A Tension within Code Comparisons

Marie Gueguen*

Abstract

It has recently been argued that robustness is not a sufficient criterion for deeming a cosmological simulation reliable.[1] These arguments, however, focused on a form of robustness called "convergence studies", which only constitute what Orzack and Sober (1993) refer to as "internal robustness", i.e., robustness found within a single model, as opposed to that obtained by comparing the predictions of distinct models. The questions thus arises: can a stronger form of robustness be found in cosmology and astrophysics? And how would it fare compared to convergence studies? Code comparisons, which compare the outcomes of simulations based on different codes, seem at first sight a natural candidate for such a role. In this paper, I will argue however that a tension within code comparisons in astrophysics prevents them to constitute an instance of robustness analysis, for the effort to make targets comparable undermines the preservation of the diversity needed for robustness analysis.

1 Introduction

Computer simulations constitute an indispensable tool of contemporary cosmology. They are necessary to extract predictions from cosmological models regarding structure formation, to design the observational surveys that will collect the data thanks to which models will be assessed (Smeenk and Gallagher 2020), but also to supplement sparse or non-existing observations (Jacquart 2020). Such ubiquity, at every stage of the scientific inquiry, should be met with a rigorous methodology for evaluating when the outcomes of such simulations faithfully track the predictions of the physical model upon which they are based. Yet, I will argue that such methods are still wanting in cosmology, which makes the question of how to assess the strengths and shortcomings of rival models in astrophysics a very tricky one.

N-body simulations have first been used in the 1970's in order to study whether grav-

*Rotman Institute of Philosophy, University of Western Ontario.Email: mgueguen@uwo.ca

[1] See (van den Bosch et al. (2017) or Gueguen (2020))

ity alone could be responsible for the formation of clusters of galaxies. The simulation of the gravitational collapse of a cloud of 300 particles detailed in Peebles 1970 was considered the first realistic simulation of cosmic structure formation. Since this initial attempt, N-body simulations have been used to represent the temporal evolution of cosmic structure at large-scale–the so-called 'cosmological simulations'[2]–, but also to track the evolution of structure at smaller scales, such as individual dark matter haloes hosting galaxies–or 'zoom-in' simulations. In both cases, many simplifications or idealizations are involved in such simulations that can create numerical artefacts. In particular, simulations have a limited mass resolution, due to a limited computational power: dark matter is substituted by fewer, but more massive particles, so as to average the density of real systems. This reduced number of particles can generate discreteness-driven effects that would not exist for real dark matter systems, which behave like continuous media. Similarly, dark matter particles in a simulation do not correspond to what is meant by 'particles' in particle physics, but to parcels of mass standing for up to millions of solar masses! When such heavy particles get close to each other, unphysical accelerations are generated that have no astrophysical counterparts, and a numerical trick called force softening must be added to remove them. Given the possible artificial consequences of these assumptions, among many others, evaluating the trustworthiness of these simulations is a difficult task to achieve. If the outcomes of galaxy formation simulations based on a model like the Cold Dark Matter model (henceforth CDM) do not fit observations, there are no straightforward answer to assess the extent to which this mismatch is problematic. Nor is there any guidance for where to go next.

Consider for instance the predictions made by simulations about the internal structure of dark matter haloes surrounding and hosting galaxies. CDM simulations predicts that halos have a central cusp. Yet, observations favour a more uniform density profile. Likewise, simulations predict several thousand satellite galaxies in a Milky-Way-size halo, but only 59 have been detected so far. How should we evaluate these discrepancies? Dark matter-only simulations are known to be unrealistic, for they discard the baryonic components of the phenomenology of galaxies. Nevertheless, many physical processes involving baryonic physics, called 'feedback', regulate the growth and shape of galaxies within dark matter haloes. Massive stars exploding as supernovae, black holes growth, jets and winds, for instance, are key processes for galaxy formation (Schaye et al., 2010), but are not implemented in dark-matter-only simulations because they happen at a scale way smaller than what a simulation can resolve[3], making it really hard to track both scales at the same time. Thus, there is a chance that this mismatch between simulation and observation could be dissolved by taking into account this missing physics. After

[2]See for instance the Millenium, the Bolshoi or Illustris simulations.
[3]Hence the name of 'subgrid' physics for such processes.

all, even if it is used as a tracer for the dark matter component, what astrophysicists *do* observe is the baryonic component, not the dark matter one. How to add baryonic feedbacks, however, is another challenging task, for only a few parameters regulating this feedback can actually be constrained on the basis of known physics. As a result, if the mismatch persists, the culprit could be the entire CDM model, but it could also be that simulations suffer from numerical artifacts resulting from the modelling of dark matter, or from artifacts stemming from the specific parametrization of the baryonic physics, or of their interplay. How can astrophysicists tell these scenarii apart?

Section 2 offers a primer on N-body simulations in astrophysics for philosophers. Section 3 discusses the role of convergence studies and code comparisons in this area, and how 'reliability' is defined in this context. In section 4, I present two recent code comparison projects, AQUILA and AGORA, and argue that none of them constitute instances of robustness analysis, for none of them succeed in satisfying the comparability and the diversity requirements that I propose as necessary conditions for robustness analysis at the same time. Given that convergence studies, as a case of robustness, fails in eliminating all artifacts and that code comparison do not qualify as instance of robustness analysis, I contend that these methodologies do not permit to assess whether a simulation is artifact-free, neither individually nor in combination. Although my paper focuses on the case of astrophysics, the lesson drawn from code comparisons can possibly be exported to many areas where simulations are crucial in extracting theoretical predictions from models whose complexity forbids the appeal to analytic calculations and blurs the line between different modules of the simulation, particularly between the model and the numerical scheme. I briefly discuss this point in the concluding remarks.

2 The methodology of N-body simulations in cosmology

This section provides a brief sketch of the different ingredients composing a simulation and the different problems that must be solved, once one has defined the kind of simulations they are interested in. This sketch is not meant to be an exhaustive introduction to N-body simulations in astrophysics, but to introduce the terminology and challenges proper to simulations in this field. This primer for philosophers is heavily indebted to the excellent introductions to the topic offered by Klypin (2017) and Bodenheimer et al. (2006).

As Parker (2011) defines it, a computer simulation is a "computer-implemented set of instructions for repeatedly solving a set of equations in order to produce a representation of the

temporal evolution of selected properties of a target system" (581). Large-volume simulations track how very small density fluctuations in the nearly homogeneous early universe evolved with time through gravitational collapse to form the large-scale structures we now observe. Zoom-in simulations re-simulate the haloes thus formed at higher resolution, to gain a better understanding of galaxy formation. Dark matter plays the essential role in the early stages of structure formation. Whereas the collapsing of ordinary, baryonic matter is opposed by the outward radiation pressure of photons, dark matter only interacts gravitationnally and is therefore not opposed by such a electromagnetic force. As a result, dark matter starts collapsing in haloes earlier than ordinary matter, thus providing the scaffolding where stars can merge with gas and form galaxies, cluster of galaxies, and all the large-scale structures of the universe. This is why the first cosmological simulations were only focusing on the dark matter and ignoring the baryonic physics, tracking the evolution of the distribution of particles using a conjunction of the collisionless Boltzmann equations for dark matter particles and the Poisson equation for the gravitational potential.

