Lastenheft SUMO-TAPAS-Kopplung

|  |  |
| --- | --- |
| Institut | Institut für Verkehrsforschung / Institut für Verkehrssystemtechnik |
| Erstellt von | Michael Behrisch |
| Beteiligte | Jakob Erdmann, Matthias Heinrichs, Anke Sauerländer-Biebl |
| Version 0.1 28.04.2014 | Aufschrieb vom ersten Treffen (MBeh) |
| Version 0.2 05.05.2014 | Zweites Treffen, Ergänzung der Datenbanktabellen (MBeh, ASB) |
| Version 0.3 06.05.2014 | Drittes Treffen (MBeh) |
| Version 0.4 12.06.2014 | Beta-Version, Überarbeitung der Struktur, Ausarbeitung der Stichpunkte, einige Präzisierungen (Tabellennamen, Fehlermaße) fehlen noch (MBeh) |
| Version 0.5 23.06.2014 | Einarbeitung der Anmerkungen (MBeh) |
| Version 0.6 30.06.2014 | Ergänzung Tests, Ausformulierung einiger Aufgaben (MBeh), Überarbeitung der Abbildungen (JErd) |
| Version 0.7  30.06.2014 | Ergänzung der Parameter-Schlüssel und der Bewertungsformel für die Messgüte (MHei) |
| Version 0.8  14.10.2014 | Korrekturen bei der DB-Definition (MHei) |
| Version 0.9 06.01.2015 | Anmerkungen zur Umsetzung (MBeh) |
| Version 0.10 19.01.2015 | Anmerkung und Korrektur zur Umsetzung (MHei) |
| Version 0.11 02.02.2015 | Ergänzung/Überarbeitung der Schnittstellendokumentation SUMO <-> Lärm, s. Abschnitt 6 (JErd) |
| Version 0.12 04.02.2015 | Ergänzung Emissionsklassenmapping, Änderungen timeline (MBeh) |
| Version 0.13 28.04.2015 | Änderungen TAZ (MBeh) |
| Version 0.14 10.06.2015 | Streichung beliebiger SUMO-Parameter in Parameter-Tabelle, Präzisierung Konfiguration der Umlegung / Postprocessing durch Python-Skripte, Präzisierung vType vs. vClass (MBeh) |
| Version 0.15 15.06.2015 | Anpassungen nach Statustreffen insbesondere zu den Zeitscheiben der OD-Matrizen (MBeh) |
| Version 1.0 25.09.2015 | Feinschliff vor Beendigung der technischen Implementation (MBeh) |
| Version 1.1 23.06.2016 | Ergänzung verschiedener Moduskombinationen in In- und Output (MBeh, MHei) |
| Version 1.2 22.08.2016 | Ergänzung Status „pending“ (MBeh, MHei) |
| Version 1.3 11.01.2017 | Anpassung der Tabelle für den Grundlastverkehr (MBeh) |
| Version 1.4 26.04.2018 |  |
| Version 1.5 04.06.2021 | Präzisierung der Ausgabeformate für Trips und OD-Matrix bei Benutzung von ÖPNV (MBeh, MHei) |

Inhalt

[1 Einleitung 5](#_Toc454443782)

[2 Technische Kopplung 5](#_Toc454443783)

[2.1 Konfiguration (Sumo Cfg und Fahrzeugtypen) 6](#_Toc454443784)

[2.1.1 Modellierung von Einschränkungen (Restrictions) 6](#_Toc454443785)

[2.2 Prozesskommunikation TAPAS -> SUMO (Input) 7](#_Toc454443786)

[2.2.1 Tabelle TAZ (param\_key: DB\_TABLE\_TAZ) 7](#_Toc454443787)

[2.2.2 Tabelle Cars (param\_key: DB\_TABLE\_CARS) 7](#_Toc454443788)

[2.2.3 Tabelle Trips (param\_key: DB\_TABLE\_TRIPS) 7](#_Toc454443789)

[2.2.4 Tabelle Representatives (param\_key: DB\_TABLE\_REPRESENTATIVES) 8](#_Toc454443790)

[2.2.5 Tabelle OD-Matrix Grundlastverkehr / Wirtschaftsverkehr (param\_key: DB\_TABLE\_ADDITIONAL\_TRAFFIC) 8](#_Toc454443791)

[2.2.6 Tabelle Timeline (core.global\_timelines) 9](#_Toc454443792)

[2.3 Prozesskommunikation SUMO -> TAPAS (Output) 9](#_Toc454443793)

[2.3.1 Tabellen ODMatrix, ODEntry (param\_key: DB\_TABLE\_SUMO\_OD\_OUTPUT DB\_TABLE\_SUMO\_OD\_ENTRY) 9](#_Toc454443794)

[2.3.2 Tabelle Trips (param\_key: DB\_TABLE\_SUMO\_TRIP\_OUTPUT) 10](#_Toc454443795)

[2.4 Prozesssteuerung 10](#_Toc454443796)

[2.4.1 Prinzipieller Ablauf 10](#_Toc454443797)

[2.4.2 Statustabelle (param\_key: DB\_TABLE\_SUMO\_STATUS) 11](#_Toc454443798)

[3 Netzgrundlage 11](#_Toc454443799)

[4 Szenarien 12](#_Toc454443800)

[4.1 Netzdatei (extrahiertes GDF (Elmar-Format 6.1) sumo xml + shapefile für taz) 12](#_Toc454443801)

[4.2 Fahrzeugtypen (sumo xml) 12](#_Toc454443802)

[4.3 Umlegungsskript 12](#_Toc454443803)

[4.4 Kantenlisten für Einschränkungen 13](#_Toc454443804)

[4.5 Postprocessingskript 13](#_Toc454443805)

[4.6 Parameter 13](#_Toc454443806)

[5 Validierung 13](#_Toc454443807)

[5.1 Zählungen (Messschleifen) 14](#_Toc454443808)

[5.2 Geschwindigkeiten (FCD) 14](#_Toc454443809)

[5.3 Reiseweiten und Reisezeiten (MID) 14](#_Toc454443810)

[5.4 Validierung der Repräsentanten 15](#_Toc454443811)

[6 Schnittstelle zu AS (Lärm) 15](#_Toc454443812)

[7 Nicht funktionale Anforderungen 15](#_Toc454443813)

[8 Software Engineering / Tests 16](#_Toc454443814)

[9 Emissionen 16](#_Toc454443815)

[9.1 HBEFA2 7 Cluster 16](#_Toc454443816)

[9.2 HBEFA2 14 Cluster 16](#_Toc454443817)

[9.3 HBEFA3 16](#_Toc454443818)

[9.4 PHEMlight 17](#_Toc454443819)

[9.5 Aggregation 17](#_Toc454443820)

# Einleitung

Inhalt dieses Dokuments ist die Spezifikation der Schnittstellen sowie der nicht funktionalen Anforderungen zur Kopplung von SUMO und TAPAS im Rahmen von VEU II sowie der zu generierenden Ausgaben für eine Lärmmodellierung. Ziel ist die Überführung der existierenden prototypischen Kopplung in ein permanent lauffähiges automatisiertes System, das auch einfach neue Szenarien ermöglicht. Der Austausch erfolgt durch Zugriffe auf eine Postgres-Datenbank und Dateien im lokalen Dateisystem.

