DFROUTER - Route estimate methods based on detector data

TeRon Nguyen,
Institute for Transportation, TU München
teron.nguyen@tum.de

Agenda

Introduction

Traffic demand is actually the key input for transportation system operation, design, analysis and planning.

Various methods have been developed for generating traffic demand:

- Direct estimation
- Model estimation
- Origin Destination matrix estimation (ODME) from traffic counts

→ Many researches have been focusing on developing ODME from traffic counts over the last 30 years

Study aiming

The study aims at **analyzing, evaluating and improving** the open-source “DFROUTER” tool.

- DFROUTER algorithm has been not **described** in detail.

- The algorithm will be well **compared** with other similar approaches.

- Suggestions for the **improvement** of DFROUTER in order to calculate routes/demand more accurately.

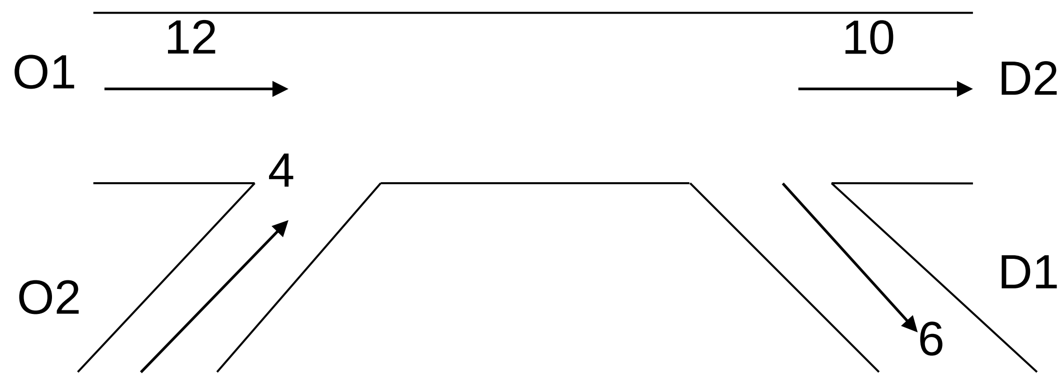
Estimation of OD matrices from traffic counts

Specifically for highway, the problem of determining of OD matrix from traffic counts can be formulated as follow:

$$\sum_i b_{ij} Q_i = O_j$$

$$\sum_j b_{ij} = 1$$

Where:
b_{ij} = proportion of trip from i to j;
Q_i = on-ramp counts (origin flows);
O_j = off-ramp counts (destination flows).



Sample highway segment

→ **the under-specification problem**

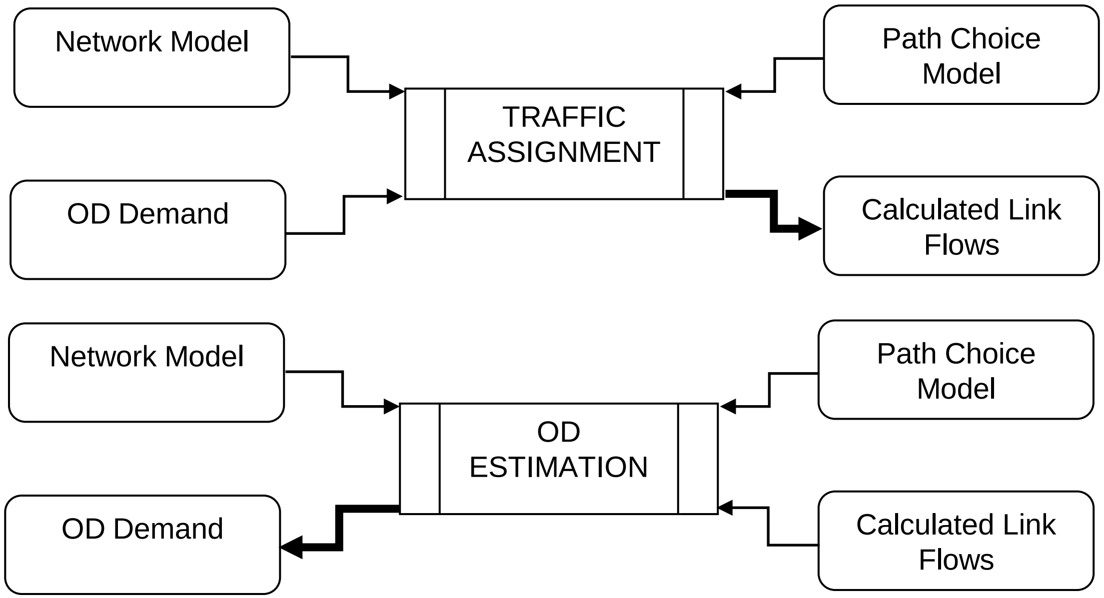
	D1	D2	Sum
O1	8	4	12
O2	2	2	4
Sum	10	6	16

