Extending the Link Transmission Model with general concave fundamental diagrams and capacity drops

Jeroen P.T. van der Gun[1,], Adam J. Pel[1], Bart van Arem[1]*

[1]Delft University of Technology, The Netherlands
[*]Corresponding author: j.p.t.vandergun@tudelft.nl

Extended abstract submitted for presentation at the
6th International Symposium on Dynamic Traffic Assignment (Sydney, 28-30 June, 2016)

1. Introduction

Kinematic wave theory consists of two main equations: the conservation of vehicles and the equilibrium flow-density relationship. Assuming that each traffic state along a road at each point in time is an equilibrium state, these combine into a single partial differential equation for the propagation of traffic along a network link. Newell (1993) proposed a solution scheme using cumulative numbers of vehicles as the primary variable, which later led to the development of the Link Transmission Model (Yperman, 2007). Daganzo (2005) implicitly shows that for triangular fundamental diagrams, this model indeed leads to the correct solution.

However, the requirement of triangular fundamental diagrams is rather restrictive. Firstly, it imposes the speed in subcritical traffic to be constant instead of more realistically, depending on the traffic density. Secondly, it impedes any discontinuity between the free-flow capacity and the queue discharge rate, i.e. a capacity drop. In this extended abstract, we therefore extend the Link Transmission Model to handle arbitrary concave fundamental diagrams, optionally including capacity drops. The resulting model, which converges to kinematic wave theory if there is no capacity drop, can be used in a network simulation and features both standing queues, with a head fixed at the bottleneck, and moving jams, including stop-and-go waves.

2. Continuous concave fundamental diagram

We start by defining a link model for the case of a continuous concave fundamental diagram $Q(k)$, i.e. without capacity drop. Alternatively, this diagram can be written as two functions $K(q)$ and $K'(q)$, describing the free-flow branch and the congested branch respectively. We define sets of relevant wave speeds $Z = [\inf \operatorname{im} dq/dK, \sup \operatorname{im} dq/dK]$ and $Z' = [\inf \operatorname{im} dq/dK', \sup \operatorname{im} dq/dK']$ respectively. The fundamental diagram is required to satisfy $\{\min Z, \max Z, \min Z', \max Z'\} \cap \{-\infty, 0, \infty\} = \varnothing$. An example fundamental diagram is depicted in Figure 1a. Note that we omit link indices on all variables for brevity.

The theoretical basis for traffic propagation along the link is formed by kinematic wave theory. The sending and receiving flows will be solved in terms of cumulative numbers of vehicles. More precisely, our algorithm relies on finding the maximum possible $N(x, t+\Delta t_x)$ at the considered end of the link $x \in \{x_0, x_L\}$ at the end of the time step under consideration, so that $N(x, t+\Delta t_x) - N(x,t)$ is the maximum number of vehicles exiting or entering the link during the time step, which simply are the sending flow $S(t)$ and the receiving flow $R(t)$ respectively.

We thus rephrased the traffic propagation problem into finding the maximum possible value of $N(x, t+\Delta t_x)$ for $x \in \{x_0, x_L\}$. To do so, we apply the variational theory developed by Daganzo (2005).

The boundary condition for this application is formed by the values of the cumulative curves in previous time steps at both link ends. We build a solution network that indicates how each boundary point may constrain the cumulative number of vehicles at our solution point $P = (x, t + \Delta t_x)$. After each time step, the boundary condition is extended with the newly found solution and the solution network is shifted to compute the next time step.

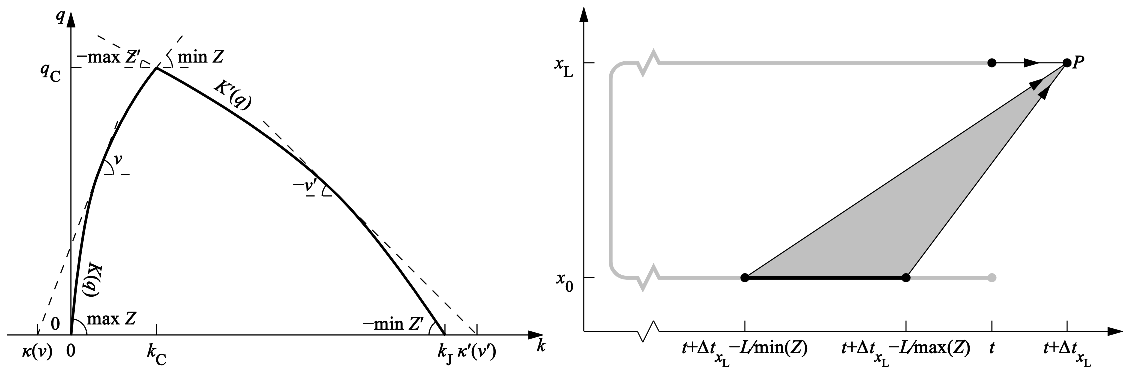

Figure 1: (a) Example fundamental diagram. (b) Example solution network for determining the sending flow, highlighting the relevant parts of the boundary and the corresponding paths to the solution point.

