University of Huddersfield Repository

Chrpa, Lukáš, Magazzeni, Daniele, McCabe, Keith, McCluskey, T.L. and Vallati, Mauro

Automated Planning for Urban Traffic Control: Strategic Vehicle Routing to Respect Air Quality Limitations

Original Citation

This version is available at http://eprints.hud.ac.uk/id/eprint/30666/

http://eprints.hud.ac.uk/

Automated Planning for Urban Traffic Control:
Strategic Vehicle Routing to Respect Air Quality Limitations

Lukáš Chrpa [a,*], Daniele Magazzeni [b], Keith McCabe [c], Thomas L. McCluskey [a] and Mauro Vallati [a]

[a] *PARK research group, University of Huddersfield, United Kingdom*
[b] *Department of Informatics, King's College London, United Kingdom*
[c] *KAM Futures, United Kingdom*

Abstract. The global growth in urbanisation increases the demand for services including road transport infrastructure, presenting challenges in terms of mobility. These trends are occurring in the context of concerns around environmental issues of poor air quality and transport related carbon dioxide emissions. One out of several ways to help meet these challenges is in the *intelligent routing* of road traffic through congested urban areas.

Our goal is to show the feasibility of using automated planning to perform this routing, taking into account a knowledge of vehicle types, vehicle emissions, route maps, air quality zones, etc. Specifically focusing on air quality concerns, in this paper we investigate the problem where the goals are to minimise overall vehicle delay while utilising network capacity fully, and respecting air quality limits. We introduce an automated planning approach for the routing of traffic to address these areas. The approach has been evaluated on micro-simulation models that use real-world data supplied by our industrial partner. Results show the feasibility of using AI planning technology to deliver efficient routes for vehicles that avoid the breaking of air quality limits, and that balance traffic flow through the network.

Keywords: Automated Planning, Urban Traffic Control, Pollution Limits, Micro Simulation

1. Introduction

In the 21st Century the world's population is expected to increase from 5.9bn in 2013 to 9.6bn by 2100. At the same time the world's population is expected to become more urbanised, in 2005 there was a 50/50 split between urban and rural population, by 2050 this is predicted to change to a 70/30 split. This huge growth in urbanisation increases the demand for housing, associated utilities and services including transport infrastructure, public transport, vehicle parking and better interchanges between the urban and national transport networks. In turn, this leads to in-creased pressure on land use and development within or around urban areas contributing to land shortages and urban sprawl. These trends present significant challenges in terms of mobility.

There are expected to be key developments in terms of growth of big data in transport, with increased deployment of roadside and vehicle sensors, and increased automation with the introduction of semi-autonomous and fully autonomous vehicles into the urban environment in the next decade. These trends are occurring in the context of concerns around environmental issues of poor air quality and transport related carbon dioxide emissions.

Current urban transport management systems help to minimise delay within day to day traffic flows us-

---

*Corresponding author. E-mail: l.chrpa@hud.ac.uk

ing controls such as self-tuning traffic light clusters, but are not designed to take into account varying vehicle types, or regional demands such as avoiding poor air quality, utilising the network fully, or working adequately in the face of unplanned exceptional events. Our goal is to show the feasibility of using automated planning to intelligently route all vehicles collectively through Urban Areas, taking into account a knowledge of vehicle types, vehicle emissions, route maps, air quality zones etc. This contrasts with the current situation where personal navigation devices produce routes for a vehicle in isolation of other travellers, and traffic controls can only react to local flows of traffic. Our approach anticipates future developments in route generation for road users, which is likely to be based on a balance between minimising the average delay of all traffic in an urban region, and observing regional constraints such as area and weather dependent air quality limits.

Whereas there has been a long record of the use of AI techniques in road transportation [16,24], with some notable exceptions, for example [1,13,26], there has been little application of AI Planning and Scheduling techniques to urban traffic control. The novelty of this paper lies in the application of automated planning techniques for strategic urban traffic control, in order to calculate routes for vehicles entering an urban area to satisfy regional-level constraints.

Micro-simulation models of road traffic are models that consider the traffic system at the level of the individual vehicle [21]. We create a micro-simulation model for an urban area, and show how planning can, by varying the vehicle flows through roads, or re-routing the highest polluting vehicles around poor air quality zones, ensure that pollution limits are respected while maximising the utilisation of network capacity, without a marked effect on the goal of delay minimisation. We test our method using various scenarios within a real urban area, and a generic urban area, incorporating poor air quality zones.

The paper is organised as follows. In Section 2 we provide the required background information, with regards to the application area and automated planning. Section 3 describes the problem requirements. Then, Section 4, we introduce the formulation of micro-simulation model of road traffic used. In Section 5 we present the method exploited for performing strategic re-routing using automated planning. Results of the experimental evaluation are presented in Section 6 and subsequently discussed in Section 7. Finally, we pro-

vide conclusions and describe future developments in Section 8.

2. Background

This section is devoted to introducing the relevant background in traffic control and automated planning.

2.1. Application Background

Within the broad topic of traffic control, in this work we focus on Urban Traffic Control (UTC).

2.1.1. Traffic Control in Urban Areas

UTC is normally the responsibility of local authorities whose aims include reducing congestion, improving journey times, increasing the reliability of the road network, safety regulation compliance and traffic pollution limitation. In a recent investigation by some of the authors involving UTC of two major European conurbations [15], it was found that the predominant goal of UTC centres is to minimise delay to the traveller. UTC is distinct from Freeway traffic management, where the number of junctions is much lower, and average speeds much higher. It is also distinct from traffic controls in rural areas, where the concentration of traffic and sensors is much lower.

UTC forms a key component of *intelligent mobility* (the safe and expeditious movement of people and goods) in urban areas. It is made up of four different levels, namely:

– **operational level**, this is the domain of the commands to individual traffic control devices;
– **tactical level**, this involves the control of individual intersections or small groups of intersection;
– **strategic level**, this involves the management of routes in and around urban areas;
– **external level**, this is the interface to control centres controlling adjacent urban, rural or interurban areas.

On a day to day basis, most UTC happens at an operational or tactical level, with the main controls available to operators being traffic light plans, advisory / warning messages displayed on variable message signs, variable speed limits and information distributed on social media and applications based on open data platforms. Notable traffic lights scheduling systems using linked or self-optimising delay reduction tech-

niques include SCOOT[1], MOVA[2] and SCATS[3] [5,20, 25]. These control traffic light clusters, changing their own phase lengths depending on current traffic conditions, have embedded triggers which can instantly adjust the behaviour of a set of lights. Such adjustment to "fixed plans" can be made at peak times of the day, usually to cope with the vagaries of rush hour. However, given several weeks notice, if operators want to increase flow in a certain direction, due to for example a large football match, they can act at a strategic level by composing a fixed plan to increase the green light phase through a particular corridor of traffic lights. These plans are typically constructed manually, and can be quite complex to create and validate.

Even the most advanced urban traffic management centres face major challenges, and have shortfalls in what they can deliver. One example of such a challenge is unbalanced distribution of traffic in the road network. In particular, one might observe that in rush hours some roads might be overcrowded by traffic while some other roads are nearly empty. This is mainly caused by the fact that traffic is navigated via the same route between given way-points. Therefore, alternative routes that might not be shorter but faster because of little traffic are not or only rarely exploited, and hence the full capacity of the Network is not used. Another shortfall, as reported in [13], is that there are no automated methods to deal with significant unexpected events such as the sudden loss of a road. Self tuning traffic lights can deal with changes in traffic flow, but fail in the face of serious congestion caused by an incident.

