User's manual for the `Cech-scale` program

Jesús F. Espinoza[*] (`jesus.espinoza@mat.uson.mx`)
Rosalía G. Hernández (`rosalia.hdez@mat.uson.mx`)
Héctor A. Hernández (`hectorhdez@gmail.com`)
Beatriz Ramonetti-Valencia (`beatrizramonetti92@gmail.com`)

Mathematics Department - UNISON
September 2018

Introduction

The goal of this manual is to serve as a guide for the program `Cech-scale`, a complementary software for the article *A numerical approach for the filtered generalized Cech complex* (`https://arxiv.org/abs/1809.08175`).

Given a (finite) disk system:

$$M = \{D_i(c_i; r_i) \subset \mathbb{R}^d \mid r_i > 0 \ \&\ 1 \leq i \leq m \ \&\ m \geq 3\},$$

we call M a Vietoris-Rips system if $D_i \cap D_j \neq \emptyset$ for every $1 \leq i, j \leq m$; additionally, if $\bigcap_{i=1}^m D_i \neq \emptyset$, then M is called a Čech system.

For $\lambda > 0$, M_λ is defined as $M_\lambda = \{D_i(c_i; \lambda r_i) \mid D_i \in M\}$. The Vietoris-Rips scale of the system M, denoted by ν_M, is defined as

$$\nu_M = \inf\{\lambda \mid M_\lambda \text{ is a Vietoris-Rips system}\}.$$

The Čech scale of the system M, denoted by μ_M, is defined as

$$\mu_M = \inf\{\lambda \mid M_\lambda \text{ is a Čech system}\}.$$

If M is a disks system in the plane, the program `Cech-scale` calculates μ_M and the (unique) intersection point

$$\{c_M\} = \bigcap_{1 \leq i \leq m} D_i(c_i; \mu_M r_i).$$

The program can also calculate μ_M and c_M for systems with three disks, and higher dimension than 2.

[*] Responsable del proyecto

Contents

1 Getting and compiling the program

- The source code of the `Cech-scale` program is written in `C++` and is freely available in `https://github.com/gcs-unison/Cech-scale`.

> The `Cech-scale` program is distributed under the GNU General Public License v3.0.

- Once finished downloading or cloning the source code of the project, it can be compiled using cmake.

> Under a GNU/Linux distribution, the project can be compiled typing the following commands:
>
> ```
> cmake -H. -Bbuild -DCMAKE_BUILD_TYPE=Release
> cd build
> make
> cd ..
> ```

2 Executing the program

- The `Cech-scale` program reads a disk system M, given by,

  $$M = \{D_i(c_i; r_i) \subset \mathbb{R}^d \mid r_i > 0 \;\&\; 1 \leq i \leq m \;\&\; m \geq 3\}.$$

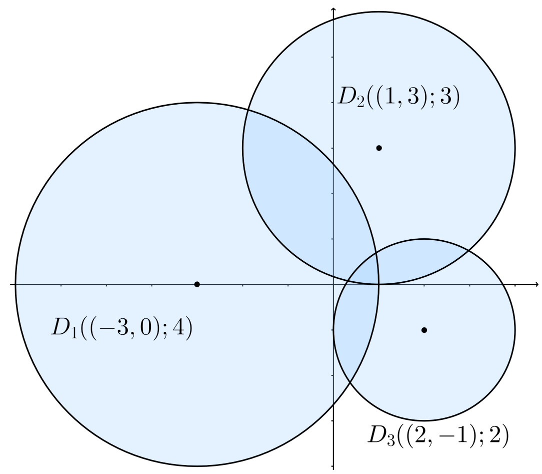
  By default, the disk system must be in a text file named `disks_system.txt`, under the `textfiles` directory of the project.

- The format of the file `disks_system.txt` must be as follows

  $$\begin{array}{l} m\ d \\ c_{11}\ c_{12}\ \cdots c_{1d}\ r_1 \\ c_{21}\ c_{22}\ \cdots c_{1d}\ r_2 \\ \cdots \\ c_{m1}\ c_{m2}\ \cdots c_{md}\ r_m \end{array}$$

  where m is the number of disks of the disk system M, d is the dimension of the space, $c_i = (c_{i1}, c_{i2} \ldots, c_{id})$ and r_i are the center and radius of the i-th disk $D_i(c_i; r_i) \in M$, respectively.

