WORKING PAPER

ITLS-WP-15-21

A lossless spatial aggregation procedure for a class of capacity constrained traffic assignment models incorporating point queues

By
Mark P.H. Raadsen, Michiel C.J. Bliemer
and Michael G.H. Bell

November 2015

ISSN 1832-570X

INSTITUTE of TRANSPORT and LOGISTICS STUDIES

The Australian Key Centre in
Transport and Logistics Management

The University of Sydney

Established under the Australian Research Council's Key Centre Program.

NUMBER: Working Paper ITLS-WP-15-21

TITLE: **A lossless spatial aggregation procedure for a class of capacity constrained traffic assignment models incorporating point queues**

ABSTRACT: In this paper two novel spatial aggregation procedures are proposed. A network aggregation procedure based on a travel time delay decomposition method and a zonal aggregation procedure based on a path redistribution scheme. The effectiveness of these procedures lies in the fact that they, unlike existing aggregation methods, exploit available information regarding the application context and the characteristics of the adopted traffic assignment procedure. The context considered involves all applications that require path and inter-zonal travel times as output. A typical example of such applications are quick-scan methods, which have become increasing popular in recent years. The proposed procedures are compatible with a class of traffic assignment procedures incorporating (residual) point queues. Furthermore, one can choose to combine network aggregation with zonal aggregation to increase the effectiveness of the procedure. Results are demonstrated via theoretical examples as well as a large-scale case study. In the case study it is shown that network loading times can be reduced to as little as 4% of the original situation without suffering any information loss

KEY WORDS: ***Zonal aggregation, network aggregation, traffic assignment, travel time decomposition, quick scan applications***

AUTHORS: **Raadsen, Belimer and Bell**

CONTACT: INSTITUTE OF TRANSPORT AND LOGISTICS STUDIES (H73)

The Australian Key Centre in Transport and Logistics Management

The University of Sydney NSW 2006 Australia

Telephone: +612 9114 1824

E-mail: business.itlsinfo@sydney.edu.au

Internet: http://sydney.edu.au/business/itls

DATE: November 2015

Introduction

At the end of the 20th century many traffic assignment applications suffered from lack of available computing power. As a consequence, standalone aggregation procedures emerged to overcome this problem. Their objective: Reduce the computational burden by imposing some form of simplification. In recent years however, most of the technical obstructions have been removed and interest in this type of aggregation methodology has declined. However, there are many other reasons why one could consider applying aggregation, for example stability in results, lack of data on a disaggregate level, consistency between granularities or identifying and removing redundant detail which, as a desirable side effect, also greatly reduces computation times. In this paper we focus on this last situation.

To identify redundant detail in traffic assignment, aggregation should also no longer be thought of as a standalone procedure separate from the choices made regarding traffic assignment. Instead it should be a combined effort, which links the application context, traffic assignment procedure and aggregation methodology. Here, the application context considered includes all applications that utilise traffic assignment to obtain inter-zonal travel times. A typical example of such applications are quick-scan methods. Quick-scan methods perform hundreds of simulation runs with near identical demand on a fixed infrastructure. The results are used by policy makers to identify a small subset of scenarios worthy of further investigation. Such applications typically adopt conventional static capacity restrained network loading. This is done not because this is the best choice, but because these models require the least computation time. Unfortunately, they are known for delivering poor travel times. Therefore we argue that for such applications a more sophisticated class of traffic assignment models is required. The additional cost is then more than compensated for by the aggregation procedure proposed.

The aggregation procedure proposed is compatible with a class of traffic assignment procedures that makes the following simplifying assumptions: (i) Undelayed travel time is assumed constant, i.e. independent of flow. (ii) Delay is assumed to be flow dependent, such that (i) and (ii) are consistent with a triangular fundamental diagram (Newel, 1993). (iii) Queues exist, but are assumed to occupy no space, i.e. point queues. (iv) A single time period T is considered. (v) The set of paths P is assumed to be fixed and given.

The generic form of the travel time function $\tau_p\left(f_p\right)$ for a path $p \in P$ compatible with this class of models is shown in (1). Note that from here on we assume that the cost of a path is synonymous with its travel time.

$$\tau_p\left(f_p\right) = \tau_p^{\neg delay} + \tau_p^{delay}\left(f_p\right), \qquad (1)$$

where f_p denotes the desired path flow, $\tau_p^{\neg delay}$ denotes the constant undelayed component (free flow travel times) and $\tau_p^{\text{delay}}\left(f_p\right)$ denotes the flow dependent delay component.

Based on these assumptions we make the following contributions: (i) A novel network aggregation procedure based on travel time decomposition. (ii) A novel zonal aggregation procedure that can be applied on top of the network aggregation procedure, utilising a zonal redistribution scheme. (iii) The aggregation procedures proposed can be lossless. (iv) Practical applicability is demonstrated via theoretical examples and a real world large scale case study. Preliminary results show computational gains exceeding an order of magnitude.

The outline of this paper is as follows. Sections 2 and 3 discuss the literature on relevant traffic assignment procedures and available aggregation methods to date. Section 4 illustrates the relation between point queues and travel time. Sections 5 and 6 contain the proposed network aggregation and zonal aggregation procedures, respectively. Section 7 provides the results of the Gold Coast large scale application case study and we conclude in Section 8 with a summary and further research.

1. Static traffic assignment in the literature

Traffic assignment methods typically find solutions by balancing supply (road infrastructure and routes) and demand (number of trips on paths between origin and destination) via some equilibrium condition. Zones represent the places where demand is loaded onto the network, through a set of origins $r \in R$ and set of destinations $s \in S$ .Total demand D_{rs} is assumed given for each origin-destination pair. Multiple paths $p \in P_{rs}$ can exist per origin-destination pair such that the total set of paths is denoted by $P = \bigcup_{(r,s)} P_{rs}$.

The supply is represented by the transport network, which in turn is defined as a directed labelled graph $G = (N, A, \mathsf{L}_n)$, with nodes $n \in N$ and links $a \in A$, where $R, S \subseteq N$. All links have a length L_a (*km*), maximum speed ϑ_a (*km/h*) and capacity C_a (*veh/h*). Link $a_1(n_1, n_2)$ denotes a directed link a_1 between nodes n_1 and n_2. The set L_n holds the available node labels mapped via partial function $lbl : N \rightarrow \mathsf{L}_N$.

