Cosmic voids are underdense regions, below 30% of the background cosmic density, in the large-scale universe. They live surrounded by enormous filaments, walls, and superclusters. The diameters of the cosmic voids, regularly, are in the range $20 - 50 \, h^{-1} \, Mpc$ [1]. The most popular cosmic void, due to its amazing size with a diameter of $50 \, h^{-1} \, Mpc$, is Böotes void discovered by Kirshner and his colleagues in 1981 [2].

In general, the study of large scale structure involves observational and simulate data. The simulate data, regularly, come from N-body simulations. These simulations are based on a specific theoretical model of cosmology; its relevance lies in comparing and contrasting the observational data with the theory. Nowadays, amazing amounts of data have been generated by space experiments such as SDSS [3][4] and 2dFGRS [5][6]. To find cosmic voids in galaxy catalogs or N-body simulations data (mocks) are necessary computational techniques that allow identifying regions with a lower density than the background cosmic density. Then, a coordination between computer science and physics cosmology is very important. Thus, the search of cosmic voids is about a data science scenario. Within the repertoire of computational tools, we have algorithms and data structures that are useful for finding cosmic voids, in particular, the spatial and geometrical types.

The aim of this work is to mention some spatial algorithms that have been used for cosmic voids finding and to propose an alternative in data structures with octrees. The organization of this paper is as follows. The cosmological relevance of cosmic voids is presented in section II. In section III, we present a brief description of data sets that are used to find cosmic voids. In section IV, it is mentioned the common algorithms of cosmic voids finding. In the section V, we explore the octree possibility as an alternative of cosmic void finder. Finally, in section VI the conclusions are presented.
2. Cosmological relevance of cosmic voids

The cosmic voids represent an interesting scenario for cosmology science. In the first instance, they are the most abundant structure in the large-scale universe. The 95% of the volume of the universe is composed of cosmic voids. Its almost empty nature suggests a government of Dark Matter and Dark Energy [1][7][8]. Furthermore, the cosmic void dynamics is very singular regarding the famous superdense large scale structure, like walls and filaments. Absence of stronger gravitational forces inside cosmic voids provides that galaxies evolve slower [1]. Therefore, the study of void galaxies helps to understand the formation of galaxies in general. In addition, the luminosity of the galaxies found in the cosmic voids, contradicts the prediction of the ΛCDM model for the expected luminosity of galaxies in underdense regions [1].

To determine that a point in the cosmic web has a density lower than the background cosmic density, it is convenient to use a dimensionless quantity, called density contrast:

$$\delta(x, t) = \frac{\rho(x, t)}{\rho(b)} - 1$$

(1)

where ($x$,t) is the density in a x point at the t time, and b is the background cosmic density. Therefore, this parameter helps to analyze the density variation of a cosmic region with respect to the mean cosmic density. In Table 1, we can see the values that the density contrast can take.

| $\delta$     | Description       |
|--------------|-------------------|
| $-1$         | Empty             |
| $0$          | Background cosmic density |
| $-1 < \delta < 0$ | Underdensity   |
| $\delta > 0$ | Superdensity      |

The cosmic voids have also given rise to alternative theories, not completely discarded, to explain the accelerated expansion of the universe [9][10]. These theories consider that our galaxy can lie at the center of a giant cosmic void and the observable effects of cosmic acceleration are due to our privileged position [9].

In the other hand, as well as there are standard rules (BAO) and standard candles (SN1a) to test cosmological models, there are also standard spheres. Cosmic voids offer the possibility of being used as standard spheres through the Alcock-Paczyński cosmological test [11] and because they have approximately a spherical shape [12][13][14]. Hence the importance of finding them in the catalogs of galaxies.

3. Data catalogs

As already mentioned, there are two types of data catalogs: observational and mocks from N-body simulations. We are lucky, because there are enough galaxy catalogs, of both types, available on Internet. In particular, the Large-Scale Structure Galaxy Catalog of the SDSS is a very good option to obtain observational data from BOSS (Baryon Oscillation Spectroscopic Survey) [15][16]. On the other hand, the most representative case of simulated data is contained in the Millenium simulation [17][18].

The SDSS survey covers 8452 square degrees, with spectra of $10^6$ galaxies and $10^5$ quasars, among other data. Meanwhile, the Millenium simulation is one of the largest simulations of
the history using more than 10 billion particles for its purpose. The output data of Millenium simulation needs about 25 terabytes of storage. To find cosmic voids, the galaxies can be idealized as points along a three-dimensional cosmic web. The galaxies catalogs, among other attributes, contain the celestial coordinates (right ascension and declination) of each galaxy and its cosmology redshift. Thus, we have a three-dimensional cosmic region inhabited by galaxy dots. The next step, because our computational interest, is to find underdense regions in this cosmic region with the appropriate algorithm.

4. Algorithms
Due to the spatial patterns present in the cosmic web, the algorithms used to find cosmic voids in galaxy catalogs, in general, are geometric ones. Clustering, density regions, Voronoi and Delunay regions are recurrent words in this type of algorithms.