Alle Interfacedefinitionen bestimmen das Minimum an Daten, das zur Verfügung stehen soll. D.h. insbesondere werden alle Spalten in Datenbanktabellen sowie Attribute in XML-Dateien immer über ihre Namen und nie über ihre Positionen referenziert. Die erstellten Tabellen / Dateien können über dieses Minimum hinaus weitere Spalten / Attribute enthalten. Identifizierte Aufgaben sind im Dokument gelb hinterlegt.

Das folgende Bild zeigt kurz den Gesamtsystemaufbau.



Dieses Dokument beschreibt zuerst die technischen Voraussetzungen der Kopplung zwischen SUMO und TAPAS in Form einer Dokumentation der nötigen Tabellen- und Dateiformate sowie des Steuerungsprozesses und beschreibt anschließend die grobe Struktur von Szenarien und der verwendeten Netze. Die zu Grunde liegenden Versionen sind SUMO 0.21.0 und TAPAS 0.9.3

# Technische Kopplung

Im Folgenden werden die Datei- und Tabellenformate für die Kopplung von TAPAS und SUMO beschrieben. Nach der allgemeinen Konfiguration werden beide Richtungen der Kommunikation einzeln dargestellt. Wenn von Input geredet wird, ist die Kommunikation von TAPAS nach SUMO gemeint, Output ist die Rückrichtung. Für das Erstellen der Tabellen und der entsprechenden Konfigurationseinträge (siehe 2.1) ist in der Regel der Datenersteller verantwortlich, d.h. für den Input VF für den Output TS.

Negative Werte bei Reisezeiten/-weiten stehen für nicht existente Verbindungen.

## Konfiguration (Sumo Cfg und Fahrzeugtypen)

Die Konfiguration steht als Freitext in der Datenbanktabelle public.simulation\_parameters:

|  |  |  |
| --- | --- | --- |
| **Spalte** | **Typ** | **Beschreibung** |
| sim\_key | Varchar | Referenz auf Simulation |
| param\_key | Varchar | Schlüsselname |
| param\_value | Varchar | Schlüsselwert |

Welche Dateien genutzt werden, wird nur über ein Basisverzeichnis konfiguriert (siehe Kapitel zur Szenariendefinition), der entsprechende param\_key ist SUMO\_TEMPLATE\_FOLDER. Mehrere Simulationen können ein Basisverzeichnis nutzen (aber nur nacheinander). Die zu simulierenden Zeitscheiben stehen als Parameterarray in einer Tabelle die unter dem param\_key DB\_TABLE\_MATRIXMAPS zu finden ist. Dort liegt unter dem in DB\_NAME\_MATRIX\_TT\_MIT\_BASE definierten Schlüssel (matrixMap\_name) ein Array von Fließkommazahlen (matrixMap\_distribution), der die Endstunden beschreibt. Als maximale Auflösung wird eine halbe Stunde angenommen.

Der Parameter DB\_HH\_SAMPLE\_SIZE gibt an, welcher Anteil der Bevölkerung in TAPAS simuliert wurde (Wert zwischen 0 und 1). Die Hochskalierung erfolgt in SUMO durch das Duplizieren der entsprechenden Fahrzeuge, aber es werden nur die „echten“ Trips zurückgeschrieben.

Fahrzeugtypen werden in SUMO als XML definiert (s. <http://sumo.dlr.de/wiki/Definition_of_Vehicles,_Vehicle_Types,_and_Routes#Vehicle_Types>) und über ihre ID aus der Triptabelle s.u. referenziert.

Die Namen der meisten Ein- und Ausgabetabellen sind für das jeweilige Szenario spezifisch und stehen in der Parametertabelle, die entsprechenden Schlüssel (param\_key) stehen in den jeweiligen nächsten Abschnitten.

### Modellierung von Einschränkungen (Restrictions)

Für Standardeinschränkungen (Verbot bestimmter Fahrzeugtypen auf einzelnen Straßen) kommen die in SUMO definierten vClass zum Einsatz, siehe http://sumo.dlr.de/wiki/Definition\_of\_Vehicles,\_Vehicle\_Types,\_and\_Routes#Abstract\_Vehicle\_Class. Sind weitergehende Einschränkungen nötig, um Umweltzonen / MAUT etc. abzubilden, werden die existierenden Fahrzeugtypen mit einer anderen vClass dupliziert und die Netze entsprechend beim Aufbau des Szenarios angepasst (siehe Szenarien). Momentan sieht SUMO nur zwei „custom“ vClasses zu diesem Zweck vor, diese Zahl ist bei Bedarf zu erhöhen oder es sind weitere vClasses wie „vip“ zu nutzen die im zu Grunde liegenden Netz keine Verwendung finden. Die entsprechend duplizierten Typen werden statisch mit den geänderten vClasses definiert und sind Teil des Basisszenarios. Auf die entsprechenden vClasses sollte auch bei Definition der Netzparameter Bezug genommen werden (siehe Szenarien).

Eine sehr einfache Typendatei könnte wie folgt aussehen:

<routes>

<vType id="PKW" length="5" maxSpeed="70" vClass=”vip”/>

<vType id="PKW\_restricted" length="5" maxSpeed="70" vClass=”passenger”/>

</routes>

Für die Umlegung werden nur wenige Fahrzeugtypen genutzt, um die Komplexität der OD-Matrix gering zu halten. Detaillierte Fahrzeugtypen (inklusive Emissionsklassen) können jedoch im Nachgang (bzw. im Postprocessingschritt) wieder aus den TAPAS-Eingangsdaten rekonstruiert werden, da jede Einzelfahrt (von TAPAS generierter Trip) identifizierbar bleibt und damit ein Fahrzeug zugeordnet werden kann. Dies gilt nicht für die Repräsentanten zum Auffüllen leerer TAZ-Beziehungen und den Grundlastverkehr. Entsprechende Änderungen am Grundlastverkehr können nur als Änderungsraten an die entsprechenden Modelle zurückgegeben werden.

