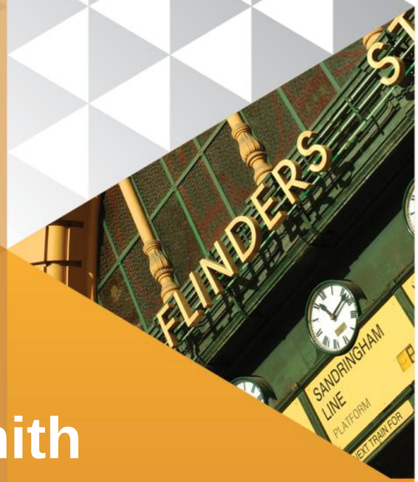
Veitch Lister Consulting Pty Ltd




Zenith Model Framework
Papers – Version 3.0.0

# Paper I – Zenith Transit Assignment Algorithm

June 2014

Page Intentionally Left Blank

Zenith Model Framework Papers – Version 3.0.0

Paper I – Zenith Transit Assignment Algorithm

Final Report



| Date | Revision | Prepared By | Checked By | Approved By | Description |
|---|---|---|---|---|---|
| 04/06/2014 | A | JC/TV | AA | JC | Prepared as part of Melbourne recalibration |
| | | | | | |

# Executive Summary

The Zenith Models are a family of four step transport models, developed by Veitch Lister Consulting (VLC) and implemented in the OmniTRANS software package for a range of Australian cities and regions. This document is one in a series of working papers that collectively describe the model structure and framework of the Zenith Model; in particular, this document describes the Zenith Transit Assignment (ZTA) Model.

The aim of the ZTA Model is to predict the routes taken by all travellers using the public transport system. The model takes into account factors such as service frequency, route travel time, complementary and competing routes, stop locations, route stopping patterns, fares, interchanges, in-vehicle overcrowding, and parking supply constraints at park and ride facilities.

# Contents

# List of Figures



# List of Tables

Page Intentionally Left Blank

# 1 Introduction

## 1.1 Background

This Technical Note is one of a series of papers that collectively describe the Zenith Travel Model. Zenith is a classical four step model, implemented in the OmniTRANS software package for a range of Australian cities and regions.

This Technical Note details the methodologies which underpin the Zenith Transit Assignment (ZTA) component of the Zenith model. The ZTA is a novel mode chaining assignment algorithm that was pioneered by Veitch Lister Consulting and which forms part of the commercial transport modelling package 'OmniTRANS'.

The remainder of this document is structured as follows:

- Section 2 describes the objectives of ZTA
- Section 3 describes the model inputs
- Section 4 describes the modelling process
- Section 5 describes the model outputs
- Section 6 describes the limitations of the model

# 2 Objectives and Context

The aim of the ZTA model is to realistically estimate the routes chosen and travel costs incurred by all users of the public transport network as they travel within the modelled area. The model is sensitive to the effects of stop placement, route frequency, travel time, carriage crowding and fares.

Within the overall Zenith model process (illustrated in Figure 1 below), assignment is the final step. However, the particular role played by assignment is more clearly illustrated in Figure 2, which groups the model's numerous components into two: the demand model and the assignment model.

The **demand model**, which encompasses all of the model's components *except assignment*, is concerned with estimating the level of demand (number of trips) between each origin / destination (OD) pair, by the various trip purposes, travel modes and periods of the day.

The **assignment model** estimates the route chosen for each trip output from the demand model, and outputs travel costs (travel time, toll, fuel, fare, etc.), which are subsequently fed back to the demand model (leading to an iterative process between the demand and assignment models).

Within the assignment model, the ZTA is responsible for the assignment of travellers to the public transport network (rail, bus, ferry, tram, etc.). The assignment of cars and commercial vehicle passengers is handled by the Static Traffic Assignment, which takes place before the ZTA (to allow traffic related congestion to determine the travel speeds of on-road busses and trams).