The well known Wardrop equilibrium (Wardrop, 1952) states that equilibrium is reached when no traveller can unilaterally switch routes and decrease their cost. Popular solution algorithms for this deterministic user equilibrium (DUE) condition were developed by Frank and Wolfe (1956) and Beckmann et al. (1956). Alternative definitions emerged that generalised the DUE to a stochastic user equilibrium (SUE) approach (Daganzo and Sheffi, 1977), where perceived cost is used instead of experienced cost. One can formulate the SUE problem as a variational inequality problem (Smith, 1979; Fisk 1980). In the formulation shown in (2) a conditional logit (route choice) model (McFadden, 1973) is adopted, with scaling parameter μ , see also Chen (1999).

$$\sum_{r\in R}\sum_{s\in S}\sum_{p\in P_{rs}} -\left(\tau_p^*\left(f_p^*\right)+\ln\left(f_p^*\right)/\mu\right)\left(f_p - f_p^*\right) \le 0 \quad ,\forall \mathbf{f} \in \Omega, \tag{2}$$

with set Ω containing all feasible path flow vectors $\mathbf{f}$. The path costs and flows of the equilibrium solution are denoted by $\tau_p^*\left(f_p^*\right)$, f_p^*, respectively. In addition, each $\mathbf{f} \in \Omega$ is subject to the following constraints:

$$D_{rs} = \sum_{p\in P_{rs}} f_p, \tag{3}$$

$$f_p \ge 0, \quad \forall p \in P_{rs}, r \in R, s \in S. \tag{4}$$

In traditional capacity restrained static assignment models, strong assumptions are made to solve the DUE or SUE: (i) Queues are modelled inside bottlenecks rather than in front of them, (ii) flows can exceed capacity and (iii) travel time is modelled through link performance functions, such as the BPR function (Bureau of Public Roads, 1964) that require calibration and are only realistic in uncongested situations (Bliemer et al. 2013).

Attempts have been made to address these limitations. For example by introducing side constraints where link flow q_a can no longer exceed link capacity, i.e. $q_a \leq C_a$. A penalty function or Lagrange multiplier then diverts flow from congested links, see Larsson and Patriksson (1995), Bell (1995), Yang and Yagar (1994), Shahpar et al. (2008) or Smith (1987). In most of these models, the penalties imposed are interpreted as queuing delay. However, the queue imposing this delay is not constructed from the demand, it is a mathematical construct and might therefore not be necessarily realistic.

1.1 Residual point queue models

In residual queuing models queues are constructed from the actual demand. A residual queue is defined as the difference between flow entering a link and flow exiting a link in the time period modelled. Lam and Zhang (2000) propose a residual queueing model that computes the travel time delay via the average time spent in the queue (Akçelik and Rouphail, 1993), unfortunately it does not include a node model and queues appear inside the bottleneck instead of in front of it. Bifulco and Crisalli (1998) compute link reduction factors $\alpha_a, a \in A$, based on link exit capacities. This implicitly determines the queue lengths upstream of bottlenecks. Like Lam and Zhang, there is no node model to distribute flows on intersections. Both these models assume queues to be vertical, i.e. point queues. In order to compute link path flows q_{ap}, one multiplies the original path flow by each reduction factor encountered:

$$q_{ap} = f_p \delta_{ap} \prod_{a' \in \eta_{ap}} \alpha_{a'}, \quad p \in P_{rs}, a \in A, r \in R, s \in S, \tag{5}$$

where δ_{ap} represents the link-path incidence indicator, taking on a value of one when path p includes link a and zero otherwise. The set η_{ap} contains all links on path p up to link a. To obtain the desired turning flow s_{ab}, one simply sums all path flows traversing the turn:

$$s_{ab} = \sum_{r \in R} \sum_{s \in S} \sum_{p \in P_{rs}} \delta_{bp} q_{ap}, \quad a \in A_n^{in}, b \in A_n^{out}, n \in N, \tag{6}$$

where sets A_n^{in}, A_n^{out} denote the set of links entering and exiting node *n*, respectively with $A_n = A_n^{in} \cup A_n^{out}$. Attempts have been made to model queues horizontally instead of vertically. The main benefit of this extension is to allow for spillback (van Vliet, 1982; 4Cast; 2009, Bliemer et al., 2012). However, these models are operational in nature, are lacking a node model and their mathematical properties remain unknown. Based on these findings, a residual (point) queuing model is considered most suitable given the application context of quick-scan methods. It supports more sophisticated travel time functions compared to traditional static models, without the problems that result from adopting a horizontal queueing model.

The model adopted here is based on Bliemer et al. (2014). It has the following characteristics: (i) It is derived from a dynamic first order network loading model (Gentile, 2011), (ii) residual point queues reside in front of bottlenecks, (iii) the travel time function is compatible with (1) and delay is based on residual queue length, (iii) it includes a proper first order node model (Tampere et al., 2011), (iv) it adopts the SUE approach outlined in equations (2-4). See Section 4 for more detail.

2. Spatial aggregation in the literature

Two types of spatial aggregation are identified in the literature: *network aggregation* which impacts on the road infrastructure, i.e. nodes and links, and *zonal aggregation* which aggregates the zones (and virtual links connecting the zones to the network).

1.2 Network aggregation

One can distinguish between *network extraction* and *network abstraction* methods within the network aggregation paradigm (Chan, 1968; Connors and Watling, 2008). Network extraction directly removes nodes and links from the network. Examples of this type of procedure can be found in Chang et al. (2002), Long and Stover (1967) and Bovy and Jansen (1983). Network abstraction on the other hand, replaces nodes and links with something else, in order to mimic the original situation. Chen (1968) argues that network extraction is an undesirable approach because it reduces capacity on the network, unrealistically diverts traffic and breaks network connectivity. Based on these findings Chan (1976) proposes a network abstraction method using zonal bypasses adopting a traditional static network loading procedure.

Network decomposition can be regarded as an alternative to network aggregation, instead of aggregating the network it decomposes the network in order to simplify the problem. Examples of this approach, proposing a traffic transfer decomposition technique, can be found in Barton and Hearn (1979) and Hearn (1984), where they reformulate the DUE as a mathematical programming problem. For a comprehensive literature review on network topology decomposition (mostly related to queuing theory) we refer to Osorio and Bierlaire (2009). All aforementioned methods only consider static capacity restrained traffic assignment and do not take the application context into account.

1.3 Zonal aggregation

In an urban planning or demand modelling context zonal aggregation is associated with the procedure on how to construct zones. This is also known as *zoning effects* (Openshaw and Taylor, 1979; Paez and Scott 2004). However, this research considers a traffic assignment context, as such the original zoning structure is assumed given and we only look at the *scaling effect* of zoning. This entails grouping of existing zones. This type of zonal aggregation is termed the "*spatial aggregation problem*" by Daganzo (1980a). Daganzo solved the problem by embedding a zonal aggregation procedure into the Frank and Wolfe (1956) algorithm. A generalised version extending this to a continuum approach (Newell, 1979) also exists (Daganzo, 1980b).

Some studies combined a zonal aggregation and network aggregation procedure to create a more comprehensive approach (Jeon et al., 2010, Bovy and Jansen, 1983). They remove categories of links deemed not important enough. As a second step the zones, residing within the "holes" formed by the network aggregation procedure, are grouped. Chang et al. (2002) adopt a similar approach, only they group zones according to functional groups. The main problem of these methods is twofold: They only consider the traditional static traffic assignment approach and more importantly, no justification is provided on why removing certain links, or grouping certain centroids is reasonable. Finally, all combined network and zonal aggregation approaches rely on network extraction which is an undesirable approach.

As a result of the problems regarding existing aggregation procedures, practitioners revert to manual adjustments based on common sense to alter their transport networks. Consequently, the statement made by Friesz (1985): "*(network) aggregation has been practiced in an essentially ad hoc fashion since the advent of widespread use of network transportation planning models*" is still valid.