One of the most successful algorithms to find cosmic voids is ZOBOV (Zones Bordering On Voidness) [19]. Colberg et al [20] shows that this type of algorithm has the advantage to define a better geometrical shape for cosmic voids. Other algorithms use spheres or cubes superimposed and do not have accurate edges for cosmic voids. Later algorithms are based on improved ZOBOV versions, a good example is VIDE [21]. The main ideas of the two algorithms that make up ZOBOV will be, roughly, presented: Voronoi tessellations and watershed.

In first place, for each galaxy (which is a point in a three-dimensional space) ZOBOV uses Qhull C++ code for calculation of every Voronoi galaxy region. This region, for each generator point, is defined as all points closer to that generator point to any other. In math words, a $T_i$ Voronoi region for a $p_i$ generator point, is:

$$T_i = \{ x \in \mathbb{R}^n : \text{dist}(x, p_i) < \text{dist}(x, z_j), \forall j \neq i \}$$

(2)

Then, ZOBOV assigns a density value for each Voronoi region in a simple way: $1/V(i)$, where $V(i)$ is the volume of the Voronoi region around the $i$ point. After that, ZOBOV sends joins each generator point (with its Voronoi region) to its lower density neighbor. There are zones with different values of density. The low-density regions could be called voids; but ZOBOV takes another step to obtain more accurate results: the watershed algorithm.

![Figure 1. First steps in ZOBOV: a) Sparse points; b) Voronoi regions for each point with density assigned; lighter green represents denser regions. Image from [19].](image1)

![Figure 2. Last steps in ZOBOV: c) zones of galaxies; d) determination of cosmic voids. Image from [19].](image2)
It is worthy to mention that Voronoi tessellations have a very long deck of applications. For example, in data mining (k-means algorithm), optimization, image compression and, inclusive, in biology [22]. On the other hand, the watershed algorithm is very common in the image processing. Watershed algorithm, as its name suggest, consider a gray scale image as a topographic relief. Then, returning to the context of ZOBOV, the watershed algorithm allows to unite previously generated zones in larger regions and, those that are below a certain low density, consider them as cosmic voids. The schema of ZOBOV is showed in the Figure 1 and Figure 2.

Although algorithms such as ZOBOV work well, Voronoi tessellation must be calculated for each point. The BOSS galaxy catalog in SDSS has around $10^6$ galaxies, so the required computing power is very high. So, we are looking for other alternatives to find cosmic voids.

5. The proposal: Octrees, spatial structures

The octrees are the generalization in three dimensions of quadtrees. Both are known as hierarchical spatial data structures or tree codes. They are based on the principle of recursive decomposition and map spatial data in image databases [23]. In cosmology and astrophysics, these structures are very used in N-body simulations, because at the end of the 1980s, Barnes and Hut proposed an algorithm using octrees that improved complexity for $O(n^2)$, of previous algorithms, to $O(n \log n)$ [24]. However, in N-body simulations, the objective is to calculate gravitational forces while, in the context of cosmic voids finding, we need to determine regions of low density. Fortunately, the octrees are also used in image processing and allow high precision in volume mapping [25][26].

Basically, given a cubical volume, it is recursively subdivided into eight congruent disjoint cubes (called octants). Then, at each step, the resulting cubes are evaluated for certain criteria. In the case of a three-dimensional object reproduction, the criterion is the presence or absence of the object. Then, in a tree representation, the criteria and their respective values for each octant are indicated in binary form (see the Figure 3). The situation with cosmic voids is slightly different that in the case of the Figure 3. In the example of Figure 3, a cubic region is divide into octants, then each octant is evaluated with the question: is totally fully filled, totally empty or partially filled? If the answer is completely filled or empty, do not subdivide it again and, in the representation of the three, this is indicated with black and white, respectively. On the other hand, if the octant has an intermediate state, it is subdivided and the entire cycle is repeated. For cosmic voids, we need to find regions with a negative density contrast, so the question changes to: is the density of the region lower than the background density? If the answer is positive, then the region is not subdivided again. In counterpart, when the octant has a higher density, the recursive decomposition is applied again.

![Figure 3. a) 3D object, b) octree block decomposition, c) tree representation [23].](image-url)
We must establish the background cosmic density $\rho$ and the expected minimum size $\text{min_diam}$ of the diameters of cosmic voids. Remember that, cosmic voids have less than 30% background cosmic density. Also, we assign the density value $\text{dens_oct}$ for each octant in the same way as in ZOBOV. Let $L$, the length of the sides of the octant, a pseudo code is:

```python
while L > min_diam:
    Divide the cubic volume in octants.
    for each octant:
        if dens_oct > background_cosmic_density:
            return octant
        else:
            continue
```

Note that, if the octant density is less than the background cosmic density, then it is no longer considered to be subdivided again, so it will be represented as a leaf of the tree. To join underdense octants to form cosmic voids candidates, we must find his neighbor leaf; there are several algorithms that allow us this, like bottom-up neighbor finding [23].