How to distribute particles within the simulated volume depends on the problem to solve. If the simulation is a cosmological one, then the distribution of particles will be nearly homogeneous and particles will have the same mass. If the intent is to simulate a smaller region with higher resolution, then many small particles will be distributed in the region of interest, with a few large ones in the rest of the volume. The second problem that arise is to calculate gravitational forces between dark matter particles. The way they are calculated differ from one code to another. In Peebles (1970), these forces were calculated using direct summation, i.e., by summing up all contributions from all particles–hence the name of 'Particle-Particle codes'. This technique is almost abandoned nowadays, given its computational cost:the number of operations needed to calculate the forces scales as N^2. One can, instead of summing the contributions of all particles, appeal to a Particle-Mesh code using a three-dimensional mesh covering the cubic domain of the simulation. The idea is simple: calculate the density field for every node of the mesh, using a technique called Cloud-in-Cell density assignment[4]; solve the Poisson equation for the gravitational potential; advance velocities, coordinates and time and repeat for every time step. This code not only discretizes time but also space, as it covers the domain of the simulation with a mesh. The advantage of Particle-Mesh methods is that the resolution can be increased wherever needed by adapting the mesh size and placing smaller cubic cells in the regions of interest, a technique known as 'Adaptative Mesh Refinement'. Moreover, but the Poisson equation is solved through Fast Fourier Transformations methods, thanks to which the

[4]The rough idea of the Cloud-in-Cell technique is to calculate the distance between the center of the mesh cells and the particle and to assign, based on this distance, a weighted fraction of the total particles mass to the nearest 2, 4 or 8 mesh cell centers. See for instance Klypin (2017).

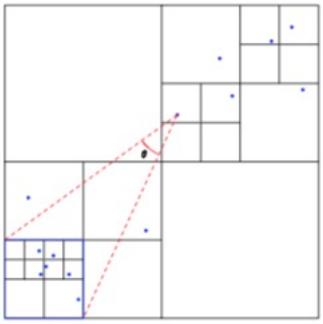

Figure 1: An example of particle grouping algorithms for TREE codes from Klypin 2017. The red dashed lines show the opening angle θ for a particle close to the centre and a given cell (the blue square). The opening angle $\theta = l/d$ at which this cell is seen by the particle pointed at by the red dashed lines must be greater than a given threshold for the blue square to 'open'. If the cell opens, then the contributions of the cell's children (i.e., the force for each particle will be treated individually) will be taken into account rather than the those of the cell as a group. In this particular figure, whether the cell opens or not will depend on the chosen threshold.

computation time is reduced to $NlogN$, with N the number of cells. Another popular way of calculating gravitational forces resides in tree-codes, that do not calculate the contribution of individual particles or the field density but instead group particles hierarchically and replace individual contributions with a single multipole force for the whole group. An algorithm is used to group particles, based on a given threshold per cell–e.g., if the threshold is one particle per cell, and the number of particles in a cell exceeds this number then the cell will split into smaller cubic cells, and so on until each cell contains only one particle. After the tree is constructed, the information about the mass distribution is stored in each cell. Forces can then be found by testing, for each cell of size l, whether the opening angle $\theta = l/d$ at which it is seen by particles in the leaf[5] at distance d is greater than a specified threshold. In this case the force contributions from the cell are ignored and the cell 'opens', so that the opening angles of the cell's children are tested instead (see figure 2). The force contribution from the cell is accepted once the opening angle is smaller than the specified threshold–as an example $GADGET-2$ chose as a threshold the following condition:

$$\theta = \frac{\lambda}{d} = \sqrt{\frac{\alpha g}{[\frac{GM}{d^2}]}} \tag{2.0.1}$$

The idea is to treat individually only the nearest particles, while the distant ones are treated collectively, as a group, to diminish the overall computational cost.

For each of these codes, there are essentially four purely numerical parameters that need to be calibrated:

[5] A leaf is simply a cell that contains more than a specified number f particles.

- The mass resolution N_p, which refers to the number of particles used in the simulation;

- The time-step Δt: the N-body problem in astrophysics consists of solving the Newtonian equations of motions and finding the velocities and coordinates of a number N of massive particles only interacting through Newtonian gravity, given their initial coordinates and velocities. If r_i and m_i are the coordinates and masses of the particles, then the equations of motion that must be integrated are:

$$\ddot{r_i} = -G \sum_{j=1, i \neq j}^{N} \frac{m_j (r_i - r_j)}{|r_i - r_j|^3} \tag{2.0.2}$$

  In the simplified case[6] where one would use Euler integration method to find the new coordinates r_i of the i^{th} particle $r_i(t + \Delta t) = r_i(t) + \Delta t v_i(t)$, then the time-step Δt is simply the time step between the initial time and the later time at which coordinates and velocities must be found.

- The force accuracy θ: the accuracy of the force computation has a distinct meaning depending on the codes. In a tree code, it corresponds to the opening angle θ above which force contributions are ignored. In a Particle-Mesh code, it corresponds to the size of the grid.

- The force softening ϵ: real dark matter particles are substituted by heavy particles, which means that gravitational forces can generate very large accelerations when two particles get very close to each other. Force softening is used to smooth the gravitational potential and suppress these accelerations below a typical distance–the 'softening length'. Like force accuracy, force softening has a different meaning for different codes: mesh codes define softening based on the size of the cell elements, while tree codes use a method referred to as 'Plummer softening" and replace the distance $\Delta r_{ij}^2 = |r_i - r_j|$ in equation 2.1.1 with the expression $(\Delta r_{ij}^2 + \epsilon^2)^{1/2}$. Note that force softening is one of this kludge that prevents a clear distinction between the physical model and the numerical scheme. Indeed, there is surprisingly little physics within a dark matter-only simulation. Once the cosmological parameters have been specified, and the gravitational forces or field taken into account, only numerical parameters are left that must be calibrated. Yet, force softening, which is a purely numerical trick, interferes with the gravitational laws as implemented in the simulations in order to counter discreteness effects, i.e., to prevent divergent pairwise forces and suppress large-angle deflections.

[6] The Euler integrator is not used in realistic simulations because of its low accuracy and replaced by a second-order integration scheme, 'Leapfrog'. See Klypin 2017, section 4 for a detailed discussion of both integrators and their merits.

The optimal parametrization differs for different codes, as differ the meaning of parameters such as force accuracy or force softening.

More realistic 'zoom-in' simulations take into account not only the dark matter component of simulations, but also the baryonic physics–dissipation, heating and cooling of gas, formation of stars and supermassive black holes, magnetic fields, and so on– that potentially affect the distribution of the dark matter. These so-called 'hydrodynamics' simulations take up the incredibly difficult challenge of tracking physical processes playing a role in galaxy formation at scales greater than 100 Megaparsecs *while at the same time* describing the physics of star formation at sub-parsec scale, knowing that both scales considered independently are challenges on their own and that the coupling of these scales makes the task even more complex. Different codes adopt different strategies for solving this task. Different teams tend to prioritize different energy feedback for instance: some focus on supernovae feedback, some others on black holes feedback, others on both. They can also make different choices to model the feedback: supernovae feedback, for instance, can be injected as thermal energy in the interstellar medium (ISM), or as kinetic energy, or by temporarily decoupling the gas from the ISM, etc (Scannapieco et al. (2012)). Finally, most of this subgrid physics is constrained on empirical grounds only, i.e., by the ability of simulations to reproduce the galaxy phenomenology. In sum, different teams can use different codes in that they have different gravity and hydrodynamical solvers, model different feedback processes, or model the same but in different ways, and adopt a different parametrization for each of the subgrid physics components.

3 Robustness analysis in astrophysics: convergence studies and code comparisons

Nowadays, a limited number of methods are used by astrophysicists to assess the reliability of N-body simulations: (1) convergence studies, (2) code comparisons, (3) semi-analytic solutions and (4) observation matching. Starting with the latter, (4) observation matching consists in testing the ability of simulations to reproduce known galaxy morphology. Astrophysicists have had remarkable success in reproducing a catalog of galaxies, with simulations so realistic that it is difficult to differentiate them from real galaxies (see figure 1 below).

