

Topological Data Analysis – group project

Sensors

Nejc Kišek, Žan Klaneček

June 7, 2018

1 Introduction

In this project we are trying to find optimal parameters for a sensor network which covers the whole surface of the earth. The network is determined by two parameters:

- r , which is the distance over which the sensors can communicate with each other – each sensor can communicate with other, if they are at most r away.
- R , which is the radius of the surrounding area in shape of a circle where the sensor can gather data.

Given the coordinates of each sensor, we had to determine the lowest possible parameters r and R so that:

- the sensor network is connected,
- the sensor network covers the whole sphere (surface of the Earth).

Furthermore, once the parameters r and R are established, obsolete sensors should be removed. These are sensors whose removal would not change the connectivity and coverage of the sensor network. Lastly, a data generator that distributes n points evenly on the surface of a sphere will be described. With this generator, parameters r and R of the sensor network will be as small as possible.

2 Topological solution

In this section theoretical background for our problem is described. Firstly, if we want to achieve sensor network connectivity (all sensors are connected), we should have a look at Vietoris-Rips complex. Such r (as small as possible) should be determined, that Vietoris-Rips graph on a given distribution of sensors is connected (i.e. has one component). The properties of Vietoris-Rips are described with following relations:

- sensors: $S_i = (r_i, \phi_i, \theta_i)$,
- sensor connections $\{S_i, S_j\} \subset S; d(S_i, S_j) \leq 2r$,
- $F \subset S$ is a simplex in $VR_r(S)$, if $\text{diam } F \leq 2r$,

where r is yet unknown parameter of sensor network and $d(S_i, S_j)$ is Euclidean distance between two sensors. Secondly, if we want to achieve coverage of the entire sphere, Čech complex characteristics will come in handy:

- sensors: $S_i = (r_i, \phi_i, \theta_i)$,
- $B_R(x)$ closed ball with radius R around x ,
- $C_R = \{\sigma \subset S, \cap_{x \in \sigma} B_R(x) \neq \emptyset\}$.

In this case, such R (as small as possible) should be determined, so that the Euler characteristic of the Čech complex is the same as that of a sphere. However, in practice, we rather determined first two Betti numbers which should be equal to 1 and 0, meaning there is only one component with no holes.

3 Distribution of sensors

In Figure 1 the pre-given distribution of sensors is visualized. The major difference between distribution of points in sensors01 and sensors02 is around the poles.

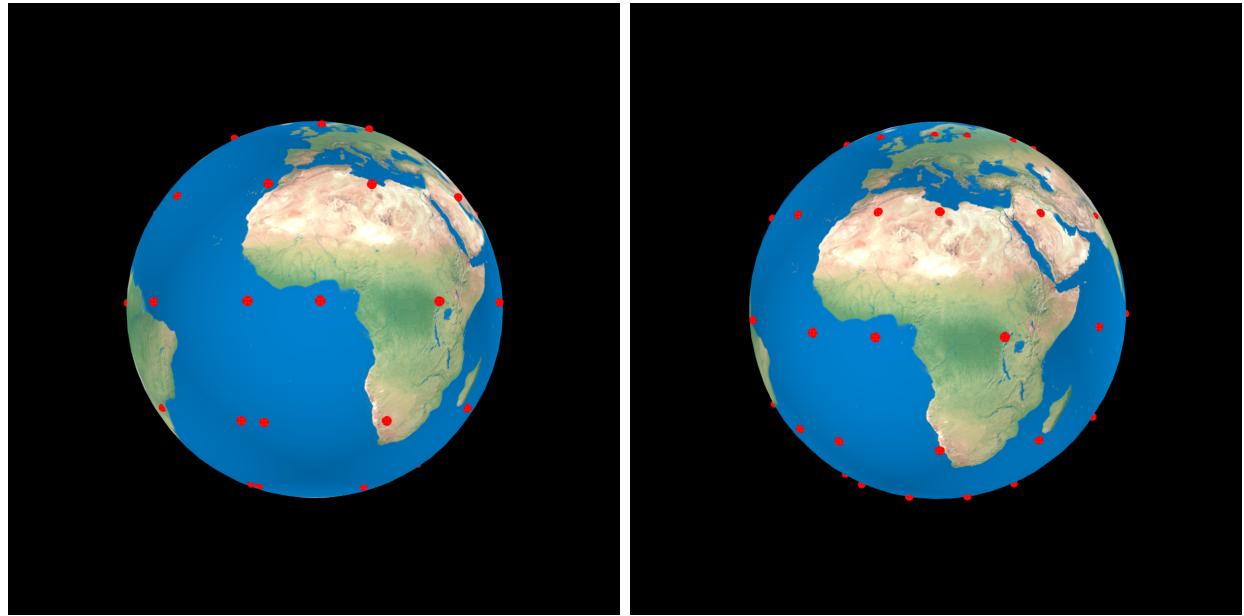


Figure 1: Locations of the sensors

4 Connectivity

TODO: VR complex, increasing r until we get 1 component.