Apart from the manual creation of fixed, connected traffic light plans (e.g. enabling a "corridor" effect as mentioned above) UTC can do little to influence traffic at a strategic or external level, such as flow balancing through different parts of a region. In particular, there are currently little or no controls that an operator can enact on a day by day basis to change traffic behaviour to reduce the traffic through specific areas of a network, or to select different types of vehicle to travel using different routes. A recent review [4] identified areas where the operation of UTC could be enhanced, highlighting the issues above. In particular, it emphasised the lack of air quality considerations in UTC technologies, as discussed in the next section.

2.1.2. UTC in the Future

Looking towards the future, UTC is often considered within the "Smart City" initiatives, as its function is to make better use of the physical infrastructure of a city, and potentially reduce usage of "environmental capital". With the use of personalised information services widespread, vehicle route choice is becoming increasingly dependent on automated route finding software, especially for commercial vehicles. Currently these routes are enacted by the driver, but in future transport vehicles will follow calculated routes autonomously. Commercial fleets have a clear motivation[4]; they are planning to install software for their vehicles that will take into account the characteristics of their vehicle and local authority constraints on routes, which in combination with minimising delay criteria will be used to decide routes. In other words, important regional criteria such as air quality will be taken into account in future route navigation systems.

2.1.3. Air Quality in Urban Areas

In many areas of the world there are in force strict controls on air quality, and local authorities can be fined if an area exceeds the fixed threshold[5]. Naturally, it is within the centre of urban areas where the air quality is poorest, with vehicles by far being the biggest polluter. For example, in 2012 there were 2720.9 km of roads of the United Kingdom that exceeded the EU air quality standards. While this is expected to decrease to 501.2 km by the deadline of 2020, 501.2 km of road exceeding health standards for air quality still represents a significant health issue for the UK.

The parameters that govern air quality in an urban area due to traffic, are:

– **area specific**, certain areas have poor air quality intrinsically, mainly because they are shielded from the prevailing wind, hence the pollution cannot be dispersed easily;
– **vehicle specific**, the level of emissions depends on the amount and type of traffic. For example diesel buses, large delivery vans and trucks are about 15 times more polluting than cars;
– **weather dominated**, if there is little wind in the direction that will disperse the pollutants, or in particular if the day is still, then the problems arise. Wind usually disperses pollutants, but it is

also possible that it could accumulate it into certain areas.

Hence, an area would only likely have air quality problems on certain days when the weather was of a certain type and the volume of traffic was high, and of the wrong sort. Potential management controls at a strategic level include routing higher level polluting vehicles away from the pollution. This is very locality-dependent, e.g., assuming we have a north - south corridor, and a line of high buildings going north - south. Then on days of dangerous air quality, a potential strategy would be to route higher polluting vehicles on the west side of the buildings, since the prevailing wind may disperse them. The east side will be shielded by the buildings, hence will be more likely to be a problem area.

2.2. Automated Planning

Automated Planning deals with the problem of finding a totally or partially ordered sequence of actions that transform the environment from a given initial state to some of the goal states [14].

In this work we focus on *Temporal Planning*, a subclass of Automated Planning that reasons with deterministic actions whose execution takes time (i.e., "durative actions") in a deterministic and fully observable environment.

For describing the planning domain model (environment, actions) and planning problems (objects, an initial state, a goal) we use PDDL 2.2 [10,17], which is supported by several planning engines. The environment in PDDL 2.2 is described by *predicates* and *numeric fluents*. *Actions* are specified via their execution *duration*, *preconditions* which are logical expressions that must hold in order to make the action executable, and *effects* which are sets of literals or fluent assignments that take place when the action is executed. In PDDL 2.2, preconditions can take place just before the action is executed, during the action execution or just before finishing execution of the action. Similarly, effects can take place just after the action is executed, or just after finishing action execution [10]. A *Planning Domain Model* consists of definitions of predicates, numeric fluents and actions. A *Planning Problem Description* consists of a set of objects, an initial state (a set of grounded predicates and fluent assignments), and a set of goals (logical expressions). Notice that "Time Initial Literals" that allow facts to become true at a predefined timestamps are supported by

PDDL 2.2 [17]. A *plan* is a set of triples in the form of ⟨timestamp,action,duration⟩ such that executing these actions in the corresponding timestamps for a given duration (it must always be possible) transforms the environment from the initial state to some state where all the goals are satisfied (i.e. a goal state).

3. Problem Requirements

We performed the study for this paper with an ITS (intelligent transport system) consultant, who is also a co-author of the paper, and were helped by members of, and financially supported by, a European Network in future transport systems[6]. We visited and investigated the current capability, and future needs, of two major conurbations, both with over 1 million citizens. We gathered that in the near future there will be a requirement for software that can work at a UTC strategic level, by producing routes for vehicles through urban areas in partnership with the traveller, the urban traffic management authority, and, in the case of fleet vehicles, the fleet-owning company. The routes produced would need to make full use of the road network, and be aimed at minimising delay. In certain cities where pollution is a problem, it would be necessary, on weather-determined days, for the software to route vehicles in order to avoid breaking the pollution limits.

Whereas fixed plans have been used at a strategic level in UTC, they are created only for expected, well specified disturbances such as large city events, and require much preparation. Hence, it is recognised that the problem outlined is too complex for manual solution. Further, the requirements demand that the solution is flexible enough for potential day-to-day changes in goals and initial state characteristics, e.g., changes to road topology, or goal optimisation constraints, and unexpected events such as sudden road closures. Also, the solution method needs to reason at a strategic level, using the whole of the network within the area, and using knowledge-based features such as types of vehicle and their emission characteristics. To show the feasibility and advantages of using automated planning to generate individual routes, we have produced domain models which embodied a micro-simulation model of traffic, as specified in the next section.

An example scenario in which our approach will fit, is as follows. Vehicles travelling in a region obtain a route for their whole journey, departing from some origin O to some destination D, using a conventional navigation algorithm, but where the calculated routes will be visible by the UTC centres in the region. UTC will filter the routes on whether or not they intersect one of their controlled regions. Assuming the vehicle enters a controlled region at point A, and exiting B, then the UTC centre's software will have the authority to change the route. The part of the route from A to B is recalculated by UTC's planning software, to take into account the authority's regional goals and constraints (in our case air quality, network capacity, vehicle type), and replaces the old section of the route from A to B. Thus, re-routing causes minimal disruption as it focuses only on re-routing vehicles that traverse the urban area. To deal with the delay between a route being requested and the actual travelling of that route, a "plan - validate - replan" approach can be taken [9]. After a route is given, when vehicles arrive at the edge of the UTC centre region, if the time is different from the expected one a checker such as VAL [12] can be used to check all constraints are still satisfied, otherwise the route can be replanned for that vehicle.

4. Problem Formulation

In this section we provide a detailed description of the PDDL 2.2 encoding of the UTC problem, and the extensions needed in order to allow a domain-independent planner to consider pollution constraints.

4.1. The Static and Dynamic Model of a Region of the Road Network

We present a formulation of a micro-simulation model of road traffic [21], that is one that considers traffic flow at the level of the individual vehicle (this is necessary as routes need to be calculated for each vehicle). It is worth pointing out that such models in the area of ITS, and Transport Studies in general, are designed to be executed and their behaviour observed. Here we are using such a model *as the input to a reasoning system* (i.e. a planner). In the former, the behaviour of individual traffic is already determined by the model and its semantics, whereas in the latter we are expecting the planner to supply a key part of the picture - the vehicles' routes.

