

On the SatPath to Discovery

Transport for London – Ken Fox & Callum Hale

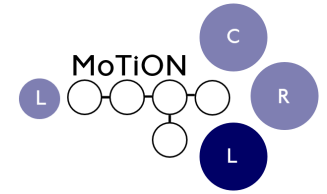
SATURN ANNUAL UGM – MARCH 2024

On the SATPATH to Discovery
Transport for London

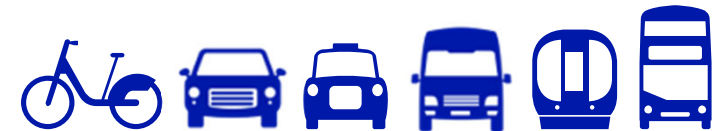


EVERY JOURNEY MATTERS

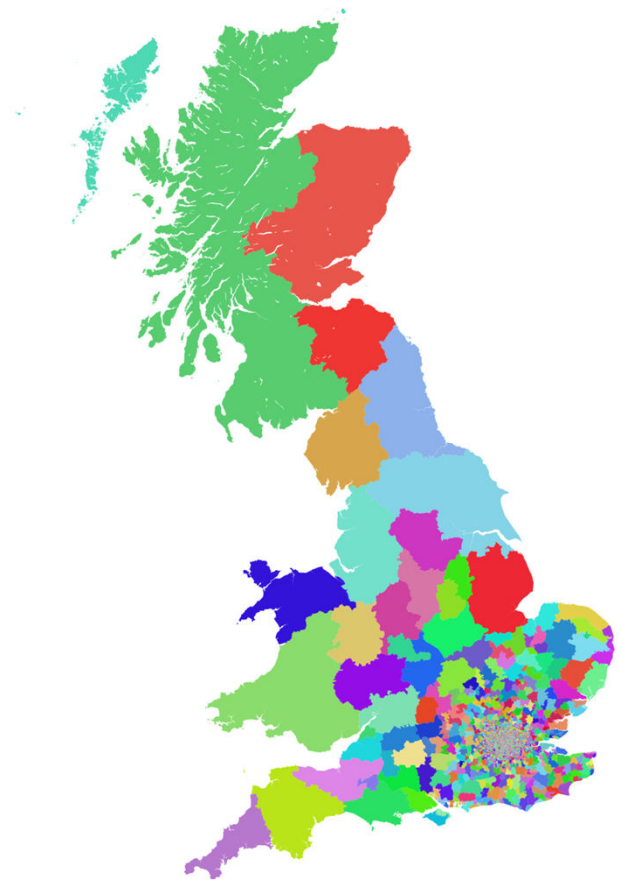
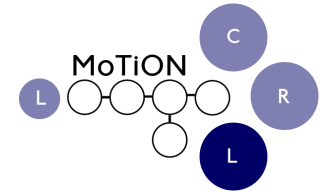
London Highway Assignment Model (LoHAM)



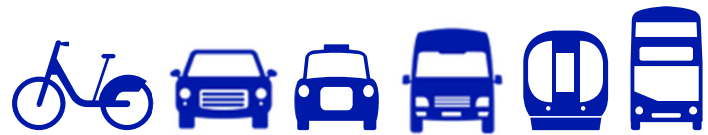
- LoHAM is the highway assignment component of MoTiON
- Produced with SATURN
- LoHAM has been developed since 2008
 - First version P1 – Released 2011
 - Now on version P5.1 – Released February 2023



LoHAM Zones Great Britain

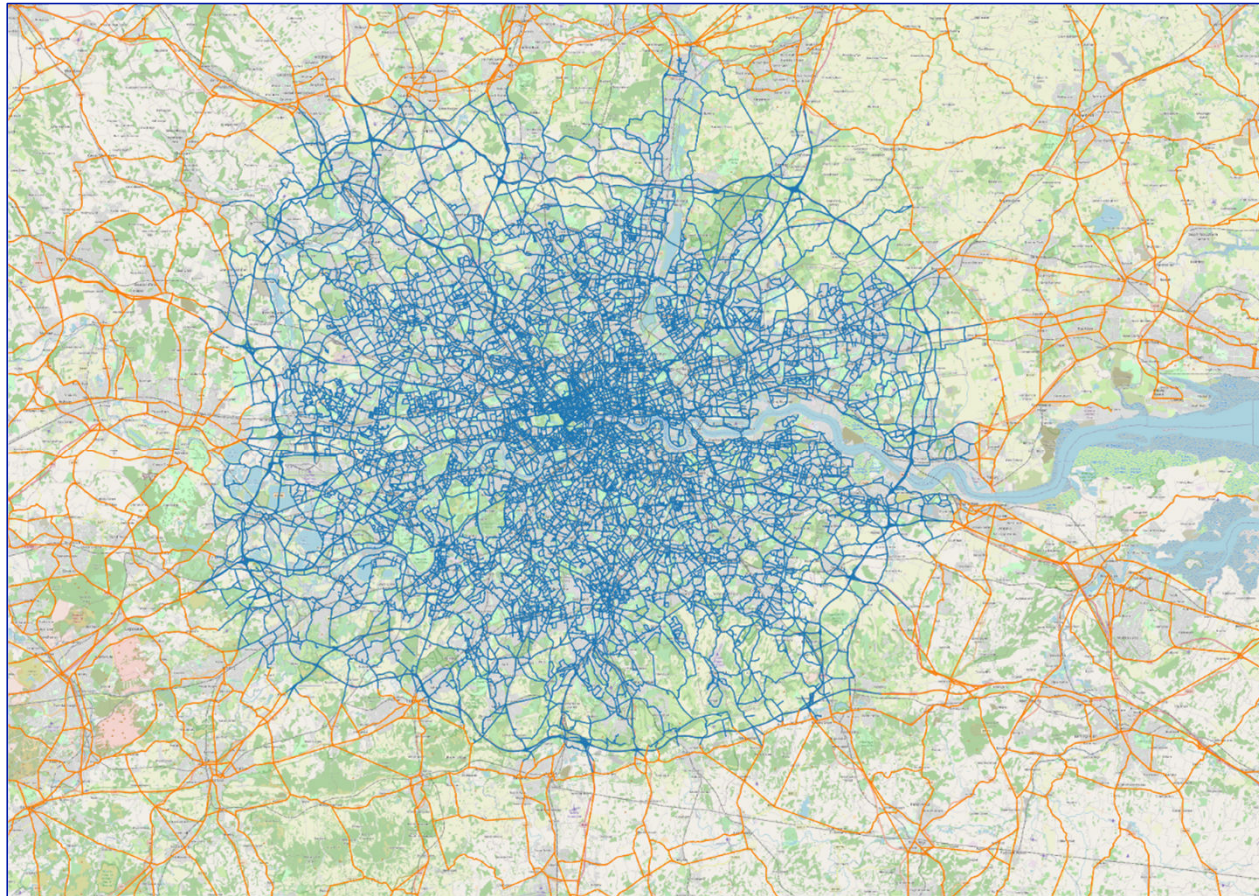
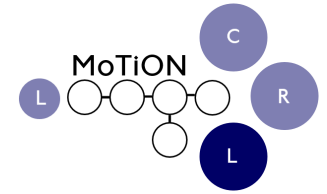