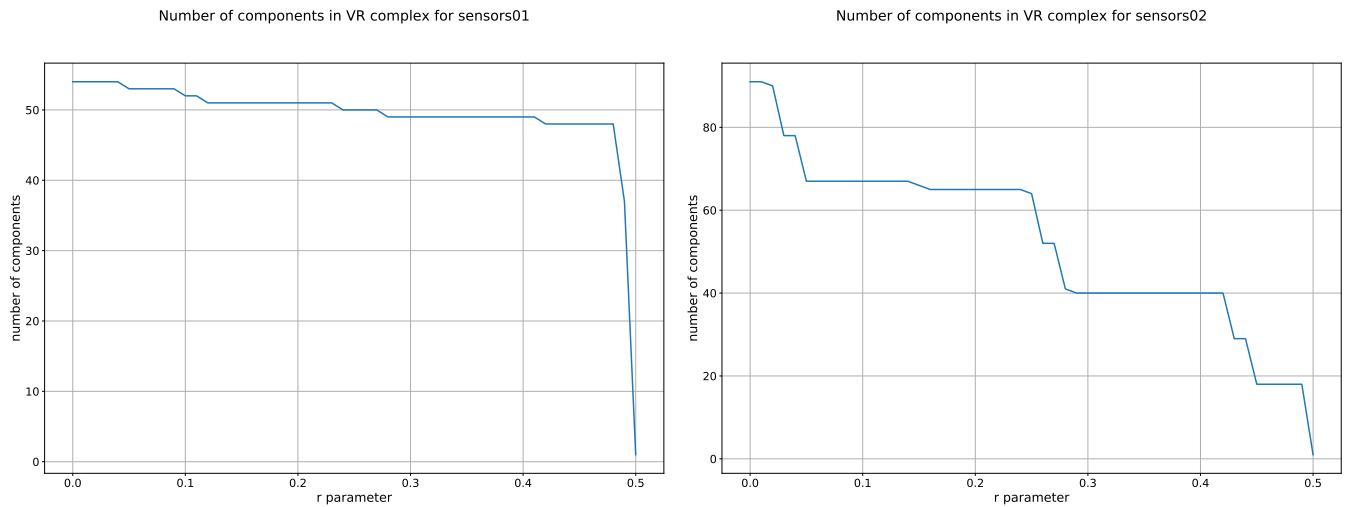


Figure 2: Number of components as we increase the r parameter

TODO: malo komentarja na graf

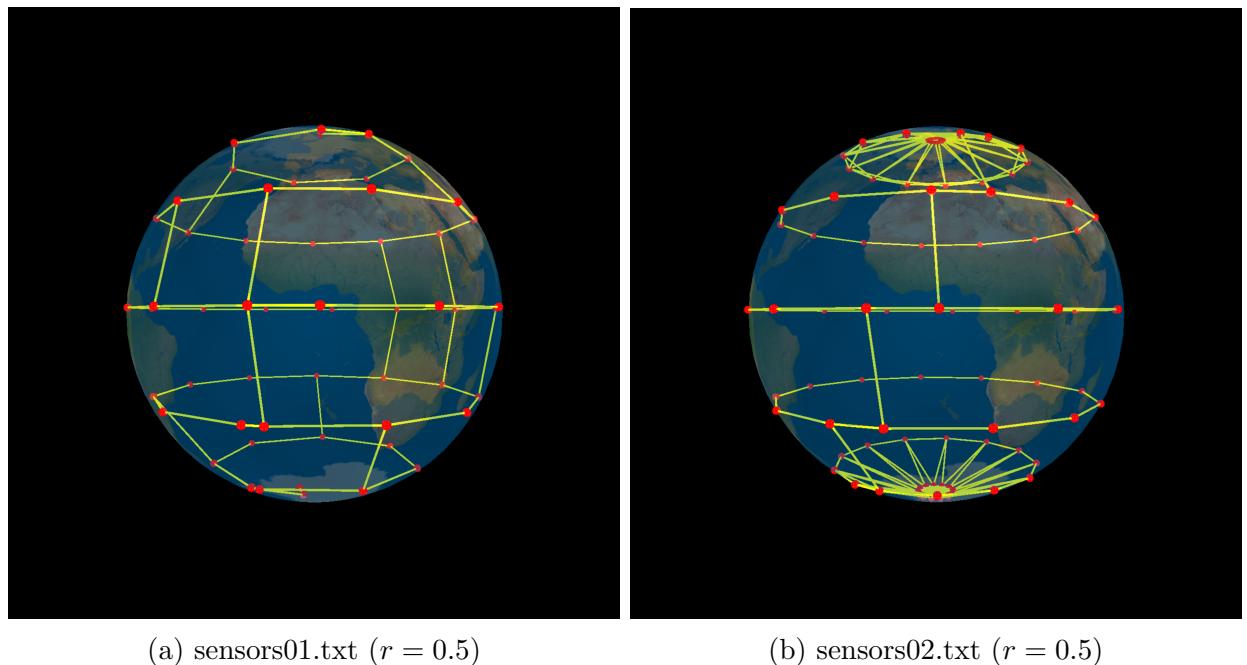


Figure 3: Connections between sensors

5 Coverage

TODO: Čech complex, increasing r until Betti number b_1 is 0 (zero cycles/holes). plots (homology, barcode), 3d image