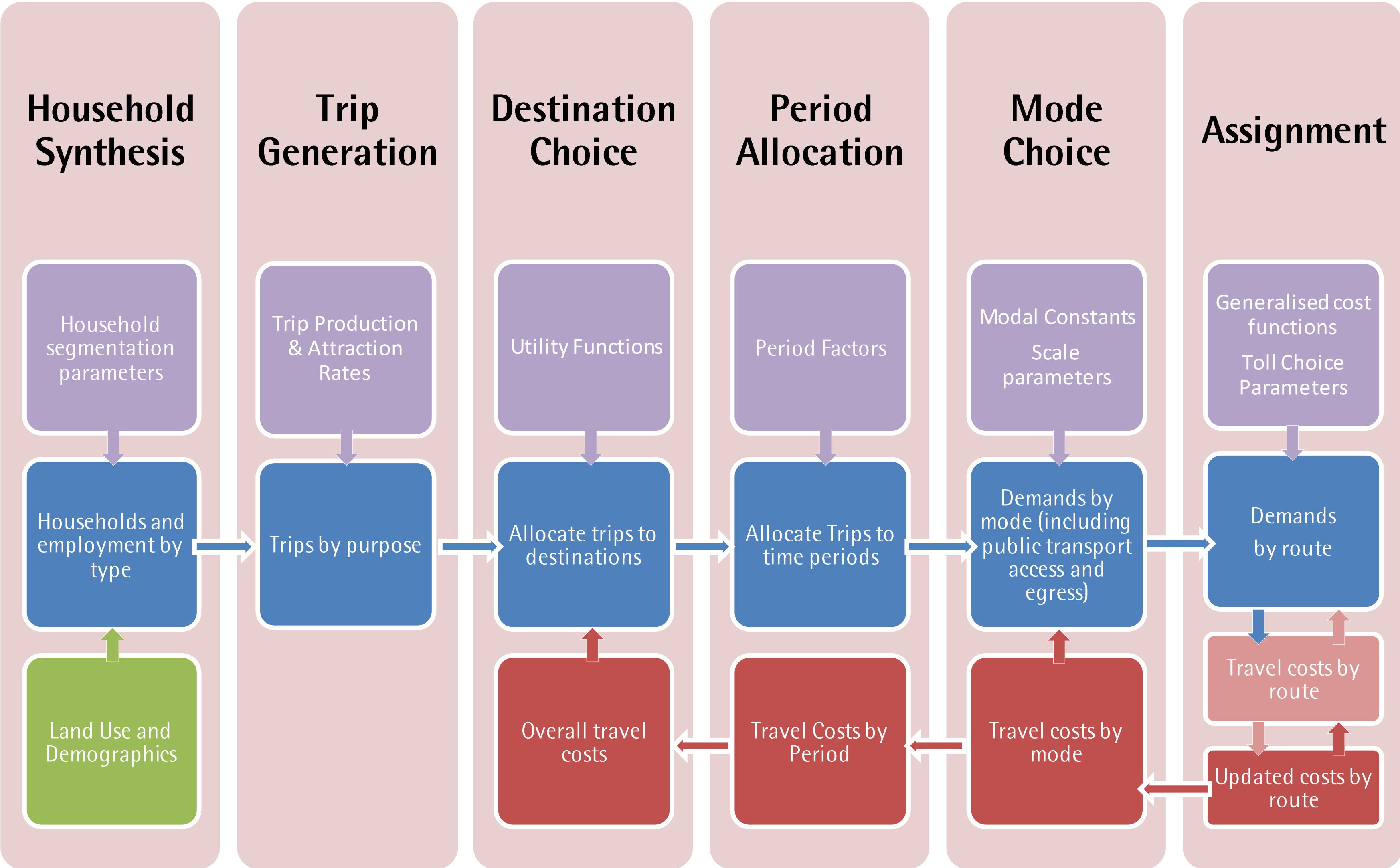



*Figure 1 – Zenith Model Process*

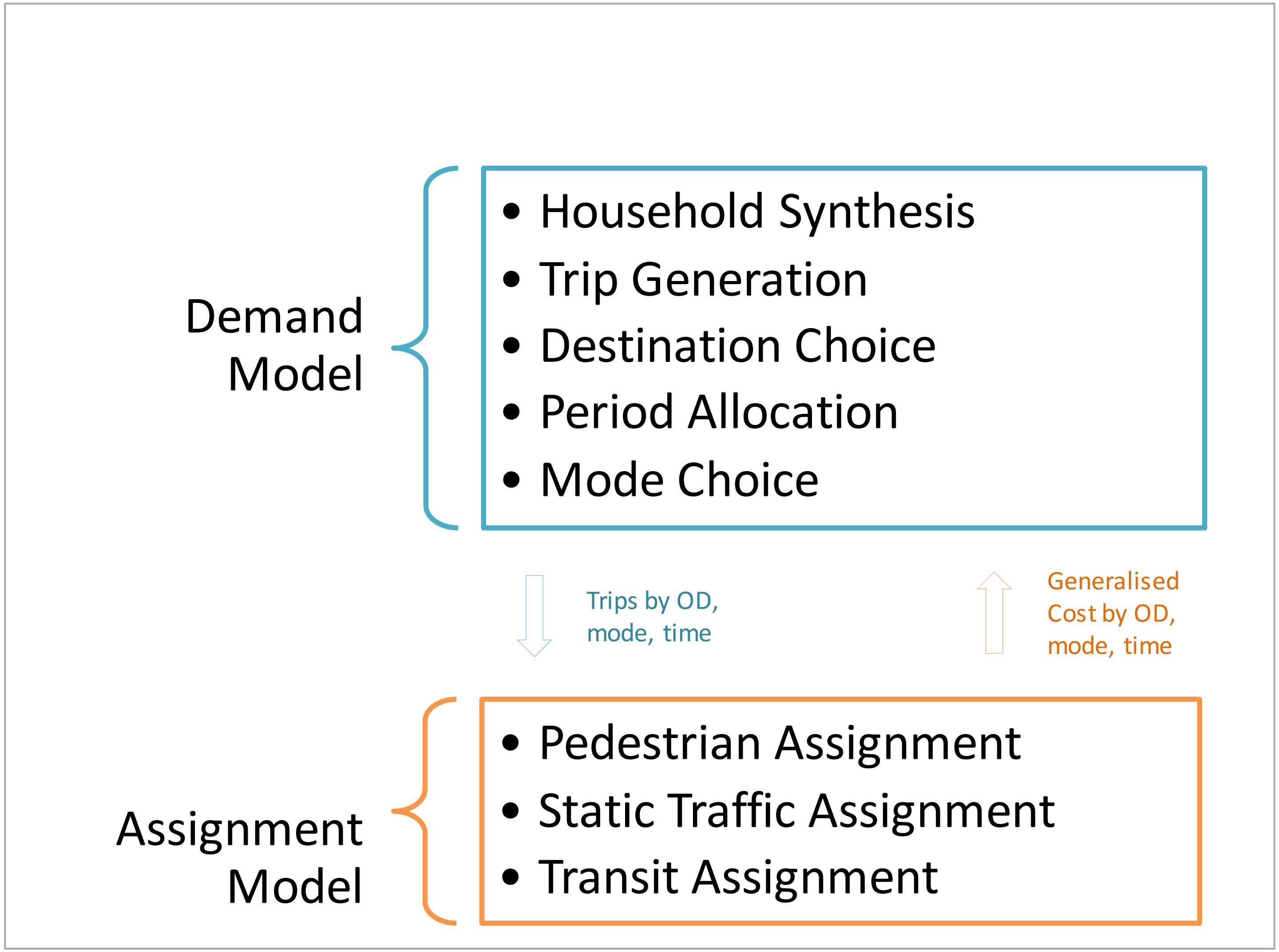



*Figure 2 – The Steps of the Zenith Model can be grouped into "Demand" and "Assignment" steps*

# 3 Inputs

This section describes the inputs to the Zenith Transit Assignment (ZTA) model.