5,421

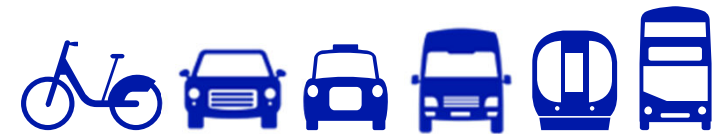


KEEPING LONDON MOVING

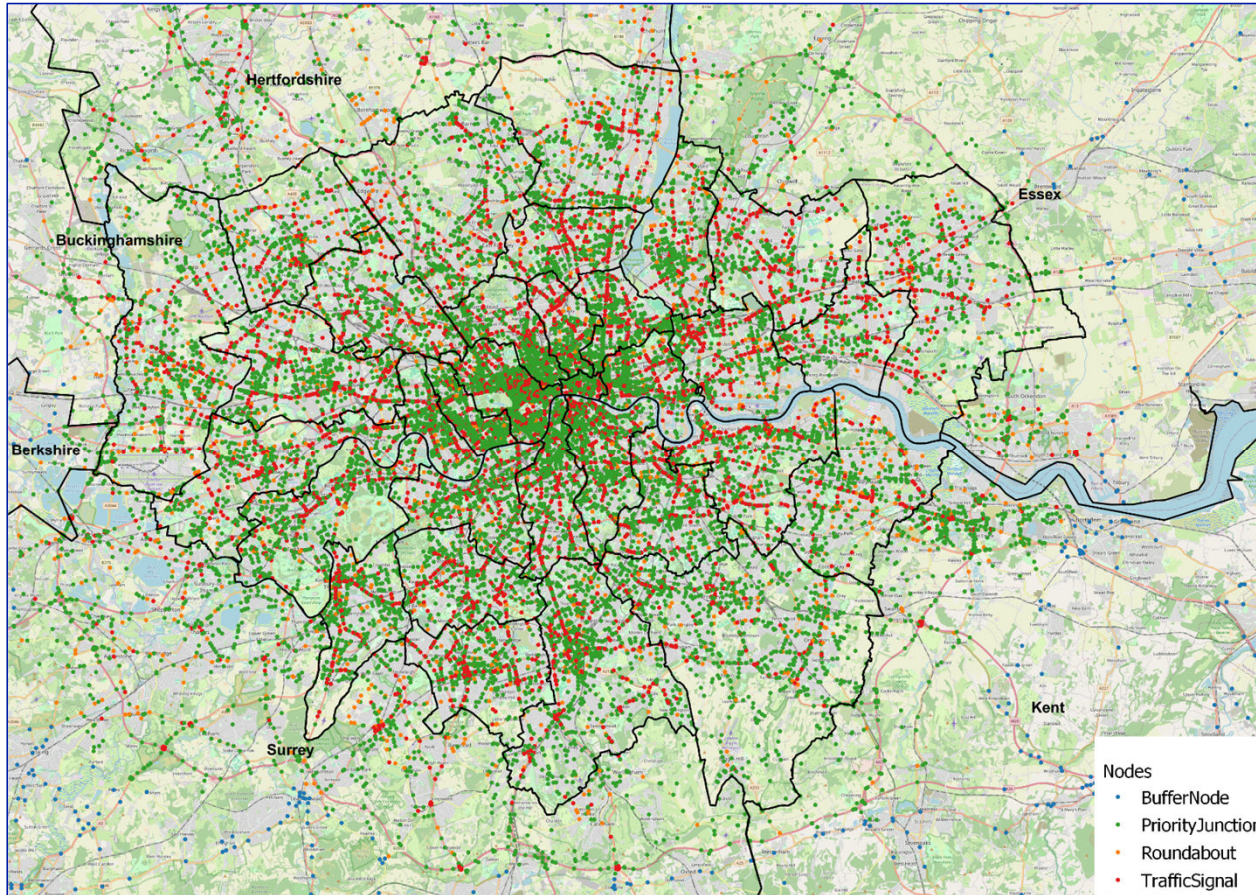
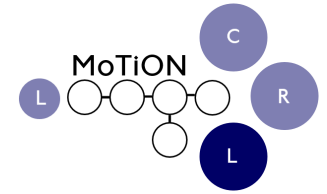
LoHAM Simulation and Buffer Links



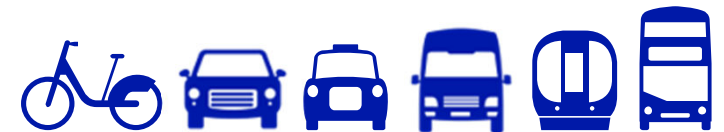
101,389

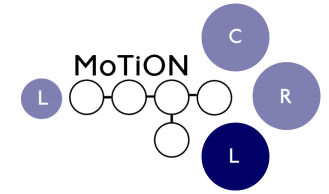


LoHAM Junctions



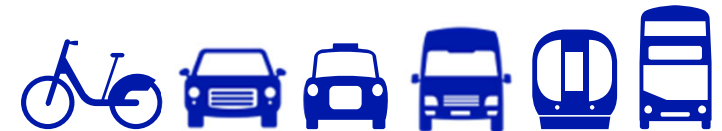
Simulation Nodes	25,780
Priority Junctions	15,722
Roundabouts	1,499
Traffic Signals	5,174
External Nodes	3,385
Buffer Nodes	4,746

