



**Universidad**  
de La Laguna



Trabajo de Fin de Máster

---

# The effect of IllustrisTNG AGN feedback model on galaxy size

---

*Efecto del modelo de AGN feedback de IllustrisTNG  
en el tamaño de galaxias*

**Andrés García-Serra Romero**

Tutorizado por:

**Dr. Claudio Dalla Vecchia**

**Dr. Andrea Negri**

**Máster en Astrofísica**

**2022-2023**

*Especial agradecimiento a Leti, Chema, Jorge y Javi,  
por apoyo y ayuda en muchos momentos de incertidumbre.*

*A mi familia: Bea, Carlos, Pau y Miguel,  
por su apoyo emocional e incondicional.*

*A mis tutores: Claudio y Andrea,  
por su ayuda cercana y su paciencia.*

*Y finalmente a Nnette, allá donde estés,  
espero que el cosmos te esté cuidando.*

## RESUMEN

Los procesos radiativos provenientes de los agujeros negros supermasivos en el centro de galaxias son la principal causa de la disminución de la formación estelar así como uno de los principales mecanismos que rigen la evolución de las galaxias tal y como las conocemos.

Los procesos de *feedback* surgen cuando el agujero negro central acreta masa de los alrededores del ISM (*Inter Stellar Medium*), formando un disco de acreción que, por fuerzas de cizalladura, disipa la energía cinética de estas estrellas como energía térmica irradiada al ISM. La luminosidad de los agujeros negros ( $L_{\text{BH}}$ ) es proporcional a su tasa de acreción. Estos agujeros negros centrales son conocidos como AGN (*Active Galactic Nuclei*) y los procesos radiativos presentes como *feedback*, o realimentación.

Según la tasa de acreción de los AGN existen dos principales modos de *feedback*. El modo "radio" se caracteriza por una tasa de acreción baja y una acreción de gas caliente en los alrededores del ISM siguiendo una geometría esférica. Este modo se caracteriza por la emisión en ondas de radio y en él domina el conocido como *feedback* mecánico o cinético, caracterizado por *jets* de partículas y lóbulos de radio de plasma relativista que interactúan de forma mecánica con el ISM, provocando la expulsión del material cercano al AGN. Por otra parte, el modo "cuásar" (de *quasar*, *quasi-stellar radio source*) está caracterizado por una tasa de acreción más rápida con geometría discal y un mecanismo dominado por radiación, con fugas de material breves y repentinas, seguidas de rápidos procesos de acreción que permiten la emisión de estos fenómenos de radiación, siguiendo un esquema pulsante de emisión y acreción. En el caso particular en el que la emisión del cuásar se da en la línea de visión del observador, este se denomina blazar, un objeto más brillante con las mismas características de emisión pero con una luminosidad mucho mayor.

En esta tesis utilizaremos los resultados obtenidos de la relación de tamaño-masa de galaxias del proyecto de simulaciones EAGLE y realizaremos los mismos cálculos y relaciones para otro set de datos proveniente de las simulaciones IllustrisTNG, puesto que ambas simulaciones tienen modelos de *feedback* muy distintos. Esto permite estudiar el impacto de estos diferentes modelos en la relación tamaño-masa y, de forma indirecta, saber el impacto del *feedback* de AGN en el tamaño de las galaxias.

Respecto a los modelos de *feedback* de AGN, en ambos casos se estima la tasa de acreción como una comparación entre la tasa de acreción de Eddington y la de Bondi, pero en el caso de IllustrisTNG se emplean dos modos de *feedback*, mientras que en EAGLE sólo se emplea uno. En el caso de IllustrisTNG se contempla un límite para el cambio entre ambos modelos escalado a cierta masa; de esta forma se fuerza que el modo de *feedback* cambie para galaxias de mayor masa y sea mucho más agresivo. En el caso de EAGLE existe un único modo de *feedback* que libera cierta cantidad de energía por el agujero negro a las partículas cercanas del ISM, siguiendo una probabilidad de absorción por parte de éste. Ambos modelos

utilizan algoritmos similares para la creación (*seeding*) de agujeros negros, aunque el modelo de IllustrisTNG presenta más masa inicial de los agujeros negros frente a EAGLE, con  $8 \times 10^5 M_{\odot} h^{-1}$  frente a  $10^5 M_{\odot} h^{-1}$ .