## 3.1 Transit Network Definition

The transit network is defined using the following attributes.

- **Service frequency** – service frequency is a key determinant of expected waiting time, which is a key factor in route choice. Service frequency is expressed in services (i.e. vehicles) per hour, and is defined separately for each modelled time period (typically AM peak, PM peak, inter-peak and off-peak).
- **Route geometry –** the path taken by each route, recorded as an ordered set of links.
- **Route stopping locations** – the locations (nodes) at which each route stops to allow passengers to board or alight. Stops can be encoded as either:
    - Board and alight
    - Board only
    - Alight only
- **Route travel time** – in the Zenith model, route travel time consists of two parts:
    - **Link travel times –** for each link traversed by each route, a travel time is calculated. If the link is shared with road traffic, then the travel time is calculated as a factor (generally 1.0) of the congested road speed calculated by the traffic assignment for the relevant time period. If the link is separated from road traffic (e.g. a rail way line, ferry route, or bus-way), then the link travel time is determined using the free flow speed defined for transit during the relevant time period.
    - **Stop dwell times** – for each transit mode (e.g. train, bus, etc.), a default dwell time is defined. At each location where a route stops, the relevant default dwell time is added to the route travel time.
- **Seating and crush capacities –** for each route, a vehicle seating capacity and a vehicle crush capacity can be defined. These capacities are definable for each modelled time period.
- **Park and ride parking capacities and on-street parking environment descriptors –** the number of off-street car parking spaces available at each park and ride facility, together with a categorical variable describing the on-street parking environment surrounding each station.
- **Stop group –** each stop is assigned to a stop group. The idea is to group similar stops together, which allows Zenith to include a more sophisticated route choice algorithm. Specifically, Zenith includes a nested logit model over alternative access stops (with the nests defined by the stop groups). Typically, the stop group is defined by the main transit mode operating at the stop (e.g. train, bus, tram).

## 3.2 Access / Egress / Interchange Network Definition

Access to the transit system, egress from the transit system, and interchange between transit stops is facilitated by the underlying integrated network definition which is also used in the Static Traffic Assignment and in the assignment of pedestrians and cyclists.

For each link where walking is permitted, a walking speed is defined. For each link where cars are permitted to drive, a congested traffic speed, calculated by the Static Traffic Assignment, is used as input to the transit assignment. The congested traffic speed varies by time period.

Vehicle turn bans are also definable and are taken into account in the transit assignment.

## 3.3 Passenger Demands

The ZTA model takes as input a set of input trip matrices.

Separate trip matrices are produced for each combination of access mode / egress mode.

Most Zenith models include the following combinations:

- Walk Access / Walk Egress
- Car Access / Walk Egress
- Walk Access / Car Egress

However, more recent Zenith models split car access into park and ride and kiss and ride, resulting in the following combinations:

- Walk Access / Walk Egress
- Park and Ride Access / Walk Egress
- Walk Access / Park and Ride Egress
- Kiss and Ride Access / Walk Egress
- Walk Access / Kiss and Ride Egress

An important implication of this is that the ZTA model is not responsible for determining the access and egress modes to be used for each transit trip. Rather, it is the role of the Mode Choice Model, using skims calculated by the ZTA model for each access / egress mode combination, which determines the access and egress modes for each transit trip.

Because the ZTA model is run separately for each time period, separate trip matrices are defined per time period.

## 3.4 Model Parameters

### 3.4.1 Generalised Cost

In the Zenith model, generalised cost is typically defined in terms of local monetary currency (i.e. cents). Generalised cost parameters are then used to convert each component of the transit journey into that unit of cost.

The following model parameters control the calculation of generalised cost:

- **Default value of time –** a default value of time is defined for all passengers in the unit of generalised cost (e.g. cents) per hour.
- **Value of <u>in-vehicle</u> travel time factors –** a factor is defined per transit mode (e.g. train, bus), and is multiplied by the default value of time to produce the value of in-vehicle travel time for each mode.
- **Value of <u>access / egress</u> time factors** – a factor is defined for each access / egress mode (typically walking and car), and is multiplied by the default value of time to produce the value of time for each access / egress mode.
- **Value of <u>waiting time</u> factor –** this factor is multiplied by the default value of time to produce a value of time for waiting time.
- **Fare factor –** this factor is multiplied by the value of the transit fare prior to its inclusion in the utility function for each alternative. A value of 1.0 is typically used.
- **Access / egress penalties** - a unique penalty is defined for each access mode / transit mode combination (i.e. walk to train). A unique penalty also exists for each transit mode

/ egress mode combination (i.e. train to walk). In Zenith it the standard procedure is to set access and egress penalties equal to each other (i.e. the penalty for walk to train is set equal to the penalty from train to walk). These access and egress penalties act as alternative specific constants in the utility for each alternative. It is worth noting that it is only the differences (in absolute terms) between the penalties that matter. All of the penalties could be increased or decreased by a fixed amount, and the assignment model would remain unchanged. An equal and opposite change would be made to the modal constants within the mode choice model to ensure equal results in mode choice.

- **Transfer penalties** – transfer penalties are definable for each pair of transit modes (e.g. bus -> train). In Zenith it is standard to define unique penalties for each combination (rather than permutation) of modes. So, the penalty for bus -> train is typically set equal to the penalty for train -> bus.
- **Stop choice and route choice parameters** – these parameters are used to distribute passengers over stops and routes using a logit choice model which is described later in this document.
- **In-vehicle crowding cost curves –** the effects of in-vehicle over-crowding are taken account of by increasing the value of in-vehicle travel time for route sections which have loads approaching and exceeding capacity. The amount by which the value of time is increased is different for passengers who are standing compared with passengers who are seated. The in-vehicle crowding methodology is discussed further in Section **Error! eference source not found.Error! Reference source not found.**
- **Parking supply cost curves –** car parking availability - and its implications on rail station access choice - is reflected through the use of impedance functions. These are also discussed further in Section **Error! Reference source not found.**.

