The Astrophysical Journal, 860:118 (16pp), 2018 June 20 https://doi.org/10.3847/1538-4357/aac263

Blooming Trees: Substructures and Surrounding Groups of Galaxy Clusters

Heng Yu (余恒)[1,2,3], Antonaldo Diaferio[2,3], Ana Laura Serra[4], and Marco Baldi[5,6,7]
[1] Department of Astronomy, Beijing Normal University, Beijing, 100875, People's Republic of China
[2] Dipartimento di Fisica, Università di Torino, Via P. Giuria 1, I-10125 Torino, Italy
[3] Istituto Nazionale di Fisica Nucleare (INFN), Sezione di Torino, Via P. Giuria 1, I-10125 Torino, Italy
[4] On leave from the Dipartimento di Fisica, Università di Milano, Via Celoria 16, I-20133 Milano, Italy
[5] Dipartimento di Fisica e Astronomia, Alma Mater Studiorum Università di Bologna, via Gobetti 93/2, I-40129 Bologna, Italy
[6] INAF—Osservatorio Astronomico di Bologna, via Gobetti 93/3, I-40129 Bologna, Italy
[7] INFN—Sezione di Bologna, viale Berti Pichat 6/2, I-40127 Bologna, Italy
Received 2017 September 15; revised 2018 March 31; accepted 2018 May 1; published 2018 June 19

Abstract

We develop the Blooming Tree Algorithm, a new technique that uses spectroscopic redshift data alone to identify the substructures and the surrounding groups of galaxy clusters, along with their member galaxies. Based on the estimated binding energy of galaxy pairs, the algorithm builds a binary tree that hierarchically arranges all of the galaxies in the field of view. The algorithm searches for buds, corresponding to gravitational potential minima on the binary tree branches; for each bud, the algorithm combines the number of galaxies, their velocity dispersion, and their average pairwise distance into a parameter that discriminates between the buds that do not correspond to any substructure or group, and thus eventually die, and the buds that correspond to substructures and groups, and thus bloom into the identified structures. We test our new algorithm with a sample of 300 mock redshift surveys of clusters in different dynamical states; the clusters are extracted from a large cosmological *N*-body simulation of a ΛCDM model. We limit our analysis to substructures and surrounding groups identified in the simulation with mass larger than $10^{13}\,h^{-1}\,M_{\odot}$. With mock redshift surveys with 200 galaxies within $6\,h^{-1}$ Mpc from the cluster center, the technique recovers 80% of the real substructures and 60% of the surrounding groups; in 57% of the identified structures, at least 60% of the member galaxies of the substructures and groups belong to the same real structure. These results improve by roughly a factor of two the performance of the best substructure identification algorithm currently available, the σ plateau algorithm, and suggest that our Blooming Tree Algorithm can be an invaluable tool for detecting substructures of galaxy clusters and investigating their complex dynamics.

Key words: galaxies: clusters: general – large-scale structure of universe – methods: numerical – methods: statistical

1. Introduction

According to the standard cold dark matter (CDM) paradigm, large cosmic structures form by merging of smaller structures (Colberg et al. 1999, 2005). In this hierarchical universe, galaxy clusters form at later times, and, at the present time, some clusters are still accreting mass by merging. A clear signature of this process is the presence of substructures in the galaxy density distribution, in the X-ray and radio emission, or in the dark matter distribution inferred from gravitational lensing (e.g., Yu et al. 2016, and references therein). Therefore, investigating the properties of substructures can constrain the models of structure formation and evolution (Geller & Beers 1982; Mohr et al. 1996; Natarajan et al. 2007; Okabe et al. 2014; Mohammed et al. 2016; Yu et al. 2016), the connection between galaxy properties and environment (e.g., Hwang et al. 2012; Pranger et al. 2013; Agulli et al. 2016; Utsumi et al. 2016; Oguri et al. 2018), and even the nature of dark matter (e.g., Harvey et al. 2015; Kummer et al. 2018; Robertson et al. 2017).

Identifying dynamically distinct substructures in galaxy clusters is not a trivial task. Most methods identify substructures in the galaxy density distribution based on spectroscopic data (see Yu et al. 2015 for a brief review). Among these methods, those relying on the hierarchical clustering analysis appear to be particularly efficient.

The hierarchical clustering analysis is a general statistical method. It is designed to partition a system into optimally homogeneous subgroups on the basis of empirical measures of similarity (see Everitt et al. 2011 for a detailed description). Materne (1978) first applied a hierarchical clustering analysis to astronomical data to identify groups of galaxies. Serna & Gerbal (1996) introduced the pairwise binding energy to link galaxies in the field of view (FoV) of a cluster and arrange them in a binary tree. Building a binary tree is a standard method to quantify the hierarchical structures of the entire system. This approach does not rely on any morphological assumption or dynamical state, and it is thus suitable for analyzing dynamically complex self-gravitating systems, like galaxy clusters.

In 1997, Diaferio & Geller (1997) introduced the caustic method to estimate the mass profile of galaxy clusters in their outer regions, where the dynamical equilibrium assumption does not necessarily hold. In the detailed illustration of the algorithm of the caustic method, where galaxies are arranged in a binary tree, similar to the procedure suggested by Serna & Gerbal (1996), Diaferio (1999) first proposed the identification of a σ plateau on the main branch of the tree to locate the cluster and return a list of cluster members and cluster substructures. Serra et al. (2011) provide detailed and complete statistical tests of the caustic technique and propose a refined and more robust version of this σ plateau algorithm.

The efficiency of the σ plateau algorithm to identify the cluster substructures in *N*-body simulations is shown in Yu et al. (2015), who emphasize a unique feature of this algorithm: unlike other methods for the identification of substructures with

spectroscopic data, like the Dressler & Shectman (DS) method (Dressler & Shectman 1988; Solanes et al. 1999; Knebe & Müller 2000; Aguerri & Sánchez-Janssen 2010; Dressler et al. 2013) or the KMM (Bird 1994; Colless & Dunn 1996; Barmby & Huchra 1998) or DEDICA (Pisani 1996; Ramella et al. 2007) algorithms, the σ plateau algorithm gives an unambiguous association of galaxies to individual substructures; it thus enables the estimation of the substructure properties, like size, velocity dispersion, and mass.

With this feature available, we can apply the strictest criterion for comparing the substructures identified in three dimensions in an N-body simulation with the substructures identified in redshift space: in mock redshift surveys with 200 galaxies within $3R_{200}$ from the cluster center, where R_{200} is the usual radius of the sphere whose average density is 200 times the critical density of the universe, the σ plateau algorithm recovers ∼30%–50% of the real substructures, depending on the mass and the dynamical state of the cluster (Yu et al. 2015). This performance is unprecedented. The algorithm was successfully applied to the galaxy distribution in the FoV of the cluster A85 (Yu et al. 2016): it provided a unique understanding of the complex dynamics of this cluster when combined with the bulk motions of the intracluster medium in different regions, as inferred by the redshift measurements derived by X-ray spectroscopy.

Despite its very good performance, the σ plateau algorithm actually overlooks the crucial fact that cluster substructures can have widely different velocity dispersions. Here, we present the Blooming Tree Algorithm, a new algorithm that takes this fact into account and thus represents a significant improvement over the σ plateau algorithm. We show how this more sophisticated algorithm substantially doubles the substructure identification efficiency. In addition, the Blooming Tree Algorithm returns the list of the groups in the cluster outskirts along with their members. This feature provides a fundamental tool that enables a quantitative investigation of the merging and accretion history of galaxy clusters (Rines et al. 2001; Lemze et al. 2013; De Boni et al. 2016).

In Section 2, we describe the cosmological N-body simulation and the mock redshift surveys of the galaxy cluster fields we use to test the method. We describe the Blooming Tree Algorithm and its results in Sections 3 and 4, respectively. In Section 5 we compare the performance of our new technique with the σ plateau algorithm. We conclude in Section 6.

2. The Mock Redshift Catalogs of Simulated Clusters

We use the Coupled Dark Energy Cosmological Simulations (Baldi 2012), the largest set to date of N-body simulations that model the interaction between a dark energy scalar field and the CDM fluid. Here, we only consider the simulation of the standard ΛCDM model with fiducial WMAP7 parameters (Komatsu et al. 2011). The simulated volume is a cube of 1 comoving h^{-1} Gpc on a side ($h = H_0/100\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$ is the dimensionless Hubble constant), containing 1024^3 CDM particles with mass $5.84 \times 10^{10}\,h^{-1}\,M_\odot$ and the same number of baryonic particles with mass $1.17 \times 10^{10}\,h^{-1}\,M_\odot$. We only consider the dark matter particles: we assume that, in the real universe, galaxies are unbiased tracers of the velocity field of the dark matter particles. In fact, both N-body simulations (e.g., Diaferio et al. 2001; Diemand et al. 2004; Gill et al. 2004, 2005) and observations (e.g., Rines et al. 2008, 2016) indicate that any velocity bias between galaxies and dark matter is smaller than 10%.

Halos are identified with the friends-of-friends (FoF) algorithm (Huchra & Geller 1982; Davis et al. 1985), which links particles with distances less than the linking length l_{FoF} to form a group. We adopt the standard linking length $l_{\mathrm{FoF}} = 0.2\,l_{\mathrm{mean}}$, with l_{mean} the mean interparticle separation, corresponding to the overdensity at virialization $\delta_v = \rho/\rho_{\mathrm{b}} = 185$ (Audit et al. 1998), with ρ_{b} the mean background density. In this procedure, the FoF halos are identified using the CDM particles as primary tracers and then linking baryonic particles to the group of their closest CDM neighbor.

We also identify the 3D substructures of the halos in the simulations with SUBFIND (Springel et al. 2001), whose algorithm is based on the overdensity and the gravitational binding energy of the particles (see Baldi 2012 for further details). With the mass of a 3D substructure we always indicate its total mass, namely the sum of the mass of the particles (both CDM and baryons) that are gravitationally bound to that substructure as identified by SUBFIND.

We consider a sample of 100 FoF halos at redshift $z = 0$ within the mass range 10^{14}–$10^{15}\,h^{-1}\,M_\odot$ with the aim of covering the variety of dynamical states; specifically, we consider 50 "merging" halos and 50 "normal" halos. We choose the 50 merging halos whose mass is closest to $5 \times 10^{14}\,h^{-1}\,M_\odot$ and that contain a substructure whose mass is at least half the mass of the halo core, where the core is the SUBFIND substructure whose center coincides with the halo center. Among the remaining halos in 10^{14}–$10^{15}\,h^{-1}\,M_\odot$, we select the 50 normal halos whose mass is closest to $5 \times 10^{14}\,h^{-1}\ M_\odot$ and that are not merging halos. The masses of our 100 halos are in the range $(4.17\text{–}6.39) \times 10^{14}\,h^{-1}\,M_\odot$, with a median mass $4.93 \times 10^{14}\,h^{-1}\,M_\odot$.

We locate each halo at the center of the simulation box, exploiting the periodic boundary conditions. To mimic the observation of real clusters, we assign the celestial coordinates $(\alpha, \delta) = (6^h, 0°)$ and a redshift distance $cz = 36{,}000\,\mathrm{km\,s^{-1}}$ to the halo center. Around the halo, we consider a rectangular prism enclosing the volume corresponding to a solid angle that at the halo distance ensures we cover a square area $12\,h^{-1}\,\mathrm{Mpc}$ wide. The volume is centered on the halo, and it is $140\,h^{-1}\,\mathrm{Mpc}$ deep. The resulting FoV is $1.^\circ6 \times 1.^\circ6$. For each halo, we apply this procedure to three orthogonal directions. Since the halos are generally not spherically symmetric, for our statistical purposes we can consider these three catalogs as independent mock clusters. So we obtain 300 mock redshift catalogs.

The observational volumes we extract from the simulation typically contain $\sim 6 \times 10^4$ particles. To get the number of particles close to a realistic number of observable galaxies, we randomly sample the dark matter particles until we obtain a given number of particles N_c within a sphere of radius $6\,h^{-1}\,\mathrm{Mpc}$ from the halo center, corresponding to $\sim 5R_{200}$ for the halos of our sample. This procedure yields mock catalogs with different numbers of total particles N. To explore the effect of particle sampling, we build catalogs with $N_c = (50, 100, 150, 200, 250, 300)$. Additionally, we only retain particles in the mock catalogs whose redshift is within $\pm 4000\,\mathrm{km\,s^{-1}}$ from the halo redshift.