	D1	D2	Sum
O1	10	2	12
O2	0	4	4
Sum	10	6	16

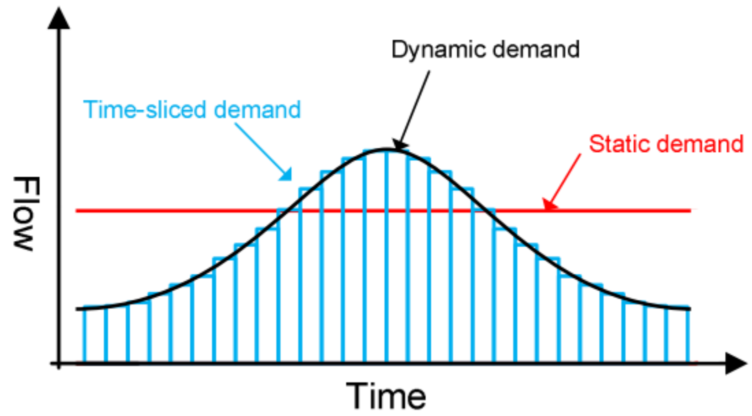
	D1	D2	Sum
O1	6	6	12
O2	4	0	4
Sum	10	6	16

	D1	D2	Sum
O1	9	3	12
O2	1	3	4
Sum	10	6	16

Estimation of OD matrices from traffic counts



Relationship between estimation of O/D flows with traffic counts and traffic assignment [Cascetta, 2001]



Traffic demand representation by traffic count [Bert, 2009]

Traffic assignment	Un-congested	Congested
OD adjustment	Static	Dynamic

Four cases for OD estimation

Static ODME

Estimate average OD demand for long-time transport planning and design purpose

The link volumes
$$V_a = \sum_{ij} p_{ij}^a T_{ij}$$

Two optimization functions

min $$F(T,V) = \gamma_1 F_1(T,\hat{T}) + \gamma_2 F_2(V,\hat{V})$$

- minimization the distance between estimated OD matrix and the target OD matrix

- minimization the difference between estimated link flows and observed link flows

Dynamic ODME

Consider OD flows with time variation for short-term strategies traffic control and management

Non-DTA-based approaches: intersection or small freeway segment

$$y_j(k) = \sum_{1}^{k} \sum_{i=1}^{M} b_{ij}(k) \times q_i(k)$$

DTA-based approaches: large urban or freeway network

Upper level

min $$F(T,V) = \gamma_1 F_1(T[t],\hat{T}[t]) + \gamma_2 F_2(V_{lj}[t],\hat{V}_{li})$$