For example, the disk system $M = \{D_1, D_2, D_3\}$, in the following picture,



must be captured in the text file `disks_system.txt` using the format:

```
3 2
-3 0 4
1 3 3
2 -1 2
```

It should be noted that the file `disks_system.txt` must be inside the `textfiles` directory and the `textfiles` directory must be inside the same directory as the `Cech-scale` program.

2.1 Disk system in a plane

If the disk system M consists in a collection of disks in the plane, $M \subset \mathbb{R}^2$, then the `Cech-scale` program allows any number of disks. The maximum number of disks is constrained by the specifications of the system (hardware) where the program is executed.

In this case, the contents of the file `disks_system.txt` must be as follows:

$$m \; 2$$
$$c_{11} \; c_{12} \; r_1$$
$$c_{21} \; c_{22} \; r_2$$
$$\cdots$$
$$c_{m1} \; c_{m2} \; r_m$$

with $m \geq 3$.

2.2 Disk systems in $\mathbb{R}^d$, with $d > 2$

If the disk system M consists of a collection of disks in an euclidean space with dimension higher than 2, then the `Cech-scale` program only allows disk systems with three disks.

In this case, the contents of the file `disks_system.txt` must be as follows:

$$\begin{array}{l} 3\ d \\ c_{11}\ c_{12}\ \cdots c_{1d}\ r_1 \\ c_{21}\ c_{22}\ \cdots c_{1d}\ r_2 \\ c_{31}\ c_{32}\ \cdots c_{3d}\ r_3 \end{array}$$

with $d \geq 2$.

3 Results interpretation

The results of the `Cech-scale` program are saved in the file `cech_results.txt`, inside the `textfiles` directory.

> If the file `cech_results.txt` does not exists, then the program `Cech-scale` creates it; and if `cech_results.txt` already exists, then the program will overwrite it.

The `cech_results.txt` file contains: the Čech scale μ_M of the disk system M and the intersection point

$$\{c_M\} = \bigcap_{1 \leq i \leq m} D_i(c_i; \mu_M r_i).$$

There are two possible outcomes for the Čech scale value, according the Vietoris–Rips scale value, as we discuss in the following two subsections.

3.1 The Čech scale agrees with the Vietoris–Rips scale

The Vietoris–Rips scale of a disk system M, given by:

$$M = \{D_1(c_1; r_1), D_2(c_2; r_2)\},$$

is

$$\nu_M = \frac{\|c_1 - c_2\|}{r_1 + r_2}.$$

In fact, in this case: $\mu_M = \nu_M$.

In general, the Vietoris–Rips scale of the disk system

$$M = \{D_i(c_i; r_i) \subset \mathbb{R}^d \mid r_i > 0 \;\&\; 1 \leq i \leq m \;\&\; m \geq 3\},$$

is given by

$$\nu_M = \max_{1\leq i<j\leq m} \left\{ \frac{\|c_i - c_j\|}{r_i + r_j} \right\}.$$

Calculating the Vietoris-Rips scale is relatively simple and efficient (in the `Cech-scale` program). The program initially calculates ν_M, then verifies if M_{ν_M} is a Čech system, in other words, whether it fulfills the non-empty intersection property:

$$\bigcap_{1\leq i\leq m} D_i(c_i; \nu_M r_i) \neq \emptyset.$$

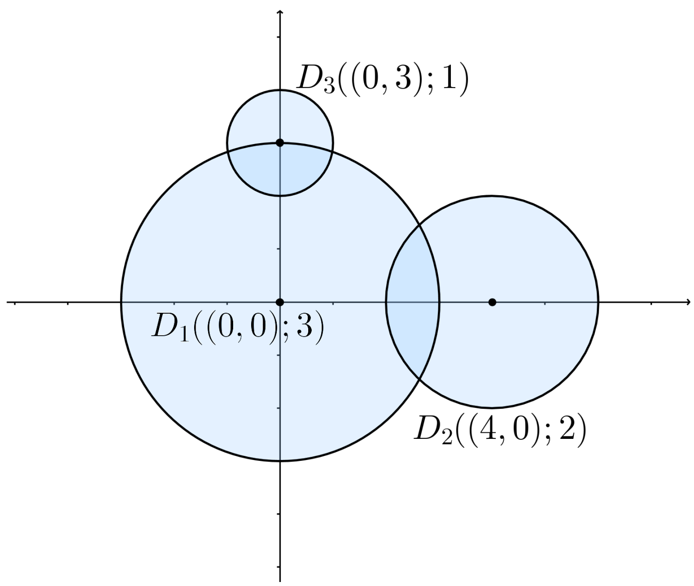
In such case, the equality between the Vietoris-Rips and Čech scale is established in the results file `cech_results.txt`, with the message:

```
The Cech scale agrees with the Vietoris-Rips scale.
```

Then, the file displays the value of the Čech scale μ_M, and the intersection point c_M.

For example, for the disk system

$$M = \{D_1((0,0);3), D_2((4,0);2), D_3((0,3);1)\}$$

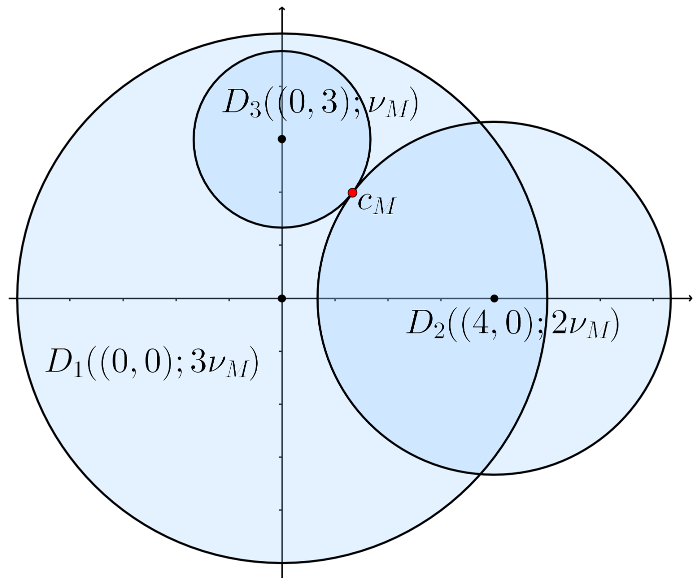


The program `Cech-scale` generates the file `cech_results.txt` with with the following information:

```
The Cech scale agrees with the Vietoris-Rips scale.
Cech scale: 1.66667

The intersection point:
(1.33333, 2)
```

The results imply that, by rescaling the disk system by the scale $\nu_M = 1.666667$, the system M_{ν_M} would be a ech system.



3.2 The Čech scale is greater than the Vietoris-Rips scale

The general and non trivial case, happens when the Čech scale μ_M of the disks system M is greater than the Vietoris-Rips scale ν_M. In this case, `Cech-scale` implement a numerical method to calculate μ_M.

For example, for the previous disk system:

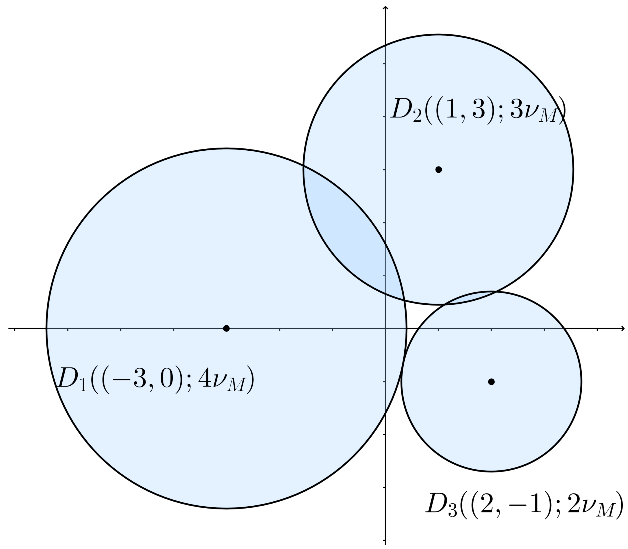
$$M = \{D_1((-3,0);4), D_2((1,3);3), D_3((2,-1);2)\},$$

the program `Cech-scale` writes in the file `cech_results.txt` the following results:

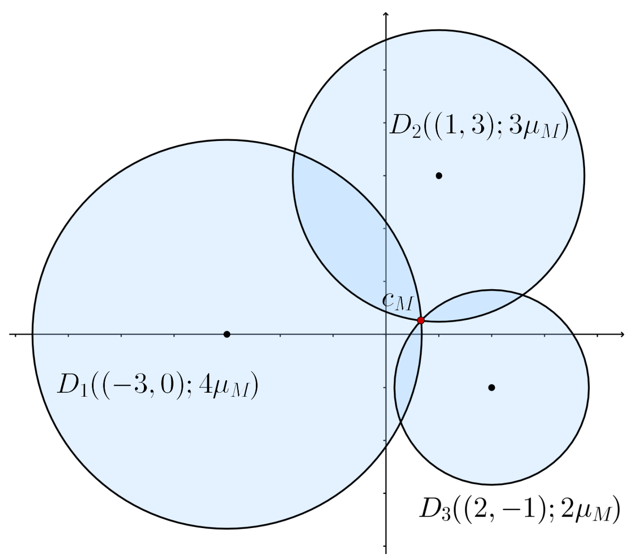
```
The Cech-scale is greater than the Vietoris-Rips scale.
Vietoris-Rips scale: 0.849837
Cech scale: 0.91884

The intersection point:
(0.665882, 0.263803)
```

The rescaled system, using the Vietoris-Rips scale, can be visualized as:



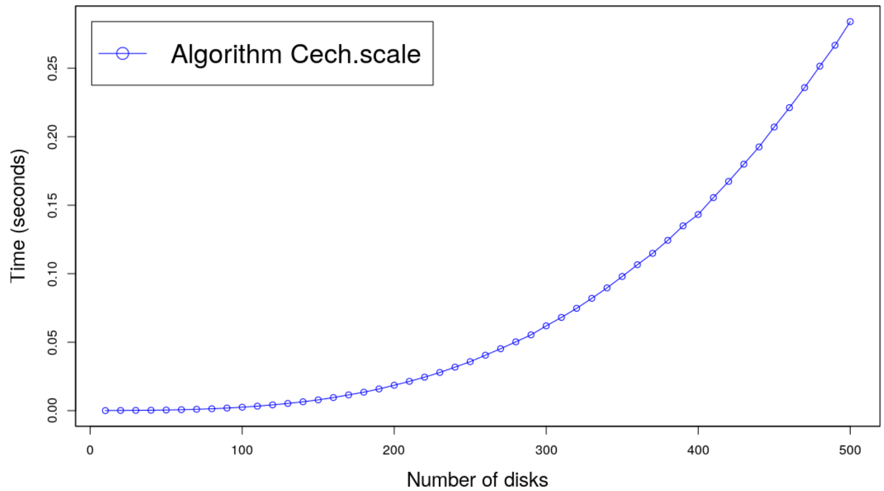
While, rescaling the radii by $\mu_M =$ `9.188403e-01`, the disks system M_{μ_M} has the following configuration:



4 Efficiency of the `Cech-scale` program

In order to calculate the average execution time that the `Cech-scale` program takes in determining the Čech scale μ_M and the intersection point c_M, disk systems were generated (with a uniform distribution) and measured the execution time of each one. The benchmarking was done on an Inten Xeon 3.4 GHz under a GNU/Linux distribution. No paralelization was used.

The following graph shows the average execution time of the `Cech-scale` program, measured with the `clock()` function of the `C` standard library. On disk system with $10n$ disks, $1 \le n \le 50$, 10,000 repetitions were measured for each system.



Likewise, the average execution time of the `Cech-scale` program, with random disks systems in the euclidean space $\mathbb{R}^{200d}$, for $1 \le d \le 50$, were measured. Also in this case, 10,000 repetitions were made.