By randomly sampling the dark matter particles, the number of members of a 3D substructure in the mock catalog can be substantially reduced or even vanish. We only consider 3D substructures that have at least 10 particles appearing in the FoV. An example is shown in Figure 1. It is a mock catalog with $N_c = 100$ and $N = 304$.

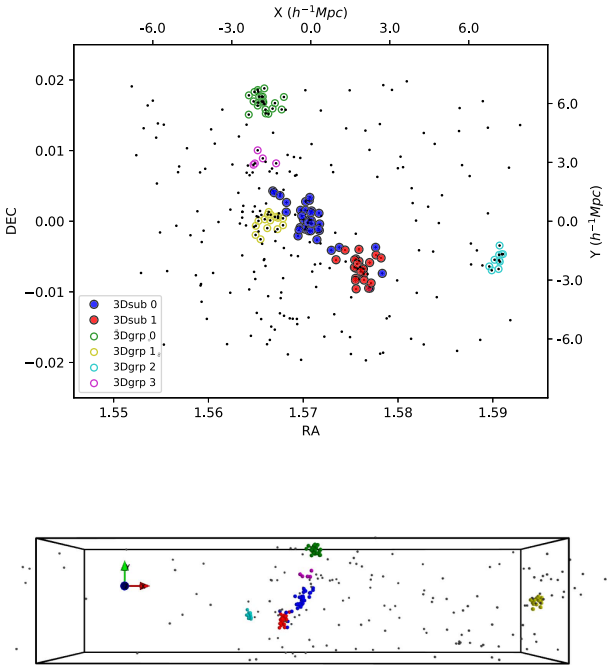

Figure 1. The upper panel shows the distribution of the 3D substructures and the surrounding groups of a merging halo on the plane of the sky. The axis labels show both the projected celestial coordinates R.A. and decl., in radiants, and the comoving coordinates in the *N*-body simulation. The blue solid dots show the particles in the halo core at the center of the FoV (sub 0 in the inserted legend). The red solid dots show the members of the largest 3D substructure that identifies this halo as a merging halo (sub 1 in the legend). The colored open circles show the members of the surrounding groups, as listed in the inserted legend. The remaining black points show the remaining particles in the FoV. The lower panel provides a 3D perspective of the system. We overplot a box with dimensions $14 \times 14 \times 60\ h^{-3}$ Mpc^3.

The properties of our mock catalogs are listed in Table 1. The third to fifth columns list the medians and percentile ranges of the number of particles N in the FoV as a function of N_c. These mock redshift surveys are comparable to recent large galaxy surveys of clusters and their surroundings, such as CIRS (Rines & Diaferio 2006) and HeCS (Rines et al. 2013). The seventh column is the total number of 3D substructures $n_{\rm sub}$, with at least 10 member particles in the FoV and with mass larger than $10^{13}\,h^{-1}\,M_\odot$. This threshold is the minimum 3D substructure mass set by the number of luminous galaxies that can be detected in current typical surveys: a $10^{13}\,h^{-1}\,M_\odot$ substructure is expected to contain a handful of galaxies brighter than $10^{10}\,h^{-1}\,L_\odot$ at most. Table 1 also lists the total number of halos $n_{\rm cl}$ and the total number of groups $n_{\rm grp}$, namely the 3D halos found by the FoF algorithm, that surround each individual central halo. As expected, the number of 3D surrounding groups appearing in the FoV increases with increasing N_c, whereas the number of 3D substructures appears to be less sensitive to N_c. Hereafter we refer to the halo at the center of the FoV as the *cluster* and to the particles in the FoV as the *galaxies*.

The Blooming Tree Algorithm identifies three different kinds of structures: (1) substructures, (2) cores, and (3) surrounding groups. The substructures only contain main cluster members, namely the galaxies linked by the FoF algorithm; the core is the substructure that contains the cluster center; the surrounding groups are structures containing galaxies that are not members of the main cluster. Below we generically indicate any of these three kinds of structures as *structures*, unless specified otherwise.

Table 1
Number of Particles N and Structures in the FoV

N_c	Type	N			$n_{\rm cl}$	$n_{\rm sub}$	$n_{\rm grp}$
		10%	50%	90%			
50	Normal	127	176	257	150	21	233
	Merging	128	173	268	150	133	274
	Total	128	176	264	300	154	507
100	Normal	250	362	502	150	48	532
	Merging	258	364	523	150	185	550
	Total	256	364	515	300	233	1082
150	Normal	400	527	747	150	73	821
	Merging	397	526	800	150	217	855
	Total	399	527	776	300	290	1676
200	Normal	522	709	1004	150	83	1090
	Merging	506	717	1048	150	244	1144
	Total	515	716	1010	300	327	2234
250	Normal	653	882	1279	150	82	1357
	Merging	649	889	1268	150	255	1391
	Total	649	886	1279	300	337	2748
300	Normal	795	1044	1571	150	102	1592
	Merging	783	1066	1528	150	276	1595
	Total	795	1063	1562	300	378	3187

3. The Blooming Tree Algorithm

3.1. Tree Construction

According to hierarchical clustering models, clusters of galaxies form by the aggregation of smaller systems accreting from the surrounding region. To a good approximation, galaxies are collisionless objects during cluster merging, and the transfer of mechanical energy to galaxy internal degrees of freedom is negligible. If the 3D binding energy can be fairly represented by their projected values, we can infer the internal structures of a cluster based on a hierarchical clustering analysis, where we adopt the galaxy projected binding energy as the similarity. Although the 3D and projected binding energies of an individual pair might be largely discrepant from each other, these two quantities in a sample of pairs are strongly correlated, supporting our adoption of the projected binding energy in the hierarchical clustering analysis.[8]

[8] Nonparametric statistical tests between the 3D binding energy of individual galaxy pairs in our simulation and their 2D binding energy estimated with Equation (1) demonstrate that the correlation between the two quantities is strong: for our sample of $\sim 1.4 \times 10^7$ pairs, we find the Spearman and Kendall rank correlation coefficients $r = 0.50$ and $\tau = 0.46$, respectively, which have a probability $P < 10^{-30}$ to appear for an uncorrelated sample. These coefficients increase to $r = 0.79$ and $\tau = 0.62$, again with probability $P < 10^{-30}$, if we limit the sample to bound pairs, namely pairs with negative 3D binding energy. To find a probability $P > 10^{-30}$, we need to take a random subsample of less than 100 pairs: for a subsample of 36 pairs, we find $P = 4 \times 10^{-8}$ and $P = 4 \times 10^{-9}$ for the Spearman and Kendall correlation coefficients, respectively, showing that the correlation remains robust even for relatively small pair samples.

We perform the hierarchical clustering analysis by building a binary tree as follows (see Diaferio 1999 and Serra et al. 2011 for further details):

i. Initially each galaxy is a group g_α.
ii. The binding energy $E_{\alpha\beta} = \min\{E_{ij}\}$, where E_{ij} is a projected binding energy between the galaxy $i \in g_\alpha$ and the galaxy $j \in g_\beta$, is the similarity associated with each group pair g_α, g_β. The projected binding energy is estimated with the equation

$$E_{ij} = -G\frac{m_i m_j}{R_p} + \frac{1}{2}\frac{m_i m_j}{m_i + m_j}\Pi^2, \quad (1)$$

where R_p is the pair projected separation, Π is the line-of-sight velocity difference, and $m_i = m_j = 10^{12}\,h^{-1}\,M_\odot$ is the typical total mass of a luminous galaxy.[9]
iii. The two groups with the smallest binding energy $E_{\alpha\beta}$ are replaced with a single group g_γ, and the total number of groups is decreased by one.
iv. The procedure is repeated from step (ii) until only one group is left.

At this stage, all of the galaxies, now the leaves, are arranged in a binary tree that quantifies their hierarchical relationship. When plotting the dendrogram representing the binary tree (Figure 2, top panel), the node index, namely the similarity, or the binding energy in our case, is adopted as the quantity shown on the vertical axis (Serna & Gerbal 1996). However, in our case, the vertical axis can show more information when a different quantity is displayed.

Here we choose to show the velocity dispersion of each node, estimated with all of the leaves hanging from that node, because it is a physical property directly related to the depth of the gravitational potential well of a bound structure. The velocity dispersion of the nodes is not always a monotonic function when walking from the root to the leaves; therefore, unlike the projected binding energy, displaying the velocity dispersion on the y axis of the dendrogram might produce branches that intersect each other, as shown in the bottom panel of Figure 2. However, this choice is more advantageous than displaying the similarity, because it generally separates different bound structures more clearly.

The dendrograms shown in Figure 2 correspond to the merging cluster in the FoV shown in Figure 1: the member galaxies of the core (in blue) and of the main substructure (in red) are mainly separated into two branches. However, due to projection effects, the members of the same structure are not always close to each other on the dendrogram. Surrounding groups (with green, yellow, cyan, and magenta circles in Figure 1) also appear as distinct branches of the binary tree. The goal of the next step of our Blooming Tree Algorithm is to identify the branches corresponding to these structures.

Incidentally, we remark here that in the language of the standard cluster analysis (e.g., Everitt et al. 2011; Hennig et al. 2015), our binary tree construction is based on a single-linkage hierarchical algorithm. Being based on an estimate of the pairwise gravitational binding energy of galaxies, which is the fundamental physical quantity for identifying gravitational structures, our approach is the best physically motivated method and compensates for known shortcomings of the single-linkage method, like the tendency of connecting independent structures, similar to the standard friends-of-friends algorithm (Huchra & Geller 1982). The physical interpretation of the linkage in other standard agglomerative methods, less prone to this shortcoming, would necessarily be, in this context, more vague and questionable.

[9] At this stage, we refrain from including different masses for galaxies of different luminosity to avoid the introduction of additional degrees of freedom: the mass-to-light ratio depends on the galaxy morphological type and luminosity, which, in turn, depends on the fixed angular aperture generally used for the photometric measurement; therefore, including the connection between the measured luminosity and the galaxy mass requires a nonnegligible number of parameters.

3.2. Buds: Binding Energy Minima

Substructures hang from different nodes of separated branches. They may have different velocity dispersions and different binding energies because they have different masses. The identification of these structures requires the identification of the proper branches of the binary tree.

To locate the minima of the gravitational potential wells, we consider the binding energy of all of the leaves. An example is shown in the lower inset in the bottom panel of Figure 2. This binding energy shows deep minima corresponding to real structures and fluctuations mostly caused by chance alignments. To suppress this noise, we smooth the profile with a box filter that is five leaves wide. The blue solid curve in the inset shows the smoothed profile. We call buds the local minima of this smoothed binding energy profile. Smoothing algorithms more sophisticated than the box filter are clearly possible, but they are unnecessary for our only purpose here of finding the buds from the local minima of the binding energy curve. The buds identify the branches that will be searched for the identification of the real structures, as illustrated in the next step.

3.3. Branch Searching and Blooming Buds

Let us consider the dendrogram in the bottom panel of Figure 2, where the vertical axis shows the velocity dispersion of each tree node, and let us walk from the leaves to the root on a given branch by moving from one node to its parent node. Being a binary tree, this path implies adding a leaf at each step. We see that the velocity dispersions of the nodes on the same branch either are basically unaffected by the step toward the root or vary substantially. In other words, when moving from the leaves to the root, the velocity dispersion on a given branch does not generally grow smoothly, but it shows either sudden jumps or plateaus. These plateaus indicate the presence of structures. The original version of the σ plateau algorithm identifies one single plateau on the main branch of the tree and isolates the cluster and its substructures (Diaferio 1999; Serra et al. 2011; Yu et al. 2015). In systems with complex dynamics, the single plateau might not actually be flat or there might be several plateaus. Identifying a single plateau is thus not obvious: a too-large threshold may erroneously associate distinct substructures into a single substructure, whereas a too-small threshold may separate an individual substructure into smaller clumps.