## Prozesskommunikation TAPAS -> SUMO (Input)

### Tabelle TAZ (param\_key: DB\_TABLE\_TAZ)

|  |  |  |
| --- | --- | --- |
| **Spalte** | **Typ** | **Beschreibung** |
| taz\_id | Int | Eindeutige TAZ-ID |
| taz\_num\_id | Int | Eindeutige TAZ-ID in der Datei |

Die Tabelle liegt im core Schema. Ihre Inhalte werden bisher nicht benötigt. In allen Referenzen in der Datenbank wird die Spalte taz\_id genutzt, zur Referenz auf die TAZ in Dateien wird die taz\_num\_id genutzt. Es muss sichergestellt werden, dass die Referenzen auf die IDs immer gültig sind. Die Geometrien der TAZ sind nicht in dieser Tabelle enthalten sondern werden separat per Datei geliefert.

### Tabelle Cars (param\_key: DB\_TABLE\_CARS)

|  |  |  |
| --- | --- | --- |
| **Spalte** | **Typ** | **Beschreibung** |
| car\_id | Int | Eindeutige Fahrzeug-ID |
| kba\_no | Int | Größenklasse des Fahrzeugs analog KBA |
| engine\_type | Int | Kraftstofftyp (0-Benzin, 1-Diesel, 2-Gas, 3-BEV (Batterie), 4-PHEV (Plugin-Hybrid)) |
| car\_key | Varchar | Flotten-ID |
| emmision\_type | Int | Emissionsklasse (Euro-Norm) |

Die Tabelle liegt im core Schema. Die Tabelle beschreibt die verwendeten Fahrzeuge insbesondere hinsichtlich ihrer Emissionseigenschaften, die car\_id und der car\_key bilden gemeinsam den Primärschlüssel. Die zu verwendende Flotte steht in simulation\_parameters unter DB\_CAR\_FLEET\_KEY.

Die Abbildung von den obigen Parametern auf SUMO-Emissionsklassen erfolgt wie im Kapitel Emissionen beschrieben.

### Tabelle Trips (param\_key: DB\_TABLE\_TRIPS)

|  |  |  |
| --- | --- | --- |
| **Spalte** | **Typ** | **Beschreibung** |
| p\_id | Int | Eindeutige Personen-ID |
| hh\_id | Int | Eindeutige Haushalts-ID |
| start\_time\_min | Int | Losfahrzeit minutengenau |
| lon\_start | Double | WGS84 Grad |
| lat\_start | Double | WGS84 Grad |
| taz\_id\_start | Id | Referenz auf die TAZ-Tabelle |
| lon\_end | Double | WGS84 Grad |
| lat\_end | Double | WGS84 Grad |
| taz\_id\_end | Id | Referenz auf die TAZ-Tabelle |
| sumo\_type | Varchar | Referenz auf die Typliste |
| activity\_duration\_min | Int | Dauer der Aktivität am Ziel in Minuten |
| mode | Int | Modus |
| is\_restricted | Bool | Zugangsbeschränkung aktiv |
| car\_type | Int | Referenz auf die Cars-Tabelle |
| usable\_modes | Array von Enumvalues | Liste von Werten aus (“walk”, “bike”, “car“, “passenger“, “public transport”, “taxi”, “train”, “car sharing”, “bike sharing” , “ride sharing”) |

Die Tabelle liegt im public Schema und der Name enthält neben dem Eintrag aus DB\_TABLE\_TRIPS noch den sim\_key als Suffix (z.B. "berlin\_trips\_2014y\_12m\_19d\_10h\_21m\_21s\_735ms"). Die Tabelle beschreibt die Trips aus TAPAS, die p\_id, hh\_id, start\_time\_min bilden gemeinsam den Primärschlüssel. Start- und Zielort werden bei Erstellung der Eingabedaten für die Simulation durch TS sollten möglichst genau aufgrund der Geokoordinaten auf das Netz gemappt (Tool bereits vorhanden). Die Startzeit darf in einem Fünf-Minuten-Intervall variiert werden, es muss jedoch die Referenz auf die Originalstartzeit erhalten bleiben (SUMO erlaubt sekundengenaue Angaben).

Die Abbildung von Wegeketten ist nötig, das Fahrzeug kann aber unterwegs wechseln. Dazu wird von TS ein Tool entwickelt, welches Personen mit Plänen entsprechend der Wegekettenmodellierung in SUMO erzeugt. Die Activity\_duration muss zwingend eingehalten werden. Relevante Modi werden je Szenario definiert (param\_key: SUMO\_MODES) (bisher werden Taxi und „normale PKW“ herausgefiltert; entsprechen den Modus-IDs: 2(MiV) und 4(TAXI)). Die Festlegung des in SUMO zu nutzenden Fahrzeugtyps erfolgt immer über den sumo\_type-Eintrag, der car\_type hat lediglich informativen Charakter (bzw. dient zur initialen Erstellung der möglichen SUMO Typen für das Postprocessing, siehe Cars-Tabelle).

### Tabelle Representatives (param\_key: DB\_TABLE\_REPRESENTATIVES)

|  |  |  |
| --- | --- | --- |
| **Spalte** | **Typ** | **Beschreibung** |
| Id | Int | Household oder location id |
| taz\_id | Int | Referenz auf die TAZ-Tabelle |
| representative\_coordinate | geometry | Punktposition des Repräsentanten |

Die Tabelle liegt im core Schema. Diese Tabelle dient dazu, Repräsentanten für einzelne Traffic Analysis Zones (auch Traffic Assignment Zones, TAZ) festzulegen, falls in den Trips nicht genügend echte Start- oder Zielorte zur Verfügung stehen, um aussagekräftige Mittelwerte für Reisezeiten und -weiten zu bestimmen (siehe Abschnitt 2.3.1). Die Repräsentanten werden von VF festgelegt (angestrebt: maximal 5 pro TAZ in Abhängigkeit von der Varianz, siehe Validierungskapitel). Die Auswahl, welche Repräsentanten-paare gerechnet werden, um eine ausreichende Datenbasis sicherzustellen, erfolgt auf TS-Seite. Der Primärschlüssel umfasst nur das id Attribut, die zur besseren Identifizierung bei -1 000 000 000 anfangen und absteigend gezählt werden. Das Mapping auf das Netz erfolgt wie in den Trips.