This method has turned very useful to interpret and design observational programs, for instance for interpreting the astrometric, spectroscopic and photometric *Gaia* surveys of the Milky Way stellar population–that is, in order to have a substitute to theoretical predictions to

Figure 2: On the left, a mock simulated galaxy from the FIRE simulations. On the right, a Hubble Space Telescope picture of the Pinwheel galaxy. Image credit: FIRE,Phil Hopkins's research group/ESA and NASA.

compare with the results of these surveys. As mentioned above, there is, however, an obvious obstacle to trusting such a method *as a method for assessing the reliability of simulations*: the subgrid physics relied upon to implement the baryonic physics can often only be calibrated on empirical grounds, i. e., calibrated arbitrarily to fit observed galaxy properties. Thus, their success in reproducing the galaxy phenomenology does not tell us much about their trustworthiness[7][8]. In some rare cases, codes can also be tested against analytical solutions. Efstathiou et al. (1985), for instance, tested their P^3M codes for discreteness effects by focusing on plane-wave collapse simulations–an ideal situation for checking whether a collisionless system with one-dimensional perturbations remains one-dimensional as it should, given the simplicity of the simulation and the existence of an exact solution up to shell-crossing (Zel'Dovich (1970),Shandarin and Zeldovich (1989)). The last two methods, convergence studies[9] and code comparisons, are motivated by a common idea, that of robustness analysis. In the former, a property that remains invariant under a variety of assumptions is deemed a 'robust', trustworthy property, in that it does not depend on the specifics of the codes and can thus be relied upon (Levins, 1966; Wimsatt, 1981; Weisberg, 2006). The 'variety of assumptions' considered is not of the same kind in convergence studies and code comparisons. The former test the reliability of a given code by looking for properties resisting to a change in the calibration, i. e., to a change of values of the purely numerical parameters. Code comparisons, on the other hand, look for properties across different codes; in other words, codes are cross-checked in order to search for properties that remain the same and are therefore independent from the assumptions upon which different codes

[7]See also Parker 2011 for a similar point in climate sciences.
[8]See for instance Sanderson et al. (2020)
[9]The term convergence study or numerical study is that used by astrophysicsts. As the reader will see, the definition of convergence studies overlaps exactly with the definition of 'internal robustness' given by Orzack and Sober or 'parameter robustness' in the words of Weisberg and Reisman (2008),p.115. For convenience, I will keep using the terminology of astrophysicists for discussing simulations of structure formation.

are based. The former constitute an internal form of robustness, searching for predictions that remain invariant within a single code despite different values assigned to numerical parameters , whereas the latter is an external one, attempting to track invariant predictions across different codes. Convergence studies and code comparisons, at first sight, seems to resemble what the verification step of the V & V procedure aim to capture. V & V, i.e., verification and validation, is a two-step procedure whose goal consists respectively of ensuring that numerical 'errors', including the necessary discretization of dark matter, do not have any significant impact of the outputs of simulations, and to check whether the simulation output agrees with the empirical data. In other words, verification consists of assessing whether the equations have been solved accurately, while the validation step warrants that the rights equations have been solved –that the model is adequate. This distinction, however, relies on the possibility of distinguishing the physical model from the computational problem. As it has been argued in Winsberg (2009), Winsberg (2010) and Lenhard and Winsberg (2010) however, such a distinction vanishes away once the simulation model has reached a certain degree of complexity: when a simulation outcome does not agree with observational data, it becomes impossible to determine whether the blame reside in the physical assumptions or in the numerical errors, notably because those two become too entangled and the modularity too 'fuzzy' for a sharp distinction to still apply. Astrophysicists themselves tend to use these terms imprecisely, in particular when explaining the aim of their codes comparison and the conclusions that can be drawn from them[10]. For this reason, I will avoid this terminology altogether and focus on the vocabulary that is explicitly at the heart of the task that astrophysicists assign to convergence studies and code comparisons, i.e., that of eliminating all sources of numerical artifacts[11]. Hence, 'reliable' in the context of this paper will be defined as 'free of artifacts significantly impacting the results of simulations', meaning that the outputs of simulations do not faithfully tracks the consequences of the physical model they implement but rather are distorted by numerical errors.

Let us take a closer look to convergence studies. Many codes have been developed since

[10] Ludlow et al. (2020), mentions that convergence studies are considered necessary to "validate the robustness of the a particular numerical result" (p.2). Likewise, the Santa Barbara Cluster code comparison (Frenk et al. (1999)) identifies the comparison to known analytic solution as a form of validation of a simulation.

[11] See for instance Power et al. (2003): "Extreme care is thus needed to separate numerical artefacts from the true predictions of the CDM model. In order to validate or 'rule out' the CDM cosmogony, we must be certain that model predictions on the relevant scales are accurate, robust, and free of systematic numerical uncertainties. Although there have been some recent attempts at unravelling the role of numerical parameters on the structure of simulated dark matter haloes, notably in the work of Moore et al. (1998), Knebe et al. (2000); Klypin et al. (2001) and Ghigna et al. (2000), the conclusions from these works are still preliminary and, in some cases, even contradictory (p.15)". Later, they add:"Understanding the origin of such disparate conclusions and the precise role of numerical parameters is clearly needed before a firm theoretical prediction for the structure of CDM haloes on $\simeq$ kpc scales may emerge.Motivated by this, we have undertaken a large series of numerical simulations designed to clarify the role of numerical parameters on the structure of simulated CDM haloes. In particular, we would like to answer the following question: what regions of a simulated dark matter halo in virial equilibrium can be considered reliably re-solved?" (p. 16) See also Ludlow et al. (2020): " 'convergence criteria' (...) can be used to disentangle aspects of simulations that are reliably modelled from those that may be affected by numerical artifact." (p.20)

the 1970s to model the evolution of a cloud of dark matter particles, based on different ways to calculate gravitational forces. For each of these codes, there are purely numerical parameters that must be calibrated that are not constrained at all, or poorly so, by the physics implemented in the simulations. Hence, the aim of convergence studies is to define the conditions under which the structure of a simulated halo of dark matter does not depend on the value assigned to these numerical parameters and can thus be deemed 'appropriately resolved'. Can convergence studies be trusted as a method for discriminating robust, physical predictions against numerical artifacts? The problem is that such a method assumes that if simulations converge toward a common prediction, despite different values assigned to numerical parameters, then the physics must be responsible for these predictions and not the numerical scheme. In other words, it assumes that these two components can be easily separated and that convergence is a reliable criterion to draw the line between the physical and the numerical. As highlighted by Gueguen (2020) however, convergence cannot be a sufficient criterion for reliability. Tests performed by astrophysicists such as Melott et al. (1997), Efstathiou et al. (1985), van den Bosch and Ogiya (2018) or Baushev (2015) show that convergence fails in ruling out artifacts, especially artifacts due to the discretization of dark matter and the subsequent two-body scattering or collisionality that can result from the discreteness of dark matter in such simulations. That convergence is not by itself a sufficient criterion is actually no longer a controversial criterion, as pointed out in van den Bosch and Ogiya (2018) (p.), Scannapieco et al. (2012)(p.1739), although it is still acknowledged as necessary. Moreover, not only convergence is not sufficient but in some cases numerical artifacts can be fully responsible for the convergence of the results. Baushev et al. (2017), for instance, has shown that simulations suffer from discreteness-driven collisional effects which not only are responsible for the cuspy density profile observed in simulations[12], but also guarantee their convergence. Hence, convergence studies might be at best a good starting point for assessing whether simulations succeed in tracking the logical consequences of the physical model, but do not constitute a definite answer to the question of whether one should trust their results when evaluating the correctness of the CDM model, and clearly do not allow to eliminate all sources of artifacts.