In this paper a method is proposed that explicitly takes the application context into account, considers a residual queuing model instead of a capacity restrained model and proposes an automated procedure for both network and zonal aggregation that does not rely on network extraction. By doing so a first step is made in defying Friesz' statement in this day and age.

3. Point queues and travel time

The proposed aggregation procedure is suitable for the entire class of point queue models that satisfy (1). However, in this paper the aggregation is specifically tailored to Bliemer et al. (2014). While not a contribution of this paper, the theory behind this model is reiterated to be able to explain the concepts behind the novel aggregation procedure. First, undelayed travel time is discussed. In Bliemer et al. (2014), undelayed travel time in (1) is simply defined as:

$$\tau_p^{\neg delay} = \sum_{a \in A} \frac{\delta_{ap} L_a}{\vartheta_a}. \tag{7}$$

The delay component on the other hand, is more complex. It depends on the flow via link reduction factors, similar to Bifulco and Crisalli (1998). However, in Bliemer et al. (2014) reduction factors are the result of a node model (Tampere et al., 2011). Recall that this factor signifies the portion of flow that is allowed to exit the link. In Equation (8) the node model is represented by function Γ_n, resulting in reduction factor set $\left[\alpha_a\right]_{a \in A_n^{in}}$.

$$\left[\alpha_a\right]_{a \in A_n^{in}} = \Gamma_n\left(s_{a'b'}, C_{a'}, C_{b'}, a' \in A_n^{in}, b' \in A_n^{out}\right), \quad n \in N. \tag{8}$$

To illustrate the concept of reduction factors consider the corridor network of Figure 1(a) containing a single path $p \in P_{rs}$, with $f_p = D_{rs} = 3000$ (veh/h), with simulation period $T = 1h.$ All nodes in Figure 1(a) have at most a single in-link and single out-link, simplifying the node model function $\Gamma_n(\cdot)$ to $\alpha_a = \min\left\{1, C_b / s_{ab}\right\}$, with $a \in A_n^{in}, b \in A_n^{out}, n \in N$. Clearly, the first four links can accommodate the offered flow resulting in reduction factor of one, i.e. no reduction. The reduction factor on link a_5 is smaller than one due to the node model, which matches the flow to the available capacity on the next link such that $\alpha_5 = \min\{1, 2000/3000\} = 2/3$. The flow offered at the end of link a_6 is obtained via (5) and (6); $s_{5,6} = 2000$. Applying the node model again results in $\alpha_6 = \min\{1, 1000/2000\} = 1/2$, which yield a final flow on links $a_{7\ldots10}$ of 1000 veh/h reaching destination *s*. The other 2000 vehicles that departed from *r* reside on the network as point queues. Queues are assumed to grow linearly in time. Thus, the first vehicle on a link does not encounter a queue, while the last vehicle experiences a queue length of $q_a = s_a(1-\alpha_a)$ while the first encounters no queue at all.

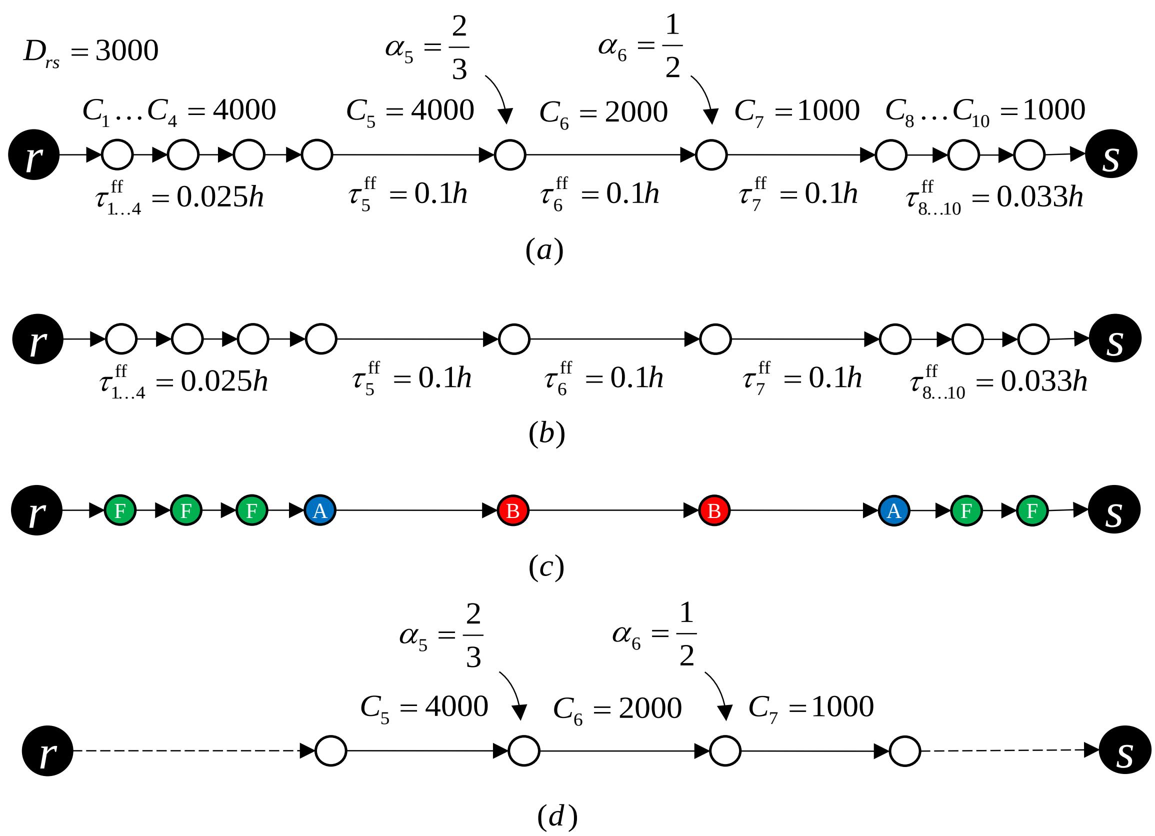


Figure 1(a) Example corridor network including capacities, demand, queues and free flow travel times. (b) Decomposed free flow network. (c) Labelled original network, (d) Decomposed delay network.

To compute the link delay caused by the queue, most point queue models follow the work of Payne and Thompson (1975). They use the length of the queue and dissipation rate (link exit capacity) to compute the link delay, i.e. $q_a / \alpha_a s_a$, where s_a represents the desired link outflow of link *a*. One can then halve the delay to obtain the average link delay. Link delays are summed to obtain the total path delay. While this sounds attractive, it is not realistic. Applying this approach to the example network, the average delay is $1/2\left(1000/2000 + 1000/1000\right) = 3/4\,h$. Intuitively, one can see this is incorrect: 3000 vehicles enter while only 1000 vehicles can exit the network after one hour. This means that 2000 vehicles are queueing. It will take another 2 hours to empty the network, i.e. the last vehicle is delayed by 2 hours, while the first vehicle is not delayed. Hence, the average delay experienced is 1 hour. In Bliemer et al.(2014) it is shown that the actual delay, which is consistent with queuing theory, is elegantly captured via (9).

$$\tau_p^{delay}\left(f_p\right)=\frac{T}{2}\left(\frac{1}{\prod_{a\in A}\delta_{ap}\alpha_a}-1\right),\quad p\in P. \tag{9}$$

We refer to Tampere et al. (2011), or Figure 2 in Bliemer et al. (2014) for details on how to apply the node model $\Gamma_n\left(\cdot\right)$ in more complex situations.