6. Final comments and conclusions

Geometric algorithms to calculate the Voronoi regions calculate the distances between each point (galaxy) and its neighbors for each iteration, which implies many calculations, even more when it comes to millions of points as is the case of the catalogs of galaxies. Since the regions with low density that define the cosmic voids contain several galaxies, we can think of other methods in which a complex process, such as the estimation of the Voronoi regions, is not carried out for each point. In this study, we conclude that an alternative to be seriously considered to find cosmic voids is in the octrees data structure. Its nature can improve efficiency and reduce computing costs. We have as a future work to implement the pseudo code and to program the corresponding tree structures.

Acknowledgments

This work was partially supported by CONACyT-Mexico scholarship, BEIFI-IPN and the SIP20180252 grant by IPN. Special to Dr. Gilberto Martínez Luna and Dr. Adolfo Guzmán Arenas from CIC-IPN.

References

[1] van de Weygaert R and Platen E 2011 Cosmic Voids: structure, dynamics and galaxies International Journal of Modern Physics: Conference Series Vol. 1, pp. 41-66. World Scientific Publishing Company.
[2] Kirshner R P, Oemler Jr A, Schechter P L and Shectman S A 1981 A million cubic megaparsec void in Bootes The Astrophysical Journal, 248, L57-L60.
[3] http://www.sdss.org/
[4] York D G et al 2000 The sloan digital sky survey: Technical summary The Astronomical Journal, 120(3), 1579.
[5] http://www.2dfgrs.net/
[6] Colless M et al 2001 The 2df galaxy redshift survey: spectra and redshifts Monthly Notices of the Royal Astronomical Society, 328(4), pp 1039-1063.
[7] Hahn O, Porciani C, Carollo C M and Dekel A 2007 Properties of dark matter haloes in clusters, filaments, sheets and voids. Monthly Notices of the Royal Astronomical Society, 375(2), pp 489-499.
[8] Bos E P, van de Weygaert R, Dolag K and Pettorino V 2012 The darkness that shaped the void: dark energy and cosmic voids. Monthly Notices of the Royal Astronomical Society, 426(1), pp 440-461.
[9] Clifton T and Ferreira P G 2009 Does dark energy really exist? Scientific American, 300(4), pp 48-55.
[10] Alexander S et al 2009 Local void vs dark energy: confrontation with WMAP and type Ia supernovae Journal of Cosmology and Astroparticle Physics, 2009(09), pp 025.
[11] Alcock C and Paczyński B 1979 An evolution free test for non-zero cosmological constant Nature, 281(5730), pp 358-359.
[12] Sutter P M et al 2012 A first application of the Alcock-Paczynski test to stacked cosmic voids The Astrophysical Journal, 761(2), pp 187.

[13] Lavaux G and Wandelt B D 2012 Precision cosmography with stacked voids The Astrophysical Journal, 754(2), pp 109.

[14] Sutter P M et al 2014 A measurement of the AlcockPaczyski effect using cosmic voids in the SDSS Monthly Notices of the Royal Astronomical Society, 443(4), pp 2983-2990.

[15] Mao Q et al 2017 Cosmic Voids in the SDSS DR12 BOSS Galaxy Sample: the AlcockPaczyski test The Astrophysical Journal, 835(2), pp 160.

[16] https://data.sdss.org/sas/dr12/boss/

[17] Springel V et al 2005 Simulations of the formation, evolution and clustering of galaxies and quasars. nature 435(7042), pp 629.

[18] https://wwwmpa.mpa-garching.mpg.de/galform/virgo/millennium/

[19] Neyrinck M C 2008 ZOBOV: a parameter-free void-finding algorithm. Monthly notices of the royal astronomical society 386(4) pp 2101-2109.

[20] Colberg J M et al 2008 The AspenAmsterdam void finder comparison project Monthly Notices of the Royal Astronomical Society 387(2), pp 933-944.

[21] Sutter P M et al 2015 VIDE: the Void IDentification and Examination toolkit Astronomy and Computing 9, pp 1-9.

[22] Du Q, Gunzburger M, Ju L and Wang X 2006 Centroidal Voronoi tessellation algorithms for image compression, segmentation, and multichannel restoration Journal of Mathematical Imaging and Vision 24(2), pp 177-194.

[23] Samet H 1988 An overview of quadtrees, octrees, and related hierarchical data structures Theoretical Foundations of Computer Graphics and CAD pp. 51-68. Springer, Berlin, Heidelberg.

[24] Barnes J and Hut P 1986 A hierarchical O (N log N) force-calculation algorithm. nature, 324(6096), pp 446.

[25] Hornung A et al 2013 OctoMap: An efficient probabilistic 3D mapping framework based on octrees Autonomous Robots, 34(3), pp 189-206.

[26] Wurm K M et al 2011 Hierarchies of octrees for efficient 3d mapping Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on (pp. 4249-4255). IEEE.