Proceedings of the Twenty-Seventh International Conference on Automated Planning and Scheduling (ICAPS 2017)

Embedding Automated Planning within Urban Traffic Management Operations

Thomas L. McCluskey and **Mauro Vallati**
School of Computing and Engineering
University of Huddersfield, UK
{t.l.mccluskey,m.vallati}@hud.ac.uk

Abstract

This paper is an experience report on the results of an industry-led collaborative project aimed at automating the control of traffic flow within a large city centre. A major focus of the automation was to deal with abnormal or unexpected events such as roadworks, road closures or excessive demand, resulting in periods of saturation of the network within some region of the city. We describe the resulting system which works by sourcing and semantically enriching urban traffic data, and uses the derived knowledge as input to an automated planning component to generate light signal control strategies in real time. This paper reports on the development surrounding the planning component, and in particular the engineering, configuration and validation issues that arose in the application. It discusses a range of lessons learned from the experience of deploying automated planning in the road transport area, under the direction of transport operators and technology developers.

Introduction

Traffic Operators use traffic control systems in large urban areas to perform the crucial role of tackling road congestion and minimising traffic related environmental effects. Conventional road traffic signal management techniques, such as traffic-responsive systems like SCOOT (Taale, Fransen, and Dibbits 1998) and SCATS (Chong-White, Millar, and Shaw 2012), or fixed time light strategies optimised using historical data, work reasonably well in normal or expected conditions. Currently, software systems in the urban traffic management area tend to be based around a syntactic, product specific integration of data, which at best share data externally at a relational database level. They have a vertical systems design, and though eminently configurable within the range of their function, they are not integrated at a horizontal level with the overall function of the urban management centre where they operate. Within urban traffic management and control (UTMC) operations this perpetuates the status quo of recurrent system replacement, rather than system evolution. The context of the novel application of planning we describe in this paper is developing semantic technology in order to better capture and exploit real-time and historical urban data sources, while pursuing a higher level of data integration. We aim to make UTMC systems less brittle and more adaptable by raising the level of traffic control software integration via semantic component interoperability. In doing this we have the longer-time aim of utilising an *autonomic* approach to UTMC in particular, and road transport support in general, as developed in the EU's transport network ARTS [1]. Results of the Network supported the idea of the construction of a semantic systems level for UTMC, consistent with previous work on integrating decision support within semantic technologies(Blomqvist 2014; Antunes, Freire, and Costa 2016). Among the benefits of a higher level of information integration are a more joined up UTMC capability, where the flexibility of a knowledge level representation gives the opportunity to use general AI techniques such as automated planning to provide a more intelligent approach to tackle UTMC issues.

Within this context, we present a novel AI Planning application addressing a well known functional drawback of established UTMC tools referred to above: they do not work adequately in the face of exceptional or unexpected conditions affecting urban regions (containing many hundreds or thousands of road vehicles). In these cases, Transport Operators may struggle to find a strategy intervention tailored to solve the unexpected situation. Creating such strategies –which involves changes to traffic signal timings over a period of time– is a manual task that may take several days or weeks, and it is therefore infeasible to hand craft one in real-time. For example, transport operators may want to reduce traffic concentrations in a targeted urban area to ameliorate effects of predicted road traffic pollution; or optimise the flow of saturated road links due to an emergency road closure; or produce a strategy to deal with a forthcoming complex situation (e.g. optimising the light timings to deal with the combination of a concert, a football match and some emergency roadworks).

The planner used in the project was the domain independent planner UPMurphi (Della Penna et al. 2009), which inputs models in PDDL+ (Fox and Long 2006). To produce a working executable of UPMurphi which generated UTM strategies in real-time, we had to perform several cycles of iterations over the engineering of the PDDL+ mod-

Copyright © 2017, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

[1] EU's COST Action 1102 "Autonomic Road Transport Support Systems" (ARTS) www.cost-arts.org

els, starting from the model proposed in (Vallati et al. 2016). The quality of the strategies output from the planner was evaluated firstly by hand inspecting the strategies to check that they were "sensible". In this case the strategies were inspected to check they embodied common sense plans in them. Secondly, their effect was compared against optimised strategies derived from historical data by simulating their execution using both the AIMSUN micro-modelling software[2], and the off-the-shelf SUMO ("simulation of urban mobility") (Krajzewicz et al. 2012) micro-modelling software. In each, transport engineers compared the results of simulations using both automated planning generated strategies, and optimised strategies derived from historical data. In both these simulators, run by different members of the consortium, the planner-generated strategies produced sufficient savings to convince the consortium to aim to adopt AI planning within a product to generate strategies for exceptional events in busy urban areas. On the other hand, the study highlighted several challenges to be overcome before a fielded implementation could be achieved, and resulted in a range of lessons learned. The implications of taking this approach are, we believe, of a step changing nature for UTM. Currently we are engaged in the adaptation of the system to deal with a wider range of effector actions (rather than only control signal change) and a more sophisticated, flexible goal language.