A region of the road network can be represented by a directed graph, where edges stand for road sections and vertices stand for either junctions, entry or exit points. Intuitively, vehicles enter the network in entry points, and exit the network in exit points. Each road section has a given length and capacity (i.e. a maximum number of vehicles it can serve). In junctions, we must consider that vehicles cannot go through it in some directions simultaneously, otherwise they moght collide. In our case, we will consider a simplistic case that at most one vehicle can pass through the junction at once. Notice that such a case reflects 4-stop junctions that are very common in US urban areas.

Definition 4.1. *Let (V, E) be a directed graph such that $\forall v \in V : (indeg(v) = 0 \Rightarrow outdeg(v) = 1) \land (outdeg(v) = 0 \Rightarrow indeg(v) = 1)$[7]. Edges in E represent one-way (or one direction of two-way) road sections. A vertex $v \in V$ represents:*

- *an entry point if $indeg(v) = 0$*
- *an exit point if $outdeg(v) = 0$*
- *a junction otherwise*

Let $C : E \to \mathbb{N}$ be a function representing road section capacity and $l : E \to \mathbb{R}^+$ be a function representing road section length.
Then $\mathcal{N} = \langle V, E, C, l \rangle$ is a Road Network.

This structure for the road network represents the static part of the environment.

To capture a current traffic situation which forms part of the dynamic part of the environment, we have to consider time. Let T be a set of time-stamps. Then, we can define functions referring to positions of vehicles in the road network as well as usage of the roads. Let $use : E \times T \to \mathbb{N}_0$ be a function referring to a number of vehicles on the road section in a given time-stamp. Clearly, it must hold that $\forall t \in T, \forall r \in E : use(r, t) \leq C(r)$. For capturing the position of vehicles on road sections in a given time-stamp, we define relations $head$ and $tail$ such that $head \subseteq X \times E \times T$ and $tail \subseteq X \times E \times T$ (X is a set of vehicles). Lastly, we have to capture situations when in a give time-stamp vehicles are ready to enter the network in a given entry point or have just exited the network in a given exit point. For this purpose we define relations $ready$ and $exited$ such that $ready \subseteq X \times V \times T$ and $exited \subseteq X \times V \times T$.

---

[7]$indeg(v)$ represents the number of incoming edges to v and $outdeg(v)$ represents the number of outgoing edges from v

We define four planning operators that are used to navigate vehicles through the network. We assume the operator to be applied in a time-stamp t and its application to last Δt. Notice that Δt might vary even for different instances of the same operator (e.g driving through different road sections may take different amount of time). For the *release-vehicle* and *exit-vehicle* operators we assume instantaneous effects (Δt=0).

- An operator *release-vehicle*(x, v, r) releases a vehicle x at the entry point v to the head of the road section r. As a precondition it must hold that r is an outgoing edge from v, $ready(x, v, t)$ is true and $C(r) > use(r, t)$. The effect of this operator is that $use(r, t) = use(r, t) + 1$ and $head(x, r, t)$ becomes true.
- An operator *drive-through-junction*(x, v, r_1, r_2) navigates a vehicle x from the tail of the road section r_1 through the junction v to the head of the road section r_2. As a precondition it must hold that r_1 is an incoming edge and r_2 an outgoing edge from v, $tail(x, r_1, t)$ is true, $\forall t' \in (t; t+\Delta t) : C(r_2) > use(r_2, t')$, and no instance of operator *drive-through-junction*(x', v, r_1', r_2') such that $x \neq x'$ is being executed in $\langle t, t+\Delta t \rangle$. The effect of this operator is that $tail(x, r_1, t)$ becomes false, $head(x, r_2, t+\Delta t)$ becomes true, and $use(r_1, t) = use(r_1, t) - 1$ and $use(r_2, t+\Delta t) = use(r_2, t) + 1$.
- An operator *drive*(x, r) moves a vehicle x from a head of r to its tail. As a precondition it must hold that $head(x, r, t)$ is true. The effect of this operator is that $head(x, r, t)$ becomes false, $tail(x, r, t+\Delta t)$ becomes true.
- An operator *exit-vehicle*(x, v, r) allows a vehicle x to leave the network in the exit point v. As a precondition it must hold that r is an incoming edge to v and $tail(x, r, t)$ is true. The effect of this operator is that $tail(x, r, t)$ becomes false, $exited(x, r, t)$ becomes true, and $use(r, t) = use(r, t) - 1$.

We consider frame axioms that, informally speaking, keeps the same value of a function (or a truth value of a relation) between consecutive time-stamps unless some operator changes it. For instance, $use(r, t) = use(r, t')$ ($t' > t$) if no corresponding instance of an operator modifying the use function is executed in between t and t'.

A planning problem is specified by a road network, which captures the static part of the problem, initial

```
(:durative-action DRIVE-THROUGH-JUNCTION
:parameters
    (?c - car ?r1 - road ?j - junction
    ?r2 - road ?z1 - zone ?z2 - zone)
:duration (= ?duration 1)
:condition (and
    (at start (tail ?c ?r1))
    (at start (to ?r1 ?j))
    (at start (from ?r2 ?j))
    (at start (in ?r1 ?z1))
    (at start (in ?r2 ?z2))
    (at start (<= (use ?r2)
            (capacity ?r2)))
    (at start (free ?j))
    (at end (<= (pol-level ?z2)
  (- (pol-limit ?z2) (pol-produce ?c)))))
)
:effect (and
    (at start (decrease (pol-level ?z1)
            (pol-produce ?c)))
    (at end (increase (pol-level ?z2)
            (pol-produce ?c)))
    (at start (not (free?j)))
    (at start (decrease (use ?r1) 1))
    (at end (increase (use ?r2) 1))
    (at start (not (tail ?c ?r1)))
    (at end (head ?c ?r2))
    (at end (free?j)))
)
```

Fig. 1. The PDDL encoding of the drive-through-junction operator. The pollution constraints are emphasized in italics.

and goal positions of vehicles (the use function is appropriately initialised), which captures the dynamic part of the problem. Timed Initial Literals [17] are used to represent situations when a vehicle is ready to enter the network later. For example, a vehicle x is ready to enter the network at the entry point e in a time-stamp 5. Then, we represent it (in PDDL2.2) as `(at 5 (ready x e))`. Solution plans are in our case optimised for "make-span", i.e., we minimise a cumulative time needed to navigate vehicles from their initial to their goal positions.

4.2. Extending the Problem Formulation for Pollution Constraints

The problem formulation introduced in the previous subsection can be extended in order to comply with pollution constraints. We divide an urban region into zones, depending on the geography of the region, and taking into account daily weather (wind speed and direction) we assume a limit on the amount of

emissions from vehicles can be calculated each day, for each zone - we call this the pollution limit. Further, we assume that each road section belongs to exactly one zone. Formally speaking, given a road network $\mathcal{N} = \langle V, E, C, l \rangle$ we define a set of zones $Z = \{z_1, z_2, \ldots, z_k\}$ such that $\bigcup_{i=1}^{k} z_i = E$ and $\forall i, j, i \neq j : z_i \cap z_j = \emptyset$. For each zone we define a pollution limit as a function $polLimit : Z \rightarrow \mathbb{N}$. For each vehicle we define its emission as a function $emis : X \rightarrow \mathbb{N}$ (X is a set of vehicles). Then, we define a function $polLevel$ that represents a current pollution level in a given zone in a given time-stamp, i.e., $polLevel : Z \times T \rightarrow \mathbb{N}_0$. Clearly, it must hold that $\forall t \in T, \forall z \in Z : polLevel(z, t) \leq polLimit(z)$.