Para nuestro estudio utilizaremos una definición de radio de galaxia que no es dependiente de la distribución de luz de ésta sino de la morfología y extensión de la misma. Esta definición, propuesta en la literatura, se basa en el radio en el que la densidad de estrellas es igual a  $1 M_{\odot}/\text{pc}^2$ , nombrado  $R_1$ . Recientemente esta definición de radio ha sido propuesta como una alternativa más intuitiva para delimitar el tamaño real observable de galaxias frente a otros indicadores de tamaño como el radio a magnitud superficial 29 mag/arcsec<sup>2</sup> ( $R_{29}$ ) o el radio que contiene la mitad del total de la masa estelar de la galaxia ( $R_H$ ,  $R_{50}$ ), los cuales son altamente dependientes del perfil de densidad o luminosidad de la galaxia.

Para el desarrollo de la tesis hemos tomado los datos de IllustrisTNG y hemos seleccionado un set de galaxias con masas estelares entre  $10^{8.5} M_{\odot}$  y  $10^{14} M_{\odot}$ . Hemos realizado un pre-procesamiento de los datos previo al cálculo del radio de cada galaxia utilizando la definición de  $R_1$  y hemos recogido los resultados junto con otros parámetros necesarios para su análisis en archivos binarios del tipo ".hdf5". Después, mediante otro fichero de análisis hemos desarrollado estos datos, obteniendo la relación tamaño-masa de las galaxias seleccionadas y comparando los resultados obtenidos con los de EAGLE.

Los resultados obtenidos tienen gran similitud con los obtenidos previamente en EAGLE, pese a tener un nivel de dispersión mayor, que hemos atribuido a la dispersión de las masas de las galaxias seleccionadas. Esto nos lleva a la conclusión de que el *feedback* de AGN no tiene efecto sobre el tamaño de las galaxias utilizando esta definición de radio de galaxia, por lo que esta definición es útil en el caso de querer estudiar el tamaño de galaxias de forma independiente a los AGNs centrales.

También hemos observado un decaimiento en la tasa de formación estelar en las galaxias con masas superiores a  $10^{10.5} M_{\odot}$  que hemos atribuido al cambio del modo de AGN *feedback* de IllustrisTNG, que fuerza el cambio en el modo de *feedback* mediante un escalado a la masa del agujero negro. Este fenómeno ha sido observado previamente en la literatura y podemos verlo presente también en la relación tamaño-masa mencionada anteriormente.

La estructura del trabajo consta de un capítulo introductorio que plantea el problema inicial a resolver e introduce los distintos modelos de *feedback* con más detalle. A continuación, el segundo capítulo expone los datos utilizados en el proyecto, explicando los distintos tipos de archivos y formatos presentes en la simulación IllustrisTNG así como los parámetros de interés y la ordenación de los datos utilizados. El tercer capítulo presenta la metodología utilizada para el desarrollo del proyecto, introduciendo en primer lugar el hardware y software utilizado y entrando en más detalle en los distintos códigos desarrollados y la explicación de la rutina de cálculo de  $R_1$ , todo esto apoyándose en distintos diagramas de flujo para una mayor comprensión de los algoritmos. En el cuarto se enuncian los resultados obtenidos, mostrando la relación tamaño-masa y el estudio de las dispersiones de

la misma, así como una breve discusión de las distintas figuras obtenidas y su comparación con la literatura. Finalmente el último capítulo recoge las conclusiones del trabajo, que verifican la validez de los resultados obtenidos, así como algunas ideas y posibles mejoras para la continuidad del proyecto.