Context

The work reported was one of the deliverables of a funded project within a consortium consisting of the University of Huddersfield, a major transport authority (Transport for Greater Manchester – TfGM), a large technology supplier (BT), and two SMEs: KAM Futures and Infohub Ltd. KAM Futures performed the project management, Infohub performed independent validation of the results, TfGM supplied the area, the problem and raw data, and BT's systems were used to integrate the data. The overall aims were in the context of developing smart city technology, taking advantage of the wide range of data available in a modern urban area. In particular the project focussed on exploiting real-time data sources to pursue better congestion control and air quality management. The consortium represented the diversity of the data using semantic technologies, to enable the integration into a unified form through a common, high-level vocabulary. Semantic technologies enable the enrichment of data sources, linking them with additional information, thus providing context and aiding data cleansing. Within the context of this development, we targeted the better control of traffic in exceptional circumstances as an area where we could show the benefit of semantically-enriched data.

Approaches to Region-wide Traffic Control

Generating a detailed strategy of interventions, such as changes to traffic signal timings over a period of time, to manage an emergency situation in *in real time* is considered to be beyond the capacity of human operators. Transport operators need the ability to produce regional strategies in real time which will deal with abnormal or unexpected events such as road closures. These cause huge delays and decreased air quality because of excessive congestion and stationary traffic. The existing conditions and set of corrective goals required to deal with these events are so varied that detailed strategies are impossible to draw up a priori in a large, dense urban area. Approaches to region-wide traffic control has been trialled using model predictive control (MPC) strategies and optimisation (Lin 2011; Dotoli, Fanti, and Meloni 2006; van den Berg et al. 2004). This line of research uses a control theory approach which, given an adequate dynamical model, can be used to derive a solution that can give continuous responses to changing inputs. Under changing state conditions, researchers have designed MPC algorithms which can continuously adjust the controlled features (here signal timings) to optimise some goal in real time. This approach tends to be less flexible than a goal-directed AI approach, however, as a solution needs to be designed, implemented and tuned using a specific model of traffic flow and a specific objective function. Additionally it is less scrutable, as it generates strategies over a restricted time horizon. For instance, there is a need in UTM to be able to achieve more focussed or detailed goals than simply minimising delay - in particular operators may want to generate strategies that avoid problems with air quality caused by traffic congestion.

While there are several examples of the application of general AI techniques to road traffic monitoring and management (Various 2007; Miles and Walker 2006), the trial of UTM systems embodying an AI planning engine within a real urban traffic management centre, with an evaluation performed by transport operators and technology developers, is novel. On the scheduling side, however, the SURTRAC project utilises a distributed scheduling system which controls traffic signals in urban areas (Xie, Smith, and Barlow 2012). In SURTRAC, each intersection is controlled by a scheduling agent that communicates with connected neighbours to predict future traffic demand, and to minimise predicted vehicles waiting time at the traffic signal. It is currently being trialled in Pittsburgh, USA, with its distributed approach suggesting good scale-up but less goal flexibility than if utilising a centralised AI planner. Two recent lines of research showed the feasibility of using AI planning to generate actions to deal with unexpected circumstances in complex urban traffic control situations[3]. Gulic et al's system (Gulić, Olivares, and Borrajo 2016) involves joining together a SUMO simulator (Krajzewicz et al. 2012) to an AI Planner, via a monitoring and execution module called the "Intelligent Autonomic System". The planning representation was done using PDDL 2.1 (Fox and Long 2003b), with no explicit representation of vehicles in the planner. Instead, traffic concentrations on road links are represented by relative density descriptors, such as very-low, low, medium and high. Traffic light change actions are enumerated to cover all

[2]https://www.aimsun.com/

[3]These two achievements were recognised in October 2015 as they were joint winners of the the Second COST ARTS Competition. https://helios.hud.ac.uk/cost/comp2.php

the ways that a particular configuration would effect the arrangements of road links. By abstracting away from explicit counts of vehicles, the system can deal with regions containing thousands of vehicles. Also, the close coupling with SUMO demonstrates the use of monitoring and replanning very effectively, and allows exhaustive testing of the system under sets of disturbances (vehicle influx, road closures).

The second work (Vallati et al. 2016) was inspired by works such as (Lin 2011; van den Berg et al. 2004), where traffic is modelled using "flows", and then analysed through model-predictive controllers. Vallati et al. exploit PDDL+ for encoding a flow model of vehicles through traffic-light controlled junctions. The length of traffic light phases are under the control of the planner, that can decide to prioritise some traffic flows, in order to reach specified goals (a phase determines which of the flows through that junction are on and have traffic flowing). Goals are specified in terms of numbers of vehicles desired on some critical road links. Encoded problems are then solved using the UPMurphi solver, extended with domain-specific heuristics. Their experimental analysis demonstrated that UPMurphi could solve traffic problems containing thousands of vehicles, in response to exceptional conditions. They showed the efficacy of the resulting strategy by comparing its execution against fixed time and reactive approaches, using SUMO.

Description of the Project

The initial phase of our collaborative project concentrated on the semantic enrichment of the data. The raw data was taken from transport and environment sources and integrated into a *Data Hub*, using the RDF triple format and a data ontology. The method was to take real time feeds and process them until they produced logical facts about a traffic scenario, which could serve at part of an initial sate of a AI planner. Figure 1 depicts the abstract system architecture used to test the generation and operation of the control strategies. To work towards that, however, the data enrichment and strategy generation had to be tested in a real scenario, hence rather than taking in real-time current data, we adjusted the system so that what would be translated into the current state would be from *historical* data. This would allow checking the performance of the system against the observed performance from historical data, in order to evaluate it off-line.