Here, we implement a new algorithm that combines three pieces of information that are missing from the original version of the σ plateau algorithm: the line-of-sight velocity dispersion σ_v of the leaves hanging from a node, the number n of these leaves, and the distribution of the leaves on the sky; σ_v and n are combined in the average velocity dispersion σ_v/n, because, when two branches corresponding to two distinct structures

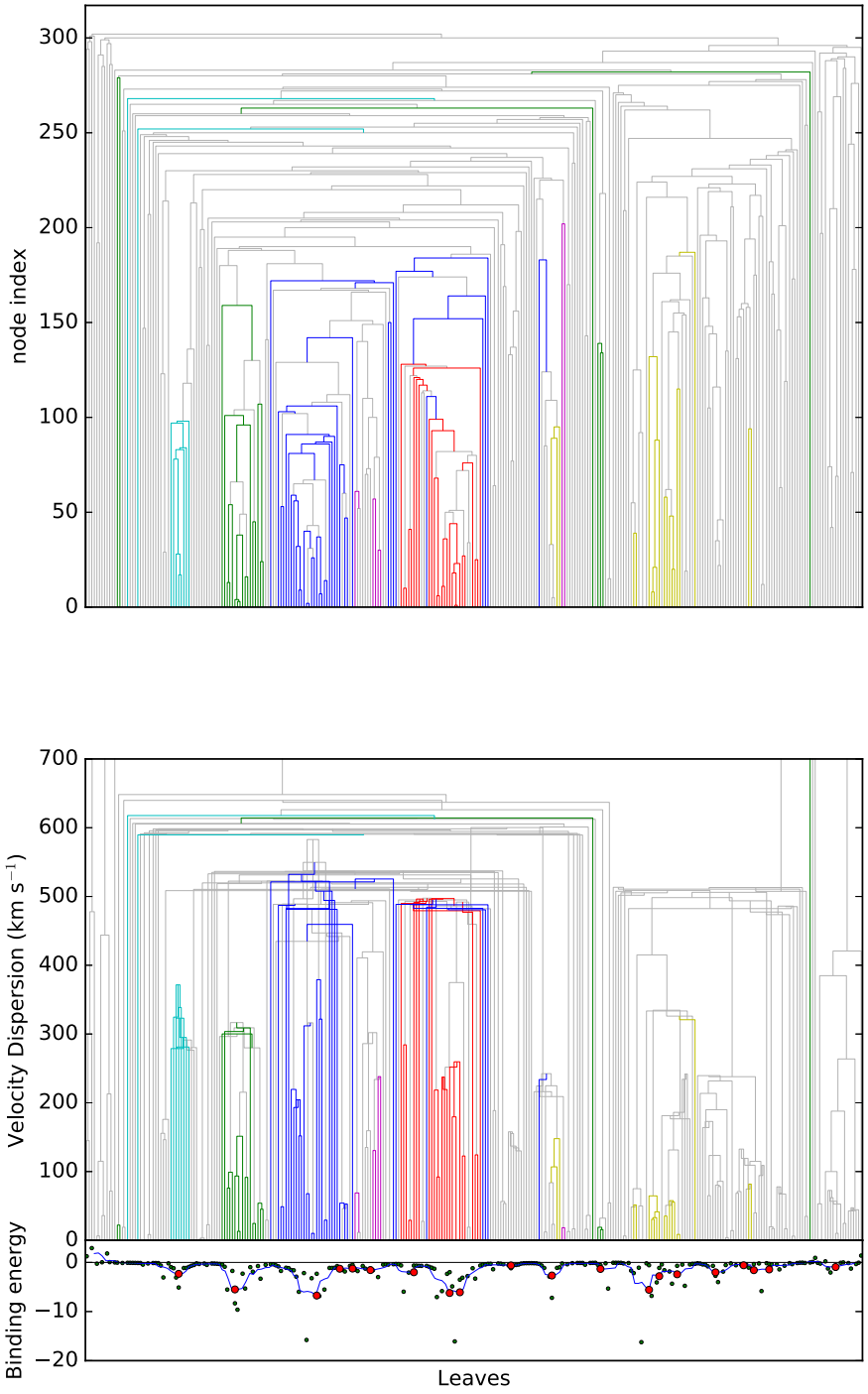

Figure 2. Two dendrograms illustrating the binary tree of the merging cluster shown in Figure 1 with 304 galaxies in the FoV. The galaxies are the leaves of the tree at the bottom of each dendrogram. All galaxies belonging to different structures are shown with different colors, with the same color code as in Figure 1. In the top panel, the y axis of the dendrogram displays the node index: larger indexes indicate less bound branches. In the bottom panel, the y axis of the dendrogram displays the node velocity dispersion. The black dots, one for each leaf, in the lower inset in the bottom panel show the binding energy profile of the binary tree: the blue solid curve is obtained by smoothing the profile with a five-leaf-wide box filter; the red dots show the local minima, or buds, whereas the horizontal solid black line indicates a binding energy equal to zero.

with similar σ_v and n merge, the branch of the combined structure has σ_v comparable to the original structures, but σ_v/n is reduced by a factor of $\sim$2. Therefore, the relation σ_v versus node index is roughly flat, whereas the relation σ_v/n versus node index shows a rapid decrease. We also take the distribution of the galaxies on the sky into account, because this piece of information is crucial to identifying a real 3D structure, as we show below.

To implement these three diagnostics, we consider the mass of a system estimated with the virial theorem

$$M_v = \frac{4}{3}\pi R_v^3 \delta_v \rho_c = \alpha \frac{3\sigma_v^2 R_v}{G}, \quad (2)$$

where R_v is the virial radius, the average density of the system is δ_v times the critical density $\rho_c = 3H^2/8\pi G$, with H the

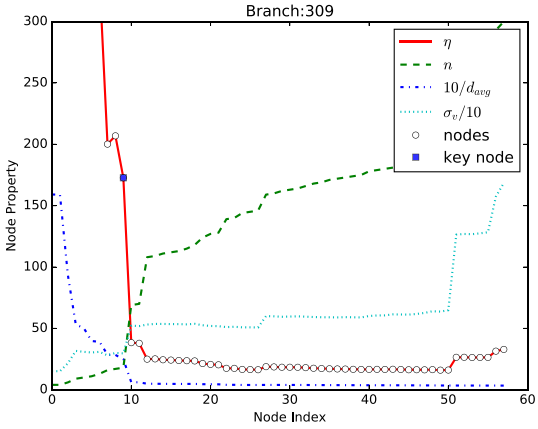

Figure 3. Node properties of one branch (grp 0, green) of the binary tree of Figure 2 from the leaves to the root; $d_{\rm avg}$ and σ_v are rescaled to fit into the plot. The blue square indicates the key node where we cut the branch.

Hubble parameter, and α is a numerical factor of order unity that depends on the mass distribution within the system; hereafter, we neglect this constant, because it is irrelevant for our purpose. All self-gravitating systems in dynamical equilibrium thus satisfy the relation

$$\frac{\sigma_v}{R_v} = \sqrt{\frac{4}{9}\pi G \delta_v \rho_c} = \sqrt{\frac{\delta_v}{6}} H, \tag{3}$$

unlike random associations of unrelated galaxies. The trend of σ_v/R_v against the node number will thus show a discontinuity when walking on the binary tree from a bound structure to a region containing unrelated galaxies. As a robust estimate of R_v, we use the 2D average distance $d_{\rm avg} = 2\,R_v/n^{1/2}$, with

$$d_{\rm avg} = \frac{\Sigma_{i\neq j}\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{n(n-1)}, \tag{4}$$

where $x_{i,j}$ and $y_{i,j}$ are the Cartesian coordinates of the galaxies i and j on the plane of the sky.

By replacing R_v in Equation (3) with $d_{\rm avg}$ and dropping the factor 2, which is irrelevant for our purpose, we finally adopt the following quantity to identify the structures:

$$\eta = \frac{\sigma_v}{n^{1/2}\, d_{\rm avg}} \ {\rm km\ \ s^{-1}\,Mpc^{-1}}. \tag{5}$$

Figure 3 shows the typical trends of some of the quantities defined above with the node index; σ_v and $d_{\rm avg}$ have been arbitrarily rescaled to fit into the plot. The figure shows that all three quantities, the velocity dispersion σ_v, the galaxy number n, and the average distance $d_{\rm avg}$, increase when the branch includes more galaxies. In passing, we note that σ_v has an upper limit, outside the range shown in Figure 3, because we limit the redshift range of the sample to ± 4000 km s^{-1}. Also, a small $d_{\rm avg}$ makes η very large for compact systems.

When, by walking along a branch corresponding to a real structure, we start including interloping galaxies, the increased number of galaxies and the increased average distance are not compensated for by a proportional increase of σ_v: η will thus abruptly decrease with the node index. This downward jump of η can be used as a diagnostic for the identification of the structures. However, the amount of the decrease can substantially vary from case to case, depending on physical conditions,

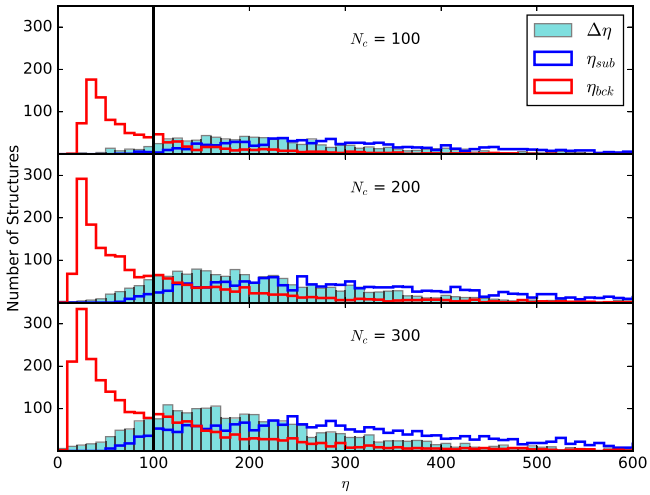

Figure 4. The distribution of $\eta_{\rm sub}$ from all 3D structures with at least 10 particles (open blue histogram), the distribution of $\eta_{\rm bck}$ from the region centered on each structure and extending to a radius five times larger than the radius of the 3D structure (open red histogram), and the distribution of $\Delta\eta = \eta_{\rm sub} - \eta_{\rm bck}$ (solid cyan histogram). The vertical line indicates our threshold $\Delta\eta = 100$. Each panel is for a different number N_c of member galaxies in the cluster, as indicated in each panel.

like local density, richness, and mass of the structure, and observational constraints, like completeness and survey density. A more suitable diagnostic is $\Delta\eta$, the difference between the value of $\eta_{\rm sub}$ associated with the structure and the value of $\eta_{\rm bck}$ associated with background region surrounding the structure.

We define $\eta_{\rm sub}$ as the value of η for each 3D structure with at least 10 galaxies in the FoV, and $\eta_{\rm bck}$ as the value of η of the region centered on each 3D structure and extending to a projected radius R_5, which is five times larger than the radius of the structure, estimated by the radius of the smallest circle enclosing the structure on the plane of the sky. To estimate $\eta_{\rm bck}$, we use Equation (5), where now n is the number of galaxies within the circle of radius R_5 and σ_v is their velocity dispersion. This definition of $\eta_{\rm bck}$ quantifies how the value of η of a 3D structure is affected by the inclusion of galaxies appearing in the surrounding area projected on the sky, which are likely to be unbound to the structure. This definition is more appropriate than, for example, choosing a random area in the FoV in the neighborhood of the structure, because the galaxy distribution is highly inhomogeneous on these cosmic scales, and the probability of selecting another structure in the random field is not negligible; the value of $\eta_{\rm bck}$ would thus be not representative of a random sample of unrelated galaxies.

Figure 4 shows that the distribution of $\eta_{\rm sub}$, the blue open histogram, is rather flat and mostly extends beyond $\eta_{\rm sub} = 100$; on the contrary, most $\eta_{\rm bck}$, the open red histogram, tend to be closer to zero because $d_{\rm avg}$ increases more rapidly than n and σ_v. The distribution of $\Delta\eta = \eta_{\rm sub} - \eta_{\rm bck}$, the solid histogram, qualitatively resembles the distribution of $\eta_{\rm sub}$.

