Adaptive and Collaborative Agent-based Traffic Regulation Using Behavior Trees

Arthur Casals, Assia Belbachir, Amal El Fallah-Seghrouchni

Adaptive and Collaborative Agent-based Traffic Regulation Using Behavior Trees

Extended Abstract

Arthur Casals
Sorbonne Université
Paris, France
IPSA
Ivry-sur-Seine, France
arthur.casals@lip6.fr

Assia Belbachir
IPSA
Ivry-sur-Seine, France
Sorbonne Université
Paris, France
assia.belbachir@ipsa.fr

Amal El Fallah-Seghrouchni
Sorbonne Université
Paris, France
amal.elfallah@lip6.fr

ABSTRACT

In this paper, we propose a self-adaptive approach to build a smart traffic light management dealing with intersections. This approach relies on the multiagent systems architecture, suitable to support a distributed and collaborative mechanism of regulation while taking into account dynamic changes in the traffic flow. In our solution, the agents model the intersections and can decide how long is the duration of traffic lights according to their perception of the traffic flow. Each `intersection agent` uses a behavior tree to update the traffic light status (*i.e.* switch from green to red lights and vice-versa), changing the duration of each status dynamically, according to the number of cars perceived in each intersection. We also demonstrate how dynamic traffic control policies can be used in a collaborative scenario to regulate traffic flow.

KEYWORDS

Behavior trees; Traffic regulation; Smart Mobility

ACM Reference Format:

Arthur Casals, Assia Belbachir, and Amal El Fallah-Seghrouchni. 2020. Adaptive and Collaborative Agent-based Traffic Regulation Using Behavior Trees. In *Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020), Auckland, New Zealand, May 9–13, 2020,* IFAAMAS, 3 pages.

1 INTRODUCTION

Traffic congestion is characterized by slower speeds, longer trip times, and increased vehicular queuing on road networks. Since congestion can have cumulative effects on neighboring roads, it is natural to conclude that alleviating traffic in a single intersection can also affect neighboring intersections. From a macro-level perspective, optimizing traffic through the use of dynamic traffic lights control is a problem that requires a network of traffic lights to be taken into account simultaneously.

In this sense, Vilarinho *et al* [11] described various steps to analyze and design a multiagent system for real-time traffic control at isolated intersections. According to their description, each isolated intersection was modeled as an individual multiagent system. Another work conducted by Fleck *et al* [5] proposed a quasi-dynamic adaptive system for a single intersection, modeled as a stochastic hybrid system. In both works, the cumulative effects on neighboring road networks were not studied.

There are also other studies using techniques such as Petri nets [8, 10, 13] and Finite State Machines (FSM) [6, 7, 9]. Regardless of the approach, these techniques aim at providing an *adaptive* mechanism for traffic lights, considering (i) the dynamic conditions of traffic (such as the number of cars flowing through an intersection) and (ii) the effects of the changes resulting from the adaptation process. Adapting the traffic lights duration according to dynamic changes in traffic requires a process of decision making based on the observed traffic conditions. Nevertheless, neither FSMs nor Petri nets formalisms can easily capture the dynamics of the traffic conditions in complex scenarios.

Recently, BTs have been used as an alternative to FSMs due to their increased robustness and flexibility [4]. One of the main advantages of BTs over FSMs is their modularity: a single tree can be composed of different modular behaviors, which also means that it is easy to reuse a modular behavior across different trees. Another advantage of BTs over FSMs is related to concurrency. FSMs do not allow the execution of parallel states; BTs, on the other hand, offer this possibility, allowing the modeling of more complex behaviors.

In this paper, we propose an *adaptive* and *collaborative* architecture to regulate the traffic congestion for *several intersections* using *multiagent* paradigm. Our architecture combines agents and BTs concepts as follows: we model each intersection as an agent endowed with a BT that implements the dynamic reasoning and decision making of the agent. Due to the cumulative effects of traffic congestion on intersections, we introduce `intersection agents` that collaborate among themselves within the multiagent system in order to mitigate the cumulative effects of traffic congestion. We implemented a multiagent simulator to demonstrate how the architecture would fare when compared with another existing multiagent based traffic regulation system. We then proceed to implement a dynamically regulated scenario, where traffic regulation policies can be imposed dynamically in order to enforce specific control conditions in a controlled area. The chosen urban area for this simulation was Place Charles de Gaulle, modeled in SUMO.