The Main Data Sources

As a basis for exploring exceptional or emergency traffic conditions, we chose to use historically averaged traffic data from a time/day when the road links were most congested: morning rush hour, between 8am and 9am on a non-holiday weekday. The main data source was the Saturn system (Simulation and Assignment of Traffic to Urban Road Networks[4]). From this and other transport engineer documentation records our partners extracted, for the selected region within the urban area, the following:

1. the topology of the road links (a link is a uni-directional part of a road between two junctions).

[4]http://www.saturnsoftware.co.uk/saturnmanual

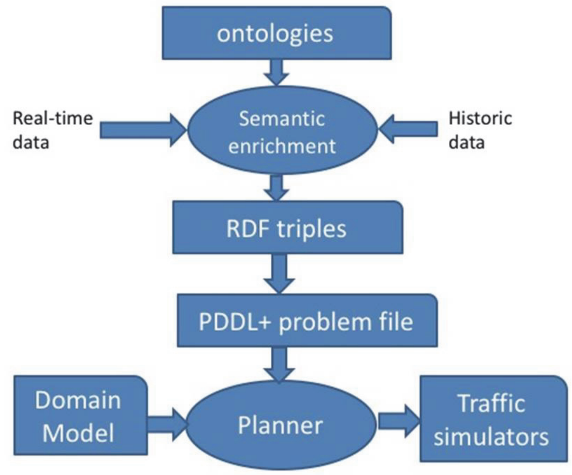

Figure 1: An Abstract View of the System Test Architecture

2. The vehicle capacity of all the road links. This is given in numbers of "passenger car units" –PCU– which takes into account the differing size of vehicles.
3. The average traffic flows between links in number of PCU's per second. This number represents the number of vehicles flowing through a particular junction at a certain time of day, when the corresponding traffic signal phase is green. A special case of this are flows in and out of boundary junctions.
4. The traffic signal position, phases of signals, minimum and maximum time that a signal phase can be set for.
5. Intergreen timings between each of the phases of the signals. Intergreen intervals are used between two traffic light phases for clearing the intersection from vehicles, and allowing pedestrian crossings. Their duration is dependent on the phase and junction, and varies between 0 and 25 seconds. Such length is calculated accordingly to the size and shape of the intersection, the expected speed of vehicles, and the expected time needed by turning vehicles for clearing the junction. All flows are considered off during the intergreen.
6. The state of the network at a certain instance, that the strategy is expected to start from: number of vehicles on each link, and the settings of each signal phase.

These data items made up the initial state of a problem file in planning terms. The goal language of the planner is what the actions in the domain model can effect. In this case the goals are made up of numerical constraints denoting predicates on the occupancy levels of road links.

Choice of Technology

The previous work on using AI planning for traffic control established that the representation of traffic through the Network needs to be performed at a macroscopic level to cope with large volumes of traffic. Our main choice was between using a fixed discretization of the link density (using a sequence of density descriptors) with a discrete PDDL representation of actions (Gulić, Olivares, and Borrajo 2016), or using a numeric representation of link density and explicit continuous flow processes in PDDL+. The length of the project constrained us to recent work that had proved itself in a similar scenario: an alternative choice, in the longer term, might be to investigate the use of non-PDDL models, such as timelines (Cesta, Fratini, and Pecora 2008).

An advantage in using the continuous PDDL+ - based approach over the classical PDDL approach was its accuracy, i.e. the representation contained exact counts of vehicles, and modelled continuous change of vehicle numbers on road links during green times. Although a planner reasoning with PDDL+ would in the end need to discretize, the advantage of PDDL+ is that this discretization level does not need to be decided on in advance. We made the decision to follow the PDDL+ approach on the basis of the accuracy and granularity of its continuous representation: the PDDL+ encoding was closer to the representation used in traffic simulation as shown by the data sources above. Also, PDDL+'s representational accuracy supports the extension of the system to incorporate other available "interventions" (such as variable speed limits). Typically these interventions are specified by their impact on the traffic in terms of flows of vehicles that are affected. One drawback is that there are very few PDDL+ planners available because of the higher complexity of mixed discrete / continuous planning.

Engineering the PDDL+ Model

Populating the initial state and goal for the planning tasks required by our collaborators using the PDDL+ model inherited from the earlier research (Vallati et al. 2016) proved infeasible [5]. As well as using UPMurphi, we tried other planners capable of inputting a form of PDDL+, such as DReal (Bryce et al. 2015) and Popf (Coles and Coles 2014), but with no success. Given the provided output, it looked reasonable to assume they had memory related issues, due to the large number of PDDL+ processes and events involved. To overcome this problem, we also experimented with UPMurphi using larger RAM, by using HPC cluster facilities. This provided no significant scale up. The input scenarios contained hundreds of road links and vehicle flows through complex junctions, which was clearly beyond the capability of the available planning technology using our existing model. For instance, junction 1202 in Figure 2, contains a cycle of seven traffic signal phases, where each phase is defined by a different set of traffic flows being active.