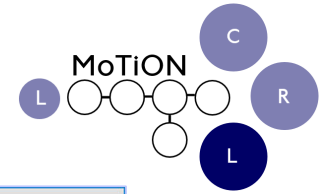




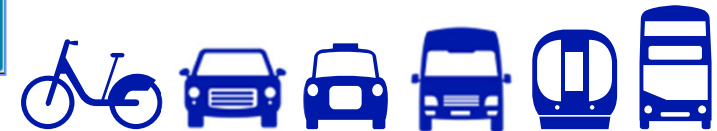
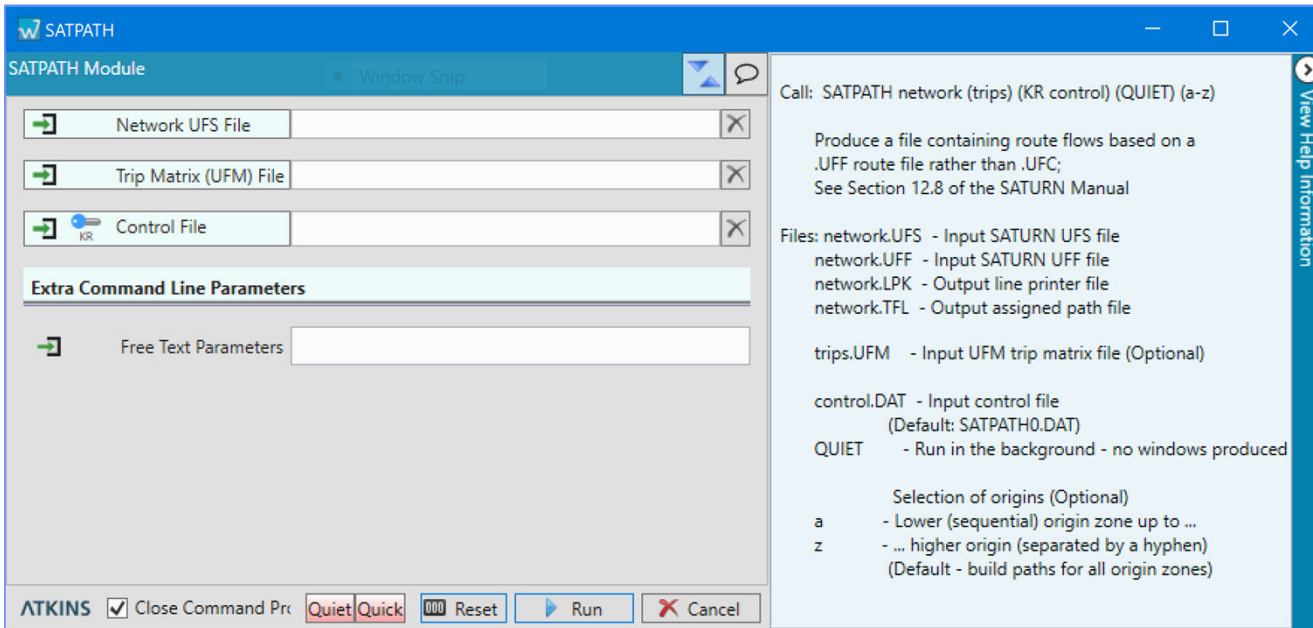
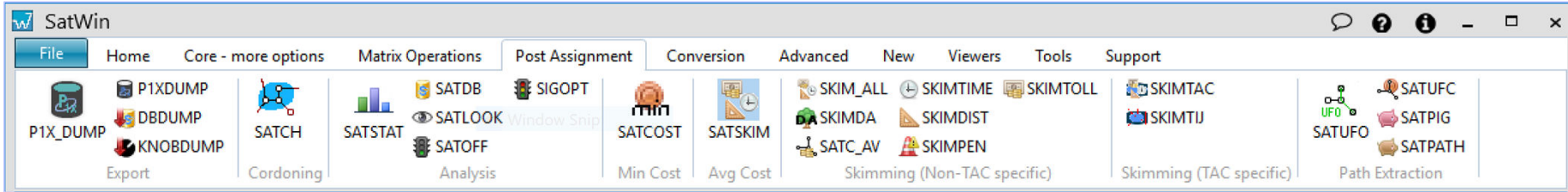
SATPATH (See Section 12.8 of the SATURN Manual)

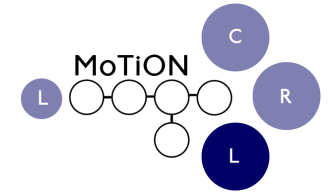
- **SATPIG** is an ad-hoc program to *produce a file of origin-destination route flows from a SATURN assignment*. Originally used to provide route flows for micro-simulation models. It used the SATURN assignment UFC file. However, it doesn't work with TAC networks.
- **SATPATH** is a much updated version of SATPIG introduced in version 11.6.03A to take advantage of .UFF files to rebuild trees, spider networks, etc. which runs very much faster and will cope with very much larger networks. Its initial format is aimed at creating files which will be of use to TfL so that its output format has been newly designed for that purpose. SATPATH works with both TAC and non-TAC networks.





Running SATPATH

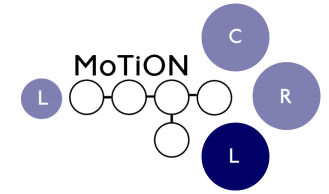




Running SATPATH

- **SATPATH requires as input the new SATURN 11.6.03A .UFF file (Produced by assignment with SATURN2F or SATUFF with a UFS)**
- **File sizes for LoHAM (they are big!)**
 - LoHAM 6UC .UFS (139,441 KB)
 - LoHAM 6UC .UFF (5,244,316 KB) - 38 times bigger than UFS
- **UFF file takes a while to produce after main assignment**
 - For LoHAM 6UC (about 45 mins after 140 min assignment)
- **Produce .TFL file using SATPATH with UFF file**
 - .TFL file is in CSV format
 - Output large (for 6UC LoHAM has 760,356,432 lines and is 371,786,775 KB) - 71 times bigger than UFF
 - Takes about 350 mins to produce from UFF
- **Current TfL Computers - Xeon(R) Gold 6254 @ 3.10 GHz (48 processors), 184 GB RAM.**

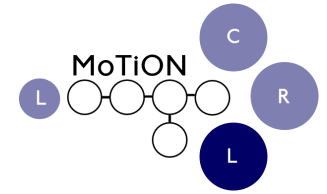




.TFL files

- **A text CSV file with the assigned paths and user class flows on them**
- **Records of 2 lines for each path**
 - Record/Line 1
 - The sequential origin number
 - The origin name
 - The sequential destination number
 - The destination name
 - Path counter (i.e. 1 for first path, 2 for second path)
 - Number of nodes included in record 2
 - Origin TAC and Destination TAC
 - For each User Class
 - Fraction of total for current path to total for same O/D pair
 - UC Flow (PCU/hr)
 - Record/Line 2
 - The full set of nodes per O-D path (Origin Zone,N1,N2,N3,....Nn-1,Nn,Destination Zone)





SATPATH .TFL Output File of Path Flows (First 6 paths)