Building upon the proofs in Daganzo (2005), we can derive a finite set of space-time paths that form an exact solution network, as illustrated in Figure 1b. This solution method is more general than those proposed by Yperman (2007) for piecewise-linear diagrams and Gentile (2010) for continuously-differentiable diagrams and better at reproducing acceleration fans or rarefaction waves.

3. Fundamental diagram with capacity drop

We then extend the model so that we can include a capacity drop. A breakdown of traffic, activating a capacity drop, must occur at a node, i.e. a discontinuity in the road infrastructure. The capacity drop will be active on the downstream end of a link if and only if its sending flow is not fully accepted by the downstream node model, i.e. if the node model triggers a queue on the link. By applying a node model without memory effects and using inverted-lambda style fundamental diagrams like Figure 2a, we permit the head of a queue to move upstream. Additionally, we model another capacity drop on the upstream end of an outgoing link if too much traffic is trying to enter it. This ensures that the queue discharge rates before and after a discontinuity in an inhomogeneous road are both taken into account.

If the node model triggers congestion, we first reduce the sending flows to the queue discharge rates on the relevant incoming links, if applicable, due to their capacity drops being enabled. To do so, we need to define a transitional traffic state on the incoming link that serves as a transition from the (possibly varying) inflow state to the queue discharge state. Hence we create a stop-and-go wave with traffic state (k_S, q_S) as depicted in the fundamental diagram in Figure 2a, which is some predefined point on the congested branch of the fundamental diagram satisfying $k_S > k_C$. This requires that the congested branch is linear for $k < k_S$. Next, if these reduced sending flows are still too large for the node to accept, we also reduce the receiving flows to their queue discharge rates on the relevant outgoing links, if applicable. This results in standing queues in front of the node, preceded by the previously mentioned transitional traffic state if necessary. If the congested traffic after the transition has a higher density than k_S, the transitional traffic state is adjusted to match that.

The transitional traffic state effectively originates from a single point (x_L, T), and its physical length increases or decreases depending on whether the inflow is higher or lower than the queue discharge rate. According to shockwave analysis, the finite maximum shockwave speed of the tail of the transitional traffic state is $w_S = (q_C - q_S)/(k_C - k_S)$. Figure 2b illustrates such a transition.

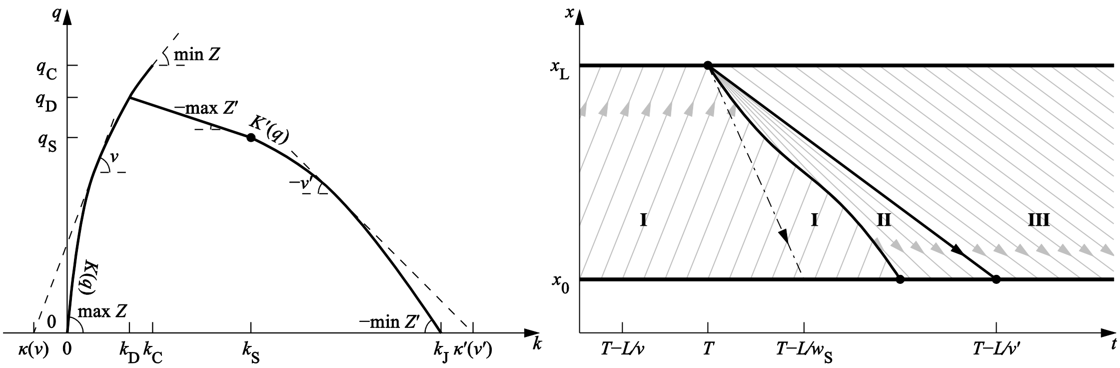

Figure 2: (a) Example fundamental diagram including a capacity drop. (b) Unrelated example transition from free-flow states I with flows higher than the queue discharge rate, to congested states III, via transitional congested state II. State II will not get a lower density than the first state III.