The three panels in Figure 4 show the distributions of $\eta_{\rm sub}$, $\eta_{\rm bck}$, and $\Delta\eta$ for three values of the N_c member galaxies in the cluster, to mimic different densities of the redshift survey. The value of N_c slightly affects these distributions: the 10th, 50th, and 90th percentiles of the $\eta_{\rm sub}$ distributions are (160, 298, 574), (139, 297, 600), and (127, 288, 608) for $N_c = 100$, 200, and 300, respectively; similarly, the percentiles for the $\eta_{\rm bck}$ distributions are (32, 66, 227), (24, 70, 260), and (21, 73, 286),

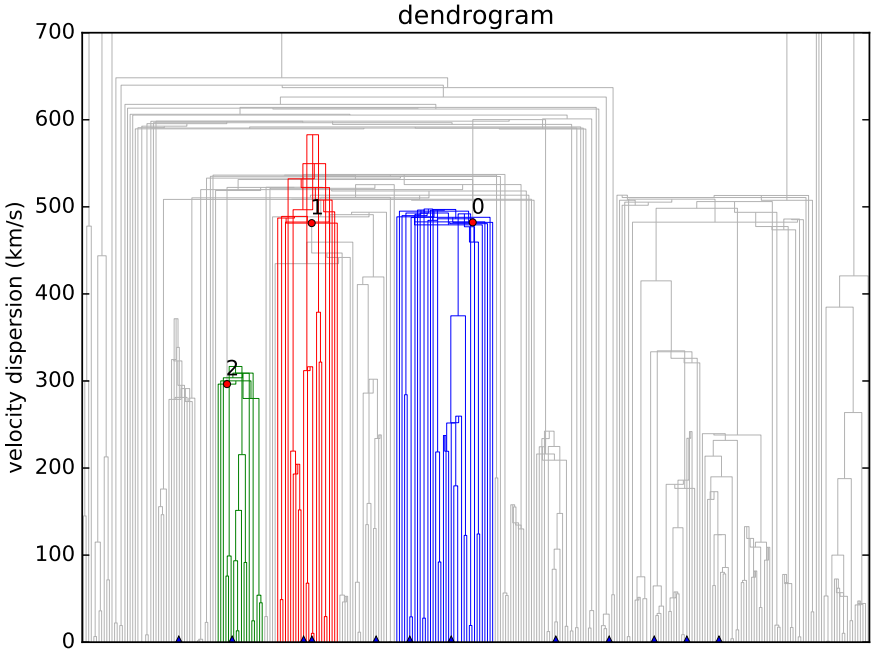

Figure 5. Dendrogram displayed with the velocity dispersion of every node on the y axis. All branches having a bud, or a local minimum of the binding energy, are labeled with blue triangles at the bottom of the panel. All key nodes found with the $\Delta\eta$ threshold, which have bloomed, are highlighted with red dots. The stems of their member leaves are indicated with different colors.

and for $\Delta\eta$ are (103, 213, 415), (89, 195, 404), and (69, 183, 400). To filter out random associations of unrelated galaxies and identify the list of structure candidates, we adopt a threshold for $\Delta\eta$. In Section 4.1, we define the success rate and completeness of the sample of 2D structures identified with the Blooming Tree Algorithm; we show how these two quantities vary with the threshold $\Delta\eta$. For $\Delta\eta = 100$, which is close to the 10th percentile of the $\Delta\eta$ distribution, the completeness is maximized. We thus choose this threshold for our following analysis. We have also tested that other values of this threshold, in the range $\Delta\eta = (50,120)$, leave our results substantially unaffected.

Given the similarity of the trends of $1/d_{\rm avg}$ and η shown in Figure 3, we might believe that the same argument used for η could be applied to $1/d_{\rm avg}$, and that $d_{\rm avg}$, rather than η, could be used as a diagnostic. This is not the case, however, because the two parameters σ_v and n, contained in η, partly balance the variation of $d_{\rm avg}$ with different densities of the redshift survey and different densities of the system itself: $d_{\rm avg}$ varies from compact to loose groups, which are both real systems, and it is different in the center and in the outer regions of clusters. Therefore, a single threshold on $d_{\rm avg}$, or $\Delta d_{\rm avg}$, is unable, unlike $\Delta\eta$, to identify structures in different environments and with different observational constraints.

To identify the 2D structures from the redshift data alone, the algorithm proceeds as follows: we explore all branches showing a bud or a local minimum of the binding energy profile. We compute $\Delta\eta$ along each branch from the leaves to the root, and we label as a key node the node before the last downward variation larger than the threshold $\Delta\eta = 100$. The key node from which at least six galaxies hang and no other key node hangs identifies a 2D structure: in other words, the bud associated with the branch containing this key node becomes a flower, and our tree blooms.

Figure 5 shows an example of this binary tree analysis for the same system shown in Figure 1. The corresponding distribution on the sky of the identified structures is shown in Figure 6: visually, the rich substructures and groups appear to be recovered at the proper position and with their proper size. In the next section, we compare the properties of the 2D structures with the 3D structures and provide a statistical analysis of the performance of our structure identification method.

4. Performance of the Blooming Tree Algorithm

4.1. Success Rate and Completeness

The results shown in this subsection are for the full sample of 50 merging clusters and 50 normal clusters projected along three orthogonal lines of sight and sampled with six different values of N_c for a total number of $(50 + 50) \times 3 \times 6 = 1800$ mock catalogs. We show the dependence of these results on the cluster dynamical state and on the FoV sampling in Section 4.2.

To quantify whether a 2D structure corresponds to a 3D structure, we make a one-to-one comparison between the members of the 2D structures identified with the Blooming Tree Algorithm and the members of the 3D structures identified with SUBFIND. The possibility of this one-to-one comparison is unique to the σ plateau algorithm and the Blooming Tree Algorithm. A single 2D structure may contain members belonging to different 3D structures or none.

For each 2D structure, we compute

$$f_{\rm 3D} = \frac{n({\rm mem_{3D}} \in {\rm mem_{2D}})}{n_{\rm 2D}}, \quad (6)$$

where $n_{\rm 2D}$ is the total number of members of the 2D structure, and $n({\rm mem_{3D}} \in {\rm mem_{2D}})$ is the largest number of particles, among the $n_{\rm 2D}$ members of the 2D structure, that are also members of a single 3D structure with $n_{\rm 3D}$ members.

The upper panel of Figure 7 shows the differential (histogram) and cumulative (solid line) distributions of $f_{\rm 3D}$.

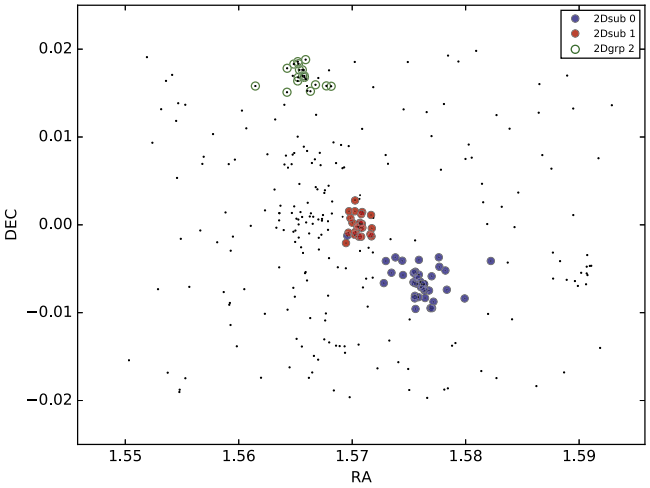

Figure 6. Distribution on the sky of the 2D structures identified with the Blooming Tree Algorithm. The color code is the same as in Figure 5.

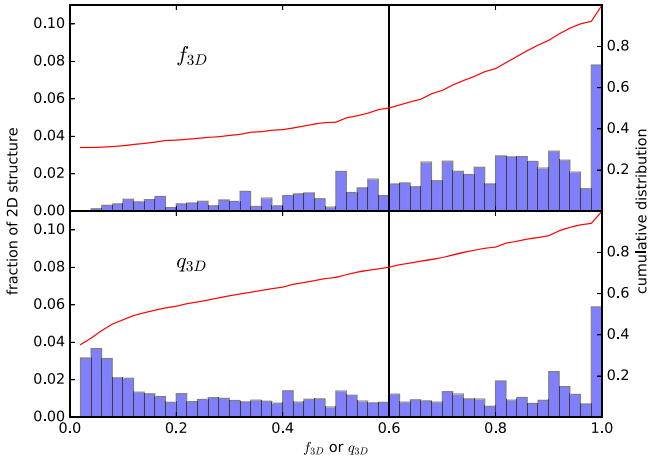

Figure 7. The upper panel shows the distribution of f_{3D}, the largest fraction of the total number of members of a 2D structure that are also members of a single 3D structure. The red solid line is the cumulative distribution function, whose value is shown on the right vertical axis. The lower panel shows the distribution and the cumulative distribution function of q_{3D}. In both panels, we omit the bar corresponding to a ratio f_{3D} or q_{3D} smaller than 0.02 for clarity, but its value can be read from the cumulative profile: it is 0.41 for f_{3D} and 0.45 for q_{3D}. The black vertical line indicates our threshold 0.6.

The initial value of the cumulative distribution is 0.41, and we thus see that 59% of the 2D structures contain at least one member of a 3D structure. The vertical line shows the value $f_{3D} = 0.6$ and crosses the cumulative distribution at the value 0.68, implying that 32% of the 2D structures have $f_{3D} > 0.6$. The median of f_{3D} is 0.38. When $f_{3D} > 0.6$, we assume that a 2D structure is successfully associated with a 3D structure. We thus define the 2D structures with $f_{3D} > 0.6$ as *successful* 2D structures. It can happen that different 2D structures contain members of the same 3D structure. This event occurs for 4% of the 2D structures. In these cases, we take the 2D structure containing the largest number of the 3D structure members as the possible match to the 3D structure.

We also compute

$$q_{3D} = \frac{n(\mathrm{mem}_{3D} \in \mathrm{mem}_{2D})}{n_{3D}} \quad (7)$$

to quantify to what extent a 3D structure is included in a 2D structure: in fact, q_{3D} is the ratio between the largest number of particles, among the n_{2D} members of the 2D structure, that are also members of a single 3D structure with n_{3D} members, and n_{3D} itself. Therefore, unlike f_{3D}, q_{3D} keeps the information on those 3D structures that are mostly or fully contained in a 2D structure, even if the 2D structure is not successful. These 2D structures might contain too many interlopers or might be too composite to be considered successful 2D structures; however, they still contain a substantial fraction of a 3D structure, and they have thus succeeded in locating its presence. The lower panel of Figure 7 shows the differential (histogram) and cumulative (solid line) distributions of q_{3D}: by looking at the values of the cumulative distribution indicated on the right vertical axis, we see that 5.9% of the 2D structures contain complete 3D structures, 11.9% of the 2D structures contain more than 90% of the members of a single 3D structure, and 36.0% of the 2D structures have q_{3D} larger than 0.6. According to Equation (7), when $q_{3D} > 0.6$, more than 60% of the members of the 3D structure are included in the 2D structure.

According to our definitions, a one-to-one correspondence between a 2D structure and a 3D structure occurs when the conditions $f_{3D} > 0.6$ and $q_{3D} > 0.6$ are satisfied at the same time on the same pair of 2D and 3D structures. This combined condition occurs for 52% and 61% of the successful 2D structures for the normal and merger cluster samples, respectively. The remaining fraction of successful 2D structures, with $q_{3D} < 0.6$, are almost exclusively associated with the cluster cores: 39% and 25% for the normal and merger cluster samples, respectively. In principle, a one-to-one correspondence would also be guaranteed when $f_{3D} > 0.5$ and $q_{3D} > 0.5$. However, we prefer to adopt those more restrictive thresholds to suppress the effect of noise around the 0.5 value of the thresholds.

To quantify the performance of our Blooming Tree Algorithm, we define the success rate and the completeness of the 2D structure sample. The success rate is the ratio between the number of successful 2D structures and the total number of 2D structures:

$$\text{Success Rate} = \frac{\text{No. of 2D structures with } f_{3D} > 0.6}{\text{Total no. of 2D structures}}. \quad (8)$$

To estimate the completeness, we only consider the successful 2D structures ($f_{3D} > 0.6$). Each successful 2D structure has an associated 3D structure. The completeness is the ratio between the number of these identified 3D structures and the total number of 3D structures in the FoV:

$$\text{Completeness} = \frac{\text{No. of identified 3D structures}}{\text{Total no. of 3D structures}}. \quad (9)$$

Figure 8 shows the success rate as a function of n_{2D}, the number of members of the 2D structures (solid line), and the distributions of n_{2D} (histogram). The success rate, namely the probability that a 2D structure identifies a 3D structure, clearly is proportional to n_{2D} in the range $n_{2D} \lesssim 80$. The success rate flattens out, on average, at larger n_{2D}. We remove those 2D structures that are unlikely to correspond to 3D structures by setting a lower limit to n_{2D}. We adopt the threshold $n_{2D} = 10$: according to Figure 8, for $n_{2D} \geqslant 10$, the success rate is always larger than 30%.