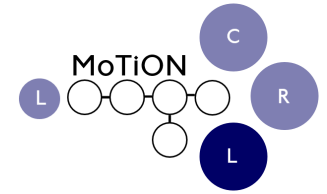
Origin Destination Path No n Path Nodes TAC 1 TAC 2 For each UC {Fraction of total, UC Flow (PCU/hr)}

	Origin	Destination	Path No	n	Path Nodes	TAC 1	TAC 2	For each UC {Fraction of total, UC Flow (PCU/hr)}																		
1	1	10001	2	10002	1	11	1	1	1	0.00288	1	0.000965	1	0.001768	1	0.000592	1	0.028817	1	0.27336	1	0.063987	1	0.030111	1	
2	10001	12845	12846	13237	12580	12813	13238	13239	13240	13237	12846	12845	10002													
3	1	10001	3	10003	1	9	1	1	1	0.00945	1	0.00316	1	0.033681	1	0.011285	1	0.190936				1	0.077505	1	0.036473	1
4	10001	10403	10201	10128	10205	10283	10206	10137	10011	10013	10003															
5	1	10001	2	10003	2	9	1	1	1											1	0.103061					
6	10001	10403	10404	10281	10128	10205	10206	10137	10011	10013	10003															
7	1	10001	4	10004	1	7	1	1	1	0.007461	1	0.0025	1	0.009665	1	0.003238	1	0.10571				1	0.116581	1	0.054862	1
8	10001	10403	10281	10128	10124	10257	10256	10394	10004																	
9	1	10001	4	10004	2	8	1	1												1	0.049933					
10	10001	10403	10404	10281	10128	10124	10257	10256	10394	10004																
11	1	10001	5	10005	1	8	1	1	1	0.005998	1	0.00201	1	0.002555	1	0.000856	1	0.112502				1	0.051833	1	0.024392	1
12	10001	10403	10281	10128	10205	10283	10206	10137	10011	10005																

Origin Zone, Node 1, Node 2, Node 3, Node 4, Node 5, Node 6, Node 7, Node 8, Node 9, Node 10, Node 11, Destination Zone

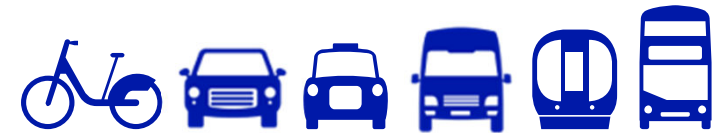


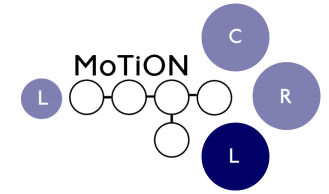
KEEPING LONDON MOVING



Potential Analysis

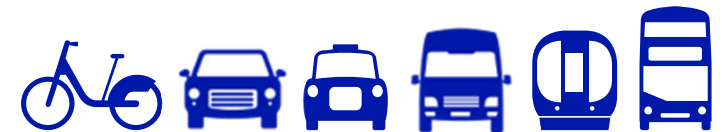
- Trip Length Distributions
 - Area, e.g. within a Borough, GLA, CCZ etc
 - User Class (Car, ULEZ Compliant, Taxi, HGV, ZEV etc)
- Road Use in Area
 - Origin / Destination
 - User Class
- Analysis can be done with any general purpose programming language.
- I have been using Python



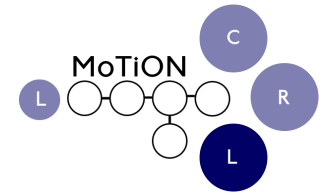


Example Bespoke Python Script - Area Trip Lengths Tool

- Developed to see how trip lengths vary within areas in London
- In particular look at trip lengths and trip length distributions within Central, Inner and Outer London and External to London.
- To answer this need to know the paths that each vehicle makes within the network, the origin, the destination and whether and how the path goes through each area. i.e. the information in the new .TFL output file
- Have written a Python script to analyse .TFL files to answer this question



Example Bespoke Python Script- Area Trip Lengths Tool



– Inputs

- Model link lengths - from .UFN file via SATDB
- Model link areas (**C**entral, **I**nners, **O**uter, **eX**ternal) - by using QGIS with SATVIEW shapefile.
- Paths and path flows from .TFL file

– Processing

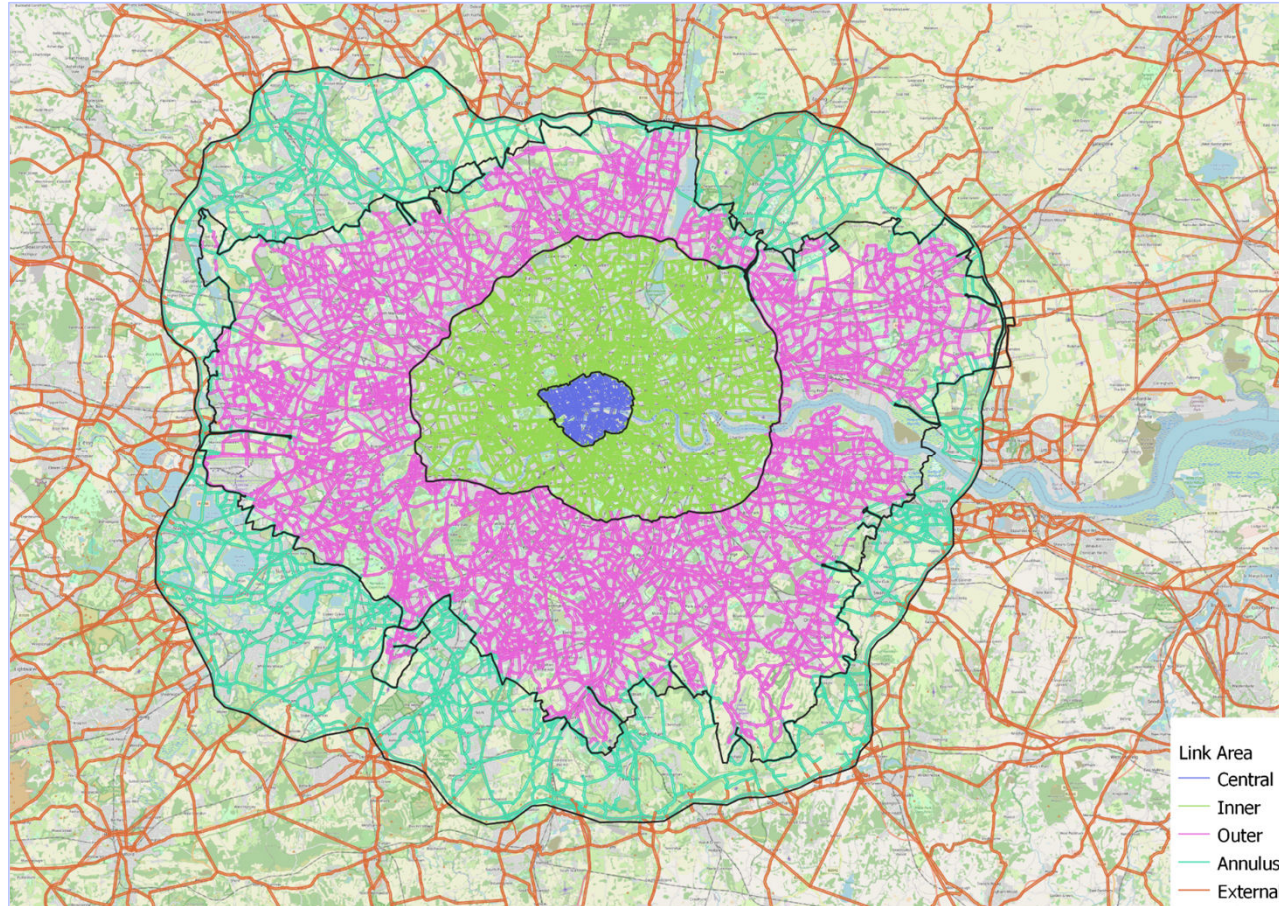
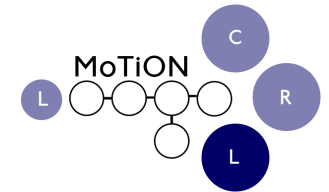
- Go through paths and calculate km travelled on each path link
- Categorise using Origin Zone, Destination Zone, link Area and User Class and sum totals

