A macroscopic first-order lane-group-based signalized node model

Raheleh Yahyamozdarani*[1], Chris M.J. Tampère[2]

[1] Ph.D. student, KU Leuven Institute for Mobility LIM, KU Leuven, Belgium

[2] Associate professor, KU Leuven Institute for Mobility LIM, KU Leuven, Belgium

SHORT SUMMARY

In macroscopic dynamic network loading algorithms, node models that find the possible transfer flow at signalized intersections often disregard the role of turning lanes. This may result in finding wrong solutions when different turning lanes have different green times. This paper extends the recently presented node model of (Yahyamozdarani & Tampère, 2022a) to a lane-group-based signalized node model. The presented node model can consider different green times for different turns coming from the same link while it does not impose any extra requirements on the link model. In fact, the link model remains ignorant of the lane compositions. Consequently, the signalized node model would be more accurate regarding the exact green time of each turn while the computational burden is kept limited and the compatibility with the connecting link model is unchanged.

Keywords: dynamic traffic assignment, first-order signalized node model, lane-based macroscopic signalized node model, macroscopic dynamic network loading, macroscopic signalized node model, traffic flow theory

1. INTRODUCTION

In macroscopic Dynamic Traffic Assignment (DTA) different Dynamic Network Loading (DNL) algorithms have been developed for propagating traffic through a network (e.g. Durlin & Henn, (2008), Raadsen et al., (2015), Thonhofer, et al., (2018), Tsanakas, et al., (2021)). In this paper, the focus is on the deterministic DNL algorithms suitable for strategic analysis of city-wide networks.

To this extent, a DNL algorithm with coarse time dynamics (discretizing the network state with a time step Δt equal to, for instance, a few minutes) suffices. Such a DNL captures long-lasting dynamics (that last longer than Δt), while intentionally neglecting high-frequent dynamics. This can be seen as a simplification aiding efficiency and avoiding fake accuracy. Firstly, the computational budget is only allocated to the dynamics that are important for the strategic aim of the simulation, and not more accurate than that. Secondly, the details of demand or supply that cause high-frequent dynamics are usually uncertain in strategic, future scenarios and the apparently finer precision of a simulation may thus be questionable.

DNL algorithms consist of a link model, which captures forward or backward wave propagation in a street, and a node model, which determines the number of vehicles that are able to pass through an intersection respecting existing Boundary Conditions (BC's). There is a variety of link and node models in the literature that are consistent with DTA's performing within the scope of this paper (e.g. iLTM, (Himpe et al., 2016) combined with ORCA (Tampère et al., 2011) or models presented by Durlin & Henn, (2008); and Raadsen et al., (2016)). But when it comes to signalized node models the choice is very limited. To the best of our knowledge, the recently presented continuum signalized node model in Yahyamozdarani & Tampère, (2022a), called COS, is the only consistent one. Yet, the mentioned model is a link-based node model, similar to most node models implemented in the DTA context, which, as we demonstrate further, may not perform adequately in some cases.

Link-based node models simplify the interactions between vehicles coming from different lanes and consider all the lanes together as one entity. Regarding the aim of simulation considered in this paper, such a simplification is acceptable to some extent. For instance, when there is a lane exclusive for a turn movement that has a shorter/longer green time than the rest of the lanes coming from the same incoming link, a link-based signalized node model has to make approximations that might be too restraining/loosening. And thus, result in a wrong queue length, wrong travel time estimation and consequently wrong route choice. The impact of exclusive lane turns on macroscopic signalized node models is not studied sufficiently in the literature due to other major deficiencies that needed to be covered. However, COS node model is comprehensive and only minor deficiencies such as lane compositions remain unsolved, which led us to extend the mentioned model to a lane-based one.

2. METHODOLOGY

In any intersection with more than one outgoing (or receiving) link, the proportion of vehicles that go towards each outgoing link is determined by a route choice algorithm. A node model takes those turning fractions as an input. In the context of DNL, traffic flow is commonly considered uniform, as well as turning fractions. When at an intersection exclusive turn lanes are disregarded, the distribution of turning fractions over lanes is implicitly assumed to be like Fig. 1. a: a homogenous combination of turns would be allowed from all the lanes. In this case, as soon as one turn is blocked because of the First-In-First-Out principle all other turns would also be blocked (FIFO principle guarantees that vehicles leave a link in the same order as they arrived at the end of that link). FIFO is commonly guaranteed by obeying Conservation of Turning Fractions (CTF), meaning the turning fractions should remain the same whatever the total number of vehicles that are able to pass through the intersection. If exclusive turn lanes are disregarded, all the turns of a signalized intersection would be bound by CTF, respecting physical limitations; assuming they all have simultaneous green, the COS node model is capable of solving such a case.
Wherever an exclusive lane turn exists, however, the actual location where CTF should be respected, is not at the intersection but upstream of it where vehicles diverge into their turning lanes, as shown in Fig. 1. b. In this case, downstream of the diverge point turning fractions are not homogenous anymore and thus, blockage of one turn does not result in *immediate* blockage of all other turns. The blocked turn can only block others when its queue reaches the diverge point. In this case, if the intersection is signalized, different lane turns may have different green times and the COS node model would not be able to solve the problem. The presented signalized node model in this paper is intended to solve such cases. However, those cases that COS already covered can be seen as a special case of the newly presented node model with only one lane.

