Unravelling cosmic velocity flows: a Helmholtz-Hodge decomposition algorithm for cosmological simulations

David Vallés-Pérez\textsuperscript{a,}\textsuperscript{*}, Susana Planelles\textsuperscript{a}, Vicent Quilis\textsuperscript{a,b}

\textsuperscript{a}Departament d’Astronomia i Astrofísica, Universitat de València, E-46100 Burjassot (València), Spain
\textsuperscript{b}Observatori Astronòmic, Universitat de València, E-46980 Paterna (València), Spain

Abstract

In the context of intra-cluster medium turbulence, it is essential to be able to split the turbulent velocity field in a compressive and a solenoidal component. We describe and implement a new method for this aim, i.e., performing a Helmholtz-Hodge decomposition, in multi-grid, multi-resolution descriptions, focusing on (but not being restricted to) the outputs of AMR cosmological simulations. The method is based on solving elliptic equations for a scalar and a vector potential, from which the compressive and the solenoidal velocity fields, respectively, are derived through differentiation. These equations are addressed using a combination of Fourier (for the base grid) and iterative (for the refinement grids) methods. We present several idealised tests for our implementation, reporting typical median errors in the order of \(1\% - 1\%\), and with 95-percentile errors below a few percents. Additionally, we also apply the code to the outcomes of a cosmological simulation, achieving similar accuracy at all resolutions, even in the case of highly non-linear velocity fields. We finally take a closer look to the decomposition of the velocity field around a massive galaxy cluster.

Keywords: turbulence, large-scale structure of Universe, galaxies: clusters: intracluster medium, galaxies: clusters: general, methods: numerical, Adaptive Mesh Refinement

1. Introduction

Cosmological structures and, in particular, galaxy clusters, which constitute the most massive structures which have had time to collapse under their own gravity, are dynamically interesting objects from several perspectives. The non-linearity of the evolution of their baryonic component (i.e., ordinary matter, most of which is in the form of a hot, tenuous plasma known as the intra-cluster medium, ICM) couples the different scales in the evolution of cosmic inhomogeneities, producing a plethora of complex hydrodynamical phenomena, such as shock waves and turbulence. Turbulence is an intrinsically multi-scale phenomenon, since bulk motions trigger (magneto-)hydrodynamical instabilities (e.g., \cite{1}) which cascade down to smaller scales until they get dissipated by viscous effects.

Turbulent motions can be \cite{2} and have been recently \cite{3,4} measured on a number of nearby galaxy clusters from X-ray surface brightness fluctuations, which have in turn been connected to signatures of particle acceleration and diffuse radio emission in the ICM \cite{5,6}. Future X-ray facilities, like ATHENA\textsuperscript{1}, will offer unprecedented insight into the dynamics of turbulent motions in the ICM \cite{7}. However, the precise theoretical and numerical description of turbulence in these vast structures is still matter of ongoing research (e.g., \cite{8,9,10,11}, just to highlight a few recent numerical studies).

Splitting the (turbulent) velocity field in its compressive and solenoidal components, i.e., performing a Helmholtz-Hodge decomposition (HHD), is a crucial step towards exploring the role of turbulence in the ICM, since these components play fundamental and distinct roles in the evolution of cosmic structures. Thus, while the solenoidal component is likely the major responsible for the amplification of cosmic magnetic fields \cite{12,13}, compressive turbulence has an important role in generating weak shocks which have consequential effects on the magnetic and thermal evolution of the cluster (e.g., \cite{14,15}). These two components also differ in their spatial distribution, the former tending to be more volume-filling \cite{16,17,18}, and even in their spectrum, steeper for the compressive component \cite{8,19,20,21}. The distinction between the solenoidal and compressive component of the velocity field is also of utmost importance to model the acceleration of cosmic ray particles in the ICM, which will become a vibrant field of observational research with the advent of the new generation of radio telescopes (e.g., SKA \cite{22}). In this line, a lot of effort has been recently put in modelling the acceleration by compressive and solenoidal modes in magnetohydrodynamics (MHD; \cite{23,24,25,26}).

Cosmological hydrodynamical simulations, as many

\textsuperscript{*}Corresponding author

Email addresses: david.valles-perez@uv.es (David Vallés-Pérez), susana.planelles@uv.es (Susana Planelles), vicent.quilis@uv.es (Vicent Quilis)

\textsuperscript{1}www.the-athena-x-ray-observatory.eu
other applications in computer fluid dynamics, need a huge dynamical range to be resolved in order to, for example, form realistic galaxies in a cosmological environment [26,27] or adequately describe turbulence in the ICM (see, e.g., [28] for some graphical examples of the effects of resolution on the ability to capture instabilities; see also [29,30], who present detailed studies of stratified, ICM-like turbulence in numerical grids of varying resolution). While Lagrangian codes are inherently adaptive, Eulerian codes based on high-resolution shock-capturing (HRSC) techniques are especially capable of handling shocks and other types of discontinuities, which are pervasive in the formation of cosmological structures ([31], for a review). That is why, among several other options (see, for example, [32] for a broad review), Adaptive Mesh Refinement (AMR) codes are especially suited for cosmological structure formation.