---

### ABSTRACT

The radiative processes emitted by supermassive black holes (SMBHs) are world-wide known to have an important effect on the evolution and star formation history of their host galaxy. Active Galactic Nuclei (AGN) feedback is thought to be the main mechanism of star formation quenching in galaxies. In this project, we study the impact of the AGN feedback on the size evolution of galaxies. For this, we use a galaxy size definition that is only dependent on the galaxy morphology and not on the galaxy light distribution. We use the results obtained with the EAGLE simulation data and compute the same calculations for an algorithm with a very different AGN feedback model, which is that of the IllustrisTNG simulation project. We find very similar results in the mass-size relation of both simulations, with a central slope of  $\beta = 0.34 \pm 0.01$  in the case of IllustrisTNG. The overall results of both simulations are very similar, indicating that AGN feedback does not impact the size of galaxies using  $R_1$  as a galaxy size indicator. We also find a large scatter of the fitting residuals of the mass-size relation in IllustrisTNG, which is believed to be caused by the IllustrisTNG galaxy stellar mass distribution. Lastly, an important abrupt change in the star formation rate is addressed for IllustrisTNG simulations for stellar masses of  $10^{10.5} M_{\odot}$ , also visible in the mass-size relation.

# Contents

<b>1</b>	<b>Motivation</b>	<b>7</b>
1.1	Background . . . . .	7
1.2	Cosmological simulations . . . . .	8
1.3	IllustrisTNG and EAGLE simulations . . . . .	8
<b>2</b>	<b>Data</b>	<b>13</b>
2.1	File types . . . . .	14
2.1.1	Snapshots and stellar data . . . . .	14
2.1.2	Friends-of-friends files and subhalo data . . . . .	15
2.1.3	The offsets file . . . . .	16
2.2	Simulation volumes and resolutions . . . . .	17
2.3	Cosmology and units . . . . .	18
<b>3</b>	<b>Methodology</b>	<b>19</b>
3.1	Software and hardware tools . . . . .	19
3.2	The code . . . . .	20
3.2.1	Original EAGLE analysis code . . . . .	20
3.2.2	The new IllustrisTNG input and analysis code . . . . .	20
3.3	File organization . . . . .	26
<b>4</b>	<b>Results and discussion</b>	<b>28</b>
4.1	Mass-size relation . . . . .	28
4.2	Mass-size residual distributions and scatter . . . . .	30
<b>5</b>	<b>Conclusion</b>	<b>33</b>

# Chapter 1

## Motivation

*This chapter of the thesis serves as an introduction and background on the cosmological simulation basics, as well as a description of the questions that the project is aiming to answer. We start by introducing the framework of the thesis and address the motivation paper and its definition of galaxy size ( $R_1$ ), comparing it to different common practice galaxy sizes, such as  $R_e$ . Then, we will shortly introduce the types of hydrodynamical simulations and the EAGLE and IllustrisTNG simulation projects, focusing on their different AGN feedback models.*

---

### 1.1 Background

The impact of supermassive black holes (SMBHs) on the evolution of their host galaxies is well known as feedback (AGN). For many years the effect of the AGN feedback processes has been studied as the principal method of star formation regulation (Clavijo-Bohórquez et al. 2023; Rosito, M. S. et al. 2021) and quenching of early type galaxies (Nesvadba, N. P. H. et al. 2010; Wylezalek & Zakamska 2016) both in empirical observations and numerical  $\Lambda$ CDM models. The outflows emanating from these AGNs can also wipe the gas away from the central region of their host galaxy (McNamara & Nulsen 2012) as these outflows are the consequence of the accretion of gas, transforming its kinetic energy to thermal energy and sometimes this energy is even larger than the binding energy of the host potential energy.

AGN feedback can then be related to the evolution of galaxy sizes, as this process of expelling gas via mechanical feedback has an important impact on the size of the galaxies and their evolution. This is highly dependent on the definition of galaxy size.

The most used galaxy size definition both in empirical observation and hydrodynamical cosmological simulations is the effective radius ( $R_e$ ), which is the radial distance that encloses half the total light emitted by the galaxy, variable with the wavelength of observation (Kennedy et al. 2015). This definition is very useful to characterize a galaxy by its luminosity profile, but not for many other purposes. Along with this approach, a very intuitive galaxy size definition was proposed by Trujillo et al. (2020), which is defined to enclose all the mass inside the radius where the stellar density is equal to  $1 M_{\odot}/\text{pc}^2$  (named  $R_1$ ).