### Tabelle OD-Matrix Grundlastverkehr / Wirtschaftsverkehr (param\_key: DB\_TABLE\_ADDITIONAL\_TRAFFIC)

|  |  |  |
| --- | --- | --- |
| **Spalte** | **Typ** | **Beschreibung** |
| p\_id | Int | Eindeutige Personen-ID |
| hh\_id | Int | Eindeutige Haushalts-ID |
| start\_time\_min | Int | Losfahrzeit minutengenau |
| lon\_start | Double | WGS84 Grad |
| lat\_start | Double | WGS84 Grad |
| taz\_id\_start | Id | Referenz auf die TAZ-Tabelle |
| lon\_end | Double | WGS84 Grad |
| lat\_end | Double | WGS84 Grad |
| taz\_id\_end | Id | Referenz auf die TAZ-Tabelle |
| sumo\_type | Varchar | Referenz auf die Typliste |
| activity\_duration\_min | Int | Dauer der Aktivität am Ziel in Minuten |
| mode | Int | Modus |
| is\_restricted | Bool | Zugangsbeschränkung aktiv |
| car\_type | Int | Referenz auf die Cars-Tabelle |

Die Tabelle liegt im core Schema. Diese Tabelle definiert den über den eigentlichen TAPAS-Input (siehe Tripstabelle) hinausgehenden Hintergrundverkehr. Die entsprechenden Eingaben werden analog zur Trips-Tabelle behandelt und in die Umlegung mit einbezogen. Primärschlüssel ist das Tripel p\_id, hh\_id, start\_time\_min. p\_id und hh\_id sind dabei fortlaufende negative Zahlen. Die beiden taz\_id-Spalten können auch negative Zahlen enthalten, die dann den (negierten) Zell-IDs im Deutschlandmodell entsprechen. Der sample\_size Parameter findet auf diese Nachfrage keine Anwendung.

Binnenwirtschaftsverkehr wird bisher nicht berücksichtigt, da keine entsprechenden Eingabedaten vorliegen.

### Tabelle Timeline (core.global\_timelines)

|  |  |  |
| --- | --- | --- |
| **Spalte** | **Typ** | **Beschreibung** |
| interval\_end | Double | In Stunden seit Mitternacht |
| timeline\_id | Int |  |
| share | Double | Anteil am Gesamtvolumen |
| description | Varchar | Freitextbeschreibung |

Die Tabelle enthält die Zeitreihen für eine Nachfragematrix, Primärschlüssel ist das Paar timeline\_id, interval\_end, jede Zeile enthält den Anteil für die jeweilige Stunde am Gesamtverkehr (z.B. timeline 0, interval\_end 7, share 0.3 bedeutet in der Timeline 0 sind von 6-7 Uhr, 30% der Fahrzeuge unterwegs). Die ID der Timeline ist im Abschnitt 2.2.3 angegeben. Alle Simulationszeiten sind immer modulo 24h zu behandlen, das heißt eine Fahrt zu Simulationssekunde 1 kommt in die gleiche Zeitscheibe wie eine Fahrt zu Sekunde 86401.

## Prozesskommunikation SUMO -> TAPAS (Output)

### Tabellen ODMatrix, ODEntry (param\_key: DB\_TABLE\_SUMO\_OD\_OUTPUT DB\_TABLE\_SUMO\_OD\_ENTRY)

|  |  |  |
| --- | --- | --- |
| **Spalte** | **Typ** | **Beschreibung** |
| taz\_id\_start | Serial | ID der Startzelle |
| taz\_id\_end | Serial | ID der Zielzelle |
| sumo\_type | Varchar | Referenz auf die Typliste |
| is\_restricted | Bool | Zugangsbeschränkung aktiv |
| interval\_end | Double | In Stunden seit Mitternacht |
| entry\_id | Serial | Referenz für die Daten Tabelle |
| trip\_source | Enum traffic\_source | („tapas“, „passenger“, „goods“, „business“) |

|  |  |  |
| --- | --- | --- |
| **Spalte** | **Typ** | **Beschreibung** |
| entry\_id | Int | Referenz in die ODMatrix-Tabelle |
| travel\_time\_sec | Double | Reisezeit in Sekunden |
| distance\_real | Double | Zurückgelegte Distanz in m |
| traveltime\_stddev | Double | Standardabweichung der Reisezeit |
| distance\_stddev | Double | Standardabweichung der Distanz |
| realtrip\_count | Int | Anzahl echter Trips (inklusive Klone bei sample\_size < 1) |
| representative\_count | Int | Anzahl zusätzlicher Repräsentantentrips |
| used\_mode | Enum value | Wert aus (“walk”, “bike”, “car“, “passenger“, “public transport”, “taxi”, “train”, “car sharing”, “bike sharing” , “ride sharing”) |

Die Tabellen liegen im temp Schema und der Name enthält neben dem Eintrag aus DB\_TABLE\_SUMO\_OD\_OUTPUT bzw. DB\_TABLE\_SUMO\_OD\_ENTRY noch den sim\_key und die Iteration als Suffix (z.B. "sumo\_od\_2014y\_12m\_19d\_10h\_21m\_21s\_735ms\_1"). Diese Tabellen enthalten die aggregierten Reisezeiten / -weiten für alle OD-Beziehungen. Primärschlüssel ist für die OD\_OUTPUT Tabelle die Kombination aus Quell- und Ziel-TAZ, sowie Typ, Restriction und Intervallende, für OD\_ENTRY die entry\_id und die used\_modes. Liegen für eine solche Kombination weniger als fünf Trips vor, werden Paare von Repräsentanten hinzugenommen, wobei Restrictions nur berücksichtigt werden, wenn das Szenario insgesamt welche hat. Es sollte immer Realtrip\_count + Representative\_count >= 5 gelten (vorausgesetzt es sind genug Repräsentanten definiert). Die Auswahl der Repräsentanten kann zufällig erfolgen. OD-Paare aus dem Durchgangs- und Wirtschaftsverkehr werden nicht durch Repräsentanten ergänzt.

Die Reisezeiten und Reiseweiten werden nur für einzelne Modi bestimmt, Moduskombinationen werden durch TAPAS hergestellt.

Kann es auf Grund von Restrictions keine Fahrten zwischen zwei OD-Bezirken geben, werden beide count-Werte auf 0 gesetzt und eine negative Reisezeit hinterlegt.