– Outputs

- Summary totals to CSV file



LoHAM CIOX Links by Area



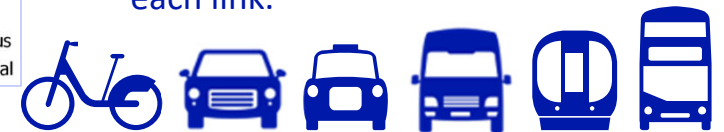
Use QGIS to classify links by Area

LoHAM Network User Links
shapefile from SATVIEW

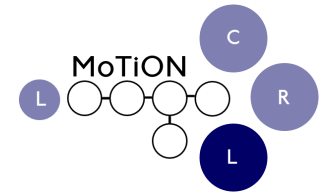
User Link lengths via SATDB (part of
the “standard outputs” produced
with another Python Script and
included in TfL Model Pack GIS
folder)

CIOX Area Boundaries shapefile
from TfL (included in Model Pack GIS
folder)

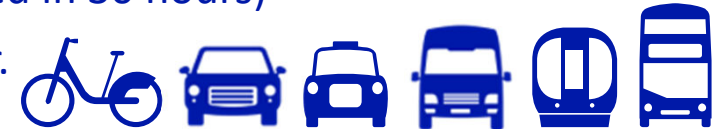
Use QGIS toolbox “Join Attributes by
Location” to associate CIOX area to
each link.



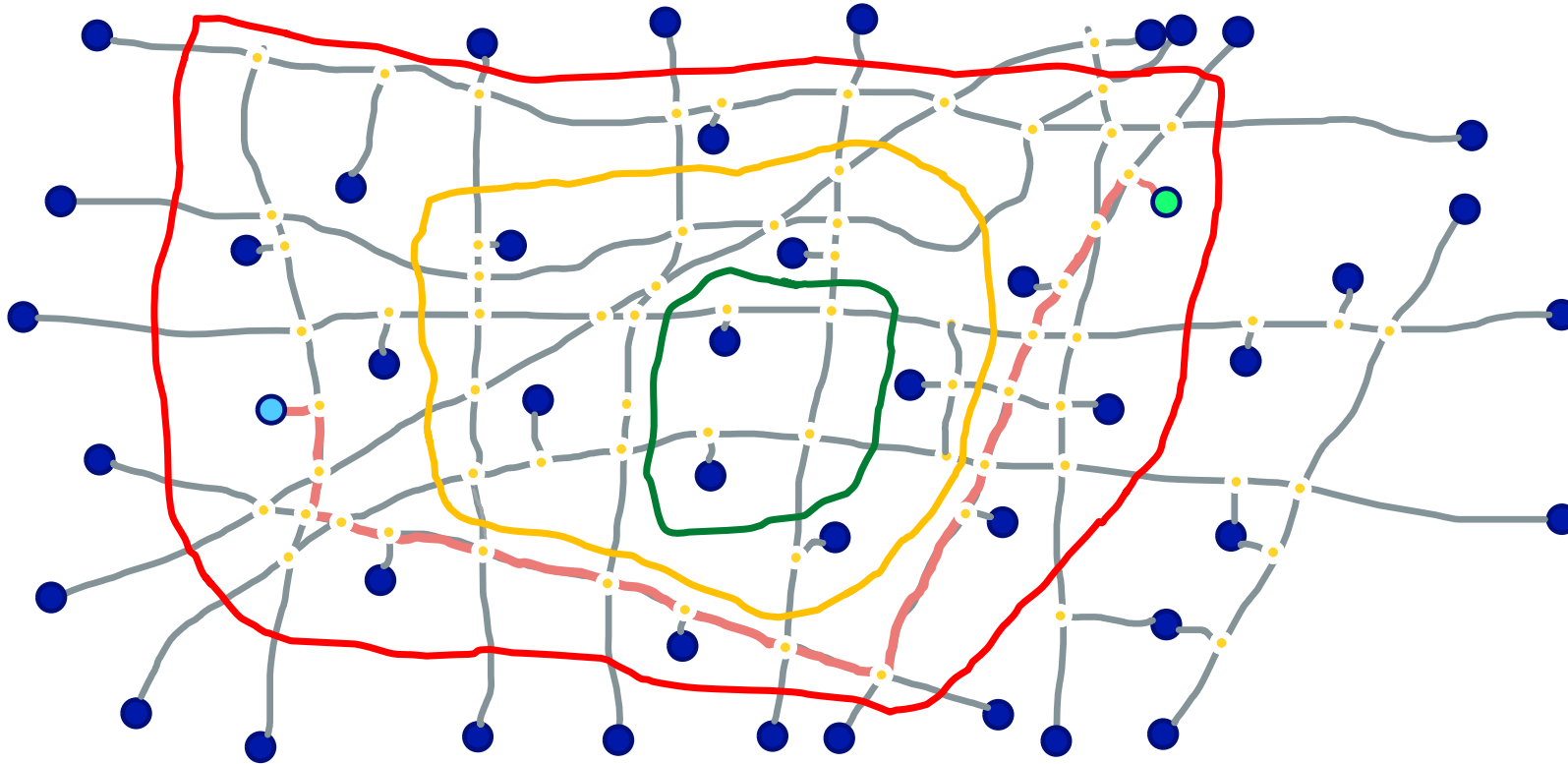
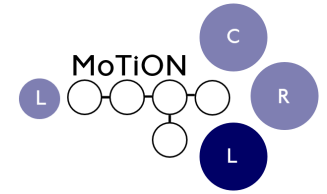
Analysing the TFL file