Convergence studies, however, constitute a mere internal form of robustness, according to which "a numerical prediction of a model is said to be robust if its value does not depend much (or at all) on variation in the value of the input parameters" (Orzack and Sober 1993, 540). Yet, while useful in testing the impact of different parameters and the sensitivity of the model to specific parameter values, such robustness analysis is considered by these authors as

[12]For a detailed discussion of discrepancies between observations and dark matter simulations, notably on the core-cusp density profile problem or the missing satellite problem, see e.g.Bullock and Boylan-Kolchin (2017).

a weaker form of robustness, that is 'no sure sign of truth". By contrast, a stronger form of robustness is attributed to "a property that a proposition has in virtue of its invariance across models" (ibid, 540). Given that the aim of code comparisons is to find invariant properties across different codes, it seems to code comparisons are in a much better position to determine when the outcomes of simulations can be trusted or not. Parker, in her 2011, has however thrown some sand in this too-beautiful-to-be-true wheels. In the case of climate science simulations, she argues, the agreement of different models used to simulate future climate on a specific prediction has no special epistemic significance. She offers two reasons to justify this claim. First, the ensemble of models under comparison is not designed to span a given parameter space, but rather is an ensemble of opportunity. In other words, the ensemble is not one that explore the full uncertainty range associated with, say, the modelling of clouds, but nothing more than the sum of groups willing to participate to this comparison project. Nor do their performance in reproducing the climate features of the past guarantee these model's truth-capturing abilities, for the number of unconstrained parameters values in these models is such that it is easy to tune[13] them so as to reproduce these features. Parker's criticism of the epistemic relevance of robust predictions in climate science can be exported to simulations in astrophysics in a straightforward way. Code comparisons reunites the teams that are willing to participate. Thus, the ensemble of codes under scrutiny in such comparison is not constructed so as to ensure that the comparison bears on models independent in a appropriate way, or spanning a wider or different region of the parameter space. Likewise, the ability of these codes to reproduce known galaxies does not say much about their truth-capturing skills, for the number of unconstrained degrees of freedom in the modelling of the baryonic physics insures that the fit-to-data can always be recovered or inflated. Thus, an agreement across simulations in astrophysics should not be interpreted as having any epistemic significance or to indicate that their outcomes faithfully reflect the underlying physics. Put differently, an agreement across codes does not guarantee that a particular simulation outcome is a genuine theoretical prediction of the CDM cosmology rather than the consequence of a numerical error. Yet, one could still argue that in principle, the possibility exists of constructing an ensemble of codes such that robust predictions would reliably track some truths about the growth of structure in our universe. In this paper, I argue that such a perfectly constructed ensemble would still have to overcome an important obstacle: there is a tension inherent to code comparisons in astrophysics that prevent them to qualify as instances of robustness analysis. Even in such a perfectly constructed ensemble, 'agreement' would not necessarily mean 'robustness'. Robustness is usually considered as qualifying a property that remains

[13] Parker defines the tuning of a model as 'making ad hoc changes to its parameter values or to the form of its equations in order to improve the fit between the model's output and observational/reanalysis data' (Parker 2011, 587).

invariant across models based on different assumptions. What this 'different' amounts to is a controversial matter: some require an agreement across diverse models and their independence, but it is not clear how to unambiguously and unequivocally define this independence. For the purpose of this paper, I will not even require that the ensemble of codes under comparison meet this strong demand of independence, but merely that the ensemble exhibits a diversity such that one can claim, upon observing agreement, that the outcomes of the simulations do not depend on the specifics of the codes, but on the physical model they all implement. That is, I will only demand that two requirements be satisfied:

- Comparability: Simulation runs bear on comparable targets. No lesson could be drawn from a disagreement among codes if the simulated systems are not intended to be the same.
- Diversity: Codes participating in a code comparison differ from each other with respect to at least one of their components and this difference is preserved along the code comparison enterprise. The construction of the ensemble under comparison allows to test the impact of, if not all the assumptions, the ones that are most likely to introduce artifacts in the simulations.

I will argue that recent code comparisons projects do not even satisfy this weakened requirements. In the case of astrophysical simulations, these two requirements are in tension with each other: making codes comparable undermines the project of examining a set of sufficiently diverse models, such as to exclude all possible sources of artifacts.

4 A Hard Choice: Comparability, Diversity and Robustness

In this section, I present two examples of code comparisons. I chose to focus on these two specific projects for two reasons: first, they constitute two of the most recent ones, and of the most important ones in terms of the number of teams involved. Second, they are based on sharply different methodologies, both of which can teach us a lot about robustness analysis. AGORA, the first project I will introduce, was actually motivated by a strong disagreement with the strategy adopted by the AQUILA project, that I will detail subsequently (see 4.3). While AQUILA insists on comparing state-of-the-art codes with their favorite parametrization on the same dark matter halo, AGORA put the effort on insuring that codes actually simulate the same system– that the enterprise is an apple-to-apple comparison. I argue that none of the

methodologies satisfy the minimal requirements of comparability and of diversity that would provide the grounds for robustness analysis. This means that, although code comparisons can teach us a lot about hydrodynamics simulations, they do not allow to exclude all sources of artifacts in simulations and thus to assess whether they are reliable.

4.1 The AGORA Project: making targets comparable

The Assembling Galaxies Of Resolved Anatomy (hereafter AGORA) project was launched in July, 2012, in an impressive attempt to insure that comparisons across codes are actually possible, i.e., that apples are compared to apples[14]. This project currently gathers fourteen teams across the world, with more than 150 participants and 60 institutions involved[15]. The project targets galaxy formation simulations, which require to model baryonic physics. As I mentioned earlier, the methodology of AGORA is driven by the idea to make sure that different codes actually target identical systems, i.e., that code comparisons are actually possible. Indeed, hydrodynamical simulations have numerical parameters left unconstrained, but also unconstrained degrees of freedom in implementing the physics of the cooling, the shocks, or even more significantly of the interstellar medium (ISM). So, not only different codes have different preferences for deciding on the values of numerical and subgrid parameters, but parameters may sometimes not have the same significance across different codes–as force softening does not have the same meaning in tree codes and in mesh-based codes. If codes do not incorporate the same energy feedback (black holes? Supernovae? Cosmic Rays? All of them?) and do not inject them in a similar way– be it the form of the feedback or its parametrization–, how could any lesson be drawn from an agreement or a disagreement across codes? This is why the core of the AGORA project is to develop a framework guaranteeing that codes share their initial conditions and astrophysical packages and can be read using common analysis toolkit.

The project consists in comparing the outcomes of 5 codes, given a common cosmological background: GADGET, GASOLINE, ART ENZO and RAMSES, based on two sets of initial conditions generated by the platform MUSIC (Hahn and Abel 2013) for haloes with $z = 0$ ranging from 10^{10} to 10^{13} solar masses,i.e., from dwarf galaxies as well as of galaxy groups. GADGET and GASOLINE are based on a tree algorithm (and mesh methods for long range forces in the former) for solving gravity, while the fluids are represented using smoothed-particle hydrodynamics (SPH). ART, ENZO and RAMSES, on the other hand, constitute three examples

[14] Private correspondence with Ji-Hoon Kim, first author of the two AGORA publications and project coordinator.

[15] Information retrieved from the website of the project https://sites.google.com/site/santacruzcomparisonproject/outline on March, 4th 2019.

of adaptative mesh refinements codes. The first set of initial conditions corresponds to galaxies forming with a quiescent merger history, while the other describes a violent one, with many mergers between $z = 2$ and 0. A low-resolution large volume is simulated with cosmological parameters chosen in accordance with the ΛCDM cosmology and the WMAP results, while the astrophysical package for gas-cooling, UV background, the stellar initial mass function and mass loss, star formation and supernovae energy feedback is implemented through a common modules such as GRACKLE, which provides a standardized primordial chemistry for H and He as well as a cooling library, or CLOUDY, a photoionization code that provides caluclated rates for metal cooling and photoheating. The subgrid physics parameters that cannot be handled through common modules suach as GRACKLE or CLOUDY, especially those regulating stellar feedback processes, are calibrated on a common isolated disk galaxy scenario whose initial conditions are given by the platform MAKEDISC. The idea is to vary the feedback parameters and the mass and spatial resolutions until all the teams succeed in simulating a realistic disk galaxy. Eventually, the outputs of each codes are used as inputs for the common analysis platform yt[16], which serves to analyze the data in terms of defining and examining field quantities at every point in space and quantities constructed from whole regions in space. Halos are identified within the data set by grouping particles of dark matter depending on their distance to each other, and yt can analyze quantities for these halos such as their density aligned with their angular momentum. Yt is also used for visualizing the outcomes of simulations in 2D or 3D.

