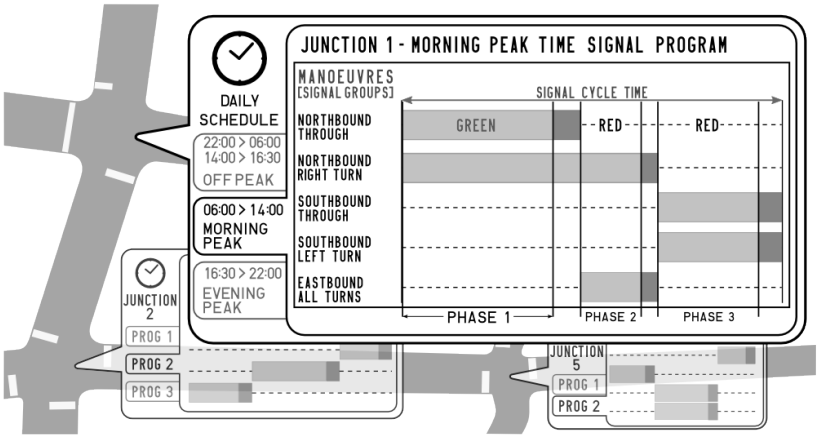

FIGURE 1.1 – Elements of a network-wide signal plan: a daily schedule specifies the signal programs running at each intersection. The sequence and duration of signal phases repeats over the course of every signal cycle as specified by the different signal programs, administering junction capacity amongst the expected traffic flows. During each phase, a set of compatible manoeuvres is allowed through while the others remain closed.

Signal Phases

Traffic signals exist mainly to separate conflicting traffic flows competing for the right of way at a road intersection. The natural way of doing so is to bundle compatible (e.g. non-secant) manoeuvres which may be safely performed simultaneously into signal phases, so that the corresponding flows may be allowed through the junction in turn. Phases are the fundamental blocks of a signal program, and are usually repeated in the same order at every signal cycle, although some signalisation systems provide phase skipping, usually as part of their public transport prioritisation strategy. Manoeuvres may pertain to different modes of transport, meaning that cars, trams and pedestrians are taken into joint consideration and can be given the right of way during the same signal phase.

Consider a junction, i.e. a network node $j \in \mathrm{N}$ where it is possible to perform a given set of manoeuvres Y_j . The generic manoeuvre $y \in \mathrm{Y}_j$ may be:

- a turn, from an arc $a \in \mathrm{A}_j^-$ of the node's backward star, to a forward star arc $b \in \mathrm{A}_j^+$;
- a tram crossing or similar transport system specific operation;

- a pedestrian crossing affecting one or more arcs either entering or leaving the junction.

In order to present a straightforward definition of *manoeuvres* in relation to junction layout and signalisation, the focus will henceforth be on the movement of private vehicles only, unless otherwise specified. It shall be clear that the principles of manoeuvre compatibility illustrated in this manner in Figure 1.2 may be easily generalised to different and etherogeneous modes of transport, such as public transport, pedestrians and bicycles.

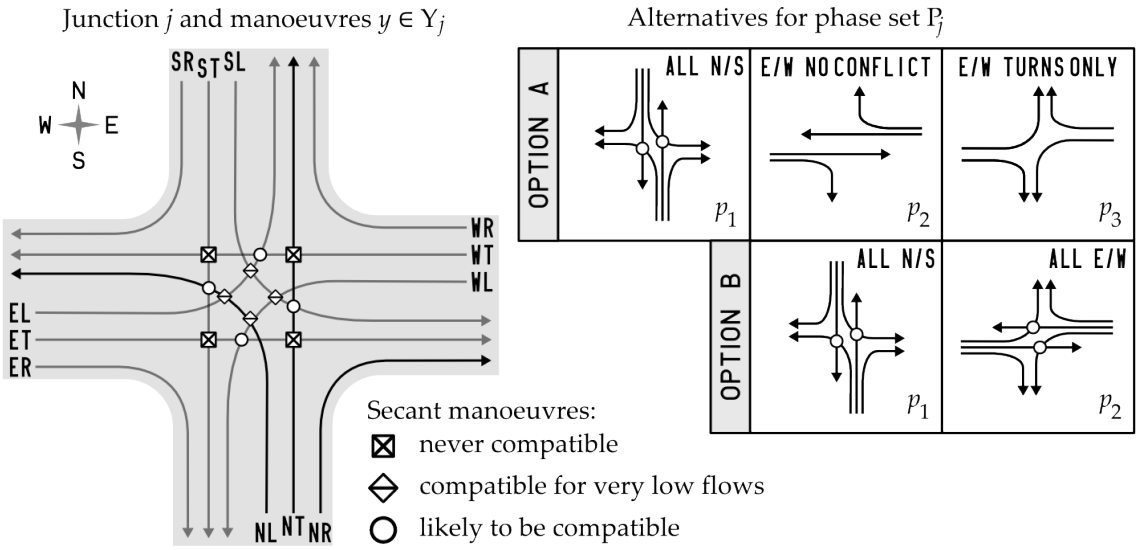

FIGURE 1.2 – Manoeuvres at an intersection, conflict areas and possible phasing options: option A avoids direct conflicts between Eastbound (E-) and Westbound (W-) manoeuvres, as would be desirable if high volumes were expected along that direction; option B favours a lower number of phase changes (less time lost) assuming flows to be such that left turning vehicles have space to wait at the middle of the intersection, until the oncoming through flow decreases enough to let them cross.

Given the layout of a junction j, different manoeuvres may or may not be safe to perform simultaneously, as exemplified in Figure 1.2. This information, which may well depend on the flow conditions, is easily represented by a square Boolean matrix where rows and columns correspond to each manoeuvre and elements comply with the following rule:

$$\delta_{yz} = \begin{cases} 1 & \text{if } y \text{ and } z \text{ are compatible} \\ 0 & \text{otherwise} \end{cases} \quad \forall y, z \in \mathrm{Y}_j \quad . \tag{1.1}$$

Each possible subset of manoeuvres $p \subseteq \mathrm{Y}_j$ potentially identifies a *signal phase*. A viable set of phases P_j for the junction however must belong to the space of *feasible* signal phases, i.e. all possible sets of manoeuvres contained in the power set $\wp(\mathrm{Y}_j)$ whose elements are mutually compatible according to (1.1) . The union of all phases must also include every available manoeuvre at least once.

Formally, P_j must therefore comply with the following properties:

$$\mathrm{P}_j = \left\{ p \in \wp(\mathrm{Y}_j) : \prod_{y \in p} \prod_{z \in p} \delta_{yz} = 1 \right\} \quad , \quad \bigcup_{p \in \mathrm{P}} p = \mathrm{Y}_j \quad . \tag{1.2}$$

Clearly, the power set $\wp(\mathrm{Y}_j)$ contains sets of manoeuvres that, although compatible and technically feasible, make little practical sense. The selection of an optimal set of phases P_j satisfying relation (1.2) with respect to a specific objective (e.g. minimum total delay for given demand flows) is a combinatorial bi-level problem, usually solved through a *what-if* approach in which the selection of a good set of phases remains largely a traffic engineer's task.

Conceptually, the determination of signal phases is thus driven by the interactions between manoeuvres. From a practical point of view, however, administration of the right of way by means of traffic signals cannot transcend the junction layout. For example, it is only possible to separate manoeuvres into different phases if each has a dedicated lane that allows vehicles to queue for it without hindering traffic that is headed elsewhere. In fact, as everyday experience testifies, traffic signals do not allow or prohibit manoeuvres directly, but rather regulate vehicle egress from lanes (or lane groups) dedicated to specific sets of manoeuvres.

Each lane or group of adjacent lanes a sharing the same manoeuvre set $\mathrm{Y}_a \in \mathrm{Y}_j$ can be conceptually assimilated into a *lane group*: a single independent arc $a \in \mathrm{A}_j^-$ of the node backward star. Let A_p be the set of lane groups which are given the green light during signal phase p, and Y_a the manoeuvres that can be performed from lane group a. The set of manoeuvres allowed during phase p is therefore

$$p = \bigcup_{a \in \mathrm{A}_p} \mathrm{Y}_a \quad . \tag{1.3}$$

The set of manoeuvres Y_a specific to each lane group a is relevant for the determination of the arc effective outflow capacity, which may be affected by partial conflicts with other manoeuvres allowed during the same phase. The HCM (2010) manual **??** presents practical methods for quantifying such effects.

Signal Programs

A signal program contains the state switching times for all signals at a given junction. For signal planning and optimisation, it is practical to view the program as a succession of signal phases with specific durations, as portrayed in Figure 1.1: during each phase a set of arcs are open, allowing users to carry out the corresponding manoeuvres, while the others arcs remain closed and accumulate queues.

A program for junction j consists therefore of a cyclic set of instructions spanning a period called *cycle time* t_j^C: given a phase set P_j, these specify the start and end of each signal phase with respect to the beginning of the signal cycle.

Transitions between subsequent phases are usually enacted via pre-timed signal state change sequences that handle the closure of a set of lane groups before opening the next.

Daily Schedule

It is common practice to tailor several signal programs to the traffic conditions normally observed at different times of the day, in order to meet each scenario with the best possible allocation of resources. The daily schedule defines the sequence of programs that each junction will run over the course of the day.

Cycle Time

The cycle time t_j^C is the *period* of the signal program, i.e. the time lapse between two occurrences of the same signal phase at a given junction. It affects the average delay and the level of saturation at which the intersection may operate. In general, longer cycle times imply larger average delays, but increase the total throughput, which may be necessary to deal with high demand flows by attenuating the effects of the time lost in signal phase changes.

Effective Green Shares

The nominal duration of each phase t_p is seldom exploited by demand flows at the full capacity of the corresponding arcs: even assuming that vehicles are not held back by downstream congestion, it is necessary to account for some transient phenomena affecting the performance of a junction.

As the signals turn green at the beginning of each phase, some time is lost before the queuing vehicles start moving, and some more passes before the flow through the stop line reaches the arc capacity. On the other hand, if a lane group remains open during two subsequent phases such effects will be smaller, in proportion. After taking into account all delays and extensions, the portion of cycle time during which a given lane group a may allow traffic onto the junction at full capacity is referred to as its *effective green share*. The absolute and relative durations of effective green experienced by lane group a during phase p are denoted respectively as:

$$g_{a,p} \in [0, t_p] \qquad \text{and} \qquad \gamma_{a,p} = \frac{g_{a,p}}{t_j^C} \quad . \tag{1.4}$$

It is not uncommon to have a lane group open during more than one phase: typically, an approach experiencing high traffic volumes is given the right of way over two or more consecutive phases without incurring further lost time in the phase change.

The effective green of each arc a is then calculated from the total effective green time it gathers over all relevant phases:

$$g_a = \sum_{p \in \mathrm{P}_j} g_{a,p} \quad \text{with} \begin{cases} 0 < g_{a,p} \leq t_p & \text{if } a \in \mathrm{A}_p \\ g_{a,p} = 0 & \text{otherwise} \end{cases} \quad \text{and} \quad \gamma_a = \frac{g_a}{t_j^C} \quad . \tag{1.5}$$

Signal Offset

When multiple signals are involved, it is important to consider that vehicles that cross a signalised junction become packed into *platoons*, which will eventually reach yet another signal-controlled stop line: adjusting the relative timing of adjacent junctions so that platoons meet a green light greatly affects the average delay incurred by the user.

Synchronisation issues are addressed by defining a global time reference, with all junctions sharing the same cycle time or integer fractions thereof. Each junction may then have all of its phase switching times anticipated or delayed in order to operate in concert with the neighbouring ones. The amount of time t_j^O, by which the beginning of a cycle at one junction j lags or leads the global reference instant, is referred to as a positive or negative *offset*, respectively.

1.3 Signal Setting

Long before microprocessors and sensors made adaptive real-time traffic control an everyday reality, the notion of signal plan optimisation identified with a range of techniques for designing good signal plans based on historical demand flows, which will henceforth be referred to as *offline* signal setting.

It is worth noting that such methods are not only still used for planning, but lie at the core of several adaptive signal setting approaches: once a *signal setting policy* is chosen to determine the best signalisation parameters for given traffic conditions, it makes little difference from the methodological point of view whether the input variables are determined from historical data or fed in real-time by sensors.

Naturally, the notion of offline planning does not imply that the dynamic interaction between signal setting and driver behaviour can be disregarded: for example, the assumption often made that route choices are fixed and unaffected by signal settings has warranted the formulation of planning strategies which have proven quite patently inadequate in the real world, as first discussed in Dickson (1981). While optimisation of a single junction for given flows may be a relatively simple problem with an analytical solution, devising a plan for an entire network is an entirely different task.