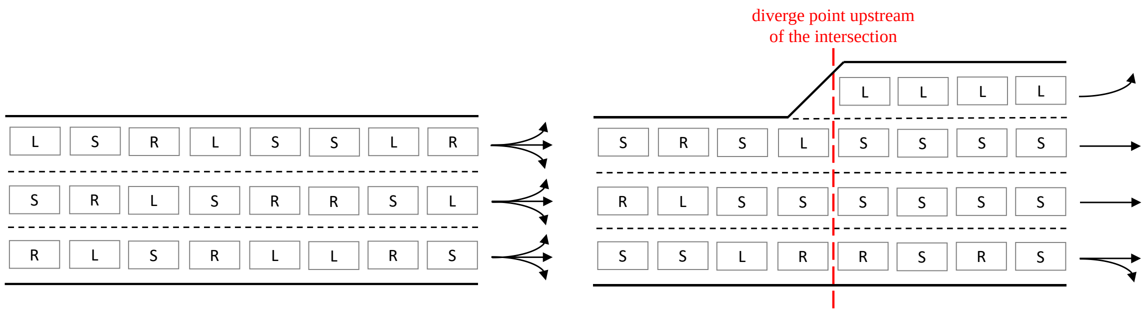

Fig. 1. Turning fraction compositions (a) without and (b) with including exclusive lane turns

A naïve solution could be to simulate the exact length of the pocket lane and to add an extra diverge node, where CTF is observed. No modification to COS would be required but it is not practical because of the calculational burden that it adds to the whole simulation (one extra node plus a few very short links for each approach to a signalized intersection with turn lanes). Instead, we propose a signalized node model that considers incoming (or sending) link as a set of lane groups allowing each lane group to have different green times. Yet within each link, all the lane groups are connected through CTF. For instance, in the case of Fig. 1. b. there are three different lane groups shown in Fig. 2, each of which may have simultaneous or other green times. As implicitly mentioned, the presented node model distinguishes between lane-groups, not lanes. Thus, we call it Lane-group-based-COS (L-COS). Notably, the link model collaborating with the node model is ignorant about the lane compositions and therefore remains uninfluenced.

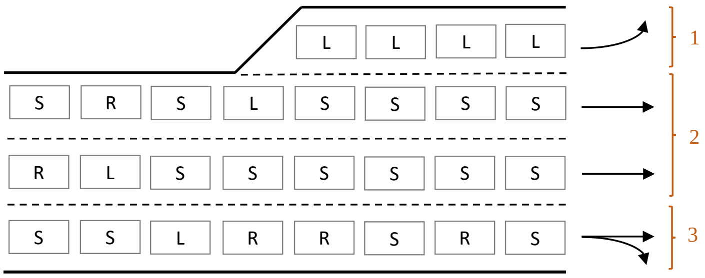

Fig. 2. Number of lane-groups when there is an exclusive lane turn

The solution that we propose relies on the assumption that the DNL is not aimed at capturing high-frequent dynamics but rather slowly evolving peak phenomena, so that in case of saturation of the intersection, the queue quickly builds up beyond the saturated turn to reach the diverge point and blocks the whole link upstream. Therefore, it is justified to neglect the length of the pocket lane and assume that all the lane groups are connected through CTF.

As L-COS shares many properties with COS, we refer for a detailed explanation to Yahyamozdarani & Tampère, (2022a) and focus here only on the main algorithm of L-COS. Notably, their algorithms and some of the sub-algorithms are fundamentally different, for more detail we refer to Yahyamozdarani & Tampère, (2022b).

