

Rijkswaterstaat
Ministerie van Infrastructuur en Milieu

MARPLE

Description and Manual

Date 13 November 2026
Status Belongs to MARPLE version 4.0.0

MARPLE

Description and Manual

Date 13 November 2026
Status Belongs to MARPLE version 4.0.0

Colophon

Published by	Rijkswaterstaat
Information	dr.ir. Henk Taale
Telephone	06-21649806
Performed by	Witteveen + Bos and Henk Taale
Layout	Henk Taale
Date	13-11-2026
Status	Final
Version Number	MARPLE 4.0.0

Table of Contents

1 General 7
1.1 Background 7
1.2 Reading guide 8

2 MARPLE 9
2.1 Model structure 9
2.2 Model characteristics 10
2.3 Steps in the model 10
2.4 Routes 10
2.5 OD-matrix 11
2.6 Optimising signal control plans 12
2.7 Assignment 13
2.7.1 No assignment 13
2.7.2 Deterministic equilibrium assignment 13
2.7.3 Stochastic equilibrium assignment 13
2.7.4 Equilibrium assignment with events and/or route information 14
2.8 Convergence 14
2.9 Output 14
2.10 Calculation time 14

3 Parameters 16
3.1 General 16
3.2 Assignment 17
3.3 Local control 18
3.4 Anticipatory control 19
3.5 Routes 19
3.6 Vehicle 20
3.7 Resuming an assignment or simulation 21
3.8 Simulation of incidents and road works 22
3.9 Toll 23
3.10 Time penalty 24
3.11 Graphs 24
3.12 Output 26
3.13 Emission parameters 27

4 Network and OD-matrix 28
4.1 Network parameters 28
4.2 Links 28
4.3 Nodes 30
4.4 Origins 31
4.5 Destinations 31
4.6 OD-matrix 31
4.7 Distribution 32
4.8 Routes 33
4.9 Paths 34
4.10 Traffic signal control 34
4.11 Ramp metering 35

4.12 Route information 36
4.13 User classes 36
4.14 Time penalty user classes 37
4.15 Time penalty network types 38
4.16 Initial traffic flows 38
4.17 Events 38
4.18 Links for selected link analysis 39
4.19 Node coordinates 40

5 Output 41
5.1 Introduction 41
5.2 Results file 41
5.2.1 General 41
5.2.2 Intermediate results per iteration 41
5.2.3 Results per link 42
5.2.4 Results per route 42
5.2.5 Results per path (route part) 42
5.2.6 Results for controllers 42
5.2.7 Output for sub-networks 42
5.2.8 Delay in zones 42
5.2.9 Number of vehicles 42
5.2.10 Network output 43
5.2.11 Emission output for the network 43
5.3 ZoneDemand.txt 43
5.4 ResultingCapcity.txt 43
5.5 SelectedLink.txt 43
5.6 SelectedLinkExtended.txt 44
5.7 LMSOutput.txt 44
5.8 MARPLE log file 45
5.9 Plots 45
5.10 Binary output files 45
5.11 JSON output files 46
5.12 Control output and input file 46
5.13 Event output and input file 46
5.14 Extra output files 47

1 General

1.1 Background

MARPLE is the abbreviation of Model for Assignment and Regional Policy Evaluation. It is a dynamic traffic assignment model that can be used to calculate the traffic flows and travel times in networks. Networks can consist of both the main road network and the provincial and urban road network. The model was originally developed within the framework of PhD research[1]. This research was looking at the interaction between traffic control and route choice in a traffic network. The aim of the research was to manage traffic, taking into account the route choices of road users (anticipatory traffic management). To determine the impact of anticipatory network management. the iterative process, as shown in the diagram in Figure 1, was developed.

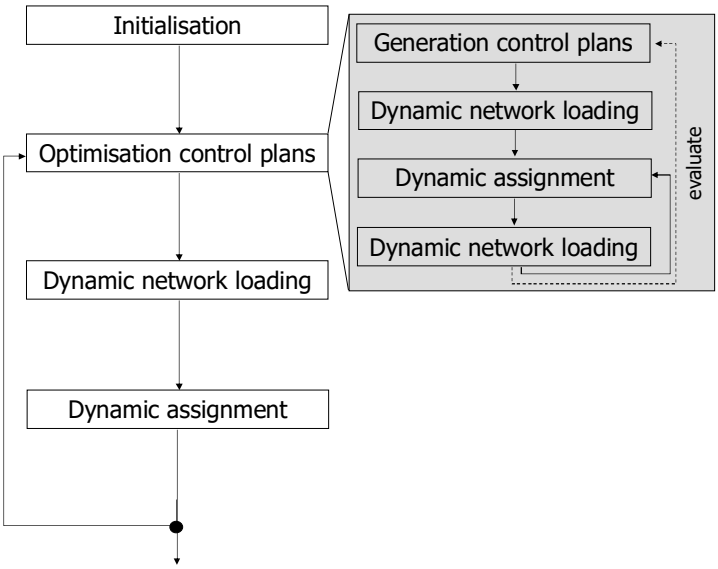

Figure 1: Schematic picture of anticipatory control

Anticipatory means that route choice was taken into account in the optimisation of the traffic control plans. This method provided good results[2], but had the disadvantage of being computationally demanding. It uses a genetic algorithm, which means that many simulations are required to determine the effect of the scheme. Therefore, a fast and simple model was needed and that is how MARPLE was developed. However, also less calculation-intensive options are available within MARPLE to optimise schemes, for example based on Webster's formulas.

[1] H. Taale (2008) *Integrated Anticipatory Control of Road Networks – A game theoretical approach*, PhD thesis TU Delft, TRAIL, Delft, November 2008, ISBN 978-90-5584-108-0.
[2] H. Taale and H.J. van Zuylen (2003) *The Effects of Anticipatory Control for Several Small Networks*, Proceedings of the 82nd Annual Meeting of the Transportation Research Board, Washington D.C.

1.2 Reading guide

This document describes in general terms how MARPLE works (chapter 2), specifies the general parameters (chapter 3) and describes how to specify the network and its properties (chapter 4). This includes, for example, the network parameters, the OD-matrix, the controls and the DVM measures in the network. Chapter 5 describes the output.

2 MARPLE

2.1 Model structure

To perform a dynamic traffic assignment, the model goes through a number of steps. These steps are illustrated in Figure 2.

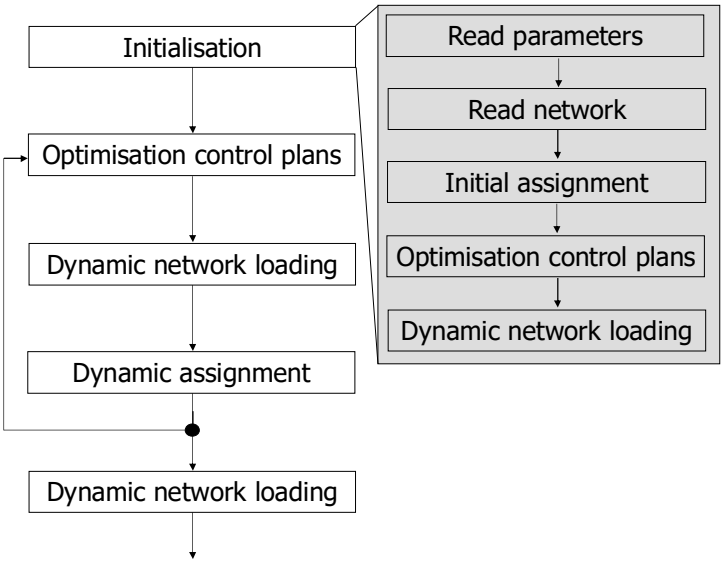

Figure 2: Steps in MARPLE

First, the file containing general parameters is read and processed. Then the file containing the network description is read. This file also contains the OD-matrix. Next, using the file *Routes.txt*, the system checks whether routes are already available and up to date. If so, the OD-matrix and routes from this file are used. If not, a new OD-matrix is generated on the basis of some of the parameters and a set of routes is determined for each OD pair.

The traffic is distributed over the available routes with an initial assignment, based on distance or free flow travel time. This creates certain traffic volumes in the network and with these link volumes, existing control plans can be optimised. A simulation is carried out to determine the traffic flow operations in the network and the corresponding indicators (volumes, speeds, travel times, network indicators, etc.) are calculated.

If iterations are desired (based on the assignment type), the signal control plans are optimised again with the prevailing traffic volumes. Then the network traffic dynamics are determined and the travel times (or travel costs) on the routes are used to reassign the traffic to the available routes. It is dynamic, because this is done for each time period. This iterative process is carried out until an equilibrium situation is reached, for which the user can influence the number of iterations and the deviation from the 'real' equilibrium.

2.2 Model characteristics

- Travel times on links are calculated with travel time functions.
- These functions are based on the degree of saturation (I/C ratio), using formulas from the literature[3] to determine the capacity. So, there is no direct relationship between speed and density. A number of parameters of these functions are adjustable (see subsection 3.6).
- The traffic flow on a link depends on the travel time and the capacity of the (controlled) links.
- The model takes into account the available space on a link and blocking back.