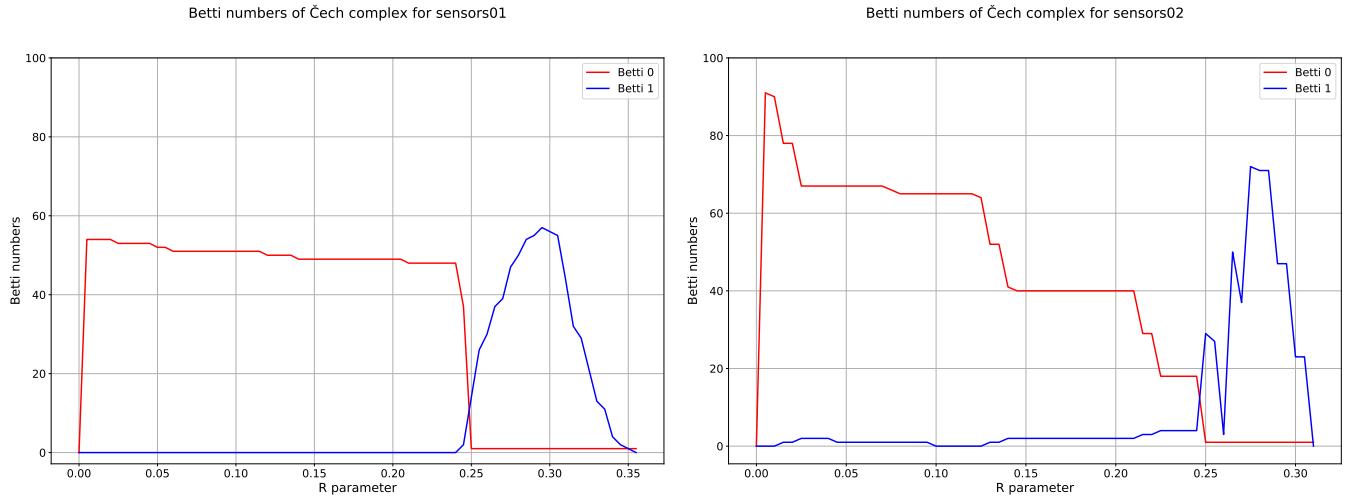


Figure 4: b_0 and b_1 as we increase the R parameter

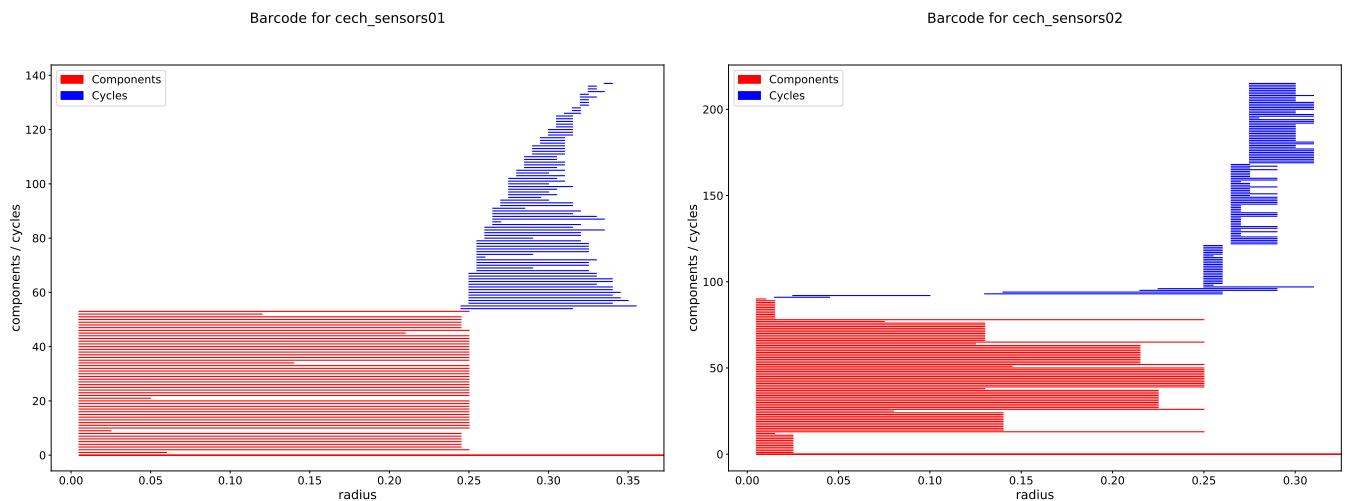
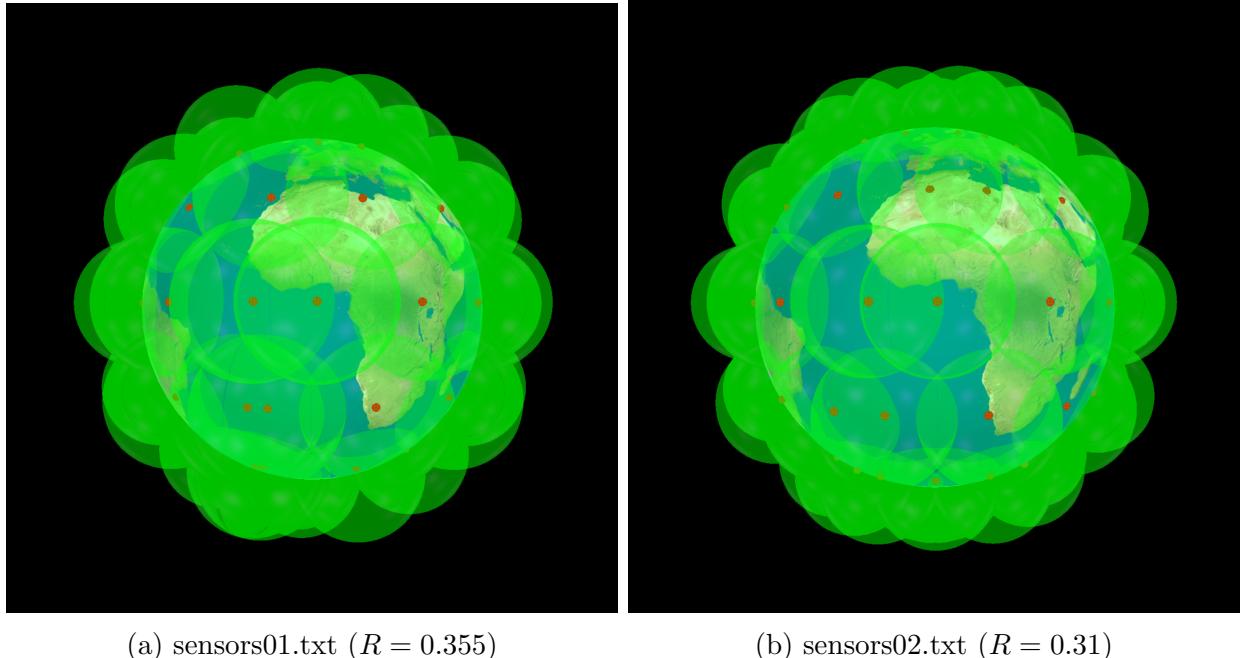


Figure 5: Barcodes for the Čech complex

TODO: malo komentarja na graf



(a) sensors01.txt ($R = 0.355$)

(b) sensors02.txt ($R = 0.31$)

Figure 6: Connections between sensors

6 Data generator

Lastly, let's have a look at the data generator that distributes n points evenly on the surface of a sphere. In order to meet this requirement we should look on points as electrons that are bound on the surface (i.e. in spherical coordinates have fixed radius). The natural state would be the state with the lowest electrostatic-potential energy V (eq. (2)). This is due to the fact that electrons repel from each other according to Coulombs inverse square law (eq. (1))

$$|\mathbf{F}_{ij}| = k_e \frac{e^2}{|\mathbf{r}_i - \mathbf{r}_j|^2}, \quad (1)$$

$$V = \sum_{i \neq j} V_{ij} \propto \sum_{i \neq j} \frac{1}{|\mathbf{r}_i - \mathbf{r}_j|}, \quad (2)$$

where k_e is Coulombs constant, e charge of electron and \mathbf{r}_i local vector for $i-th$ point. It should be emphasized that potential energy was determined as sum of inversed Euclidean distances between all points.