Given the additional constraints, some of the planning operators have to be amended by incorporating additional preconditions and/or effects in order to fulfill the constraints. Again, we assume the operators to be applied in a time-stamp t and its application to last Δt time.

- *release-vehicle*(x, v, r, z) is modified such that in precondition it must hold that $r \in z$ and $polLimit(z) \geq polLevel(z, t) + emis(x)$. The effects are extended by adding $polLevel(z, t) = polLevel(z, t) + emis(x)$.
- *exit-vehicle*(x, v, r, z) is modified such that in precondition it must hold that $r \in z$. The effects are extended by adding $polLevel(z, t) = polLevel(z, t) - emis(x)$.
- *drive-through-junction*$(x, v, r_1, r_2, z_1, z_2)$ is modified such that in precondition it must hold that $r_1 \in z_1$, $r_2 \in z_2$ and if $z_1 \neq z_2$, then $\forall t' \in \langle t; t + \Delta t \rangle : polLimit(z_2) \geq polLevel(z_2, t') + emis(x)$. If $z_1 \neq z_2$ the effects are extended by adding are extended by adding $polLevel(z_1, t) = polLevel(z_1, t) - emis(x)$ and $polLevel(z_2, t + \Delta t) = polLevel(z_2, t) + emis(x)$

The change in the pollution level in a given zone is approximated in these operators by increasing it when a vehicle enters a road in that zone, and decreasing it when the vehicle leaves the zone, to approximate the pollution dispersion. The *drive-through-junction*, encoded in PDDL, is depicted in Figure 1.

5. Strategic Re-Routing through Planning

In the section we introduce the two-step approach used for dealing with strategic re-routing.

5.1. A Two Step Approach

As previously introduced, we consider a scenario where vehicles already have a planned route for their journey, using conventional routing software. For all vehicles whose routes are detected to intersect the controlled urban region, their routes are re-planned for that region, in order to utilise the available road network (which would respect any problems resulting from sudden changes in capacity due to e.g. a road incident leading to blockages). It is the responsibility of the UTC to monitor the level of pollution emissions of vehicles traversing the urban area, and, if necessary, to re-route some of these vehicles in order to maintain the pollution level in the area within the safety limit. In this case a new planning problem is dynamically created by considering the expected entry points of all vehicles into the urban area. Timed Initial Literals are used for this purpose, by modelling both the exact entry point and the expect time of arrival of each vehicle. A new plan is then generated for these vehicles, but this time pollution levels in the urban area must be kept below the safety thresholds, therefore some vehicles may be re-routed.

Finally, the two plans need to be merged, by replacing actions in the initial plan referring to the vehicles that have been re-routed with the corresponding actions in the re-routing plan. It is easy to observe that the final plan is guaranteed to satisfy the pollution limit constraint, as all the vehicles traversing the urban area have been routed while taking the pollution constraints into account. In order to guarantee that at most one vehicle passes through each junction at the same time, and that road capacities are respected, we adopt a delay strategy for the vehicles that have been re-routed. A plan and validate approach is used [9] to dynamically check that the solutions are valid. In summary, dynamic problem formulation and planning are used to accomplish the UTC task of strategic re-routing.

A simple example scenario is provided in the following section to illustrate the method.

5.2. Illustrative Example

Figure 2 shows an example scenario, where eight vehicles v1...v8 have to reach their exit points, and can choose to traverse the urban area (highlighted in green) or use the external ring road. Figure 2 (a) shows the initial plan containing the best path for each vehicle. Given the entry and exit points of each vehicle, and without taking pollution emissions into account,
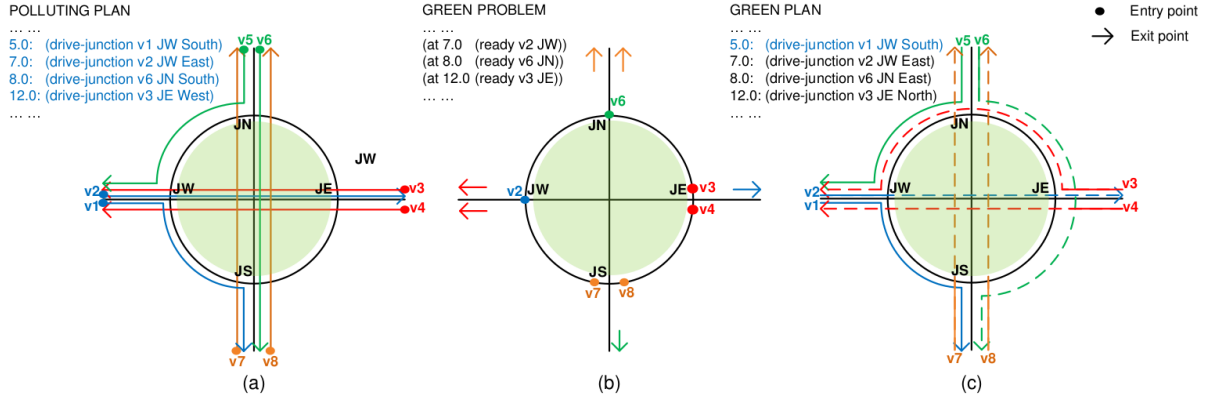
Fig. 2. Example of re-routing: eight vehicles have to reach their exit points, either passing through the city centre or using the external ring road.

for vehicles `v1` and `v5` it is better to use the ring road, while for the other vehicles the shortest path traverses the urban area. The fragment of the corresponding plan (`POLLUTING PLAN`) focuses on vehicles `v1`, `v2`, `v3` and `v6` and shows the time points at which they traverse the junctions at the borders of the urban area[8].

In the next phase, shown in Figure 2 (b), from `POLLUTING PLAN` a new problem is generated, with the objective to re-route the vehicles traversing the urban area in case their pollution emissions exceed the safety limit. To this end, only the vehicles entering the urban area are considered in the new problem, while vehicles `v1` and `v5` are ignored. As shown in the fragment of the new problem (`GREEN PROBLEM`), for vehicles `v2`, `v3` and `v6`, TILs are used to set their new entry points, while the exit points remain the same. The pollution limit constraint is added to the new problem and a new plan is generated, as shown in Figure 2 (c). Here, dotted lines represent the new generated paths. Note that not all vehicles need to be re-routed, as for example vehicles `v2` keeps traversing the urban area. On the other hand, vehicles `v3` and `v6` are re-routed to the ring road in order to keep the pollution emissions in the urban area below the safety limit. The new plan can then be merged with the initial plan, as shown in the fragment of `GREEN PLAN`. Note that the new plan is guaranteed to satisfy the pollution constraint after the merging with the `POLLUTING PLAN`, as from the latter only the vehicles not entering the urban area are considered.

---

[8]For brevity, the action parameters are replaced by the traversed junction and the next direction.