2.3 Steps in the model

- Initialise all variables and the starting link loads (if requested);
- Determine the free travel time and capacity for each link;
- Determine for each node the splitting rate (per period), this is done based on destinations;
- For each time step:
  - Handle the specified events and change link variables;
  - Determine the travel time and delay for each link;
  - Determine the *desired* outflow for each link;
  - Determine the inflow for each node;
  - Determine the remaining space for each link;
  - Determine the inflow for each link (route choice);
  - Determine the *possible* outflow for each link (blocking back);
  - Determine the queue for each link;
  - Determine the volume, speed and density for each link;
- Calculate travel time and delay per route and sub-route and per time step and time period;
- Calculate network indicators (also for sub-networks): TDT (total distance travelled), TTS (total time spent), TD (total delay), etc.

2.4 Routes

Possible routes between OD pairs are automatically generated and written to the file *Routes.txt*. An existing *Routes.txt* is only overwritten if the location or name of the network file has changed, or if one of the following parameters has changed: *ThresFlow*, *ScaleFlow*, *nrRand*, *nrRoutes* or *nrTimePeriods* (see chapter 3 for a description of the parameters). Finally, new routes are also generated if a file *Detours.txt* exists that is newer than the route file (see section 4.8). This file contains the specification of possible detours. A route file that is going to be overwritten, is copied to the file *RoutesOld.txt*. If the generation of routes is not necessary, the existing file *Routes.txt* is used. This reduces the calculation time.

[3] Akçelik, R. (2003) *Speed-Flow Models for Uninterrupted Traffic Facilities*, Technical Report, Akçelik and Associates Pty Ltd, Melbourne, Australia.
Transportation Research Board (2000) *Highway Capacity Manual*, National Research Council, Washington D.C.
Wu, N. (2001) *A universal procedure for capacity determination at unsignalized (priority-controlled) intersections*, Transportation Research Part B, **35** (6) 593-623.

Routes.txt also contains the OD matrix. It is checked whether the total demand in the network file is equal to the value written in the *Routes*.*txt* file. If this is not the case, the file is generated again. It is possible that the total demand remains the same, but that the distribution over time changes. This is not checked! Therefore, if the OD matrix is changed in this way, delete the existing *Routes.txt* so that new routes can be created, which are consistent with the OD matrix.

Routes are generated per OD pair. First the real shortest route is determined. This can be in distance or in free travel time (parameter *linkCost*), where the free travel time can be taken including or excluding delays at junctions. This is controlled with the parameter *junctionDelay.*

To calculate multiple routes, a random error term is then added to the travel time/distance of each link in the network and the shortest route is determined again. In this way, *nrRand (e.g.* 50) routes are generated. From these generated routes, it is checked whether there are any routes that are almost the same. If the links in two routes are overlapping for more than a certain percentage, the two routes are considered to be equal and only the shortest route is used. The value for this percentage is specified with the *linkEqual* parameter. Of the remaining routes, the *k* shortest routes are used as alternative routes between an OD pair (*k* = *nrRoutes*). This method has the disadvantage that the free travel time is used as the basis. Therefore, some important routes may fall outside the set of the routes.

2.5 OD-matrix

When reading the OD matrix, the minimum flow per OD pair is considered. With the parameter *ThresFlow* a threshold value can be specified. OD pairs with lower flows (considered over all time periods) are removed from the set of OD pairs and no routes are calculated for those OD-pairs. For large OD matrices with lots of very small OD flows, this can save calculation time. To compensate for the removed low OD flows, the flows of all OD pairs with the same origin and all OD pairs with the same destination are increased. This can be an equal increase or an increase proportional to the size of the OD relationship (specified with parameter *ODdist*). If it is not possible to increase the OD pairs, for example because there is no OD pair with the same origin or destination and sufficient flow, the total OD matrix is increased with a proportional factor. This is done by automatically changing the *ScaleFlow* parameter.

In addition, it is possible to change the traffic demand of certain OD relations via the selected link analysis. In the network, certain links can be specified for which a selected link analysis should be performed. For these links also a percentage can be specified. At the start of the simulation it is checked whether this percentage is smaller or larger than zero. A selected link analysis is then carried out for the considered link and the OD relations that use that link have their traffic demand reduced or increased by the specified percentage (see section 4.16). This option is useful to quickly calibrate an OD matrix on a number of measuring points. Of course, one should be careful while doing this and this is not a replacement for a full scale OD matrix estimation and calibration. But it can help to calibrate the traffic situation in a small and specific part of the network.

Because a dynamic OD matrix is used, there are two possibilities for a demand profile (see Figure 3): a profile where the demand is continuous during the period but can change every period (solid line) or a profile, where the demand changes every time step, making it more continuous (dashed line). This option can be specified with the *DemandPar* in the network file.

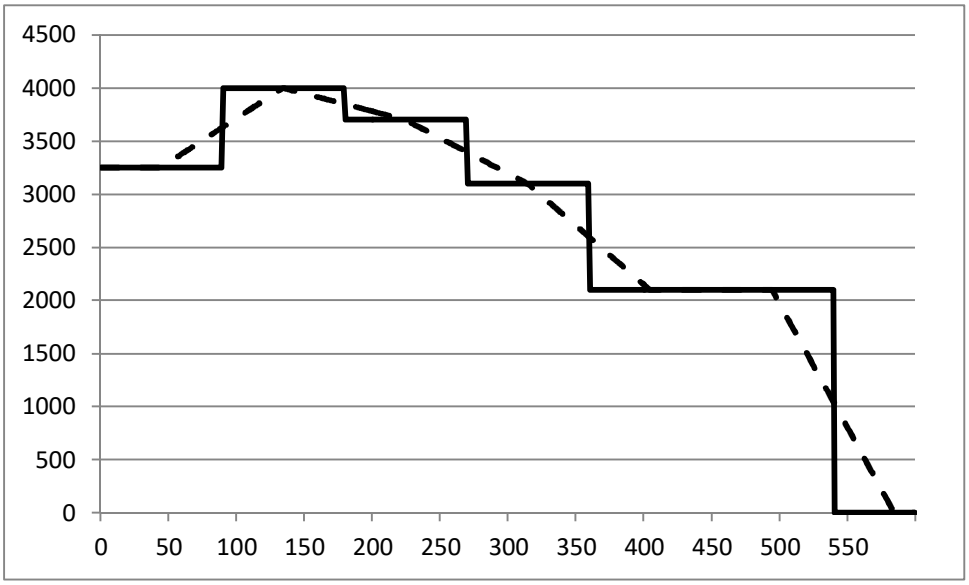

Figure 3: Possible demand profiles

2.6 Optimising signal control plans

During the simulation, the intersection signal control plans in the network are optimised, but only if the parameter *Optimisation* is larger than zero. This applies to nodes of type 3 (controlled node) and type 4 (metering node). Here, the green times of the various phases in the control plan are optimised within specified minimum and maximum times. To simplify the calculation, every incoming link is considered a separate phase.

The procedure is as follows:

- Determine the controlled links;
- Determine the flow on those links per time period;
- For each controlled intersection, determine the green times and cycle times per simulation period or simulation step. By default, this is done with Webster's formulas, but other algorithms are also available;
- For ramp metering:
  - Determine the upstream flow and the downstream (remaining) capacity;
  - Determine the metering time based on the amount of traffic allowed to enter the motorway, taking into account a specified minimum and maximum metering time. The algorithm is a demand-capacity algorithm (allowed on-ramp flow = capacity downstream - flow upstream).

2.7 Assignment

The initial assignment of the traffic to the available routes (at the start of the simulation or assignment) is on distance (*initialAssign=0*) or the free flow travel time. If travel time is used, the free flow travel time can be chosen including (*initialAssign=1*) or excluding (*initialAssign=2*) the extra travel time at controlled intersections, roundabouts or priority junctions. The initial assignment itself is done with the C-logit model[4].

For the regular assignment (in the iterative process) there are three possibilities (parameter *Assign*): no assignment (*Assign* = 0), deterministic assignment (*Assign* = 1) or stochastic assignment (*Assign* = 2).

2.7.1 No assignment

The traffic is distributed over the available routes via the initial assignment. Then, the simulation is carried out and travel times and network indicators are calculated. After this the simulation stops and the results are written in the output file.

2.7.2 Deterministic equilibrium assignment

With this type of assignment, a deterministic user equilibrium is calculated according to Wardrop's principle: each traveller cannot unilaterally improve his travel time (or travel costs) by changing route. Also, each unused route has a higher travel time than the used routes, for which the travel times are equal. The assumption here is that every traveller is perfectly informed about the expected travel time.

2.7.3 Stochastic equilibrium assignment

If the assumption of perfect information from the deterministic equilibrium assignment is weakened, the stochastic assignment is the alternative. In the equilibrium situation, the travel times on the available routes are not exactly, but approximately equal. A parameter (parameter *teta*) determines how 'approximately' this is. The assumption is now that in equilibrium, the perception of the travel time is the same. By weakening the assumption of perfect information, it is generally assumed that the stochastic equilibrium assignment gives more realistic results than the deterministic one.