Previous studies of ICM turbulence have already implemented HHD algorithms. For example, several works using uniform grids (or fixed refinement strategies which, ultimately, allow to resolve the object of interest within a constant resolution) perform the decomposition in Fourier space, where it simply reduces to linear algebra projections (e.g., [10,33]). Additionally, [10] also confront this method with solving a Poisson equation (in Fourier space) to find a scalar potential for the compressive velocity component, reporting more accurate results for the first method. However, any of these two procedures, because of their usage of FFT algorithms, can only be applied to regular, uniform grids, which are not the common use in cosmological simulations.

In this paper, we propose, implement and test a new algorithm for performing the Helmholtz-Hodge decomposition in a multi-scale AMR grid, which can therefore be applied to the outcomes of a full-cosmological simulation without the need of performing resimulations of specific objects or constrained simulations. Our method decomposes the velocity fields by solving elliptic partial derivative equations (PDEs), which can be addressed iteratively using a wide variety of well-known algorithms (e.g., [34]). Nevertheless, although our primary focus is the application to cosmological structure formation, we emphasize that the approach presented in this work can be directly applied to any application of block-structured AMR, and readily extended to particle-based simulations through a suitable interpolation scheme.

The rest of the manuscript is organised as follows. In Sec. 2 we describe our method for performing the decomposition and discuss its numerical implementation. In Sec. 3 we present and describe a set of tests to validate the accuracy of the algorithm, while in Sec. 4 we apply it to the complex velocity field of a cosmological simulation. Last, in Sec. 5 we summarise and present our conclusions.

2. Description of the method

The algorithm is based on the Helmholtz-Hodge decomposition (see, e.g., [35,36,37]), which allows to univocally split any velocity field in three terms,

\[ \mathbf{v} = \mathbf{v}_{\text{comp}} + \mathbf{v}_{\text{rot}} + \mathbf{v}_{\text{harm}} \]

where \( \mathbf{v}_{\text{comp}} \) is the compressive (or irrotational, \( \nabla \times \mathbf{v}_{\text{comp}} = 0 \)) velocity field, \( \mathbf{v}_{\text{rot}} \) is the purely rotational (or solenoidal, \( \nabla \cdot \mathbf{v}_{\text{rot}} = 0 \)) velocity field and \( \mathbf{v}_{\text{harm}} \) is the harmonic velocity field (both irrotational and solenoidal, thus satisfying \( \nabla^2 \mathbf{v}_{\text{harm}} = 0 \) and \( \mathbf{v}_{\text{harm}} = \nabla \chi \), with \( \nabla^2 \chi = 0 \)).

The harmonic component can be shown to be identically null in any domain with periodic boundary conditions, and therefore we will no longer consider it. Because of their defining properties, the compressive component can be written as the gradient of a scalar potential, \( \mathbf{v}_{\text{comp}} = -\nabla \phi \), while the rotational component can be derived from a vector potential, \( \mathbf{v}_{\text{rot}} = \nabla \times \mathbf{A} \). From this, it is easy to derive that the scalar and vector potentials can be computed, respectively, as the solutions of the following elliptic PDEs,

\[ \nabla^2 \phi = -\nabla \cdot \mathbf{v} \]
\[ \nabla^2 \mathbf{A} = -\nabla \times \mathbf{v} \]

which are formally equivalent to a set of four decoupled Poisson equations (one for \( \phi \) and one for each cartesian component of \( \mathbf{A} \)) whose sources are the divergence and the components of the curl of the overall velocity field.

Once the potentials have been obtained, the compressive and rotational components of the velocity field can be obtained through differentiation. Note that, in principle, it would only be necessary to find one of the potentials (either \( \phi \) or \( \mathbf{A} \)), since the other velocity component could be then derived by subtracting to the total velocity (Eq. 1 with \( \mathbf{v}_{\text{harm}} = 0 \)). Nevertheless, we have chosen to compute all the potentials in order to keep track of the associated numerical errors.

2.1. Numerical implementation

We have designed a code to perform such decomposition in a multi-resolution, block-structured AMR velocity field. As mentioned before, our code can be easily applied to the outcomes of any AMR simulation (cosmological or not), or even to a particle-based code, from which the continuous velocity field can be defined on an ad-hoc AMR grid structure through a smoothing method (e.g., a particle-mesh, [38]).

In our particular implementation, for the base ( coarsest) level, \( \ell = 0 \), taking advantage of the periodic boundary conditions, Poisson equations are solved in Fourier space.\(^2\)

\(^2\)Note, however, that this is not a requirement of the method. We solve Poisson’s equations in Fourier space as cosmological simulations of sufficiently large volumes typically implement periodic bound-
where they just reduce to a multiplication by the Green’s function. The basic procedure followed to solve Poisson’s equations in the base grid can be described in the following steps:

1. The fast Fourier transform (FFT) of the source (right-hand side terms in Eqs. [2] or [3]) in the base grid is computed, yielding a set of coefficients $F_{i,m,n}$.