This section introduces the fundamentals of network signalisation design, describing the methods commonly used to determine the foremost features of a signal program, including cycle time, offsets and green share allocation.

1.3.1 Performance of Isolated Signalised Junctions

only a more specific case of junction

The concept of *performance* of a signalised junction may be defined in several ways, but in general terms it represents a gauge of the interaction between supply and demand with respect to a choice of metrics. As such, it depends on the junction physical layout, on the distribution of vehicle arrivals in time and on the signal that regulates their departure times.

Several flow models were introduced in scientific literature to reproduce arrival and departure phenomena. For all signal planning purposes, traffic flow is usually assimilated to a fluid stream according to the *macroscopic* paradigm, which differs substantially from the microscopic approach where the trajectory of each single vehicle is explicitly considered.

More specifically, vehicle *departures* from a stop line are modelled as a uniform flow. If the *arrival* flows are sufficiently lower than capacity, their inherent random component can be neglected and they are also considered deterministic. Conversely, if stochasticity of arrival flows is significant, as it occurs when they approach the relevant arc capacity, or are very low, a random component is added to the simple deterministic model as in ref Webster, (1956) . This section will present the basic relationships between signal timing variables and junction performance with reference to the simple deterministic model.

Queues and Queue Clearance

Consider a single arc (lane group) $a \in \mathrm{A}_j^-$ entering a signalised junction $j \in \mathrm{N}$, with a constant demand flow of vehicles q_a arriving over the entire cycle. The flow can only be

discharged onto the junction during the effective green time, at the constant saturation flow rate $\hat{q}_a$ given by the arc capacity and possibly degraded due to conflicts with other arc flows. The *flow ratio* between demand and saturation is denoted as:

$$\phi_a = \frac{q_a}{\hat{q}_a} \quad . \tag{1.6}$$

During the rest of the cycle, the departure rate is zero and vehicles have to stop, forming a *queue*, which has to be discharged during the next green phase if it is not to grow indefinitely.

The saturation flow $\hat{q}_a$ must therefore be sufficient to serve the queue accumulated over the red phase, which has duration $t_j^C - g_a$, in addition to the flow of vehicles that keep arriving during the green phase g_a.

This relationship is illustrated in figure 1.3 and may be formalised by considering the following expression for the *queue clearance time* in terms of the signal timing and flows just described:

$$t_a^Q = \frac{q_a \left(t_j^C - g_a\right)}{\hat{q}_a - q_a} = \frac{\phi_a(1-\gamma_a)}{1-\phi_a}\, t_j^C \quad , \quad \forall a \in \mathrm{A}_j^- \quad . \tag{1.7}$$

Vehicle Stops

In this context, it makes sense to assume that vehicles will stop if they reach the stop line during the red phase or if they have to join the back of a queue that has yet to be fully discharged, although this is a slightly conservative approximation as the back of the queue might not be standing still during the green phase.

The number of vehicles that end up stopping (or significantly slowing down) during every signal cycle can therefore be expressed as

$$n_a = q_a \left(t_j^C - g_a + t_a^Q\right) = \hat{q}_a\, t_a^Q \tag{1.8}$$

where the right-hand side equality is justified simply by the definition of clearance time t^Q given by equation (1.7) under the assumption that standing vehicles will discharge onto the junction at the maximum possible flow rate during the effective green phase.

This in turn leads to the theoretical definition of the *stop ratio*, an essential metric indicating what fraction of the total flow of vehicles will have to stop at the junction:

$$\omega_a^S = \frac{\hat{q}_a t_a^Q}{q_a t_j^C} = \frac{1-\gamma_a}{\phi_a} \quad , \tag{1.9}$$

which is proportional to the red share of the cycle time and increases as the arrival rate approaches the discharge capacity. Quite obviously for values of $\phi_a \geq 1$, but also if $\gamma_a < \phi_a$ queues cannot be fully discharged at every cycle, and all vehicles end up stopping: in this case, the queue can grow indefinitely.

Average Delay

Assuming constant arrival and departure rates, the total delay experienced at each cycle by all users from a given approach a corresponds to the integral over time of the queue size (the

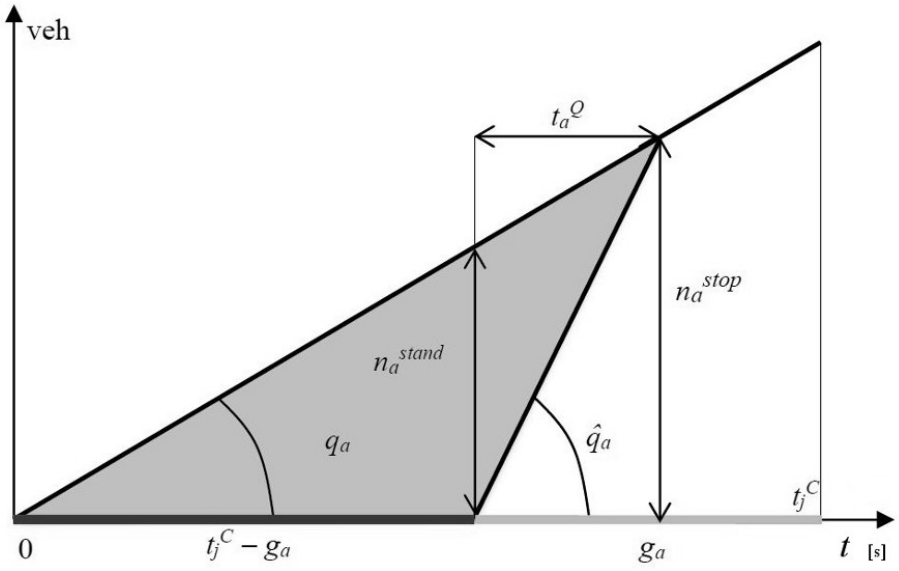

FIGURE 1.3 – Geometric determination of stopped vehicles and queue clearance for one approach given the relevant demand flow, saturation flow, cycle and green time. The grey triangle between the arrival cumulative, the departure cumulative and the horizontal axis covers the number of vehicles queuing at any given moment. Notice that the number of standing vehicles $n_a^s tanding$ at the beginning of the effective green does not account for all vehicles that need to stop $n_a^s top$ according to the approximation given by equation (1.8).

area of the greyed out triangle in Figure 1.3), whence the average delay ω_a^D per vehicle is found to be

$$\omega_a^D = \frac{\left(t_j^C - g_a\right)\left(\hat{q}_a\, t_a^Q\right)}{2\,\left(q_a\, t_j^C\right)} = \frac{\left(t_j^C - g_a\right)^2}{2\,\left(1-\phi_a\right)\,t_j^C} \quad , \tag{1.10}$$

using (1.7) for the queue clearance time t_a^Q.

Clearly, the above equation (1.10) assumes no standing queues at the end of a cycle. More complex delay functions can be obtained by considering stochastic fluctuations of arrival flows ref (Webster, 1958) . Flows exceeding the arc capacity require the introduction of either simulation models or empirical adaptations of analytical models, such as the coordinate transformation method introduced by ref Kimber and Hollis (1979) and later adopted by the popular ref HCM traffic manual (2010) .

Critical Flow Ratio and Saturation

The saturation flow characterising each lane group depends on various factors, such as

- total road width,
- visibility,
- conflicts with other manoeuvres served during the same phase,
- presence of dedicated turn bays to alleviate such conflicts.

Conflicts are particularly relevant to left turns, or turns encroaching a pedestrian crossing: scrupulous phase planning can minimise the number and entity of such conflicts.

The flow ratio ϕ_a quantifies the expected demand on a given lane group a in relation to its *nominal* saturation capacity. The saturation level χ_a is determined by the ratio of demand flow to its *outflow capacity*, which is further limited by the signal, inasmuch as each arc can only be open for a limited share of the available green time:

$$\chi_a = \frac{q_a}{\gamma_a \, \hat{q}_a} = \frac{\phi_a}{\gamma_a} \quad . \tag{1.11}$$

For values of $\gamma_a < \phi_a$ the saturation level is above 100 % and the flow cannot be served, leading to queues that grow indefinitely until demand drops.

When multiple lane groups are to be open simultaneously during phase p, the *critical flow ratio* ϕ_p is given by the approach which is relying most heavily on the phase in question. The concept is formalised in equation (1.12) by scaling the flow ratio of each approach in proportion to the share of its green time represented by the current phase.

In other words, in searching for the maximum flow ratio, only the share of flow that each lane group must serve during the specific phase is considered:

$$\phi_p = \max \left\{ \phi_a \frac{\gamma_{a,p}}{\gamma_a} \mid a \in \mathrm{A}_p \right\} \quad , \tag{1.12}$$

whence conversely the *critical lane group* of phase p is also identified as

$$\mathrm{A}_p^* = \left\{ a \in \mathrm{A}_p \mid \phi_p = \phi_a \frac{\gamma_{a,p}}{\gamma_a} \right\} \quad . \tag{1.13}$$

The *critical saturation* of signal phase p is obtained by applying (1.11) to its critical lane group:

$$\chi_p = \frac{q_p}{\gamma_{\mathrm{A}_p^*}} \quad , \tag{1.14}$$

noting that in the particular case where each lane group is only open during a single phase, critical saturation occurs on the one registering the highest flow ratio.

Since different lane groups may experience different effective green shares, should be calculated using the effective green experienced by the same lane group during that phase, which is practically considered the *phase effective green*:

$$g_p = g_{\mathrm{A}_p^*,p} \quad . \tag{1.15}$$

Finally, the total *junction flow ratio*, which gives a measure of how busy the intersection really is, can be calculated as the sum of the critical flow ratios over all phases of the signal cycle:

$$\phi_j = \sum_{p \in \mathrm{P}_j} \phi_p \quad . \tag{1.16}$$

Lost Time

Driver reactions are not instantaneous, and vehicles take a finite amount of time to accelerate and clear the junction. This implies that a non-negligible share of the signal cycle goes wasted, since demand is not served efficiently during the phase transitions:

- at every phase start, a few seconds pass before vehicles can flow at full capacity, causing a *start-up time loss*;
- at every phase end, sufficient time must be allowed for vehicles to clear the junction before others may safely carry out a conflicting manoeuvre, which represents a *clearance loss*.

The start-up loss may be reduced by helping drivers to react more promptly, e.g. using a pre-green amber light or red count-down timers, which also seem to alleviate the stress of being stuck in a queue ref . The clearance loss may only be mitigated by an accurate choice of signal phase sequence for given traffic conditions or, wherever possible, by appropriate modification of the junction layout, e.g. implementation of protected turn bays.

The total lost time t_j^L then depends on phase design and sequence, which in turn should be tailored to the geometry of junction j in relation to the expected traffic conditions. Each phase contributes its own time losses $t_{j,p}^L$ to the total lost time, which may be quantified by the following relation between the effective phase green and the phase duration:

$$t_p^L = t_p - g_p \quad . \tag{1.17}$$

The total time loss and the total effective green thus account for the whole signal cycle period:

$$t_j^C = t_j^L + \sum_{p \in \mathrm{P}_j} g_p \quad . \tag{1.18}$$

1.3.2 Formulation of the Signal Setting Problem

Conflicting sets of manoeuvres compete for the right of way at road intersections, and the main purpose of signalization is to distribute the junction capacity amongst them.

It follows naturally that the allocation of green time to signal phases is the single most important step in signal setting: the cycle must be allotted according to the relative distribution of demand, lest the junction capacity go wasted and unnecessary queues form on critical approaches.

As far as fixed timing is concerned, optimal allocation of green time is a straightforward process, yet it can be undertaken according to a number of different principles: early studies aimed to develop analytical equations, while modern simulation based methods rely on heuristics to shape the signal setting around a cost function that formalises the chosen signal setting policy. The next sections provide a general formulation of the problem and a few examples of objective implementation through different setting policies.

Lagrangian Formulation

The Signal Setting of junction j can be formulated as an optimisation problem, i.e. to find effective green durations for each phase and cycle time that minimise an objective function while complying with a set of constraints.

A popular choice of cost function may be the average delay at the intersection, given by the weighted average vehicle delay ω_a^D on all lane groups.
Delay on each lane depends according to equation (1.10) on effective green shares, cycle length, and the relevant flows q_a as illustrated in section 1.3.1.

For average delay optimisation of a junction j, consider a well-designed phase sequence P_j ensuring minimal conflicts and time losses. The signal program is then fully characterised by a vector of effective phase green shares $\mathbf{g}_{\mathrm{P}_j} \in \mathbb{R}^{|\mathrm{P}_j|}$ together with the cycle time t_j^C.