In MARPLE, the well-known C-logit route choice algorithm is used[4]. This algorithm explicitly takes into account possible overlap in routes (a shortcoming in other stochastic algorithms). For smoothing the route flows (necessary to ensure convergence), a slightly modified version of the Multiple Successive Average method (MSA) is used. The flows (*SmoothFlow*), and if needed also the green times (*SmoothG*), are smoothed with

$$f(j) = f(j-1) + \delta(j)\big(f(j) - f(j-1)\big),$$

where $\delta(j)$ is given a slightly different value than in normal MSA.

[4] Cascetta, E and Nuzzolo, A and Russo, F and Vitetta, A. (1996) *A Modified Logit Route Choice Model Overcoming Path Overlapping Problems: Specification and some Calibration Results for Interurban Networks*, Proceedings of the 13th International Symposium on Transportation and Traffic Theory, Lyon, France, pp. p. 697-711.

2.7.4 Equilibrium assignment with events and/or route information

The disadvantage of an equilibrium assignment is that it includes assumptions about the degree of information of road users. Therefore, it is normally not a good option to simulate incidents and route information, because road users will already experience the consequences of the incident in the assignment and will (partially) automatically adjust their route. This problem can be solved by running a normal equilibrium assignment first. Then, an extra simulation with an incident, or extra iterations with route information, or a combination of incidents and information is carried out. This gives the opportunity to properly simulate the effect of incidents, variable message signs and the combination thereof (see further section 3.7).

2.8 Convergence

The criterion for convergence of assignment is calculated in MARPLE on the basis of flows. For each OD pair, the difference between the old and new (from the last and previous iteration) flows for each route is calculated. This difference is expressed as a percentage of the total demand (per OD pair) and indicates the amount of traffic that has changed routes. The assignment is considered converged if the maximum value of this percentage (calculated over all OD pairs and time periods) is smaller than a specified threshold (e.g. 1%). The threshold value can be set via the parameter *ConvError*.

2.9 Output

The output of the model consists of the following data (per time period):
- Flow, speed, standard deviation of the speed and density per link;
- Travel time and delay per OD relation, route and route part (if specified);
- Travel time, delay and speed per route and route part (if specified).

And also network indicators for the entire network and for each network type:
- Number of vehicles;
- Distance travelled;
- Time spent;
- Average speed;
- Delay.

In addition to the traffic engineering output, MARPLE is extended with output for emissions. Based on the VERSIT model of TNO, emission factors are derived that fit the macroscopic nature of MARPLE[5]. With these emission factors, the emission of CO, CO_2, HC, NO_x and PM10 is calculated at network level.

2.10 Calculation time

The calculation time and memory usage for a simulation or assignment depends mainly on:
- The number of routes and therefore also the number of OD pairs;

[5] Gerdien Klunder and Ulrike Stelwagen (2012). *Manual EM4MARPLE – Emission Module for RBV – Version 1.0*, TNO, February 2013.

- The number of links in the network;
- The number of time steps into which the simulation period is divided.

The number of routes is highly dependent on the number of OD pairs with a traffic demand larger than zero. Limiting the number of OD pairs included in the simulation results in a smaller number of routes and therefore less memory usage and calculation time.

To limit the number of OD pairs, the limit for the minimum flow of an OD pair (*ThresFlow*) can be used. It is also advisable to merge links with the same characteristics as much as possible (recognisable in OmniTRANS by the shape points) so that there are fewer links, fewer nodes and fewer splitting rates in the model.

The size of the time step (and thus the number of time steps) must be adjusted to the travel time on the shorter links. Choose here a time step as large as possible (*LTimeStep*), but make sure that there are no or only a limited number of links in the network whose travel time is shorter than the time step. MARPLE can handle this, but it is not yet clear what effect this has on the results.

3 Parameters

The parameters for the simulation and the assignment are specified in a separate file. For MARPLE simulations with the OmniTRANS plugin this file is called *marple_parameters.txt*, but the file can also be named differently. The parameters are grouped in a number of different clusters. The most important clusters are: general parameters (found in //General), assignment parameters (found in //Assignment), parameters for local control (//LocalControl), route parameters (//Routes), vehicle parameters (//VehPar) and output parameters (//Output).

The subsections below give an overview of all clusters of parameters and explain the meaning of each of these parameters. Also the default value in the OmniTRANS MARPLE-plugin is mentioned.

3.1 General

Cluster	//General	
Parameter	**Meaning**	**Default**
Assign	Parameter to indicate the type of assignment: 0 = no assignment; 1 = DDUE (deterministic dynamic user equilibrium); 2 = SDUE (stochastic dynamic user equilibrium).	2
Optimisation	Parameter to determine the type of optimisation for the green times: 0 = no optimisation; 1 = local optimisation with local algorithm per simulation period; 2 = local optimisation with local algorithm per time step; 3 = anticipatory optimisation. The optimisation is done only for the signal controlled intersections for which it is indicated that the green times must be optimised (node type = 3). The local optimisation algorithm is determined with *LContrMethod*.	1
Metering	Parameter to switch ramp metering in the network on or off: 0 = no ramp metering; 1 = local ramp metering with capacity-demand algorithm. Ramp metering only applies to nodes which are specified as metering nodes (node type = 4).	1
SmoothG	Parameter to smooth the green times, using the actual and green times from the previous iteration: 0 = no smoothing; 1 = smoothing with $g(j) = g(j-1) + \delta(j)\big(g(j) - g(j-1)\big)$.	0

Cluster	//General	
Parameter	**Meaning**	**Default**
SmoothFlow	Parameter to smooth the route flows, using the actual route flow and the ones from the previous iteration: 0 = no smoothing; 1 = smoothing with $f(j) = f(j-1) + \delta(j)\big(f(j) - f(j-1)\big)$.	1
DelayType	Parameter to change the reference speed used for the calculation of delay on links: 0 = delay is calculated based on maximum speed; >0 = delay is calculated based on of the maximum speed or the imposed speed (as is used with events, see section 4.17).	1
InitialFlow	Parameter to determine the initial state (flows) in the network (at the start of the simulation): 0 = initial link flows are zero; 1 = initial link flows are equal to the flows for the first time period.	1
ThresFlow	Minimum demand threshold for an OD pair to be met in one (or more) time periods. The minimum demand must always be larger or equal than 0.	1
ConvErr	Maximum accepted difference in the flow on a route between two iterations of the assignment. Expressed as a percentage of the total demand. If for all routes and time period the difference in flows is less than this value, the assignment is considered to be converged.	1
minCounter	Minimum number of iterations for the assignment.	5
maxCounter	Maximum number of iterations for the assignment.	30

Notes

- *Assign*: If the value is 0 (no assignment), the traffic is assigned according to the free travel times on the different routes or according to the distances (see parameter *initialAssign*)
- *Metering*: Checks whether there are metered nodes present in the network. If not, this parameter is set to 0;
- *ThresFlow*: Preferably use a value of 1. If necessary, use 2 or more to further reduce the number of routes (and thus the calculation time). See also section 2.4 about determining the routes for each OD pair.
- *minCounter*, *maxCounter*: use 0 and 5 for a quick calculation and 5 and 30 for a normal calculation.
- The assignment is stopped when *ConvErr* is met, or when the maximum number of iterations is reached.

3.2 Assignment

This cluster is used to set the parameters of the assignment algorithm.

Cluster	//Assignment	
Parameter	**Meaning**	**Default**
VI-Rho	Parameter for the projection method, the method used if *Assign* = 1 (deterministic assignment).	10
CF-Beta	Parameter of the C-logit model used, if *Assign* = 2	1
CF-Gamma	Parameter of the C-logit model used, if *Assign* = 2	2
Kirchhoff	Parameter for the type of logit assignment used: 0 = stochastic assignment with overlap in routes (C-logit); 1 = assignment according to Kirchhoff's principle; 2 = stochastic assignment without overlap in routes.	0
initialAssign	Parameter for initial assignment: 0 = Initial assignment based on distance; 1 = Initial assignment based on free travel times, including delays at intersections; 2 = Initial assignment based on free travel times, excluding delays at intersections.	1
mobScan	Parameter to assign traffic according to the Mobility Scan: 0 = no assignment according to the Mobility Scan; 1 = assignment according to the Mobility Scan. This is an extra option to do an assignment. First, a regular assignment is done. From this assignment the travel times are derived. Using the free flow travel times and the travel times from the assignment, two routes are determined. Then traffic is equally distributed among these two routes (50%-50%).	0

3.3 Local control

Parameters for (real-time) optimisation of the controlled intersections.

Cluster	//LocalControl	
Parameter	**Meaning**	**Default**
LContrMethod	The optimisation algorithm for local control by time period and time step: 1 = Webster (default); 2 = Smith's P0; 3 = Genetic algorithm; 4 = Backpressure algorithm; 9 = Input in the network file of dynamic green times; 99 = Input from an external file of dynamic green times.	1
LOptPeriod	The length of the period (in seconds) over which local optimisation is performed. Typically in the order of minutes.	300

Cluster	//LocalControl	
Parameter	**Meaning**	**Default**
AdjustSat	Automatic adjustment of exit capacities for controlled links to take into account turning lanes for left and right turning traffic. 0 = do not adjust automatically; 1 = automatic adjustment based on total incoming traffic and largest outgoing flow.	0

Notes

- If *LContrMethod* = 99 the simulation stops every control period (*LOptPeriod*) to generate the file *LinkOutput.csv* in which the current link data is stored. After that it waits for the file *InputControls.csv* in which new control information is given to the simulation. The format of these files is specified in section 5.12.
- At the start of the simulation the file *InputControls.csv* should be available, but it is deleted after every control period.