For the link model, this means we effectively add a new path to the solution network for the receiving flow to create the transitional traffic state once downstream congestion occurs. On the other hand we must remove backward paths when the downstream link end is uncongested, so that flows above the discharge rate can be sustained. For both, a set Λ must be maintained for each downstream link end, containing the times it is congested, not including the queue discharge state. Special attention is also needed for the dissolution of congestion. If congestion resolves at the downstream link end at time θ, then the link outflow will be constrained to q_D until some later time Θ, which is assumed ∞ until it is set to a finite value by a queue dissolution procedure as it detects all congestion on the link has dissolved. The latter procedure, which must be invoked for the link at the start of every upstream and/or downstream time step, also reduces Λ to prevent dissolved queues from affecting the upstream link end.

4. Resulting link model algorithms

The algorithms shown on the next page implement our extended link model, supporting arbitrary concave fundamental diagrams, optionally with a capacity drop, subject to the restriction that in case of a capacity drop the congested branch is linear for $k < k_S$. To complete the network loading model, these have to be supplemented with an extended node model, which must decide whether or not the capacity drop occurs on each of its incoming links. In the full study, we also provide the detailed proofs and derivations of our algorithms, present a suitable node model supporting capacity drops and investigate numerical examples.

References

Daganzo, C. (2005). A variational formulation of kinematic waves: basic theory and complex boundary conditions & solution methods. *Transportation Research Part B, 39*, 187-196 & 934-950.

Gentile, G. (2010). The General Link Transmission Model for Dynamic Network Loading and a comparison with the DUE algorithm. *New Developments in Transport Planning: Advances in Dynamic Traffic Assignment*, 153-178.

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

Yperman, I. (2007). *The Link Transmission Model for Dynamic Network Loading.* Katholieke Universiteit Leuven.

Sending flow algorithm:

- $t_1 := t + \Delta t_{x_L} - L / \min Z$.
- $N\left(x_L, t + \Delta t_{x_L}\right) := N\left(x_0, t_1\right) - \kappa\left(\min Z\right) L$.
- $t_2 := \lfloor t_1 \rfloor_{\Delta t_{x_0}}$.
- Loop:
  - $t_2 := \min\left(t_2 + \Delta t_{x_0}, t + \Delta t_{x_L} - L / \max Z\right)$.
  - If $t_1 = t_2$:
    - Exit the loop.
  - $q := \frac{N(x_0,t_2) - N(x_0,t_1)}{t_2 - t_1}$.
  - If $t_1 + \frac{L}{\min V(q)} < t + \Delta t_{x_L} < t_2 + \frac{L}{\max V(q)}$:
    - $N\left(x_L, t + \Delta t_{x_L}\right) :\le N\left(x_0, t_1\right) + q\left(t + \Delta t_{x_L} - t_1\right) - K\left(q\right) L$.
  - $N\left(x_L, t + \Delta t_{x_L}\right) :\le N\left(x_0, t_2\right) - \kappa\left(\frac{L}{t + \Delta t_{x_L} - t_2}\right) L$.
  - $t_1 := t_2$.
- If $t + \Delta t_{x_L} < \Theta$:
  - $N\left(x_L, t + \Delta t_{x_L}\right) :\le N\left(x_L, \theta\right) + q_D\left(t + \Delta t_{x_L} - \theta\right)$.
- $S\left(t\right) := N\left(x_L, t + \Delta t_{x_L}\right) - N\left(x_L, t\right)$.

Receiving flow algorithm:

- $N\left(x_0, t + \Delta t_{x_0}\right) := N\left(x_0, t\right) + q_C \Delta t_{x_0}$.
- $t_1 := t + \Delta t_{x_0} + L / \max Z'$.
- $\lambda := \left(t_1 \in \Lambda\right)$.
- If λ :
  - $N\left(x_0, t + \Delta t_{x_0}\right) :\le N\left(x_L, t_1\right) + \kappa'\left(\max Z'\right) L$.
- Else:
  - $t_2 := \lceil t_1 \rceil_{\Delta t_{x_L}}$.
  - Loop:
    - $t_2 := t_2 - \Delta t_{x_L}$.
    - If $t_2 \le t + L / \max Z'$:
      - Exit the loop.
    - If $t_2 \in \Lambda$:
      - $N\left(x_0, t + \Delta t_{x_0}\right) :\le N\left(x_L, t_2\right) + \kappa'\left(\max Z'\right) L + q_D\left(t_1 - t_2\right)$.
      - Exit the loop.
- $t_2 := \lfloor t_1 \rfloor_{\Delta t_{x_L}}$.
- Loop:
  - $t_2 := \min\left(t_2 + \Delta t_{x_L}, t + \Delta t_{x_0} + L / \min Z'\right)$.
  - If $t_1 = t_2$:
    - Exit the loop.
  - If $\left(t_1, t_2\right) \subseteq \Lambda$:
    - $q' := \frac{N(x_L,t_2) - N(x_L,t_1)}{t_2 - t_1}$.
    - If $t_1 - \frac{L}{\max V'(q')} < t + \Delta t_{x_0} < t_2 - \frac{L}{\min V'(q')}$:
      - $N\left(x_0, t + \Delta t_{x_0}\right) :\le N\left(x_L, t_1\right) + q'\left(t + \Delta t_{x_0} - t_1\right) + K'\left(q'\right) L$.
  - If $t_2 \in \Lambda$:
    - If λ :
      - $N\left(x_0, t + \Delta t_{x_0}\right) :\le N\left(x_L, t_2\right) + \kappa'\left(\frac{L}{t_2 - t - \Delta t_{x_0}}\right) L$.
    - Else:
      - $q' := \min\left(\frac{N\left(x_L, t_2 + \Delta t_{x_L}\right) - N\left(x_L, t_2\right)}{\Delta t_{x_L}}, q_S\right)$.
      - If $\frac{L}{t_2 - t - \Delta t_{x_0}} < \min V'\left(q'\right)$:
        - $N\left(x_0, t + \Delta t_{x_0}\right) :\le N\left(x_L, t_2\right) + q'\left(t + \Delta t_{x_0} - t_2\right) + K'\left(q'\right) L$.
      - Else:
        - $N\left(x_0, t + \Delta t_{x_0}\right) :\le N\left(x_L, t_2\right) + \kappa'\left(\frac{L}{t_2 - t - \Delta t_{x_0}}\right) L$.
      - $\lambda :=$ true .
  - $t_1 := t_2$.
- If $\neg\lambda$:
  - $t_2 := \lfloor t_2 \rfloor_{\Delta t_{x_L}}$.
  - Loop:
    - $t_2 := t_2 + \Delta t_{x_L}$.
    - If $t_2 \ge t + \Delta t_{x_0} + L / w$:
      - Exit the loop.
    - If $t_2 \in \Lambda$:
      - $q' := \min\left(\frac{N\left(x_L, t_2 + \Delta t_{x_L}\right) - N\left(x_L, t_2\right)}{\Delta t_{x_L}}, q_S\right)$
      - $N\left(x_0, t + \Delta t_{x_0}\right) :\le N\left(x_L, t_2\right) + q'\left(t + \Delta t_{x_0} - t_2\right) + K'\left(q'\right) L$.
      - Exit the loop.
- $R\left(t\right) := N\left(x_0, t + \Delta t_{x_0}\right) - N\left(x_0, t\right)$.

Congestion dissolution algorithm:

- If $\Lambda \ne \varnothing$:
  - Loop:
    - $t_L := \min \Lambda + \Delta t_{x_L}$.
    - $t_0 := \max\left(t_0, \left\lfloor t_L - \frac{L}{\min V(q_D)} \right\rfloor_{\Delta t_{x_0}} + \Delta t_{x_0}\right)$.
    - If $t_0 > t$:
      - Exit the loop.
    - If $N\left(x_0, t_0\right) - k_D L \le N\left(x_L, t_L\right) + q_D\left(t_0 - t_L\right)$:
      - $\Lambda := \Lambda \cap \left[t_L, \infty\right)$.
      - If $\Lambda = \varnothing$:
        - $\Theta := t_0 + \frac{L}{\min V(q_D)}$.
        - Exit the loop.
    - Else:
      - $t_0 := t_0 + \Delta t_{x_0}$.

Notes: The values of θ , Θ and t_0 persist over time steps, with initial values $\Theta := 0$ and $t_0 := -\infty$. The compound assignment operation $a :\le b$ is short for $a := \min\left(a, b\right)$. The floor-to-multiple-of operator $\lfloor a \rfloor_b$ means $\lfloor a / b \rfloor \cdot b$ and the ceil-to-multiple-of operator $\lceil a \rceil_b$ means $\lceil a / b \rceil \cdot b$. The following functions are used:

$$\kappa\left(v\right) = \min_{q \in domK}\left(K\left(q\right) - \frac{q}{v}\right) \qquad V\left(q\right) = \left\{v \in Z \,\middle|\, \kappa\left(v\right) = K\left(q\right) - \frac{q}{v}\right\}$$

$$\kappa'\left(v'\right) = \max_{q' \in domK'}\left(K'\left(q'\right) - \frac{q'}{v'}\right) \qquad V'\left(q'\right) = \left\{v' \in Z' \,\middle|\, \kappa'\left(v'\right) = K'\left(q'\right) - \frac{q'}{v'}\right\}$$