### 3.4.2 Path Building Parameters

The following parameters control various aspects of the path building algorithm:

- **Access / egress stop search –** for each origin / destination travel zone, a set of feasible access / egress stops is determined. A unique set of stops is determined for each access / egress mode (e.g. walk / car) reflecting that walking long distances is uncommon. To control the search for access / egress stops, a set of search parameters are defined:
  - **Minimum search distance**. With this criterion the modeller can specify that all stops within a certain distance from the zone (through the network) are included in the set of potential stops. This is particularly relevant for walk and bicycle access, where all stops within a reasonable walking or cycling distance should be considered. This property is definable separately for each access mode.
  - **Minimum size of the set of potential stops**. With this criterion the modeller can specify that the set of potential stops must be of a certain minimum size. In the case of remote travel zones this property can help to prevent such zones from becoming disconnected from the transit network. This property is definable separately per access mode.
  - **Minimum number of stops for specific modes**. With this criterion the modeller can specify that the set of potential stops (for each zone) should contain at least a minimum number of stops for a specific transit mode. For example, it is common to specify that the set of potential stops for car access should contain at least 1 train station. This property is definable separately per access mode.
- **Maximum number of interchanges –** the modeller is able to specify the maximum number of interchanges that can be considered as part of a transit journey (typically 3). This helps to reduce model run time with minimal loss of accuracy.

### 3.4.3 Route Choice Parameters

As will be discussed in Section 4, the ZTA model employs a multi-layered nested logit model to represent the choice between alternative paths through the transit network.

Four scale parameters can be defined, one for each layer of nesting:

- Choice between Access Stop *Groups*
- Choice between Access Stops
- Choice between Egress Stop *Groups*
- Choice between Transit Routes

The choice of parameters is very important, as it determines how sensitive the model will be to changes in generalised cost.

### 3.4.4 Algorithmic parameters

- **Number of Iterations –** the Zenith ZTA employs an iterative process, which allows crowding levels and park and ride demand to reach equilibrium. The number of iterations is fixed by the user, to facilitate “like-for-like” comparisons between scenarios.

# 4 Modelling Process

## 4.1 Overview

The algorithm employed by the ZTA model takes the following high-level form:

*For each assignment iteration*

*For each access / egress mode combination*

*Assign for all origin / destination pairs*

## 4.2 Assignment Iterations

The ZTA model can be run iteratively in order to model the effects of the following capacity constraints:

- Capacity constraints on public transport vehicles
- Parking supply constraints at park and ride facilities

Other forms of capacity constraint (i.e. on-platform) are not currently accounted for.

At the conclusion of each assignment iteration, the assigned passenger demands from the current iteration are averaged with the passenger demands from all previous iterations. The resulting averaged passenger loads are used to recalculate perceived generalised costs relating to over-crowding on public transport vehicles and at park and ride facilities, prior to the following iteration.

Normally, the ZTA model employs a volume averaging technique which weights each iteration equally although non-equal weighting schemes can be specified by the modeller if required.

## 4.3 Inclusion of Multiple Access and Egress Modes

Within each assignment iteration, the ZTA model loops over each logical combination of access and egress modes.

In most Zenith models, three combinations of access and egress modes are considered:

1. Walk Access / Walk Egress
2. Car Access / Walk Egress
3. Walk Access / Car Egress

However, more recent Zenith models split car access into park and ride and kiss and ride, resulting in the following combinations:

- Walk Access / Walk Egress
- Park and Ride Access / Walk Egress
- Walk Access / Park and Ride Egress
- Kiss and Ride Access / Walk Egress
- Walk Access / Kiss and Ride Egress

The ZTA model effectively performs a separate assignment for each access mode / egress mode combination. The model inputs which vary across access / egress modes are:

- The network used to build paths between zone centroids and potential access / egress stops (see Section 3.2)
- The path building parameters used to control the selection of potential access and egress stops for each travel zone (see Section 3.4.2)
- Access and egress penalties (see Section 3.4.1)
- The trip matrices which define origin / destination passenger demands (see Section 3.3)

## 4.4 Assignment for all origin / destination pairs

### 4.4.1 Algorithm

Within each assignment iteration, and for each access / egress mode combination, the ZTA model assigns passenger demands for each origin / destination pair to the network. This section provides a conceptual overview of the algorithm used to perform the assignment. A formal mathematical specification of the ZTA algorithm is contained in Appendix A.

The assignment algorithm is a multi-path algorithm, which assigns passengers along multiple routes based on a series of nested choices. The choices considered by the algorithm are presented in Table 1 below.

| Order | Choice | Calculation |
|---|---|---|
| 1 | Choice of which type of stop to board at (i.e. train station, bus stop, etc.) | Multi-choice (logit) |
| 2 | Choice of which stop to board at (i.e. a particular train station) | Multi-choice (logit) |
| 3 | Choice of which type of stop to finally egress at (i.e. train station, bus stop) | Multi-choice (logit) |
| 4 | Choice of which transit line (service) to board | Multi-choice (modified logit) |
| 5 | Choice of which stop to alight at | Single choice (stop with lowest generalised cost) |
| 6 | Choice of next action (egress directly to destination, or interchange at same stop, or interchange to other stop) | Single choice (action with lowest generalised cost) |
| 7 | Choice of which stop to re-board at (if walk interchanging) | Single choice (action with lowest generalised cost) |

*Table 1: The succession of choices considered by the ZTA algorithm*

The algorithm considers each choice in order, and then loops iteratively back to choice number 4 (choice of transit line) each time that the passenger is to board at a stop. The algorithm ends when the passenger choices to egress in step 6.