3.4 Anticipatory control

Parameters for (real-time) anticipatory optimisation of the controlled intersections.

Cluster	//AntControl	
Parameter	**Meaning**	**Default**
nrAssign	The number of iterations in the assignment in anticipatory control to predict the route choice.	1
nRTperiods	The number of time periods for which a forecast is made in real-time anticipatory control. *Not used at this moment*.	1
optCriterion	The optimisation criterion used in anticipatory control: 0 = total delay in the network; 1 = total delay in the network + entrance delay; >1 = different combinations of delay and vehicles in network.	1
LTimeStepOpt	Time step (in seconds) used in the simulations for the anticipatory optimisation. Less accurate if this is higher than *LTimeStep*.	60

3.5 Routes

Cluster	//Routes	
Parameter	**Meaning**	**Default**
nrRoutes	The maximum number of routes used per OD pair.	4
nrRand	Number of routes generated from which the *nrRoutes* shortest routes are selected.	50
scaleFac	Scaling factor for the random error term used to generate the routes. More than 99% of the generated routes are shorter than (1+3*scaleFac) times the shortest route.	0.66

Cluster	//Routes	
Parameter	**Meaning**	**Default**
linkCost	Parameter used for calculating the shortest route: 0 = calculation based on distance; 1 = calculation based on free travel time.	1
linkEqual	Maximum percentage of links that are equal in two routes to distinguish these routes. If the percentage is higher, the routes are considered equal (1.0=100%).	0.85
junctionDelay	Parameter that determines whether intersection delay is included in the calculation of shortest routes: 0 = intersection delay is not taken into account; 1 = intersection delay is included.	0
ODdist	Parameter which determines how the traffic demand of small OD-relationships (smaller than *ThresFlow*) are distributed to the other OD-relationships: 0 = uniform; 1 = in proportion to the size of the remaining relationships.	1

Notes

- *linkCost* has the same kind of function as *initialAssign*; however, *linkCost* is used in the generation of the routes and *initialAssign* has to do with the assignment. See also section 2.4.

3.6 Vehicle

Cluster	//Vehpar	
Parameter	**Meaning**	**Default**
VehLen	The average vehicle length used in the simulation.	7.5
TruckV	The free speed of trucks. This parameter is only used for the retrospective calculation of travel times for trucks.	80
minV1	Minimum speed for links with free speed over 90 km/h.	10
minV2	Minimum speed for links with free speed below 90 km/h.	10

Cluster	//Vehpar	
Parameter	**Meaning**	**Default**
Ja1 ... Ja9	Parameters to indicate the speed at congestion for different types of links:	
	Ja1 = speed in congestion for links with free speed > 110 km/h;	85
	Ja2 = speed at congestion for links with 90 < free speed <= 110 km/h;	75
	Ja3 = speed at congestion for links with 70 < free speed <= 90 km/h;	60
	Ja4 = speed at congestion for links with 60 < free speed <= 70 km/h;	50
	Ja5 = speed at congestion for links with 50 < free speed <= 60 km/h;	40
	Ja6 = speed at congestion for links with 40 < free speed <= 50 km/h;	20
	Ja7 = speed at congestion for links with 30 < free speed <= 40 km/h;	15
	Ja8 = speed at congestion for links with 20 < free speed <= 30 km/h;	10
	Ja9 = speed at congestion for links with 0 < free speed <= 20 km/h.	10

3.7 Resuming an assignment or simulation

Cluster	//Resume	
Parameter	**Meaning**	**Default**
ResumeSim	Parameter to determine whether the assignment or simulation should be resumed with the network state of a previous iteration or simulation. 0 = do not resume. 1 = resume simulation. 2 = resume simulation per time step. *Not used at this moment*.	0
ResInputFile	File that contains the network state from a previous assignment or simulation.	
ResOutputFile	File that is used to store the network state of the current assignment or simulation.	

Notes

- A simulation can be started with the network state from the end of a previous assignment or simulation. This can be useful for example for the simulation of several days after each other. Normally, this cannot be done in one simulation due the limits of memory use.

3.8 Simulation of incidents and road works

Cluster	//EventSimPar	
Parameter	**Meaning**	**Default**
EventSimType	Parameter that determines whether an additional assignment should be made after the normal assignment to simulate incidents or road works, without or with extra information. 0 = no additional simulation of certain events, 1 = additional simulation/assignment of certain events, 2 = additional simulation/assignment of certain events with available route information, 3 = additional simulation/assignment with route information only. For options 2 and 3 is route information only available for the links whose *route info* parameter on the *//VMSinfo* record is set to 1. If for this record the *incident* parameter is also set to 1, only information during incidents is simulated. For this additional simulation/assignment the type of assignment *EventSimAssign* is used with *EventSimNrIter* number of iterations. The extra assignment is done only for the traffic using the links for which information is given.	0
EventSimAssign	The type of assignment to be used for the additional simulation of events. 0 = no assignment; 1 = DDUE (deterministic dynamic user equilibrium); 2 = SDUE (stochastic dynamic user equilibrium).	2
EventSimNrIter	Maximum number of iterations for the assignment.	1
EventExternal	Parameter to determine if external events are read from the file *inputEvents.csv*. 0 = no external events are read, 1 = reading external events, every event period.	0
EventPeriod	Length of time period for which external events are read	300

Notes

- For the simulation of incidents and road works with or without route information, it is important to assume an existing situation and continue the calculation with that, instead of running a new equilibrium assignment. Especially for incidents this is not realistic. To make this possible, the user can specify that after the normal equilibrium assignment, an extra assignment should be performed. In doing so, the *route flows* from the output of the normal equilibrium assignment are copied to a new network file as *initial flows* (see section 4.16). This network file is used in the extra assignment, and, moreover, the incident or road work events specified by the user are included (see section 4.17). These events are not used in the normal equilibrium assignment. The user can also specify whether it should be a simulation or an assignment with a specific number of iterations.
- The new network file is saved with the same name as the old network file, but with the addition *"-Incident"*. This also applies to the output file.

- Another option is that the user can specify that extra information about those events (incidents or road works) is available for the road users (*EventSimPar* = 2). Then, after the simulation/assignment with the extra events, another simulation/assignment is run in which only the OD relations with a route over the links with route information are assigned again.
- The output file at *EventSimPar* = 2 is saved with the same name as the normal output file, but with the addition '*-Incidents-VMS.txt'.*
- In addition, it is possible that there are no events, but that extra information is available for certain links (*EventSimPar* = 3). In this case, after the normal simulation/assignment, an additional simulation/assignment is run for which only those OD relations with a route over the links with route information are re-assigned.
- If *EventExternal* = 1 the simulation stops every event period (*EventPeriod*) to generate the file *LinkOutput.csv* in which the current link data is stored. After that it waits for the file *InputEvents.csv* in which new event information is given to the simulation. The format of these files is specified in section 5.12.
- At the start of the simulation the file *InputEvents.csv* should be available, but it is deleted after every event period.

3.9 Toll

Cluster	//TollPar	
Parameter	**Meaning**	**Default**
TollType	Parameter to determine the type of tolling: 0 = no tolling; 1 = tolling on every link; 2 = tolling on specific links.	
ValTime1..4	Value of Time for 4 different user or vehicle types	

Notes

- For *TollType* = 1 there should be a cluster *//NetTolls* in the network file. This cluster specifies for every sub-network (*Nettype*) and every time period the tolling price in euro per kilometre.
- For *TollType* = 2 in the network file the *//LinkTolls* should be specified, which gives for certain links, for every time period and the 4 user or vehicle types, the price for that link.

3.10 Time penalty

Cluster	//TimePenaltyPar	
Parameter	**Meaning**	**Default**
TimePenaltyType	Parameter to determine the type of tolling: 0 = no time penalty; 1 = time penalty for links with flow-capacity ratio larger than *IC-threshold*; 2 = time penalty for links belonging to certain network types (*nettype)*.	
IC-threshold	Value for the flow-capacity ratio for *TimePenaltyPar* = 1 (0-100%).	
TimePenNrIter	Number of extra iterations to run with an active time penalty	

Notes

- This option gives the opportunity to add extra travel time to certain links or sub-networks (*nettype*) with the goal to persuade traffic to change their route. In that way it is possible to decrease for certain bottlenecks of certain areas in the network. Also, for this option first an equilibrium assignment is done and after that the extra time (here called a penalty) is added to the specified links or links belonging to the specified sub-network. Then, extra iterations (*TimePenNrIter*) are run to determine the effect.
- For *TimePenaltyPar* = 1 all links in the simulation are checked if the flow-capacity ratio is above *IC*-threshold. If this is the case for certain links and time periods and extra travel time is added to the routes to which these links belong. The extra travel time is specified for certain user classes with *//UserClassTP* (see section 4.14).
- For *TimePenaltyPar* = 2 all routes with links belonging to certain sub-networks are penalised with extra travel time. The sub-networks are defined with *//NetTypePenalty* (see section 4.15). The amount of extra travel time is specified per user class with *//UserClassTP* (see section 4.14).