2 SCENARIO

Let us consider an urban scenario where each intersection has its traffic lights controlled by intelligent agents. Each agent is capable of perceiving how many cars are queued in each lane of the intersection. This agent is also capable of informing the neighbor

Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020), B. An, N. Yorke-Smith, A. El Fallah Seghrouchni, G. Sukthankar (eds.), May 9–13, 2020, Auckland, New Zealand.

intersection about how many cars are queued in each lane and receiving similar information from its neighbors. With this information, the agent is capable of adjusting the traffic lights (red and green) duration according to the current traffic flow: depending on how many cars are queued, the red light duration can be shortened to increase the flow in a specific lane, for example.

For illustration, let us consider a free-form disposition of intersections, such a roundabout. Each lane can be either classified as *inbound* (the traffic flows towards the center of the roundabout) or *outbound* (the similar opposite). In a typical situation, we could make all the agents collaborate among themselves and optimize the traffic flow in the roundabout as a whole. However, during rush hour, for example, we only want to optimize the route between three of the roads connected to the roundabout. Optimizing traffic in a specific route, in this scenario, could lead to a degradation of the traffic flow in the other roads - which could be acceptable depending on the objective at hand. Ultimately, if specific routes can be optimized, the set of effective neighboring intersections can also be considered as dynamic.

3 CONCEPTUAL FRAMEWORK

In our multiagent system (MAS), we implemented a distributed mechanism that allows multiple `Intersection agents` to collaborate and to adapt the duration of the traffic light according to the observed environment changes (*i.e* traffic flow). The agents compute adjustments (γ) of their traffic lights and communicate them to their `agents intersections` neighbors. The agents are context-aware and can dynamically adapt to the continuous changes in the environment. The *decision-making module* processes the perceived information and decides which action to be performed to adapt the traffic light duration. To make decisions, the agents rely on BTs as the basis for the deliberation mechanism. A *Communication* module allows an `Intersection agent` to communicate with other `Intersection agents` by message passing while *perception* module manages the sensors and their inputs. Each `intersection agent`:

- Has a set of views $\mathcal{V}_i \subset \mathcal{V}$ for an intersection $i \in \mathcal{I}$ (all intersections in the model). The number of cars on each light is acquired as an individual perception. In the case of a perception fault, a previously stored value associated with conditions similar to the current context can be retrieved and used.
- Computes the adequate duration of the traffic light (red and green) using an Adaptive Value Tracking (AVT) [12]. We model this computation as $avt.adjust()$. The computed value is embedded in γ_i, where $\gamma_i \leftarrow avt.adjust()$. The computation of γ_i used the computed γ_j from its intersection neighbor as follows: $\gamma_i \leftarrow \eta_{v_i^1} - \eta_{v_i^2} + \eta_{v_i^3} - \eta_{v_i^4} + \gamma_j$. In the case where no value for γ_j is received, the calculation is based on a locally stored historical value.
- Compares the duration of γ_i for each traffic light in the intersection with the current BT policies P_i. The traffic lights are then updated accordingly. In the case the computation fails, the BT can remain unchanged, or it can use previously stored parameters depending on the scenario in place. Thresholds can also be parameterized for each of the traffic lights.
- Sends the calculated adjustments to each of its neighbor intersections, modeled for a neighbor j as $sendAdjustIntersec(j, \gamma_i)$.

4 IMPLEMENTATION AND DISCUSSION

In order to test the effectiveness of our developed model, we implemented a simulator for the proposed architecture and tested it using different scenarios. We also compared our approach to the one developed in [2]. We divided the experiments into three parts: (i) implementing a simulator using intersection agents capable of using BTs in their decision-making process; (ii) simulating scenarios using both the previous agent model [2] and the one currently being proposed, comparing the results from both simulations to verify the effectiveness of the new model; and (iii) simulating a scenario for Place Charles De Gaulle (a roundabout), located in Paris, with dynamic optimization rules to illustrate a more complex situation (as described in section 2).

Our simulator was implemented using two different tools: SUMO [1] and JADE [3]. We divided our experiments into three different parts. In all cases, the agents used a parametrized BT. The first one, using a single intersection, was aimed at verifying if the proposed model would work as expected. The second part of our experiment compared traffic flow convergence when using the two different models: the original one [2] and the proposed BT-based model. Our original work used different types of agents to process all the information and make the decision related to changing the duration of the traffic lights for a given intersection. The current model uses only one agent to do the same, relying on a more complex decision-making mechanism.

The third part of our experiment was intended to observe how the proposed architecture would behave in a scenario involving dynamic optimization conditions. We recreated Place Charles de Gaulle as a SUMO model and used it in two different situations: in the first one, similarly to the previous part of the experiment, each agent considered the information from its direct neighbors to optimize the traffic flow as a whole. In the second situation, we defined a group of three intersection agents in different parts of the roundabout (not physically consecutive) to optimize a specific route. This configuration was possible due to the insertion of a dynamic policy mechanism, capable of updating the stored policies in each of the agents.