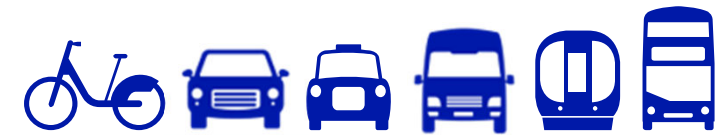
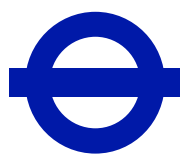
- Bespoke Python code, Python 3.9.11 (64 bit) via Anaconda.
- For Trip Length Analysis is over 500 lines of code.
- Simple logic
 - Read data on each path from .TFL file (2 lines per path)
 - Go through each path link by link
 - Work out what areas (Central, Inner, Outer, Annulus, External) the path goes through.
 - Calculate the length of the path covered in each area.
 - Classify the path and tally up according to areas, lengths and user class flows.
 - Calculate sums and trip length distributions (0 to 50+ km in 1 km blocks).
 - When all processed output the mean, minimum, maximum and distributions to a CSV file.
- Quite slow to run. (6UC LoHAM with 380,178,215 paths processed in 30 hours)
- Runs more than three times quicker if you use the PyPy compiler.



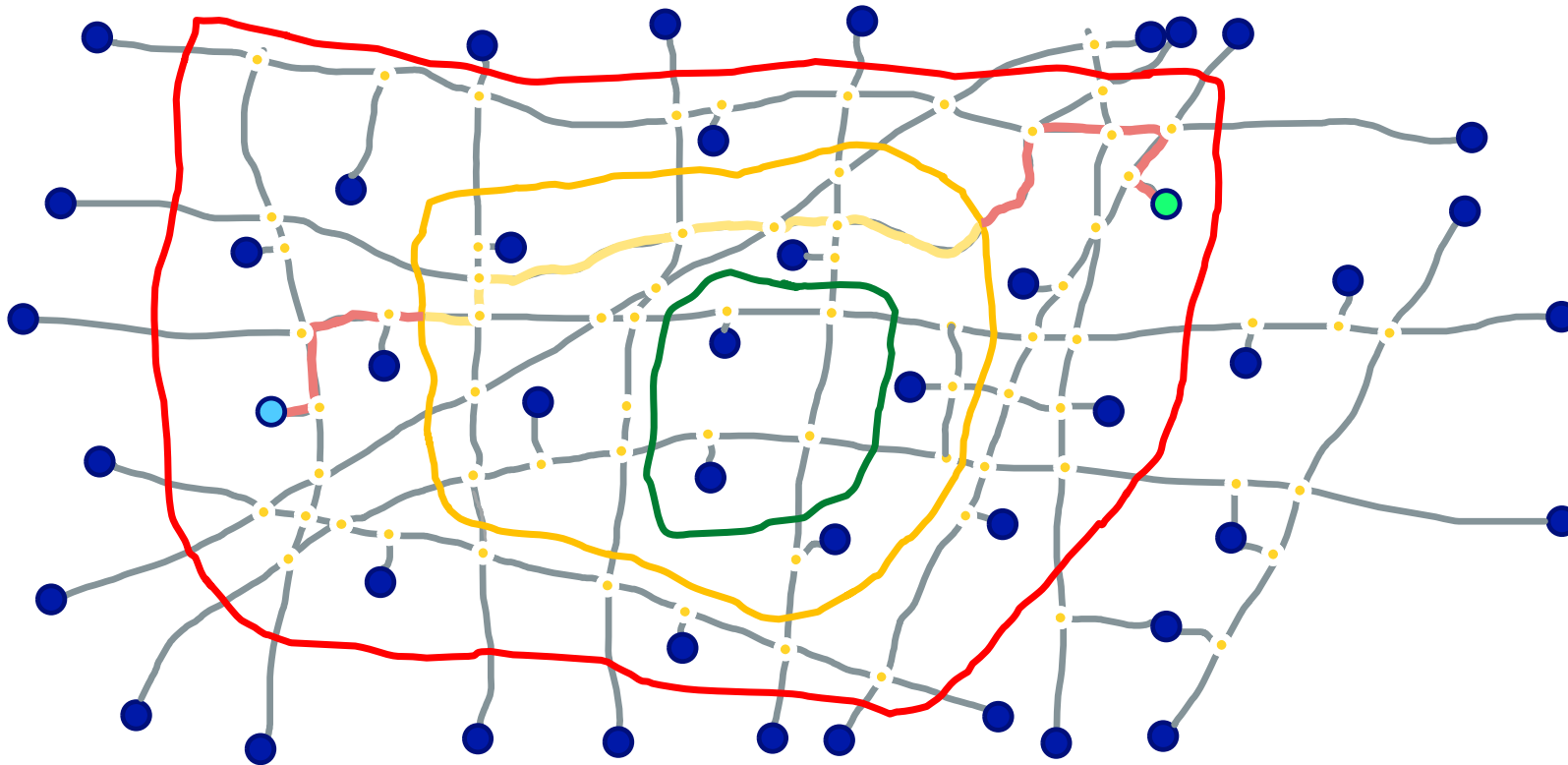
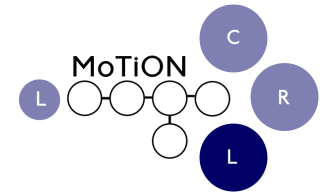
Categorising paths



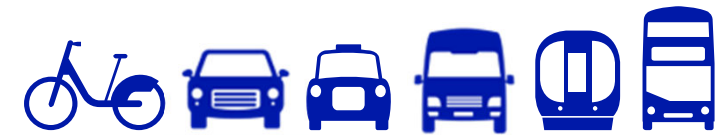
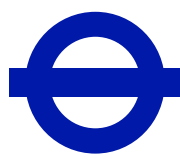
- Zone
- Junction
- Road
- Central
- Inner
- Outer



