With services like Google Maps, finding the fastest route from A to B has become quick, cheap, and easy. Not just for driving but walking, cycling and public transport too. But in the field of transport studies, we often want not only a single route, but thousands or millions of routes. This is where we hit a problem for services such as Google or the [Open Route Service](https://openrouteservice.org/), that usually only allow a limited number of free routes per day (typically around 1000). So routing is either very time consuming or expensive. Another problem is that we may not be interested in the current travel options, but how those options may change in the future. Such as, after a new bridge has been built or a new bus timetable has been introduced. Therefore, researchers can find it useful to run their own routing services where they have more control and can produce as many routes as necessary.

The aim of the OpenTripPlanner for R package to make it easy to set up and use a multimodal trip planner in R. [OpenTripPlanner](https://www.opentripplanner.org/) (OTP) is an open-source multimodal trip planner written in Java. It uses OpenStreetMap for walking, cycling, and driving directions and uses [GTFS](https://developers.google.com/transit/gtfs) files for public timetables. The R package makes it easy to set up and use OpenTripPlanner on your local computer or connect to a server running OpenTripPlanner.

**Prerequisites and Installation**

OTP requires Java 8 specifically, not any earlier or later versions. Once you have Java installed, install the package in the normal way.

install.packages("opentripplanner")

library(opentripplanner)

**Building your first trip planner**

The package comes with demo data for the Isle of Wight, which can be downloaded and built using a few simple commands. For a full explanation, including how to customize your setup and build OTP for other places.

path\_data <- file.path(tempdir(), "OTP") # Make a folder to store the data

dir.create(path\_data)

path\_otp <- otp\_dl\_jar(path\_data) # Download the OTP

otp\_dl\_demo(path\_data) # Download the demo data

log1 <- otp\_build\_graph(otp = path\_otp,

dir = path\_data) # Built the demo data into a graph

log2 <- otp\_setup(otp = path\_otp,

dir = path\_data) # Start OTP

The whole process should only take a few minutes to run, and then the OTP web interface will launch in your browser. You can use the web interface just like any other journey planner, but to get the results into R you need to connect R to the OTP and then make your requests through R.

otpcon <- otp\_connect() # Connect R to OTP

route <- otp\_plan(otpcon, # Route between two lon/lat coordinates

fromPlace = c(-1.17502, 50.64590),

toPlace = c(-1.15339, 50.72266),

mode = c("WALK","TRANSIT"))

If you have the tmap package installed, you can view the route within RStudio. Notice the use of sf::st\_zm(route). This is because tmap does not currently support XYZ coordinates.

library(tmap) # Load the tmap package

tmap\_mode("view") # Set tmap to interactive viewing

qtm(sf::st\_zm(route), lines.lwd = 3,

lines.col = "mode") # Plot the route on a map

The package supports many more features such as batch routing, isochrones, and geocoding, There are also instructions on how to set up OTP for any area of the world.