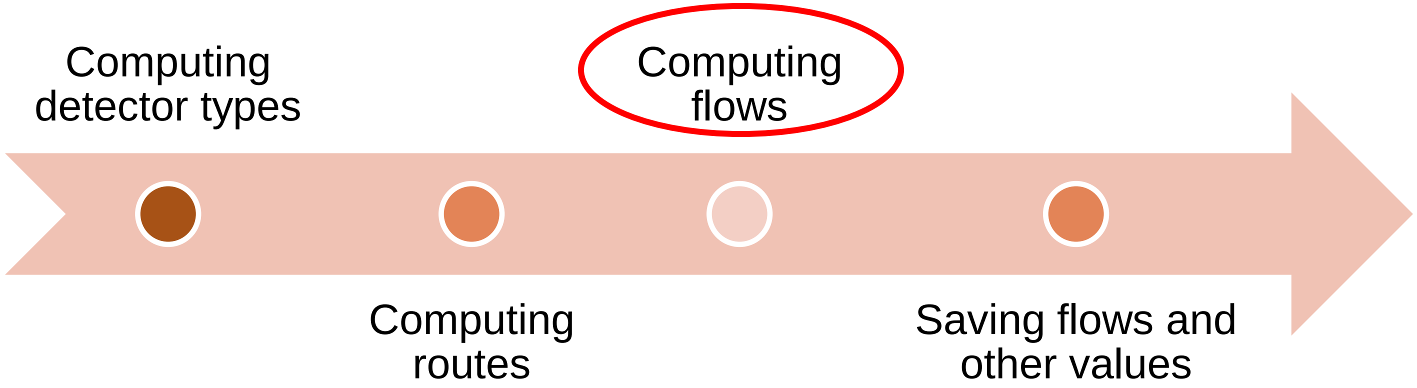
Lower level

$$V_{lj}[t] = assign(T[t]) = \sum_{t=1}^{j} \sum_{od} m_{lj}^{od,t} T[t] + \varepsilon(l,t)$$

SUMO suite and DFROUTER

DFROUTER routing module has been designed specifically to highway scenarios based on the idea that most highways are well equipped with induction loops, measuring each of the highways' entering and leaving flows.

From this information regarding vehicle types, flows and speeds DFROUTER is able to rebuild vehicle amounts and routes.



The initial DFROUTER was not a major work, but rather **a tool in a larger system for calibration** the microscopic simulation scenarios.
→ adding and removing vehicles to / from the simulation at the measurement points so that they match the real-world counts → need routes

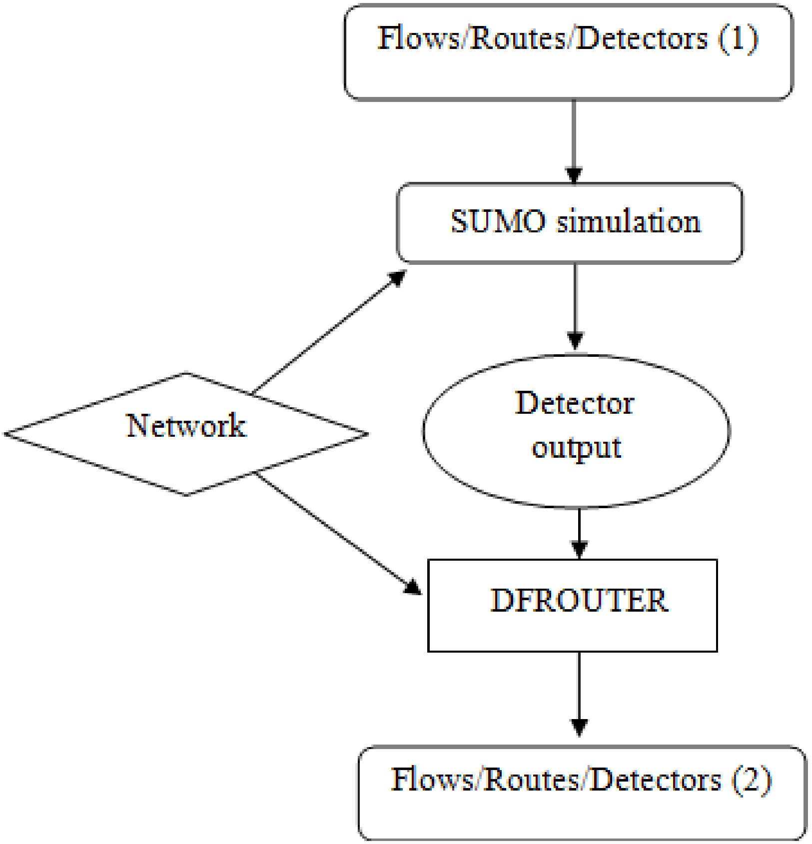
Methodology

The research methods focus on three key areas:

1. A more formal description of DFROUTER

2. How does the algorithm compare to similar approaches?

3. How could the algorithm be improved in order to estimate routes more accurately?



General framework to analyze DFROUTER tool

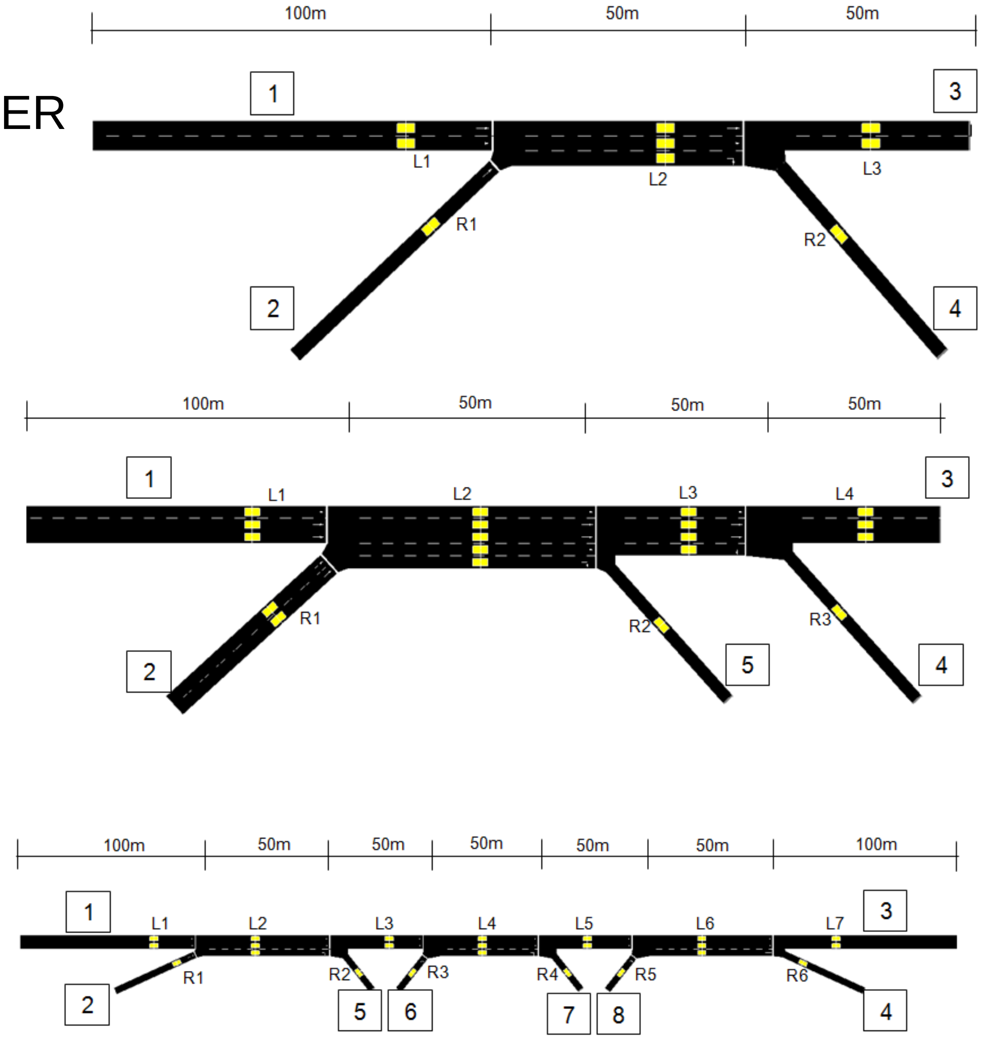
Results

1. A more formal description of DFROUTER

-The algorithm works well whenever the network is fully covered with detectors and generates routes comprising all OD pairs. The algorithm could not detect the missing routes.

-Missing of in-between detectors does not cause a big estimation problem as long as the source and sink detectors are present.

- Basically the estimated probabilities are identical to flow proportions at destinations, therefore sink detectors are the decisive elements in flow computation.