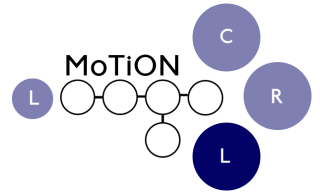
Categorising paths



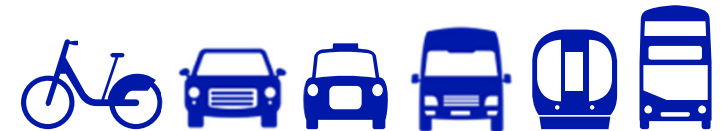
- Zone
- Junction
- Road
- Central
- Inner
- Outer



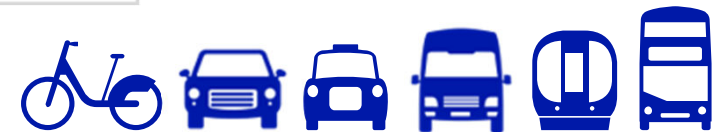
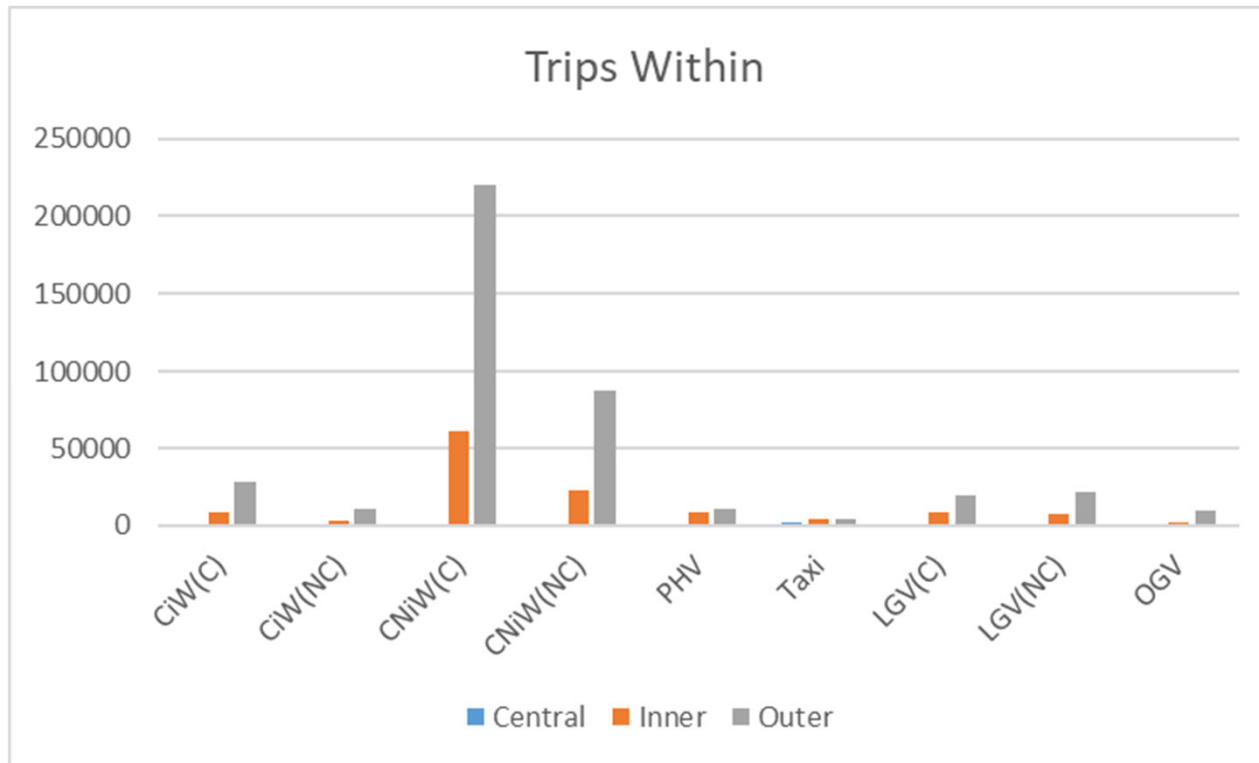
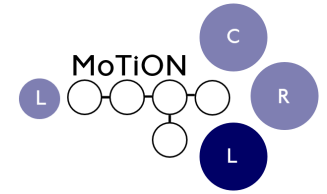
Summary outputs produced for each Path Category & User Class



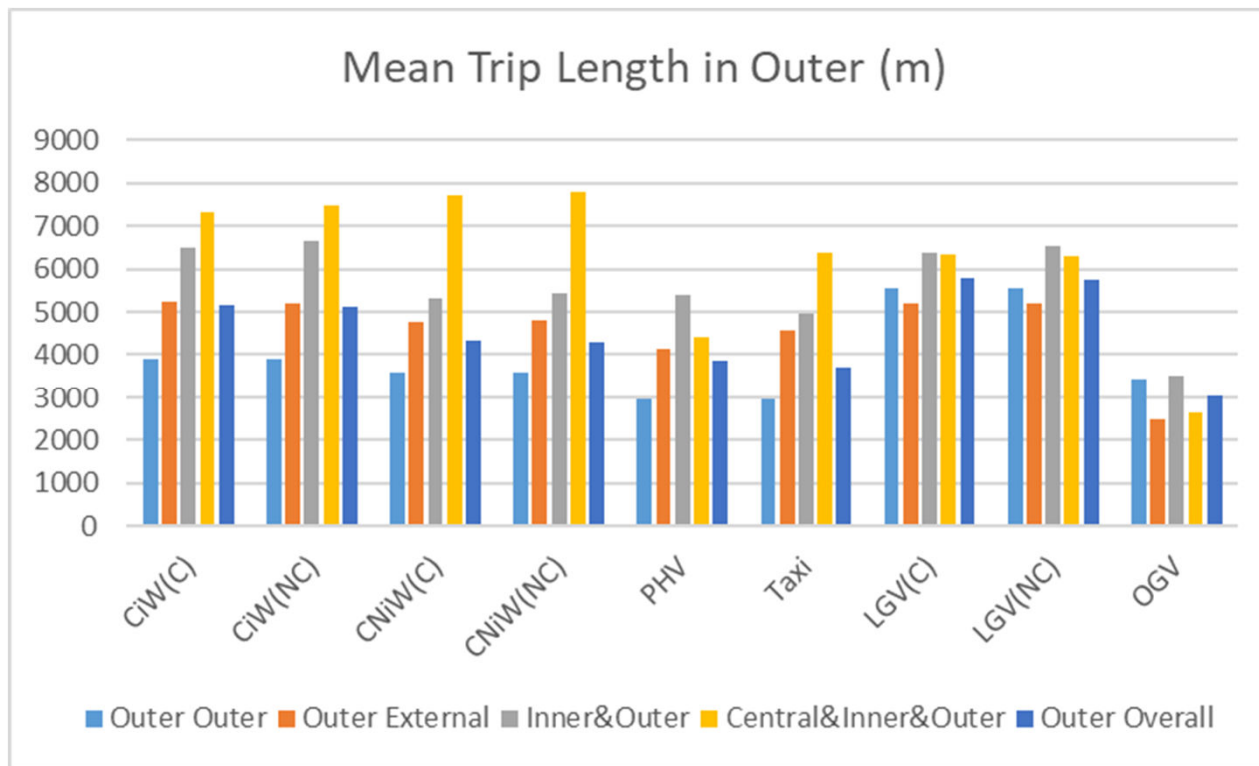
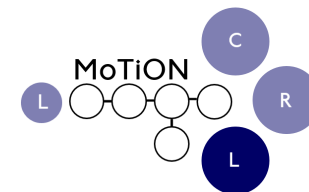
- For each user class and area
 - Trips
 - Mean, Minimum and Maximum trip length (m)
 - Trip length distribution
- For each path type area regime
 - Central, Inner, Outer
 - Outer (Outer, External)
 - Central and Inner (Central, Inner)
 - Central, Inner and Outer (Central, Inner, Outer)