2. Poisson’s equation is then solved in Fourier space by multiplying by the Green’s function, $G_{i,m,n}$: $\phi_{i,m,n} = G_{i,m,n} F_{i,m,n}$. The Green’s function is given by:

   $$ G_{i,m,n} = \frac{(\Delta x/2)^2}{\sin^2 \left( \frac{\pi i}{\Delta x} \right) + \sin^2 \left( \frac{\pi m}{\Delta y} \right) + \sin^2 \left( \frac{\pi n}{\Delta z} \right)} $$

   where $\Delta x$ is the cell side length and the domain is discretised in $N_x \times N_y \times N_z$ cells [38].

3. The inverse FFT of $\phi_{i,m,n}$ yields the sought potential at the base level.

In subsequent refinement levels, $\ell > 0$, Poisson equations have to be solved taking into account the boundary conditions imposed by the coarser grids the refined patches are embedded into. In order to do so, we use a successive over-relaxation procedure (SOR; see, for example, [34]) on the discretised Poisson equation.

Each AMR patch is first initialised (both in the boundary and in the interior cells) by linear interpolation from the values of the potential at the best-resolved lower level patch available. Then, the interior cells are iteratively updated in a chessboard pattern as

$$ \phi^{\text{new}}_{i,j,k} = \omega \phi^*_{i,j,k} + (1 - \omega) \phi^{\text{old}}_{i,j,k}, $$

where

$$ \phi^*_{i,j,k} = \frac{1}{6} \left[ \phi_{i+1,j,k} + \phi_{i-1,j,k} + \phi_{i,j+1,k} + \phi_{i,j-1,k} + \phi_{i+1,j+1,k} + \phi_{i+1,j-1,k} + \phi_{i-1,j+1,k} + \phi_{i-1,j-1,k} \right], $$

being $\Delta x$ the cell size at the given refinement level, $f_{i,j,k}$ the source term and $1 < \omega < 2$ the over-relaxation parameter. In order to boost convergence, we set $\omega$ according to the Chebyshev acceleration procedure [34]. Aiming to avoid undesirable boundary effects due to the interpolation of the potential boundary conditions, we extend the patches with 3 fictitious cells in all directions, so that these boundary conditions are enforced slightly far away from the region of interest.

Once $\phi$ and $A$ are known, the velocity components are found by finite differencing the potentials as defined before. We compute the derivatives (both of the velocity and of the potentials) using an eighth-order scheme with a centered stencil of up to 9 points, which provides robust values of the differential operators of the velocity fields and mitigates the impact of spurious noise.

The code is parallelised according to the OpenMP standard directives. Our implementation is freely available through its GitHub repository [4].

3. Tests

Aiming to assess the robustness of our HHD method and its implementation, we have designed a battery of tests focused on quantifying to which extent the code is able to accurately identify and disentangle the compressive and rotational velocities. We describe such tests in Sec. 3.1 and examine their results in Sec. 3.2.

3.1. Description of the tests

For the four tests described below, we have first established a simple AMR grid structure, i.e., a set of patches at different refinement levels that could reasonably mock the ones generated in an actual simulation.

We have considered a cubic domain of unit length, with origin at the center of the box, and we have discretised it with $128^3$ cells. For each octant, we establish a first refinement patch with twice the spatial resolution covering the central 1/8 of the octant volume. For example, in the first octant ($0 < x, y, z < 1/2$), we set up a patch at $1/8 < x, y, z < 3/8$ with $64^3$ cells (and likewise for the remaining 7 octants). Then we add a level $\ell = 2$ patch in the central 1/8 volume of each $\ell = 1$ patch, and continue recursively.

In the first three tests we consider up to $\ell_{\text{max}} = 10$ refinement levels, providing a peak resolution of $7.63 \times 10^{-6}$ (relative to the box size, which is normalised to 1). Figure 1 presents graphically the grid structure employed in these tests. Below, we describe the velocity fields that we have seeded on these grids.

3.1.1. Test 1: constant divergence field

For the first test, we consider a purely compressive velocity field with constant divergence, given in cartesian and spherical coordinates by the analytical expression

$$ \mathbf{v} = \omega_0 (x \hat{u}_x + y \hat{u}_y + z \hat{u}_z) \equiv \omega_0 \mathbf{r} \hat{u}_r. $$

This field has $\nabla \cdot \mathbf{v} = 3\omega_0$ and $\nabla \times \mathbf{v} = \mathbf{0}$, and hence $\mathbf{v} = \mathbf{v}_{\text{comp}}$ and $\mathbf{v}_{\text{rot}} = \mathbf{0}$. It is easy to find an analytic expression

\[\text{https://www.github.com/dvallesp/vortex/}\]
for the scalar potential, $\phi = -\frac{\omega_0}{4\pi} (x^2 + y^2 + z^2) + C$, with $C$ any arbitrary real constant. Likewise, the vector potential ought to be $A = \nabla \chi$, with $\chi$ an arbitrary scalar function or, in particular, $A = 0$. We have set $\omega_0 = 0.01$.