Hence, we re-engineered the PDDL+ model in a more efficient way. This entailed minimising the size of groundings produced by the planner at compile time, in particular the number of processes that were grounded. We had sufficient success with the re-engineered PDDL+ that we were able to represent features that made the model more realistic, such as the addition of intergreen processes to all phases of all the junctions, and the introduction of processes representing

[5] UPMurphi's compilation resulted in a segmentation fault, as the grounding of the problem proved too large

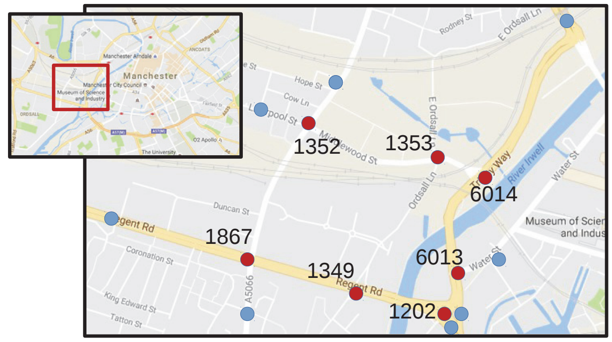

Figure 2: The Modelled Area (large picture) and the position of the modelled area with regards to the city centre (small picture, red-limited area). Blue points indicate the sources (destinations) of incoming (outgoing) vehicles.

roadworks.

We adopted a systematic approach, starting from a simple network, and proceeded to expand the network while iteratively testing the function of the plan generation capability. The final modelled region, which was judged to be large enough for trialling purposes by the consortium, is shown in Figure 2, and abstracted in Figure 3. Junctions are identified by "Saturn" numbering. Directed links are identified by the concatenation of the names of their start and end junctions.

- The model consists of 15 junctions and 34 road links: 7 junctions are controllable junctions (in red) and the 8 outer junctions are not modelled as controllable, but act as a boundary to the region.
- The controllable junctions have 69 controllable flows in total between them. For instance, junction "1349" has 12 flows, as a vehicle has a choice of 3 exits when entering from any of the 4 directions.
- Each controllable junction has between 2 and 7 variable-time signal phases.
- Each traffic signal phase has a fixed intergreen period between its end and the start of the next phase.
- Roadworks could be placed in links as follows: they were modelled as simple junctions with 2 flows, one off and one on at any point in time. The intergreen would vary in size depending on the size of the roadworks. A similar model could be used for pedestrian crossings. In both cases, however, the introduction would add two extra links and two extra process flows to the total.
- Boundaries to the region are modelled as a single vertex. Each road section connected with the boundary has an assigned traffic flow, which corresponds to the number of vehicles entering or leaving the region per time unit.

This configuration was at the edge of the limit for the final version of the planning system (engineered PDDL+ model and UPMurphi) –to add extra roadworks, for example, we would need to abstract outgoing process flows (this would abstract any limit on the volume of traffic leaving via an "out" link) leading from the region. Where the abstracted

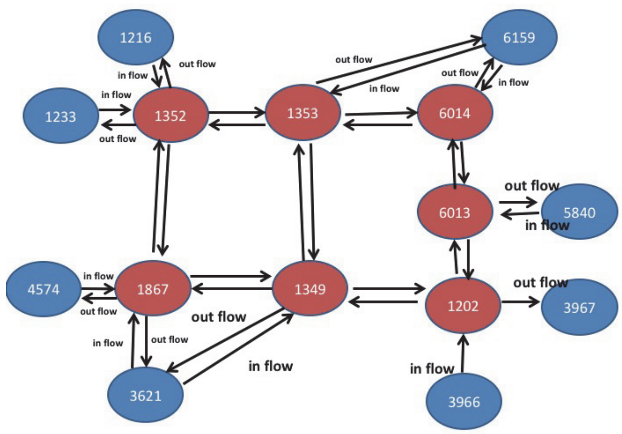

Figure 3: Modelled Flows and Links in the Target Region: Abstracted View. Blue vertices are modelled as part of the *outside* region.

outgoing process flows were not near road links involved in goals, this would have little or no effect on the result.

As a comparison, beside all the extensions needed for modelling intergreens, in/out-flows, etc., the largest test presented in previous work(Vallati et al. 2016) involved 9 junctions with between 2–3 light phases, 21 links, and 27 controllable flows.

A Description of the PDDL+ Model

A region of the *road network* can be represented by a directed graph, where edges stand for *road sections* and vertices stand for *intersections*. One vertex is used for representing the *outside* of the modelled region. Intuitively, vehicles enter (leave) the network from road sections connected with the outside. Each road section has a given maximum *capacity*, i.e. the maximum number of vehicles that can be, at the same time, in the road, and the current number of vehicles of a road section, which is denoted as a *queue*. In our formulation, we consider only intersections that are controlled by traffic lights, as they are those under the control of traffic controllers.