Um die Anzahl der Einträge gering zu halten, soll die Anzahl der Fahrzeugtypen beschränkt werden. Die Abbildung von Fahrzeugtypen auf Emissionsklassen ist im Abschnitt Emissionen beschrieben. Möglicherweise werden auch die Typen am Ende aggregiert und es wird nur nach Restricted unterschieden.

Optional können die Daten um routengenaue Informationen ergänzt werden. Dies geschieht jedoch nicht in der Datenbank, sondern in Dateien im Amitran-Format (s. <http://sumo.dlr.de/trac.wsgi/browser/trunk/sumo/data/xsd/amitran/od.xsd>).

### Tabelle Trips (param\_key: DB\_TABLE\_SUMO\_TRIP\_OUTPUT)

|  |  |  |
| --- | --- | --- |
| **Spalte** | **Typ** | **Beschreibung** |
| p\_id | Integer | Eindeutige Personen-ID |
| hh\_id | Integer | Eindeutige Haushalts-ID |
| start\_time\_min | Int | Startzeit wie im Input (ohne Delay) |
| clone\_id | Int | Index des Klons bei Skalierung (sample\_size < 1), Default: 0 |
| travel\_time\_sec | Array von Doubles | Reisezeit in Sekunden pro Modus |
| distance\_real | Array von Doubles | Zurückgelegte Distanz in m pro Modus |

Die Tabelle liegt im temp Schema und der Name enthält neben dem Eintrag aus DB\_TABLE\_TRIP\_OUTPUT noch den sim\_key und die Iteration als Suffix (z.B. "sumo\_trip\_2014y\_12m\_19d\_10h\_21m\_21s\_735ms\_1"). Im Unterschied zu vorherigen Kopplungen wird der Output hier getrennt von dem Trip-Input (aus der Tabelle DB\_TRIPS) behandelt. Diese Tabelle enthält Einzelfahrzeugdaten, p\_id, hh\_id, start\_time\_min, clone\_id bilden den Primärschlüssel, wobei die ersten drei zum Input aus der dortigen Triptabelle passen müssen.

Für die Arrays travel\_time\_sec, distance\_real gilt folgende Reihenfolge für die Indizes. “walk”, “bike”, “car“, “passenger“, “public transport (journey time)”, “access\_pt”, “initial waiting time”, “egress\_pt”, “interchanges (time = waiting time, distance = number of transfers)”, “taxi”, “train”, “car sharing”, “bike sharing” , “ride sharing”. Dabei müssen nicht alle Einträge befüllt sein, wenn einzelne Modi nicht genutzt werden, können vom Ende her Nullen weggelassen werden.

## Prozesssteuerung

Die Konfiguration erfolgt wie oben beschrieben über Einträge in der simulation\_parameters Tabelle. Für ein Szenario ist mindestens der Name des Vorlagenverzeichnisses (param\_key: SUMO\_TEMPLATE\_FOLDER) und des zu erzeugenden Basisverzeichnisses (param\_key: SUMO\_DESTINATION\_FOLDER) wichtig. Des Weiteren lässt sich mittels param\_key „DELETE\_TEMP“ (lösche die temporären Daten jeder Iteration) und „DELETE\_INTERMEDIATE\_RESULTS“ (lösche nach der letzten Iteration auch die Zwischenergebnisse) der Umgang mit temporären Daten konfigurieren. Fehlen diese Parameter wird als Defaultwert false angenommen.

### Prinzipieller Ablauf

* Datenbank periodisch auf neue Aufträge prüfen (liegt in der simulation\_parameters ein Eintrag vor (Kombination aus sim\_key und iteration), dessen letzter Eintrag „pending“ in der Statustabelle ist)
* Wenn neuer Auftrag (ohne vorhandenes Arbeitsverzeichnis)
  1. Vorlage in neues Arbeitsverzeichnis klonen
  2. u. U. Netz neu bauen und Fahrzeugtypen duplizieren um Restriktionen umzusetzen
  3. Shapes in TAZs mit Kanten umsetzen
* Neue Iteration:
  1. Mapping Trip-Koordinaten auf das Netz
  2. Trip-Dateien generieren
  3. Start der Umlegung (Z.B. duaIterate, oneshot, evtl. mit komplexerem Aufruf für dynamische Steuerung der Simulation (ITS))
  4. Bei Beendigung der Umlegung gehen die OD-Matrix und die Trips in die Datenbank und der Status (s.u.) wird auf finished gesetzt
* Wenn letzte Iteration
  1. Postprocessing, z.B. Durchlauf mit detaillierten Emissionsklassen, mit ITS-Maßnahmen wie GLOSA, Erzeugung von Trajektorien, …

Die entsprechenden Werkzeuge dazu sind teilweise als Standard-SUMO-Werkzeuge bzw. aus VEU (I) vorhanden. Die noch offenen Aufgaben hier sind insbesondere (falls nicht explizit anders vermerkt, werden sie von TS bearbeitet):

* Erstellung eines allgemeinen Steuerungsskripts (zur Abarbeitung aller Schritte) inklusive der Statuseinträge in die Datenbank
* Abbildung von Netzrestriktionen in der Netz- und der Fahrzeugtypendatei
* Erweiterung der Umlegungsskripte für den Start adaptiver Simulationen
* Datenbankeinträge für Trips und OD-Matrix erzeugen
* Postprocessing für die Trajektorienerzeugung

### Statustabelle (param\_key: DB\_TABLE\_SUMO\_STATUS)

|  |  |  |
| --- | --- | --- |
| **Spalte** | **Typ** | **Beschreibung** |
| Sim\_key | Varchar | Referenz auf die Simulations-ID |
| Iteration | Int |  |
| Status\_time | Timestamp |  |
| Status | Varchar | Freitextfeld |
| Msg\_type | Enum sumo\_status („message“, „warning“, „error“, „fatal“, “finished”, “started”, “pending”) | “error” meint aktuelle Iteration abgebrochen, “fatal”: Totalabbruch, „finished“: erfolgreiche Beendigung der Iteration „started“ beim Start der Sumo-Iteration, „pending“ wenn der Lauf bereit zur (erneuten) Umlegung ist |

Die Tabelle enthält die Statuseinträge für die einzelnen Iterationen, Primärschlüssel ist sim\_key, status\_time. Eine Iteration für die bereits Daten in dieser Tabelle liegen wird nicht neu gestartet (d.h. um sie neu zu starten muss nicht nur der ITERATION Schlüssel in den simulation\_parameters richtig gesretz sein sondern auch die entsprechenden Einträge in dieser Tabelle gelöscht werden).