A test of this set-up is then performed to ensure that "1) each participating code can read the common "zoom-in" initial conditions generated by the MUSIC code, 2) that each code can perform a high-resolution cosmological simulation within a reasonable amount of computing time, and 3) that the simulation output be analyzed and visualized in a systematic way using the common analysis yt platform" (Kim et al. 2013, 11). The test consists of the dark-matter only simulation of a galactic halo of intermediate size at $z = 0$. This 'proof-of-the concept' test shows great agreement overall, especially on the mass distribution around the central halo, the target halo mass, and the density profiles[17]. The flagship paper thus concludes on a rather optimistic note about the possibility of comparing different codes:

> We have found that the dark matter density profiles as well as the general distributions of matter exhibit good agreement across codes, providing a solid foundation for future hydrodynamic simulations. Throughout the test we have demonstrated the practical

[16] See Turk et al. 2010 for a detailed introduction to yt.

[17] The agreement does not hold for the substructure mass distribution. Possible culprits identified by the authors are a) a small deviation in density distribution evolving into a significant difference, b) a timing mismatch in the numerical integration of the equations of motion, c) an intrinsic difference in solving Poisson equation–i.e., in the gravity solvers.

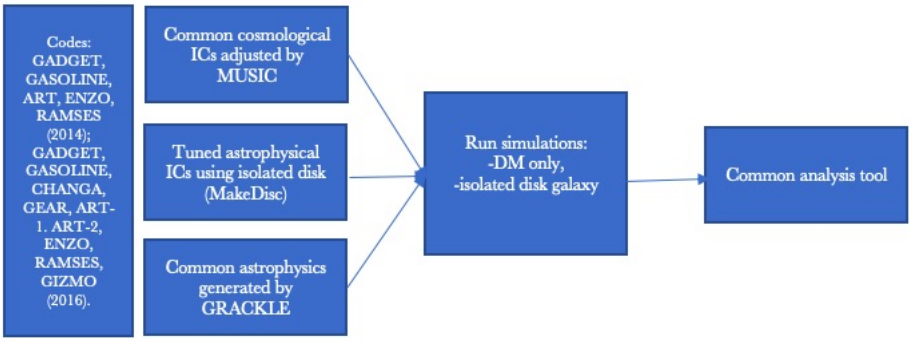

Figure 3: The methodology of the AGORA project. GADGET, GASOLINE, CHANGA and GEAR are Smoothed-Particle Hydrodynamics (SPH) codes; Art-I, Art-II, ENZO and RAMSES Adaptative-Mesh Refinements codes (AMR); and Gizmo a mesh-free code.

> advantage of our common initial conditions and analysis pipeline by showing that each code can read the identical 'zoom-in' MUSIC initial conditions and that each simulation output can be analyzed with a single *yt* script independent of the output format. By doing so, we have produced evidence that the cumbersome barriers in comparing galaxy simulations can be, and are, removed.

The second paper, published in 2016, extended this methodology to nine codes total[18], whose convergence was tested on an isolated Milky Way-size disk galaxy and its properties such as gas/stellar disk morphology and kinematics, the thermal structure of the ISM, or the star formation relation. The conclusion of this paper is even more optimistic than that of AGORA-I, which insists on establishing the comparability of the targets, but is also of a different nature. This time, the authors assert that the comparability of the codes in conjunction with their agreement warrant the *robustness* of the predictions upon which an agreement is found and their reliability :

> Our experiment reveals the remarkable level of agreement between different model simulation tools despite their codebases having evolved largely independently for many years. It is also reassuring that our computational tools are more sensitive to input physics than to intrinsic differences in numerical schemes, and that predictions made by the participating numerical codes are reproducible and likely reliable. If adequately designed in accordance with our proposed common parameters (e.g, cooling, metagalactic UV background, stellar physics, resolution (...)), results of a modern high-resolution galaxy formation simulation are likely robust (Kim et al. 2016, 26).

[18] CHANGA, GASOLINE, GADGET and GEAR for the SPH codes and ART-1, ART-2, ENZO, RAMSES for the AMR codes, and finally GIZMO, a mesh-free code. See the details of each codes in section 5 of Kim et al. 2016.

4.2 Is the diversity requirement satisfied?

The AGORA project developed an impressive and very much needed apparatus to ensure that codes could actually be compared, i.e., to ensure that code comparisons are 'apple-to-apple comparisons'. This attempt to develop common platforms, including common initial conditions (ICs), common astrophysical packages and common analysis tools, was indeed required to make sure that outcomes of simulations were representing the same system. Furthermore, a lot was learned about the codes themselves through this enterprise and will still certainly be in the future. For instance, the AGORA teams decided to take what they called a '0 Myr snapshot' when testing the effectiveness of common platforms, i.e., a snapshot immediately after the ICs were read, in order to determine whether different codes were reading the ICs consistently. This step not only allowed to correct all the possible ways in which the ICs could be misread by the codes–wrong units, wrong definition, wrong convention–[19], but also permitted to gain some insights about intrinsic differences between mesh-based codes like ART I and II, ENZO, RAMSES, GIZMO and particle-based ones like GADGET, GEAR, CHANGA or GASOLINE. An example of this is illustrated by figure 1 of Kim et al. 2016, that shows the differences between these hydro-solvers for the surface density of disk galaxies: particle-based codes seem "to smooth out the strong density contrast in the ICs at the edge of the initial gas disk" (2016, 10), due to the way the density is reconstructed from the positions of particles in these codes. Nevertheless, it does not follow from the comparability of codes that shared predictions are robust. In this context, the effort to make the comparison possible is actually in tension with the diversity required for the search of robust properties and the exclusion of all sources of artifacts.

Very early in the development of simulations in cosmology, some astrophysicists have pointed out the fact that code comparisons are useless in those cases where the source of numerical artifacts is a shared assumption. Discreness-driven artifacts, for instance, find their origin in the discreteness of dark matter due to mass resolution limitations, a necessary assumption of all codes to this day. Such discreteness-driven artifacts seeds spurious fluctuations whose consequences have been detected in the form of two-body relaxation, mass segregation effects or spurious fragmentation of haloes in the numerical experiments comparing outcomes of simulations to known exact solutions performed in Efstathiou and Eastwood (1981), Melott et al. (1997), Binney and Tremaine (2011), or more recently in Wang and White (2007), among many others[20], but never in code comparisons, with good reasons: a common assumption leads to a common error. The same could be said about the periodicity of the simulated volume, or to a lesser[21] extent

[19] The nature of these corrections is not detailed in the paper itself, but was in private correspondence with Ji-Hoon Kim.

[20] See for instance the comments on this last paper by Melott

[21] Force softening is used in all hydrodynamical simulations but has a different 'meaning' in each codes, as force

about force softening, Thus, code comparisons are already exposed to an important flaw: the impossibility of assessing the impact of an unavoidable assumption such as the discretization of the dark matter. The problem is that the effort undertaken by the AGORA teams to make a comparison across codes possible indeed succeed in making the targets comparable, but they also amplify the problem that was already at the heart of code comparisons: that is, to introduce even more common assumptions whose impact cannot be assessed through a codes comparison, given that they are common to each code and thus are not themselves tested. To be sure, I do not claim that implementing a common astrophysics package–a similar rate for metal cooling, for instance –or common initial conditions is a problem *per se*. On the contrary, doing so is necessary to satisfy the comparibility requirement as defined above. The problem is that it is done through extra codes or modules which themselves have problematic assumptions, including that of the discreteness of the dark matter.

How can one guarantee that these common tools and their own idealizations or numerical errors are not introducing new sources of numerical artifacts, since these elements themselves are not systematically varied in the codes comparison? Consider the example of MUSIC (MUlti-Scale Initial Conditions), for instance, which is conceived as a way to generate common cosmological initial conditions for zoom-in simulations readable by all the codes involved in the comparison. By initial conditions, one must understand here, roughly speaking, the distribution of particles in a given volume–their initial load in a CDM+ baryonic physics context. Remember that for zoom-in simulations, one must first carry out a low-resolution simulation of a large volume, in order to have an idea of the large-scale cosmic environment where haloes structure will develop, before zooming-in on a particular object at higher resolution. The challenge is thus to generate initial conditions for both scales: since the density perturbations in the cold dark matter are responsible for structure formation, and these perturbations extend from solar system size to giga parsec scales, a reliable simulation of an individual halo in its cosmic environment must be able to track both smaller scale perturbations directly impacting the halo structure and the large-scale perturbations. The aim of MUSIC is to provide multi-scale initial conditions by generating "Gaussian random fields that follow a prescribed power spectrum and act as source terms for density and velocity perturbations in Lagrangian perturbation theory" (Hahn and Abel 2013, 2).[22] MUSIC itself is a code, based on a number of assumptions, idealizations and numerical parameters, including the discreteness of dark matter, or the force softening, that expose

softening does not have any physical meaning but is defined through the specific assumptions meant to prevent divergent forces.