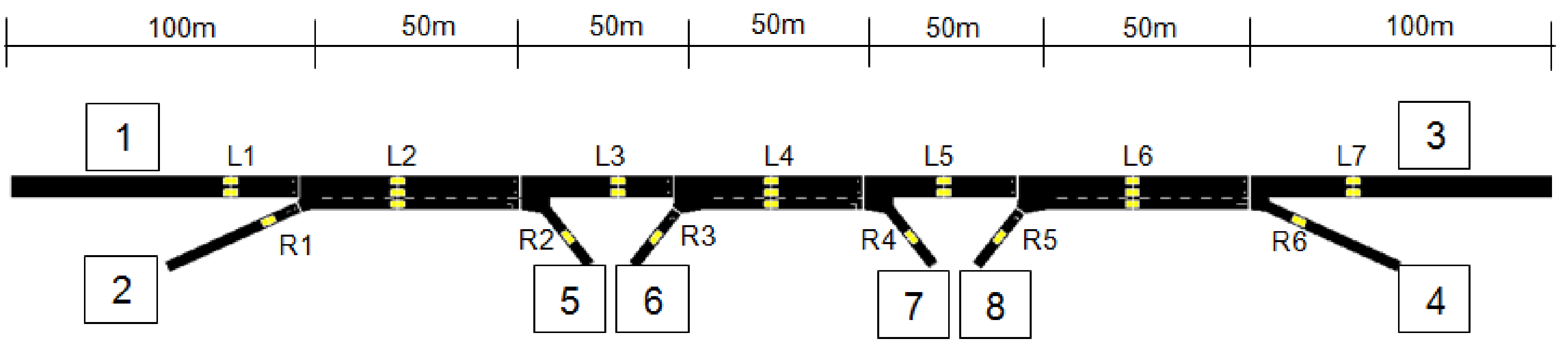
Testing of 3 Cases

1. A more formal description of DFROUTER

Step 1: For all routes starting from source or between detectors to sink detectors → determining split edges (the legs go out of a junction) having detectors on them.

Step 2: Calculating proportion of flow on split edges using detector data → each split edge contains different probability, the others have probability = 1,0 as default.

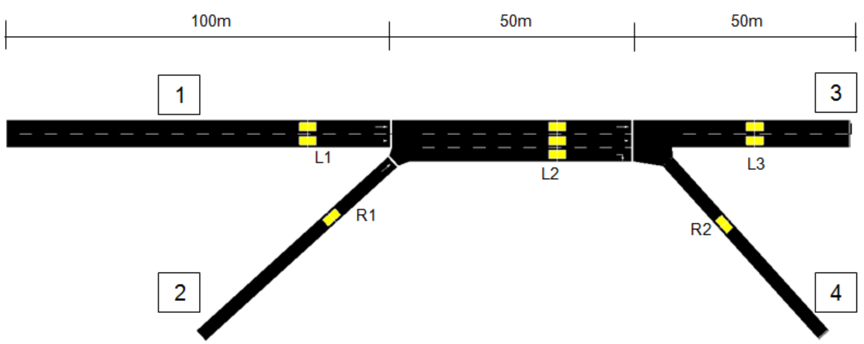
Step 3: For only routes starting from source detectors → calculating destination distribution by multiplying all flow probabilities on all edges constructing that route.



1. A more formal description of DFROUTER

Pros:

-simple and fast calculation in any kind of network.
-requires basically source and sink detector data.



Cons:

-The algorithm could not work successfully in the case of missing detectors, especially detector data on split edges.
-Probability of missing flow is overestimated to 1.0 as default.
-Under-specified problem .

However the absent flow from one detector can be deduced by subtraction of all inflows to all outflows.

→ Additional information like a priority matrix or a specific route assignment is required to compute routes/demand properly.

2. How does the algorithm compare to similar approaches?

DFROUTER generates route/demand data based merely on flow proportions on split edges. This method works similarly to a static OD matrix estimation, but:

- not any constraints between link flows

- not any optimization function

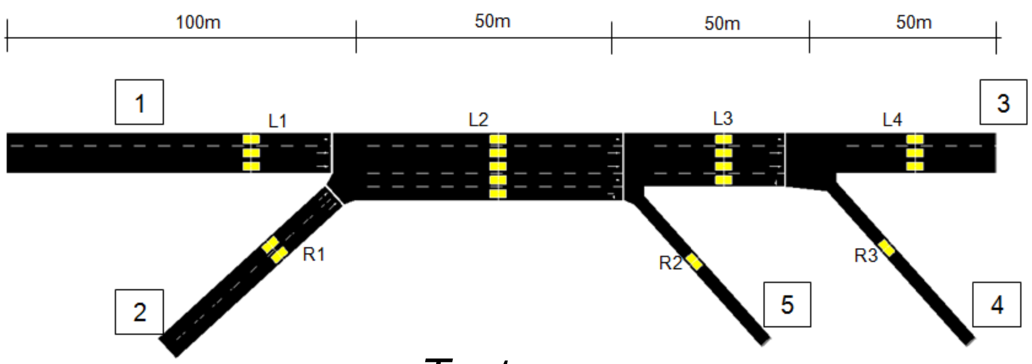
- no congestion effects and travel time between origin and destination

→***It does not work as an OD estimator***
→ ***can not be compared with OD matrix estimation methods such as Maximum Entropy, Minimum Information, Bayesian Inference, Generalised Least Square, etc.***