Traffic in intersections is distributed by *flow rates* that are defined between each couple of road sections. Given two road sections r_x, r_y, an intersection i, and a traffic signal phase p such that r_x is an incoming road section to the intersection i, r_y is an outgoing road section from i, and the flow is active (i.e., has green light) during phase p. Flow rates stand for the maximum number of vehicles that can leave r_x, pass through i and enter r_y per time unit. For the sake of simplicity, we assume that vehicles going in the same direction move into the correct lane, thus not blocking other vehicles going in the different directions.

Intersections are described in terms of a sequence of traffic signal phases. Specifically, intersections *contain* a signal phase, and phases are connected using a *next* predicate. According to the active traffic phase, one (or more) flow rates are activated, corresponding to the traffic lights that are turned green. For each phase, the *minimum* and *maximum* phase length is specified. Within this range, the planner can decide whether to stop the phase currently active, or not. Between two subsequent signal phases, an *intergreen* interval is specified. The model was encoded so that some intersections can be declared as not under the control of the planner, by introducing a *controllable* predicate. Intersections are regulated using the following PDDL+ constructs:

- An action *switchPhase(p,i)* is used by the planner for stopping the currently active phase p in intersection i, if the intersection i is controllable, and minimum phase time of p (increased by the *keepPhase* process) has been reached. This action is the "tool" allowing the planner to affect the traffic flows. The only effect of this action is of activating a *trigger* for the intersection i.
- An event *triggerCatcher(p,i)* is activated when the trigger of intersection i is activated, during the traffic phase p. The event stops the current traffic phase, resets the *phaseTime* to zero, and turns on the next intergreen phase.
- A process *keepPhase(p,i)* is used for "keeping" the traffic phase p on intersection i active, and measuring the time the phase is kept on. This process is started when the *activePhase* predicate of p is set to true, and automatically stops when the phase time has reached the maximum allowed value, or the phase has been de-activated by the planner. Similarly, a *keepIntergreen(p,i)* process is used for keeping the intergreen, after traffic phase p, active.
- An event *maxPhaseTimeReached(p,i)* is triggered when the phase p of intersection i reaches the maximum allowed time (the *keepPhase* process). The event activates the trigger predicate of i (in the same way as the *switchPhase* action does). A corresponding *maxIntergreenTimeReached(p,i)* is used for stopping an intergreen phase when the maximum time has been reached.
- A process *flowPhase(p,r1,r2)* is activated when the *keepPhase(p,i)* process is active. It is used for moving vehicles from road *r1* to road *r2* at the given flow rate. If there is no vehicle on *r1*, or *r2* is full (the number of the vehicles is the same as the capacity of *r2*), the process is stopped.

Figure 4 shows the PDDL+ encoding of the *switchPhase* action, the *triggerCatcher* event and the *keepInterGreen* process. A road section connected with the outside area can either have incoming or outgoing flows of vehicles. In the first case, vehicles from the outside region are entering the modelled area through the section, otherwise the road section is used by vehicles that are leaving the modelled area. Each road section connected with the outside has a corresponding *entering* (*leaving*) rate, that indicates the maximum flows of vehicles, in either direction, that can be served by the section. Vehicles that are going to enter the network are queued in the corresponding incoming road section, unless the road section is full. Flows of vehicles entering the network can be activated, or deactivated, using Timed Initial Literals (Fox and Long 2003a).

```
(:action switchPhase
    :parameters (?p - phase ?i - intersection)
    :precondition (and
        (controllable ?i)
        (activePhase ?p)
        (contains ?i ?p)
        (> (phaseTime ?i) (minPhaseTime ?p) ))
    :effect (and
        (trigger ?i) ))

(:event triggerCatcher
    :parameters (?p - phase ?i - intersection)
    :precondition (and
        (trigger ?i)
        (activePhase ?p)
        (contains ?i ?p))
    :effect (and
        (not (trigger ?i))
        (not (activePhase ?p))
        (activeIntergreenAfter ?p)
        (assign (phaseTime ?i) 0) ))

(:process keepInterGreen
    :parameters (?p - phase ?i - intersection)
    :precondition (and
        (activeIntergreenAfter ?p)
        (contains ?i ?p)
        (< (interGreenTime ?i) (interGreenLimit ?p)) )
    :effect (and
      (increase (interGreenTime ?i) (* #t 1 )) ))
```

Figure 4: Part of the developed PDDL+ model.

Evaluation

The consortium tested the software configuration on a range of classes of problems: to clear saturated road link(s) as soon as possible; to clear a region as soon as possible; and to clear a saturated road link with nearby road works. The idea behind the tests was that, when a problem was spotted, the normal fixed time strategy would be turned off, and replaced by the planner-generated strategy. When the plan achieved the goal, the fixed time strategy would be turned back on. All the goals in the tests below have the format: `X1 < N1 & X2 < N2 ...` , where `Xi` is the road link occupancy, and `Ni` is the desired occupancy level. Hence, in this context, clearing road links equates to lowering the occupancy to less than a certain –predefined– value.