3.11 Graphs

Cluster	//PlotPar	
Parameter	**Meaning**	**Default**
MFDplot	Parameter to generate an MFD plot: 0 = no MFD plot; 1 = MARPLE generates MFD plot.	
MFDperiod	Parameter to specify the period over which MARPLE aggregates the data for the MFD plot (minutes).	
ContourPlot	Parameter to generate a speed contour plot: 0 = no speed contours plot; 1 = MARPLE generates speed contour plot.	

Cluster	//PlotPar	
Parameter	**Meaning**	**Default**
StartTime	Parameter to specify the start time of the speed contour plot. This is to make the plot neater. The parameter is in hours on a 24-hour scale, so 6 = 06:00 and 15.5 is 15:30.	
FlowPlot	Parameter to generate a plot of link flows: 0 = no link flow plot; 1 = MARPLE generates a plot of link flows.	
SpeedPlot	Parameter to generate a plot of link speeds: 0 = no link speed plot; 1 = MARPLE generates a plot of link speeds.	
ControlPlot	Parameter to generate a plot of green and cycle times: 0 = no plot with green and cycle times; 1 = MARPLE generates plot with green and cycle times.	
TravTimPlot	Parameter to generate a plot of route travel times: 0 = no travel time plot; 1 = MARPLE generates a plot of travel times of certain routes.	
ConvPlot	Parameter to generate convergence indicators at the end of the assignment 0 = no plot with convergence indicators; 1 = MARPLE generates plot with convergence indicators.	

Notes

- The plots are saved in the same directory as the output, with the names "*MFD plot.jpg*" and "*Speed contour plot for route part xx.jpg*".
- The contour plots are generated for the specified paths (route parts) in the network file, where, where *xx* is the number of the route.
- The input for the flow, speed, travel time and signal timings plots is specified in a separate input file '*MARPLE-Graphs.txt*'. This file has the following structure:

```
//FlowPlots
;nrLinks  LinkNrs
        2     1 11


//SpeedPlots
;nrLinks  LinkNrs
       3    3 10  12


//TravelTimePlots
;nrRoutes  RouteNrs
         2    16 17
```

```
//SignalPlots
;controller signal
        3       8
        3       9
        3      11
```

3.12 Output

Cluster	//OutputPar	
Parameter	**Meaning**	**Default**
OutputFlag	Parameter to determine the type of output: 0 = minimal output; 1 = normal output; 2 = normal output and selected link analysis; 3 = normal output and output for the Dutch National Strategic Model (LMS); 4 = normal output and selected link analysis and LMS output.	1
BinaryOutput	Parameter to save binary output in MATLAB format. This output is used to write into the OmniTRANS database. 0 = no binary output; 1 = binary output generation.	0
EmissionOutput	Parameter to calculate emission output based on the MARPLE output. 0 = no output on emissions; 1 = generating output on emissions.	0

Notes

- *Output flag*
  - *Flag* = 0. Minimal output is created: output per link, per OD relation, per specified path and network indicators.
  - *Flag* = 1. The default which gives the normal output, which is the same as above, but output per route is added.
  - *Flag* = 2. Some output is above, but with the results of a selected link analysis, which are stored in the file *SelectedLink.txt*.
  - *Flag* = 3. Again as above, but additionally with output similar to the output of the Dutch National Strategic Model System (LMS). This output is stored in the file *LMS-Output.txt*.
- *Binary Output*
  - *BinaryOutput* = 0. Output consists of text files only.
  - *BinaryOutput* = 1. Output consists of binary files in the MATLAB mat format, in addition to text files. If MARPLE is called with 2 additional arguments (the path to the OmniTRANS database and the name of the variant), the program reads the binary files and writes them to the OmniTRANS database.
- *Emission Output*
  - *EmissionOutput* = 0. No output on emissions.

- *EmissionOutput* = 1. In the normal output file, in addition to the traffic indicators on network level, emissions of CO, CO_2, HC, NO_x and PM are calculated and shown.
- For more information on output see chapter 5.

3.13 Emission parameters

Parameter	Meaning	Default
TruckPerc	Percentage of freight traffic for the calculation of emissions	10
ExcelOutput	Parameter to write the emission output into an Excel file: 0 = no excel output; 1 = generate excel file with emission output.	0
binaryOutput	Parameter to write the emission output in binary files: 0 = no binary output; 1 = generate binary files (MAT format) with emission output.	0

Notes

- These parameters are used for the calculation of emissions. Because MARPLE makes no distinction between cars and trucks in the simulation, the user can specify a truck percentage here to make the calculation of emissions a little bit more realistic.
- If *ExcelOutput* = 1, the detailed emission information is stored in an Excel file with the name of the normal file, extended with '*-PlusEmissions.xlsx*'.
- If *binaryOutput* = 1, the detailed emission information is stored in a MAT file with the name of the normal file, extended with '*-PlusEmissions.mat*'.

4 Network and OD-matrix

The network is specified in a text file. This file can have any given name, but if the OmniTRANS plugin is used, this file gets the name *marple_network.txt*. It not only defines the network (links and nodes), but also other aspects that are connected to the network, such as the OD matrix, the controls (VRIs, TDIs) and other traffic management measures (events). The following sections discuss the network file. First, a line of parameters or properties is displayed with a number of associated values and then these parameters or properties are explained.

4.1 Network parameters

In the network file, a number of extra simulation parameters are defined that are specific for the network at hand.

```
//Parameters
;nrTimePeriods LTimePeriod LTimeStep ScaleFlow ScaleCap ScaleSpeed DemandPar
             6        900         20      1.00     1.00      1.00         0
```

Parameter	Meaning
nrTimePeriods	Number of simulation periods
LTimePeriod	Time duration of one simulation period in seconds
LTimeStep	Time duration of a simulation step in seconds
ScaleFlow	Global scale factor for the OD matrix
ScaleCap	Global scale factor for road capacity
ScaleSpeed	Global scale factor for the speed with free flowing traffic
DemandPar	Parameter for the structure of the demand profile: 0 = jumpy and continuous during the period; 1 = continuously changing per time step.

Notes

- The parameter *LTimePeriod* must be divisible by *LTimeStep*.
- For short links, the time step (*LTimeStep*) is allowed to be longer than the shortest travel time (order 10-20 sec), but then the accuracy of the calculations decreases (see also section 2.10).
- The different options for the parameter *DemandPar* are explained in section 2.5.

4.2 Links

```
//Links
;linknr  nettype  length  nrlanes  satflow  speed  type  CTR  nrSG  Signal(s)
;                  (m)        (pea/hr) (km/hr)
    1        1     2713          2     4200      120     0
```

```
;linknr  nettype  length  nrlanes  satflow  speed  type  CTR  nrSG  Signal(s)
;                  (m)     (pea/hr) (km/hr)
    2        1     670        2     4200     120     0
    3        1    2045        2     4200     120     0
    4        1    1046        2     4200     120     0
    5        2     500        1     2000      80     0
    6        1    1654        1     2000      80     2    1     1          1
    7        2     500        1     2000      80     0
    8        1     715        1     2000      80     2    2     1          1
    9        2     702        1     2000      70     1    3     1         11
   11        2     999        2     3800      70     0
```

Parameter	Meaning
Linknr	NTmber to identify the link
Nettype	Number of the sub-network. This parameter can be used to differentiate the output at the sub-network level.
Length	Length of the link (m)
Nrlanes	Number of lanes on the link
Satflow	Lane capacity in (pae/hour)
Speed	Speed for free flow traffic (km/h)
Type	Link type 0 = normal link, free of control of something else; 1 = controlled link (signal control at the end of the link); 2 = metered link (ramp metering at the end of the link); 3 = roundabout link (roundabout at the end of the link); 4 = priority link (priority junction at the end of the link).
CTR	Controller number (for links with type = 1 or 2)
nrSG	Number of phases that control the link (for links with type = 1 or 2)
Signals	The numbers of the signal phases (for links with type = 1 or 2)
nrConf	Number of conflicting links (for links with type = 3 or 4)
ConfLinks	The numbers of the conflicting links (for links with type = 3 or 4)

Notes

- The link number is a number used for MARPLE. In the OmniTRANS plugin other numbers are used, because in MARPLE every direction has a separate link number and for OmniTRANS the link number is the same for both directions. The relationship between these two numbering systems can be found in the file *marple_conversion.txt*.
- The number of types for network type and their meaning can be freely defined. It can be considered as sub-networks. Network types are only used to group the simulation results at network level.
- The parameter *nrSG* can be larger than 1 if a link is controlled by several signal phases.

- The parameters *CTR*, *nrSG* and *Signals* are only needed for traffic signal control and ramp metering. The parameters *nrConf* and *ConfLinks* are only needed for roundabouts and priority junctions.