With the definitions of success rate and completeness at hand, we can now show how these two quantities vary with the

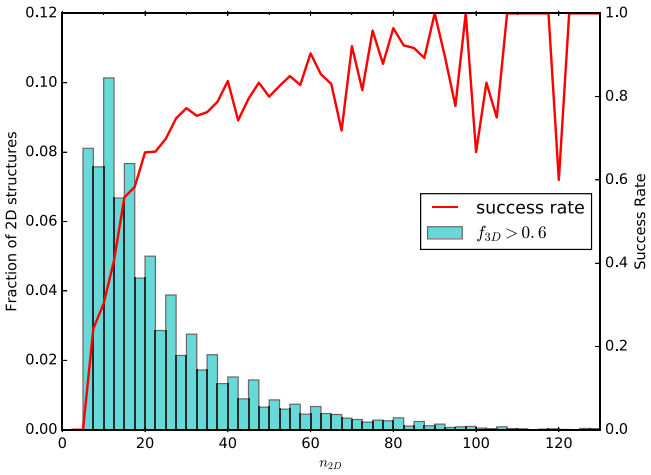

Figure 8. The cyan solid histogram shows the distribution of n_{2D} of the successful 2D structures, those with $f_{3D} > 0.6$. The solid red line shows the success rate, the ratio between the number of successful 2D structures and the total number of 2D structures as a function of n_{2D} of the 2D structures. The 15 2D structures with more than 130 members, which represent 0.2% of the total sample, are not shown for clarity.

value of the threshold adopted for $\Delta\eta$ (Section 3.3). By increasing the threshold $\Delta\eta$, the Blooming Tree Algorithm identifies a decreasing number of 2D structures, whose probability of being successful 2D structures increases. At the same time, the number of members of the 2D structures n_{2D} decreases. The combination of these two effects makes the completeness peak at $\Delta\eta \sim 100$ for $N_c > 150$, as shown in the top panel of Figure 9. On the contrary, the number of 2D structures decreases with $\Delta\eta$ more rapidly than the number of successful 2D structures; it follows that the success rate monotonically increases with $\Delta\eta$, as shown in the bottom panel of Figure 9. The two panels show that the successful rate increases at the expense of the completeness. We thus adopt the threshold $\Delta\eta = 100$, which maximizes the completeness for $N_c > 150$. The Blooming Tree Algorithm appears to be robust against the value of this threshold: we have tested that the results presented in this work for $\Delta\eta = 100$ remain substantially unaffected by adopting $\Delta\eta$ in the range (50, 120), where 50 and 120 maximize the completeness for $N_c = 50$ and $N_c = 300$, respectively.

4.2. 2D versus 3D Structures

We now illustrate the performance of our Blooming Tree Algorithm in identifying the 3D structures present in the FoV.

Table 2 lists the success rate and the completeness for our two cluster samples for different N_c, the number of galaxies within a sphere of 6 h^{-1} Mpc from the cluster center; N_{2D} is the average number of structures in the FoVs in addition to the cluster. For completeness, we distinguish between substructures, cores, and surrounding groups. For a typical density of the redshift survey $N_c = 200$ (e.g., Rines & Diaferio 2006; Rines et al. 2013), Table 2 shows, for example, that the Blooming Tree Algorithm recovers 79.6% of the real substructures and 59.8% of the surrounding groups.

Figure 10 shows the success rate against N_c for successful 2D structures ($f_{3D} > 0.6$) and for 2D structures with $f_{3D} > 0.3$. In this latter case, the success rate is the ratio between the number of 2D structures with $f_{3D} > 0.3$ and the total number of 2D structures. Figure 10 also shows the completeness for 3D

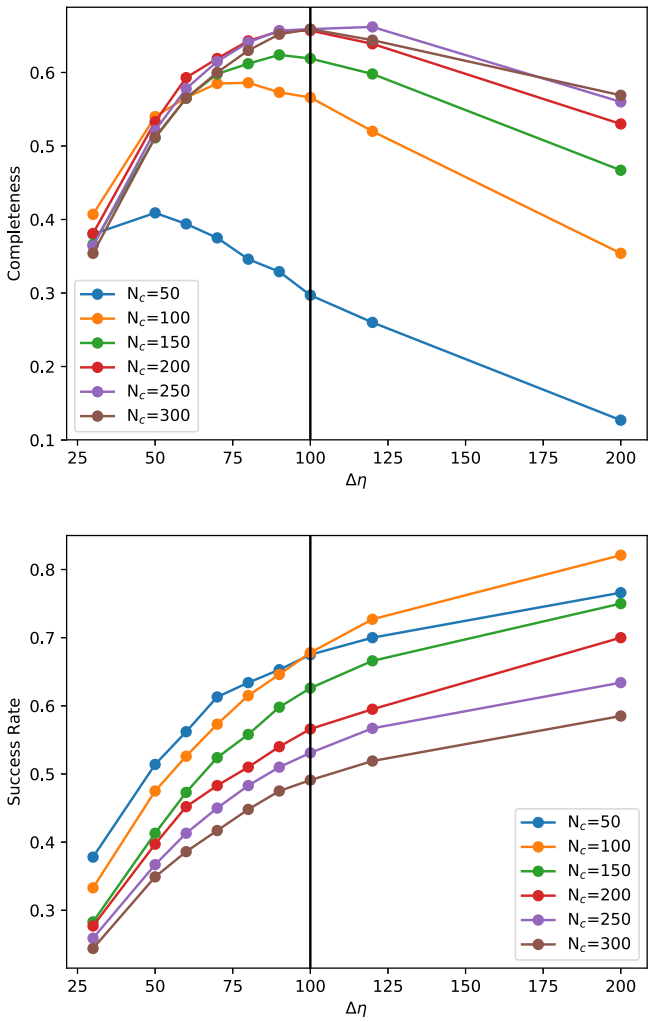

Figure 9. Completeness and success rate as a function of the threshold $\Delta\eta$.

structures associated with 2D structures with $f_{3D} > 0.6$ or $f_{3D} > 0.3$.

The success rate decreases in dense FoVs (larger N_c) because of the increasing number of chance alignments. On the contrary, the completeness basically is independent of N_c, except for the poorest fields with $N_c = 50$, where the smaller number of galaxies in the FoV reduces the probability of identifying the structures. These results show that structures are satisfactorily identified in dense fields with roughly 100–150 galaxies within 6 h^{-1} Mpc and within 4000 km s^{-1} from the cluster center: somewhat counterintuitively, increasing the number of galaxies does not increase the completeness and might actually decrease the success rate.

This result cannot be considered a shortcoming of the Blooming Tree Algorithm *tout court*: gravity is an infinite range force, and the definition of the borders of a 3D structure is debatable. The trend of the success rate with N_c also indicates that, in the presence of dense redshift surveys, the Blooming Tree Algorithm is more generous than the 3D structure identification algorithm in finding structures and assigning members to them. However, the Blooming Tree Algorithm also returns a completeness that is independent of N_c when $N_c > 100$, demonstrating the quite relevant ability of the Blooming Tree Algorithm to recover the 3D structures independently of the redshift survey density.

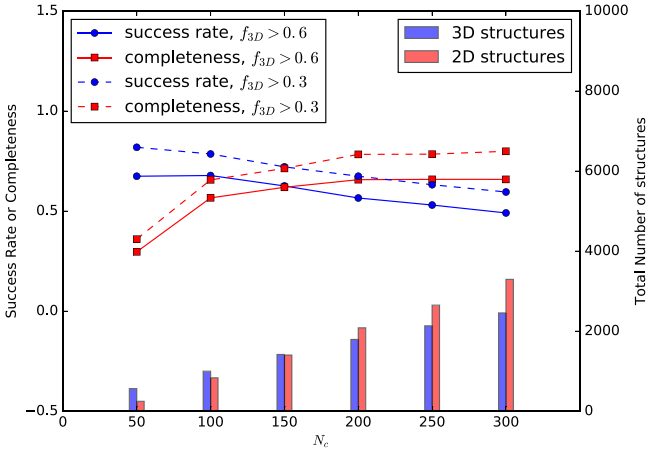

Figure 10. Success rate (blue dots) and completeness (red squares) against N_c. The blue lines show the success rate for the 2D structures with $f_{3D} > 0.6$ (solid line) and with $f_{3D} > 0.3$ (dashed line). The red lines show completeness for the 3D structures associated with the 2D structures with $f_{3D} > 0.6$ (solid line) and with $f_{3D} > 0.3$ (dashed line). The blue (red) bars at the bottom show the total number of 3D (2D) structures. The larger number of 2D structures indicates that some 2D structures are spurious; in other words, they do not correspond to any 3D structure, and the relative number of these spurious 2D structures is smallest when $N_c = 150$.

Table 2
Success Rate and Completeness ($\Delta\eta = 100$)

N_c	Cluster	N_{2D}	Success rate (%)	Completeness (%)			
				tot	core	subs	groups
50	Normal	0.6	75.0	27.6	28.7	29.1	25.5
	Merging	1.0	63.1	31.3	21.5	25.9	40.9
	Total	0.8	67.5	29.7	25.2	27.2	33.6
100	Normal	2.3	68.8	54.3	80.7	76.0	40.7
	Merging	3.3	67.1	58.5	66.0	59.8	57.0
	Total	2.8	67.8	56.6	73.3	65.6	48.8
150	Normal	3.5	65.2	53.9	92.0	77.7	43.6
	Merging	5.8	61.0	68.5	90.0	77.7	61.9
	Total	4.7	62.6	61.9	91.0	77.7	52.8
200	Normal	5.5	57.6	58.8	96.7	79.9	52.0
	Merging	8.4	56.0	71.4	90.7	79.4	67.2
	Total	7.0	56.6	65.7	93.7	79.6	59.8
250	Normal	7.1	54.2	58.8	98.7	85.8	51.3
	Merging	10.6	52.3	72.0	94.0	81.3	67.7
	Total	8.8	53.1	65.9	96.3	83.0	59.6
300	Normal	8.9	49.7	58.8	99.3	84.8	52.7
	Merging	13.1	48.7	71.9	96.7	82.5	67.7
	Total	11.0	49.1	65.9	98.0	83.3	60.3

Figure 11 shows the success rate and the completeness for the merging and normal clusters separately. The success rate drops with increasing N_c in both samples, whereas the completeness is almost unaffected by N_c. In merging systems, the galaxy distribution is more clumpy than in normal clusters. Therefore, in merging systems, the Blooming Tree Algorithm appears to be more effective, and the completeness in this sample is systematically larger than in normal clusters. The

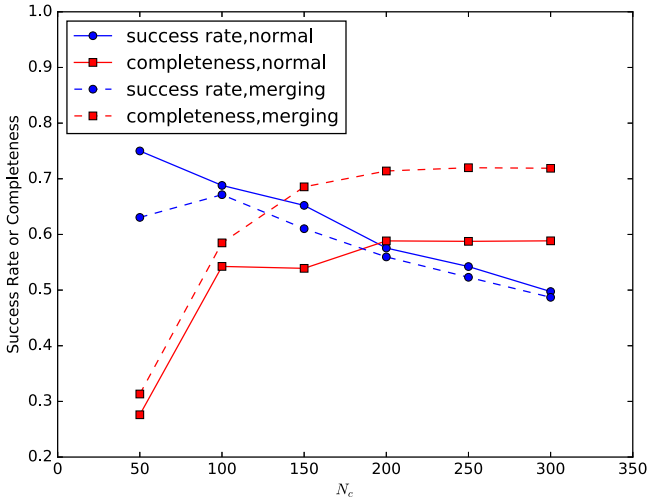

Figure 11. Success rate (blue dots) and completeness (red squares) for the merging (solid lines) and normal (dashed lines) clusters separately.