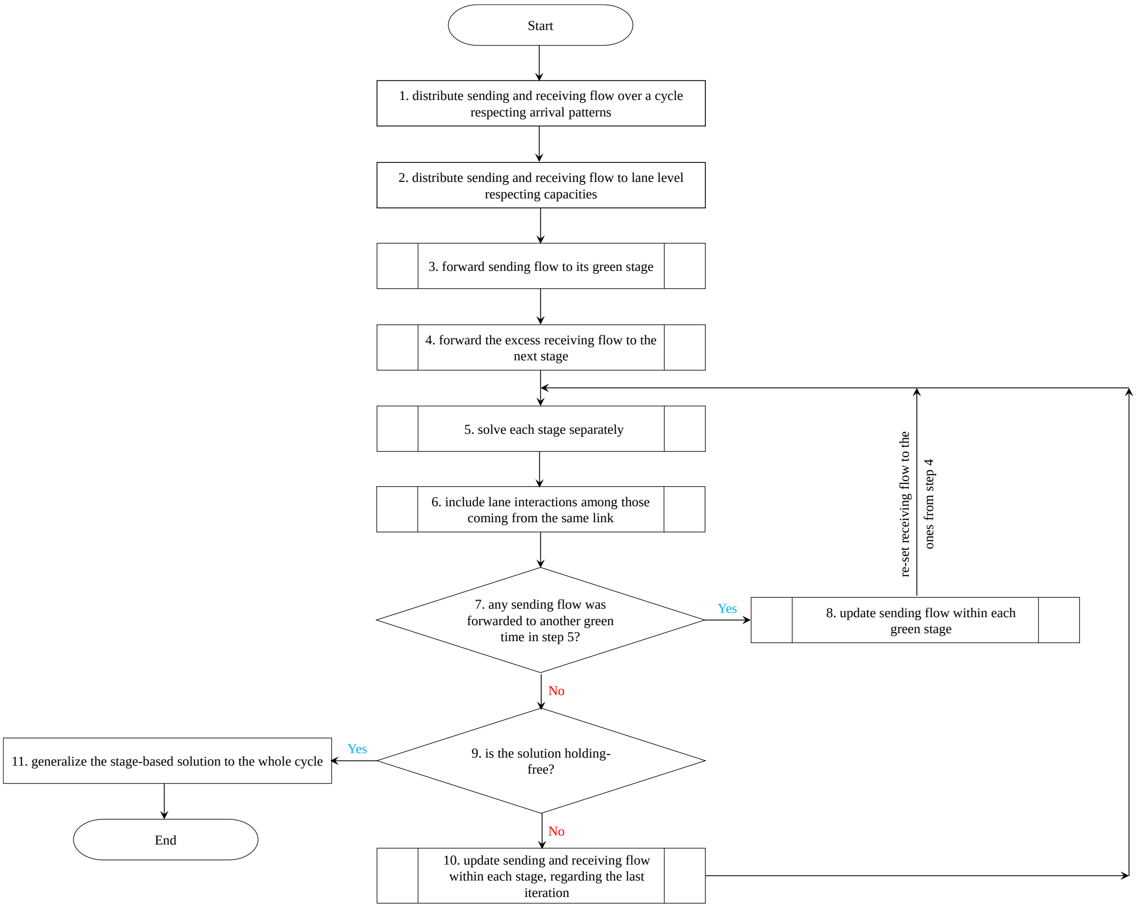

Fig. 3. The algorithm of L-COS

The algorithm of L-COS shown in Fig. 3 finds the stationary transfer flows over a signalized node, that is: after filtering all transient waves. In other words, if the time step is long enough under the same BC's, some transient queues would vanish and only those that last longer than Δt, would remain. In order to consider the conflicts among traffic flows coming from different incoming links and having green simultaneously, each *stage* is first considered independently (a stage being a time interval within which there is no change in the state of a traffic signal, e.g. all-red stage). Then, interactions among stages are included. Possible interactions among stages involve forwarding some sending/receiving flow from one stage to another because they could not be sent/consumed.

In the first step, sending and receiving flows are distributed over *stages*. The distribution follows their arrival patterns, which either need to be provided externally or are considered uniform otherwise. In the second step, stage-based sending and receiving flows are distributed among lane groups proportional to turning fractions, respecting exclusive lane turns. In the third step, sending flows arriving during red are forwarded to their (first next) green stages. Such a forwarding procedure is justified as we are disregarding transient waves (e.g. the transient queue caused by sending flow arriving during a red stage but is later served within its green stage). In the fourth step, the part of receiving flow that cannot be consumed within one stage (e.g. there is insufficient demand for it) is forwarded to another stage where it may be consumed. From the next step, the iterative part of the algorithm starts.