6. Experimental Evaluation

The aim of the experimental evaluation is to test whether the current state of the art of domain-independent planning approaches can perform the required vehicle routing in both real and generic urban scenarios. The data used in these scenarios (the relative percentages of different types of vehicles in the urban area, the relative pollution emissions of vehicles, the road topology of a urban area) has been extracted from openly available and published data. Our experimental evaluation tests whether the current state of the art of domain independent planning approaches can handle the proposed two-step approach, and the possibility to effectively plan the routes of vehicles in a micro-simulation urban context. In particular, the objectives of the particular tests are: i) testing the feasibility of the two-step approach, in terms of CPU time needed; ii) assess the distribution of the traffic on the network, according to road capacities and pollution limits; iii) demonstrate that pollution limits in critical areas can be kept under control, without major delays for the urban traffic.

In our experiment we used the well-known temporal planner POPF [7]. The planner has been run on a cluster with computing nodes equipped with 2.5 Ghz Intel Core 2 Quad Processors, 4 GB of RAM and Linux operating system. A cutoff of 3600 seconds was imposed for solving each problem.

6.1. Scenarios

In our experimental evaluation we considered the structure of the typical European city as depicted in Figure 3. There is a ring road around the city centre, that allows vehicles to quickly move between different
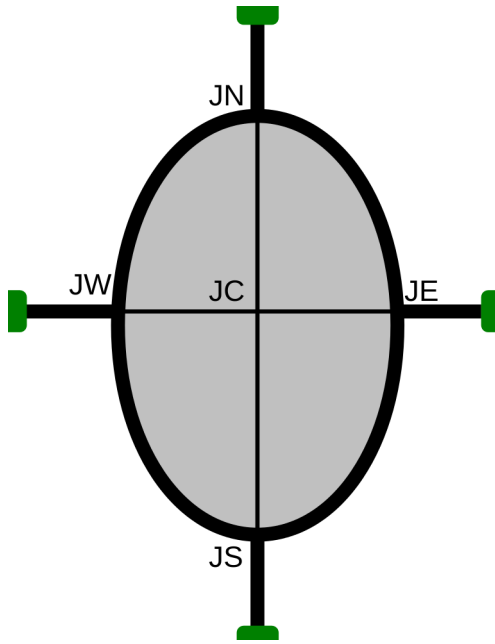
Fig. 3. The simplified Ring Road network, referred as Scenario 1. Gray area indicates the city centre, where pollution must be kept low. JN, JE, JW, JS and JC are the considered junctions. Green squares indicate entry/exit points.
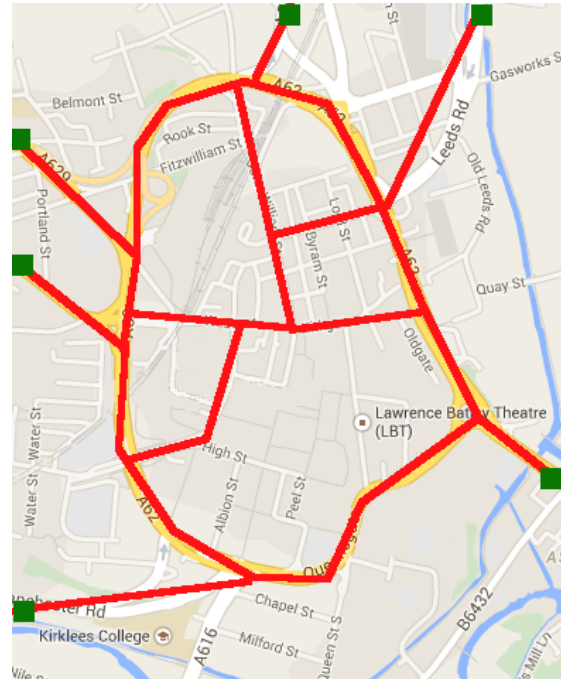

Fig. 4. The map of Scenario 2, which encodes an actual European town (namely, Huddersfield, in United Kingdom). In red the modelled network. Green squares indicate entry/exit points.

areas of the city, as well as all the intercity traffic. The ring road is elliptic, thus navigation on the north-south direction takes more time than the corresponding west-east navigation. There are four entry points to the city, which is navigable through the ring road and through the roads that cross city centre. The area within the ring road, which represents the densely populated area, as strict pollution limits, in order to limit the access for heavy vehicles, and avoid a very dense traffic.

In a second set of experiments, we encoded in our framework the structure of a real European town. Connections, as well as road length and capacity have been modelled by considering the real data. The map is shown in Figure 4. There are six entry/exit points to the network, which consists of a ring road, and six road sections that can be used for crossing or reaching the town centre.

6.2. Results

In the considered scenarios, we generated problems with the number of vehicles ranging between 20 and 50. Vehicles include cars, vans (6-10%) and heavy vehicles (7-20%), in proportion to the published averages within urban areas of these vehicle groups. They are randomly distributed among entry points, and are re-

leased at different times. In testing instances, we considered a rush hour situation, in which many vehicles are released at entry points in a very short time interval, and a more "relaxed" condition, in which vehicles appear less frequently at entry points. Considered problems in both scenarios have been designed for stressing the pollution limits in the urban area, by setting vehicles entry points on one side of the centre, and exit points on the opposite side of the network.

We followed the two-phase approach in both scenarios. We first generated a plan for navigating vehicles throughout the network while ignoring pollution limits. We then generated a new problem file, taking into account the vehicles traversing the urban area and generated a new plan for them, while considering the pollution constraint. All plans were then validated using VAL. The results are summarised in Table 1 and 2.

In Scenario 1, both finding the first plan, and re-routing vehicles are efficiently performed; both steps usually require less than 60 CPU time seconds to be solved. We also observed that makespan is not significantly increased when re-routing occurs: average makespan rise is 19.9%, and the number of re-routed cars ranges between 8 (40%) and 14 (28%). On the

other hand, a large reduction of pollution emissions in the urban area has been observed, up to 54%.

In Scenario 2, we observed very interesting behaviour of our planning framework. Only problems which include between 20 and 30 vehicles are solved; on the rest, POPF runs out of memory. On solved instances, the first step is computationally expensive, as it requires an average of 451 CPU time seconds to be solved. On the other hand, re-routing vehicles took at most 12 CPU time seconds. Number of vehicles that were re-routed ranged between 5 and 9. Furthermore, re-routing never increased the total makespan, while pollution emissions were reduced in the urban area by between 29% and 36%.

We noticed that in plans produced by both steps, i.e cases where POPF was able to solve the 2 different planning poroblems, vehicles are effectively distributed across the whole network. This reduces the delay of vehicles, since junctions and roads are less congested. Also, the good distribution of vehicles on the network, permits to spread pollution in different areas of the map, without seriously affecting a specific small area of the network.

As can be seen in Table 2, the second phase (re-planning) takes considerably less CPU-time than the first phase. With increasing complexity of scenarios the percentage of CPU-time with respect to total CPU-time drops to about 5% in Scenario 1 and to less than 1% in Scenario 2. In one case (Scenario 1, normal hour 50 cars), this trend is violated having Phase 2 CPU-time higher than Phase 1 CPU-time. This shows that planner's performance besides problems' complexity might be also affected by their structure.

Figure 5 shows an example of the route identified by our approach for a single vehicle in Scenario 1. `Vehicle1` enters the network from the north entry point (`en`), and has to reach the south exit-point. In order to respect the pollution limit of the urban area, and to limit congestion in the city centre, the vehicle is navigating the network by exploiting the western section of the ring road.

7. Discussion

In this paper, we applied centralised domain-independent temporal planning approach to generate plans for navigating vehicles through the urban areas in order to follow the air-quality constraints. Our model dealt with single vehicles, which in literature is referred to "micro simulations".