This definition is also very interesting as it does not depend on the luminosity profile of



each galaxy but on its physical extent, which presents a much more accurate size indicator when identifying an object visibly. In other words, galaxies with similar extent get similar  $R_1$  values, whereas similar values of  $R_e$  can mean very different extents, depending on the luminosity radial profile, as galaxies with a very luminous bulge or a denser central region will have a smaller effective radius regardless of their actual physical extent.

The results of the mass-size relation regarding  $R_1$  were compared to the EAGLE (Schaye et al. 2014) simulation data, finding very similar results overall. Then, the plan was to compare these results with the IllustrisTNG simulations, which have a much more aggressive AGN feedback implementation. This will help understand the impact of AGN feedback in galaxy size, regarding different size definitions.

## 1.2 Cosmological simulations

Our current understanding of the Universe is based on a model that acknowledges the significant influence of Dark Energy and Dark Matter. However, since these components cannot be directly observed, investigating their properties needs an alternative approach and interpretation. In this context, cosmological simulations emerge as an immensely powerful tool, enabling us to gain insights into the genuine formation and evolution of dark matter halos and galaxies. These simulations encompass various approaches and span different scales of length. The initial and simplest simulations only focus on dark matter and its behavior, yet they can become considerably more intricate. By incorporating gas, stars, and black holes, along with their respective physical models, we can develop what is known as hydrodynamical cosmological simulations.

Hydrodynamical simulations use dark matter, dark energy, and ordinary matter to study the evolution of the first initial overdensities present after the Big Bang happened (Planck Collaboration 2016) into the well-known Cosmic Web (Libeskind et al. 2017), which is a network of dark matter and ordinary matter organized in different halos and subhalos that interact with each other gravitationally. These simulated structures aim to have the highest level of detail, so they can be compared with real observed galaxies to test our current ideas on the formation and evolution of galaxies. By comparing and improving our simulation models we can help understand the real secrets that lie under the evolution of galaxies and the universe itself.

## 1.3 IllustrisTNG and EAGLE simulations

The IllustrisTNG simulations are performed based on solving hydrodynamical Euler equations with the moving-mesh technique used in AREPO (Springel 2010), which is superior to smooth particle hydrodynamics (SPH) techniques. The SUBFIND algorithm (Dolag et al. 2009) is the one used to find each halo and subhalo present in the simulation, which helps creating the friends-of-friends (FoF) files addressed with more detail in chapter 2.

The EAGLE simulations are performed using GADGET-3 tree SPH code, based on GADGET-2 (Springel 2005). The EAGLE simulation uses the same SUBFIND algorithm as IllustrisTNG as a method of characterizing FoF files and detecting halos and subhalos. In this case, different

types of simulation volumes are present, but only three box sizes are present with 25, 50, and 100 Mpc. The differential factor of this model is the cautious post-processing calibration of the different physical parameters, explained by [Crain et al. \(2015\)](#).

It is also important to mention that both IllustrisTNG and EAGLE simulation projects use the same initial mass function (IMF) presented by [Chabrier \(2003\)](#).

## AGN *feedback* models

As we know, AGN feedback can present two modes, each one corresponding to a different accretion rate. There is a low accretion rate mode feedback, also known as kinetic or mechanical feedback, which is commonly associated with a spherical accretion of hot gas present in the surrounding ISM, known as 'radio' mode. The high accretion rate feedback is associated with an efficient disc shaped accretion, where radiation dominates as a result of photoionization of the gas in the ISM, this last 'quasar' mode is characterized by short powerful outflows that unfuel de AGN quickly followed by a rapid accretion of mass that allows the next radiation outflow.