2. How does the algorithm compare to similar approaches?

	Method	Description
	DFROUTER	multiplying the available probabilities on edges
1	The Equally Split OD matrix	equal proportion is assigned to all destinations
2	Proportional OD matrix	the attraction of any destination is the function of the number of trips that end at that destination
3	Iterative method	balances both inflows and outflows based on an iterative fitting algorithm
4	The Gravity model	a concept that probability of very long and very short trips is low in the freeway
5	The Turning Percentage	is based on turning percentage

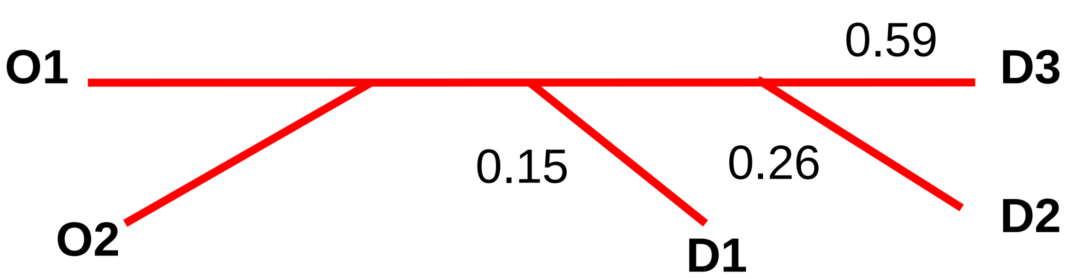
Item	Value
Section length	100, 50, 50, 50
On-ramp counts	280, 180
Off-ramp counts	70, 120, 270
Mainline counts	280, 460, 390, 270



Test case

2. How does the algorithm compare to similar approaches?

	Method	Results
	DFROUTER	
1	The Equally Split OD matrix	implausible result.
2	Proportional OD matrix	Identical to DFROUTER
3	Iterative method	Identical to DFROUTER
4	The Gravity model	depends heavily on the treatment of external stations
5	The Turning Percentage	Identical to DFROUTER



DFROUTER results

2. How does the algorithm compare to similar approaches?

The simple proportionality scheme on has the problem of over-predicting the number of very short and very long trips with 20-30% level of error (L.Nihan, 1979).

→ Due to the drawback of these methods, they are often used to generate starting solution (*seed or target, a priori matrix*) for the OD estimation problem to solve the minimization function of difference between estimated and observed link flows or OD matrix

3. How could the algorithm be improved to estimate routes more accurately?

Calculating missing data based on existing detector flows.

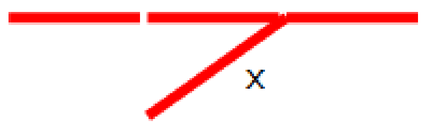
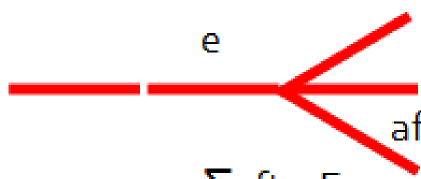
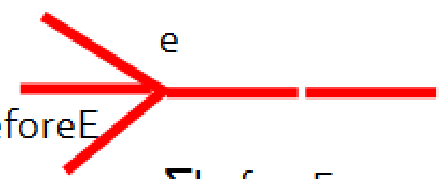
Step 1: Calculate the flow value on each edge of the highway network using recursion backward or forward.

Step 2: For all routes starting from source or between detectors to sink detectors → determining split edges after a junction.

Step 3: Calculating flow proportion of split edges based on computed flow → each split edge contains different probability.

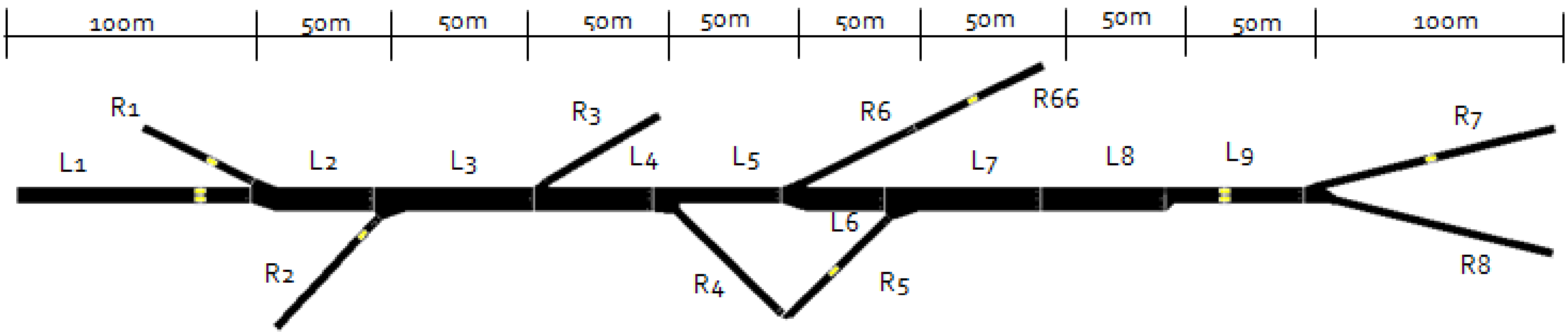
Step 4: For only routes starting from source detectors → calculating destination distribution by multiplying all flow probabilities on all edges constructing that route.