# Netzgrundlage

Folgende Netzausschnitte werden statisch in den Vorlagenverzeichnissen als Extraktionen aus Navteq 2012 Q2 (Elmar-Format) sowie daraus erzeugtem SUMO-Netz vorgegeben:

* Berlin 2010, innerhalb des Stadtgebiets komplett, mit Autobahnring und Bundes- / Landesstraßen (wahrscheinlich functional road class <= 2, Hauptstraßennetz)
* Berlin 2020, Ausschnitt wie oben, manuell erweitert um VP 2020
* Berlin 2030, Ausschnitt wie oben, manuell erweitert um VP 2030
* Ausschnitt Berlin-Mitte (Neubezirk) 2010 (umliegendes Gebiet wird nach Evaluation der microsim-Performance geklärt, Ziel ist functional road class <= 2)
  + Um das Netz mikrosimulationstauglich zu machen ist eine Netzanpassung (insbesondere der Abbiegebeziehungen und der Ampelprogramme nötig
* ~~Braunschweig Ausschnitt wie bei TS (AIM) definiert~~ (Braunschweig wurde in der Implementierung erstmal zurückgestellt)
  + ~~Hierzu müssen die Änderungen aus dem 2010er Netzes in das neue Netz eingepflegt werden~~

Zum Netz gehört auch die Beschreibung der TAZ, die als shapefile von VF zur Verfügung gestellt wird und auch als daraus erzeugte SUMO-Polygondatei vorliegt. Der Name der TAZ ist im shapefile in der Spalte „NO“ gespeichert und entspricht der „taz\_num\_id“ in der taz-Tabelle.



Zur Modellierung des Netzausschnitts Berlin-Mitte wird entweder nur der Ausschnitt fein aufgelöst (das heißt mit allen Netzkanten) bereitgestellt (oberes Bild) oder das Gesamtnetz, wobei nur der Ausschnitt aufbereitet und zur Simulation genutzt wird. Es wird zuerst die zweite Variante versucht und nur bei Performanceproblemen die erste genutzt. In jedem Fall wird Berlin-Mitte mikroskopisch simuliert.

# Szenarien

Es existieren Basisverzeichnisvorlagen entsprechend der obigen Netzausschnitte:

* Berlin 2010
* Berlin 2020
* Berlin 2030
* Mitte 2010
* ~~BS 2010~~

Information über das zu generierende Basisverzeichnis und für die Vorlage stehen in der simulation\_parameters Tabelle (siehe Konfigurationskapitel). Für ein eventuelles Szenario Berlin 2040 wird das Basisverzeichnis von 2030 verwendet.

Inhalt einer Basisverzeichnisvorlage:

## Netzdatei (extrahiertes GDF (Elmar-Format 6.1) sumo xml + shapefile für taz)

## Fahrzeugtypen (sumo xml)

Diese werden von VF definiert. Ein Beispiel sieht wie folgt aus:

<routes xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="http://sumo.dlr.de/xsd/routes\_file.xsd">

<vType id="passenger" vClass="passenger"/>

<vType id="delivery" vClass="delivery"/>

<vType id="truck" vClass="truck"/>

<vType id="passenger\_restricted" vClass="vip"/>

<vType id="delivery\_restricted" vClass="emergency"/>

<vType id="truck\_restricted" vClass="custom1"/>

</routes>

Hier werden für alle Fahrzeugtypen die jeweiligen Defaultparameter verwendet.

## Umlegungsskript

Dieses Skript wird als Funktion in einer python-Datei mit dem Namen „\_\_init\_\_.py“ im Vorlagenverzeichnis abgelegt. Die Signatur ist dabei wie folgt:

def assign\_trips(options, first\_depart, last\_depart, routes, weights)

Es generiert eventuelle sumocfg und setzt die Umlegungsparameter und startet anschließend die Umlegung. Entsprechende Standardskripte für DUA und oneshot werden hierzu zur Verfügung gestellt.

## Kantenlisten für Einschränkungen

Dies sind Listen von Kanten-IDs analog sumo selections, die evtl. einzuschränken sind. Sie enthalten in jeder Zeile eine Kanten-ID des SUMO-Netzes. Referenziert werden sie über den Namen der Datei. Alternativ können auch Elemente aus TAZ-Dateien gewählt werden, diese werden dann über den Namen der TAZ-Datei und eine angehängte ID in der Form „<Dateiname>@<TAZ\_ID>“ referenziert.

In der DB werden Maximalgeschwindigkeit und erlaubte Fahrzeugtypen eingestellt.

Für die Definition einer Einschränkung wird ein Format analog JSON gewählt: param\_key=‘sumo\_net\_parameter‘, param\_value=’{“liste.txt”:{“maxSpeed”:{“typ1”:20},”allowed”:[”typ1”, “typ3”]}, “liste2.txt”:{…}}’

## Postprocessingskript

Auch das Postprocessingskript wird als eine Pythonfunktion in „\_\_init\_\_.py“ definiert. Die Signatur ist hier:

def post(options, params, conn, routefile)

er definierten Signatur in einem separaten Modul pro Szenario definiert. Entsprechende Standardskripte für die schon vereinbarten möglichen Postprocessingmechanismen werden hierzu zur Verfügung gestellt.

## Parameter

Die unterschiedlichen Szenarien beschränken sich auf die Veränderung der genutzten Fahrzeugtypen sowie der eingeschränkten Kanten (und deren Geschwindigkeiten). Insbesondere werden in SUMO keine Reichweitenbeschränkung (etwa bei Elektrofahrzeugen) und keine MAUT-Kosten abgebildet, weil die Kostensensitivitätsparameter beim Routen nicht bekannt sind. Das Routing in SUMO erfolgt immer nur fahrzeitoptimierend, andere Zielgrößen werden nicht berücksichtigt. Die Gesamt-Berlin-Szenarien und Braunschweig werden mesoskopisch simuliert, Berlin-Mitte mikroskopisch. Dies ist jedoch nur eine Startvorgabe und im Rahmen der Konfiguration auch änderbar.

Das Postprocessing kann sowohl Zugriff auf die bei der Umlegung erzeugten Daten als auch auf zusätzliche Tabellen (z.B. für detaillierte Fahrzeugtypen) oder Simulationsausgaben (z.B. Trajektorien) nehmen. Dazu ist unter Umständen ein zusätzlicher Simulationslauf nötig, es wird jedoch kein Routing mehr vorgenommen. Zusätzlich können hier noch als Eingabe neue Fahrzeugtypen mit neuen Emissionsfaktoren verarbeitet werden. Dazu müssen die entsprechenden Faktoren noch für SUMO aufgearbeitet werden.