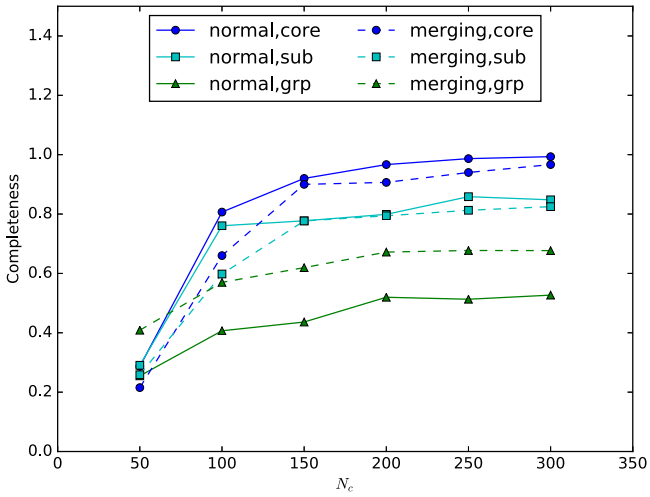

Figure 12. Completeness of different types of structures: cluster cores (blue dots), substructures (cyan squares), and surrounding groups (green triangles). Solid (dashed) lines are for normal (merging) clusters.

structures that contribute to the larger completeness in merging clusters are the surrounding groups, as shown in Figure 12.

Figure 12, in fact, shows the completeness against N_c of different types of structures in the two cluster samples: cores, substructures, and surrounding groups. In dense FoVs, the cluster cores are the easiest structures to identify, especially in normal clusters. Substructures require a trade-off between sufficiently dense FoVs, to enable their correct identification, and sufficiently sparse FoVs, to minimize the interloper contamination. Although the surrounding groups are the most frequent structures in dense FoVs, they show the smallest completeness (∼50%): they are looser systems and are easily affected by interloper contamination.

Figure 13 shows the completeness as a function of the mass of the 3D structure: massive structures with more bound members and deeper gravitational wells are more easily detected. Figure 13 shows that, in principle, we could improve the completeness of our structure sample by simply dropping the less massive structures. In the less dense FoVs ($N_c = 100$ and $N_c = 200$), the smallest mass bin is systematically larger than the second smallest mass bin: this effect is due to the fact

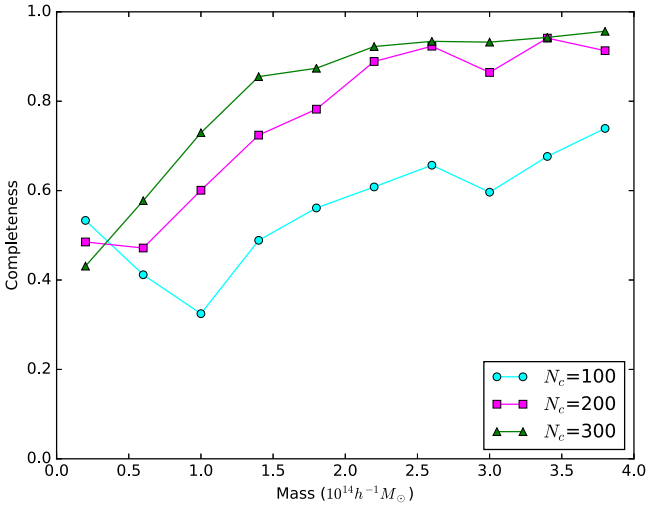

Figure 13. Completeness against the 3D structure mass for different N_c.

that we remove all 3D structures with less than 10 member galaxies from the sample of the 3D structures, as we mention in Section 2. In fact, in the densest FoV ($N_c = 300$), this phenomenon disappears.

4.3. Discussion

From Table 2 and Figure 10, we see that if we reduce the success threshold to $f_{3D} > 0.3$ (Figure 7), the success rate increases from a value between 49% and 68% to a value between 60% and 82%, and the completeness increases from a value between 30% and 66% to a value between 36% and 80%.

In addition, to be conservative, we only consider 3D structures with more than 10 particles in the fields (see Section 2). However, there are many 2D structures corresponding to small 3D structures with at least five particles; unfortunately, these small 3D structures are too small to increase f_{3D} to 0.6 and thus to make the 2D structure successful. Nevertheless, if we take them into account, the spurious 2D structures, namely 2D structures that do not contain any 3D structure members, will be reduced from the current 30%–15%.

Finally, all of these results on the success rate and completeness of our Blooming Tree Algorithm are based on the SUBFIND substructure-detecting algorithm, which is only one of the many halo-finding methods adopted in *N*-body simulations. As tested by Knebe et al. (2013), different halo-finding methods can give different numbers of structures with a mass scatter up to 20%. Therefore, our strict one-to-one comparison results are certainly affected by the reference algorithm we adopt in the 3D simulation. Especially in the central part, many 2D structures turn out to be failures, because the 3D halo finder algorithm associates those particles with the central core. However, these 2D structures might well be dynamically distinct from the core. These aspects require different analyses that go beyond the scope of the present paper.

The above discussion clearly shows that our astrophysical problem of identifying 3D structures from the three phase-space coordinates accessible to observations cannot be straightforwardly equated to the standard problems of clustering and classification problems described in the classical literature of cluster analysis (e.g., Everitt et al. 2011; Hennig et al. 2015). In particular, because of the numerous possible

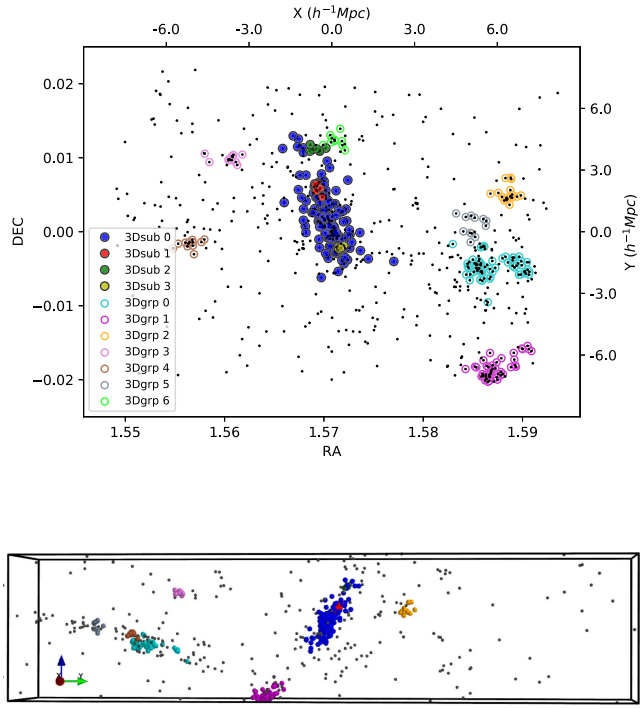

Figure 14. Distribution of the 3D structures in the FoV of a massive cluster with 722 particles. In the top panel, the projected celestial coordinates are in radiants; comoving coordinates in the *N*-body simulation are also shown. The solid dots show the members of the structures with different colors; the core particles are in blue. The open circles show the members of the surrounding groups. The bottom panel provides a 3D view within a box with dimensions $14 \times 14 \times 60\,h^{-3}$ Mpc^3. Group 6 (open green circles) is outside this box.

definitions of the 3D structures and the successful 2D structures, optimization methods based on a minimization or maximization of a single numerical quantity cannot be adopted. Similarly, adopting classical tools, like the receiver operating characteristic (ROC) curve and the area under curve (AUC) statistic to quantify the algorithm efficiency, is unfeasible, because some of the standard quantities used for their estimate, like the number of *true negative* elements of the data set, cannot be defined in our problem.

5. Comparison with the σ Plateau Algorithm

We now compare our Blooming Tree Algorithm for the identification of structures with the σ plateau algorithm. We will first illustrate the differences on a representative case, and we will then consider the statistical properties of a large cluster sample.

5.1. A Representative Case

We consider a massive cluster with $M_{200} = 8.66 \times 10^{14}\,h^{-1}\,M_{\odot}$, with 722 galaxies in the square FoV of size equal to $3\,R_{200}$; 200 out of these 722 galaxies are cluster members. The weak point of the σ plateau algorithm is its difficulty in identifying structures with widely different velocity dispersions (Yu et al. 2015). A massive cluster like the one we choose here provides structures with different mass and size and provides thus a good test for the performance of the two algorithms.

There are four 3D substructures, including the cluster core, and seven 3D surrounding groups in the FoV, as shown in Figure 14. The distribution of these structures in the binary tree, shown in Figure 15 with different colors, indicates that the

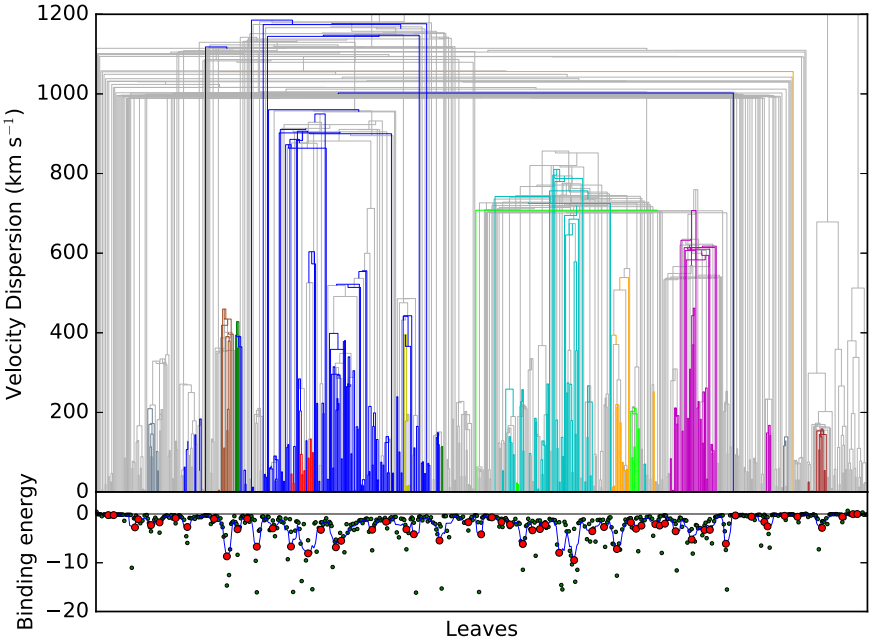

Figure 15. Binary tree of the simulated massive cluster with 722 particles in the FoV shown in Figure 14. The inset at the bottom shows the binding energy profile. The colors indicate the real members of the different 3D structures, according to Figure 14.

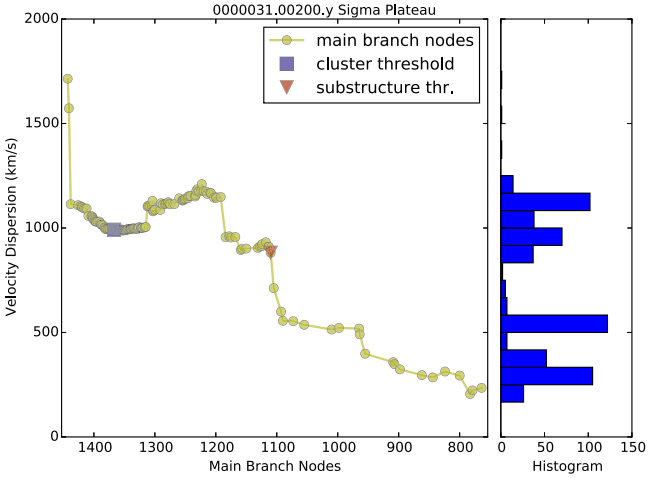

Figure 16. Velocity dispersion of the leaves of each node along the main branch of the binary tree shown in Figure 15. The blue square and the red triangle are the first and second threshold, respectively. The curve in between is the σ plateau, whose position generally corresponds to the peak of the distribution of node numbers with similar velocity dispersion shown in the right panel. This peak does not stand out clearly here because of the existence of other plateaus.

member galaxies of these structures tend to appear on the same branch of the tree; however, their velocity dispersions are in the range ∼200–1000 km s^{-1}. As mentioned above, this large velocity dispersion range is the main challenge for the σ plateau algorithm.

According to the σ plateau algorithm (Yu et al. 2015), by walking on the main branch of the tree, we determine the σ plateau shown in Figure 16. The large range of velocity dispersions, which is a consequence of the complex dynamics of clusters like this one, prevents the plateau from being flat. Therefore, locating the second threshold in this kind of system becomes rather ambiguous.