4. Network aggregation through travel time decomposition

The travel time function of (1) can be decomposed into an undelayed (free flowing) and delayed component. In this section we exploit this characteristic to decompose the network based on this premise.

1.4 Path travel time free flow decomposition

The undelayed decomposition comprises summing the undelayed link travel times for each path. Recall from (1) that the undelayed component of the path travel time is invariant to flow. Since a fixed path set *P* is used, one can simply compute the undelayed travel time once for each path via (7), store it, and reuse it for all scenarios. In the corridor example this situation is depicted in Figure 1(b): One assumes no delays exist, resulting in an undelayed path travel time of exactly $\tau_p^{\neg delay}=4\cdot\left(0.025\right)+3\cdot\left(0.1\right)+3\cdot\left(0.033\right)=0.5h$.

1.5 Path travel time delay decomposition

A node n is part of '*blocked*' set $\hat{N}$, when there exists a queue on one of its in-links:

$$\hat{N}=\{n\in N\mid \exists a\in A_n^{in}:\alpha_a<1\}. \tag{10}$$

Suppose that our quick-scan application investigates a number of scenarios. Further suppose that we can construct a scenario that contains all blocked nodes present in all other scenarios. This scenario is termed super-scenario and one can expect this to be the scenario with the highest demand.

In order for the delay decomposition to be lossless, one needs to construct a decomposed network such that the assignment still yields the same delay travel time compared to the original network. To do so, one first needs to compute the equilibrium result of the super-scenario, which is guaranteed to be able to reproduce all possible queues of all considered scenarios. Note that to obtain the correct delays nodes directly adjacent to a blocked node must also be retained in order to maintain each blocked nodes topology. Equation (11) defines this set of '*adjacent'* nodes.

$$\tilde{N}=\{n\in N\mid \exists a\in A_n,a\in A_{n'},n\neq n',n'\in\hat{N}\}. \tag{11}$$

Let us now define the node labelling based on these sets:

$$lbl\left(n \in N \setminus \left(R \bigcup S\right)\right) = \begin{cases} B & ,n \in \hat{N}, \\ A & ,n \in \tilde{N}, \\ F & ,\text{otherwise.} \end{cases} \tag{12}$$

Figure 1(c) shows the labelling on the corridor example network. To obtain the minimal decomposed delay network a method of (undelayed) node smoothing is adopted.

Definition 5.1 Node smoothing.
Given line graph $G(N, A)$ containing links $a_1(n_1, n_2), a_2(n_2, n_3)$, $a_{1\ldots2} \in A, n_{1\ldots3} \in N$. Smoothing of node n_2 is defined transforming G such that $N = N \setminus n_2$ and $A = A \setminus \{a_1, a_2\} \bigcup a_3$ with $a_3(n_1, n_3)$.

A corridor network is a line graph. As such one can obtain the desired decomposed delay network, of Figure 1(d), directly by smoothing the network for all $n \in N$ with $lbl(n) = F$. Observe that the smoothed network still contains the required topology and link path flows, such that the node model yields the same reduction factors as before. However, this approach does not suffice for a general network, because then nodes can have multiple in and out links. On the other hand, a path p can always be described as a line graph $\mathcal{P}_p$. Therefore, a two-step approach is proposed which first applies a line graph based path aggregation procedure and then constructs the desired delay decomposed (general) network from the aggregate paths.

1.5.1 Two-step network aggregation

Consider the grid network shown in Figure 2(a). To illustrate the procedure, a path $\mathcal{P}_p$ is depicted, as well as the node labels. The first step in obtaining the delay decomposed network is applying labelled path smoothing $\mathcal{P}_p / F, \forall p \in P$.

Definition 5.2 Labelled path smoothing.
Given path $p \in P$ with labelled line graph $\mathcal{P}_p(N, A, \mathcal{L}_N)$, if $\exists a_1(n_1, n_2), a_2(n_2, n_3)$, $a_{1\ldots2} \in A, n_{1\ldots3} \in N$, and $lbl(n_2) = \chi$. Labelled path smoothing of p is defined as transforming $\mathcal{P}_p$ into $\mathcal{P}_p'(N', A')$ such that $N' = N \setminus n_2$ and $A' = \{a_3\} + A \setminus \{a_1, a_2\}$, with $a_3(n_1, n_3)$. This is repeated until no more matches are found. This process is denoted via $\mathcal{P}_p' = \mathcal{P}_p / \chi$.

Figure 2(d) shows the result of path smoothing $\mathcal{P}_p' = \mathcal{P}_p / F$. Note that each replacement link a_3 has a cost of zero, because the undelayed travel time has already been captured and no queue exists on this link either.

