Запрос в PostGis

Данный запрос получает на вход мультилинию(трек) в формате wkt и находит длины совпадений данной мультилинии с транспортными путями в разрезе типов транспортных путей.

Делается это так:

строится оболочка (буффер) вокруг каждой из частей мультилинии заданной ширины (десятки метров), в результате чего получается мультиполтгон и он пересекается геометрически с транспортными путями из базы. Начало и конец пересечений поточечно проецируется на исходную мультилинию, эти точки затем соединяются, в результате получается проекция пересечений на трек. Теперь необходимо найти длины этих проекций. Здесь немного сложнее. Проекции могут пересекаться между собой и поэтому складывать их длины неправильно, а если пересекать их между собой, то из-за арифметической точности числа могут возникать ошибки, поэтому было принято решение взять небольшой буффер вокруг каждой проекции и объединить их в один мультиполигон, затем длина всей проекции вычисляется как “длина периметра буффера / 2“.

SET work\_mem TO '100MB';

DROP TABLE IF EXISTS cashed\_mline, projections;

– считать строку из %s в проекционной системе 4326 (Lat, Lon)

CREATE TEMP TABLE cashed\_mline AS

SELECT

ST\_GeomFromText('%s', 4326)

AS mline;

– разбить мультилинию на составные части, каждую часть пронумировать через path, вокруг каждой линии сдлать буффер

CREATE TEMP TABLE geoms AS

SELECT

-- ST\_AsText(ST\_Transform(

ST\_Transform((gdump).geom, 900913)

-- ,4326))

as line,

(gdump).path as path,

ST\_Transform(ST\_Buffer((gdump).geom::geography, 50)::geometry, 900913) AS poly

FROM (

SELECT

ST\_Dump(mline) AS gdump

FROM cashed\_mline

) as dumped;

– пересечь транспортные пути с полигонами

CREATE TEMP TABLE raw\_intersects AS

SELECT

geoms.path as path,

-- - ST\_AsText(ST\_Transform(

ST\_Intersection(ppol.way, geoms.poly)

-- ,4326))

AS inter,

ppol.railway as name

FROM public.planet\_osm\_line AS ppol

INNER JOIN geoms

ON ppol.railway in ('rail', 'subway',

'tram', 'monorail', 'light\_rail', 'narrow\_gauge')

AND ST\_Intersects(ppol.way, geoms.poly)

UNION

SELECT

geoms.path,

ST\_Intersection(ppol.way, geoms.poly),

ppol.route

FROM public.planet\_osm\_line AS ppol

INNER JOIN geoms

ON ppol.route in ('bus', 'trolleybus', 'share\_taxi')

AND ST\_Intersects(ppol.way, geoms.poly)

;

– преобразовать мультилинии в список из линий

CREATE TEMP TABLE intersects AS

SELECT

path,

name,

CASE

WHEN ST\_GeometryType(inter) = 'ST\_MultiLineString' THEN

(ST\_Dump(inter)).geom

ELSE

inter

END AS inter

FROM raw\_intersects

WHERE ST\_GeometryType(inter) in ('ST\_LineString', 'ST\_MultiLineString');

DROP TABLE raw\_intersects;

– вычисляем начало и конец каждого пересечения и проецируем на наш трек

CREATE TEMP TABLE start\_end AS

SELECT

ints.path,

name,

ST\_LineLocatePoint(line, ST\_StartPoint(inter)) as p1,

ST\_LineLocatePoint(line, ST\_EndPoint(inter)) as p2

FROM intersects as ints

INNER JOIN geoms

ON geoms.path = ints.path;

DROP TABLE intersects;

– соединить начало и конец спроецированных точек в линии

CREATE TEMP TABLE sub\_lines AS

SELECT

s\_e.path,

name,

ST\_LineSubstring(line, LEAST(p1, p2), GREATEST(p1, p2)) AS sub\_line

FROM start\_end as s\_e

INNER JOIN geoms

ON geoms.path = s\_e.path;

DROP TABLE start\_end, geoms;

– делаем буфер вокруг проекций и объединяем эти буферы

CREATE TEMP TABLE projections AS

SELECT

path,

name,

-- ST\_AsText(ST\_Transform(

ST\_Union(ST\_Buffer(sub\_line, 0.01))

-- ,4326))

as proj

FROM sub\_lines

GROUP BY path, name;

DROP TABLE sub\_lines;

– вычисляем длину самого трека и каждого из перечений в разрезе типов путей

SELECT

'track' as name,

ST\_Length2D\_Spheroid(mline,

'SPHEROID["GRS 1980",6378137,298.257222101]')/1000 as length

FROM cashed\_mline

UNION

SELECT

name,

sum(ST\_Length2D\_Spheroid(ST\_Transform(proj, 4326),

'SPHEROID["GRS 1980",6378137,298.257222101]'))/2000

FROM projections

group by name

order by length desc