4.3 Nodes

```
//Nodes
;nodenr type nIn  links    nOut  links        AllowedTurns
    1    1   0               1    1
    2    0   1   1           2    2  9
    3    0   2   2  7        1    3
    4    0   2   3  5        1    4
    5    2   1   4           0
    6    4   1   8           1    7
    7    4   1   6           1    5
    8    3   3   9 13 14      3    6  8   12     0 1 0 0 1 0 1 0 1
   10    1   0               1   11
   11    2   1  12           0
   12    0   1  11           2   13 14
```

Parameter	Meaning
Nodenr	Identification number of the node
Type	Type of node: 0 = normal node to connect links; 1 = origin node; 2 = destination node; 3 = controlled node for which green times can be optimised (if parameter *Optimization* > 1, see section 3.1) 4 = node with ramp metering; 5 = controlled node with a fixed-time signal control plan; 6 = roundabout with 1 lane on the roundabout; 7 = roundabout with 2 lanes on the roundabout; 8 = priority junction.
nIn	Number of incoming links that connect to the node
Links	Link numbers of the incoming links (n_i)
nOut	Number of outgoing links that connect to the node
Links	Link numbers of outgoing links (m_j)
AllowedTurns	Option for turning movements: 0 = permitted movement; 1 = prohibited movement.

Notes

- The node number is a number used for MARPLE. In the OmniTRANS plugin other numbers are used. The relationship is put in the file *marple_conversion.txt*.
- *AllowedTurns*: from each inbound link to each outbound link, in the order n_1 to m_1 until m_j ; n_2 to m_1 until m_j ; ... ; n_i to m_1 until m_j. In the example movements 9-8, 13-8, 14-6 and 14-12 are prohibited.

4.4 Origins

```
//Origins
;nrOrigins        nodenrs
        2          1  10
```

Parameter	Meaning
nrOrigins	Number of origins
Nodenrs	Numbers of the nodes that are an origin (links are only leaving the node)

Notes

- The OmniTRANS plugin uses zones that represent both origins and destinations. In the conversion to MARPLE, zones are split into origins and destinations. The relation between the different numbers can be found in *marple_conversion.txt*.

4.5 Destinations

```
//Destinations
;nrDestinations  nodenrs
              2    5  11
```

Parameter	Meaning
nrDestinations	Number of destinations
Nodenrs	Numbers of nodes that are destinations (links are only entering the node)

Notes

- The OmniTRANS plugin uses zones that represent both origins and destinations. In the conversion to MARPLE, zones are split into origins and destinations. The relation between the different numbers can be found in *marple_conversion.txt*.

4.6 OD-matrix

```
//OD table
;origin destination  nRoutes  Routenrs  timeperiod 1  -  timeperiod n
     1           5         1    1        3100  3800  3500  3000  2000
     1          11         1    2         150   200   200   100   100
    10           5         2    3  4     1000  1500  1200  1000   600
```

Parameter	Meaning
Origin	Origin number (node number)
Destination	Destination number (node number)
nRoutes	Number of routes for the respective OD relationship
Routesnrs	Route numbers for the OD relationship

Parameter	Meaning
timeperiod 1 - n	Flow per time period (veh/hour), n = *nrTimePeriods*

Notes

- If *nRoutes* is 0 for all OD pairs and there are no route numbers given (*routenrs* is left open), the routes are generated and saved in the file *Routes.txt* (see section 2.4).

4.7 Distribution

```
//Distribution
;origin   destination   type   timeperiod 1   -   timeperiod n
  1          5           1       70    80   100   100   100   100
  1          5           2       10     5     0     0     0     0
  1          5           3       10     5     0     0     0     0
  1          5           4       10    10     0     0     0     0
 10          5           1      100   100   100   100   100   100
 10          5           2        0     0     0     0     0     0
 10          5           3        0     0     0     0     0     0
 10          5           4        0     0     0     0     0     0
```

Parameter	Meaning
Origin	Origin number (node number)
Destination	Destination (node number)
Type	Types of traffic: 1 = commuting; 2 = business traffic; 3 = other passenger traffic; 4 = freight traffic.
timeperiod 1 - n	Percentage of total demand of OD pair per time period (%), n = *nrTimePeriods*

Notes

- The distribution is only used for the application of MARPLE with the Dutch National Model or with tolling.
- If tolling is used (see section 3.9), the distribution is used to determine different route costs for the 4 different traffic types and the assignment takes these route costs into account, instead of only the travel time.

4.8 Routes

```
//Routes
;Routenr   nrLinksRoute          Links
        1              4          1    2    3    4
        2              3          1    9   12
        3              5         11   13    6    5    4
        4              6         11   14    8    7    3    4
```

Parameter	Meaning
Routenr	Route number
NrLinksRoute	Number of links included in the route
Links	Numbers of the links forming part of the route

Notes

- If routes are defined in the network file, these are used for the simulation. If not, it is checked whether the file *Routes.txt* exists and whether it can be used. The check is on the location and name of the network file and the values for the parameters *ThresFlow*, *nrRoutes*, *nrRand, scaleFac, junctionDelay* and *nrTimePeriods*. These should be consistent with the current values in the parameter and network file. If that is the case, the *Routes.txt* used. If not, new routes are generated using the Dijkstra algorithm. These routes are then used and saved in (a new) *Routes.txt*. An existing *Routes.txt* is copied to the file *RoutesOld.txt*.
- *Routes.txt* also contains the OD matrix and (if present) the distribution matrix. However, there is no check whether the OD matrix has changed! The total demand is compared, the sum of all OD cells, but not if there are changes from cell to cell. Therefore, if the OD matrix has been changed, the existing *Routes.txt* should be deleted, so that a new *Routes.txt* is created.
- For road works it could be convenient to include alternative routes in the network, which are not likely to be found with the normal procedure of generating routes. For this situation, the user can specify a separate file called *Detours.txt*. This file specifies the alternative routes. When reading the existing routes, it checks whether the file with diversions is newer. If so, new routes are generated, taking the detours into account. These are added to the set of routes per relevant OD pair. So for every detour the routes are checked for the specified combination of links. If this exists a new route is generated including the alternative set of links in that route. The file *Detours.txt* has the following format:

```
//Detours
 ;detournr    nrlinks    links
 ;alternative nrlinks    links
        1          5    45 46 128 236 23
                   7    38 123 22 345 346 347 569
```

4.9 Paths

```
//RouteParts
;RoutePnr nrLinksRouteP        Links
       10              5        1   9   8   7   3  4
       11              2        2   3
```

Parameter	Meaning
RoutePnr	Number of the path
NrLinksRouteP	Number of links that are part of the path
Links	The numbers of the links which are part of the path

Notes

- Paths can be used to define a part of a route or a path through the network. For each path, output is generated (see section 5.2.5).

4.10 Traffic signal control

```
//TrafSignals
;controller    signal     green      cycle     mingr      maxgr
;                         (sec)      (sec)     (sec)      (sec)
       3        11          14         97         7         40
       3         8          40         97         7         40
       3         9          33         97         7         40
```

Parameter	Meaning
Controller	Number of the traffic signal controller
Signal	Number of the signal control phase (or signal group number)
Green	Green time (s)
Cycle	Cycle time of the controller (s)
Mingr	Minimum green time (s)
Maxgr	Maximum green time (s)

```
//DynTrafSignals
;controller      signal  green time (sec)
;                        timeperiod 1 - timeperiod n
       3           11    14   14   14   14   14   14   14
       3            8    40   20   30   20   10   40   40
       3            9    33   33   33   33   33   33   33
```

Parameter	Meaning
Controller	Number of the traffic signal controller
Signal	Number of the signal control phase (or signal group number)

Parameter	Meaning
Green	Green time (s), number of values has to correspond with the number of time periods.

Notes

- The specification of //DynTrafSignals is not needed, but if is used, also //TrafSignals has to be specified. This is to determine the lost time per signal, which is then used to calculate the cycle time in case of dynamic green times.
- If dynamic green times are wanted, make sure that the node type of the signalled node is set to 3, otherwise the green times remain fixed.

4.11 Ramp metering

```
//RampMeters
;controller signal green cycle mingr maxgr uplink downlink percInc algCap
;                 (sec)  (sec)  (sec)   (sec)                  (%)
        1      1    12     12      2      12      2       3       3      4000
        2      1    12     12      2      12      3       4       2      3600
```

Parameter	Meaning
Controller	Number of the ramp metering controller
Signal	Number of the metered phase
Green	Green time (s)
Cycle	Cycle time of the controller (s)
Mingr	Minimum green time (s)
Maxgr	Maximum green time (s)
Uplink	Link number of the link on the main carriageway upstream of the metered on-ramp
Downlink	Link number of the link on the main carriageway downstream of the metered on-ramp
PercInc	percentage of capacity increase of the downstream link due to metering
AlgCap	capacity used by the metering algorithm to calculate the amount of traffic to be allowed to enter the motorway

```
//DynRampMeters
;controller  signal  green time (sec)
;                    timeperiod 1 - timeperiod n
        1      1      4    8    7    6    6    6    6
        9      1      5    6    6    9    6    6    6
```

Parameter	Meaning
Controller	Number of the ramp metering controller
Signal	Number of the metered phase

Parameter	Meaning
Green	Green time (s), number of values has to correspond with the number of time periods.

Notes

- The numbers of the traffic signal controllers and the ramp controllers must be unique.
- The signal number for ramp metering can be anything, because there are no conflicts.
- If *Metering* = 0 (see section 3.1) the green time will be equal to the cycle time and then all traffic is allowed to pass unhindered.
- If *percInc* and *algCap* are not specified, respectively the values 0 and the capacity of the downlink are used.
- The specification of //DynRampMeters is not needed, but if is used, also //RampMeters has to be specified. This is to determine the lost time per signal, which is then used to calculate the cycle time in case of dynamic green times.