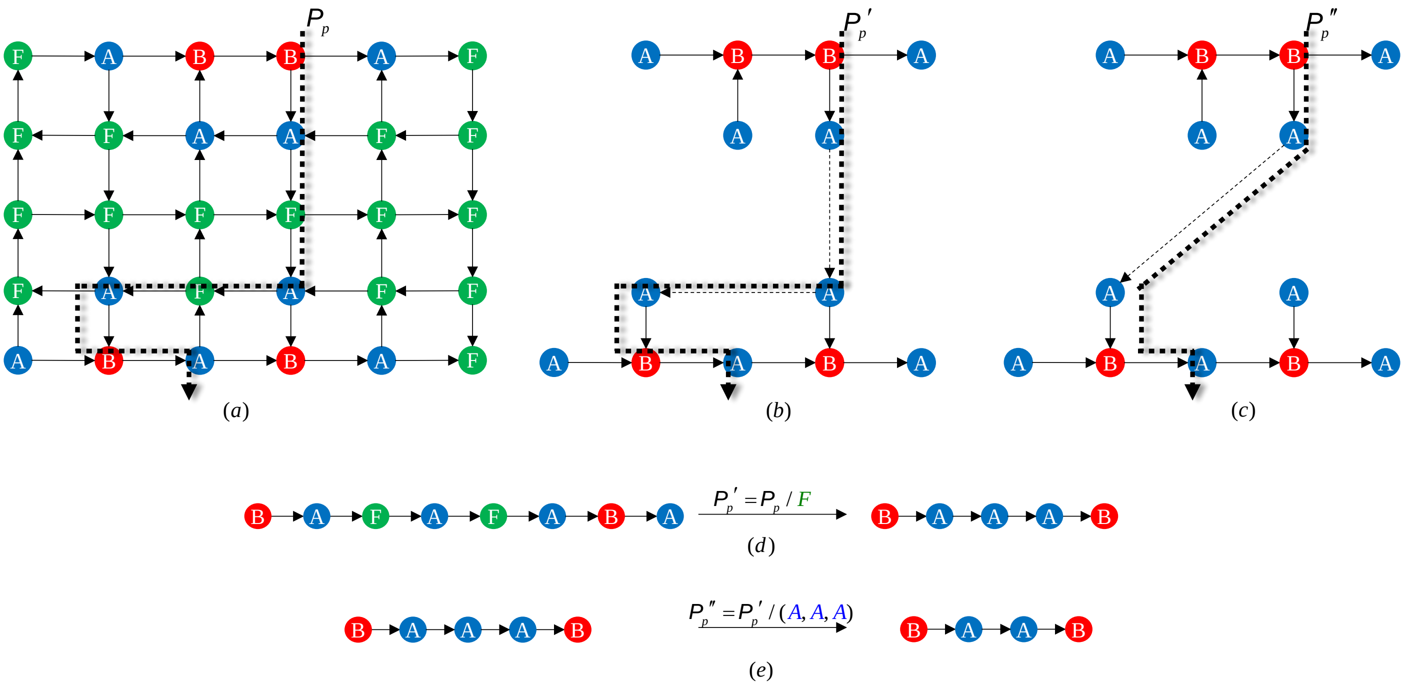


Figure 2(a) Equilibrated grid network including labels. (b) Delay decomposed network based on smoothing. (c) Delay decomposed network after conditional smoothing. (d) Smoothing of path P_p. (e) Conditional smoothing for adjacent labelled nodes.

The second step: Obtaining delay decomposed network G' becomes trivial, once all the paths have been smoothed. One simply takes the union of all smoothed paths (13), see also Figure 2(b).

$$G' = \bigcup_{p \in P} P_p / F. \tag{13}$$

1.6 Path aggregation

The delay decomposed network contains the minimum network required to yield the same solution as the original network. However, it probably still contains superfluous path links, because paths might traverse multiple '*adjacent*' labelled nodes in a row without encountering an actual blocked node, see Figure 2(d). By definition an '*adjacent*' labelled node n is not blocked, i.e. $\alpha_a = 1, a \in A_n^{in}$. Consequently, one can remove '*adjacent*' labelled nodes that reside between other '*adjacent*' labelled nodes on the path level. This is achieved through conditional labelled path smoothing:

Definition 5.3 Conditional labelled path smoothing.
Given labelled line graph $P_p(N, A, L_N)$, if $\exists a_1(n_1, n_2), a_2(n_2, n_3)$, $a_{1..2} \in A, n_{1..3} \in N$, with $lbl(n_1) = \chi_1$, $lbl(n_2) = \chi_2$ $lbl(n_3) = \chi_3$. Conditional labelled path smoothing of path $p \in P$ is defined as transforming P_p into $P_p'(N', A')$ such that $N' = N \setminus n_2$ and $A' = \{a_3\} + A \setminus \{a_1, a_2\}$

with $a_3(n_1, n_3)$ *.this process is repeated until no more matches are found. This is denoted via* $P_p' = P_p / (\chi_1, \chi_2, \chi_3)$.

Applying conditional labelled path smoothing $P_p'' = P_p' / (A, A, A)$, with $P_p' = P_p / F$, is depicted in Figure 2(e). This results in paths with even less links, see also Figure 2(c).

5. Zonal aggregation through redistribution

The effectiveness of the proposed zonal redistribution scheme proposed relies heavily on identifying complete path overlap. Which is defined as follows:

Definition 6.1 Complete path overlap.
Paths p_1, p_2, *with their respective line graph representations* $P_{p_1}(N, A, L_N), P_{p_2}(N', A', L_{N'})$ *are considered to overlap completely, when their respective sets of nodes and links are equal, i.e.* $N = N', A = A'$.

Suppose there are ten paths in the network all with a demand of *d* and one removes nine paths leaving a single path with an updated demand of 10*d*. Then, in case all original paths overlap completely, the delay travel time would remain the same regardless of any bottlenecks present on these paths. Now suppose these ten paths still overlap, except for their origins and destination and subsequent first and last link (connector links). Then, if and only if, all connector links of all paths represent the exact same travel time, one can still remove nine paths and keep one path with the combined demand without affecting the travel time delay computation. Also note that the origin and destination are not important with respect to the travel time computation, so redistributing paths to different zones is a possibility. This is the reasoning that underpins the proposed zonal redistribution scheme.

Zonal redistribution is only applied to the delay decomposed network G' and aggregate paths $P_p'', p \in P$. Note that all connector links are assumed not to directly connect to a blocked node (which represents poor network design). As such they have been smoothed as a result of the delay decomposition procedure and carry no delay, i.e. zero cost. Consequently, in the delay decomposed network, besides all paths that overlap completely, paths that overlap completely except for their connector links can also be grouped. To also accommodate the latter, paths are first redistributed to a new '*redistributed*' zone, such that their connector links also overlap. Then one identifies completely overlapping paths and replaces them with a unique redistributed path. Finally a surjective mapping is maintained between original paths and unique redistributed paths to ensure consistency during route choice.

a. Redistributed zones

Let us first define the sets of redistributed origin and destination zones $\tilde{R}, \tilde{S}$, respectively. Such that there exists a bijection between $\tilde{N}$ and $\tilde{R}$, as well as between $\tilde{N}$ and $\tilde{S}$. This is formalised through bijective functions $\rho : \tilde{N} \to \tilde{R}$ and $\sigma : \tilde{N} \to \tilde{S}$ such that:

$$\tilde{R} = \left\{ \rho(n_1), \rho(n_2) \dots \rho(n_{|\tilde{N}|}) \right\}, \quad n \in \tilde{N}, \tag{14}$$

$$\tilde{S} = \left\{\sigma(n_1), \sigma(n_2) \ldots \sigma(n_{|\tilde{N}|})\right\}, \quad n \in \tilde{N}. \tag{15}$$

In addition links are added between each node $n \in \tilde{N}$ and its origin $r \in \tilde{R}$ and destination $s \in \tilde{S}$, such that (16) and (17) are satisfied, see also Figure 3(a) and (b).

$$\exists a\left(\rho\left(n\right), n\right) \in A, \forall n \in \tilde{N} \tag{16}$$

$$\exists a\left(n, \sigma\left(n\right)\right) \in A, \forall n \in \tilde{N} \tag{17}$$

b. Redistributing paths

The assignment used to obtain path travel time delays based on zonal redistribution, adopts a path set $\tilde{P}$ instead of original path set *P*. Path set $\tilde{P}$ contains only unique redistributed paths. A redistributed path is defined as follows:

Definition 6.2 Redistributed path.
Given original path $p \in P_{rs}, r \in R, s \in S$ with $\mathsf{P}_p(N, A, \mathsf{L}_N)$ and connector links $a_1(r, n_1), a_1(n_2, s) \in A, n_{1\ldots2} \in \tilde{N}$. Then redistributed path p' of original path p is defined as $p' \in P_{r's'}, r' \in \tilde{R}, s' \in \tilde{S},$ with $\mathsf{P}_{p'}(N', A', \mathsf{L}_{N'})$ where $N' = \{r', s'\} + N \setminus \{r, s\}$ and $A' = \{a_1'(r', n_1), a_2'(n_2, s')\} + A \setminus \{a_1, a_2\}$ given $r' = \rho\left(n_1\right), s' = \rho\left(n_2\right)$.