3.1.2. Test 2: constant curl field

Analogous to the previous one, we have also considered the case of a purely uniformly rotating velocity field, which can be analytically given in cartesian and cylindrical coordinates as:

$$\mathbf{v} = \omega_0 (-y \hat{u}_x + x \hat{u}_y) \equiv \omega_0 \rho \hat{u}_\phi$$  \hspace{1cm} (8)

It is straightforward to show that $\nabla \cdot \mathbf{v} = 0$ and $\nabla \times \mathbf{v} = 2\omega_0 \hat{u}_z$. This velocity field is generated by the potentials $\phi = C$, being $C \in \mathbb{R}$ a free constant, and $A = -2\omega_0 \rho \hat{u}_x + \nabla \chi$, with $\chi$ any arbitrary scalar function. As in Test 1, we set $\omega_0 = 0.01$.

These two previous tests (Test 1 and Test 2) are aimed to validate the reconstruction of pure velocity fields (either compressive or rotational, respectively), i.e., to estimate the magnitude of the errors involved in the procedure described in Sec. 2.1 in idealised situations where there is no cross-talk between rotational and compressive components. Last, note that even though these two velocity fields are not periodic in the mock simulation domain, this will only affect a negligible amount of cells in the domain’s boundary.

3.1.3. Test 3: compressive and rotational, periodic field

We have designed a third test, aimed to assess the effects of the cross-talk between the compressive and the rotational components. We have considered the velocity field:

$$\mathbf{v} = \left[ \sin (2\pi x) + \sin (4\pi y) + \sin (6\pi z) \right] \hat{u}_x +$$
$$+ \left[ \sin (6\pi x) + \sin (2\pi y) + \sin (4\pi z) \right] \hat{u}_y +$$
$$+ \left[ \sin (4\pi x) + \sin (6\pi y) + \sin (2\pi z) \right] \hat{u}_z$$  \hspace{1cm} (9)

The compressive part corresponds to the terms of angular frequency $2\pi$, while the higher frequency ones constitute the rotational component. Also in this case, it is easy to find analytical solutions to $\phi$ and $A$:

$$\phi = \frac{1}{2\pi} \left[ \cos (2\pi x) + \cos (2\pi y) + \cos (2\pi z) \right] + C$$  \hspace{1cm} (10)

$$A = \left[ -\frac{1}{4\pi} \cos (4\pi z) + \frac{1}{6\pi} \cos (6\pi y) \right] \hat{u}_x +$$
$$+ \left[ -\frac{1}{4\pi} \cos (4\pi x) + \frac{1}{6\pi} \cos (6\pi z) \right] \hat{u}_y +$$
$$+ \left[ -\frac{1}{4\pi} \cos (4\pi y) + \frac{1}{6\pi} \cos (6\pi x) \right] \hat{u}_z + \nabla \chi.$$

3.1.4. Test 4: ICM-like mock velocity field

While the previous tests have checked the ability of the code to reconstruct idealised solenoidal, compressive and mixed velocity fields, we have implemented a last test aimed to assess the ability of the code to capture and reconstruct variations on a broad range of spatial frequencies. The test is, in part, inspired by the one presented by [10, App. A.1.2], but with some differences aimed to mix both velocity fields (instead of generating a purely solenoidal or compressive field), while still having an analytic solution of the HHD to compare with the numerical results.

We have generated our mock, ICM-like velocity field according to the procedure described below:

1. We consider a uniform grid of $(N_x \times 2^{\text{max}})^3$ cells. In that grid, we compute the velocity field $\mathbf{v} = \mathbf{v}_{\text{comp}} + \mathbf{v}_{\text{rot}}$, with:

$$\mathbf{v}_{\text{comp}} = \sum_{i=x,y,z} \sum_{n=\text{min}}^{\text{max}} A_{\text{comp}}^n \sin \left( \frac{2\pi n}{L} x_i + \psi_{\text{comp},i}^n \right) \hat{u}_i$$  \hspace{1cm} (12)

$$\mathbf{v}_{\text{rot}} = \sum_{i=x,y,z} \sum_{j \neq i} \sum_{n=\text{min}}^{\text{max}} A_{\text{rot}}^n \sin \left( \frac{2\pi n}{L} x_j + \psi_{\text{rot},ij}^n \right) \hat{u}_i$$  \hspace{1cm} (13)

being $L$ the box side length ($L = 1$ in our case), $A_{\text{comp}}^n$ ($A_{\text{rot}}^n$) the amplitude of the mode of frequency $n$ of the compressive (solenoidal) component, and $\psi_{\text{comp},i}^n$ ($\psi_{\text{rot},ij}^n$) the initial phases. For the initial phases, we

![Figure 1: AMR grid structure for the tests in Sec. 3. The figure represents a slice at $z = 0.25$, with the colors encoding the highest refinement level at each point in the $x$-$y$ plane. The white, dashed lines indicate the cartesian $x$ and $y$ axes.](image-url)
have generated 9 sets of $N_{\text{max}} - N_{\text{min}} + 1$ random numbers, uniformly sampled from the interval $0 < \psi < 2\pi$. The amplitudes are generated so that the compressive (solenoidal) component follows a Burgers [39] (Kolmogorov [40]) spectrum, $E(k) \propto k^{-2}$ ($E(k) \propto k^{-5/3}$). We set $N_{\text{min}} = 2$ and $N_{\text{max}} = 1024$, so that our mock velocity field presents solenoidal and compressive fluctuations over scales differing almost 3 orders of magnitude. In order for both components to be relevant, we fix the amplitudes so that $A_{n}^{\text{rot}} = A_{n}^{\text{comp}}$ for $n = 64$. We note that, while the velocity field generated according to this procedures is not the most general one (e.g., one could add oblique plane waves), it is challenging enough in order to show the capability of our code to handle a broad range of spatial frequencies in close-to-realistic conditions.