The problem takes the following form:

$$\begin{aligned} \underset{\mathbf{g}_{\mathrm{P}_j},\, t_j^C}{\mathbf{min}} \qquad & \omega_j^D = \sum_{a \in \mathrm{A}_j^-} \omega_a^D q_a \\ \text{subject to} \quad & t_j^C - t_j^L = \sum_{p \in \mathrm{P}_j} g_p \\ & g_p \geq \phi_p\, t_j^C \quad \forall p \in \mathrm{P}_j \end{aligned} \tag{1.19}$$

finish copying in lagrangian approach

Webster Optimal Solution

The first and foremost formulation of optimal signal settings to lift the assumption of uniform vehicle arrivals is due to Webster (1958). The approach is based on a queueing system with Poissonian arrivals and a constant service rate equal to the capacity $\gamma\hat{q}$ of the signalised lane group. The average delay given for the steady state case by equation (1.10) was extended to obtain a more complete delay function for random arrivals, with an additional empirical term needed to improve the fit with *experimental* observations.

To simplify the optimisation problem, a reasonable green share allocation policy (widely known as *Equisaturation Policy*) was chosen. This revolves around the idea that an equitable distribution of green share is obtained when all critical manoeuvres operate at the same saturation level: the higher the demand for a manoeuvre *with respect to the capacity* of the relevant infrastructure, the higher the green share allocated to the corresponding signal phase.

Furthermore, Webster worked under the assumptions that no over-saturation occur and average demand *flows* are stable, i.e. and path choices made by road users are in no way a consequence of the signal setting. This assumption was removed by later scholars who tackled the global optimisation signal setting who route choice problem ref (Smith, 1984; Cipriani and Fusco, 2008) .

Under the equisaturation policy, all phase saturation levels at a given junction are equal by definition. The *available green time* can simply be allocated proportionally to the critical flow ratio of each phase:

$$\gamma_p = \frac{\phi_p}{\phi_j} \frac{t_j^C - t_j^L}{t_j^C} \quad \forall p \in \mathrm{P}_j \tag{1.20}$$

which yields meaningful results provided that the junction total flow ratio does not exceed its maximum value of 1 and the cycle time is sufficiently long to amortise the lost time.

The approach can be extended to design for specific (not necessarily even) saturation values for each phase by rearranging equation (1.11) and solving for the green share: this may have practical sense in order to design a higher tolerance to high arrival rates into a given phase e.g. if it is strategically more important to keep queues at a minimum on a certain set of lanes than it is elsewhere.

With this green share setting policy in place, the problem of minimising the average delay is reduced to a single variable function of the cycle length.
The resulting solution for the cycle time that minimises average delay under probabilistic arrivals is rather complex and was approximated it through an empirical formula, widely known as the Webster optimum cycle time:

$$t_j^{C,Webster} = \frac{\frac{3}{2}\, t_j^L + 5}{1 - \phi_j} \quad . \tag{1.21}$$

Notice from equations ref e:mincycle and (1.21) how the cycle time invariably grows with the total flow ratio of the junction. It is also possible to extend ref e:mincycle to get a target saturation level χ_j for the junction:

$$t_j^C(\chi_j) = \frac{t_j^L}{1 - \frac{\phi_j}{\chi_j}} \quad , \tag{1.22}$$

or even a vector $\vec{\chi}_{\mathrm{P}_j}$ of critical saturation level values each phase, as in

$$t_j^C(\vec{\chi}_{\mathrm{P}_j}) = \frac{t_j^L}{1 - \sum_{p \in \mathrm{P}_j} \frac{\phi_j}{\chi_j}} \quad . \tag{1.23}$$

It should be evident that saturation values greater than 1 correspond to *oversaturated* conditions, under which the demand flows are not met with sufficient capacity and queue buildup is inevitable: such traffic conditions require radically different timing approaches. The rule of thumb mentioned in ref HCM (2008) and generally followed in practice is that signals should be timed so that lanes operate at saturation levels below 0.85, allowing sufficient margin to deal efficiently with most possible traffic fluctuations, and discharge any queues within a few signal cycles.

1.4 Signal Coordination

Traffic light coordination between adjacent junctions is an essential aspect of an optimal signalisation plan, with disposition of *green waves* as its most notable and popular feature. Traffic in fact mostly travels along a limited number of main corridors, commonly referred to as *arteries* carrying *arterial traffic*.

It has long been accepted as a reasonable compromise to minimise user discomfort along those, rather than taking on the much more intricate problem of reducing the total network delay. Although, undeniably, being able to drive through a streak of green signals already goes a long way towards improving the quality of a trip from the user point of view, signal coordination chiefly serves the purpose of ensuring an efficient use of the available infrastructure.

[1] By City of San Francisco - Public domain (via Eric Fischer), CC BY-SA 3.0, `https://commons.wikimedia.org/w/index.php?curid=34715929`

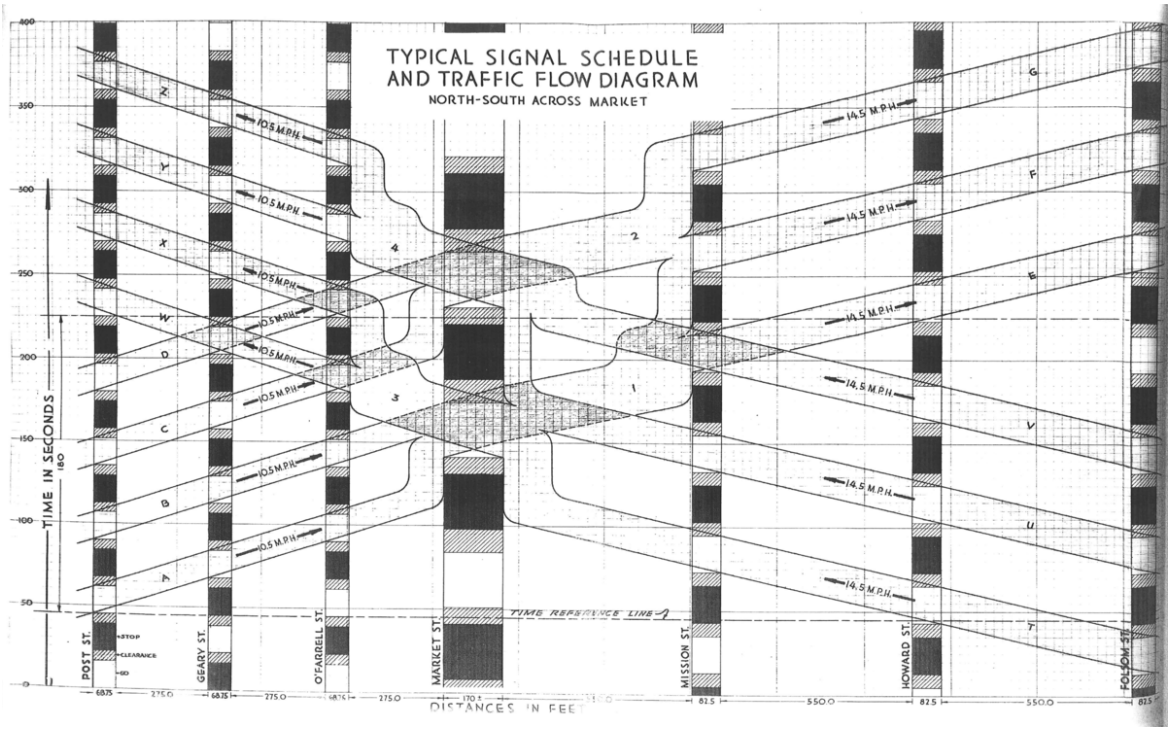

FIGURE 1.4 – Early signal synchronisation along a San Francisco arterial road, circa 1929. Bands A through T represent vehicle platoons [1].

It is in fact of the utmost importance to avoid unnecessary signal-induced delays and stops which could rapidly bring traffic to a grinding halt, even under rather mild conditions which the network could otherwise cope with.

The search for a coordination solution that maximises usability of urban arteries under specific traffic conditions is still mostly carried out offline — as it was for the first attempts at smart arterial signalization, such as the pen-and-paper method portrayed in Figure 1.4. To this end, a wide variety of methods have been the object of intensive research since the early 1980s, ranging from simple analytical approaches to heuristics.

Analytical methods have brought about a number of popular applications which are still in use despite the fact that they mainly apply to low congestion scenarios; more complex methods, which account for demand flows and their propagation along the arterial, can deal with heavy congestion related phenomena, but invariably require a more detailed network model and rely on computationally demanding simulations rather than a closed-form problem formulation. An overview of the most prominent approaches to the signal coordination problem is given in the following sections.

1.4.1 The Traffic Corridor

The fulcrum of signal coordination is the *traffic corridor* (i.e. an arterial road, as defined in the previous section) selected for its strategic relevance. Since the flow on the corridor is supposedly much higher than on its side roads, it is deemed acceptable to concentrate optimisation efforts on the arterial traffic conditions, as improvements will benefit the largest number of road users.

A traffic corridor C may be defined as an *ordered* set of n *connected* arcs:

$$\mathrm{A} \supset \mathrm{C} = \{a_1, a_2, \ldots, a_n\} \qquad \text{with} \quad \begin{cases} a_{i-1} \in \mathrm{A}^-_{a_i} & \forall i > 1 \\ a_{i+1} \in \mathrm{A}^+_{a_i} & \forall i < n \end{cases} \quad . \tag{1.24}$$

Although all nodes along the corridor are, strictly speaking, junctions, it makes sense in this context to define the ordered subset $\mathrm{J_C}$ of the m *signalised* junctions that actually regulate the flow on the corridor. This may be formalised as

$$\bigcup_{a \in C} \{\mathrm{N}^-_a, \mathrm{N}^+_a\} \supset \mathrm{J_C} = \{j_1, j_2, \ldots, j_m\} \quad \text{such that} \quad \forall j \in \mathrm{J_C} \quad \exists y \in \mathrm{P}_j | \{\mathrm{A}^-_y, \mathrm{A}^+_y\} \subset \mathrm{C} \tag{1.25}$$

where it is simply stated that a corridor node is considered a relevant *signalised junction* if features one *signalised* manoeuvre $y \in \mathrm{P}_j$ whose origin and destination lanes $\{\mathrm{A}^-_y, \mathrm{A}^+_y\}$ both lie on the corridor (with the exception of the first node of the corridor, which may be included in $\mathrm{J_C}$ as long as it regulates at least one turn onto the corridor, and the last one if the corridor outflow may be affected by its signal).

Coordination of junctions $\mathrm{J_C}$ is handled by offsetting their local timing instructions (as described at the end of section 1.2), i.e. anticipating or delaying all phase changes rigidly without altering the necessary green shares determined on the basis of average demand flows. The global offset values (with respect to an arbitrary global time reference) of the junctions of corridor C may be represented by a vector $\mathbf{t}^O_\mathrm{C}$.

Furthermore, it is assumed that all junctions of the corridor share the same cycle time, so that in the context of signal coordination the symbol t^C_C refers to all junctions, and may be even used without the subscript C.

1.4.2 Bandwidth Maximisation

In relation to arterial traffic, the concept of *progression bandwidth* emerges as a measure of the quality of a green wave setup along a *corridor* and can be defined as the duration of the time window through which a vehicle may enter the artery and travel its entire length without encountering red lights nor standing queues.

By reducing delays and number of stops along the most critical paths, bandwidth maximisation is a relatively straightforward but effective way to help the system meet user expectations about traffic fluidity, mitigating the stress associated with driving in a congested urban environment. Moreover, this type of signal coordination has proven highly beneficial in reducing the chance of rear end collisions and red signal violations ref (Li, Tarko 2010) as well as pollution levels associated with the hiccupping stop-and-go driving often experienced under poorly coordinated signalisation.

Bandwidth maximisation has been formulated as a Linear Optimisation problem since ref Little et al. (1981) which led to development of the MAXBAND/MULTIBAND series of software solutions. These considered the offsets between junctions as the only decision variables, but provided a computationally viable method for one-way and two-way bandwidth maximisation relying solely on the target travel times between junctions and predetermined signal cycle length and green times. A more efficient solution method was introduced by ... ref Papola and Fusco (1988a) ... expand .

However, relevant discrepancies — dubbed *bandwidth degradation* — were observed between the expected outcome and the real-world performance of the signal plans generated by these

early methods. It is now universally accepted that, as ref Tsay and Lin (1988) amongst many others pointed out, the underlying models were oversimplified and no account was taken of side flows and platoon dispersion.

Proposed extensions of the original method aimed to factor in queue and side flow clearance times, to produce a more realistic bandwidth model for phase offset determination. The analytical relationship between maximal bandwidth and minimum delay problems was finally formulated in Papola and Fusco (2000) ... analytical model, delays as function of the maximal bandwidth and other variables .

At present, offline arterial progression optimisation techniques invariably rely on some formulation of the *bandwidth maximisation problem* (as in the cases illustrated in the next section), which is to say that their common objective is to maximise a *theoretical* traffic throughput, often without much consideration for network performance (and online too! the point is: nobody looks beyond the throughput but we are trying to) .