[22]More precisely, MUSIC uses Fast-Fourier Transformations convolutions to obtain the density field from a hierarchical white noise field, and an adaptative multigrid Poisson solver for displacement and velocity fields. As is usually the case in different codes, the Fast-Fourier Technique (FFT) is used for long-range forces, but given their poor resolution, are complemented by other techniques when higher resolution is needed, such as tree methods. See Hahn and Abel 2013, sections 2 and 3 for more details.

it to numerical artifacts. How do we know that MUSIC itself is free of errors and immune to numerical artifacts that would similarly impact all codes, especially when we know already that the discreteness assumption results in artifacts?[23]

Let me give an example. It had been known since Bode et al. (2001) that simulations based on Warm Dark Matter, i.e., dark matter made of particles whose thermal velocity is higher than 'cold' particles, show 'beads-on-a-string haloes', regularly spaced haloes along filaments. In the early universe, density fluctuations acted as seeds for structure formation through gravitational collapse. Such rapid motion particles, however, would have washed out small-scale inhomogeneities, and only large-scale fluctuations would have survived. Thus, structure forms top-down in such a universe: large scale structure form first, and smaller structures appear through the fragmentation of the latter. As a consequence, beads-on-a-string haloes were considered as a natural 'physical' consequence of the model. When Götz and Sommer-Larsen (2003) showed that the spacing of the haloes was equal to the grid spacing, the artificial nature of these haloes became clear: the dependence of this artefact upon the grid simply means that new techniques for distributing particles over the simulated volume would circumvent the problem and that the spurious fragmentation is anchored in the specifics of the way particles are distributed within the volume[24]. Given that MUSIC provides the seeds from which all codes can be run, if artifacts were already there in the initial conditions, how would we be able to determine whether the convergence obtained across codes is due to the physics or to these initial artifacts? Likewise, how do we know that the interplay of MUSIC and N-body codes under comparison does not create artifacts, compensate errors or amplify the effects of those already emphasized? The exact same reasoning can be applied to the code GRACKLE and its sub-module CLOUDY about the astrophysical sub-packages they implements, especially given the difficulty that astrochemists still face in reproducing the primordial chemistry, and for the analysis toolkit *yt*. *Yt* is especially worrying, given that it takes as inputs the outputs of each codes: how can we make sure that the analysis algorithm does not smooth the differences between codes at least for some given observables? These common platforms serve their role, inasmuch as they do allow for a comparison on similar grounds for all codes, but they also hinder the search for the diversity needed to proceed to a sound robustness analysis by introducing even more common assumptions whose effects remain unnoticeable. Again, what I mean here is that making targets comparable requires the addition of common elements that will be introduced through coding or modelling assumptions that themselves are not tested within the code comparison.

[23] The authors argue that MUSIC is not free of errors, but that either the errors are below a 10^{-4} significance threshold, or they are confined to the boundaries. They obtained these results, however, based on a code comparison and observations matching.

[24] Note that the Wang and White (2007) paper challenges this conclusion and suggests that the spurious haloes stems from discreteness effects

A second worry stems from the tuning of the astrophysical physics that is not handled through common platforms, like the star formation and stellar feedback parameters, and the subsequent forced agreement imposed on all codes. Star formation and stellar feedback parameters are known to be crucial in modelling the baryonic physics guiding galaxy formation, but cannot in any way be calibrated against theoretical calculations. Hence, each code proceeds to calibrate these feedback processes as they prefer, in order to reproduce an isolated Milky-way like galaxy. Different calibrations may yield similar results, given the degeneracies that exist between numerical parameters[25].

Before addressing this worry, we need to pause and look at an older code comparison undertaken in 2012. This project, called 'AQUILA', greatly contributed to motivate the methodology of AGORA, for its disappointing results were thought to entirely stem from the failure of the project to offer comparable targets.

4.3 An earlier attempt: the AQUILA project

The project AQUILA was based on a comparison among six versions of GADGET3, GASOLINE, one adaptative mesh-refinement (RAMSES) and a moving mesh (AREPO), each of these codes which has its own preferred treatment of radiative cooling, star formation and its own numerical treatment of feedback. One version of GADGET3 and RAMSES were also run three times with different subgrid physics modules: G3 only included supernovae feedback whereas G3-BH also considered the energy feedback of supermassive black holes and G3-CR that of the energy deposition of cosmic rays. Likewise, RAMSES was run with longer star formation time-scale as compared to the fiducial run[26] of RAMSES (in the RAMSES-LSFE), and also with adding the feedback energy of an active galactic nuclei associated with a supermassive black hole (see figure below). The idea was to compare the outcomes of these thirteen zoom-in of one of the haloes of the Aquarius project named 'Aq-c'.[27].

The results of this project were expressed as follows: "Although numerical convergence

[25] Schaye et al. (2015), for instance, describe an example where the supernovae feedback subgrid model is too inefficient because numerical error cause too much of the energy to be radiated away, or too much of the momentum to cancel out, or the energy/momentum to be coupled to the gas at the wrong scale. In that case, the unawereness of such numerical problems will lead to think that additional feedback processes is required, say radiation pressure. But a too efficient numerical implementation of the supernovae feedback would caue to underestimate the need for radiation pressure. Both calibration would however yield the same result, without this calibration to allow for any interesting insight into these processes (p.523)

[26] By 'fiducial', I mean the run of the ordinary RAMSES with no extra or modified prescription, which serves as a basis for comparison in the interpretation of the 'modified' version of RAMSES.

[27] Aquarius is a collaborative project similar in scope and scale to the Millenium simulation, which provides ultra-high resolution simulations of 6 Milky-way size individual dark matter haloes, named Aq-A, B, C D, E and F. See table 1 in the flagship paper of the project for more details about the haloes' properties (Springel et al. (2008)).

Table 1. Summary of code characteristics and subgrid physics.

Code	Reference	Type	UV background (z_{UV})	UV background (spectrum)	Cooling	Feedback
G3 (GADGET3)	[1]	SPH	6	[10]	Primordial [13]	SN (thermal)
G3-BH	[1]	SPH	6	[10]	Primordial [13]	SN (thermal), BH
G3-CR	[1]	SPH	6	[10]	Primordial [13]	SN (thermal), BH, CR
G3-CS	[2]	SPH	6	[10]	Metal dependent [14]	SN (thermal)
G3-TO	[3]	SPH	9	[11]	Element-by-element [15]	SN (thermal+kinetic)
G3-GIMIC	[4]	SPH	9	[11]	Element-by-element [15]	SN (kinetic)
G3-MM	[5]	SPH	6	[10]	Primordial [13]	SN (thermal)
G3-CK	[6]	SPH	6	[10]	Metal dependent [14]	SN (thermal)
GAS (GASOLINE)	[7]	SPH	10	[12]	Metal dependent [16]	SN (thermal)
R (RAMSES)	[8]	AMR	12	[10]	Metal dependent [14]	SN (thermal)
R-LSFE	[8]	AMR	12	[10]	Metal dependent [14]	SN (thermal)
R-AGN	[8]	AMR	12	[10]	Metal dependent [14]	SN (thermal), BH
AREPO	[9]	Moving mesh	6	[10]	Primordial [13]	SN (thermal)

Note: [1] Springel et al. (2008); [2] Scannapieco et al. (2005), Scannapieco et al. (2006); [3] Okamoto et al. (2010); [4] Crain et al. (2009); [5] Murante et al. (2010); [6] Kobayashi, Springel & White (2007); [7] Stinson et al. (2006); [8] Teyssier (2002), Rasera & Teyssier (2006), Dubois & Teyssier (2008); [9] Springel (2010a); [10] Haardt & Madau (1996); [11] Haardt & Madau (2001); [12] Haardt & Madau (private communication); [13] Katz et al. (1996); [14] Sutherland & Dopita (1993); [15] Wiersma, Schaye & Smith (2009a); [16] Shen, Wadsley & Stinson (2010).