Nevertheless, the results of the σ plateau algorithm, shown in the top panel of Figure 17, are satisfying. The algorithm

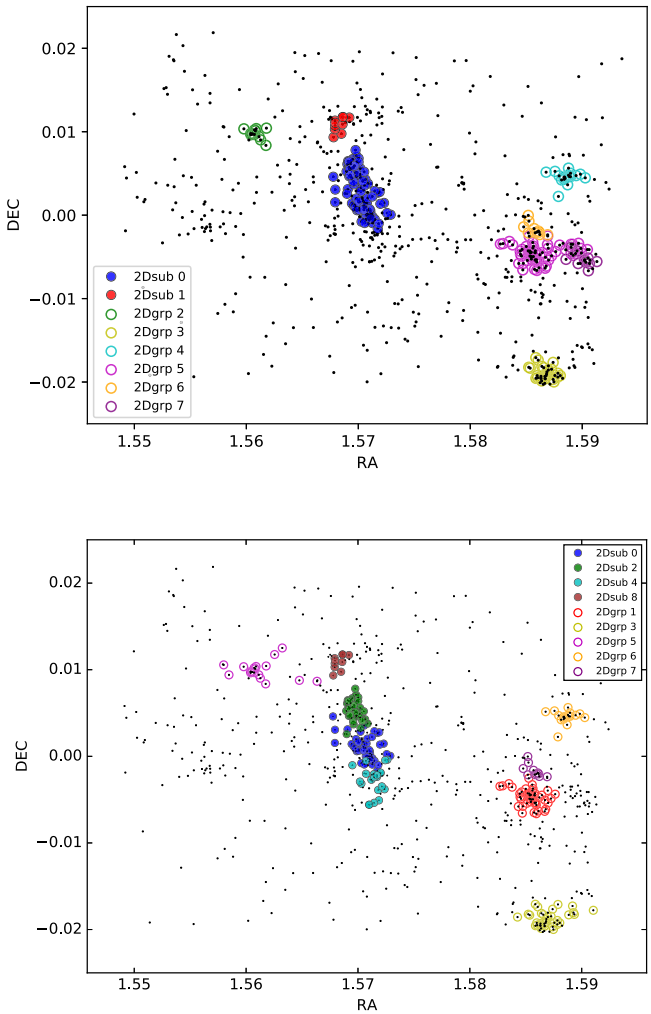

Figure 17. Sky diagrams of structures identified by the σ plateau algorithm (top panel) and the Blooming Tree Algorithm (bottom panel). Galaxies with the same color belong to the same 2D structure. The galaxies in the bottom panel are colored according to colors of the binary tree in Figure 18.

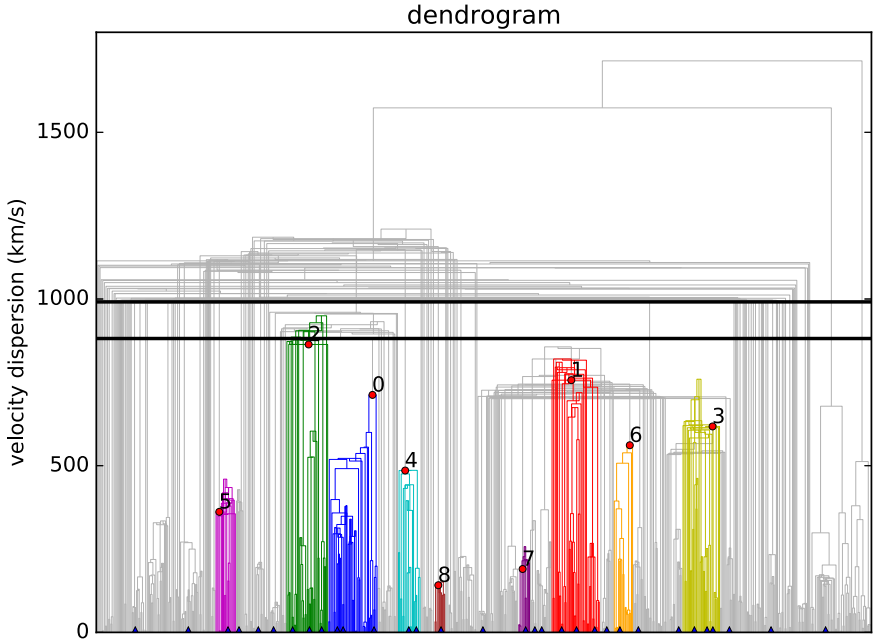

Figure 18. Dendrogram with the node velocity dispersion on the vertical axis. The blue triangles at the bottom show the buds, the binding energy minima that identify the branches that need to be searched. The red dots show the blooms, all of the key nodes found with the $\Delta\eta$ threshold, that identify the 2D structures. The stems of their member leaves are shown with different colors. The structure indexes shown on the plot are sorted according to decreasing number of members. The two horizontal black solid lines are the thresholds of the σ plateau algorithm.

recovers two substructures out of four and six surrounding groups out of seven. The first 2D substructure (2Dsub 0) is the cluster core, whereas the second 2D substructure (2Dsub 1) includes the 3D substructure 3Dsub 2; however, it does not satisfy the $f_{3D} > 0.6$ criterion and cannot be considered a successful 2D substructure. The remaining 3D substructures, 3Dsub 1 and 3Dsub 3, remain unidentified, because they are located in the cluster center and the method is unable to separate them from the core. This result is a consequence of the fact that the σ plateau algorithm cuts the binary tree with a single velocity dispersion threshold and can thus only identify structures whose velocity dispersion is close to this threshold: if a substructure has a small number of members, it cannot generate a significant plateau, it is not recognized by the algorithm as a distinct structure, and it will thus be included in a larger system.

Figure 18 and the bottom panel of Figure 17 show the results of our Blooming Tree Algorithm, based on tracing all tree branches rather than the main branch alone. The two solid lines in Figure 18 indicate the two thresholds of the σ plateau approach. The new algorithm is able to pick up structures with different velocity dispersions. It recovers all four substructures; it also recovers five out of the seven 3D surrounding groups.

The 2D substructure 2Dsub 0, which identifies the cluster core, has a velocity dispersion corresponding to the second threshold of the σ plateau. With the first key node below the σ plateau, the algorithm can identify the 2D substructure 2Dsub 2: it includes the 3D substructure 3Dsub 1 that was unidentified by the σ plateau algorithm. Finally, the 2D substructure 2Dsub 4 includes the 3D substructure 3Dsub 3, and the 2D substructure 2Dsub 8 includes the 3D substructure 3Dsub 2.

All of the 2D structures identified by the Blooming Tree Algorithm are located at positions consistent with the corresponding 3D structures. However, only four groups, 2Dgrp 1, 2Dgrp 3, 2Dgrp 5, and 2Dgrp 6, out of five satisfy the $f_{3D} > 0.6$ criterion. All three 2D substructures other than the core (the 2D substructure 2Dsub 1) are contaminated by core members and fail the $f_{3D} > 0.6$ criterion. Therefore, even when the 2D and 3D structures share the same position on the sky, this strict criterion on f_{3D} does not allow the association of the 2D structure to the corresponding 3D structure, and the 2D structure is classified as spurious. This example shows that the $f_{3D} > 0.6$ criterion guarantees a robust identification of the 3D structures, but it returns a lower limit to the performance of our algorithms for the identification of structures.

5.2. Performance on Two Cluster Samples

We analyze the two samples of simulated clusters used in Yu et al. (2015) with both algorithms. The two samples are a massive sample (M15) containing 150 mock redshift surveys, with M_{200} ranging from $0.86 \times 10^{15}\,h^{-1}\,M_{\odot}$ to $3.4 \times 10^{15}\,h^{-1}\,M_{\odot}$, and a less massive sample (M14) with the same number of clusters, with mass ranging from $0.95 \times 10^{14}\,h^{-1}\,M_{\odot}$ to $1.1 \times 10^{14}\,h^{-1}\,M_{\odot}$. Each cluster is sampled with a given number of galaxies within $3\,R_{200}$: $N_c = (100, 200, 300)$.

Figure 19 shows the results of this analysis. It shows the average number of 2D structures per cluster as a function of N_c. The left and right bars are for the σ plateau algorithm and the Blooming Tree Algorithm, respectively. The Blooming Tree Algorithm identifies five to ten times more structures than the σ plateau algorithm, with more structures identified in the M14 sample. In fact, because of our strategy for the mock catalog creation (see Yu et al. 2015), the fields of the M14 clusters are denser than the M15 fields, and detecting 2D structures is more likely. The two algorithms identify a comparable number of substructures (blue sectors of the bars), whereas the Blooming Tree Algorithm is more efficient at separating the cores into distinct structures (cyan sectors of the bars). The Blooming

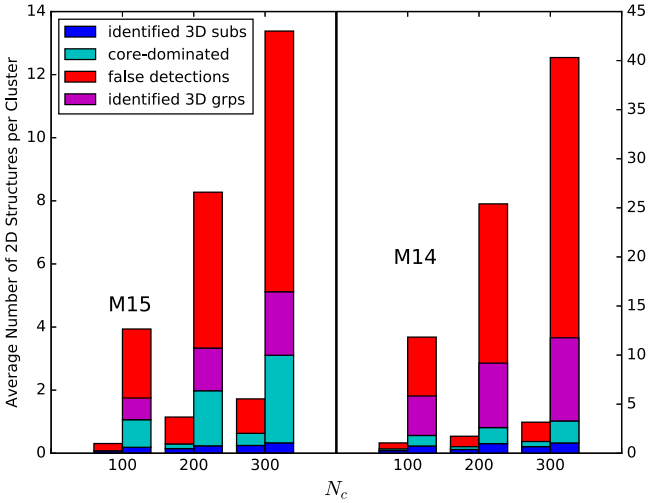

Figure 19. Average number of 2D structures per cluster as a function of N_c: left (right) panels are for the M15 (M14) sample, and left (right) bars are for the σ plateau algorithm (Blooming Tree Algorithm). The blue part of each bar shows the 2D substructures associated with 3D substructures ($f_{3D} > 0.6$); the cyan part shows the 2D structures associated with 3D cores ($f_{3D} > 0.6$); the red part shows the spurious 2D structures ($f_{3D} < 0.6$). The purple part shows the 2D groups associated with 3D groups ($f_{3D} > 0.6$): they are missing in the left bars because the 3D groups were not included in the σ plateau analysis of Yu et al. (2015).

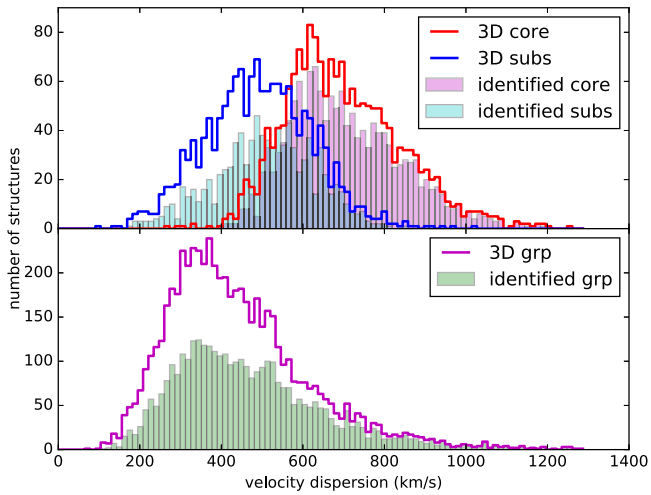

Figure 20. Distributions of the velocity dispersions of different types of structures in the combined sample of 50 normal clusters and 50 merging clusters: cores of clusters (red open histogram), substructures (blue open histogram), and surrounding groups (magenta open histogram). The solid histograms are the 3D structures identified by the successful 2D structures with $f_{3D} > 0.6$ according to the Blooming Tree Algorithm in the 300 mock catalogs created by projecting each cluster along three orthogonal lines of sight: core of clusters (magenta), substructures (cyan), and surrounding groups (green).

Tree Algorithm also identifies the surrounding groups (purple sectors), which are not included in the σ plateau analysis.

The red sectors show the false detections, the 2D structures with $f_{3D} < 0.6$: for the Blooming Tree Algorithm, they represent ∼56%–62% of the 2D structures in the massive clusters, depending on N_c, and ∼67%–77% of the 2D structures in the less massive clusters. The σ plateau algorithm has a substantially comparable performance, but on a substantially smaller number of 2D structures: those fractions become ∼40%–70% and ∼60%, respectively. These fractions would clearly decrease by adopting a weaker criterion than $f_{3D} > 0.6$. The representative case illustrated in Section 5.1 shows that this criterion might indeed be too strict.