Consider the four distinct paths depicted in Figure 3(c). After redistributing these paths, paths p_2' and p_4' overlap completely, as do paths p_1' and p_3', see also Figure 3(d). One then maps these overlapping paths to the two unique paths $p_{5\ldots6} \in \tilde{P}$ of Figure 3(e).

Definition 6.3 Unique redistributed path set.
The set of unique redistributed paths is defined as the minimal set of paths $\tilde{P}$ such that: For each path $p \in P$, with redistributed path p', $\exists p'' \in \tilde{P}$, where p'' completely overlaps with p'. This is formalised through surjective function $\psi : P \rightarrow \tilde{P}$.

The flow of each unique redistributed path is then simply obtained via:

$$f_p = \sum_{\psi(p')=p} f_{p'}, \quad p \in \tilde{P}, p' \in P \tag{18}$$

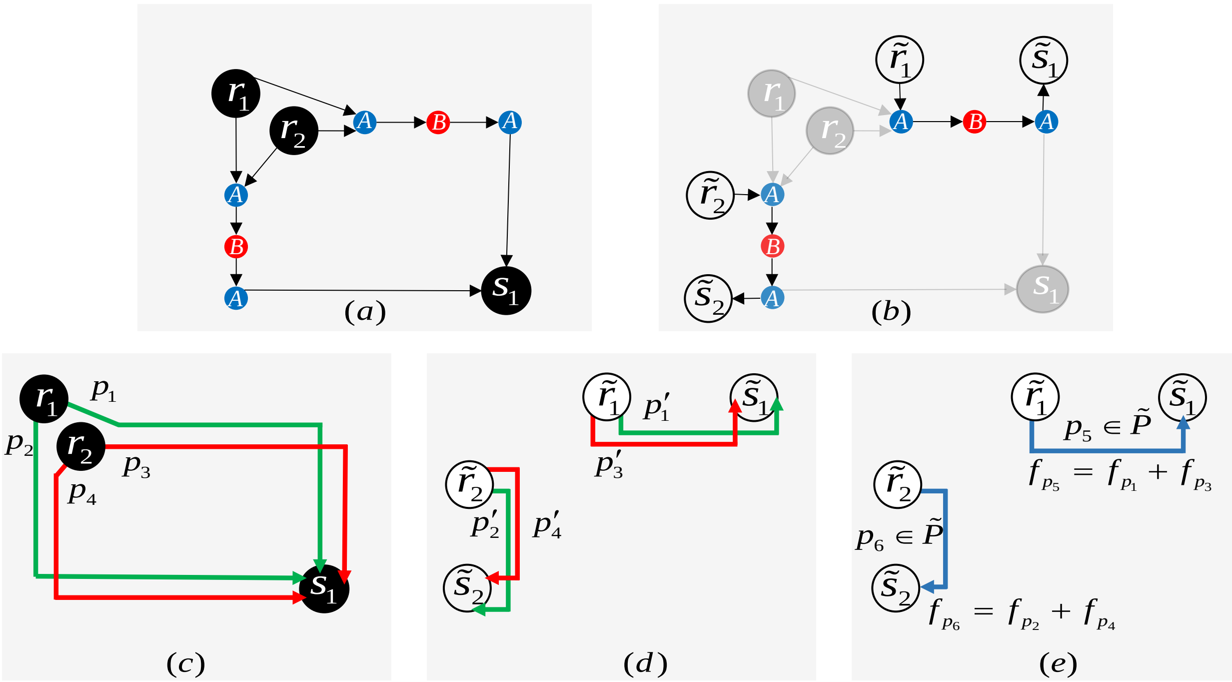


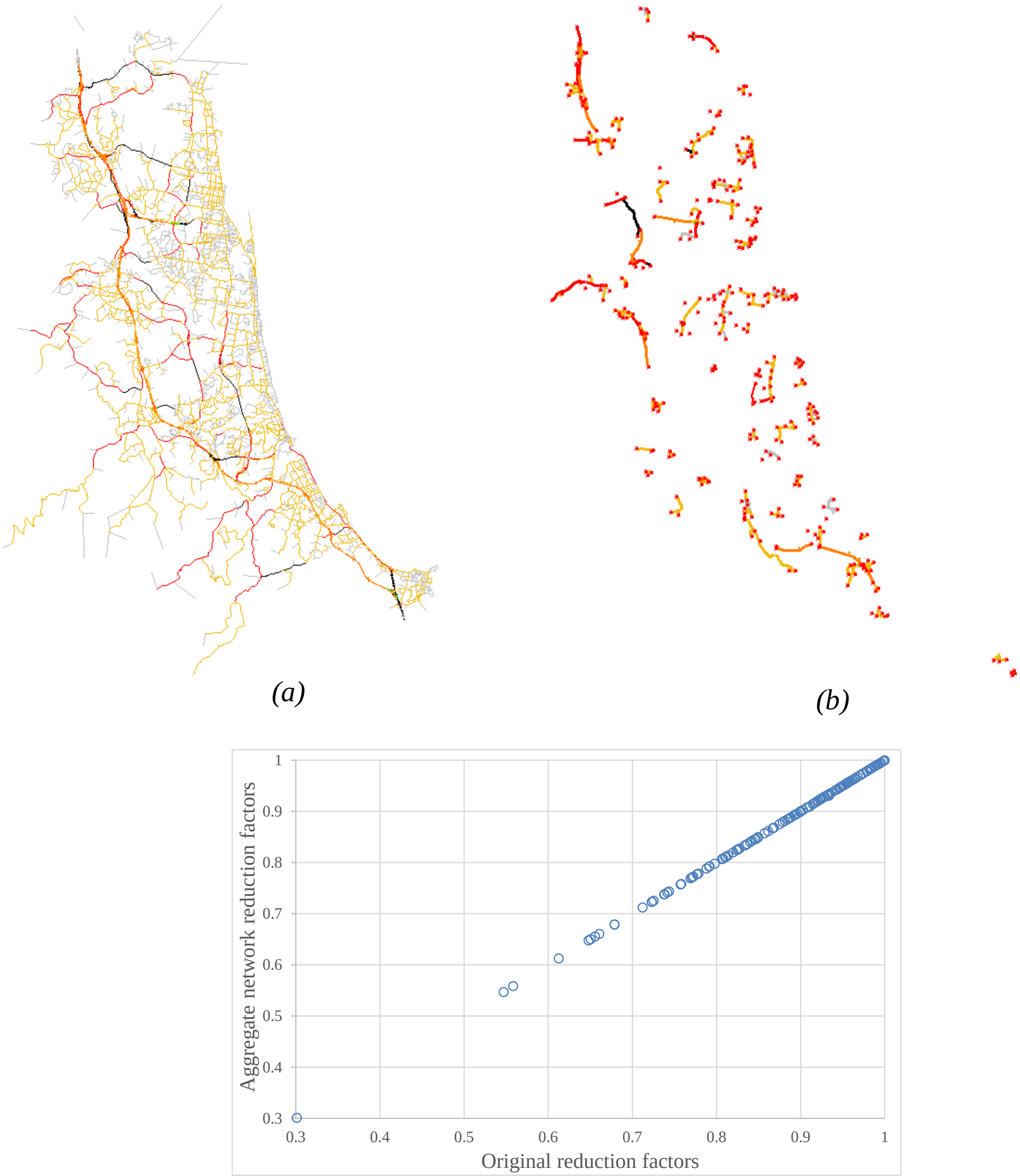
Figure 3 ***(a) delay decomposed network and original zones, (b) same as (a), only now with redistributed zones in place. (c) original paths, (d) non-unique redistributed paths, (e) unique redistributed paths.***