Figure 4: This table coming from Scannapieco et al. 2012 (1729) summarizes the different types of gravity and hydro solvers, the preferred way of calculating the radiative cooling, the kind of feedback considered (Supernovae, Black Holes, Cosmic Rays) of the different (versions of the) codes and the way it was implemented (through the injection of thermal or kinetic energy into the ISM or into the gas itself).

is not particularly good for any of the codes, reasonably good convergence is found for the properties of the stellar component, such as total mass and median age. Less well converged are the internal properties of the galaxy, such as the half-mass radius, or the fraction of stars in a rotationally supported disc. [...] Aside from these considerations, perhaps the main result of the AQUILA project is that, despite the large spread in properties spanned by the simulated galaxies, none of them has properties fully consistent with theoretical expectations or observational constraints in terms of mass, size, gas content and morphology". The claim that numerical convergence is not good enough, repeated multiple times throughout the paper, indicates that (Scannapieco et al. 2012, 1742). Looking at the diversity of the physics implemented though, the deceptive results of AQUILA seem no longer as surprising as they appear at first glance, and not even as worrying as presented by the authors of the project. The divergence of the results obtained by different codes could very well be explained by the fact that these codes do not consider the same physics to begin with. As a result, no rushed conclusion should be made about the unreliability of N-body simulations from AQUILA's divergent results–the results could be different merely because they compare different things. Hence, the AGORA project focused on building a common infrastructure to allow for a genuine comparison of similar targets.

However, AQUILA, with its own methodology, was able to deliver very interesting insights that must not be forgotten when interpreting AGORA's results, especially when focusing

on the tuning of the astrophysics based on an isolated disk scenario. In particular, the results of AQUILA about galaxy morphology highlight the importance of stellar feedback and star formation for the morphology of galaxies. The delayed star formation of the RAMSES-LSFE version of the code, with delayed star formation, shows that the later the gas turns into stars, the more prominent the disc will be. Delaying star formation gives time for the gas to accrete into a centrifugally supported structure and thus promotes the apparition of a thin disk. On the other hand, the earlier stars form, the more galaxies will tend to be spheroidal.[28] These conclusions are supported by the fact that G3 and AREPO, which share their subgrid physics but differ on their hydro solvers, both lack discs in the simulated halo, making it more likely to find the culprit in the astrophysics than in the numerical scheme; and that codes with more efficient feedback, preventing the stars to form too early, do exhibit a disc, although less prominent that the one displayed by RAMSES-LSFE. However, star formations parameters and feedback modules[29] in the AGORA project are individually tuned so as to produce a realistic disk, thus erasing the differences met in the AQUILA project:

> It is of primary importance to understand how each individual code needs to be calibrated to reproduce various observational constraints. In a comparison like the AGORA project, it is even more important to cross-calibrate stellar feedback processes of the various codes using an idealized set-up such as an isolated disk. This is precisely the goal of this second type of initial conditions: we would like to model a realistic galactic disk using our various codes and their feedback parameters and the mass and spatial resolutions. By doing so, subgrid star formation and feedback prescriptions in various code platforms will be tuned to provide a realistic interstellar and circumgalactic medium (Kim et al. 2013, 6).

How then can any conclusion be made about the final agreement of these codes on galaxy morphology? How can one determine whether this agreement stems from the reliability of the predictions made or from this forced initial agreement obtained by individually tuning the codes? This is all the more worrying that the tuning is made in the AGORA project to produce a realistic disk galaxy, when in the AQUILA project, none of the codes–including GADGET and RAMSES, which are involved in both code comparisons– were able to produce realistic results, compatible with theoretical expectations and observations. One might be concerned, in this case, by the extent to which codes have been tuned to agree on a realistic disk.

[28] See figure 4 and its interpretation, Scannapieco et al. 2012, 1731-1732.

[29] More specifically, the relevant tuned parameters includes: star formation density threshold, star formation efficiency, initial mass of star particles, stochasticity of star formation. See section 3.2. of Kim et al. 2013.

The reduction of diversity generated by the comparability requirement can be summarized as follows:

- Common idealizations or assumptions made by different codes cannot be tested within a code comparison. Such common assumptions are already present within code comparison, at least due to the discretization of dark matter. The appeal to common new modules implies the introduction of new common assumptions such as the specifics of the initial load of particles, or a specific modelling choice for the primordial chemistry of H and He, that cannot be tested within the code comparison.
- Common modules may interfere or create compensating errors among different parts of the modules that, likewise, would not be detected by this code comparison.
- The tuning of the feedback processes that cannot be constrained theoretically could although contribute erase important differences between codes, especially as the way feedback processes interfere is not well understood.
- The analysis platform *yt* uses as inputs the outputs of each code. An artifact in, say, the pairing algorithm, would affect similarly the result of eacu code, and thus again contribute to erase the differences that the different numerical solvers or different numerical schemes for implementing feedback processes may have yield otherwise.

To wrap it up, I do not consider that code comparisons can be considered as instances of robustness analysis, because of the incompatibility that a project like AGORA exemplifies between 1) making sure that the targets are made comparable and 2) providing a diverse set of codes to compare in order to eliminate all possible sources of numerical artifacts, given the addition of common assumptions that is required, and the tuning and subsequent forced agreement necessary to calibrate the physics that cannot be handled through common platforms. Such projects allows to determine whether codes comparable, but not to draw any conclusion with respect to whether predictions based on simulations are genuine predictions of the model or numerical artifacts.

5 Conclusion

In this paper, my aim was to assess whether code comparisons in astrophysics can provide more solid grounds for robustness analysis than convergence studies, such as to complement the argument against robustness analysis provided by Gueguen (2020). There are two ways

to proceed to such a comparison. Either state-of-the-art codes are compared, each team using their favourite calibration and subgrid physics–which feedback they want to implement and how. Such a methodology however fails to compare similar targets. Other code comparisons focus on developing a common infrastructure guaranteeing that codes read the initial conditions in the same way, implement, whenever possible, the same subgrid physics, and can be analysed using the same tools: in other words, that different codes actually simulate comparable targets. The problem, however, is that the common infrastructure itself becomes an unanalyzed source of artifacts. When cosmological codes agree in such a context, there is no conclusive way to determine whether the physical model or the common infrastructure are responsible for the convergence of the results. Moreover, the part of the subgrid physics that cannot be commonly implemented is in those cases arbitrarily calibrated to fit a predetermined outcome. How can we know then whether convergence across codes finds its origin in the common infrastructure, in the thorough tuning of part of the subgrid physics, or in the physical model? In sum, there is an inherent tension within code comparisons between the diversity of models that must be considered in order to find robust properties and the common infrastructure required to make the comparison possible. As long as this tension is not resolved, a code comparison cannot deliver what it is supposed to–i.e., a method for determining when simulations faithfully track the logical consequences of the physical model. I expect this tension to apply to areas beyond that of astrophysics, especially in climate sciences, where the assessment of simulations becomes extremely challenging given their complexity and potential 'fuzzy' modularity. But such a claim requires the careful scrutinization of how the comparibility requirement is satisfied in such areas, and this analysis would go beyond the scope of this paper. I am hoping that a brave reader will follow-up on the task of unravelling whether the worries invoked at the end of section 4.3 extend to this area of research similarly!

Acknowledgements

I am very grateful to Chris Smeenk for his continuous support and his helpful feedback. I would also like to thank Helen Meskhidze, John Norton and the visiting fellows of the Center for the Philosophy of Science at the University of Pittsburgh (2019-2020) for insightful discussions and two anonymous referees for their helpful feedback. This paper is based on work done while funded as a graduate student researcher under the John Templeton Foundation grant: "New Directions in Philosophy of Cosmology" (grant number 61048).