Durch die flexible Konfiguration sowohl der Umlegung als auch des Postprocessing mittels eigener Skripte ist die Schnittstelle hier sehr variabel und erlaubt auch spätere Erweiterung auf ÖPNV etc. Hierzu sind jedoch weitere statische (Netz-)Daten (Fahrpläne, Haltestellen etc.) und eventuell Erweiterungen der Datenbank nötig.

# Validierung

Folgende Datenquellen stehen zur Validierung zur Verfügung:

* Zählungen im Basisjahr 2010 an ausgewählten Stellen in Berlin und BS (Autobahn und Zufahrten), Daten der Bast (Tageswerte)
* Messschleifendaten des Landes Berlin (Quelle: VEU I)
* Stadtschleifen BS (Zählungen und Geschwindigkeiten ab 2013)
* FCD Berlin und BS
* MID 2008: Reiseweitenverteilung (evtl. auch Reisezeiten)
  + Berlin (ohne Durchgangsverkehr)
  + Braunschweig nur im Rahmen eines Raumtyps verfügbar, Verwendbarkeit ist zu überprüfen

Die entsprechenden Daten (Messschleifen und FCD) sind auf das aktuelle Netz neu zu verorten.

Alle Datenquellen werden als korrekt vorausgesetzt, nur wenn es während der Validierung Hinweise auf Unstimmigkeiten gibt, wird stichprobenartig auf Plausibilität geprüft.

Im Folgenden werden die einzelnen Vergleiche kurz dargestellt. Die in den jeweiligen Abschnitten angegebenen Zielwerte sind lediglich als Ideen zu verstehen. Im Ergebnis wird in jedem Fall ein Bericht mit einer differenzierten Darstellung der beobachteten Abweichungen sowie Maßnahmen zur Verbesserung der Güte stehen. Dabei wird in der Simulation die Kalibrierung der Modellparameter (tau, jam\_factor, …) im Vordergrund stehen.

Während die entsprechenden Daten zum großen Teil schon vorliegen sind die entsprechenden Auswertungsskripte in allen Fällen noch zu erstellen. Zur Visualisierung entsprechender Daten liegt in SUMO schon ein Korpus an Skripten / Simulationsvisualisierungen vor, die entsprechend angepasst werden können.

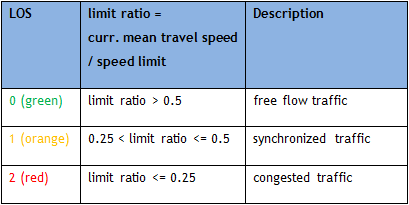
## Zählungen (Messschleifen)

An allen Kanten an denen Zählungen vorliegen, werden die Belastungen auch in der Simulation erfasst und auf Stundenscheiben aggregiert (ausgefahrene Fahrzeuge in der Messstunde). Dabei werden Fahrzeugtypen analog zu den Messdaten unterschieden. Unterschiedliche Fahrzeugklassen werden hierzu mitunter geeignet zusammengefasst.

Der Vergleich erfolgt an allen Kanten für alle Einzelstunden und für den Mittelwert mittels des GEH (s. <http://en.wikipedia.org/wiki/GEH_statistic>). Mögliches Ziel wäre ein GEH kleiner 5 für mindestens 80% der Kanten für alle Einzelstunden.

## Geschwindigkeiten (FCD)

Zum Vergleich der Geschwindigkeiten werden die stündlichen Mittelwerte sowohl der Mess- als auch der Simulationsdaten auf gut abgedeckten Kanten (mindestens 50 Messungen) auf einen dreistufigen Level Of Service (LOS) gemäß folgender Tabelle aggregiert.



Zur Berechnung der Abweichung werden die absoluten Abweichungen im LOS für alle Einzelstunden pro Kante addiert. Ein mögliches Ziel wäre eine Abweichungssumme kleiner 10 für 80% der Kanten. Ein mögliches Problem sind mangelnde Informationen über Busspuren in den genutzten Gebieten, die Taxi-FCD verfälschen.

Geschwindigkeitsinformationen der Messschleifen werden nicht zur Validierung genutzt.

## Reiseweiten und Reisezeiten (MID)

Reiseweiten und -zeiten können als tagesgenaue Mittelwerte (mit den Einschränkungen für die Regionen wie oben) für das gesamte Untersuchungsgebiet aus der MID entnommen werden. Zum Vergleich werden diese in verschiedene Klassen eingeteilt (z.B. 0-2km, 2-5km, 5-10km, die genaue Aufteilung wird von VF vorgegeben) und die entstehenden Anteilsvektoren aus Simulation und MID verglichen. Das Ähnlichkeitsmaß für die Anzahl aller Trips in den entsprechenden Distanzkategorien/Modus-Paare verglichen mit den Referenzwerten aus der MiD2008 ist das Quadrat des Pearson-Korrelationskoeffizienten

wobei die Ähnlichkeit 1 vollständige Übereinstimmung angibt. Angestrebt wird eine Übereinstimmung von mindestens 0.8.

Für die Reisezeiten wird analog vorgegangen, nur enthalten die Einträge der Vektoren nicht den Anteil sondern die mittlere Reisezeit für die entsprechende Reiseweite.

## Validierung der Repräsentanten

Momentan wird in TAPAS ein korrigierter Luftlinienabstand *k* zur Bestimmung der Reisezeit zweier Ziele in verschiedenen TAZ genutzt. Um diesen zu validieren, werden die simulierten Reisezeiten *t* mit diesen Werten verglichen. Die mittlere Abweichung zwischen *k* und *t* zwischen zwei TAZ sollte nicht größer sein als 10%. Ein mögliches Ziel für eine konvergierte Simulationsschleife wäre, dass 80% der tatsächlich durchgeführten Trips um weniger als 10% in der Reisezeit von der durchschnittlichen Reisezeit zwischen den TAZ abweichen.

Zur Evaluation des Netzes wird die Varianz der Reisezeiten aller Startpunkte innerhalb einer TAZ zu den Nachbar-TAZ (Abweichung vom Luftlinienfaktor) geprüft. Auf dieser Grundlage wird die Anzahl der nötigen Repräsentanten bestimmt (hohe Varianz bedeutet Notwendigkeit vieler verschiedener Repräsentanten).

