MaaS Global – Transport emission estimation 2019

Method for calculating transport emissions by Whim customers

Executive summary

The 'Mobility as a Service' offer of 'MaaS Global' in the form of the app 'Whim' not only aims to provide the user with a pleasant experience of moving around but also to encourage them to move more consciously in the sense of sustainability. With its Whim solution, MaaS Global aims to provide an attractive alternative for car ownership.

Based on MaaS Global's commitment to provide sustainable mobility, its full carbon footprint was calculated for the first time for 2019 according to the GHG Protocol, together with an external consultancy Positive Impact. To support the footprint calculations for the trips taken by Whim customers, a 'Sustainable Mobility API' was developed by 'MaaS Global'. It is a tool to calculate carbon dioxide (equivalent) emissions for a given transport trip and provide a simple interpretation of the result. This project is intended to be used to help raise awareness about the environmental cost of transportation choices, so people can make better-informed decisions. Beyond MaaS Global's CO2 calculations, the 'Sustainability Mobility API' is also an invitation for the open source and MaaS community to jointly develop further the industry's CO2 impact.

This methodology statement was developed to support the emission estimation of Whim passengers in 2019. It presents the rationale behind the method adopted for calculating emissions arising from different passenger transport modes.

CO2 emissions

The data used to calculate the CO2 emissions comes directly from users who buy individual “tickets” through the app while providing us with details about their location and destination. When the origin and destination information is lacking, such as for Whim Wheel bookings, we use statistical models as a basis. The sections below describe how we approached specific cases for our main user groups.

‘Whim Wheel’ bookings

Users can purchase public transport tickets through the so-called ‘Whim Wheel’. Those bookings are not connected to any itinerary and hence have neither mode nor distance information. In order to estimate carbon emissions regardless, we calculate the distribution of transport modes from public transport bookings with itinerary information and use the same distribution for those bookings without. For the distances, we approximate daily averages by mode and use those to interpolate distance information where they are missing.

All those calculations are done separately by plan type as their user bases and consequently uses of transport modes differ.

Pay-as-you-go users

All bookings of the 'pay as you go' users (PAYG) are used for direct carbon estimation, where necessary with mode and distance interpolate as per the above method. This user group represents about 20% of the app users.

Urban / Urban 30 users

The current users of the 'Urban30' subscription cannot be measured directly for all of their trips, because they can use public transport freely without having to enter their start and destination in the app so that in their case no distances can be calculated. However, we have more detailed travel data for ‘Urban’ subscribers prior to April 2019.

In order to estimate the carbon footprint of current ‘Urban 30’ users, we calculated some population statistics for the 2018-2019 Urban users and extrapolated those statistics against subscriber counts for post-April 2019 Urban 30 users. This was a best-effort approach to estimating the carbon footprint of these users and is not without tradeoffs.

Emissions calculation

The CO2 emissions are calculated with the [‘Sustainable Mobility API’](). The ‘Sustainable Mobility API’-project consists of a Python library and HTTP API for estimating the environmental impact of personal mobility. With the help of that Python library, it is possible to calculate CO2 emissions for a transport trip based on the distance and mode.

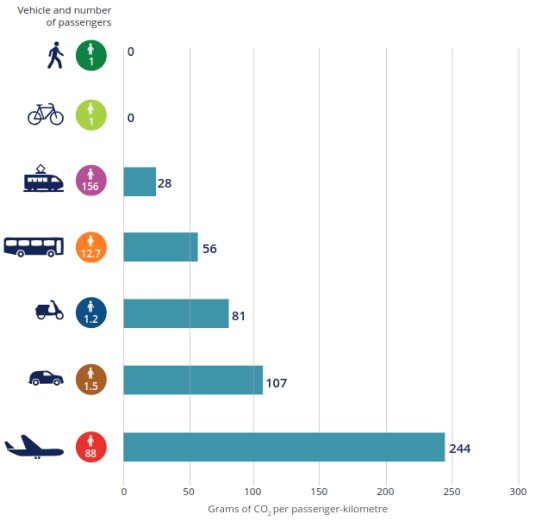
The following transport modes are distinguished:

- light rail,
- small car,
- large car,
- e-scooter, and
- bus.

In the Python library, the emissions per vehicle kilometre and the occupancy rate are stored for each mode.

The emission factors and occupancy rates are featured in the [EU 2016 TERM report]():

Figure 5.2 **Specific CO_2 emissions at average occupancy for various transport modes, 2014**



Note: The addition of more passengers (the loading factor) results in fuel consumption — and hence also CO_2 emissions — penalties as the vehicle becomes heavier, but the final figure in grams of CO_2 per passenger is obviously lower. This effect is significant for CO_2 emissions from passenger cars and two-wheelers. For other vehicles, which are generally much heavier, this effect is insignificant. The inland ship emission factor is estimated to be 240 g CO_2/km, but data availability is still not comparable with that of other modes.

Own estimations based on the UNFCCC and the EU Greenhouse Gas Monitoring Mechanism (see TERM 02 indicator) and total activity (pkm) from DG MOVE pocketbook 2016 (DGMOVE, 2016). Rail emissions include those from diesel and electricity powered trains at European level compiled by the International Union of Railways (UIC). PRIMES is used for aviation CO_2 emissions. Linear interpolation of the PRIMES data, available in five-year steps, is needed for the intermediate years.

It is worth mentioning that within the library the emissions are used per vehicle-kilometre and not per person-kilometre. This is the case because the occupancy rate should be included in the equation as a variable adjustable for the user.

The emissions are then calculated as follows:

*Estimated CO2 emission [g] = CO2 per vehicle km [g/km] * distance [km] / occupancy [-]*