Table 1

Number of cars, makespan extension after re-routing, number of redirected cars, pollution level reduction after re-routing, and total CPU time of problems from Scenario 1 and Scenario 2, in rush hours or "normal" time.

Scenario 1 – Rush hour				
# cars	Δ M	Redir	Δ Poll.	CPU time
20	+42.9%	7	-53%	5.5
30	+0.0%	10	-17%	12.7
40	+27.5%	14	-34%	26.1
50	+0.0%	14	-54%	46.6

Scenario 1 – Normal				
# cars	Δ M	Redir	Δ Poll.	CPU time
20	+33.3%	7	-53%	5.8
30	+0.0%	10	-22%	13.6
40	+0.0%	11	-46%	28.3
50	+54.7%	14	-34%	106.2

Scenario 2 – Rush hour				
# cars	Δ M	Redir	Δ Poll.	CPU time
20	+0.0%	5	-29%	247.3

Scenario 2 – Normal				
# cars	Δ M	Redir	Δ Poll.	CPU time
20	+0.0%	5	-29%	230.7
30	+0.0%	9	-36%	890.8

Our approach is two phased. In Phase 1, plans are generated for individual vehicles without considering pollution constraints whereas capacity and collision avoidance constraints are considered. The main benefit of our centralised approach is that vehicles were distributed throughout the whole road network and thus minimising chance of having some road congested. It is worth noting that it is common in urban road networks that in rush hours main roads are congested while other roads have low traffic. From this perspective, our approach might be beneficial since the whole road network can be utilized and thus main roads can be decongested. The main disadvantage of our approach, at the moment, is its scalability. We believe that the current model can be improved by introducing more constraints (e.g. whether a vehicle can enter a specific road), and/or by "hierarchizing" it.

In Phase 2, pollution constraints are considered and routes re-planned for vehicles whose initial routes violate these constraints. As the results showed Phase 2 takes considerably less CPU-time than Phase 1. Therefore, we might be generally able to re-plan routes for a larger number of vehicles (than in Phase 1) in reasonable time. Importantly, Phase 2 can be independent on Phase 1, i.e., it is only necessary to have initial routes (plans) for the vehicles. It gives a high level of flexibility since these routes can be obtained, for example,

```
10:(release-vehicle vehicle1 en north1 zout) [0.001]
10:(drive vehicle1 north1 zout) [5.000]
15:(drive-through-junction vehicle1 north1 JN ringnw1 zout zout) [1.000]
16:(drive vehicle1 ringnw1 zout) [9.000]
25:(drive-through-junction vehicle1 ringnw1 JW ringsw2 zout zout) [1.000]
26:(drive vehicle1 ringsw2 zout) [9.000]
35:(drive-through-junction vehicle1 ringsw2 JS south2 zout zout) [1.000]
36:(drive vehicle1 south2 zout) [5.000]
41:(exit-vehicle vehicle1 south2 es zout) [0.001]
```

Fig. 5. An example of PDDL plan produced by our approach for a single vehicle, in the Scenario 1 network. `en` and `es` stand respectively for entry/exit point North and South. `zout` indicates the outer zone of the ring road, i.e. not critical from the pollution perspective. `ring` prefix indicates the different sections of the modelled ring road. In the plan shown, `vehicle1` enters the area from the north entry-point, navigate the network by using the western sections of the ring road, and then exit the area from the southern exit-point.

Table 2

Number of cars, CPU time per phase of problems from Scenario 1 and Scenario 2, in rush hours or "normal" time. Fourth column reports the percentage of vehicles that has been redirected in phase 2.

Scenario 1 – Rush hour			
# cars	Phase 1	Phase 2	% of Phase 2
20	4.51	1.00	18%
30	11.57	1.10	9%
40	23.94	2.21	8%
50	43.76	2.40	5%
Scenario 1 – Normal			
# cars	Phase 1	Phase 2	% of Phase 2
20	4.76	0.99	17%
30	12.53	1.09	8%
40	26.74	1.51	5%
50	46.35	59.88	56%
Scenario 2 – Rush hour			
# cars	Phase 1	Phase 2	% of Phase 2
20	230.18	0.48	0%
Scenario 2 – Normal			
# cars	Phase 1	Phase 2	% of Phase 2
20	246.85	0.47	0%
30	877.83	12.92	1%

from SATNAV systems that are becoming more ubiquitous nowadays. It is worth noting that the planning domain model for Phase 2 is similar to the planning domain model used for Phase 1 (it additionally considers pollution constraints), so improvements to the Phase 1 model (as discussed in the previous paragraph) can be also used for the Phase 2 model.

In summary, using centralised route planning for individual vehicles has some benefits. Whereas using current route planning system focus on planning of the shortest or fastest route for an individual vehicle, our approach is able to coordinate multiple vehicles that

go through the controlled urban area and can influence each other. For example, in Scenario 1 a vehicle goes from the Northern entry point to the Southern exit point. When the traffic is low, the best option is to go through the town centre. When the traffic is heavier and pollution constraints apply, the vehicle has to take either the Eastern or the Western ring road. Intuitively, it is useful, especially in heavy traffic, to (evenly) distribute vehicles that go from North to South between the Eastern and the Western ring roads. Centralised planning is able to make such a decision and thus optimise traffic flows in urban areas.

7.1. Directions for Future Work

The experiments demonstrate the extent to which our approach is able to efficiently route vehicles, and re-route them when necessary to respect pollution limits. In Scenario 2 at least, we get an idea of the complexity level that current planning technology can reach. The results also showed that the second phase of our planning process takes a small fraction of the total planning time. This indicates that scalability of our approach can be better if initial routes of the vehicles are known. The initial routes can be calculated by SATNAV in each individual vehicle which is entering the controlled area. This can at least to some extent alleviate the necessity of conducting the first phase since SATNAV systems are becoming more widespread.

The key question for future work is how our approach can be deployed and exploited in the corresponding real-world context. With advanced SATNAV systems that are becoming more and more widespread, it is possible to influence behaviour of drives by altering routes through the urban areas. The high-level idea of how the system can work is as follows. When

a vehicle is approaching the controlled urban area, its SATNAV system shares the current route with the controller of the area. The controller either confirms the current route, or suggest to update the route in order to avoid highly-polluted zones of the controlled area. In the latter case, the SATNAV system updates the current route accordingly to the indications given by the controller.

However, there are some fundamental issues that have to be addressed. Firstly, the drivers have to co-operate with the system, in other words, take routes suggested by the system. Such an issue can be alleviated by mass deployment of autonomous cars. Also, we believe that nowadays majority of drivers rely on the SATNAV and thus it is likely that they will obey routes the SATNAV suggests. Secondly, the model has to be reasonably accurate, i.e., the pollution limits are met (or close to be met) in the real situation, and be efficient, so routes through the controlled urban area can be calculated in a few seconds for all vehicles entering the area.