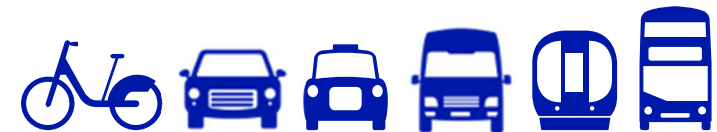
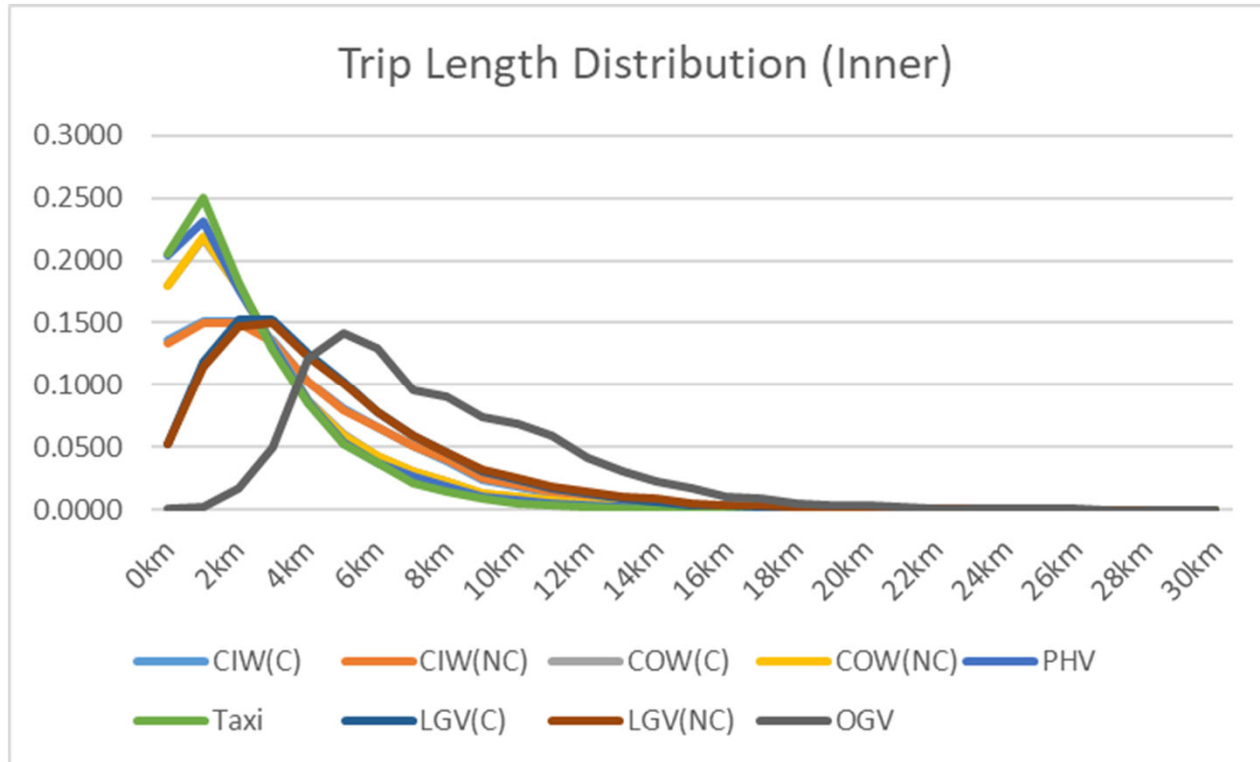
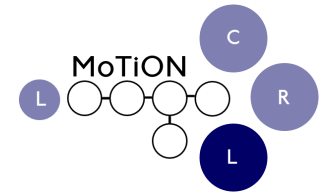
Example summary output

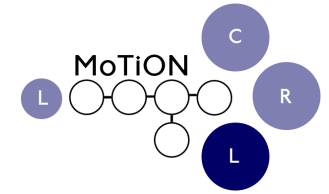


Example summary output



Example summary output





SATPATH Analysis - Conclusions

- **Developing a set of bespoke Python scripts to analyse LoHAM assignment results via SATPATH**
- **It allows analysis to be carried out that can't be done via standard SATURN outputs**
- **Will give a greater understanding of the routing effects on the highway network of any modelled schemes and changes in demand, e.g. Planning forecast vs Hybrid; Changes to areas of interest.**
 - By Area (e.g. Borough, Region, etc)
 - By User Class (e.g. Car In Work, Car Not In Work, Goods Vehicle, Zero Emissions Vehicle, Taxi, etc)
- **If you would like any of our Python Scripts then just request them. They are bespoke scripts for answering specific questions, but they may help with writing your own software for analysing your own model's .TFL outputs.**



SATURN ANNUAL UGM – MARCH 2024

Contact us

hams@tfl.gov.uk



EVERY JOURNEY MATTERS

thank you



UNIVERSITY OF LEEDS

FVVB Ltd

 **AtkinsRéalis**