# Schnittstelle zu AS (Lärm)

SUMO liefert Trajektorien in Form von XML Dateien (SUMO –fcd-output 🡪 <http://sumo.dlr.de/wiki/Simulation/Output/FCDOutput>), die Positionen und Geschwindigkeiten für jeden Zeitschritt sowie mit einfacher Nachprozessierung auch Beschleunigungen enthalten. Der Fahrzeugtyp (inklusive des verwendeten Emissionsmodells) ist nicht in diesem Format enthalten. Er kann jedoch mittels der Fahrzeug-ID aus der Datenbank bezogen werden. Die Fahrzeug-ID in SUMO hat dabei das Format *p\_id*\_*hh\_id*\_ *start\_time\_min*. Die genauen eigenschaften des Fahrzeug können aus der Cars-Tabelle oder aus der Definition des sumo\_typs inder entsprechenden XML-Datei entnommen werden.

Es ist zu beachten, dass diese Daten nur aus der mikroskopischen Simulation (Untersuchungsgebiet Berlin Mitte) hinreichend genau bestimmt werden können. Die Spezifikation der Schnittstelle erfolgte im Zuge des 3-wöchigen Besuchs des Mitarbeiters Roland Schuster von AS bei TS/VF im Dezember 2014.

# Nicht funktionale Anforderungen

Ziel der Arbeiten ist eine nutzbare Simulationsschleife aus TAPAS und SUMO auf einem der VF-Simulationsrechner. Dabei soll ein Lauf von SUMO und TAPAS jeweils innerhalb 24h durchführbar sein, wobei 8 Prozessoren, 16GB RAM und 1 TB Plattenplatz zur Verfügung stehen.

Mögliche Schritte zur Laufzeitoptimierung ohne Veränderung der Lösung / des Laufzeitverhaltens sind:

* Einsatz einer SSD / Ramdisk
* Parallelisiertes Routing
* Simulationsinternes Routing

Nimmt man Veränderungen in der Lösung in Kauf sind folgende Maßnahmen möglich

* Makroskopische Startlösung
* One shot
* Abbruchkriterien DUA

Diese werden nicht alle umgesetzt sondern zuerst bezüglich ihrer Wirksamkeit abgeschätzt und vermutlich etwa in der oben beschriebenen Reihenfolge getestet bis die gewünschten Performance-Anforderungen erfüllt werden können.

# Software Engineering / Tests

Es kommen automatische Tests auf der Basis von Texttest (<http://www.texttest.org/>) zum Einsatz. Der von TS entwickelte Teil der Software wird im SiP-Repository abgelegt <http://129.247.218.249:8080/svn/SiP/trunk/projects/tapas>, die entsprechenden Tests unter <http://129.247.218.249:8080/svn/SiP/trunk/tests/scenario/tapas>. Das Bugtracking erfolgt mittels Trac <http://129.247.218.249/tracsip/> wobei eine eigene Komponente für die TAPAS-Kopplung angelegt wird. Für die automatisierten Tests kann die auf dem Server ts-simulation8 (129.247.218.50) installierte PostGres-Datenbank genutzt werden. Hierzu ist noch ein kleines Testszenario mit entsprechenden Inputdaten zu spezifizieren.

# Emissionen

Die Fahrzeuge werden in Größenklassen eingeteilt wobei die folgende Abbildung gilt:

|  |  |
| --- | --- |
| **Klasse** | **Kba\_no** |
| Small | 1, 2, 101 |
| Medium | 3, 4, 9, 10, 102 |
| Large | 5, 6, 7, 8, 95, 103 |
| Transporter | 11, 12, 104 |

Die Abbildung von den in der Cars-Tabelle definierten Typen auf die SUMO-Emissionsklassen erfolgt je nach verwendetem Emissionsmodell nach folgender Aufstellung. Der vollständige Name einer Emissionsklasse setzt sich dabei aus dem Modell und der spezifischen Klasse zusammen, z.B. „HBEFA2/P\_7\_7“ oder „HBEFA3/PC\_G\_EU5“:

## HBEFA2 7 Cluster

Für Fahrzeuge der Größenklasse Transporter wird bei Benzin die Emissionsklasse P\_7\_3 gewählt, für Diesel P\_7\_5. Alle anderen Fahrzeuge werden als P\_7\_7 klassifiziert.

## HBEFA2 14 Cluster

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Größenklasse** | **Emissionsklasse / Kraftstoff** | | | | | |
|  | Euro0/1 Benzin | Euro0/1 Diesel | Euro2/3 Benzin | Euro2/3 Diesel | Euro4+ Benzin | Euro4+ Diesel |
| Small | P\_14\_14 | P\_14\_7 | P\_14\_9 | P\_14\_8 | P\_14\_9 | P\_14\_8 |
| Medium | P\_14\_14 | P\_14\_10 | P\_14\_9 | P\_14\_8 | P\_14\_9 | P\_14\_8 |
| Large / Transporter | P\_14\_14 | P\_14\_10 | P\_14\_13 | P\_14\_4 | P\_14\_13 | P\_14\_8 |

## HBEFA3

Die Größenklassen Small und Medium werden als PKW (PC) geführt und die Größenklassen Large und Transporter als leichte Nutzfahrzeuge (LDV). Die weitere Zuordnung erfolgt direkt nach Euro-Norm und Kraftstoffart und der Klassenname ist PC\_*X*\_EU*Y* bzw. LDV\_*X*\_EU*Y*, wobei *X* ein G für Benzin oder D für Diesel sein kann und *Y* eine Ziffer von 0 bis 6 (einschließlich) entsprechend der Euro-Norm.

## PHEMlight

Die Größenklassen Small und Medium werden als PKW geführt und die Größenklassen Large und Transporter als leichte Nutzfahrzeuge (LNF). Die weitere Zuordnung erfolgt direkt nach Euro-Norm und Kraftstoffart und der Klassenname ist PKW\_*X*\_EU*Y* bzw. LNF\_*X*\_EU*Y*\_I, wobei *X* ein G für Benzin oder D für Diesel sein kann und *Y* eine Ziffer von 0 bis 6 (einschließlich) entsprechend der Euro-Norm.

## Aggregation

Um eine Reduktion der Fahrzeugklassen zu erreichen, wird für die OD-Matrizen im Output eine weitere Aggregation in insgesamt vier Klassen vorgenommen wobei nur noch nach Kraftstoffart und Euro-Norm >= 4 (oder nicht) unterschieden wird.