The assignment procedure based on the redistributed path set (and flows) produces the same link flows compared to the situation without applying zonal redistribution, only now using the minimal set of paths. Computation time is reduced significantly, since the computational burden predominantly depends on the number of paths. However, redistributing paths does complicate finding equilibrium. Redistributed zones host a multitude of paths originating from different original zones. Therefore, route choice cannot be applied based on the set of paths per redistributed origin-destination pair. Instead, route choice must still be applied to the original origin-destination pairs and paths. Each original paths' delay travel time is now obtained via mapping $\psi : P \rightarrow \tilde{P}$, as shown in (19).

$$\tau_p\left(f_p\right)=\tau_p^{\neg delay}+\tau_{p'}^{delay}\left(f_{p'}\right), \quad \forall p \in P, p'=\psi\left(p\right). \tag{19}$$

6. Real world case study

In this section, the combined network and zonal aggregation procedure is applied to the large-scale Gold Coast network (Queensland, Australia), kindly provided by Veitch-Lister Consultancy. This network contains: 2987 nodes, 5076 links, 1067 origins and 1067 destinations. A hypothetical (one hour) morning peak origin-destination matrix is utilised, containing well over 120,000 veh/h. This demand is assumed to represent the super-scenario. Path set P is generated a-priori, based on Fiorenzo-Catalano et al. (2004): Resulting in a conditional SUE. Finally, the scale parameter of the route choice is set to $\mu = 5$ and the Method of Successive Averages (MSA) is used to smooth iteration results, using a smoothing factor $\lambda = 0.7$ (in line with Polyak (1990)). The proposed aggregation procedures are implemented in the StreamLine framework, provided by DAT.mobility.



(a) *(b)* *(c)*

Figure 4 ***(a) Original Gold Coast network, (b) Gold Coast network after delay decom-position and zonal redistribution, (c) comparing reduction factors between original and aggregate equilibrium result (80 iterations).***

The original network is depicted in Figure 4(a). After equilibrating the super-scenario, the network and zonal aggregation procedures are applied to obtain the delay decomposed, zonal redistributed network. The result is shown in Figure 4(b). To illustrate that the equilibrated aggregate network yields the same result as the original network, the resulting reduction factors of the two assignments are compared in Figure 4(c) with original reduction factors α_a on the x-axis and aggregate reduction factors α'_a on the y-axis.

Clearly, the reduction factors in Figure 4(c) have converged. Further proof of this is given when examining norm $\|\tilde{\boldsymbol{\alpha}}\| = 0.00366$, which is defined as:

$$\|\tilde{\boldsymbol{\alpha}}\| = \left(\sum_{a \in A} \left(\alpha_a - \alpha'_a \right)^2 \right)^{1/2}. \tag{20}$$

The effect of applying the proposed aggregation procedure(s) on network, zone and path topology are extensive. As a result computation times are reduced accordingly, mainly due to the decrease in both size and number of paths that need to be loaded in each iteration. Table 1 outlines the results obtained in the Gold Coast case study (for an 80 iteration assignment simulation).

Table 1 Effects of aggregation on computation times and network, zone and path topology

Network	*Nodes*	*Blocked nodes*	*Paths*	*Total path links*	*Total network loading time (s)*
Original	2987	148	1,221,446	55,852,786	2178.48
Delay decomposition	473	148	1,221,446	12,919,862	902.57
+Zonal redistribution	473	148	99,528	1,142,338	88.62

Applying only delay decomposition halves the network loading time on this network. This can be attributed to the reduced number of links per path (10.6 vs 45.7 on average). However, it is not until the combined procedure (zonal redistribution and delay decomposition) is employed that the potential of this approach becomes apparent. The number of paths drops to less than 10% of the original (<100,000 vs 1,221,446), due to using unique redistributed path set $\tilde{P}$ instead of original path set P. As a result, the time required to complete network loading is a mere 4.1% of the original computation time, meaning one can now compute twenty-five scenarios instead of one, in the same amount of time.

7. Conclusions and future research

a. Conclusion

In this paper two novel spatial aggregation procedure are proposed that are suitable for a class of traffic assignment models compatible with (1). In contrast to existing aggregation procedures, the novel methodology is tailored towards a specific application type, i.e. travel time oriented applications. Using this information and utilising characteristics of the adopted class of compatible traffic assignment model the aggregation procedures can be optimised and are shown to be lossless, provided a super-scenario can be identified.

Practical implications of this procedure are twofold: (i) Practitioners can either consider more scenarios in the same amount of time, which improves the reliability and robustness of recommendations made or, (ii) practitioners have more time investigating the subset of scenarios that require further investigation, improving the accuracy of the studies. As an added benefit, there is no more reason to adopt traditional capacity restraint traffic assignment models in quick-scan methods, from a purely computational point of view.

Results on the Gold Coast case study show that aggregate network loading only takes 4.1% of the original network loading time. This is achieved by a network delay decomposition method that reduces path size and a zonal redistribution scheme that identifies and removes overlapping aggregate paths such that only a minimal set of paths remains.

b. Further Research

The proposed procedure is guaranteed to be lossless when a super-scenario can be identified. While one might expect this is the scenario with the highest demand, in practice this is not always the case. Models, like the Bliemer at al. (2014) model, that do not have monotonic increasing cost functions do not necessarily generate more blocked nodes with higher demand. Identifying the super-scenario is therefore not trivial. Further research is needed to: (i) See if a procedure exists that quickly identifies the super-scenario for models with non-monotonic cost functions, (ii) in case no super-scenario is identified, estimate the amount of information loss suffered when still applying the proposed aggregation procedure.

It would also be of interest to investigate the effects of this aggregation procedure on other compatible traffic assignment models and compare then. This would provide further insight in the robustness and generalisability of the methodology.

Acknowledgements
This research is partly funded by the Australian Research Council Linkage Project LP130101048.