Make it clear that the slack band concept is unorthodox but original, and implemented in the context of this work.

Mixed Integer Linear Programming Approach

Illustrate De Nunzio 2015 Variable Speed Limit Bandwidth LP optimiser

The Slack Band Approach

Illustrate the slack bandwidth generalisation. The idea of *slack* bandwidth is an answer to the very strict definition of bandwidth given at the beginning of section 1.4.2, according to which only the band running through all junctions counts for something, implying that:

- if one passing phase is particularly short, coordination between longer green phases may be disregarded (discuss fringes in MILP approach) ;
- in bi-directional optimisation, maximisation of the return band may prioritise a very narrow band that *just* makes it through all junctions (possibly degrading the main band significantly) over a very wide band divided in two or more chunks.

To avoid this we consider the total band between any pair of junctions: illustrate algorithm.

1.5 Advanced Offline Signal Planning

Describe heuristic approaches, and complex offline performance optimisation. Anticipate how the present work will build and improve upon those. TO REVIEW.

The simple signal setting problems presented so far are quasi-convex, but more realistic traffic models that include and quantify global performance indicators such as total delay introduce an inherent non convexity, better addressed with the aid of heuristic methods.

With the increase in computing power availability, metaheuristics have seen a substantial rise in popularity as means to overcome the inherent limitations of analytical formulations: heuristic approaches to this class of problems involve the generation of a large — yet manageable, compared to the dimensions of the search space — number of candidate timing

solutions, the effects of which are then simulated to evaluate their fitness. At each iteration, a variety of methods ranging from Genetic Algorithms to Simulated Annealing and Particle Swarm Optimisation can then be used to modify and combine the most successful solutions into a new set of candidates.

Such methods are particularly suited for solving obscure problems as they require no attempt to establish an explicit correlation between the control variables and the desired outcome. Rather, they rely on the assumption that if any relevant phenomena can be modelled with sufficient accuracy and a performance index can describe the degree of achievement of the optimization objectives, then the system can be made to naturally evolve towards an optimal solution.

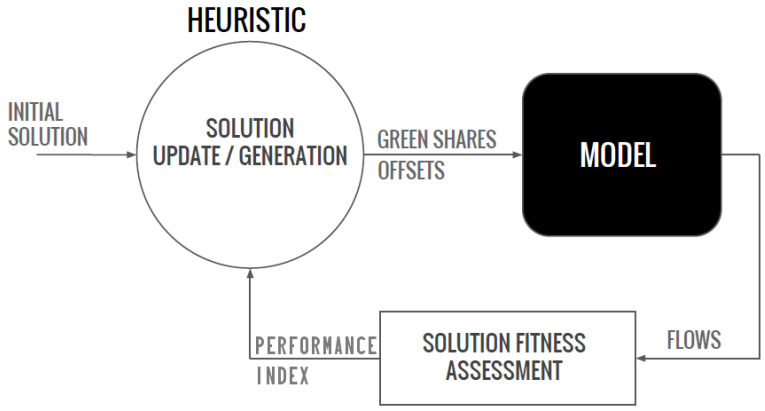

FIGURE 1.5 – Conceptual information flow in a heuristic approach to signal optimisation

It is therefore obvious that the model used to assess the fitness of candidate solutions should represent a sensible trade-off between speed and completeness: the real-world performance will inevitably be disappointing if the optimisation could not account for relevant traffic phenomena that were simplified out of the solution assessment, while on the other hand the need to evaluate huge numbers of candidate solutions calls for a lean and fast method to predict the outcome of a given timing plan. Heuristics that depend heavily on the choice of initial conditions often use maximum bandwidth solutions as starting point in the search for minimum total delay, to shave off convergence time and increase the quality and applicability of solutions.

This approach has been taken most notably by the Transport Research Laboratory, the UK based institution that since Robertson (1969) has been developing the TRAffic Network StudY Tool, which was born as a software tool to minimise stops along arterial roads while accounting for reasonably realistic vehicle behaviour, and was gradually extended to model ever more complex phenomena. Today, TRANSYT can handle pedestrian flows, optimise green shares as well as junction offsets and include actuated signals, all the while monitoring a custom set of network-wide performance indicators that can implement whatever policy the traffic administration desires. The optimisation relies on the availability of a complete transportation network model, possibly including detailed junction geometry. A range of search algorithms can be used to explore complex timing solutions, which are then evaluated using either micro– or macrosimulation models. Earliest version of Transyt implemented a simple hill climbing algorithm that explored the non convex performance function by executing a predetermined set of short and long steps to vary each control variable in both directions alternatively. At each step, the changed value of the control variable is kept if it improved

the performance index. Park et al. (1999) introduced a traffic signal optimization program for oversaturated intersections consisting of two modules: a genetic algorithm optimizer and mesoscopic simulator. Colombaroni et al. (2009) devised a solution procedure that first applies a genetic algorithm and then a hill climbing algorithm for local adjustments. The fitness function is evaluated by means of a traffic model that computes platoon progression along the links, their combination and possible queuing at nodes through analytical delay formulations. Colombaroni et al. (2010) extended the model to design optimal signal settings of a synchronised artery with predetermined rules for dynamic bus priority. A visual example is given in Figure 7 of the genetic algorithm representation of signal settings as a chromosome population.

Metaheuristics often see applications in traffic signal engineering that reach beyond ordinary signal planning, and have more than once played an important role in research by aiding the formalisation of less intuitive correlations between signal settings and traffic behaviour. Gentile and Tiddi (2009) use a Genetic Algorithm to venture out into the yet uncharted territory of arterial synchronisation under heavy congestion and queue spillback. To predict the outcome of candidate signal plans, the heuristic method relies on the General Link Transmission Model (Gentile 2015), which implements the Kinematic Wave Theory to allow accurate simulation of traffic dynamics and model physical blockage of links, while requiring sufficiently short computation times to deal with the very large number of solutions to be evaluated. The optimisation reveals a crucial difference between subcritical and supercritical flow conditions: while in the former case the optimal green wave is led as usual by the flow velocity, the same approach proves completely ineffective under supercritical conditions, which oppositely demand that the backwards propagating jam wave speed should set the pace of upstream signals, to ensure that the residual capacity of saturated links is fully exploited.

It must be noted that the level of detail taken into account when using metaheuristics comes at a heavy cost in terms of computation speed, which restricts (has so far restricted! reformulate around present work) the functionality of this type of software to that of advanced yet offline development tools. As long as ordinarily accessible computing power remains insufficient for true real time functionality, advanced optimization suites are staying on top of the game by attempting to streamline the interactions between the development environment and the street-level equipment, e.g. providing offline optimisation based on real time readings and quick and simple deployment of new plans.

Such efforts are driving this type of software towards a sort of mid-term, day-to-day real time adaptivity which, possibly in combination with a true real-time program selection logic may well prove to be an effective approach to slowly and steadily improve the signalization of large urban networks (meh, cut) .

Chapter 2

Smart Signals

This chapter presents adaptive signalisation approaches. TO REVIEW. .

Over the years, many attempts have been made to render the signalisation system of urban networks capable of reacting autonomously to the traffic conditions, to address the mutable nature of demand. In this context, the term optimisation is used in its broader sense of choice of the best option, whether this is picked out of a set of previously planned solutions, tailored on-the-fly onto the current traffic conditions, or simply the result of a sequence of best possible actions evaluated individually: the main features of each class of very different approaches will be illustrated in the following sections.

The one thing that all responsive traffic control systems have is the need to perceive the traffic state on the network by means of detectors. The type and quantity of information required for different optimisation approaches may vary, but in the end it always boils down to one, or a combination, of the following quantities:

flow : the number of vehicles crossing a road section in a given amount of time

occupancy : the percentage of time the sensor spends occupied by a vehicle

velocity : the average speed of the vehicles through a road section

2.1 Adaptive Signalisation

Why do we need adaptive signals?

- responsive to short term fluctuations, can allocate green efficiently
- reactive to unexpected flows deviating from statistical forecast
- reactive to special events and accidents

2.2 Traffic Actuated Signals

2.2.1 Automatic Plan Selection

relevant because allows very careful optimisation of a scenario to devise a general plan that makes sense under even extreme circumstances. copy and review

2.2.2 Traffic Actuated Control

relevant because more efficient and reliable than any offline optimisation but currently hardly ever does much more than increasing throughput. it could benefit from a more network conscious perspective. copy and review

2.3 Real Time Signal Plan Generation

copy and review

Real time optimisers that perform plan generation are a class of proactive signal control systems that, based on current traffic conditions, seek to develop an optimal plan to apply in the immediate future, either from first principles or by continuous update of an existing pre-timed plan. While each plan is played out, the system gathers information to make the next.

This mode of operation is often referred to as rolling horizon, and in order for the system to respond effectively (i.e. to capture and react to rapid changes in traffic conditions) the rolling horizon time step should be reasonably short, which imposes austere constraints on the optimisation methods. Some real-time optimisers with a very short rolling horizon step update the signalisation plan at every cycle, so that their behaviour may appear indistinguishable from that of an actuated controller.

It is important however to understand the clear conceptual difference between the two: actuated controllers perform second-by-second decisions about the best action to perform instantly, while the systems considered in this section plan ahead, producing fully featured signal plans made of cycle times, offsets and green shares deemed optimal for dealing with the traffic conditions observed.

2.3.1 Incremental Analytical Optimisation

The most prominent member of this category is the *Split Cycle and Offset Optimisation Technique* developed for research purposes in Glasgow, and first applied there in 1975 under the acronym SCOOT by which it is now popular all over the world, counting over a hundred active installations. It revolves around a centralised control unit which generates plans based on a real-time traffic snapshot gathered from detectors. The signalisation plans are continuously updated, with a frequency in the order of one to three cycle times, and may concern the entire network or *regions* thereof which are expected to feature homogeneous traffic conditions.
copy and review

2.3.2 Linear Quadratic Optimal Control

copy and review

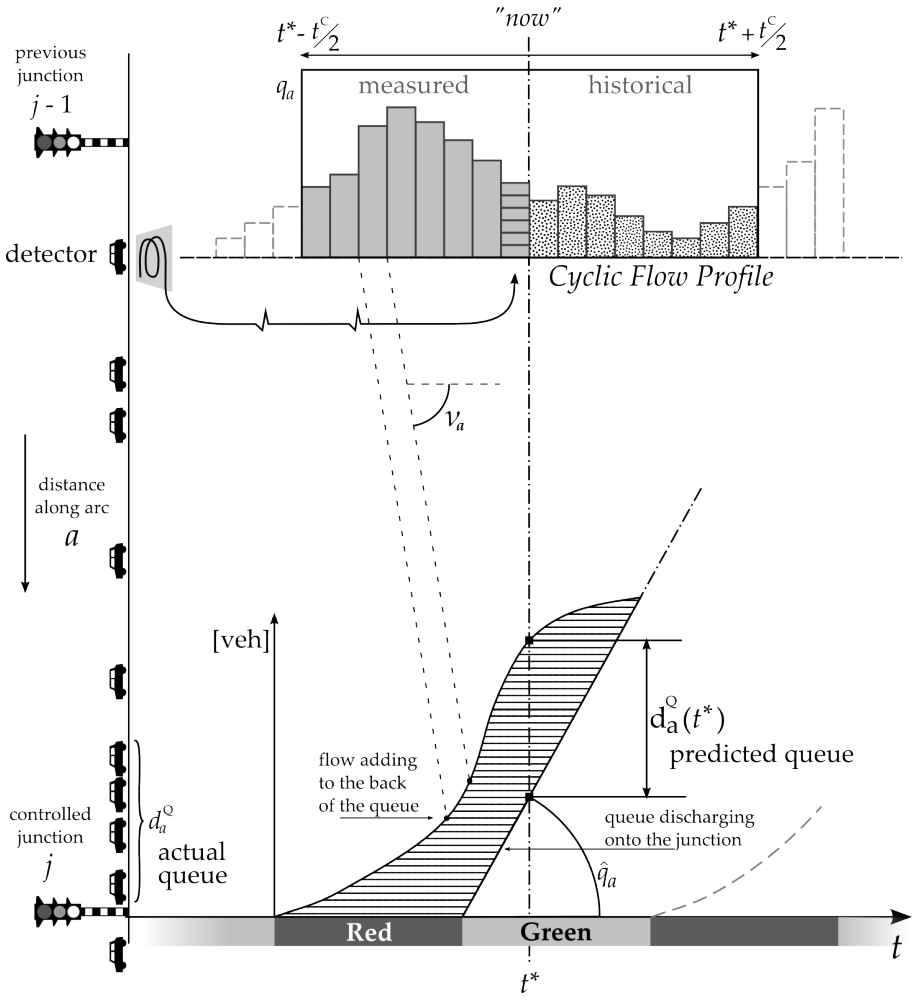

FIGURE 2.1 – SCOOT Cyclic Flow Profiles and queue prediction: detector readings are used to update the flow profile, which is integrated to predict the queue forming at the downstream junction during the red phase. The information may prompt the system to anticipate or delay a phase change in order to accommodate the measured demand.