References

Baushev, A., L. del Valle, L. Campusano, A. Escala, R. Muñoz, and G. Palma (2017). Cusps in the center of galaxies: a real conflict with observations or a numerical artefact of cosmological simulations? *Journal of Cosmology and Astroparticle Physics 2017*(05), 042.

Baushev, A. N. (2015). The real and apparent convergence of n-body simulations of the dark matter structures: Is the navarro–frenk–white profile real? *Astroparticle Physics 62*, 47–53.

Binney, J. and S. Tremaine (2011). *Galactic dynamics*, Volume 20. Princeton university press.

Bode, P., J. P. Ostriker, and N. Turok (2001). Halo formation in warm dark matter models. *The Astrophysical Journal 556*(1), 93.

Bodenheimer, P., G. P. Laughlin, M. Rozyczka, T. Plewa, H. W. Yorke, and H. W. Yorke (2006). *Numerical methods in astrophysics: an introduction*. Taylor & Francis.

Bullock, J. S. and M. Boylan-Kolchin (2017). Small-scale challenges to the λcdm paradigm. *Annual Review of Astronomy and Astrophysics 55*(1), 343–387.

Efstathiou, G., M. Davis, S. White, and C. Frenk (1985). Numerical techniques for large cosmological n-body simulations. *The Astrophysical Journal Supplement Series 57*, 241–260.

Efstathiou, G. and J. Eastwood (1981). On the clustering of particles in an expanding universe. *Monthly Notices of the Royal Astronomical Society 194*(3), 503–525.

Frenk, C., S. White, P. Bode, J. Bond, G. Bryan, R. Cen, H. Couchman, A. E. Evrard, N. Gnedin, A. Jenkins, et al. (1999). The santa barbara cluster comparison project: a comparison of cosmological hydrodynamics solutions. *The Astrophysical Journal 525*(2), 554.

Götz, M. and J. Sommer-Larsen (2003). Galaxy formation: Warm dark matter, missing satellites, and the angular momentum problem. In *The Evolution of Galaxies*, pp. 47–50. Springer.

Gueguen, M. (2020). On robustness in cosmological simulations. *Philosophy of Science 87*(5), 1197–1208.

Hahn, O. and T. Abel (2013, November). MUSIC: MUlti-Scale Initial Conditions.

Jacquart, M. (2020). Observations, Simulations, and Reasoning in Astrophysics.

Kim, J.-h., T. Abel, O. Agertz, G. L. Bryan, D. Ceverino, C. Christensen, C. Conroy, A. Dekel, N. Y. Gnedin, N. J. Goldbaum, et al. (2013). The AGORA high-resolution galaxy simulations comparison project. *The Astrophysical Journal Supplement Series 210*(1), 14.

Kim, J.-h., O. Agertz, R. Teyssier, M. J. Butler, D. Ceverino, J.-H. Choi, R. Feldmann, B. W. Keller, A. Lupi, T. Quinn, et al. (2016). The AGORA high-resolution galaxy simulations comparison project. II. Isolated disk test. *The Astrophysical Journal 833*(2), 202.

Klypin, A. (2017). Methods for cosmological N-body simulations.

Lenhard, J. and E. Winsberg (2010). Holism, entrenchment, and the future of climate model pluralism. *Studies in History and Philosophy of Science Part B: Studies in History and Philosophy of Modern Physics 41*(3), 253–262.

Ludlow, A. D., J. Schaye, M. Schaller, and R. Bower (2020). Numerical convergence of hydrodynamical simulations of galaxy formation: the abundance and internal structure of galaxies and their cold dark matter haloes. *Monthly Notices of the Royal Astronomical Society 493*(1), 2926–2951.

Melott, A. L. (2007). Comment on 'Discreteness effects in simulations of Hot/Warm Dark Matter' by J. Wang & SDM White.

Melott, A. L., S. F. Shandarin, R. J. Splinter, and Y. Suto (1997). Demonstrating discreteness and collision error in cosmological n-body simulations of dark matter gravitational clustering. *The Astrophysical Journal Letters 479*(2), L79.

Orzack, S. H. and E. Sober (1993). A critical assessment of Levins's The strategy of model building in population biology (1966). *The Quarterly Review of Biology 68*(4), 533–546.

Parker, W. S. (2011). When climate models agree: The significance of robust model predictions. *Philosophy of Science 78*(4), 579–600.

Peebles, P. (1970). Structure of the Coma cluster of galaxies. *The Astronomical Journal 75*, 13.

Power, C., J. Navarro, A. Jenkins, C. Frenk, S. D. White, V. Springel, J. Stadel, and T. Quinn (2003). The inner structure of ΛCDM haloes—i. A numerical convergence study. *Monthly Notices of the Royal Astronomical Society 338*(1), 14–34.

Sanderson, R. E., A. Wetzel, S. Loebman, S. Sharma, P. F. Hopkins, S. Garrison-Kimmel, C.-A. Faucher-Giguère, D. Kereš, and E. Quataert (2020). Synthetic gaia surveys from the fire cosmological simulations of milky way-mass galaxies. *The Astrophysical Journal Supplement Series 246*(1), 6.

Scannapieco, C. e. a., M. Wadepuhl, O. Parry, J. Navarro, A. Jenkins, V. Springel, R. Teyssier, E. Carlson, H. Couchman, R. Crain, et al. (2012). The Aquila comparison project: the effects of feedback and numerical methods on simulations of galaxy formation. *Monthly Notices of the Royal Astronomical Society 423*(2), 1726–1749.

Schaye, J., R. A. Crain, R. G. Bower, M. Furlong, M. Schaller, T. Theuns, C. Dalla Vecchia, C. S. Frenk, I. McCarthy, J. C. Helly, et al. (2015). The eagle project: simulating the evolution and assembly of galaxies and their environments. *Monthly Notices of the Royal Astronomical Society 446*(1), 521–554.

Shandarin, S. F. and Y. B. Zeldovich (1989). The large-scale structure of the universe: Turbulence, intermittency, structures in a self-gravitating medium. *Reviews of Modern Physics 61*(2), 185.

Smeenk, C. and S. C. Gallagher (2020). Validating the universe in a box. *Philosophy of Science 87*(5), 1221–1233.

Springel, V., J. Wang, M. Vogelsberger, A. Ludlow, A. Jenkins, A. Helmi, J. F. Navarro, C. S. Frenk, and S. D. White (2008). The Aquarius project: the subhaloes of galactic haloes. *Monthly Notices of the Royal Astronomical Society 391*(4), 1685–1711.

Turk, M. J., B. D. Smith, J. S. Oishi, S. Skory, S. W. Skillman, T. Abel, and M. L. Norman (2010). Yt: A multi-code analysis toolkit for astrophysical simulation data. *The Astrophysical Journal Supplement Series 192*(1), 9.

van den Bosch, F. C. and G. Ogiya (2018). Dark matter substructure in numerical simulations: a tale of discreteness noise, runaway instabilities, and artificial disruption. *Monthly Notices of the Royal Astronomical Society 475*(3), 4066–4087.

van den Bosch, F. C., G. Ogiya, O. Hahn, and A. Burkert (2017). Disruption of dark matter substructure: fact or fiction? *Monthly Notices of the Royal Astronomical Society 474*(3), 3043–3066.

Wang, J. and S. D. White (2007). Discreteness effects in simulations of hot/warm dark matter. *Monthly Notices of the Royal Astronomical Society 380*(1), 93–103.

Weisberg, M. and K. Reisman (2008). The robust volterra principle. *Philosophy of science 75*(1), 106–131.

Winsberg, E. (2009). Computer simulation and the philosophy of science. *Philosophy Compass 4*(5), 835–845.

Winsberg, E. (2010). *Science in the age of computer simulation*. University of Chicago Press.

Zel'Dovich, Y. B. (1970). Gravitational instability: An approximate theory for large density perturbations. *Astronomy and astrophysics 5*, 84–89.