References

4Cast, (2009). Qblok_2004: toedelingsprocedure LMS/NRM.
Akçelik, R. and Rouphail, N.M. (1993). Estimation of delays at traffic signals for variable demand conditions. Transportation Research Part B: Methodological, 27(2), pp.109–131.
Barton, R.R. and Hearn, D., 1979. Network aggregation in transportation planning models - final report.
Beckmann, M., McGuire, C.B., Winsten, B.W. (1956). Studies in the economics of transportation, New Haven CT, USA: Yale University Press.
Bell, M.G.H., 1995. Stochastic user equilibrium assignment in networks with queues. Transportation Research Part B: Methodological, 29, pp.125–137.
Bifulco, G. and Crisalli, U. (1998). Stochastic user equilibrium and link capacity constraints: formulation and theoretical eveidencies. In Proceedings of the European Transport Conference. pp. 85–96.
Bliemer, M.C.J., Brederode, L. Wismans, L. Smits, E-S. (2012). Quasi-dynamic network loading: adding queuing and spillback to static traffic assignment.

Bliemer, M.C.J., Raadsen, M.P.H., Romph de, E. Smits, E-S. (2013). Requirements for Traffic Assignment Models for Transport Planning : A Critical Assessment, Australian Transport Research Forum (ATRF).
Bliemer, M.C.J. Raadsen, M.P.H., Smits, E-S. Zhou, B. Bell, M.G.H. (2014). Quasi-dynamic traffic assignment with residual point queues incorporating a first order node model. Transportation Research Part B: Methodological, 68, pp.363–384.
Bovy, P. and Jansen, G. (1983). Network aggregation effects upon equilibrium assignment outcomes: An empirical investigation. Transportation Science, 17(3), pp.240–261.
Bureau of Public Roads, United States (1964). Traffic assignment manual, Washington.
Chan, Y., Follensbee, K.G., Manheim, M.L., Mumford, J.R. (1968). Aggregation in transport networks: An Application of Hierarchical Structure., M.I.T. dept. Civil Engng Res. Report No R68-47.
Chan, Y., 1976. A method to simplify network representation in transportation planning. Transportation Research, 10(3), pp.179–191.
Chang, K.T., Khatib, Z., Ou, Y. (2002). Effects of zoning structure and network detail on traffic demand modeling. Environment and Planning B: Planning and Design, 29, pp.37–52.
Chen, H-K. (1999). Dynamic travel choice models, a variational inequality approach, Berlin: Springer-Verlag.
Connors, R.D. and Watling, D.P. (2008). Aggregation of Transport Networks Using Sensitivity Analysis. In Proceedings of the European Transport Conference. Association for European Transport.
Daganzo, C.F. and Sheffi, Y. (1977). On Stochastic Models of Traffic Assignment. Transportation Science, 11(3), pp.253–274.
Daganzo, C. F. (1980). An equilibrium algorithm for the spatial aggregation problem of traffic assignment. Transportation Research Part B: Methodological, 14(3), pp.221–228.
Daganzo C.F. (1980b). Network representation, continuum approximations and a solution to the spatial aggregation problem of traffic assignment. Transportation Research Part B: Methodological, 14(3), pp.229–239.
Fisk, C. (1980). Some developments in equilibrium traffic assignment. Transportation Research Part B: Methodological, 14(3), pp.243–255.
Fiorenzo-Catalano, S., van Nes, R., Bovy, P.H.L. (2004). Choice Set Generation for Multi-modal Travel Analysis. European Journal of Transport and Infrastructure Research, 4(2), pp.195–209.
Frank, M. and Wolfe, P. (1956). An algorithm for quadratic programming. Naval Research Logistics quarterly, 3, pp.95–109.
Friesz, T.L. (1985). Transportation network equilibrium, design and aggregation: Key developments and research opportunities. Transportation Research Part A: General, 19, pp.413–427.
Gentile, G. (2011). The General Link Transmission Model for Dynamic Network Loading and a comparison with the DUE algorithm. In New Developments in Transport Planning: Advances in Dynamic Traffic Assignment (Chapter 8).
Hearn, D. W. (1984) Practical and theoretical aspects of aggregation problems in transportation planning models. Transportation Planning Models. New York, Elsevier.
Jeon, J-H., S-Y. Kho, J.J. Park, D-K. Kim (2012). Effects of spatial aggregation level on an urban transportation planning model. KSCE Journal of Civil Engineering, 16, pp.835–844.
Lam, W. and Zhang, Y. (2000). Capacity-constrained traffic assignment in networks with residual queues. Journal of transportation engineering, (April), pp.121–128.
Larsson, T. and Patriksson, M. (1995). An augmented lagrangean dual algorithm for link capacity side constrained traffic assignment problems. Transportation Research Part B: Methodological, 29, pp.433–455.

Long, G.D. and Stover, V.G. (1967). The effect of network detail on traffic assignment results. Research Report Number 60-11, Texas Transportation Institute, College Station, Texas.

McFadden, D. (1973). Conditional logit analysis of qualitative choice behavior. In Frontiers in Econometrics. pp. 105–142.

Newell G.F. (1979). Traffic flow on transportation networks, Cambridge Massachusetts: MIT Press.

Newell G.F. (1993). A Simplified Theory of Kinematic Waves In highway Traffic, Part II: Queuieng At Freeway Bottlenecks. Transportation Research Part B: methodological. Volume 27B issue 4, 1993, pp.289-303.

Openshaw, S. and Taylor, P.J., 1979. A million or so correlation coefficients: three experiments on the modifiable areal unit problem. statistical applications in spatial sciences, pp.44–127.

Osorio, C. and Bierlaire, M., 2009. An analytic finite capacity queueing network model capturing the propagation of congestion and blocking. European Journal of Operational Research, 196(3), pp.996–1007.

Páez, A. and Scott, M., 2004. Spatial statistics for urban analysis: A review of techniques with examples. GeoJournal, 61, pp.53–67.

Payne, H.J., and Thompson, W.A. (1975). Traffic assignment on transportation network with capacity constraints and queuing. 47th National ORSA/TIMS North Am. Meeting.

Polyak, B.K. (1990). New method of stochastic approximation type. Automated Remote Control, 51(7), pp.937–946.

Shahpar, A.H., Aashtiani, H.Z., Babazadeh, A., 2008. Dynamic penalty function method for the side constrained traffic assignment problem. Applied Mathematics and Computation, 206(1), pp.332–345.

Smith, M.J. (1979). The existence, uniqueness and stability of traffic equilibria. Transportation Research Part B: Methodological, 13(4), pp.295–304.

Smith, M.J. (1987). Traffic control and traffic assignment in a signal-controlled network with queueing. In N. H. Gartner & Wilson M., eds. Proceedings of the Tenth International Symposium on Transportation and Traffic Theory. Elsevier, pp. 61–68.

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, 45(1), pp.289–309.

Yang, H. and Yagar, S. (1994). Traffic assignment and traffic control in general freeway-arterial corridor systems. Transportation Research Part B: Methodological, 28, pp.463–486.

Van Vliet, D. (1982). SATURN: a modern assignment model. Traffic engineering and control, 23, pp.575–581.

Wardrop, J.G. (1952). Some theoretical aspects of road traffic research. ICE Proceedings: Engineering Divisions, 1, pp.325–362.