Four of the seven choices listed in Table 1 are calculated via multinomial logit models. The succession of choices forms what is mathematically a nested logit model, with the expected utility derived from lower level choices feeding back to higher level choices via the "inclusive value" or "logsum" calculation. Because of the need to pass information from lower level choices to upper level choices, the path building algorithm used by the ZTA builds "backwards" from each destination (rather than forward from each origin).

Three of the seven choices listed in Table 1 are "single choice", meaning that the alternative with least generalised cost is chosen. This presents something of a limitation, though this limitation is mostly addressed by the inclusion of choice #3, which causes a split between passengers in terms of what type of stop they will choose to finally egress from.

### 4.4.2 Generalised Cost

A key ingredient of all the choice models is the calculation of *generalised cost*. Generalised cost is a unit of *perceived cost* which combines monetary costs (i.e. public transport fares) with non-monetary costs (i.e. travel time).

All of the choice models employed by the ZTA algorithm use the generalised cost of each alternative to compute the allocation of demand to each alternative.

For the non-transit legs of a transit trip (i.e. access, egress and interchange) generalised cost is defined as a linear function of two elements: travel time and distance.

For the transit legs of a transit trip, generalised cost is defined as a linear function of in-vehicle travel time, waiting time, penalty, fare, in-vehicle crowding costs, and park and ride impedance.

The definable parameters used in the generalised cost function were listed in Section 3.4.1.

### 4.4.3 Transit Line Choice Model

The Transit Line Choice Model is responsible for calculating the proportion of passengers who will board each transit line at a given stop, for a given destination.

The probabilities for boarding each transit line at each stop are calculated using formula (6).

$$P_{lsij} = \frac{F_l e^{-\lambda C_{lsij}}}{\sum_{x \in L_s} F_x e^{-\lambda C_{xsij}}} \qquad (6)$$

Where:

| | |
|---|---|
| $P_{lsij}$ | fraction for line l at stop s to reach j from i. |
| $F_l$ | frequency of line l. |
| $C_{slij}$ | generalised costs when using line l at stop s to reach j from i. |
| $L_s$ | set of feasible transit lines at stop s. |
| $\lambda$ | service choice parameter. |

The set of feasible transit lines at a stop, $L_s$, is a subset of the full set of transit lines passing through a stop. Specifically, the set $L_s$ is limited by the following conditions:

- Transit lines which take passengers away from their destination are excluded; and
- If a transit line takes longer to reach the destination than another transit line, such that it would be less costly (in terms of generalised cost) to wait the full headway for the other transit line, rather than board the current transit line, then the current transit line is excluded.

#### 4.4.4 Effective Frequency and Expected Waiting Time

The effective service frequency at a stop, given all available services, is also calculated as a function of all of the candidate transit lines, is calculated using the formula below.

$$F_s = \sum_{l \in L_s} \frac{F_l e^{-\lambda C_l}}{e^{-\lambda C_{\min}}}$$

Given the calculated effective frequency, the expected waiting time at a stop is calculated as:

$$W_s = \min\left(\frac{60\alpha}{F_s}, MAXWAIT\right)$$

The parameter alpha is modeller defined, and is typically set to 0.5, which is consistent with the passengers and services arriving randomly at the stop. The parameter MAXWAIT is also modeller defined, and is generally set to 60 minutes.

#### 4.4.5 Stop choice model

For each origin / destination zone pair, the probability of first boarding the transit system at different stops is calculated. The calculation employs a standard multinomial logit model, as follows.

$$P_{sij} = \frac{e^{-\theta C_{sij}}}{\sum_{x \in S_i^A} e^{-\theta C_{xij}}} \tag{7}$$

Where:

| | |
|---|---|
| $P_{sij}$ | fraction of travellers that choose stop $s$ to reach j from $i$. |
| $S_i^A$ | set of candidate stops for a given origin $i$. |
| $C_{sij}$ | total generalised cost for travelling from stop $s$ to reach j from $i$. |
| $\theta$ | logit scale factor for stop choice |

The set of stops considered as alternatives is bounded so that it does not include every stop in the network. Specifically, a set of stops is identified which are feasible for each origin, given the access mode being used. The criteria used to determine the set of feasible stops were described in Section 3.4.2.

#### 4.4.6 Modelling of capacity constraints

The ZTA is capable of modelling the effects of two forms of capacity constraint:

- Capacity constraints on public transport vehicles
- Parking supply constraints at park and ride facilities

The effect of each capacity constraint is modelled through the use of impedance functions which act to increase the perceived generalised cost of alternatives which are experiencing high demand in comparison to their capacity.

The following sub-sections describe the calculation of impedance for each type of capacity constraint.

#### *4.4.6.1 In-Vehicle Capacity Constraints*

In highway assignment a volume delay function is employed to model the (monotonically decreasing) relationship between the number of vehicles utilizing a road and the speed of these vehicles. Higher volumes lead to lower speeds, causing an increase in travel time.

Crowding onboard public transport vehicles does not cause travel time to increase but instead increases the disutility (or discomfort) of that travel time.

The factor by which the disutility of travel time is increased is calculated by a “crowding function”:

$$f_{crowding} = F(V_C)$$

where $f_{crowding}$ is the crowding factor, $V_c$ is the volume capacity ratio and $F$ is the crowding function. The default transformation is derived from ATC guidelines (Vol 4, P70), a graphical representation of which can be seen in  where the green points denote the values used to specify the function.