The Blooming Tree Algorithm improves the completeness of the σ plateau algorithm by roughly a factor of three. In the M14 sample, the Blooming Tree Algorithm identifies 68% of the cluster substructures and cores, compared to 25% of the σ plateau. In the M15 sample, the improvement is even more dramatic, with a completeness of 77% for the Blooming Tree Algorithm and 5%–20% for the σ plateau.

Overall, the M15 sample has larger success rates than M14, because in massive clusters, cores and substructures are more massive and are thus easier to identify. In addition, the fields of the M14 clusters are denser than the M15 fields, and the probability of detecting 2D structures that do not correspond to any 3D substructure increases.

As mentioned above, the origin of the different performances between the two algorithms is the wide distribution of the velocity dispersions of the structures. Figure 20 shows the distributions of the velocity dispersions of the three kinds of structures in our combined sample of 50 normal clusters and 50 merging clusters: cores (red open histogram), substructures (blue open histogram), and surrounding groups (magenta open histograms). Unlike the σ plateau algorithm, which searches the main branch alone and only relies on the velocity dispersion of the leaves on the main branch, the Blooming Tree Algorithm searches all tree branches and combines three quantities, velocity dispersion, richness, and size of each node, into the physically motivated quantity η (Equation (5)) to define the identification criterion $\Delta\eta > 100$ (discussed in Sections 3.3 and 4.1). By this deeper analysis of the physical properties of the branches of the binary tree, the Blooming Tree Algorithm can identify structures with largely different velocity dispersions, as shown by the solid histograms in Figure 20. This feature does not belong to the σ plateau algorithm, as shown in Figure 14 of Yu et al. (2015).

6. Conclusion

We present the Blooming Tree Algorithm, a new algorithm for the detection of cluster substructures and surrounding groups with optical spectroscopic data. The Blooming Tree Algorithm is a substantial improvement over our previous procedure, the σ plateau algorithm, whose performance is described in Yu et al. (2015).

Both algorithms first arrange the galaxies in the FoV in a binary tree according to the estimate of the binding energy of each galaxy pair, and they search for structures associated with the individual branches of the tree. Neither algorithm thus requires an assumption on the geometry of the systems, on their velocity field or on their dynamical state, or an initial guess of the position and size of the structures to identify.

The Blooming Tree Algorithm improves over the σ plateau algorithm because it searches all branches of the binary tree and relies on a physically motivated combination of velocity dispersion, richness, and size of the candidate structures to identify them. Unlike the σ plateau algorithm, which only searches the main branch alone and relies on the velocity dispersion alone, these improvements enable the Blooming Tree Algorithm to identify structures with widely different velocity dispersions and to increase substantially the efficiency of the structure identification. It also enables the Blooming Tree Algorithm to identify easily the galaxy groups present in the

cluster outskirts. This ability is relevant for the quantitative investigation of the merging and accretion history of galaxy clusters (Rines et al. 2001; Lemze et al. 2013; De Boni et al. 2016).

Because both methods are based on the arrangement of the galaxies in a hierarchical binary tree, the identified structures are naturally nested into each other, and by walking on the tree branches, we can actually separate each individual structure into smaller and smaller dynamically distinct substructures. The binary tree also easily returns a list of the members of the identified structures, a piece of information that is necessary to estimate the properties of the structures, like velocity dispersion, size, and mass.

We test the Blooming Tree Algorithm on 300 mock redshift surveys of 100 clusters of mass $\sim 5 \times 10^{14}\, h^{-1}\, M_{\odot}$ extracted from an N-body simulation of a ΛCDM model. We consider 50 *merging* clusters, whose most massive substructure other than the cluster core has a mass at least half the mass of the core, which is the substructure whose center coincides with the cluster center. We also consider 50 *normal* clusters, where that condition is not verified. We also explore mock surveys of different density by varying the number N_c of galaxies within a sphere of radius $6h^{-1}$ Mpc from the cluster center in the range $N_c = [50–300]$. We only consider substructures and surrounding groups with mass larger than $10^{13}\, h^{-1}\, M_{\odot}$.

A substantial fraction of the 3D structures are correctly identified by the Blooming Tree Algorithm. Disregarding the case $N_c = 50$, which returns too-sparse redshift surveys, the completeness of the substructure catalogs is ∼80% for both the normal and merging clusters. A large completeness is also obtained for the surrounding groups, with ∼50% and ∼60% for the normal and merging clusters, respectively.

The completeness is almost independent of N_c, provided that $N_c > 50$. The density of the redshift survey affects the number of spurious structures, because with a larger number of galaxies in the FoV, the probability of detecting false structures increases. In fact, when increasing N_c from 50 to 300, the success rate, the fraction of 2D structures that correspond to real 3D structures, drops from 75% to 50% for the normal clusters, and from 63% to 49% for the merging clusters.

These results are rather impressive, because the clusters are extracted from the N-body simulation without any particular criterion in addition to the mass of the cluster and of the largest substructure.

When considering substructures, surrounding groups, and cluster cores, the Blooming Tree Algorithm has an overall completeness of ∼60% and a success rate in the range ∼50%–70%, substantially larger than the completeness and success rate of the σ plateau algorithm. This latter algorithm has a completeness ∼30%–50% and a success rate ∼15%–20%, depending on the mass and the dynamical state of the cluster (Yu et al. 2015).

Clearly, for each galaxy, we can only measure three phase-space coordinates out of six, and the estimate of the binding energy of each pair of galaxies, on which the Blooming Tree Algorithm is based, can be heavily affected by projection effects. In addition, the galaxy peculiar velocities are unknown, and the redshift difference entirely contributes to the estimated kinetic energy of the galaxy pair. Nevertheless, the good performance of the Blooming Tree Algorithm shows that these limitations little affect the identification procedure and that this algorithm is a powerful tool to identify the substructures of clusters and their surrounding groups when dense redshift surveys of clusters, like CIRS (Rines & Diaferio 2006) and HeCS (Rines et al. 2013), are available. The Blooming Tree Algorithm can thus be a powerful tool to infer the merging history of clusters, investigate their dynamics, and constrain their formation models (Yu et al. 2016).

A.D. thanks Margaret Geller for fruitful discussions and suggestions. We sincerely thank an anonymous referee whose comments and suggestions urged us to improve substantially the presentation of our results. We acknowledge support from the grant I@UNITO of the Italian Ministry of Education, University and Research assigned to the University of Torino, the INFN grant InDark, the National Natural Science Foundation of China under Grant 11403002, the Fundamental Research Funds for the Central Universities, and the Scientific Research Foundation of Beijing Normal University. M.B. also acknowledges the financial contribution by the PRIN INAF 2012 "The universe in the box: multiscale simulations of cosmic structure," and the support from the Italian Ministry for Education, University and Research (MIUR) through the SIR individual grant SIMCODE, project number RBSI14P4IH.

ORCID iDs

Heng Yu (余恒) https://orcid.org/0000-0001-8051-1465
Antonaldo Diaferio https://orcid.org/0000-0002-4986-063X
Marco Baldi https://orcid.org/0000-0003-4145-1943

References

Aguerri, J. A. L., & Sánchez-Janssen, R. 2010, A&A, 521, A28
Agulli, I., Aguerri, J. A. L., Sánchez-Janssen, R., et al. 2016, MNRAS, 458, 1590
Audit, E., Teyssier, R., & Alimi, J.-M. 1998, A&A, 333, 779
Baldi, M. 2012, MNRAS, 422, 1028
Barmby, P., & Huchra, J. P. 1998, AJ, 115, 6
Bird, C. 1994, ApJ, 422, 480
Colberg, J. M., Krughoff, K. S., & Connolly, A. J. 2005, MNRAS, 359, 272
Colberg, J. M., White, S. D. M., Jenkins, A., & Pearce, F. R. 1999, MNRAS, 308, 593
Colless, M., & Dunn, A. M. 1996, ApJ, 458, 435
Davis, M., Efstathiou, G., Frenk, C. S., & White, S. D. M. 1985, ApJ, 292, 371
De Boni, C., Serra, A. L., Diaferio, A., Giocoli, C., & Baldi, M. 2016, ApJ, 818, 188
Diaferio, A. 1999, MNRAS, 309, 610
Diaferio, A., & Geller, M. J. 1997, ApJ, 481, 633
Diaferio, A., Kauffmann, G., Balogh, M. L., et al. 2001, MNRAS, 323, 999
Diemand, J., Moore, B., & Stadel, J. 2004, MNRAS, 352, 535
Dressler, A., Oemler, A., Jr., Poggianti, B. M., et al. 2013, ApJ, 770, 62
Dressler, A., & Shectman, S. A. 1988, AJ, 95, 985
Everitt, B. S., Landau, S., Leese, M., & Stahl, D. 2011, Cluster Analysis (5th ed.; New York: Wiley)
Geller, M. J., & Beers, T. C. 1982, PASP, 94, 421
Gill, S. P. D., Knebe, A., & Gibson, B. K. 2005, MNRAS, 356, 1327
Gill, S. P. D., Knebe, A., Gibson, B. K., & Dopita, M. A. 2004, MNRAS, 351, 410
Harvey, D., Massey, R., Kitching, T., Taylor, A., & Tittley, E. 2015, Sci, 347, 1462
Hennig, C., Meila, M., Murtagh, F., & Rocci, R. 2015, Handbook of Cluster Analysis (Boca Raton, FL: CRC Press)
Huchra, J. P., & Geller, M. J. 1982, ApJ, 257, 423
Hwang, H. S., Geller, M. J., Diaferio, A., & Rines, K. J. 2012, ApJ, 752, 64
Knebe, A., & Müller, V. 2000, A&A, 354, 761
Knebe, A., Pearce, F. R., Lux, H., et al. 2013, MNRAS, 435, 1618
Komatsu, E., Smith, K. M., Dunkley, J., et al. 2011, ApJS, 192, 18
Kummer, J., Kahlhoefer, F., & Schmidt-Hoberg, K. 2018, MNRAS, 474, 388
Lemze, D., Postman, M., Genel, S., et al. 2013, ApJ, 776, 91
Materne, J. 1978, A&A, 63, 401
Mohammed, I., Saha, P., Williams, L. L. R., Liesenborgs, J., & Sebesta, K. 2016, MNRAS, 459, 1698
Mohr, J. J., Geller, M. J., & Wegner, G. 1996, AJ, 112, 1816

Natarajan, P., De Lucia, G., & Springel, V. 2007, MNRAS, 376, 180
Oguri, M., Miyazaki, S., Hikage, C., et al. 2018, PASJ, 70, 26
Okabe, N., Futamase, T., Kajisawa, M., & Kuroshima, R. 2014, ApJ, 784, 90
Pisani, A. 1996, MNRAS, 278, 697
Pranger, F., Böhm, A., Ferrari, C., et al. 2013, A&A, 557, A62
Ramella, M., Biviano, A., Pisani, A., et al. 2007, A&A, 470, 39
Rines, K., & Diaferio, A. 2006, AJ, 132, 1275
Rines, K., Diaferio, A., & Natarajan, P. 2008, ApJL, 679, L1
Rines, K., Geller, M. J., Diaferio, A., & Kurtz, M. J. 2013, ApJ, 767, 15
Rines, K., Mahdavi, A., Geller, M. J., et al. 2001, ApJ, 555, 558
Rines, K. J., Geller, M. J., Diaferio, A., & Hwang, H. S. 2016, ApJ, 819, 63
Robertson, A., Massey, R., & Eke, V. 2017, MNRAS, 465, 569
Serna, A., & Gerbal, D. 1996, A&A, 309, 65
Serra, A. L., Diaferio, A., Murante, G., & Borgani, S. 2011, MNRAS, 412, 800
Solanes, J. M., Salvador-Solé, E., & González-Casado, G. 1999, A&A, 343, 733
Springel, V., White, S. D. M., Tormen, G., & Kauffmann, G. 2001, MNRAS, 328, 726
Utsumi, Y., Geller, M. J., Dell'Antonio, I. P., et al. 2016, ApJ, 833, 156
Yu, H., Diaferio, A., Agulli, I., Aguerri, J. A. L., & Tozzi, P. 2016, ApJ, 831, 156
Yu, H., Serra, A. L., Diaferio, A., & Baldi, M. 2015, ApJ, 810, 37