2. Then, we compute this total velocity field onto the AMR grid structure defined at the beginning of Sec. 3.1. In this case, we maintain $N_x = 128$ as the resolution of the base grid, and limit the number of refinement patches to $\ell_{\text{max}} = 4$, which is still enough to show the multi-scale capabilities of our code. When computing the value of the velocities on the base level or on AMR levels $\ell < \ell_{\text{max}}$, we average over the uniform grid cells enclosed in the coarser volume element.

This process consistently generates a mixed, solenoidal and compressive, velocity field presenting similar scaling features as the ICM over almost 3 decades in spatial frequency. Therefore, it can robustly show the capability of the code to handle multi-scale (and multi-frequency) velocity signals.

3.2. Results

For each test, we have validated the performance of the code by computing a series of error statistics, which we present below. Let $\mathbf{v}$ be the input velocity field, for which our algorithm returns its compressive and rotational components, $\mathbf{v}_{\text{comp}}$ and $\mathbf{v}_{\text{rot}}$. Thus, the algorithm recovers a total velocity field $\tilde{\mathbf{v}} = \mathbf{v}_{\text{comp}} + \mathbf{v}_{\text{rot}}$ which might differ from the original, $\mathbf{v}$, due to the numerical error in the processes involved, namely finite-differencing the velocity field, integrating the elliptical equations and finite-differencing the potentials.

In Tests 1 and 2, where only one velocity component (compressive and rotational, respectively) was present, we have quantified the error in reconstructing these velocity fields by computing the cell-wise relative error as

$$\varepsilon_c (v_{\text{cell}}) = \frac{\sum_{i=1}^{3} \left( \frac{v_{i}^{\text{cell}}}{v_{\text{cell}}} \right)^2}{\sum_{i=1}^{3} \left( \frac{v_{i}^{\text{cell}}}{v_{\text{cell}}} \right)^2}$$

where the subindex $i$ runs over the three cartesian components. Note that we add a small constant, $\epsilon$, in the denominator of the relative error in $v_{i}^{\text{cell}}$ to prevent the overestimation of the error due to the cells with velocities close to 0. We set $\epsilon = 10^{-2} \max (|v_{i}|)$. For each refinement level, $\ell$, we have computed the median error over all the cells, which we use as an estimate of the velocity reconstruction error. We also consider the error percentiles 5, 25, 75 and 95 in order to give a confidence interval (CI) for this error.

The velocity field in Test 1 (Test 2) has null rotational (compressive) component. We have checked this by computing the median value and the 5, 25, 75 and 95 percentiles of $|v_{\text{rot}}|/|v|$ ($|v_{\text{comp}}|/|v|$).

The results for Test 1 and Test 2 are summarised in Figures 2 and 3, respectively. The upper panels show $\varepsilon_c (v)$ as defined above. Both in the constant divergence and the constant rotational test, the median errors are small (typically lower than 1%), and even the 95-percentile errors do not exceed 1% at any refinement level. The base level, for which FFT is used, has much more precise results, with median relative errors below $10^{-6}$. For the AMR levels, we only find a very slight trend to increase the error in more refined levels.

The lower panels in Figs. 2 and 3 show $|v_{\text{rot}}|/|v|$, for
Test 1, and $|v_{\text{comp}}|/|v|$, for Test 2. For these highly-idealised scenarios, there is virtually no cross-talk between the different components: the rotational velocity in Test 1 accounts for less than $10^{-8}$ in more than 95% of the cells at any level. Likewise, more than 95% of the cells in Test 2 have compressive velocities less than $10^{-7}$ relative to the total velocity magnitude.

In Test 3, as both the compressive and the rotational velocity components are present, we have quantified the relative error in disentangling and reconstructing each of these by applying Eq. 14 separately to each component. We present its results in Figure 3.

For the compressive velocity, the typical relative errors are below $3 \times 10^{-3}$, while the rotational velocity presents median relative errors in the range $10^{-4} - 10^{-3}$. For the AMR levels, $\ell \geq 1$, the magnitude and behaviour of the errors resembles what has been seen before for Tests 1 and 2. Remarkably, in this example where both components are present, the base level does no longer exhibit a much more precise result, but it shows errors in the same order as the ones in the AMR levels.