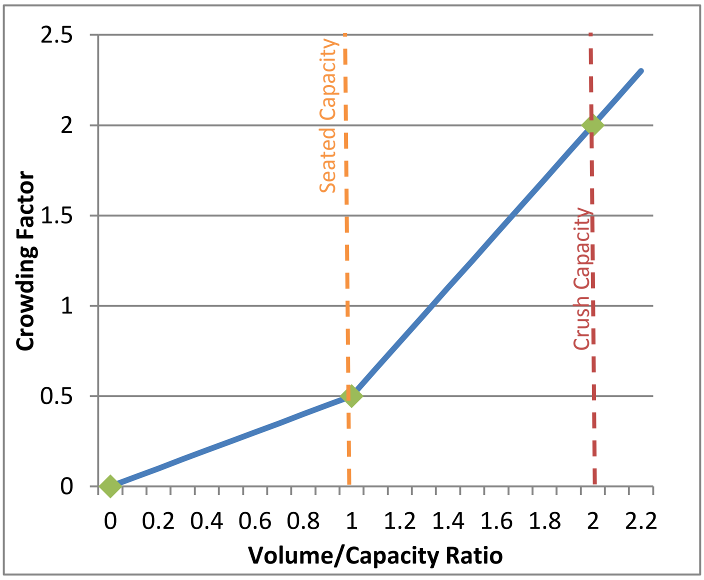



*Figure 3 – Crowding function*

In the figure, a volume / capacity ratio of 2 is used to represent crush capacity.  However, in the model, crush capacity is definable to any number larger than the seated capacity.

#### *4.4.6.2 Park and Ride Capacity Constraints*

The impedance applied to park and ride access users incorporates the availability of both dedicated off-street and surrounding on-street parking, as well as the level of parking utilisation. The cost curve applied is determined by a station’s “parking environment” category, which is assigned based on the quantity of surrounding on-street parking. One of three “parking environment” types can be applied: “No Parking”, “Parking Constrained” or “Overflow”.

The impedance function is applies as follows:

- If the utilisation (within a given time period) of the station’s off-street parking is less than 100% (i.e. the volume / capacity ratio – or V/C ratio – of the parking facility is less than 1), then the applied cost will be dependent on the level of the V/C ratio, and the “parking environment” category

- Once the station’s off-street parking capacity has been exceeded (i.e. the V/C ratio is greater than 1), the cost continues to increase and is now determined by the absolute number of parked vehicles over capacity. Again, this rate of cost increase is reliant on the station’s “parking environment” category.

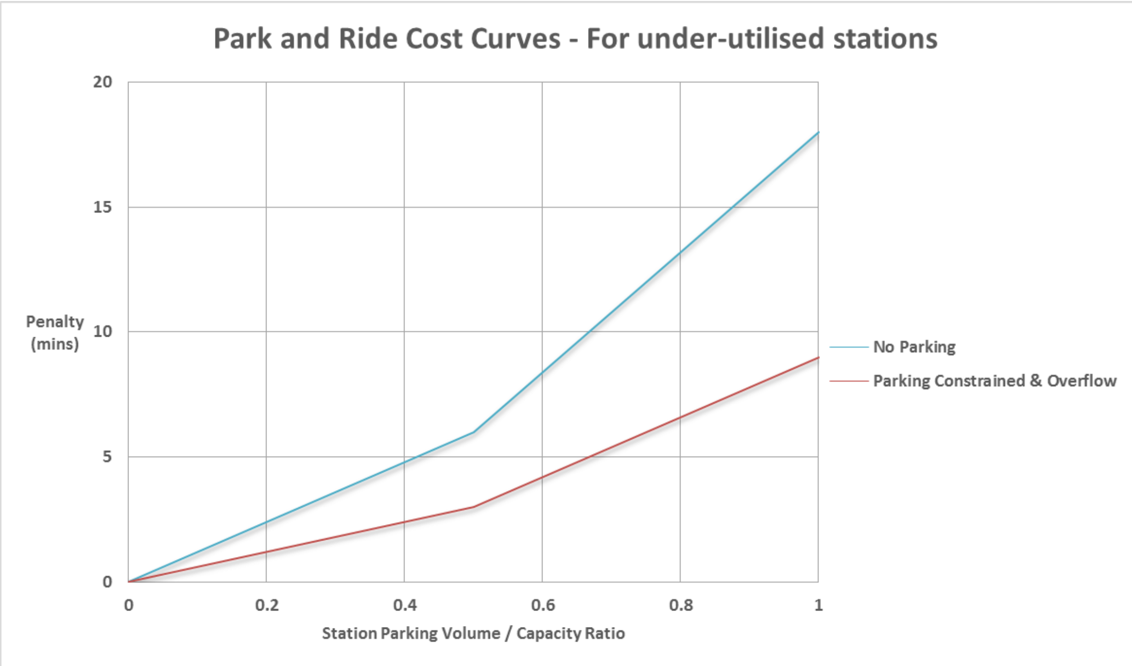



*Figure 4 – Parking supply used to define an impedance function for station parking V/C ratios of less than 1.0 – By “parking environment” category*