The model we developed in this paper requires some features (e.g., temporal reasoning, numeric fluents, timed initial literals) that are not supported by many planning engines. Therefore, there is a need for developing advanced planning engines that are able to efficiently handle such features. Furthermore, in the current discrete model we make some assumptions on how the pollution level of different zones changes over time. Namely, the pollution level in a given zone is increased (decreased) when the vehicle enters (leaves) that zone. In reality, the pollution level is subject to continuous change, that depends on several factors, such as the amount of time each vehicle is in a given zone, or the current wind speed and direction. A more realistic modelling of this scenario requires the use of PDDL+ features [11] to model such a continuous change. In particular, PDDL+ continuous processes can be used to model the pollution dispersion due to the wind, or the flows of vehicles through a given road. However, it should be noted that existing systems often support only a limited number of PDDL+ features. For example, COLIN [6] or a SAT-based approach [19] does only support linear processes, which is hardly the case of the pollution dispersion. UPMurphi [8,9] or dReal [2] are possible options, however, the former performs blind search and thus its scalability is very questionable, while the latter does not support exogenous events. Hence, the absence of effective PDDL+ solvers is the major obstacle in using PDDL+ for modelling our scenario.

Clearly, increasing expressiveness of the model often has a detrimental impact to scalability. Increasing computational complexity might not be a major problem since even classical planning (the simplest form of AI planning) is intractable. The problem is, however, in lack of advanced planning engines that can handle non-classical planning problems (as mentioned before in the PDDL+ case). Therefore, when modelling our scenario we should keep in mind that we can use very expressive planning (e.g. PDDL+) on very simple problems. A possible solution to this issue is to "hierarchize" the model. Since the pollution constraints are specified in zones, i.e., parts of urban area, the high-level model can determine upper bounds on the number of vehicles going through particular zones such that pollution constraints are met. Then, we can provide high-level plans for vehicles in terms of which zones they have to pass on their ways from given entry points to given exit points of the controlled area such that the number of vehicles in each zone follows the calculated limits. In the lower level models, we can distribute vehicles among the roads such that the roads are not congested (or their congestion is minimised). Then, the lowest level models can be used for calculating routes for individual vehicles throughout the controlled area. In should be mentioned that higher levels of the hierarchy would refer to macro simulation while lower levels of the hierarchy would refer to micro simulation. An effective hierarchical model would have to besides being accurate consist of particular models that are relatively simple to solve, so the scalability issue is properly addressed, and these particular models "connect" with each other. The latter property is important for obtaining feasible and good quality plans. How to "hierarchise" the current model is an interesting direction for future work, where we indeed believe that we utilize lessons learnt in this work.

Alternatively, there is a possibility to develop and use domain-dependent planning techniques that can be tailored to our application. This might be an efficient approach, but on the other hand, such an approach may be less flexible to changes in requirements.

Decentralised (multi-agent) planning might be another option. Agents (vehicles) might be capable to plan their own routes taking into account the air-quality constraints. This involves some control units responsible for controlling pollution restricted zones that collect data from vehicles passing through the zones and communicate these data with vehicles willing to enter these zones. A possible issue for such an approach might be collision avoidance in junctions.

While, the centralised approach can deal with this issue easily, the decentralised approach will have to consider communication between vehicles willing to pass through the same junctions. It might cause some hardly expectable delays and thus it might be hard to determine when a vehicle leaves the zone.

7.2. Related Work

In recent years, automated planning techniques have been exploited for dealing with different aspects of traffic-related tasks. This is also confirmed by the presence of traffic-related benchmarks in the 2014 edition of the International Planning Competition[9] [22], which is one of the major event of the planning community.

Jimoh et al. [13] introduced the idea of using automated planning in urban traffic control as a planning aid to be used in exceptional circumstances, e.g. in situations where roads within a network of roads become blocked due to some unanticipated incident. This demonstrated that it was feasible to use current planning technology to produce re-routing of vehicles using a simple micro-simulation model. The aim of the work, however, was to provide a supplement to existing tactical UTC, rather than providing a strategic solution at a regional level.

Recently, Vallati et al. [23] developed a PDDL+ model for controlling traffic lights in urban areas. The model considers flow of vehicles rather than individual ones and aims to decrease congestion of roads. It has been demonstrated that this approach is efficient especially in cases when an unexpected event (e.g. traffic accident) occurs.

The work of Xie et al. [26] is also aimed at urban traffic control, though again works at a tactical level. Their research has led to the deployment within an urban area of a real-time, adaptive traffic control system called SURTRAC which utilises scheduling techniques to synchronise a group of traffic light clusters, by viewing intersection control optimisation as a scheduling problem. Botea et al. [1] modelled multi-modal journey planning (i.e. generating plans using more than one mode of transport) as a non-deterministic planning problem, and created a heuristic planner for generating contingent plans to advise a user on using combinations of transport (e.g. walking, getting a bus). While this work is relevant to ur-

ban areas, it is aimed more at personalised planning rather than strategic vehicle routing. Similarly to the approach proposed by Botea et al., Caff et al. [3] addressed the multi-modal journey problem by exploiting planning techniques. In doing so, they relaxed the uncertainty of the problem, and presented a PDDL formulation representing temporally expressive actions encompassing the multiple schedule for an action homogeneously. Their work shows how the majority of the temporal conditions can be translated by exploiting numeric fluents and constraints.

Finally, Shah et al. [18] provided a model for dealing with road traffic accident management, a problem which is orthogonal to the general urban traffic control task.

8. Conclusion

In this paper we examined the current situation in UTC, where management of traffic is largely done at an operational or tactical level, with personal vehicle route planning performed in isolation of other travellers and of the regional UTC centre.

We investigated the feasibility of applying automated planning to UTC at the strategic level, where individual vehicles are taken into account at a regional level, a development foreseen in the near future for organised (potentially autonomous) vehicle routing. This is seen as having a range of advantages for traffic management, such as the enforcing of regional constraints, and the even distribution through, and utilisation of, network capacity. We formulated the problem in temporal planning, and created a two stage solution which calculated routes for vehicles in an initial pass, replanning when air quality limits dictates. The results demonstrated the feasibility of the approach on problems of the complexity demonstrated by the real urban area, though considering larger areas / amounts of traffic will clearly challenge current planning engines.

For the future, we propose to further validate our proposed solution in collaboration with our industrial partners. Lessons we learned from this work will give invaluable insights into developing a hierarchy of models that will alleviate the aforementioned scalability issue that is the most limiting factor for a possible deployment of the model into the real-world traffic management systems. We also plan and explore promising approaches which will extend the method's scope, like the exploitation of decentralised (multi-agent) strategies for planning vehicle routes that might also bring

some benefits into reducing traffic pollution in urban areas.

References

[1] A. Botea, E. Nikolova, and M. Berlingerio. Multi-modal journey planning in the presence of uncertainty. In *Proceedings of the International Conference on Automated Planning and Scheduling, ICAPS*. AAAI, 2013.

[2] D. Bryce, S. Gao, D. J. Musliner, and R. P. Goldman. SMT-based nonlinear PDDL+ planning. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA.*, pages 3247–3253, 2015.

[3] S. C. M. Caff, F. D. Mauro, and E. Scala. A numeric PDDL based approach for temporally constrained journey problems. In *Proceedings of the IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*, 2014.

[4] I. Cenamor, L. Chrpa, F. Jimoh, T. L. McCluskey, and M. Vallati. Planning & scheduling applications in urban traffic management. In *Proceedings of PlanSIG 2014, Annual Workshop of UK Planning & Scheduling Special Interest Group*, 2014.

[5] C. Chong-White, G. Millar, and S. Shaw. SCATS and the environment study: definitive results. In *Proceedings of the 19th World Congress on ITS, Vienna, Austria*, 2012.

[6] A. J. Coles, A. Coles, M. Fox, and D. Long. COLIN: planning with continuous linear numeric change. *J. Artif. Intell. Res. (JAIR)*, 44:1–96, 2012.