3. How could the algorithm be improved in to estimate routes more accurately?

	Recursion forward	Recursion backward
1	e afterE  e = afterE	beforeE e  e = beforeE
2	e afterE  x e = afterE – x	beforeE e  x e = beforeE - x
3	e  afterE e = ∑afterE	e  beforeE e = ∑beforeE

Cases to consider recursion algorithm

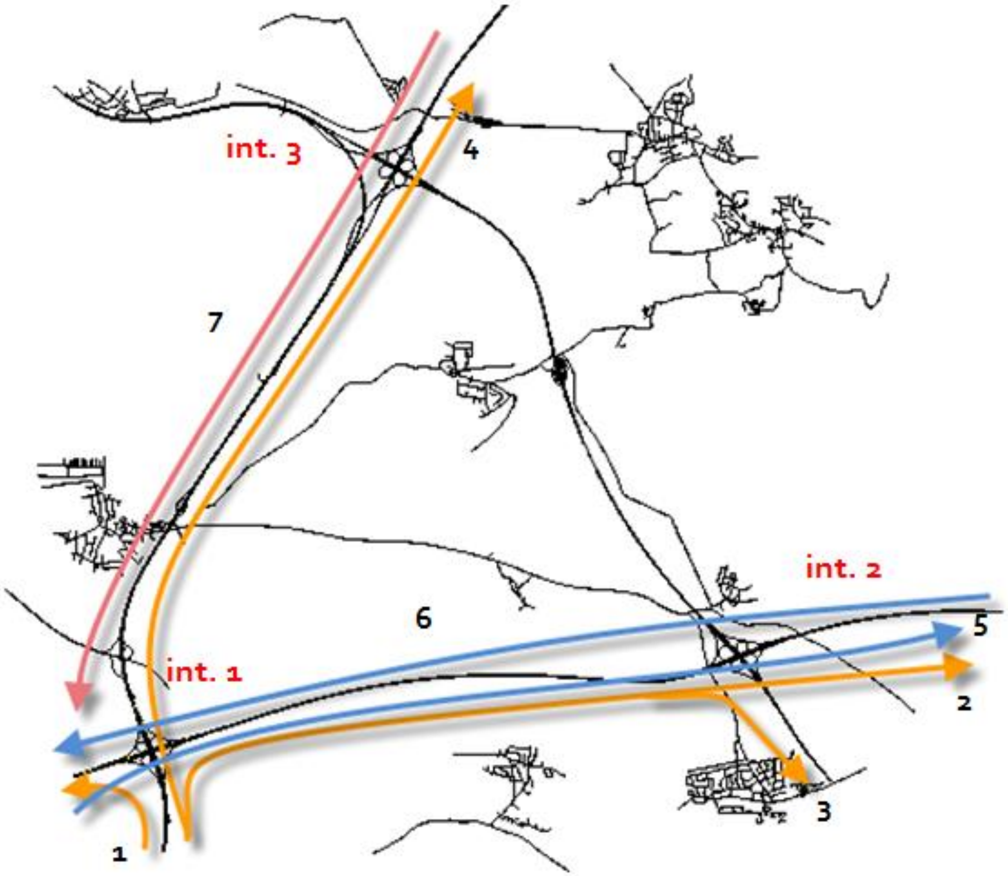
Application in an abstract network



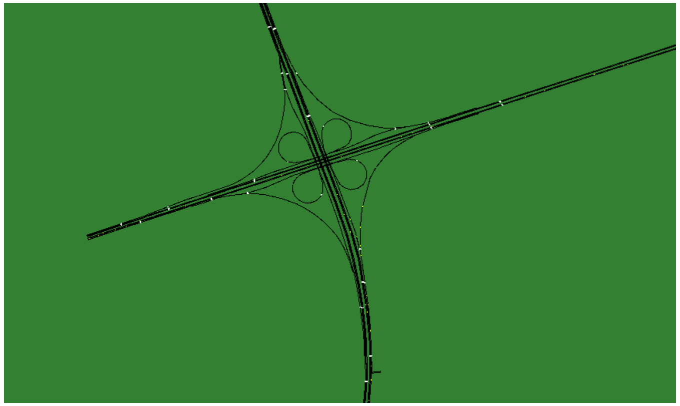
Trip	Des-counts	Des-Probability	Probability		Relative Error	
			DFROUTER	Improved DFROUTER	DFROUTER	Improved DFROUTER
From L1/R1/R2 to R3	900	0.24	1	0.23	3.22	-0.03
From L1/R1/R2 to R66	500	0.13	0.14	0.13	0.06	-0.01
From L1/R1/R2 to R7	1100	0.29	0.69	0.28	1.38	-0.03
From L1/R1/R2 to R8	700	0.18	0.69	0.20	2.75	0.09
From L1/R1/R2 to R7_1	300	0.08	0.69	0.09	7.74	0.14
From L1/R1/R2 to R8_1	300	0.08	0.69	0.06	7.74	-0.24
Total					**22.90**	**-0.19**

→The probabilities generated by the improved DFROUTER are approximate to the destination probabilities and also more accurate compared to that of the original DFROUTER

Application in a larger network



Nuremberg highway network



Application in a larger network

No	Trip	Color	Input	Probability		Relative Error	
				DFROUTER	Improved DFROUTER	DFROUTER	Improved DFROUTER
1	From 1 to 1 left	Yellow	0.16	0.16	0.16	0.00	0.00
2	From 1 to 2 straight1	Yellow	0.13	0.06	0.18	-0.54	0.38
3	From 1 to 2 right	Yellow	0.09	0.22	0.04	1.44	-0.56
4	From 1 to 3	Yellow	0.63	0.24	0.62	-0.62	-0.02
5	From 1 to 2 straight2	Yellow	1	0.28	0.82	-0.72	-0.18
6	From 2 to 1	Blue	1	1	1	0.00	0.00
7	From 3 to 1	Pink	1	0.4	0.86	-0.60	-0.14
Total						**-1.03**	**-0.51**

→ More reliable and accurate results from the improved DFROUTER are achieved

Conclusion