2.3.3 Traffic Gating

extremely relevant, it is the only approach known so far that attempts to prevent congestion by doing something counter intuitive like delaying flows upstream

Feedback Traffic Gating ref (Ekbatani 2012) is a form of actuated signal control aiming to prevent oversaturation of critical portions of the network by holding back the incoming traffic flows — using deliberately exaggerated red phases — rather than attempting to deal with the flows already trapped in a congested area. In these respects, it constitutes a simple yet innovative method to induce more efficient utilisation of the existing infrastructure, and an answer to the patent performance degradation that currently feasible real-time optimisation solutions face under saturated conditions. copy and review

Network Fundamental Diagram Formulation

Feedback Controller Design

Chapter 3

Modelling, Simulation and Optimisation Tools

This chapter presents the relevant elements of the real time traffic management framework in which the work was conducted, illustrating the most interesting features in light of their role in the optimisation. An introduction to the basic principles of the Genetic Algorithm completes the inventory of the tools used to bring together the optimisation presented in Chapter 4.

3.1 The Optima Framework

Illustrate components: model building and calibration, data model, simulation, rolling horizon forecast, visualisation, statistical data, real time data harmoniser, events interface, traffic control, field actuators.

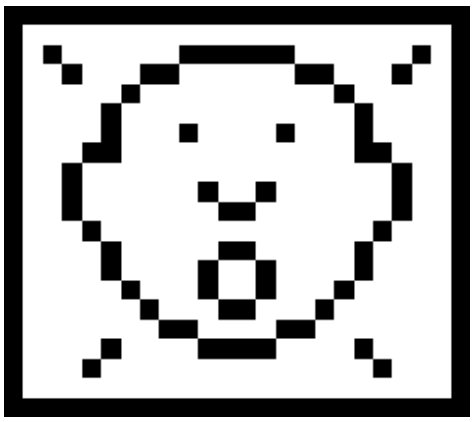

FIGURE 3.1 – the Optima framework for real time traffic management

3.2 TRE simulation engine

The proposed optimisation method relies on the macrosimulation engine known as TRE, based on the eponymous Dynamic User Equilibrium assignment algorithm. TRE lies at the

core of the Optima traffic management software suite, developed in Rome by PTV SISTeMa[1] and running in traffic control centres around the globe.
This section aims to illustrate its fundamental principles of operation, in order to clarify how they might affect the optimisation and better understand the role of the simulation engine within the architecture.

3.2.1 Continuous Dynamic Traffic Assignment

The general idea of *Traffic Assignment* is rather intuitive: it is the modelling of the interaction between *supply*, i.e. the roads, infrastructures and public transport options; and the *demand* for mobility, i.e. people that need to travel using a choice of the available resources.

Since supply is limited, its availability and performance are a consequence of the choices made by users, which in turn are affected by the perceived discomfort of travelling across the network in the state it actually is: to predict with any plausibility the way in which traffic will spread across the network, it is necessary to resolve this reciprocal influence between supply and demand.

Of the many approaches proposed to this end throughout the history of transport research, the most successful are based on the *Selfish User Equilibrium* principle first stated by Wardrop [1952]. This follows from the simple and sound behavioural assumption that every user will choose the route and mode of transport which are best for them, and implies that the most reasonably foreseeable traffic scenario is that in which no user would benefit from making a different choice: hence the notion of user *equilibrium*.

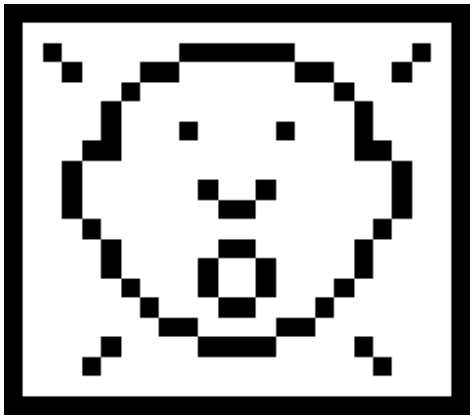

FIGURE 3.2 – General Dynamic Traffic Assignment [Yperman, 2007] p22

Even in the simplest possible *static* case, with steady demand and travel times only dependent on user choices, the equilibrium point must be found by an iterative process as illustrated in Figure 3.2. Thus, at each iteration

1 . demand is routed through the network according to the arc costs,
route flows are calculated;

2 . flows are assigned to the relevant arcs, costs are updated to account for congestion
and checked against convergence criteria;

3a. until convergence is obtained, costs are fed back into a new demand routing (step 1);

[1] ITS spinoff of Sapienza University, and partner in the present work ???

3b. when arc costs converge, it means users are confirming the route choices made in the previous iteration: the *user equilibrium* is satisfied, and the last route flows and costs calculated are the best estimate of the outcome from the given demand and supply.

In reality, demand is hardly constant throughout the day and congestion occurs as a consequence of the *history* of the system; therefore any real-world application must account for the fact that travel times and user choices evolve *dynamically* over time.

A *Dynamic Traffic Assignment* (DTA) model allows to determine the dynamic interaction of supply and demand to predict the evolution of traffic conditions over any length of time, conceptually without further complication beyond the addition of the temporal dimension.

The user equilibrium condition can still be found via the mechanism illustrated in Figure 3.2, working from the knowledge of demand and supply; with the difference that demand and user choices will be time dependent, and flows will propagate in space *and time* so that arc costs become dynamic too. Traffic is considered to be a continuum, both as far as vehicle movements and trip-maker decisions are concerned; the equilibrium is then found between time profiles of arc flows and costs.

Trip planning and route choice models are extremely relevant to the accuracy of a DTA and to the computational effort involved, but in the general DTA framework they are quite independent of each other and of the supply model.

The representation of traffic as it propagates and interacts with the network and signals, and all related phenomena within the scope of the present study, fall upon the Dynamic Network Loading model. The most suitable macroscopic model for the task at hand is the General Link Transmission Model, which will be analysed in further detail over the next few sections.

3.2.2 General Link Transmission Model

The *General Link Transmission Model*, henceforth referred to as GLTM, is a model for continuous dynamic network loading: it can be used to determine time-dependent link flows $q_{a,\tau}$ and travel times $t^t_{a,\tau}$ given the time-dependent route flows.

It is built upon the representation of traffic as a partially compressible one-dimensional fluid flowing through the network according to the principles of *Kinematic Wave Theory* (KWT), as developed independently by Lighthill and Whitham [1955] and Richards [1956].

Its origins can be traced back to the *Cell Transmission Model* first presented in relation to highway traffic by Daganzo [1994] and shortly after applied to network traffic in [Daganzo, 1995]. CTM was the first dynamic traffic representation based on hydrodynamic theory, and borrowed heavily from that field, as is most evident from the cell-based space discretisation of the road network adopted directly from computational fluid dynamics.

The need for cell discretisation was eliminated in the *Link Transmission Model* presented by Yperman, Logghe, and Immers [2005]. This innovative approach allowed dynamic network loading of large scale networks using a computationally efficient algorithm that only required calculations at intersection nodes, while solving for traffic propagation along whole links using kinematic wave theory: this allowed to do away with a significant complexity factor while still accurately modelling local flow restrictions and junction delays.

The original LTM, presented in full detail in Yperman [2007], rather simplified the wave propagation problem relying on the *simplified* kinematic wave theory proposed by [Newell,

1993], whereby only two possible wave propagation speeds are contemplated: a forwards one for free-flowing traffic, and one for the congested flow states to propagate backwards. This is a considerable approximation, as the relation between vehicle density, speed and the resulting flow is rather more complex in reality: the instrument provided by kinematic wave theory to express such relations in general is the Density-Flow Fundamental Diagram, illustrated in section 3.2.3.

While in truth the work of Yperman already improved on the simplified KWT approach to include any piecewise linear fundamental diagram, the GLTM presented in Gentile et al. [2010] was developed to extend the LTM formulation to any concave fundamental diagram, considerably improving the accuracy in representing delays due to congestion. The GLTM also uses time-varying capacity adjustments at nodes to accurately model conflicts at intersections and the so called *spillback* of traffic states from downstream links to the relevant upstream ones.

The features of the supply model described in the next sections, together with the computational efficiency and the possibility to perform a DTA in high temporal resolution, make the General Link Transmission Model an optimal candidate for the present application.

3.2.3 Link Model

In the GLTM, traffic propagates along links according to kinematic wave theory. As stated in section 1.1, links are assimilated to weighted arcs of a directed graph, and as such are one-dimensional, one-directional and homogeneous along their length, stretching between locations x^0 (the tail node) and $x^1 = x^0 + \ell$ (the head node): the actual link shape is inconsequential. As far as the arc model is concerned, there is no need to disambiguate arcs since they exist and are processed independently: arc subscripts can be dropped for ease of reading but are to be implied on all relevant quantities henceforth.

The traffic state at a specific location $x \in \left[x^0, x^1\right]$ along a link is characterised by three macroscopic variables:

flow q_x : vehicles through the link section per unit time;

density k_x : average number of vehicles per unit length;

speed v_x : average distance covered per unit time.

As is evident from their dimensions, only two of these quantities can be independent, and if two are known the third may be readily calculated using the relationship

$$v = \frac{q}{k} \quad . \tag{3.1}$$

The idea that vehicle density and speed can be completely independent, however mathematically sound, does not seem practically plausible. Kinematic wave theory provides a device for solving this contradiction as illustrated in the following section.

Fundamental Diagram

Kinematic wave theory assumes a functional relation between traffic density and flow, known as the *Fundamental Diagram* of traffic flow. It approximates the changes in the average behaviour of drivers as the road gets more crowded, and may take several forms, but invariably

follows from the properties of the road, e.g. width, slope or parking. As such it is itself, conceptually, a property of the link, although it could also be made to depend on environmental factors and driver behaviour, or be specific to a particular class of vehicles.

A generic fundamental diagram expresses the relationship between flow and density under *stationary* traffic conditions, i.e. it is derived as an equilibrium condition between flow speed and available space taking the general form

$$q = f(k) \tag{3.2}$$

which may be represented on a Density-Flow graph like the ones shown in Figure 3.3.

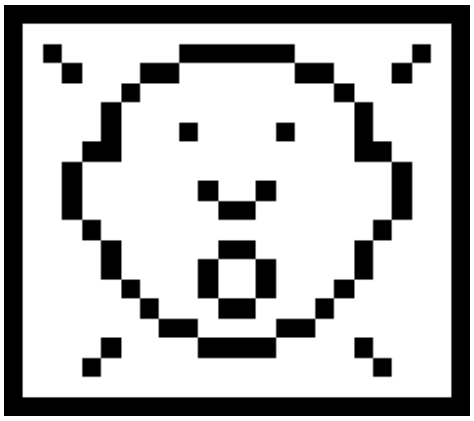

FIGURE 3.3 – Fundamental Diagram of a link, representing the functional relation between vehicular density and speed, resulting in different flow values for different congestion levels. Triangular form (left) and ??? form (right) are similarly shaped around the critical density value $\hat{k}$, the jam density k_{jam} and the free flow speed v^0.

The shape of a fundamental diagram reflects different assumptions about traffic flow dynamics, but there are a few key features that are shared by all formulations:

- when density approaches *zero* the speed approaches the maximum value attainable on the link, i.e. the *free flow* speed v^0, but the flow tends to zero;
- maximum flow occurs at the *critical density* $\hat{k}$ also referred to as the link *capacity*;
- beyond capacity, further increase in vehicular density induces a speed penalty that causes the flow to decrease;
- when vehicles reach the *jam density* k_{jam} they are packed as closely as possible, and come to a standstill;
- for any flow state on the $k - q$ curve, the speed is given by the slope of the line connecting it to the origin;
- the rising branch diagram (i.e. to the left of $\hat{k}$) represents free flowing states, the descending branch represents congested states.

In a simple triangular diagram like the one shown in Figure 3.3 (left) the speed is assumed constant at its maximum value for all subcritical states, while above capacity it decreases linearly with density. More subtle modelling may yield a diagram shape more similar to Figure 3.3 (right), where the speed is shown to decrease even in subcritical conditions as the road gets more crowded due to the natural variance in driving speed which, as more

vehicles become involved, yields a higher chance of having a slow vehicle delaying all the others *(subcritical spacing)*.