In the first two parts of the experiment, we observed that the new model behaved as expected: there was a slight difference in the convergence pace of the traffic lights duration, but both models converged to the same duration under the same conditions. In the third part of the experiment, we observed that the overall behavior in the first situation (adjacent neighbors) was similar to the previous simulations. The difference between the initial traffic light duration and the duration on convergence was significantly smaller than in the previous experiments. The original optimization mechanism was still effective in policy-based scenarios, while proportionally less when compared with the previous experiments. Another relevant observation is that the obtained values for the policy-based situation are specific to a particular configuration. Considering that we did not stress the possible configurations and policy variations for the initial model, we understand that a more elaborate study regarding this aspect must take place in the future.

REFERENCES

[1] M. Behrisch, L. Bieker J. Erdmann, and D. Krajzewicz. 2011. SUMO - simulation of urban mobility (an overview). *Proceedings of the Third International Conference on Advances in System Simulation (SIMUL 2011)*, 63–68.

[2] Assia Belbachir, Amal El Fallah-Seghrouchni, Arthur Casals, and Marcia Pasin. 2019. Smart Mobility Using Multi-Agent System. *Procedia Computer Science* 151 (2019), 447 – 454. https://doi.org/10.1016/j.procs.2019.04.061 The 10th International Conference on Ambient Systems, Networks and Technologies (ANT 2019) / The 2nd International Conference on Emerging Data and Industry 4.0 (EDI40 2019) / Affiliated Workshops.

[3] F. Bellifemine, F. Bergenti, G. Caire, and A. Poggi. 2005. *Jade — A Java Agent Development Framework*. Springer US, Boston, MA, 125–147. https://doi.org/10.1007/0-387-26350-0_5

[4] Michele Colledanchise and Petter Ögren. 2014. How behavior trees modularize robustness and safety in hybrid systems. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 1482–1488.

[5] Julia L Fleck, Christos G Cassandras, and Yanfeng Geng. 2016. Adaptive quasi-dynamic traffic light control. *IEEE Transactions on Control Systems Technology* 24, 3 (2016), 830–842.

[6] S Venkata Kishore, Vasavi Sreeja, Vibhuti Gupta, V Videesha, IBK Raju, and K Madhava Rao. 2017. FPGA based traffic light controller. In *2017 International Conference on Trends in Electronics and Informatics (ICEI)*. IEEE, 469–475.

[7] Jia Lei and Umit Ozguner. 2001. Decentralized hybrid intersection control. In *Proceedings of the 40th IEEE Conference on Decision and Control (Cat. No. 01CH37228)*, Vol. 2. IEEE, 1237–1242.

[8] Jiliang Luo, Yi-Sheng Huang, and Yi-Shun Weng. 2019. Design of Variable Traffic Light Control Systems for Preventing Two-Way Grid Network Traffic Jams Using Timed Petri Nets. *IEEE Transactions on Intelligent Transportation Systems* (2019).

[9] Sourav Nath, Chandrajit Pal, Suman Sau, Sreyanka Mukherjee, Abhishek Roy, Arghya Guchhait, and Debdatta Kandar. 2012. Design of an FPGA based intelligence traffic light controller with VHDL. In *2012 International Conference on Radar, Communication and Computing (ICRCC)*. IEEE, 92–97.

[10] Kok Mun Ng, Mamun Bin Ibne Reaz, and Mohd Alauddin Mohd Ali. 2013. A review on the applications of Petri nets in modeling, analysis, and control of urban traffic. *IEEE Transactions on Intelligent Transportation Systems* 14, 2 (2013), 858–870.

[11] Cristina Vilarinho, José Pedro Tavares, and Rosaldo J. F. Rossetti. 2015. A Conceptual MAS Model for Real-Time Traffic Control. In *Progress in Artificial Intelligence*, Francisco Pereira, Penousal Machado, Ernesto Costa, and Amílcar Cardoso (Eds.). Springer International Publishing, Cham, 157–168.

[12] K. S. Yildirim and Ö. Gürcan. 2014. Efficient Time Synchronization in a Wireless Sensor Network by Adaptive Value Tracking. *IEEE Transactions on Wireless Communications* 13 (July 2014), 3650 – 3664.

[13] Yaying Zhang, Wei Qiang, and Zhengyu Yang. 2017. A new traffic signal control method based on hybrid colored petri net in isolated intersections. *International Journal of Intelligent Transportation Systems Research* 15, 2 (2017), 98–107.