The study has been conducted to analyze the DFROUTER tool. The algorithm has been compared with other similar approaches. The study has also focused on proposing an improved algorithm, which generates traffic demand/flow probabilities more accurately.

The algorithm of flow computation has operated efficiently up to now.

The comparison of similar methods: none of them provides better output than DFROUTER.

More testing in practical highway scenarios are needed to complete the improved algorithm.

Future research:

- wrong detector data or broken detectors
- demand calculation for urban areas
- take into account time variation.

References

- BERT, E. 2009. *Dynamic urban origin-destination matrix estimation methodology.* EPFL.

- CASCETTA, E. 1984. Estimation of trip matrices from traffic counts and survey data: A generalized least squares estimator. *Transportation Research Part B: Methodological,* 18**,** 289-299.

- DANIEL KRAJZEWICZ, J. E., MICHAEL BEHRISCH, AND LAURA BIEKER 2012. Recent Development and Applications of SUMO – Simulation of Urban MObility. *International Journal on Advances in Systems and Measurements,* 5.

- L.NIHAN, N. 1979. Use of volume data to reproduce ramp-to-ramp freeway trip patterns - A pilot study. *Technical report standard, Washington State Department of Transportation,* WSDOT - 35.1.

- LINA KATTAN, B. A. 2011. TRAFFIC ORIGIN-DESTINATION ESTIMATION. *Handbook of Transportation Engineering, Volume II: Applications and Technologies, Second Edition.* McGraw Hill Professional, Access Engineering.

- MUTHUSWAMY, S., LEVINSON, D., MICHALOPOULOS, P. & DAVIS, G. 2005. Improving the Estimation of Travel Demand for Traffic Simulation: Part I.

- VAN ZUYLEN, H. J. & WILLUMSEN, L. G. 1980. The most likely trip matrix estimated from traffic counts. *Transportation Research Part B: Methodological,* 14**,** 281-293.

Thank You
for your attention and questions!