4.12 Route information

```
//VMSinfo
;linknr  routeinfo  incident  deltaTeta
     1          1          0        1.0
     2          0          1        2.0
```

Parameter	Meaning
Linknr	Number of the link where route information is available
Routeinfo	Route information available in normal situations: 0 = no, 1 = yes
Incident	Route information available in case of incidents: 0 = no, 1 = yes
deltaTeta	Increase of level of information parameter θ (see section 4.13)

Notes

- For the links for which the *routeinfo* parameter is 1, this information is taken into account in the assignment. This only applies to those relations for which a route uses the link specified. The parameter for the level of information for those relations is increased by *deltaTeta*. This means that the assignment for these relations is different from that of the other OD pairs.
- Route information for incidents only is determined by the *incident* parameter. If this is set to 1 and *EventSimPar* = 2 (see section 3.8), then a new assignment is run for the OD pairs that have routes using the link with *linknr*.

4.13 User classes

```
//UserClasses
;userclass   percentage     teta    percentage2
```

```
1        10        0.0       80
2        70        1.0       10
3        20        3.0       10
```

Parameter	Meaning
Userclass	User classes for the assignment: 1 = habitual behaviour, user keep the shortest route based on free flow travel time; 2 .. 9 = users are moderately or well informed, dependent on the value of the θ parameter.
Percentage	Percentage of traffic belonging to a user class
Teta	Parameter for the level of information
Percentage2	Percentage of traffic belonging to a user class

Notes

- For *userclass* = 1 always *teta = 0*. The distribution over the routes is done according to the initial travel times.
- For *userclass* = 2 .. 9, users have knowledge about the travel times on their available routes, but uncertainty is small or high, dependent on the value of the parameter *teta*. *Teta* says something about the level of available information: the larger *teta*, the more information.
- The number of user classes can be expanded simply by adding lines to the table.
- The second column of percentages (*percentage2)* does not need to be specified. It is used in the additional simulation of events for incidents and WIU, if needed. These percentages can therefore be different from the normal ones. If these second percentages are not specified, the first ones are used.
- For the percentages the sum should be 100.

4.14 Time penalty user classes

```
//UserClassTP
;userclass   percentage
    1            0
    2            0
    3           10
```

Parameter	Meaning
Userclass	Number of the user class
Percentage	Percentage of extra travel time for that user class

Notes

- These parameters are used in combination with *//TimePenaltyPar* (see section 3.10). If *TimePenaltyType* > 0 extra travel time is added to links with a certain flow-capacity ratio or belonging to a certain sub-network for the user classes specified here. This will have the effect that a part of the traffic will choose other routes and there is less traffic for the specified links or sub-networks.

4.15 Time penalty network types

```
//NetTypePenalty
;nettype     timeperiod 1  … timeperiod n
     2            1  1  0  0  1  1
     4            0  1  1  1  0  0
```

Parameter	Meaning
Nettype	Number of the sub-network
Yimeperiod 1 - n	0 = no time penalty for that period; 1 = time penalty for that period; n = *nrTimePeriods*.

Notes

- The *nettype* is the number of the sub-network, which is given to every link (see section 4.2).

4.16 Initial traffic flows

```
//InitialFlows
;Route        timeperiod 1    -     timeperiod n
   1          3100  3800  3500  3000  2000  2000
   2           150   200   200   100   100   100
   3           500   750   600   500   300   150
   4           500   750   600   500   300   150
```

Parameter	Meaning
Route	Number of the route
Timeperiod 1 - n	Initial flow per time period (veh/h), n = *nrTimePeriods*

Notes

- If initial route flows are defined here, they are used for the simulation. If not, the initial assignment is calculated based on distance or free travel time. This can be set in the *initialAssign* parameter of the general parameters (see section 3.2).

4.17 Events

```
//Events
;begintime    endtime    linknr     nrlanes      satflow     vfree     type
;                                                  (%)        (%)
   1200         1500       3          -1          47.6        50       1
```

Parameter	Meaning
Begintime	Start time of the event in seconds after the start of the simulation
Endtime	End time of the event in seconds after the start of the simulation
Linknr	Number of the link to which the event relates
Nrlanes	Change in the number of lanes
Satflow	Link capacity (percentage of normal capacity)
Vfree	Speed during free flow of traffic (percentage of normal speed)
Type	Type of the event: 1 = measure 2 = incident 3 = road works

Notes

- These parameters can be used to define events that are able to change network properties during the simulation.
- In the OmniTRANS plugin, events are used to model traffic management measures like peak-hour lanes.
- For event type 1, the events are used in the normal assignment and the results of that assignment include the impact of those events.
- At this moment no distinction is made between event types 2 and 3.
- For events with types 2 and 3, MARPLE follows a special procedure to get useful results. First an equilibrium assignment is run, without those events with types 2 and 3, but with event type 1 in the simulation. Then a new simulation or assignment is started with all events and with the last route flows from the first equilibrium assignment. The user can specify how many iterations are run in the second assignment (see section 3.8). For incidents, this could be one iteration and for road works several, depending on the duration of the work. A different distribution of the different user classes can also be specified (see section 4.13). This is to enable different route choice behaviour.
- The method which is used to calculate the delay with events, can be set with the variable *DelayType* of the parameters (see section 3.1).

4.18 Links for selected link analysis

```
//SelectedLinks
;sellinks  percChange
        3          5.0
       11          0.0
```

Parameter	Meaning
Sellinks	The link numbers for which a selected link analysis is made
percChange	The percentage by which the traffic demand of all OD relations using this link is adjusted.

Notes

- With this, a selection of links can be defined for which a selected link analysis is performed. For this, the *OutputFlag* must be set to 2 or 4.
- If these parameters are not specified and the *OutputFlag* is 2 or 4, a selected link analysis is made for all links. Therefore, using these parameters can save a lot of calculation time.
- If the *percChange* parameter of a selected link is not equal to 0, a selected link analysis is carried out for that link before the simulation/assignment starts. For the OD relations using this link, the traffic demand is adjusted with the given percentage (both negative and positive). This allows the user to quickly adjust the flow on a road section without having to apply an elaborate estimation method for the OD matrix. For example, if you know the measured flows and the simulation gives different results, this feature can be used the calibrate the model. Of course, this feature should be used thoughtfully.

4.19 Node coordinates

```
//NodeCoordinates
;nodenr      x-coord     y-coord
    1            0           0
    2            6           0
    3           12           0
    4           20           0
    5           26           0
    6         10.5          -2
    7         18.5          -2
    8            9          -6
   10            3          -6
   11            9          -9
```

Parameter	Meaning
Nodenr	Number of the node
x-coord	x-coordinate
y-coord	y-coordinate

Notes

- The coordinates can be used for network visualisation. OmniTRANS generates these coordinates, but also user specified ones can be used. The coordinates can then be read with other software packages such as GIS, to visualise the network.

5 Output

5.1 Introduction

Depending on the *OutputFlag* parameter (see section 3.12), one or several text files are generated with results of the MARPLE simulation. The parameter *OutputFlag* can have the following values:
0 = minimal output;
1 = normal output;
2 = normal output and selected link analysis;
3 = normal output and output for the Dutch National Model (LMS);
4 = normal output, selected link analysis and output for the National Model (LMS).

The name for the file with the minimal or normal output is free to choose if you use MARPLE as a standalone program. We use *Output.txt* here. The MARPLE plugin for OmniTRANS uses the name *marple_result.txt*. All the other files have predefined names. The following files are generated for the different values:

OutputFlag	Output.txt	SelectedLink.txt	LMS-Output.txt
0	x		
1	x		
2	x	x	
3	x		X
4	x	x	X

The following sections discuss the contents of the various output files.

5.2 Results file

In the results file *Output.txt* most of the output is written. It concerns the results of the simulation and the assignment, mostly of the final iteration. The file is divided into a number of sections, which are described below.

5.2.1 General

The general section mentions the files for which the simulation was run: simulation parameters file, network file and output file. Also, the values of all the parameters used are given, both the simulation as the network parameters.

5.2.2 Intermediate results per iteration

If the parameter *OutputFlag* is larger than or equal to 1, for each iteration the following results are given:
- Total kilometres travelled in the network (veh.km);
- The total time spent in the network (veh.hour);
- The average speed in the network (km/hr);
- The total delay in the network (veh.hour).

5.2.3 Results per link

After the first and final iteration, the results are written per link for each period (//LinkOutput):

- Density (veh/km);
- Speed (km/h);
- Standard deviation of the speed within the time period (km/h)
- Inflow and outflow (veh/hour);
- Queue length (m).

5.2.4 Results per route

If the parameter *OutputFlag* is larger than or equal to 1, the results for each route are given for each time period:

- Flow (veh/hour);
- Travel time (min);
- Delay (min);
- Speed (km/h).

5.2.5 Results per path (route part)

For each path (route part) the following results are given:

- Flow (veh/hour);
- Travel time (min);
- Delay (min);
- Speed (km/h).

5.2.6 Results for controllers

For the traffic signal controllers the green times and cycle times for each signal phase and each controller are written after the final iteration.