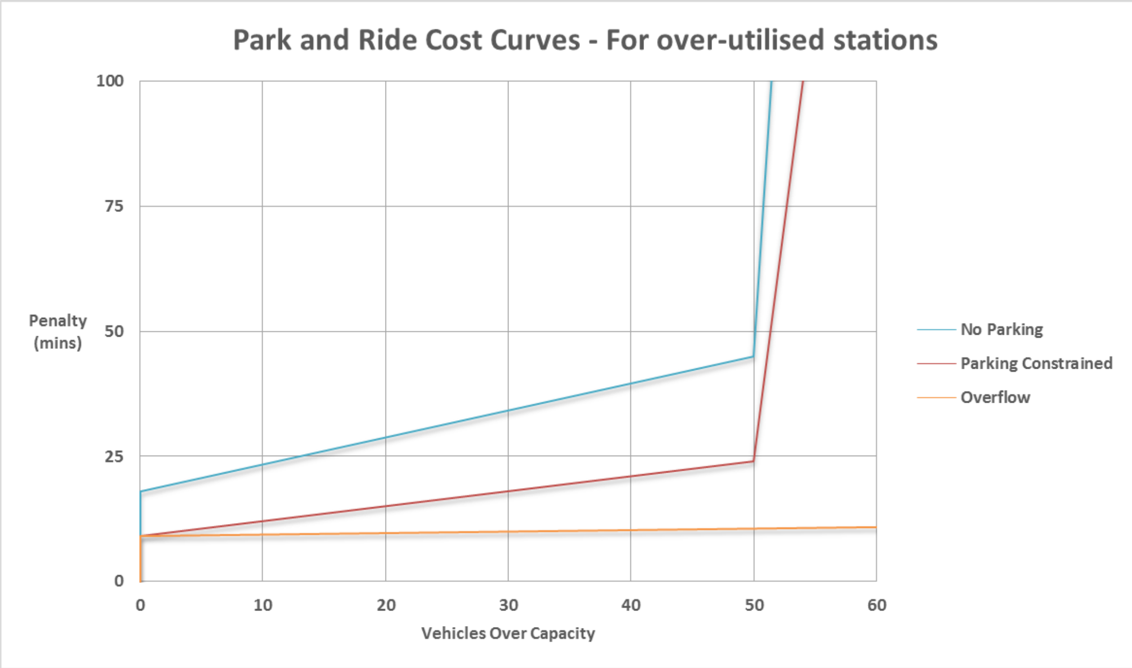



*Figure 5 – Impedance applied to all park and ride users for station parking volume / capacity ratios greater than 1.0 – by "parking environment" category*

# 5 Outputs

The standard outputs of the Zenith ZTA are:

- Link flows
    - by transit line
    - by access/egress/transfer mode
- Stop boardings, alightings and transfers
    - by transit line
    - by travel zone (first board, last alight only)
- Turning movements
- Skims (weighted average across all appropriate multi-path options)
    - Travel time
    - Distance
    - Toll
    - Fare
    - In-vehicle time / distance / cost
    - Access/Egress time / distance / cost

Some additional non-standard outputs can include:

- Cordon matrices
- Select link matrices, select link loads
- Stop-to-stop matrices

# 6 Limitations

Like all models, the ZTA has some limitations. The key limitations are:

- **Single alighting location –** due to computational complexity, the algorithm currently simplifies to an all or nothing choice on alighting location. Even though there may be a set of multiple stations that are equally likely for egress or at stop interchanging the algorithm will only choose one per path. This is generally an acceptable simplification however in some cases where the interchanging behaviour is being studied the simplification can break down – VLC are currently exploring the runtime implications for a more comprehensive set of logit choices at all stages in the PT journey.

- **Value of time is a point estimate –** within the nested logit route choice model, the parameters on time and toll are fixed point estimates (not distributions), resulting in a single value of time per user class. In reality, user perceptions of time and the value of it follow a distribution. Despite this limitation, the model is still able to model heterogeneous user preferences through the "error term" implicit in the nested logit model. In the future, we may explore the implementation of a mixed logit model, which accommodates random parameters.

# Appendix A - Algorithmic Overview

There are two key algorithms used in the ZTA:

1. Path building
2. Assignment.

The path building algorithm is a reverse propagation algorithm. It begins at a single destination, and propagates backwardly through the network of transit lines and stops. As it propagates, it calculates and stores the expected generalized cost of travel from each location in the network to the destination zone. The path building algorithm is described below.

At the completion of the path building process (for a single destination zone), the assignment algorithm begins. The assignment algorithm extracts the number of passengers travelling from each origin zone to the chosen destination from the input trip matrix, and probabilistically assigns these passengers to routes.

## Path Building Algorithm

The path building algorithm used is a reverse propagation algorithm. Beginning at a single destination, the algorithm traverses backwardly through the network, until the entire network has been explored.

*Definitions*

$L_{max}$ is the number of allowable interchanges
$R_x$ is the Rule value for x where x can be either wait or penalty
$M$ is the set of stops to be processed for the current leg
$L_s$ is the set of transit lines stopping at stop
$C_s$ is the stop cost – the cost to reach the destination from stop
$C_{l,s}$ is the line-stop cost – the cost to reach the destination from stop  using line
$C_l^{s1 \rightarrow s2}$ is the line cost between stop  and stop
$\phi_{line}$ is the scale parameter for transit line choice
$\phi_{stop}$ is the scale parameter for access stop choice.

**Step 0 – Initialisation (for a single destination zone, d)**

let $\boldsymbol{S}$ be the set of all stops

for each stop, $s$, in $\boldsymbol{S}$
 for each transit line, $l$, in $\boldsymbol{L}_s$
  set the cost of each line-stop, $C_{l,s} = \infty$
 end
 set the stop cost, $C_s = \infty$
end

let $\boldsymbol{E}_d$ be the set of egress candidate stops for zone $d$

for each stop, $s$, in $\boldsymbol{E}_d$
 add $s$ to the set of stops to be processed, $\boldsymbol{M}$
 set the stop cost $C_s$ to the travel cost from stop $s$ to zone $d$
end

set the current leg number $L$ = 0

**Step 1 – Propagate backwards along all transit lines, for all stops in M**

for each stop, $s$, in $\boldsymbol{M}$
 let $s$ be the stop where alighting occurs
 for each transit line $l$ in $\boldsymbol{L}_s$
  for each stop $s'$ prior to $s$ along the transit line $l$
   let $s'$ be the stop where boarding occurs
   calculate the cost to $d$ if travelling on line $l$ from $s' \rightarrow s$
   $C'_{l,s'} = C_l^{s' \rightarrow s} + C_s$

   compare to the currently recorded cost for boarding $l$ at $s'$
   if $C'_{l,s'} < C_{l,s'}$
    set $C_{l,s'} = C'_{l,s'}$
    add $s'$ to $\boldsymbol{M}_{next}$
   end
  end
 end
end