UPMurphi is configurable using two types of heuristics (as described in (Vallati et al. 2016)). One allows certain preconditions to be put on action choices, and another specifies the goal heuristic. In a nutshell, the first heuristic gives some guidance about when it is more promising to apply available actions, while the goal heuristic provides an estimation of the distance of the current state from a goal state. For all the tests with the configuration below, we specified only the goal heuristic, as the other heuristic did not seem to play a clear role in the success of the tests. The goal heuristic amounted to minimising the values of link occupancy in the goal expression (`X1`, `X2`, ...). The tests that completed generated strategies composed of between around 30 to several hundred actions with a makespan of several minutes, in less than 30 seconds, on a 2.5 Ghz Intel Core 2 Quad processors with 4GB of memory made available and a Linux operating system. The strategies were composed of sequences of the instantiation of the single action in the PDDL+ model: to move on a traffic a signal phase on to the next phase (respecting intergreen intervals, of course). As the simulators AIMSUN and SUMO rely on non-deterministic components for simulating traffic evolution, simulations were run five times.

Validation of generated strategies consisted of:

1. Comparison with what would be expected in a “common sense” solution, by the visual inspection of the planner-generated strategies.
2. Validation that the planner’s internal simulator (the VAL tool(Howey, Long, and Fox 2004)), the micro simulator SUMO and the micro simulator AIMSUN give similar and consistent results to each other when run with the fixed and planner-generated strategies. If the PDDL+ model was correct/sufficiently accurate, then the planner’s generated strategy was guaranteed to solve the goal when executed; and if the independent simulation tests showed that it does not, then we would conclude that the planner’s PDDL+ model was not correct or sufficiently accurate.
3. Comparison of the effects of the planner-generated strategies with a fixed strategy which had been optimised for the time of day by Transport Engineers. Clearly, this fixed strategy was not generated to deal with the exceptional event, but this was assumed a good comparison as that strategy would be operational when an event occurred.
4. Estimates of savings in terms of tail-pipe emissions.

Results

The initial test scenario was used to investigate the connection between the planner’s internal traffic model (based on flow values), and the models encoded in the SUMO and AIMSUN microsimulation packages. It was inspired by a possible scenario. Assume there was an extreme vehicle build upon a link (in our case 3966_1202) entering into the region, and the consequent air quality implications around the link were unacceptable. This problem would be to clear the link as soon as possible. It is formalised by assuming the link contains at the initial state an unexpectedly large number of vehicles (in this case, 300), and the goal state is to reduce the number to less than 10.

The common sense, approximate strategy to solve this kind of problem would be as follows. At the junction that the link leads to (in this case 1202) called the “primary junction”: give maximum green time to those light phases which allow vehicles to leave the link, and minimise those phases which do not, so that the lights will quickly cycle back to the phases letting out traffic. At the junctions that lead off from the primary junction (in this case 6013 and 1349): give at least enough green time to the links leading in from the primary junction to make sure that the links do not get congested and the increased level of traffic can go through them smoothly. This strategy may have to be repeated through junctions further away if necessary. To visually inspect the

Config.	Initial Test	3 Links	Saturated	Roadworks
Fixed	430	2845	1985	815
Planned	370	1500	1400	630

Table 1: Planned vs Optimised Configurations

quality of the strategy, members of the consortium (including transport operators) checked the generated solution to the first test: it was indeed close to this common sense solution, showing the repeating loop of green-lighting referred to above, as one would expect from a hand-generated solution.

Considering the simulation, the traffic models (AIMSUN and SUMO) were run independently by the transport authority and the SME Infohub, respectively, using the planner-output strategy and the fixed optimised strategy. In the first test, after validating that the simulations were fairly consistent, the reduction in time to clear a junction using the planner-output strategy was approximately 20% using the simulations. AIMSUM and SUMO gave similar results to each other, but tended to produce slightly longer times to clear congestion than the planner's own simulation, and tended to give better results for the planner-generated strategy than the planner's own simulation. Videos of the AIMSUN planner-generated[6] and fixed optimised[7] strategies are online. This comparison shows a slightly longer makespan than the planner's internal simulator on both configurations (compare with results in Table 1, first column).

The results of the full range of tests are shown in Table 1: "3 Links" is to clear congestion from 3 road links leading into the junctions 1867, 1349 and 1202 shown in Figure 3, where an extra 600 vehicles are entering as a result of a disturbance in another region; "Saturated" is where all the links in the region of Figure 3 are at capacity, "Roadworks" is the same configuration as the initial test, but with roadworks severely limiting flow between junctions 1202 an 1349. In each case the figures in Table 1 are the times in seconds to decongest the roads involved using the optimised fixed strategy (first row) or the planner-produced strategy (second row) using the planner's simulator. All show a marked reduction in the case of the planner-generated strategies. A common sense, approximate strategy to solve the more complex problems (columns 2-4 in Table 1) is much more difficult to formulate than for the initial test (and hence one of the reasons for automation). However, a sensible pattern appeared to exist in the planner-generated strategies, to green light the correct junctions.

Reductions in Tail-pipe emissions

The consortium investigated a simple approximation to the amount of emissions that the vehicles produced in both the planner-generated strategies and the default ones. The assumption we use is that clearing a junction (in particular, reducing it from a level of saturation as quickly as possible) will lead to a reduction in tail-pipe emissions, and hence overall pollution. We illustrate this by deriving the expected emission reduction along the strategic link 3966_1202 of our first test. The potential emission reduction achieved by the strategy has been calculated, approximately, as follows:

Emissions Reduction = (Y - X) * (E1 - E2)

- Y = Time taken for the goal to be reached by the normal strategy provided that the link is congested;
- X = Time taken for the goal to be reached by the planned strategy provided that the link is congested;
- E1 = The Emission expected given that the model is congested and the normal strategy is being used;
- E2 = The Emission expected given that the model is not congested and the normal strategy is being used.