Last, for the exceedingly complex Test 4, since the actual decomposition is still known, we follow the same procedure as in Test 3. The detailed results, presented in the same way as for the previous test, can be found in Fig. 4. Compared to Tests 1 – 3, in this case we find errors up to an order of magnitude higher, which is not surprising since in this case we have included a truly multi-scale velocity field, with signal spanning over almost 3 decades in spatial frequency. In any case, our code performs the
HHD decomposition with median errors around 1%, and not exceeding 7% even at 95-percentile level, despite the fact that we have introduced oscillations close to the grid nominal frequency.

The results for this last test, together with the ones we present below in Sec. 4.2, show the ability of our algorithm to perform the HHD in challenging, very non-linear velocity fields.

4. Application to a cosmological simulation

Last, this section focuses on the results of our HHD code when applied to the outcomes of a cosmological simulation, which is described in the paragraphs below. As opposed to the previous highly idealised tests 1 – 3, the velocity field in a full-cosmological simulation exhibits a plethora of complex features due to the non-linear nature of the equations governing its evolution (see, e.g., [26, 32, 41] for classical and recent reviews).

4.1. Simulation details

The simulation analysed in this paper has been carried out with the cosmological code MASCLET [42], and has been already employed in a series of previous works [43, 44, 45]. Here we shall introduce the main details of the simulation, while some topics which are not intimately connected to the analyses in this paper can be found in more detail in the aforementioned references.

MASCLET is an Eulerian cosmological code, implementing high-resolution shock-capturing techniques for the description of the gaseous component and an N-Body particle-mesh for dark matter (DM). Both components are built into an AMR scheme to gain spatial and temporal resolution in the regions of interest.

We have simulated a cubic domain of comoving side length 40 Mpc, assuming a flat ΛCDM cosmology with the following values of the cosmological parameters: $h \equiv H_0/(100 \text{ km s}^{-1} \text{ Mpc}^{-1}) = 0.678$, $\Omega_m = 0.31$, $\Omega_b = 0.048$, $\Omega_{\Lambda} = 0.69$, $n_s = 0.96$ and $\sigma_8 = 0.82$, which are consistent with the latest values reported by the Planck mission [46]. The domain is discretised in a base grid of $128^3$ cells, granting a harsh resolution of $\sim 310 \text{ kpc}$ at the coarsest level. Regions with large gaseous and/or DM densities can get recursively refined following the AMR scheme. We allow $n_f = 9$ refinement levels, each one halving the cell side length with respect to the previous level, providing a peak resolution of $\sim 610 \text{ pc}$. The peak DM mass resolution is $\sim 2 \times 10^6 M_\odot$, equivalent to filling the domain with $1024^3$ of such particles.

The simulation started at redshift $z = 100$, with the initial conditions set up by a CDM transfer function [47] and generated by a constrained realisation of the gaussian random field aimed to produce a massive cluster in the center of the computational domain by $z \sim 0$ [48]. By redshift $z \sim 0$, several massive clusters and groups have been formed. Besides gravity, the simulation accounts for a broad variety of feedback mechanisms, which are explained in greater detail in the cited previous works.

4.2. Performance of the code

We have run our HHD algorithm over 80 snapshots of the simulation described above, ranging from $z = 100$ to $z = 0$, and computed the cell-wise error\footnote{In this case, as we do not know beforehand the ‘true’ decomposed velocity fields to compare with the reconstructed ones, we quantify the error by comparing the reconstructed total velocity field, $\tilde{v} = \tilde{v}_{\text{comp}} + \tilde{v}_{\text{rot}}$, to the input one, as defined in Eq. (14). Since, by definition, $\tilde{v}_{\text{comp}}$ ($\tilde{v}_{\text{rot}}$) is the gradient of a scalar field (the rotational of a vector field), it is irrotational (solenoidal). We have, indeed, checked that our high-order derivatives verify this, typically much better than 1%. Therefore, as the decomposition is unique, checking that $\tilde{v} = v$ proves the validity of the method.} and its corresponding percentiles (5, 25, 75 and 95), as done in the tests in Sec. 3. Figure 6 presents the overall error statistics, defined as the median of the error statistics over all the code outputs. In order to keep track of the dispersion of the error statistics in different snapshots, we have also plotted the 80 individual error profiles in dotted lines, with the line colours encoding the redshift.
The density map shows that, by $z \sim 0$, the cluster is, indeed, relatively relaxed, sitting in the gravitational potential dominated by the dark matter component, in an approximately spherical shape. Several density discontinuities can be easily discerned, mainly corresponding to (internal) merger shocks and (external) accretion shocks. A filament penetrating quite inner radii, of around $r \sim 1\,\text{Mpc}$, is also noticeable in the density plot. The total velocity field displays great complexity, especially in the inner regions of the cluster, where the variations occur on smaller scales (both because the dynamics are more complex and because, correspondingly, the resolution is higher). The filamentary structures appear to present high velocity magnitudes, mainly pointing radially, and velocity discontinuities, hinting the presence of shocks, are ubiquitous.