In both cases it is assumed that as density increases, the available space becomes insufficient to maintain safe distances between vehicles, causing drivers to slow down *(hypercritical spacing)*. analysis of shapes given in ref tiddi

If a model is to rely on the fundamental diagram to hold for non-stationary traffic as well, it must allow vehicles to change speed instantly with infinite acceleration, as is the case with GLTM and in general with first-order implementations of KWT.
Higher order traffic phenomena such as the emergence of stop-and-go waves along the link, or fundamental diagram hysteresis (due to traffic states evolving asimmetrically when leading up to congestion or recovering from it) are knowingly neglected.

Traffic State Propagation

To understand how traffic states propagate on links, consider the *cumulative flow* $N(x,t)$, i.e. the number of vehicles that have passed location x along a link before time t.

Assuming that vehicle conservation is respected along the link, i.e. that no vehicle is created or destroyed between the tail and the head node, the trajectory of the n^{th} vehicle to enter the arc can be traced on a time-space diagram as the locus of points for which $N(x,t) = n$ as shown in Figure 3.4.

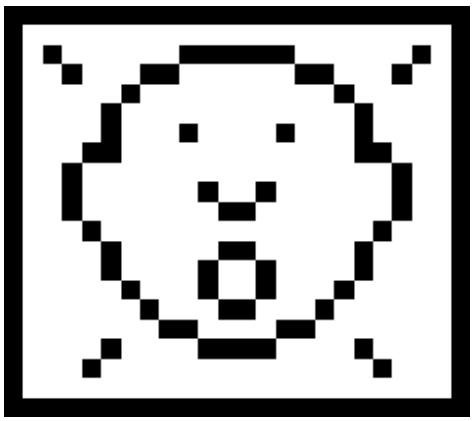

FIGURE 3.4 – vehicle trajectories

The cumulative function $N(x,t)$ is clearly discontinuous in both time and space, but it is possible to consider a smooth approximation that is differentiable in either direction without altering the essence of the phenomenon. Flow and density values at a given location and time can then be expressed as the partial derivatives

$$q(x,t) = \frac{\partial N(x,t)}{\partial t} \quad , \tag{3.3}$$

$$k(x,t) = \frac{-\partial N(x,t)}{\partial x} \quad , \tag{3.4}$$

the latter requiring a sign change simply because density is defined positive but the cumulative decreases along the positive spatial direction.

derive wave propagation speed, conservation, Newell Luke MP, shockwaves

OR MAYBE cut back on theory refer to literature and stick with the ESSENTIALS

3.2.4 Node Model

illustrate node model, specifying how IT is in charge of the implementation of signals

3.3 GLTM as Flow Simulation

The network loading and flow propagation model is central to the optimisation process proposed in this work. So far, its position in the Dynamic Traffic Assignment has been clarified, but it may be useful to recapitulate and formalise what its input and output are as a stand-alone *flow simulator* component; before proceeding to Section 3.4 where its role in relation to the Genetic Algorithm will be clarified. These are summarised in Figure 3.5 and presented in more detail in the following sections.

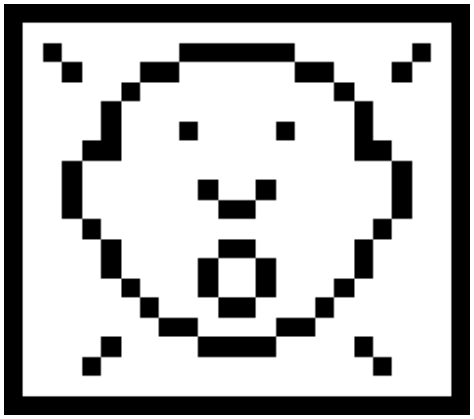

FIGURE 3.5 – Dynamic Network Loading Input and Output

3.3.1 Simulation Input

The General Link Transmission Model operates on the basis of

splitting rates resulting from the aggregation of the dynamic route flows, used by the node model to distribute the outflow of an arc to its forward star;

origin flows representing the vehicles *injected* at specific network locations at every simulation interval, according to the demand data.

It is obviously important that the input data cover the entire span of the simulation, if realistic results are to be obtained. However, the time resolution of the input data is irrelevant and the algorithm can operate with constant values as well as weighted averages where the time intervals do not correspond; in fact, it is robust even with respect to incomplete input, since it may split flows based on the relative capacity of downstream arcs, and if demand flows are unknown they will simply be assumed to be zero.

3.3.2 Real Time Data Integration

The GLTM implemented in TRE can draw real-time corrections from the OPTIMA framework, which are based on harmonised data coming from a variety of public and private sources (loops, cameras, floating car data etc.) which are integrated into the simulation as:

capacity corrections when an accident, road closure or other modification to the supply is broadcast by the authorities or inferred automatically, and the fundamental diagram of the relevant arcs is updated;

speed corrections when the speed is measured or inferred, and either the fundamental diagram is updated to match the traffic state or a flow correction is applied;

flow corrections when a real flow value (often the outflow from an arc) is available and the simulated value overwritten.

These corrections are all applied in the inner loops of Figure 3.5, during the simulation intervals for which they are relevant. Their effect is then propagated in time and space, increasing the fidelity of the simulation to the real world traffic conditions.

3.3.3 Simulation Output

Strictly speaking, the results of the Dynamic Network Loading are, for every arc, the cumulative inflow and outflow profiles (namely F and E), the cumulate number of spaces available and vehicles that reached the head of each arc (respectively G and H) defined in Sections **??** and 3.2.4.

These values refer to *instants* of the simulation span, and as illustrated in Figure 3.5 can be readily used to obtain interval averages of the following quantities:

q_a **: flow** onto the arc during each interval in vehicles per unit time;

t_a^t **: travel time** that users entering during each interval will spend on the arc;

ω_a^Q **: queue** length given as average share of the arc length;

ω_a^n **: total vehicles** on the arc during the interval.

3.3.4 Optimisation Corridor

The present work aims to optimise signal timings in relation to the performance of what is usually called a *Traffic Corridor* or *Arterial Road*, referring to a stretch of road designed or happening to carry particularly high volumes of traffic.
While the concept is not strictly related to urban traffic, it is in the urban environment that traffic corridors most often suffer significant performance degradation due to congestion, aggravated by the numerous intersections with other busy roads where consistent traffic flows compete for the right of way and must be regulated by traffic lights.

The proposed optimisation method revolves around an *Optimisation Corridor* object that essentially implements the formalisation illustrated in Section 1.4.1. It consists of an *ordered set of links* connected head-to-tail, and may run through any number of signalised intersections: the problem size is then determined exactly, since the task at hand is simply to optimise signal coordination and each junction has a predetermined program that can only be offset in time.

This definition of corridor blends seamlessly into the Optima network model as well as in TRE result computation, and allows relevant key performance indicators to be calculated from continuous network loading results, i.e. the arc profiles just introduced in section 3.3.3: the process is fully detailed in Chapter 5 where performance indicators are discussed.

The corridor object represents the interface between the optimisation and simulation processes, and allows their separation (as may be more clear from Figure 4.2), leaving the possibility to exploit the work done in this context e.g. with different optimisation methods. Whatever the optimisation procedure, the corridor defines an additional input and output for the DNL. For a corridor with n arcs and m signalised junctions (see section 1.4.1):

- the **input** is a vector of m offset values, which affect the turn capacities used by the Node Model at the relevant junctions, altering the simulated flow propagation;
- the **output** is a vector of performance indices calculated for the n arcs of the corridor, which can be aggregated into global corridor performance indices.

Although for the rest of this dissertation only one corridor will be considered, the application might easily be scaled to multiple corridors or extended to sub-networks: such efforts are beyond the scope of this experiment but their potential is discussed in **??** alongside other scalability considerations.

Finally, it should be noted that links that are not strictly part of the corridor are *not* factored into the KPI computation. Some may be considered relevant, e.g. the inroads to the corridor; however, it is far from straightforward to *automatically* determine which links should be included based solely on the corridor definition, and in general there is no guarantee that the network model should be constructed in such a way as to render it possible at all. Although *in principle* some consideration for the consequences of choices made on the corridor on the neighbouring roads may help make better decisions, this would require preprocessing of the network and the associated complications are deemed unnecessary given the current task, but will be discussed in Chapter **??**.

Return Corridor Definition

The return corridor cannot simply be defined as the sequence of links traversing the same nodes in reverse order: there is no guarantee that for any pair of subsequent nodes representing the tail and head of a given link there should exist another link joining them in the opposite direction (the network is a directed graph).

Furthermore, the two directions of a traffic corridor may well be modelled as completely disjoint sets of arcs, sharing no nodes between them.

The present approach can handle two-way optimisation without loss of generality in this respect: it is sufficient to define the return corridor in the exact same way as the primary direction, and to indicate it as *return* direction along with a weight coefficient, which can be used to scale KPI values to reflect its importance with respect to the main direction.

If the junctions traversed by the return corridor are handled by the same set of controllers as the main, the problem size remains the same and the extra computation time required to calculate the relevant KPI is negligible.
If more controllers are involved, they can be ignored (which makes little sense unless they actually cannot be controlled and adjusted remotely) or included in the optimisation, which will increase the solution space size and the time required to explore it.

3.4 Genetic Algorithm

A Genetic Algorithm is an evolutionary computation technique that explores a solution space by mimicking a process of evolution by natural selection. It is particularly suitable for

heuristic optimisation approaches as it does not rely on *a priori* knowledge of the problem.

explain right away how this applies to our case

Given a generic process or single valued function of n arguments, the algorithm identifies candidate solutions as *individuals*, each characterised by a *chromosome* $\mathbf{s}$ which is nothing but a vector of n viable input values to the process. The candidate solution chromosomes are then fed through the process or function to determine the fitness f of the corresponding individual:

$$\omega_i = \phi(\mathbf{s}_i) \qquad \text{with} \quad \mathbf{s}_i = \left\{ s_i^1, s_i^2, \dots s_i^n \right\} \tag{3.5}$$

where the subscript $i \in I$ identifies an individual member of the population, i.e. a specific vector among the pool of candidate solutions whose components are referred to as *genes*.

The population may initially be randomised or otherwise generated. Through the fundamental operations of *selection, crossover* and *mutation* (detailed in the next few sections) the best individuals are allowed to live on and pass their genes to their successors, which gradually replace the lower ranking individuals.

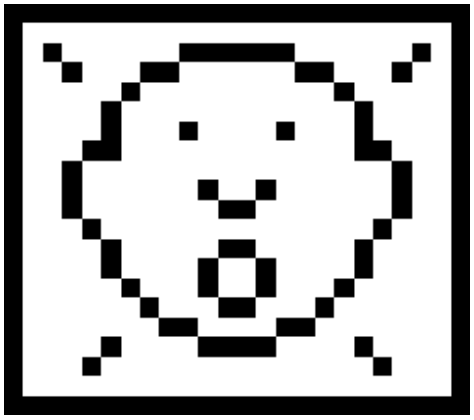

FIGURE 3.6 – The Genetic Algorithm: with each cycle, a new *generation* carrying the most successful traits of the previous one supplants the least successful individuals.

Through iterations of this mechanism, illustrated in Figure 3.6, the population undergoes an evolutionary process whereby all individuals become (on average) better suited for the process considered: there is no guarantee that a global optimum will be found, but good-enough solutions for practical applications can be obtained in relatively few generations.

There is no generalised consensus regarding the optimal population size in relation to the problem size; if anything, researchers agree that no implementation of the Genetic Algorithm can be expected to work equally well with different problem types, and that some trial and error is always required in practice: [Eberhart and Shi, 1998] provide an insightful analysis of the matter.

This work is no exception, and a study of the algorithm performance with different configurations is presented in the Results chapter. The next sections are dedicated to the formalisation of the genetic algorithm operators and will illustrate in more detail some of the design choices made for the current GA implementation.

3.4.1 Evolutionary Operators

Selection

Selection is the process whereby the survival and breeding chances of an individual are determined based on its fitness. Selection for survival is necessary because some individuals must be eliminated from the pool to make room for the new generation, and is generally applied before the other for obvious reasons, although this is not strictly necessary. Selection for breeding further enforces the inheritance of the *best* genes to the new generations. They are applied in this order in the GA implemented for this work, and will be presented accordingly.

Considering the population I_g at generation g, the selection process κ determines the subset I_g^* that survives to maturity based on the individual fitness values $\mathbf{f}$

$$I_g^* = \kappa\left(I_g, \mathbf{f}\right) \quad , \tag{3.6}$$

then the the breeding chance of each surviving individual $\mathbf{p}$ may be calculated by an independent process β

$$\mathbf{p}^\beta = \beta\left(I_g^*, \mathbf{f}^*\right) \tag{3.7}$$

where all terms have the same cardinality equal to the number of individuals in I_g^*.