### Step 2 – Update Stop Cost (for all stops in $M_{next}$)

for each stop, $s$, in $\boldsymbol{M}_{next}$
 let $s$ be a stop where boarding occurs
 calculate the probability of boarding each line
 for each line, $l$, in $\boldsymbol{L}_s$
  if transit line choice method = STANDARD
   calculate the probability of boarding line $l$,
    $P_{l,s} = \frac{f_l}{\sum_{k \in L_s} f_k}$
  else *(method = ZENITH)*
   calculate the utility, $U_{l,s} = e^{-\phi_{line} C_{l,s}}$
   calculate the probability of boarding line $l$,
    $P_{l,s} = f_l U_{l,s} / \sum_{k \in L_s} f_k U_{k,s}$
  end
 end
 calculate the combined line cost, $V_s = \sum_{l \in L_s} P_{l,s} C_{l,s}$
 calculate the combined frequency, $F_s$
 if transit line choice method = STANDARD
  $F_s = \sum_{l \in L_s} f_l$
 else
  $F_s = \sum_{l \in L_s} f_l U_{l,s} / U_s^{max}$
 end
 calculate the combined wait time,
  $T_{wait} = R_{wait} \times 1/F_s$ if wait time rule is 'factor'
  $T_{wait} = R_{wait}$ if wait time rule is 'constant'

 calculate the new stop cost, $C_s' = V_s + T_{wait}$
 if $C_s' < C_s$
  update the cost at this stop, $C_s = C_s'$
 end
end

### Step 3 – Find all walk interchange candidates

set $\boldsymbol{S}_{walk} = \emptyset$
for each stop, $s$, in $\boldsymbol{M}_{next}$
 let $\boldsymbol{S}_{walk}^s$ be the set of walk interchange stops for stop $s$
 for each stop, $s'$, in $\boldsymbol{S}_{walk}^s$
  walk occurs from $s'$ to $s$
  if $C_s + C_{walk}^{s' \to s} < C_{s'}$
   $C_{s'} = C_{walk}^{s' \to s} + C_s$
   add stop $s'$ to $\boldsymbol{S}_{walk}$
  end
 end
end

**Step 4 – Update M and iterate**

set $\boldsymbol{M} = \boldsymbol{M}_{next} + \boldsymbol{S}_{walk}$
set $L = L + 1$
if $L < L_{MAX}$
 go to step 1
end

**Step 5 – Calculate access stop choice for each origin zone**

for each origin zone, $Z_o$
 let $\boldsymbol{A}_o$ be the set of access candidate stops for zone $Z_o$
 let $Y_s$ be the cost to d from zone $Z_o$, boarding at stop $s$
 for each stop, $s$, in $\boldsymbol{A}_o$
  let $a$ be the travel cost from zone $Z_o$ to stop $s$
  $Y_s = a + C_s$
 end
 let $Q_{s,Z_o}$ be the probability of boarding at stop $s$ from zone $Z_o$
 if access stop choice model = STANDARD
  let $x$ be the stop with minimum cost $Y_x$
  all trips board at stop $x$
  $Q_{x,Z_o}$ = 1
 else (access stop choice model = ZENITH)
  use logit model
  for each stop, $s$, in $\boldsymbol{A}_o$
   $Q_{s,Z_o} = \frac{\exp(-\emptyset_{stop} \times Y_s)}{\sum_{k \in A_o} \exp(-\emptyset_{stop} \times Y_k)}$
  end
 end
end

## Assignment Algorithm

The process detailed in the previous section generates a transit path from every zone to a given destination zone which can be used to assign the travel demand to that zone. This is accomplished by distributing the demand from an origin onto the set of sensible access candidates according to their relative stop costs and then allowing the load to propagate through the network.

*Definitions*

| | |
|---|---|
| $D_{o,d}$ | is the demand between origin, o, and destination, d |
| $L_s$ | stores the current iteration load at stop s |
| $L_s^{next}$ | stores the next iteration load at stop s |
| $J_{l,s}$ | stores the current iteration load boarding line l, at stop s |

### Step 0 – Initialisation

let the set of stops with load, $\boldsymbol{S}_{Load} = \emptyset$
for each stop, $s$ in $\boldsymbol{S}$
 let the current load at stop $L_s = 0$
 let the next load at stop $L_s^{next} = 0$
 $J_{l,s} = 0$ for all lines $l$, stopping at stop $s$
end

### Step 1 – Distribute load onto stops

for each origin zone, $Z_o$
 let $\boldsymbol{A}_o$ be the set of access candidate stops for zone $Z_o$
 for each stop, $s$, in $\boldsymbol{A}_o$
  $L_s = L_s + D_{Z_o,d} \times Q_{s,Z_o}$
  add stop $s$ to $\boldsymbol{S}_{Load}$
 end
end

### Step 2 – Distribute load onto transit lines

for each stop $s$ in $\boldsymbol{S}_{Load}$
 for each transit line $l$, stopping at stop $s$
  $J_{l,s} = J_{l,s} + P_{l,s} \times L_s$
 end
end

### Step 2 – Propagate demand along line until it is cheaper to alight than stay on

for each stop $s$ in $\boldsymbol{S}_{Load}$
 for each transit line, $l$, stopping at $s$
  allocate demand to links until reaching stop $m$, where:
   $C_m < C_{l,m}$
  at stop m, if it is cheapest to:
   interchange at the stop, then:
    $L_m^{next} = L_m^{next} + J_{l,s}$

   or, walk interchange to another stop, $w$, then:
    $L_w^{next} = L_w^{next} + J_{l,s}$
   or, egress, then:
    record load on egress candidate links
 end
end

### Step 3 – Iterate

$\boldsymbol{S}_{Load} = \emptyset$
for all stops, $s$, in $\boldsymbol{S}$
 $L_s = L_s^{next}$
 $L_s^{next} = 0$
 if $L_s > 0$
  add $s$ to $\boldsymbol{S}_{Load}$
 end
end
if $\boldsymbol{S}_{Load} \neq \emptyset$
 go to step 1
end