6.1 Simulated annealing

In order to determine the minimum of electrostatic-potential energy (eq. (2)) we used algorithm for simulated annealing with some Monte-Carlo modifications. At the beginning we randomly distributed n points on the surface of a sphere with radius r and selected the initial temperature of the system T . In each iteration a random point was chosen (one point was fixed, so $n - 1$ available) and moved it according to Gaussian distribution with appropriate parameters ($\mu = 0$ and small enough σ). After that, the difference in energy was calculated ΔE . If the energy of newly distributed points was less than before (i.e. $\Delta E < 0$) the move was accepted. Otherwise ($\Delta E \geq 0$) the move was accepted with the probability $\exp(-\frac{\Delta E}{T})$. The higher the difference in energy the lower are chances that we accept the change. Similar for the temperature, the lower the temperature the lower is the probability for accepting a bad move. If for a certain temperature T enough changes N_T were accepted, we decrease the temperature T for ΔT . Continue with another (or same) random point.

6.2 Generated datasets

In this report we will include two generated datasets, for $n = 4$ and for $n = 50$. We can see that in case of four points they distribute themselves in tetrahedral geometry and for $n = 50$ we get nice homogeneous even distribution of points.

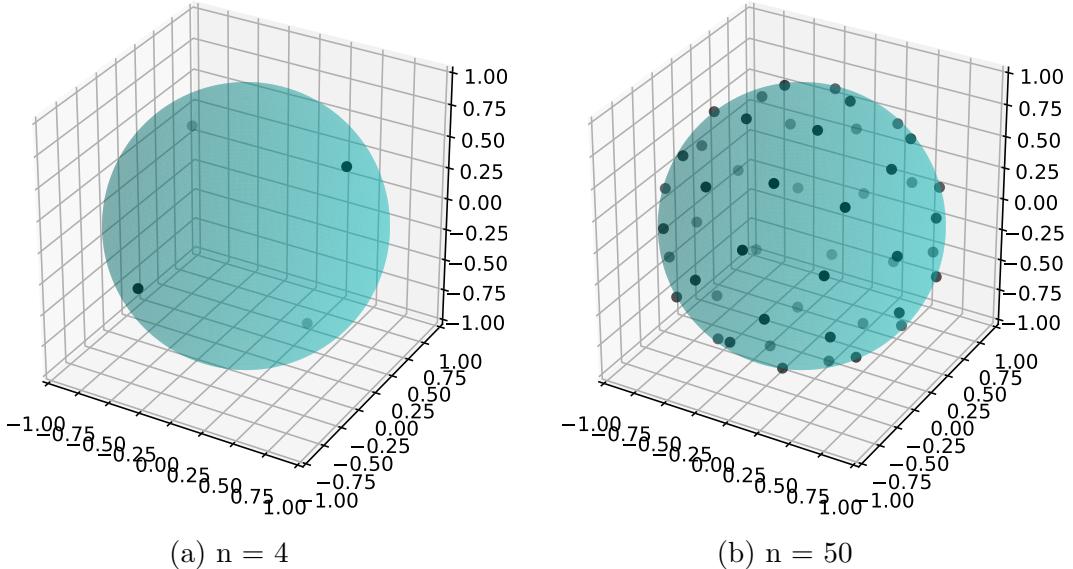


Figure 7: Even distribution of generated n points on the surface of a sphere.

7 Redundant sensors

TODO: how we remove the sensors which are not needed

8 Conclusion

TODO

9 References

References

- [1] Vietoris-Rips. https://en.wikipedia.org/wiki/Vietoris_Rips_complex (5.6.2018).
- [2] Čech-complex. https://en.wikipedia.org/wiki/Cech_complex (5.6.2018).
- [3] Dionysus. <http://www.mrzv.org/software/dionysus/> (5.6.2018).
- [4] Miniball algorithm. <https://github.com/weddige/miniball> (5.6.2018).
- [5] Vpython. <http://vpython.org/> (5.6.2018).
- [6] Lecture notes from prof. dr. Neža Mramor Kosta.