In this instance, the selection function ϕ takes the form of a dynamic step function allowing an arbitrary percentile of the k fittest individuals to make it to adulthood; the breeding chances are then determined by a linear function of the individual ranking, which can be adjusted via the ratio p_1/p_k, expressing how much more likely the top individual is to breed with respect to the least fit surviving one.

Crossover

The combination of two individuals to generate a new one is inspired by the naturally occurring event of genes *crossing over* between chromosomes during meiosis in sexually reproducing organisms. Given two chromosomes with n genes, and assuming only one random crossing-over locus x, the crossover operator ξ used to produce a new one may be formalised as follows:

$$\xi\left(\mathbf{s}_1, \mathbf{s}_2\right) = \left\{s_1^1, \ldots, s_1^{x-1}\right\} \cup \left\{s_2^x, \ldots, s_1^n\right\} \qquad \text{with} \quad x \in [1, n+1] \quad , \tag{3.8}$$

meaning that the resulting chromosome inherits the genes from one parent up to a random position, and the rest from the second parent. The same principle may be intuitively extended to multiple random crossover loci.

Mutation

Mutation is the process whereby a random gene on a solution chromosome changes value. This introduces variability in the population that is not directly related with fitness: on one side, this is beneficial as it prevents to some degree that a sub-optimal solution should take over the entire population.

3.4.2 Other Devices

3.4.3 Initial Population Seeding with Slack Bandwidth

The speed of convergence of the Genetic Algorithm is strongly influenced by the initial population. Theoretically, an infinitely large random population would contain the global optimum right from the first iteration, but with any manageable number of candidate solutions the chance of having a randomly generated solution performing well becomes very slim.

Depending on the problem size, there is a chance that a single random solution may be rather near an optimum, but in a randomly generated gene pool it will struggle to find a worthy partner, and most crossover operations will result in low fitness individuals with the exception of those which happen to inherit most genes from the successful one (which increases the average population fitness but almost exclusively through loss of diversity). This leads to slow performance improvements over the first iterations, and rather unpredictable results in the long run.

By priming the algorithm with a population of selected individuals that can be reasonably expected to perform well, obtained by some fast and cheap approximation of the problem, it is possible to greatly improve the initial performance; the downside being the risk of *driving* evolution too hard into a local optimum.

Based on the results presented in the Results chapter, section 7.1.2, it was determined that the best results for the problem at hand could be obtained by priming the population with solutions derived from the Slack Bandwidth approach presented in section 1.4.2.

To maximise the chance of obtaining good solutions right from the first generation while reducing the risk of driving the algorithm into a local optimum, the maximum bandwidth solution was cloned into 50% of the initial population while applying small random mutations and a constant shift to all values, so that the *relative* offsets (which can reasonably be expected to be nearly correct from the geometric method) could be phased over the entire signal cycle. This introduces a certain degree of diversity in the population, complemented by the remaining 50% of the initial population generated randomly.

Time Dependent Weighting of Cost Functions

Chapter 4

A Real-Time Forecast-Based Optimiser

The foremost aim of this work is to exploit the versatility and speed of advanced macroscopic traffic simulation to bring forth a heuristic approach capable of improving signal plans *in real time*. In this, it represents an attempt to bring together the best of most signal setting approaches described so far:

adaptive aiming at real-time operation, it is hoped it might guarantee a degree of adaptivity so far only expected of adaptive online signal setting approaches described in ref?? using much simpler models

accurate comprehensive model including real time data and less dependant on single sensors (see balmanual in case it says anything interesting about the stability of optima

heuristic using heuristics avoid the simplifications required to formulate analytical approaches without compromising on accuracy in evaluating the outcomes

gating by evaluating forecast traffic conditions over a look-ahead window, it should behave more like a feedback controller, accounting at least for the short-term consequences of its decisions. it should stay away from greedy solutions that may maximise the immediate efficiency of junctions while disregarding even the short-term effects which might include an increase in congestion

This chapter presents the approach in detail. anticipate sections better

4.1 Heuristic Offset Optimisation

Describe the task at hand, in terms of inputs and goals of the optimiser strictly.

4.1.1 The Look-Ahead Window

It could be done static, like everybody else, but we are in real time and we want to include short term effects (long term is still out of reach).

The size of the time window is limited by computing power. Resolution cannot be sacrificed or the fundamental events get averaged out.

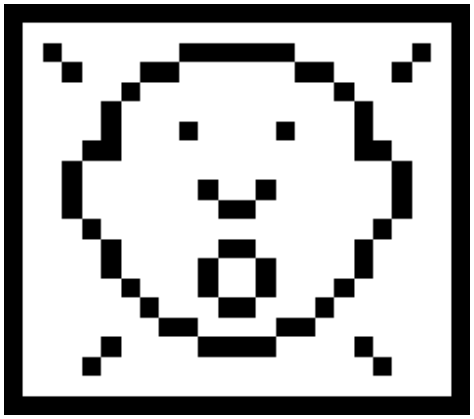

FIGURE 4.1 – The basic task is to find optimum offset values that minimise a performance index calculated over

4.2 TRE as Performance Function

Describe the interface with TRE, which is used evaluate solutions while serving as single point of contact with Optima.

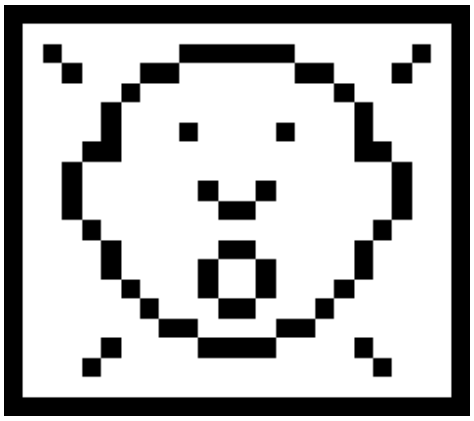

FIGURE 4.2 – Interaction between the optimiser and TRE

Condizioni dei test di TRE:

1. TRE fa un equilibrio e produce TPRB per tutto il giorno: gli intervalli sono ben più lunghi di un ciclo di semaforo quindi le svolte sono mediate e dipendono solo da domanda e green share
2. TRE salva l'istantanea dei flussi per cominciare le simulazioni di ottimizzazione a rete carica
3. Gli offset vengono modificati e per ogni individuo si fa un caricamento al secondo su una finestra temporale ristretta, partendo dall'istantanea salvata

4.2.1 Network Wide DTA

Spans the next time window for the entire network calculating average splitting rates (more than reasonable) which will be used by DNL, flow snapshot and cordon link flows.

4.2.2 Solution Evaluation with DNL

Solutions are implemented and KPI calculated.

4.2.3 Calling Method and Data Exchange

The data that needs to be passed between processes is very very little:

- opt - TRE 1 integer
- TRE - OPT corridor data
- opt - TRE solutions (a bunch of numbers: nothing)
- TRE - OPT fitness values

and can be packaged to minimise connections.

4.3 Performance and Scalability

Identify the performance bottlenecks and the most time consuming tasks.

Cite our article on parallel DTA and stress how the time hungry stuff is done only once. Quick performance calculations to show we're well in the ballpark.

Explain how parallel machines running TRE on the same model can be used to

- run multiple corridors
- evaluate more solutions in parallel complex problem space e.g. offsets+shares
- break up a slower problem like area optimisation

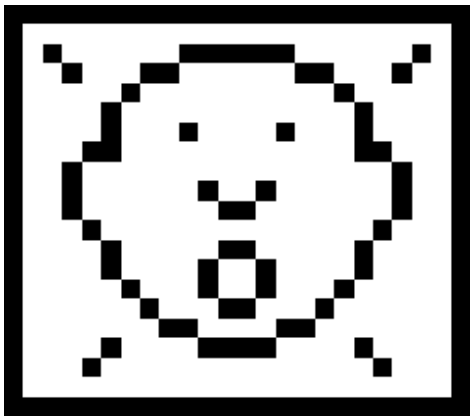

FIGURE 4.3 – Parallelisation Options

Chapter 5

Smart Objectives

Definizione degli obiettivi dell'ottimizzazione classica e di quella sostenibile.

5.1 The Optimisation Dilemma

Present the Big Question: are we really optimising? Are adaptive signals doing us any good?

- Signal optimisation sounds good but basically *identifies* with increasing the supply, increasing throughput, increasing flows.
- Not only this is of *extremely* dubious benefit to our cities, health and safety, but even in the short term it seems like there hasn't been too much thought whether it really is beneficial to traffic to maximise e.g. bandwidth if this only means piling up earlier and more heavily a few junctions down the road.
- It is true that it is extremely hard to model these effects explicitly, which is why we wish to attempt a heuristic approach that may help investigate the consequences of different choices, and maybe reverse-engineer a more structured logic.
- For the present work it is necessary to put aside the cold hard fact that the only way to alleviate traffic is to discourage people from relying on private transport.
- The plan is to try and see if at least we can be sure we are actually *reducing* the short term discomfort and external costs of driving.

Refer to the model outputs as described in section 3.3.3

5.1.1 Fundamental Quantities

The following quantities are calculated on the corridor over the entire simulation, and represent the reference quantities for calculation of key performance indicators.

The subscript T is often dropped for readability, but is *implied* for all quantities aggregated at the simulation span level and presented in the following section.

Section and Corridor Total

The *section total* is defined as the integral of the inflow to a given section of the corridor $a \in \mathrm{C}$, obtained piecewise in this case, as the total number of vehicles entered during each interval of the simulation window:

$$\omega_a^n = \sum_{\tau \in \mathrm{T}} q_{a,\tau} \Delta t^\tau \quad . \tag{5.1}$$

The *corridor total* gives an aggregate measure of how frequented the corridor is on the whole: it does not carry information on which sections are busier, but accounts for all vehicles that accessed *any* section during the simulation.

It is obtained as the cumulative total over all corridor sections, according to

$$\omega_{\mathrm{C}}^n = \sum_{a \in \mathrm{C}} \omega_a^n \quad . \tag{5.2}$$

Notice that ω_a^n implies no distinction based on whether the flows are coming from the previous section of the corridor or from a cordon arc, therefore vehicles travelling on more than one section are counted several times. This reflects the fact that the corridor is being used *more* if vehicles travel a greater portion of it than if they only were to use one section.

The total inflow index ω_{C}^n covers an important role as a *checksum*, since it ensures that any improvements in other cumulative indices are not really due to the corridor accepting fewer vehicles because of a deterioration in the traffic conditions.

User Time Spent and User Time Travelled

The most direct way to calculate how much time is spent by users on the corridor during the simulation window is to integrate the total number of vehicles present on any section over all time intervals. The total and section *User Time Spent* can be expressed as

$$\omega_{\mathrm{C}}^t = \sum_{a \in \mathrm{C}} \omega_a^t \qquad \text{where} \qquad \omega_a^t = \sum_{\tau \in \mathrm{T}} n_{a,\tau} \Delta t^\tau \tag{5.3}$$

therefore accounting for any vehicles already on the corridor at the start of the simulation, but not for the time that will be spent to get out of it beyond the end of the look-ahead window. However, since it is impossible to know how much time the vehicles have *already* spent on the corridor when the simulation begins, nor how far they have got down the arc they're found on, the time spent ω^t is not suitable for estimating the corridor performance with respect to travelled distances.

Disregarding the initial vehicles n_{C} and only considering flows that enter a corridor section during the simulation, it is possible to extrapolate from the results exactly how much time those vehicles will spend *travelling* the length of each arc, even beyond the end of the simulation. The average *User Time Travelled* is still a measure of time, but obtained from flows and travel times as

$$\omega_a^{tt} = \sum_{\tau \in \mathrm{T}} t_{a,\tau}^t \; q_{a,\tau} \; \Delta t^\tau \quad . \tag{5.4}$$

5.1.2 Performance Indicators

Minutes per Kilometre Travelled

From the user point of view, it makes sense to evaluate the performance of the corridor by considering the time required to travel the desired distance.

Referring to the User Time Travelled ω_a^{tt} expressed by (5.4) the *Minutes per Kilometre* cost function

$$\omega_{\mathrm{C}}^{T} = \sum_{a \in \mathrm{A}} \frac{\omega_a^{tt}}{\ell_a \, \omega_{\mathrm{C}}^{n}} \tag{5.5}$$

uses the travel times experienced by all users, normalised with respect to the relevant section lengths and averaged over all vehicles involved with any part of the corridor during the simulation. This gives an overall measure of the fluidity of traffic on the corridor, and has the dimensions of a time per unit length. The choice of units (and name) for this performance indicator is therefore dictated solely by human-readability: it makes sense to count minutes spent in traffic to cover one kilometre, and it is easy to refer to the fact that for an average speed of 60 km/h the value of ω^T would be 1.