### IllustrisTNG

These two AGN modes are present in IllustrisTNG AGN feedback model. The model first has a seeding mechanism that creates a BH particle in the halos that do not have one and their halo mass is greater than a certain threshold ( $5 \times 10^{10} M_{\odot} h^{-1}$ ). This BH particle is created with a mass of  $8 \times 10^5 M_{\odot} h^{-1}$  and kept near the center of potential of the halo using an ad hoc centering prescription, which consists in checking for the distance between the BH particle and the center of potential of the halo every given short period of time and shifting the BH particle to that position.

To determine which accretion mode is applied, the ratio between the black hole accretion rate and the Eddington accretion rate is used ([Begelman 2014](#)). The accretion rate of the BH is defined as the minimum accretion rate between the Bondi accretion rate and the Eddington accretion rate presented in equation 1.2. In order to have a high accretion rate feedback, the Black Hole accretion rate must exceed a fraction  $\chi$  of the Eddington accretion rate,

$$f_{\text{Edd}} \equiv \frac{\dot{M}_{\text{BH}}}{\dot{M}_{\text{Edd}}} \geq \chi, \quad \dot{M}_{\text{BH}} = \min(\dot{M}_{\text{Bondi}}, \dot{M}_{\text{Edd}}), \quad \chi = \min \left[ 0.002 \left( \frac{M_{\text{BH}}}{10^8 M_{\odot}} \right)^2, 0.1 \right] \quad (1.1)$$

where:

$$\dot{M}_{\text{Edd}} = \frac{4\pi G M_{\text{BH}} m_{\text{p}}}{\epsilon_{\text{r}} \sigma_{\text{T}} c}, \quad \dot{M}_{\text{Bondi}} = \frac{4\pi G^2 M_{\text{BH}}^2 m_{\text{p}}}{c_{\text{s}}^3} \quad (1.2)$$

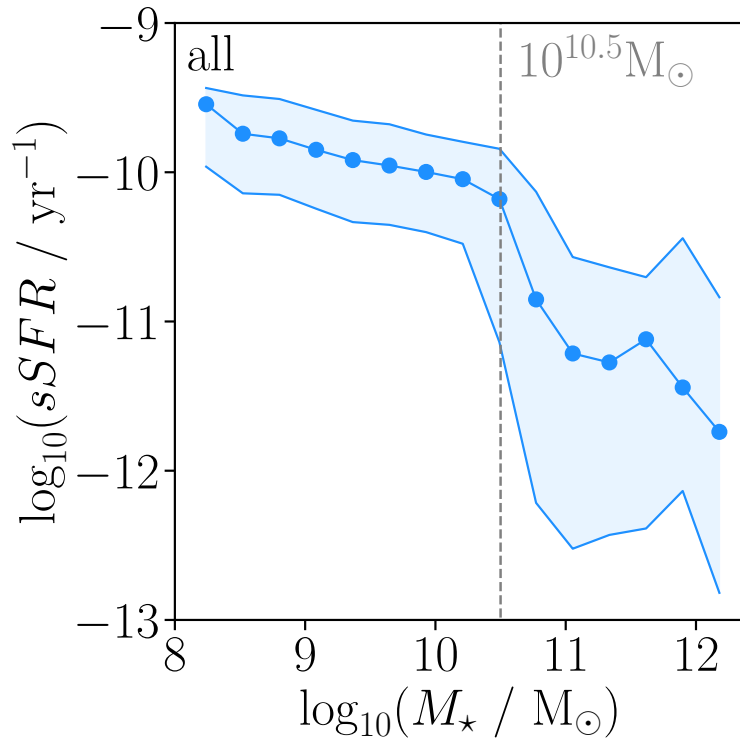
This ratio is scaled to the BH mass, which facilitates the transition of high-mass BH to the kinetic AGN mode. Also, a minimum threshold of  $\chi = 0.1$  is set so even the most massive BH can reach the high accretion state.