E1 and E2 emissions have been provided from a "capacity" case (E1) and a normal case (E2). For both, default fixed timings were used in the AIMSUN model. In the 'normal' case, Saturn demand for the 3966_1202 link was used. The overall effect of applying the planner-generated strategy was measured using the TRACI (Bare 2011) environmental impact assessment tool built into SUMO.

As well as estimating the emissions reduction in the link referred to in the goal, the emissions reduction from the overall effect of applying the strategy to the model given that certain links carry more weight (e.g. those that are in an air quality management zone) was calculated. The emissions around the link to be cleared were calculated to drop by 5%, whereas the overall drop over the region was 2.5%. It is worth stressing that these results are preliminary, however, with more testing to be done to accurately determine the effect on air quality levels.

Discussion

At the end of the project, the consortium was convinced enough by the results of using AI planning as to want to pursue field trials and potentially a software product. Using a domain independent planning engine was, in the end, adequate for showing the proof of concept of a planning-driven approach to the solution of a real problem. The method of implementation would incorporate monitoring and re-planning if needed, as the plan generation speeds during the trials made re-planning in real time feasible.

The main advantage of the AI planning approach appears to be its ability to generate a useful, readable strategy in real time to meet the needs of a new unexpected situation. This relies on the flexibility of the PDDL+ approach, as well as the speed of a planner in dealing with the specified goals. The ability to generate complete initial states and triggered goals in real time (and so be responsive to a detected event) was also a persuasive factor for the consortium. Also, new effectors such as the exploitation of variable speed limits or variable message signs (affecting traffic flows) can be added to the planner's domain model modularly, meaning that new strategies generated will contain instances of those effectors if they help achieve a goal.

In contrast, demand driven, traffic responsive controls such as SCOOT (Taale, Fransen, and Dibbits 1998) are aimed at handling cycle-to-cycle changes in demand. In response to large changes in traffic flows, SCOOT would gradually adapt and adjust the traffic signal timings, as compared

[6] https://goo.gl/st149L

[7] https://goo.gl/dNzByU

to the immediate adjustment made by the planner-generated strategies. SCOOT is dependent on its own local data sensors –the inductive loops embedded in the road surface– and it cannot respond to varied, regional goals. In contrast, SimplyfAI's planning system has the benefit of enriched and integrated sources of data, which gives it the immediate advantage of higher quality data and a wider data view.

There were many lesson learned from the challenges we had to overcome and those still outstanding: we summarise the main ones below.

- *Problems with the data:* the "meaning" of the flow values obtained from Saturn were not as we had anticipated them. While we expected these values to be the maximum number of vehicles (in PCU) that could flow assuming a queue was formed in the oncoming link, in fact they denoted the flow averaged for the particular time of day.
- *Problems with adequacy of our representation:* the PDDL+ model embodied several assumptions that made it inaccurate. Firstly, it assumed that as soon as vehicles enter into a link, they are queued at the next. Secondly, there were breaks in links that we did not model, such as roundabouts and pedestrian crossings.
- *Problems in complexity measures:* the field trial demonstrates the crucial importance of estimating accurately measures of the trial (region) size a priori, and acquiring planning machinery which would cope with that. In our case the measure of "number of vehicles in a region" was not as relevant for determining limits as other factors such as the total number of links, and consequently we were over optimistic in our expectations.
- *Problems with understanding the chosen planning engine:* several classes of scenarios when input to UPMurphi would not yield results. For example in the first class of tests, instead of raising the occupancy of a road link to 300 (well above its maximum value), the normal approach would be to increase greatly the *flow-in* value. In such a scenario we were unable to obtain an output. From extensive tests it appeared that if the goal was one in which actions could make immediate progress towards, then an answer would be extracted. On the other hand, if UPMurphi was initiated in a heuristic "canyon" it was likely that no result would be output. Given a fixed number of vehicles to start, however, the path to the goal heuristic (minimise occupancy on 3966_1202) was monotonic, which seemed to guarantee a resulting plan.
- *Problems with a purely goal directed strategy:* while the effect of a generated plan was successful for solving the goal, other junctions through the region were not optimised. In fact the light signals in other junctions in the region were all left to run to maximum (actions to move them on a phase would not be taken unless it helped towards solving the specific goal). Also, goals such as the *maintenance* of a value are desirable in some situations. For a future system, a richer goal language is needed.
- *Problems in joining up the technology:* when engineering a planning component into a larger application we naturally use the high level interface input language – in this case PDDL+. Components of the initial state are assembled automatically from the Data Hub. In our application, a different team had responsibility for producing the tool which assembles PDDL+ elements. As this work was chronologically scheduled first, there was an over-commitment to a particular target representation. The work following on from this involved configuring the planner, and required changing the PDDL+ models many times. Hence the coding of a PDDL+ assembling tool, and any work on the end to end effectiveness of the system, would need to be completed only after a final PDDL+ representation had been agreed upon.

