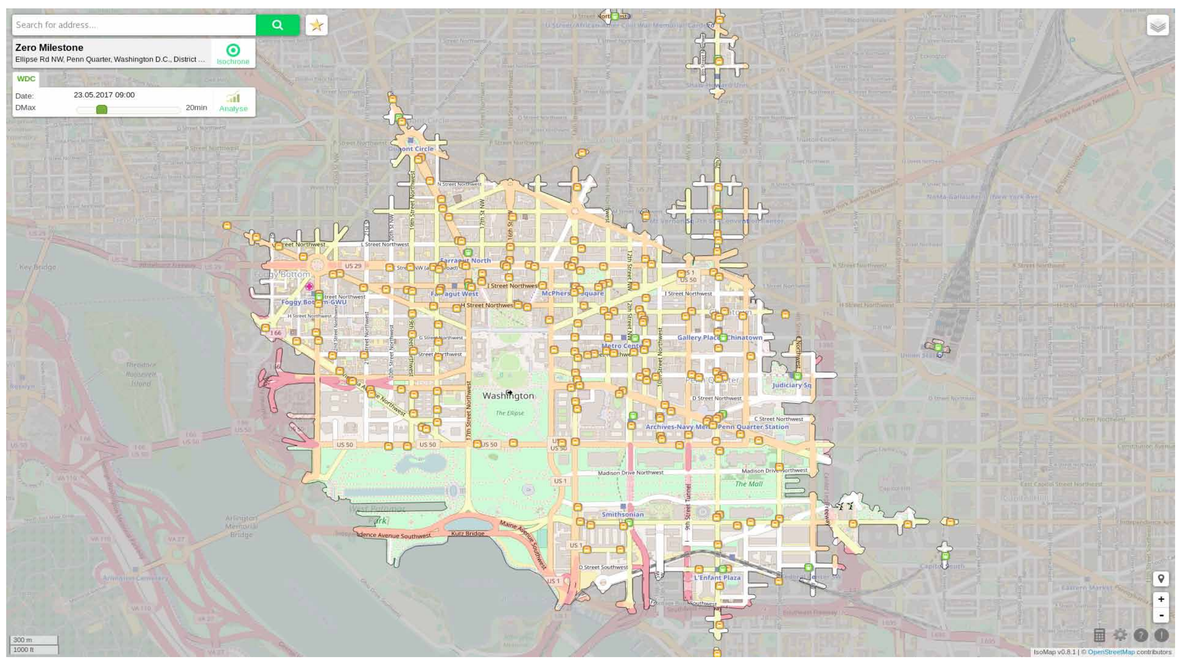
Computing Isochrones in Multimodal Spatial Networks using Tile Regions

Isochrones in multimodal spatial networks are computed in a Dijkstra-like fashion.

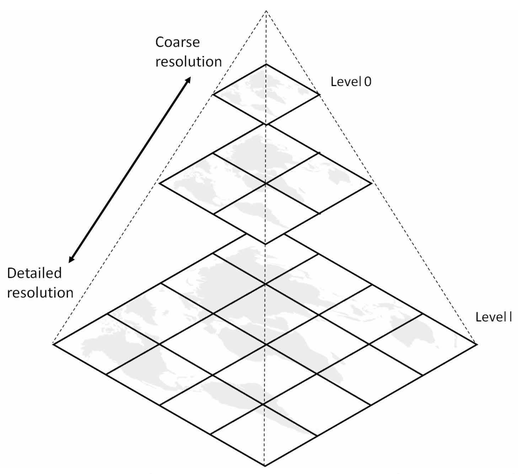


Computation time heavily depends on how data is loaded from the underlying database.

- For small networks, the entire network is first loaded in main memory (MDijkstra).
- Large networks that do not fit into it are calculated using vertex-by-vertex queries (Mine(X)).
- We propose an approach in between and to load data using chunks (MineT(X)) to find a good trade-off between memory usage and runtime.

To optimize database index usage, the chunks we propose are of rectangular shape and known (as tiles) from geo sciences. Chunk sizes can easily be varied by changing the zoom level of a tile to allow for performance improvement.