Then, for both feedback models, the energy is coupled to the cells in a small region around the BH, the recipient particles do not change other physical interactions, they adopt the following energy:

$$\dot{E}_{\text{high/low}} = \epsilon_{\text{high/low}} \dot{M}_{\text{BH}} c^2 \quad (1.3)$$

Where  $\epsilon$  is the coupling efficiency of the surrounding gas, which is set to  $\epsilon_{\text{high}}=0.02$  for quasar mode and  $\epsilon_{\text{low}}=0.2$  for radio mode. In low accretion mode, the energy is imparted as a momentum boost to the gas particles in discrete 'injection events' that occur once the BH has exceeded a certain energy accumulation threshold. For high accretion mode, the energy is continuously injected as thermal energy within the gas cells, weighted by a softening kernel. This is why this feedback mode is so inefficient. Cooling is very efficient in radiating away thermal energy. This is well described in [Dalla Vecchia & Schaye \(2012\)](#).

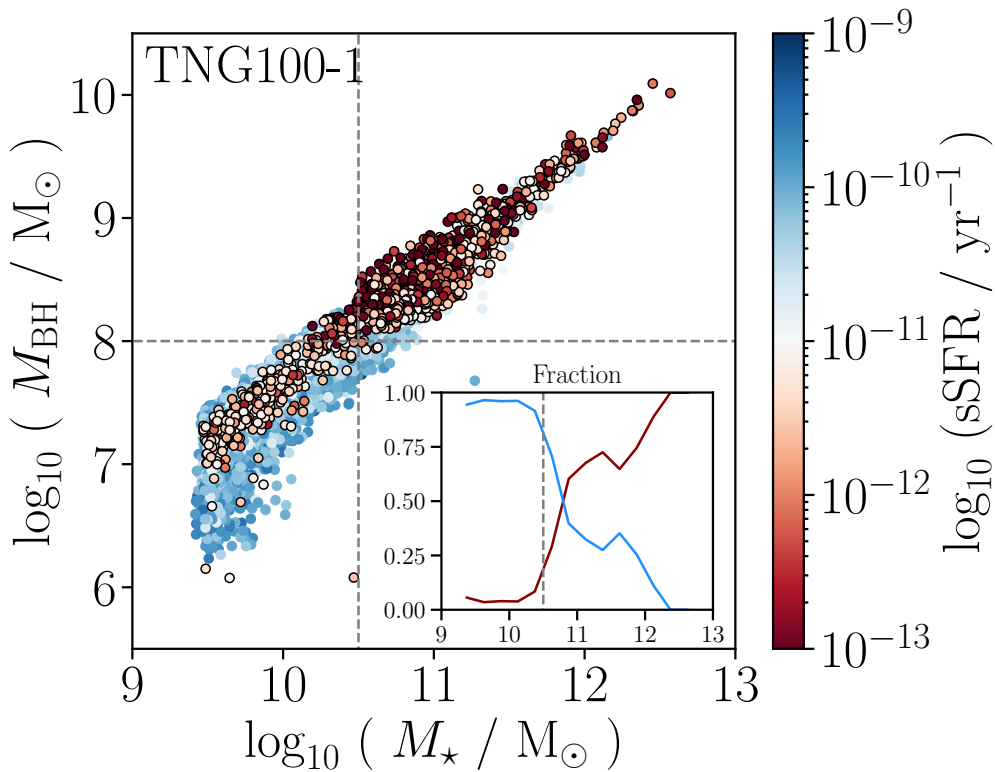
It is important to mention that in IllustrisTNG simulations there is a change in AGN feedback mode created ad hoc, directly related to the BH mass scaling mentioned above, as a BH mass of  $10^8 M_{\odot}$  corresponds to a halo stellar mass of  $10^{10.5} M_{\odot}$  ([Kormendy & Ho 2013](#)), where the star formation rate is heavily quenched in the IllustrisTNG simulations ([Zinger et al. 2020](#)). In [Figure 1.1](#) we present this sudden transition to the kinetic feedback. Data is loaded from IllustrisTNG simulations using TNG100 and TNG50 volumes combined, the use of data is addressed with more detail in [chapter 2](#).



**Fig. 1.1.** Relation of specific star formation rate (sSFR) to the stellar mass of the subhalo both axis in logarithmic scale. Each point presents the mean value of sSFR for all subhalos within each mass bin, with a total of 16 bins, where the horizontal axis coordinate of each point is the mid-bin value. The filled area around the relation corresponds to the 16th and 84th percentile of each data point.