Compressive velocities show a nearly spherically symmetric pattern, as the cluster smoothly accretes gas from its surroundings. In the outskirts, baryonic matter is accelerated (and compressed) by the gravitational pull of the cluster, until it gets shocked causing the strong accretion shocks. In comparison, the post-shock medium presents very small compressive velocities, as the shock has effectively halted the accretion flows. Part of the energy corresponding to these compressive component gets thermalised, increasing the internal energy (and temperature) of the ICM. However, another important role of shocks is the generation of vorticity (see, e.g., [13]). Indeed, inside the external accretion shocks, the solenoidal component of the velocity fields gets relevant. Eddies develop on a wide range of scales, especially in the cluster central regions. It is also interesting to note how the infalling filament mentioned before presents high values of the rotational velocity, suggesting that matter is infalling following heliocoidal trajectories. Being the main aim of this work is presenting the algorithm, we may defer a more in-depth analysis of these issues to future work.

Complementarily, in Figure 8 we present the radial profile of radial total (red line), radial compressive (blue) and radial solenoidal (green) velocities, for $20\,\text{kpc} \lesssim r \lesssim 4\,\text{Mpc} \simeq 2R_{\text{vir}}$. In the inner regions ($r \lesssim 0.1\,\text{Mpc}$) both strong radial compressive and solenoidal flows are present, bounded by an internal shock (clearly visible in the compressive velocity map of Fig. 7). For $r \gtrsim 0.1\,\text{Mpc}$, the radial compressive velocity clearly dominates. While solenoidal motions are still present, the fact that their radial component is close to zero suggests that these motions tend to occur along the tangential direction.

5. Conclusions

In this paper, we have proposed a novel method to perform a Helmholtz-Hodge decomposition in AMR velocity fields, or virtually in any description which can be smoothened over an ad-hoc hierarchy of grids. Although our primary focus has been cosmological simulations of struc-
Figure 7: Maps around a massive cluster. Each map is a mass-weighted projection of the same region around the cluster, 10 Mpc on each side and $\sim 150$ kpc thick, limiting the image resolution to $\sim 10$ kpc. Top left: Gas density (in units of the background density of the Universe). Top right: Total velocity magnitude. Bottom left: Compressive velocity magnitude. Bottom right: Solenoidal velocity magnitude. The arrows in the velocity maps represent the projection of the corresponding velocity fields in the slice plane.
ture formation, the method is general and could be easily extended to any type of hydrodynamical simulation.

Previous works in the field of numerical cosmology typically use uniform grids and work straightforwardly in Fourier space. However, this procedure requires to perform constrained simulations (or resimulations) of specific objects of interest (e.g., a galaxy cluster) in order to be able to describe it with a uniform computational grid at a reasonable computational cost. Our algorithm, instead, can be applied to full-cosmological simulations, without the need of performing resimulations and keeping the full description at the maximum resolution at each position. The performance of the code has been validated in a series of idealised tests, for which the analytical decomposition is known. Our algorithm has shown to succeed in a domain of several tens of Mpc along each direction.

For exceedingly complex velocity fields, like the ones generated by actual cosmological simulations at low redshifts, where turbulence is fully developed (e.g., [10]) and velocity fluctuates on many different scales, our tests show that the decomposition can be brought about with median errors below 1%, even resolving scales smaller than the kpc in a domain of several tens of Mpc along each direction. This procedure, whose implementation has been made publicly available (see Sec. 2.1), will allow us to further explore the dynamics of the turbulent velocity field in the ICM of large samples of clusters in future works.

Acknowledgements

We gratefully acknowledge the anonymous referees for their valuable feedback, which has helped us to improve the quality of this manuscript. This work has been supported by the Spanish Ministerio de Ciencia e Innovación (MICINN, grants AYA2016-77237-C3-3-P and PID2019-107427GB-C33) and by the Generalitat Valenciana (grant PROMETEO/2019/071). Simulations have been carried out using the supercomputer Lluís Vives at the Servei d’Informàtica de la Universitat de València. This research has made use of the following open-source packages: NUMPY [29], SCIPY [30], MATPLOTLIB [31] and YT [52].