Stop Ratio

Congestion

5.1.3 Dynamic Weighting

Tutto da riscrivere

Formalizzazione delle funzioni di costo: l'indice di coda (integrale tempo in coda per veicoli)/tempo passato nel corridoio fa

$$\frac{\sum_{a \in C} \sum_{t \in T} queu_{at}}{\sum_{a \in C} \sum_{t \in T} n_{at}}$$

non è reattivo soprattutto se gli archi sono lunghi perché il tempo in coda è comunque poco rispetto al totale

Numero di fermate non è male, calcolato come integrale dell'aumento della QUEU in ogni intervallo, corretto aggiungendo l'eventuale deflusso se la coda alla fine dell'intervallo non si è azzerata. Questo comporta degli errori (da quantificare esattamente?) negli intervalli in cui la coda comincia o finisce.

$$\sum_{a \in C} \sum_{t \in T} QUEU_{a,t} - QUEU_{a,t-1} + OFLW_{a,t} \Delta t$$

Il tutto è da normalizzare rispetto al numero di veicoli entrati su qualsiasi arco.

Altro obiettivo utile sarebbe guardare la congestione degli archi. Prendendo la congestione media si auto-pesano di più gli archi corti, il che avrebbe senso. Come indicatore della congestione, dal GLTM prendiamo la lunghezza della coda peggiore nell'intervallo rispetto all'arco $QUEL \in [0, 1]$ quindi come funzione di costo

$$Q_a^{MAX} = \sup\{QUEL_{a,t} | t \in T\} \tag{5.6}$$

Prendendo invece la lunghezza totale della coda?

5.1.4 Cost Function Correlation

Chapter 6

The Benchmark

This chapter introduces the Genetic Optimiser currently shipped with many OPTIMA systems, which will used as a sparring partner to evaluate the performance of the TRE based optimiser.

- Il modello è meglio o peggio di TRE ?
- Perché usare TRE che è molto più lento?
- come funziona BAL e cosa può fare?

Balance ha un modellino *mesoscopico* a velocità fisse e fa un taglio delle reti intorno alle junction che vuole controllare. Sugli ingressi alle sottoreti (una per ogni junction) usa profili di flusso costanti, ma se può usa i flussi uscenti di una junction per determinare i flussi entranti in una a valle (propagazione).

Il modello viene usato per ricavare le funzioni di costo FERMATE, LUNGHEZZA CODE (in realtà numero di veh in coda) e PERDITEMPO. Lui vede le code come F-E perché i suoi archi sono in realtà le corsie di svolta, ed usa una lunghezza MASSIMA per le code.

Punti di forza di Balance:

- Fa tante intersezioni
- E' veloce
- Aggiusta anche le durate degli stage

Punti deboli:

- Non è detto che le intersezioni si parlino tanto bene
- Non guarda avanti
- non vede l'arco ma solo l'approccio: forse una volta che la coda ha raggiunto il sensore per lui tutte le situazioni sono uguali, e sotto carico non gli cambia più niente
- Probabilmente tende a massimizzare la capacità dove è più richiesta, favorendo lo scorrimento ma provocando un comportamento "ingordo" che crea problemi a valle.

6.1 Traffic Model

Balance builds a two-level traffic model:

Macroscopic level : based on OD and assignment, gives the inflow values to the junction subnetworks, runs once per optimisation window

Mesoscopic level : second-by-second flow model, generates flow profiles based on macro flows and decision variables, computes performance indices

The macroscopic model performs an incremental traffic assignment, consisting of a series of partial assignments of increasing shares of the expected demand.
Each iteration assigns the corresponding share of flows according to a path search that accounts for any delay due to the flows already assigned, finally obtaining an estimate of the total flows on all arcs of the network. These may be further refined using iterative corrections if real flow measures are available.

OPTIMA may be plugged in here instead.

The mesoscopic model begins by computing an inflow profile $f_{a,\tau}$ for each approach to a junction, determining in detail the expected dynamics of the relevant inflows over the course of a signal cycle. ref scoot and formalise?

The notion of *arc* in this context corresponds to that of *lane group*, with the tail corresponding to the detector position (in accordance to the operational requirements specified in section 6.2), and the head with the signalised stop line. The flow entering the arc during interval τ is assumed to reach the stop line in a future interval τ' by propagating forwards at *constant speed*. The exit flow e_a accounts for inflow, queue, lane capacity and signal timing, but not for any downstream effects:

$$\text{with } \tau' \geq \tau \mid \tau_0 + \ell_a / v_a^0 \in \tau' \qquad e_{a,\tau'} = \begin{cases} 0 & \text{if lane group is closed} \\ \max\left(\hat{q}_a \,,\, \frac{n_{a,\tau}}{\Delta t^\tau} + f'_{a,\tau}\right) & \text{if lane group is open} \end{cases} . \tag{6.1}$$

The term $f'_{a,\tau}$ is used to model platoon dispersion along the link

$$f_{a,\tau} = \frac{1}{2t+1} \sum_{\tau''=\tau-t}^{\tau+t} f_{a,\tau''} \quad \text{where} \quad t \in \mathbb{N} = \ell_a \backslash v_a^0 \Delta t^\tau \tag{6.2}$$

this dispersion would average anything out if tt = cycle/2

describe propagation and define exit flow e

6.2 Performance Index

For each sub-network A_j around a controlled junction j, Balance evaluates a composite performance index based on vehicle delay D, number of stops S and queue lengths Q which takes the following form:

$$\omega_j^{PI}(\mathbf{q}_j, \mathbf{x}_j) = \sum_{a \in \mathrm{A}_j^-} \alpha_a^D D_a(\mathbf{q}_j, \mathbf{x}_j) \; + \; \alpha_a^S S_a(\mathbf{q}_j, \mathbf{x}_j) \; + \; \alpha_a^Q Q_a(\mathbf{q}_j, \mathbf{x}_j) \quad , \tag{6.3}$$

where the α_a terms are arc specific weights on the value of each component performance function, while $\mathbf{q}_j$ represents a generic vector of flow values at the junction and $\mathbf{x}_j$ the set of decision variables. plus there's an undocumented arc global weight

The delay, stops and queue terms are calculated by the mesoscopic model as detailed in the following sections.

Balance Delay D_a

Considering that the mesoscopic model regards lane groups as arcs, the total delay for a signal group over a given time window is approximated by the integral of time spent by vehicles on the corresponding arcs.

This is readily obtained from the arc occupancy profile during the desired time window, in turn obtained as the difference between the entry and exit cumulative flows (analogous to those defined for the GLTM in ref) which in this case can be taken to be the integrals of the entry and exit flow profiles described in section 6.1:

$$n^F_{a,t} = \sum_{\tau<t} f_{a,\tau}\Delta t^\tau \qquad \text{and} \qquad n^E_{a,t} = \sum_{\tau<t} e_{a,\tau}\Delta t^\tau \quad . \tag{6.4}$$

maybe define and use τ endpoints

The difference is then taken to represent *delayed* vehicles even though it will factor in *all* vehicles travelling down the approach lane, even if they are not in fact hindered in any way, but given the nature of f_τ described by equation (6.1) and the short length accounted for (between the sensor and the stop line) this seems like an acceptable approximation. Finally, the *total delay* is calculated over an interval T as:

$$D_{a,\mathrm{T}} = \sum_{\tau\in\mathrm{T}} \omega^n_{a,\tau}\Delta t^\tau = \sum_{\tau\in\mathrm{T}} \left(n^F_{a,\tau} - n^E_{a,\tau}\right)\Delta t^\tau \quad . \tag{6.5}$$

Balance Stops S

The number of stops is calculated

Balance Queue Length Q

Queues are

Balance Requirements

In order to ensure that the flow values used for Balance calculations are accurate, the following technical requirements should be met:

- there must be *one detector per lane* for all signalised lanes approaching a junction controlled by Balance, including pocket lanes;
- detectors must accurately count vehicles in adverse weather conditions and in case of congestion;

- detectors should be placed at least 20 m before the stop line, ideally between 40 and 50 m: in any case, the choice of placement should favour the ability to *count* vehicles accurately over optimal distance.

In addition to the above, the performance can be improved if extra detectors are used in the following non-crucial positions:

- at the edge of the network, to help calibrate the origin-destination flow estimates;
- on the forward star arcs of controlled junctions, to gather flow profiles for the prediction of arrival rates at downstream signals, to optimise inter-junction synchronisation.

6.2.1 Balance Settings

6.3 Setting equal grounds for comparison

How to put Balance and TRE on equal grounds for a significant performance comparison.

Considering the Balance requirements and settings illustrated in 6.2 and 6.2.1 respectively, it is possible to try and put Balance in operation conditions that are as close as possible to those of the TRE based Bandwidth optimiser.

This requires restricting the decision space of Balance while providing it the maximum possible detection capabilities, and further aligning the cost functions to specifically compare the efficacy of the algorithms while factoring out all expected sources of discrepancy.

Model setup

balance detectors

Vissim acceleration.

Vissim stochasticity removed.

Problems with connectors! At the start of connectors TRE is ok. At the very start of the entry links TRE is missing some but they're accounted for on the relevant connector.

ZONE	VISUM	VISSIM	TRE
1	672.5	665	635.1 +36.3
2	190	190	179.4 +10.3
3	180	180	170 +9.7
4	41.67	41	39.6 +2.3
5	91.67	91	86.6 +5.0
6	30	30	28.3 +1.6
7	30	30	28.3 +1.6

Discrepancy can be minimised by setting zero-length connectors, but not really eliminated.

Vissim flows seem capped at 1800 veh/hr regardless of the link capacity in visum.

Balance Decision Space

limiting balance to junction offsets only: Balance lavora stage based, cambiando gli istanti di inizio degli interstage. Per forzare il comportamento "solo offset" bisogna bloccare la durata degli stage, e lasciare libertà totale a tutti gli interstage.

Cost Function

Weighting out side approaches from Balance

Reproducing Balance PI in TRE

Chapter 7

Results

Tutto da scrivere:

Domanda stabile subcritica

Domanda stabile supercritica

Riduzione di capacità alla fine del corridoio per vedere se tiene le code a monte o fa andare tutti a morire in fondo.

7.1 Algorithm Parameters

This section presents results of the preliminary studies aimed at determining an optimal choice of parameters for the Genetic Algorithm. Genetic Algorithm parameters, population priming, dynamic weighting effects

7.1.1 Population Size

7.1.2 Population Priming

Slack Bandwidth section 1.4.2

7.2 Test Networks

Eight Junction corridor with side flows

Describe the features of sync8 and why it is representative of the problem.

7.3 Overall Cost Function Improvement

Demonstrate capability to improve over the SlackBand solution.

7.4 Comparison with Balance

Run hour long rolling optimisation and present in detail the evolution of traffic conditions.

7.5 Performance in Micro Simulation

Confirm effectiveness of solutions in a microsimulator, since in TRE they obviously work.

7.6 Performance

Considerations upon computation time issues and applicability, scaling. Different corridor lengths.

Bibliography

Carlos F Daganzo. The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research Part B: Methodological*, 28(4):269–287, 1994.

Carlos F Daganzo. The cell transmission model, part II : network traffic. *Transportation Research Part B: Methodological*, 29(2):79–93, 1995.

Russell C Eberhart and Yuhui Shi. Comparison between genetic algorithms and particle swarm optimization. In *International conference on evolutionary programming*, pages 611–616. Springer, 1998.

Guido Gentile et al. The general link transmission model for dynamic network loading and a comparison with the due algorithm. *New developments in transport planning: advances in Dynamic Traffic Assignment*, 178:153, 2010.

Peter Koonce, L Rodegerdts, K Lee, S Quayle, S Beaird, C Braud, J Bonneson, P Tarnoff, and T Urbanik. Traffic signal timing manual, no. Technical report, FHWA-HOP-08-024, 2008.

Michael James Lighthill and Gerald Beresford Whitham. On kinematic waves II. a theory of traffic flow on long crowded roads. *Proc. R. Soc. Lond. A*, 229(1178):317–345, 1955.

Gordon F Newell. A simplified theory of kinematic waves in highway traffic, part I: General theory. *Transportation Research Part B: Methodological*, 27(4):281–287, 1993.

Paul I Richards. Shock waves on the highway. *Operations research*, 4(1):42–51, 1956.

J G Wardrop. Road paper. some theoretical aspects of road traffic research. *Proceedings of the Institution of Civil Engineers*, 1(3):325–362, 1952.

Isaak Yperman. *The link transmission model for dynamic network loading.* PhD thesis, Katholieke Universiteit Leuven, 2007.

Isaak Yperman, Steven Logghe, and Ben Immers. The link transmission model: an efficient implementation of the kinematic wave theory in traffic networks. In *Proceedings of the 10th EWGT Meeting*, pages 122–127. Poznan Poland, 2005.