We can discern the strong quenching of star formation in galaxies with a total stellar mass above  $10^{10.5}M_{\odot}$ , as predicted. Also, at  $z=0$ , galaxies with specific star formation rate (sSFR) lower than  $10^{-11} \text{ yr}^{-1}$  are considered quenched galaxies, with very low star formation activity (Katsianis et al. 2020). As we can see after this abrupt change in the AGN feedback model, all galaxies are quenched.

Furthermore, as we wanted to make sure the effect of the BH mass scaling of  $10^8M_{\odot}$  is responsible for the quenched star formation rate, in Figure 1.2 we present the BH mass to stellar mass relation color-coded by the specific star formation rate. We highlighted the quenched galaxies with sSFR lower than  $10^{-11}\text{yr}^{-1}$  and added a histogram of the fraction of quenched galaxies in red and star forming galaxies in blue. As shown in the plot, the fraction of quenched galaxies increases abruptly after the threshold mass studied above in Figure 1.1, corresponding to a black hole mass of  $10^8M_{\odot}$ . Also, the fraction of star forming galaxies drops to zero for masses above  $10^{12.4}M_{\odot}$  approximately.



**Fig. 1.2.** Relation of Black Hole Mass ( $M_{\text{BH}}$ ) to the stellar mass ( $M_{\star}$ ) of the subhalo both axis in logarithmic scale color-coded by specific star formation rate (sSFR). Quenched galaxies (with sSFR lower than  $10^{-11}\text{yr}^{-1}$ ) are highlighted with black borders and colored in red, while star forming galaxies colored in blue. Dashed grey lines define black hole masses of  $10^8M_{\odot}$  and stellar masses of  $10^{10.5}M_{\odot}$ . The inset plot presents the fraction of each of these galaxy types (star forming and quenched) against subhalo stellar mass with units  $M_{\odot}$ .

## EAGLE

In the case of EAGLE simulations, there is only one AGN feedback mode. The first step of the model is similar to the one of IllustrisTNG BH seeding, creating a BH particle in the place

of the densest particle in each subhalo more massive than  $10^{10}M_{\odot}h^{-1}$ . This particle is transformed into a BH collisionless particle with an initial mass of  $10^5M_{\odot}h^{-1}$ , which is a factor 8 smaller than the initial BH mass in IllustrisTNG. Then, once the BH mass exceeds the initial particle mass, the accretion processes start. There is also an algorithm that prevents the BH from leaving the center of potential of their host galaxy. There is also a merging mechanism between BHs that are closer than the BH kernel size or three gravitational softening lengths.

The physics of the mass accretion are the same as in IllustrisTNG, the accretion rate follows the minimum between the Eddington rate and the Bondi rate (presented in Equation 1.2). The growth of BH mass is:

$$\dot{m}_{\text{BH}} = (1 - \epsilon_r)\dot{m}_{\text{accr}} \quad (1.4)$$

With a radiative efficiency of  $\epsilon_r = 0.1$ .

In this case, only one feedback mode is adopted, where energy is injected from the BH thermally and stochastically, with a rate of:

$$\dot{E}_{\text{BH}} = \epsilon_f \epsilon_r \dot{m}_{\text{accr}} c^2 \quad (1.5)$$

where  $\epsilon_f$  is the fraction of energy coupling with the ISM, calibrated by observations (Crain et al. 2015).

Even though the model only accounts for one type of AGN feedback, it has proven to mimic the properties of low accretion rate "radio" type feedback and high accretion rate "quasar" feedback when the BH accretion rate is whether a very small (radio) or large (quasar) fraction of the Eddington rate.