5.2.7 Output for sub-networks

For each sub-network and time period, the following results are written in separate tables:

- The total number of kilometres travelled in the network (veh.km);
- The total time spent in the network (veh.hour);
- The average speed in the network (km/hr);
- Total tolls (if specified in the input);
- The total delay in the network including zone delay (veh.hour).

5.2.8 Delay in zones

It may happen that traffic cannot enter the network due to blocking back effects, and gets stuck in a zone. To show this, an extra table is included that gives the entrance delay per zone and per time period. If this delay only increases over time, it is an indication that the traffic cannot enter the network. The delay at the entrance is then also shown per subnetwork and per time period, so that it is clear for which subnetwork this problem is the largest.

5.2.9 Number of vehicles

The number of vehicles in the network is an important figure. For each time period and in total, a table shows the traffic demand, how many vehicles are leaving and how many are arriving. This indicates the size of the traffic flows and also shows whether there are any problems with departures or arrivals. If many more vehicles are leaving than arriving, they may get stuck somewhere along the way. This can

also be seen in a second table which shows per time period (and at the start and end of the simulation) how many vehicles are in the network.

5.2.10 Network output

The network results are summarised at the end of the output file. This table gives results for the whole network and for each sub-network for the final iteration:

- Total number of vehicles (demand);
- The total number of kilometres travelled in the network (veh.km);
- The total time spent in the network (veh.hour);
- The total delay in the network including zone delay (veh.hour);
- The average speed in the network (km/hr).

Extra tables are included for the sum and average over the iterations for the following network indicators:

- The total number of kilometres travelled in the network (veh.km);
- The total time spent in the network (veh.hour);
- The total delay in the network including zone delay (veh.hour).

5.2.11 Emission output for the network

If the parameter *EmissionOutput* is equal to 1, the following results for emissions are given for the whole network and for each subnetwork:

- Total emissions CO (gr);
- Total emissions CO_2 (kg);
- Total emissions HC (gr)
- Total emissions NO_x (gr);
- Total emissions PM (gr).

5.3 ZoneDemand.txt

In this file, the total demand per period per origin and destination is written. This information can help determine which zones cannot handle the demand due to the capacity of the connected links. For these zones, the traffic cannot enter or exit the network sufficiently, causing congestion in the zone or near the zone, which is not desirable. The file also gives the links for which the demand is probably higher than the capacity for certain time periods.

5.4 ResultingCapcity.txt

In this file, the resulting capacity per link per period is written. The capacity of a link is input, but can change during the simulation due to controllers or events. This file shows the original capacity and the capacity which is used in the simulation or final iteration of the assignment.

5.5 SelectedLink.txt

This file records, per link and per period, which OD pairs from the matrix use the relevant link (in the relevant period). It also shows the flows of the relevant OD pairs on the relevant link (in the relevant period). The file format is as follows:

//SelectedLinkOutput

Linknr	Origin	Destination	1	2	3
1	1	5	3100.0	3800.0	2422.4
1	1	1	150.0	200.0	138.4
3	1	5	3094.2	1867.2	1801.6
3	10	5	256.8	232.8	298.4
4	1	5	3084.1	2085.9	2178.0
4	10	5	994.9	823.4	746.7
11	10	5	1000.0	1500.0	1200.0

5.6 SelectedLinkExtended.txt

In this file, the results of a selected link analysis are written in a tree structure. For each selected link, the amount of traffic per link and per period is shown. In a network, it is easier to visualise where the traffic of a certain link comes from and goes to. The layout of the file is as follows:

//ExtendedSelectedLinkOutput

SelLinknr	Linknr	1	2	3
1	1	3250.00	4000.00	2555.85
1	2	3100.00	3800.00	2417.70
1	3	3100.00	3800.00	2417.70
1	4	3100.00	3800.00	2417.70
1	9	150.00	200.00	138.15
1	12	150.00	200.00	138.15
3	1	3033.4	1902.16	1840.61

5.7 LMSOutput.txt

In this file, the results of the simulation are written in the (slightly modified) format of the Dutch National Model (LMS):

- Period (between 1 and *nrTimePeriods*);
- Number of the start node (start link);
- Number of the end node (end link);
- Link length (m);
- Free speed link (dm/h);
- Number of lanes;
- Link capacity (pae/hour);
- Network type (sub-network);
- Road type;
- Special purpose lane;
- Free travel time (sec)
- Type of toll points morning rush hour;
- Type of toll points evening rush hour;
- Type of toll points rest of the day;
- Total flow in period (veh/h);

- Commuting flow (veh/hour);
- Business traffic flow (veh/hour);
- Other passenger traffic flow (veh/hour);
- Freight traffic flow (veh/hour).

5.8 MARPLE log file

A separate log file is created for each run. All screen output is stored in this file, so that after each run it is saved for possible analysis. The name of this log file is *MARPLE-log-yyyymmdd-hhmmss.txt*. Where *yyyymmdd* stands for the year, month and day of the run and *hhmmss* stands for the time the run was started.

5.9 Plots

MARPLE can generate different types of plots if desired: the MFD plot, the speed contour plot, plots of link information and control information and the travel time plot. MFD stands for Macroscopic Fundamental Diagram and represents the relationship between the number of vehicles in the network and the output of the network. Both variables can be represented in different ways. In MARPLE, the number of vehicles in the network and the number of vehicles leaving the network are plotted for the specified time period.

The speed contour plots show the speed for each path (route part) as a relation of space (along the path) and time. With these plots, the location and size of traffic jams can easily be determined visually.

The link plots provide information on flow speed and queue length during the simulation period for the specified links (in the file *MARPLE-Graphs.txt*, section 3.11).

The plots with signal control information show the development of green times and cycle times of the specified signal groups (also in the file *MARPLE-Graphs.txt*, section 3.11).

If an assignment is run a plot is generated with information on the total delay in the network and the convergence error per iteration.

5.10 Binary output files

If the parameter *BinaryOutput* of the output parameters (see section 3.12) is set to 1, MARPLE also generates binary output files. This is to speed up the reading in the OmniTRANS database. The files are put in the same directory as the output file.

The binary output files are:
- FreeFlowTT.mat;
- LinkData.mat;
- FinalRouteFlows.mat;
- RouteData.mat;

- RoutePartData.mat;
- NetworkData.mat;

These files contain the same information as the normal output file of MARPLE.

Another binary file is generated if emissions have been calculated and this binary output file is requested. This file has the same name as the output file with an addition, so *Output.txt* becomes *Output_PlusEmissions.mat* or if an Excel file was requested an *Output_PlusEmissions.xlsx* file is created.

To be able to plot the requested graphs, also simulation data is stored in binary files. That is the case for:

- ControlComplete.mat;
- FlowComplete.mat;
- MFDComplete.mat;
- SpeedComplete.mat;
- TrTime.mat.

5.11 JSON output files

For future purposes output is also generated in JSON format. This includes link data in the file *LinkData.json* and the network indicators in the file *NetworkData.json*. These files are also generated for the extra simulations, as explained in the next paragraph.

5.12 Control output and input file

If signal control is specified with external files (*LControlMethod* = 99), every control period (with length specified with *LContrPeriod*) the file *LinkOutput.csv* is generated containing for every link a row with the following data: link number, length of the link, inflow (veh/hr), outflow (veh/hr) speed, density and queue length (separated by semi-colons).

The simulation expects the file *inputControls.csv* with a row for every combination of controller and signal (see sections 4.2, 4.10 and 4.11, containing the following data: controller number, signal number, green time (sec) and cycle time (sec), again separated by semi-colons.

5.13 Event output and input file

If external events are made possible (*EventExternal* = 11), every event period (with length specified with *EventPeriod*) the file *LinkOutput.csv* is generated containing for every link a row with the following data: link number, length of the link, inflow (veh/hr), outflow (veh/hr) speed, density and queue length (separated by semi-colons).

The simulation expects the file *inputEvents.csv* with a row for every link for which the characteristics are changed, containing the following data: link number, number of lanes, capacity and maximum speed, again separated by semi-colons.

5.14 Extra output files

For the simulation with events (without or with route information) and with time penalties, extra output files are generated. For every value of the parameters *EventSimType* and *TimePenaltyType* this is shown in the following table. Note that not both parameters can be larger than 0.

For the table we assume that we named the output file *Output.txt* and for the binary files we take *LinkData* as an example, but the same holds for other binary output files mentioned in section 5.10, with the exception of *FreeFlowTT.mat* and the binary files related to graphs.

EventSimType	**Text output file names**	**Binary output file names**
0	Output.txt	LinkData.mat
1	Output.txt	LinkData.mat
	Output-Incident.txt	LinkData-Incident.mat
2	Output.txt	LinkData.mat
	Output-Incident.txt	LinkData-Incident.mat
	Output-Incident-VMS.txt	LinkData-Incident-VMS.mat
3	Output.txt	LinkData.mat
	Output-VMS.txt	LinkData-VMS.mat
TimePenaltyType	**Text output file names**	**Binary output file names**
0	Output.txt	LinkData.mat
1	Output.txt	LinkData.mat
	Output-Penalty.txt	LinkData-Penalty.mat
2	Output.txt	LinkData.mat
	Output-Penalty.txt	LinkData-Penalty.mat