References

[1] J. A. ZuHone, A Parameter Space Exploration of Galaxy Cluster Mergers. I. Gas Mixing and the Generation of Cluster Entropy, ApJ 728 (1) (2011) 54. arXiv:1004.3820 doi:10.1088/0004-637X/728/1/54
[2] M. Gaspari, E. Churazov, Constraining turbulence and conduction in the hot ICM through density perturbations, A&A 559 (2013) A78. arXiv:1307.4397 doi:10.1051/0004-6361/201322995
[3] I. Zhuravleva, E. Churazov, A. A. Schekochihin, S. W. Allen, P. Arévalo, A. C. Fabian, W. R. Forman, J. S. Sanders, A. Simionescu, R. Sunyaev, A. Vikhlinin, N. Werner, Turbulent heating in galaxy clusters brightest in X-rays. Nature 515 (7525) (2014) 85–87. arXiv:1410.6485 doi:10.1038/nature13830
[4] F. Hofmann, J. S. Sanders, K. Nath, N. Clerc, M. Gaspari, Thermodynamic perturbations in the X-ray halo of 33 clusters of galaxies observed with Chandra ACIS, A&A 585 (2016) A130. arXiv:1510.08445 doi:10.1051/0004-6361/201526925
[5] D. Eckert, M. Gaspari, F. Vazza, F. Gastaldello, A. Tramacere, S. Zinner, S. Ettori, S. Paltani, On the Connection between Turbulent Motions and Particle Acceleration in Galaxy Clusters, ApJL 843 (2) (2017) L29. arXiv:1705.02341 doi:10.3847/2041-8213/aa7c1a
[6] A. Bonafede, M. Brüggen, D. Rafferty, I. Zhuravleva, C. J. Riceley, R. J. van Weeren, J. S. Farnes, F. Vazza, F. Savini, A. Willer, A. Bottom, G. Brunetti, R. Cassano, C. Ferrari, F. de Gasperin, E. Orrú, R. F. Pizzo, H. J. A. Röttgering, T. W. Shimwell, LOFAR discovery of radio emission in MACS J0717.5+3745, MNRAS 478 (3) (2018) 2927–2938. arXiv:1805.00473 doi:10.1093/mnras/sty1121
[7] M. Roncarelli, M. Gaspari, S. Ettori, V. Biffi, F. Brighenti, E. Bulbul, N. Clerc, E. Cuccetti, E. Pointecouteau, E. Rasia, Measuring turbulence and gas motions in galaxy clusters via synthetic Athena X-IFU observations, A&A 618 (2018) A39. arXiv:1805.02577 doi:10.1051/0004-6361/201833371
[8] F. Miniati, The Matryoshka Run. II. Time-dependent Turbulence Statistics, Stochastic Particle Acceleration, and Microphysics Impact in a Massive Galaxy Cluster, ApJ 800 (1) (2015) 60. arXiv:1409.3576 doi:10.1088/0004-637X/800/1/60
[9] W. Schmidt, J. F. Engels, J. C. Niemeyer, A. S. Almgren, Hot and turbulent gas in clusters, MNRAS 459 (1) (2016) 701–719. arXiv:1603.04711 doi:10.1093/mnras/stw632
[10] F. Vazza, T. W. Jones, M. Brüggen, G. Brunetti, C. Gheller, D. Porter, D. Ryu, Turbulence and vorticity in Galaxy clusters generated by structure formation, MNRAS 464 (1) (2017) 210–230. arXiv:1609.03858 doi:10.1093/mnras/stx2351
[11] L. Iapichino, C. Federrath, R. S. Klessen, Adaptive mesh refinement simulations of a galaxy cluster merger - I. Resolving and modelling the turbulent flow in the cluster outskirts, MNRAS 469 (3) (2017) 3641–3655. arXiv:1704.02922 doi:10.1093/mnras/stx882

Figure 8: Radial profiles of radial velocity for the total (red), compressive (blue) and solenoidal (green) velocity components.
[42] V. Quilis, A new multidimensional adaptive mesh refinement hydro + gravity cosmological code, MNRAS 352 (2004) 1426–1438. arXiv:astro-ph/0405389 doi:10.1111/j.1365-2966.2004.08040.x

[43] V. Quilis, S. Planelles, E. Ricciardelli, Is ram-pressure stripping an efficient mechanism to remove gas in galaxies?, MNRAS 469 (1) (2017) 80–94. arXiv:1703.09446 doi:10.1093/mnras/stx770

[44] S. Planelles, P. Mimica, V. Quilis, C. Cuesta-Martínez, Multiwavelength mock observations of the WHIM in a simulated galaxy cluster, MNRAS 476 (4) (2018) 4629–4648. arXiv:1802.09438 doi:10.1093/mnras/sty527

[45] D. Valls-Pérez, S. Planelles, V. Quilis, On the accretion history of galaxy clusters: temporal and spatial distribution, MNRAS 499 (2) (2020) 2303–2318. arXiv:2009.13882 doi:10.1093/mnras/staa3035

[46] Planck Collaboration, Planck 2018 results. VI. Cosmological parameters, A&A 641 (2020) A6. arXiv:1807.06209 doi:10.1051/0004-6361/201833910

[47] D. J. Eisenstein, W. Hu, Baryonic Features in the Matter Transfer Function, ApJ 496 (2) (1998) 605–614. arXiv:astro-ph/9709112 doi:10.1086/305424

[48] Y. Hoffman, E. Ribak, Constrained realizations of Gaussian fields - A simple algorithm, ApJ 380 (1991) L5–L8. doi:10.1086/186160

[49] T. E. Oliphant, A guide to NumPy, Vol. 1, Trelgol Publishing USA, 2006.

[50] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. Jarrod Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. Carey, Í. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, SciPy 1. 0 Contributors, SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python, Nature Methods 17 (2020) 261–272. doi:https://doi.org/10.1038/s41592-019-0686-2

[51] J. D. Hunter, Matplotlib: A 2d graphics environment, Computing in Science & Engineering 9 (2007) 90–95. doi:10.1109/MCSE.2007.55

[52] M. J. Turk, B. D. Smith, J. S. Oishi, S. Skory, S. W. Skillman, T. Abel, M. L. Norman, yt: A Multi-code Analysis Toolkit for Astrophysical Simulation Data, ApJS 192 (1) (2011) 9. arXiv:1011.3514 doi:10.1088/0067-0049/192/1/9