In step five, each stage is considered separately and the transfer flow is proposed regarding the current stage-based sending and receiving flow. Given the proposed transfers in each stage, we verify the necessity of forwarding any remaining sending and receiving flow. In step six, the

interaction between different lane groups coming from the same link is included through CTF. In other words, the most constrained lane group of a link is found and the transfer flows of the other lane groups within the same link are reduced accordingly, if required. When one lane group is bounded more strongly than other(s) within the same link, a queue builds up within that lane group and reaches the diverge point resulting in a blockage of other lane groups as well. Consequently, the transfer flow of other less constrained lanes would be less than when there is no interaction between lanes. This is justified because we only consider long-lasting queues within DNL, as was mentioned before.
In step seven, if there was any sending flow forwarded to another stage, it means not all the vehicles were in their final stage within which they may be transferred. Thus, it is possible that a part of the receiving flow that could have been consumed by them is already forwarded to another stage. Consequently, in step eight, the receiving flow is reset to the initial stage-based one and stages are considered once again by returning to step five.
Once step seven finds no more forwarding of sending flow, all the vehicles were in their final stage where they might be transferred. And thus, the only remaining check is if the solution is holding-free. The holding-free concept was introduced by Jabari (2016) which guarantees that traffic flow is allowed to move until at least one BC constrains it. In step nine, the holding-free criterion is checked. If it does not hold it means more sending flow can be transferred and the solution is not stationary yet. Thus in step ten, sending and receiving flows are updated to what has been sent/consumed during the last iteration plus what still remains. Then the algorithm goes back to step five to solve stages again. When the holding-free condition is met, no more sending flow can be transferred and the solution is final. Then in step eleven, the stage-based transfer flow found in the last iteration is aggregated to the time step duration used by the link model which wraps up the L-COS algorithm.
It is worth mentioning that the sub-algorithms and mathematical formulation of L-COS are not presented here and can be found in Yahyamozdarani & Tampère, (2022b).

3. CONCLUSIONS

Most node models designed for dynamic network loading are link-based, which is satisfactory as long as there is no turn movement with an exclusive lane at a signalized node, which has a different green time than other turns coming from the same link. In this case, a node model should be able to distinguish between lane groups allocated to each turn; otherwise, its solution is not correct. To the authors' best knowledge, the most comprehensive signalized node model presented by Yahyamozdarani & Tampère, (2022a), called COS, is still ignorant about the mentioned problem, while some (rare) other models in the state of the art might consider turn lanes but violate other consistency requirements that COS captures. Therefore, in this paper, the COS node model is extended to L-COS such that it can also include the lane compositions and different green times for each lane group. This requires a modified mathematical formulation and a significantly different solution algorithm. By considering the lane groups endogenous to the node model, no further computational burden is imposed onto the DNL algorithm because the link model can remain ignorant about the turn lane compositions (i.e. no introduction of short turning links and diverge nodes).

ACKNOWLEDGMENT

The authors herewith acknowledge FWO project G051118N "Ontwerp, analyse en optimalisatie van kruispunten van de toekomst".

REFERENCES

Durlin, T., & Henn, V. (2008). Dynamic Network Loading Model with Explicit Traffic Wave Tracking. *Transportation Research Record*, (2085), 1–11. https://doi.org/10.3141/2085-01

Himpe, W., Corthout, R., & Tampère, M. J. C. (2016). An efficient iterative link transmission model. *Transportation Research Part B: Methodological*. https://doi.org/10.1016/j.trb.2015.12.013

Jabari, S. E. (2016). Node modeling for congested urban road networks. *Transportation Research Part B: Methodological*, *91*, 229–249. https://doi.org/10.1016/j.trb.2016.06.001

Raadsen, M. P. H., Bliemer, M. C. J., & Bell, M. G. H. (2015). An efficient and exact event-based algorithm for solving simplified first order dynamic network loading problems in continuous time. *Transportation Research Part B*, *92*, 191–210. https://doi.org/10.1016/j.trb.2015.08.004

Tampère, C. M. J., Corthout, R., Cattrysse, D., & Immers, L. H. (2011). A generic class of first order node models for dynamic macroscopic simulation of traffic flows. *Transportation Research Part B: Methodological*, Vol. 45, pp. 289–309. https://doi.org/10.1016/j.trb.2010.06.004

Thonhofer, E., Palau, T., Kuhn, A., Jakubek, S., & Kozek, M. (2018). Macroscopic traffic model for large scale urban traffic network design. *Simulation Modelling Practice and Theory*, *80*, 32–49. https://doi.org/10.1016/j.simpat.2017.09.007

Tsanakas, N., Ekström, J., Gundlegård, D., Olstam, J., & Rydergren, C. (2021). Data-driven network loading. *Transportmetrica B*, *9*(1), 237–265. https://doi.org/10.1080/21680566.2020.1847213

Yahyamozdarani, R., & Tampère, C. M. J. (2022a). *The COntinuosm Signalized (COS) node model for dynamic traffic assignment* [manuscript submitted for publication], Institute for Mobility LIM, KU Leuven.

Yahyamozdarani, R., & Tampère, C. M. J. (2022b). *A Lane-group-based COntinuous Signalized (L-COS) node model for dynamic traffic assignment* [manuscript submitted for publication], Institute for Mobility LIM, KU Leuven.