Conclusions

In this paper we described the operation and results of a collaboration between a transport authority, academics, a large technology provider and two SMEs, which included in its remit the use of AI Planning to generate strategies of traffic light changes to achieve desired goals in the presence of exceptional events. The trials involved using historical data describing the traffic in the region of a large city. The strategies (timing changes of traffic signals) output were judged to be useful for dealing with exceptional situations, using both visual inspection to check that they were *sensible* and simulating their execution using two different traffic modelling software packages AIMSUN and SUMO. We believe that this is the first successful demonstration of AI Planning technology to create useful strategies for UTC where the overall control for the region chosen, the nature of the data feeds and the validation of the end result was largely in the hands of non-academic stakeholders. On the other hand, the success is limited by several factors discussed above. While the results of the plan generation component seem acceptable to the stakeholders, a certain amount of scale-up is required in terms of traffic area covered, and granularity of representation, before the project enters its next phase.

Acknowledgements

This work was supported by the ESF-funded COST Action 1102, and UK NERC grant NE/N007239/1 issued through Innovate UK. We are indebted to the members of the SimplyfAI consortium, and in particular Grigoris Antoniou and Ilias Tachmazidis who designed the semantic enrichment engine for the traffic data, and to Sam Corns and Louis Burrows who conducted the comparative simulations with AIMSUN and SUMO respectively. We would like to thank Bart De Schutter for hosting Mauro Vallati's STSM in 2015, and Daniele Magazzeni for introducing us to the implementation and configuration of UPMurphi. Finally, we acknowledge the help of members of the COST ARTS Action too numerous to name, who explained to us the finer points of Transport Engineering.

References

Antunes, F.; Freire, M.; and Costa, J. P. 2016. Semantic web and decision support systems. *Journal of Decision Systems* 25(1):79–93.

Bare, J. 2011. Traci 2.0: the tool for the reduction and assessment of chemical and other environmental impacts 2.0. *Clean Technologies and Environmental Policy* 13(5):687–696.

Blomqvist, E. 2014. The use of semantic web technologies for decision support–a survey. *Semantic Web* 5(3):177–201.

Bryce, D.; Gao, S.; Musliner, D. J.; and Goldman, R. P. 2015. Smt-based nonlinear PDDL+ planning. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, 3247–3253.

Cesta, A.; Fratini, S.; and Pecora, F. 2008. Unifying planning and scheduling as timelines in a component-based perspective. *Archives of Control Science* 18(2):231271.

Chong-White, C.; Millar, G.; and Shaw, S. 2012. SCATS and the environment study: definitive results. In *Proceedings of the 19th World Congress on Intelligent Transportation Systems (ITS)*.

Coles, A. J., and Coles, A. I. 2014. PDDL+ planning with events and linear processes. In *Proceedings of the Twenty-Fourth International Conference on Automated Planning and Scheduling, ICAPS*, 74–82.

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

Dotoli, M.; Fanti, M. P.; and Meloni, C. 2006. A signal timing plan formulation for urban traffic control. *Control Engineering Practice* 14(11):1297–1311.

Fox, M., and Long, D. 2003a. Pddl2. 1: An extension to pddl for expressing temporal planning domains. *Journal of Artificial Intelligence Research* 20:61–124.

Fox, M., and Long, D. 2003b. PDDL2.1: an extension to PDDL for expressing temporal planning domains. *Journal of Artificial Intelligence Research (JAIR)* 20:61–124.

Fox, M., and Long, D. 2006. Modelling mixed discrete-continuous domains for planning. *Journal of Artificial Intelligence Research* 27:235–297.

Gulić, M.; Olivares, R.; and Borrajo, D. 2016. Using automated planning for traffic signals control. *PROMET-Traffic&Transportation* 28(4):383–391.

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

Krajzewicz, D.; Erdmann, J.; Behrisch, M.; and Bieker, L. 2012. Recent development and applications of SUMO - Simulation of Urban MObility. *International Journal On Advances in Systems and Measurements* 5(3&4):128–138.

Lin, S. 2011. *Efficient model predictive control for large-scale urban traffic networks*. TU Delft, Delft University of Technology.

Miles, J. C., and Walker, A. J. 2006. The potential application of artificial intelligence in transport. *Journal of Intelligent Transport Systems* 153.

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

Vallati, M.; Magazzeni, D.; De Schutter, B.; Chrpa, L.; and McCluskey, T. L. 2016. Efficient macroscopic urban traffic models for reducing congestion: a PDDL+ planning approach. In *The Thirtieth AAAI Conference on Artificial Intelligence (AAAI)*, 3188–3194.

van den Berg, M.; De Schutter, B.; Hegyi, A.; and Hellendoorn, J. 2004. Model predictive control for mixed urban and freeway networks. In *Proceedings of the 83rd Annual Meeting of the Transportation Research Board*, volume 19.

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

Xie, X.-F.; Smith, S.; and Barlow, G. 2012. Schedule-driven coordination for real-time trafc network control. In *Proceedings of the 22nd International Conference on Automated Planning and Scheduling (ICAPS)*, 323–331.