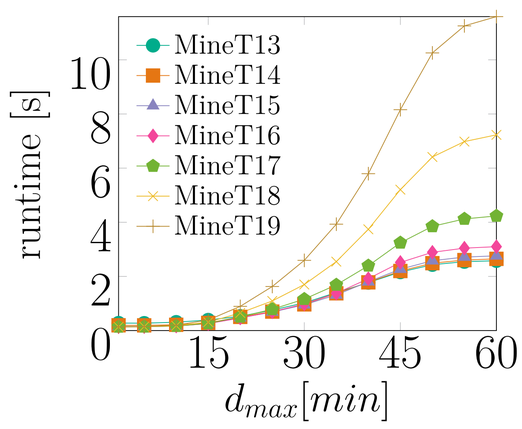
Larger regions (smaller z) load more data at once and are preferable for server machines. With increasing z the regions get smaller and therefore work better on desktop hardware (with less RAM and slower CPUs).



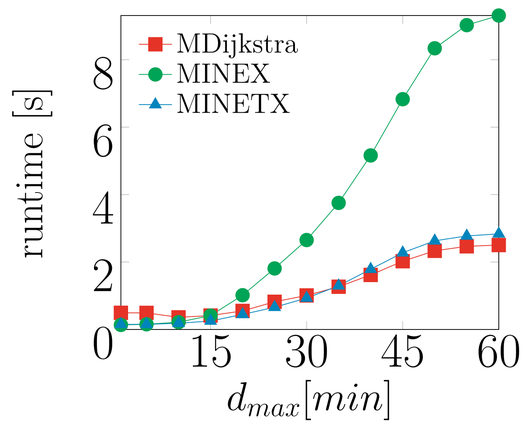
Source: R. García, J.P. de Castro, E. Verdú, M.J. Verdú and L.M. Regueras (2012). Web Map Tile Services for Spatial Data Infrastructures: Management and Optimization, Cartography - A Tool for Spatial Analysis

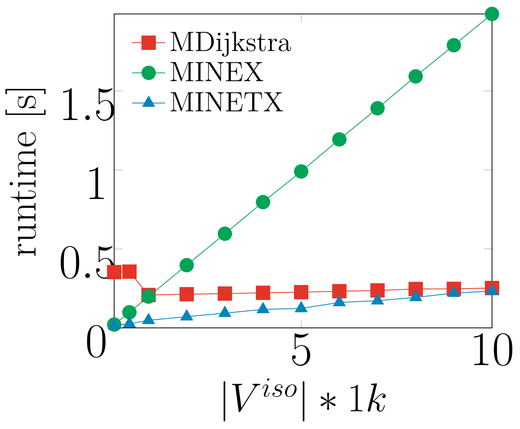
Experimental Evaluation (WDC dataset)

The optimal zoom level for each dataset can be determined with some testing and experience.

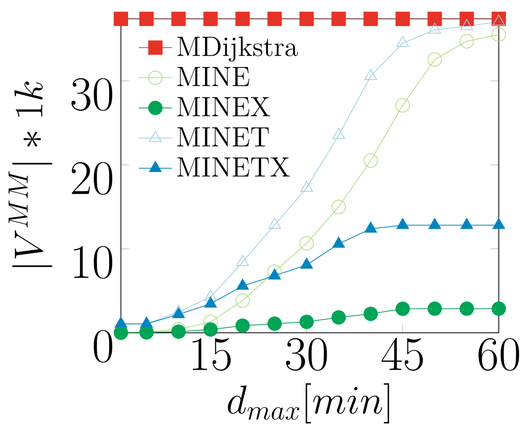


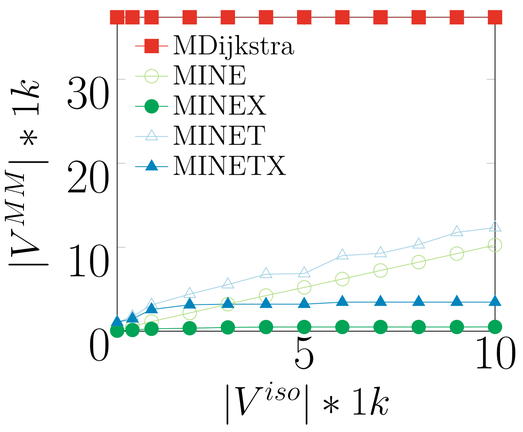
The following figures display the runtime and memory consumption for the calculation of isochrones with zoom level at 15.





Since our algorithm allows for vertex expiration, the memory footprint is kept small.





The experimental evaluation shows that the new algorithm clearly outperforms previous competitive approaches such as Mine and MineX.

Further information is available at
https://dbis-isochrone.uibk.ac.at