[7] A. J. Coles, A. I. Coles, A. Clark, and S. T. Gilmore. Cost-sensitive concurrent planning under duration uncertainty for service level agreements. In *Proceedings of the Twenty First International Conference on Automated Planning and Scheduling (ICAPS-11)*, 2011.

[8] G. Della Penna, D. Magazzeni, and F. Mercorio. A universal planning system for hybrid domains. *Appl. Intell.*, 36(4):932–959, 2012.

[9] G. Della Penna, D. Magazzeni, F. Mercorio, and B. Intrigila. UPMurphi: A tool for universal planning on PDDL+ problems. In *Proceedings of the 19th International Conference on Automated Planning and Scheduling, ICAPS*, 2009.

[10] M. Fox and D. Long. PDDL2.1: An Extension to PDDL for Expressing Temporal Planning Domains. *J. Artif. Intell. Res.(JAIR)*, 20:61–124, 2003.

[11] M. Fox and D. Long. Modelling mixed discrete-continuous domains for planning. *J. Artif. Intell. Res. (JAIR)*, 27:235–297, 2006.

[12] R. Howey, D. Long, and M. Fox. VAL: automatic plan validation, continuous effects and mixed initiative planning using PDDL. In *16th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2004)*, pages 294–301, 2004.

[13] F. Jimoh, L. Chrpa, T. McCluskey, and M. M. S. Shah. Towards application of automated planning in urban traffic control. In *2013 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pages 985–990. Institute of Electrical and Electronics Engineers (IEEE), 2013.

[14] M. Ghallab and D. Nau and P. Traverso. *Automated Planning Theory and Practice*. Elsevier Science, 2004.

[15] T. L. McCluskey and M. Vallati. Extracting information from urban traffic data to improve road traffic management. In *Final Report for Transport Catapult Innovation Voucher*, 2014.

[16] J. C. Miles and A. J. Walker. The potential application of artificial intelligence in transport. *Journal of Intelligent Transport Systems, Volume 153, Institution of Engineering and Technology*, (3), 2006.

[17] S. Edelkamp and J. Hoffmann. PDDL2.2: The Language for the Classical Part of the 4th International Planning Competition. Technical Report 195, Albert-Ludwigs-Universitat Freiburg, Institut fur Informatik, 2004.

[18] M. M. Shah, L. Chrpa, D. Kitchin, T. L. McCluskey, and M. Vallati. Exploring knowledge engineering strategies in designing and modelling a road traffic accident management domain. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, pages 2373–2379. AAAI Press, 2013.

[19] J. Shin and E. Davis. Processes and continuous change in a SAT-based planner. *Artif. Intell.*, 166(1-2):194–253, 2005.

[20] H. Taale, W. Fransen, and J. Dibbits. The second assessment of the SCOOT system in Nijmegen. In *IEEE Road Transport Information and Control*, number 21-23, 1998.

[21] M. Treiber and A. Kesting. Traffic flow dynamics. *Springer*, 2013.

[22] M. Vallati, L. Chrpa, M. Grzes, T. L. McCluskey, M. Roberts, and S. Sanner. The 2014 international planning competition: Progress and trends. *AI Magazine*, 36(3):90–98, 2015.

[23] M. Vallati, D. Magazzeni, B. De Schutter, L. Chrpa, and T. L. McCluskey. Efficient macroscopic urban traffic models for reducing congestion: a PDDL+ planning approach. In *Proceedings of the 30th AAAI conference*, 2016.

[24] Various. Artificial intelligence in transportation. *Transportation Research Circular E-C113, Transport Research Board*, 2007.

[25] R. Vincent and J. Pierce. Self-optimising signal control for isolated intersections. In *Crowthorne: Transport and Road Research Laboratory Research Report*, number 170, 1988.

[26] X.-F. Xie, S. Smith, and G. Barlow. Schedule-driven coordination for real-time trafñÄc network control. In *Proceedings 22nd International Conference on Automated Planning and Scheduling (ICAPS)*, 2012.

Appendix

A. The Domain Model

```
(define (domain utc)
(:requirements :typing :durative-actions
 :fluents :equality
 :timed-initial-literals)

(:types road junction car zone)

(:predicates
 (ready ?car - car ?e - junction)
 (exit ?e - junction)
 (free ?junction - junction)
```

```
(from ?road - road ?junction - junction)
(to   ?road - road ?junction - junction)
(head ?car - car ?road - road)
(tail ?car - car ?road - road)
(exited ?car - car ?e - junction)
(in ?road - road ?zone - zone)
)

(:functions
  (capacity ?road - road)
  (length ?road - road)
  (pollution-limit ?zone - zone)
  (use ?road - road)
  (pollution-level ?zone - zone)
  (pollution-produce ?car - car)
)

(:durative-action DRIVE-THROUGH-JUNCTION
  :parameters (?c - car ?r1 - road
               ?j - junction ?r2 - road
               ?z1 - zone ?z2 - zone)
  :duration (= ?duration 1)
  :condition (and
      (at start (tail ?c ?r1))
      (at start (to ?r1 ?j ))
      (at start (from ?r2 ?j ))
      (at start (in ?r1 ?z1))
      (at start (in ?r2 ?z2))
      (at start (<= (use ?r2)
                 (capacity ?r2)))
      (at start (free ?j))
      (at end (<= (pollution-level ?z2)
              (- (pollution-limit ?z2)
              (pollution-produce ?c))))
  )
  :effect (and
      (at start (decrease
                 (pollution-level ?z1)
                 (pollution-produce ?c)))
      (at end (increase
                 (pollution-level ?z2)
                 (pollution-produce ?c)))
      (at start (not (free?j)))
      (at start (decrease (use ?r1) 1))
      (at end (increase (use ?r2) 1))
      (at start (not (tail ?c ?r1)))
      (at end (head ?c ?r2))
      (at end (free?j)))
)

(:action RELEASE-CARS
  :parameters (?c - car ?e - junction
               ?r - road ?z - zone)
```

```
  :precondition (and
      (ready ?c ?e)
      (from ?r ?e)
      (in ?r ?z)
      (<= (use ?r) (capacity ?r))
      (<= (pollution-level ?z)
        (- (pollution-limit ?z)
        (pollution-produce ?c)))
      )
  :effect (and
      (increase
        (pollution-level ?z)
        (pollution-produce ?c))
      (increase (use ?r) 1)
      (not (ready ?c ?e))
      (head ?c ?r) )
)

(:durative-action DRIVE
  :parameters (?c - car ?r - road
               ?z - zone)
  :duration (= ?duration (length ?r))
  :condition (and
      (at start (head ?c ?r))
   (at start (in ?r ?z))
      (over all (<= (use ?r)
                    (capacity ?r)))
   (over all (<= (pollution-level ?z)
              (pollution-limit ?z)))
      )
  :effect (and
      (at start (not (head ?c ?r)))
      (at end (tail ?c ?r)))
)

(:action EXIT-CARS
  :parameters (?c - car ?r - road
               ?e - junction ?z -zone)
  :precondition (and
      (exit ?e)
      (to ?r ?e)
      (tail ?c ?r)
      (in ?r ?z)
      )
  :effect (and
      (decrease (pollution-level ?z)
              (pollution-produce ?c))
      (decrease (use ?r) 1)
   (not (tail ?c ?r))
      (exited ?c ?e))
)
)
```