Each BH particle maintains an energy reserve (just as in IllustrisTNG),  $E_{\text{BH}}$ , that is updated each simulation time step (using expression 1.5). Once this energy is at least capable of heating a fluid element of mass  $m_g$  it becomes possible that the heating process is performed, following a probability for each neighbor particle that is inversely proportional to the temperature increment of that element ( $\Delta T_{\text{AGN}}$ ), the number of neighbors ( $N_{\text{ngb}}$ ) and the main mass of these ( $\langle m_g \rangle$ ), following the relation presented in equation 1.6.

$$P = \frac{E_{\text{BH}}}{\Delta \epsilon_{\text{AGN}} N_{\text{ngb}} \langle m_g \rangle} \quad (1.6)$$

Where  $\Delta \epsilon_{\text{AGN}}$  is the change in internal energy per unit mass corresponding to the temperature increment. The time step at which  $E_{\text{BH}}$  is checked and updated is calibrated to aim for probabilities of  $P < 0.3$  and the variations of temperature  $\Delta T_{\text{AGN}}$  are in the range  $10^{8.5-9}\text{K}$ , depending on the simulation volume. A higher temperature variation value produces more energetic feedback but makes the process more intermittent.

Generally, the IllustrisTNG feedback model has proven to be much more aggressive than other hydrodynamical simulation feedback models. When compared to EAGLE simulations, the two models perform very differently, this is why IllustrisTNG was chosen for this project, as the purpose was to analyze the effect of different AGN feedback in galaxy size and compare both mass-size relations so we could better understand the behavior of two different feedback models.

# Chapter 2

## Data

*In this chapter we describe the different datasets available from IllustrisTNG. First, we introduce the different file types and their purpose, then we give an explanation of different useful parameters for each file type as well as the sorting and ordering of data in each file. After that, we introduce some very useful alternative files we have used in the analysis. Finally, the parameters for each simulation and volume are presented, as well as some indications of the cosmology used by the IllustrisTNG team.*

---

In general, data from astrophysical simulations is given in snapshots, which are in binary format (*.hdf5*) to optimize the size of the files even though they are holding extremely large amounts of data. These snapshots are images of the simulated Universe volume at a given time ( $z$ ) that contain all the data, they are distributed in different files as they will be too huge otherwise. Snapshots are basically divided into two types of files, the snapshots themselves, which contain all the particle data available to analyze in lists that are the size of the number of particles, and the friends-of-friends files, or group files, depending on the simulation this last name may differ. These last contain the information about the groups and subhalos present in the simulation like the positions of the particles that correspond to each subhalo, the most massive subhalo for each group, or more generic parameters like the number of groups and subhalos in each file and in total.

It is important to mention that these friends-of-friends files are built using a combination of L`HaloTree` (Springel et al. 2005) and SubLink (Rodriguez-Gomez et al. 2015) algorithms. The algorithms used by each simulation project are addressed in chapter 1. This halos and subhalos information is written in the friends-of-friends files, which are built parallel to the snapshot files, meaning there is one friends-of-friends file for each snapshot file in the simulation, which is made to order and input data from that particular snapshot.

In IllustrisTNG, each simulation run is divided into 100 snapshots that include all particles/cells in the volume and are classified following its redshift. Of these 100 snapshots, there are 80 of them that are called 'mini' and only present a subset of particle fields. Focusing on the other 20 snapshots that contain all the relevant information we need, we will only use the snapshot at redshift 0 (scale factor 1). Worth noting that this is enough for our objective, but for many other projects the possibility of exploring different time frames gives a very powerful tool that helps understand the evolution of galaxies and their properties.

## 2.1 File types

### 2.1.1 Snapshots and stellar data

The snapshot files are divided into two sections, the first one contains general information about the code runs, which is stored in the "Header", "Parameters" and "Configuration", and holds constants and parameters like the Hubble parameter (`HubbleParam`) or the size of the simulation box (`BoxSize`). The second section holds all the data for the `PartTypeX` structures, these are:

- `PartType0` - Gas
- `PartType1` - Dark matter
- `PartType4` - Stars and wind particles
- `PartType5` - Black holes

Particle 2 type is empty as the code was built this way, allowing it to be filled in later versions and type 4 are wind particles, which are artificial particles used to perform SN feedback models that consist of particles expelled by a supernova and interacting with other particles in their way.

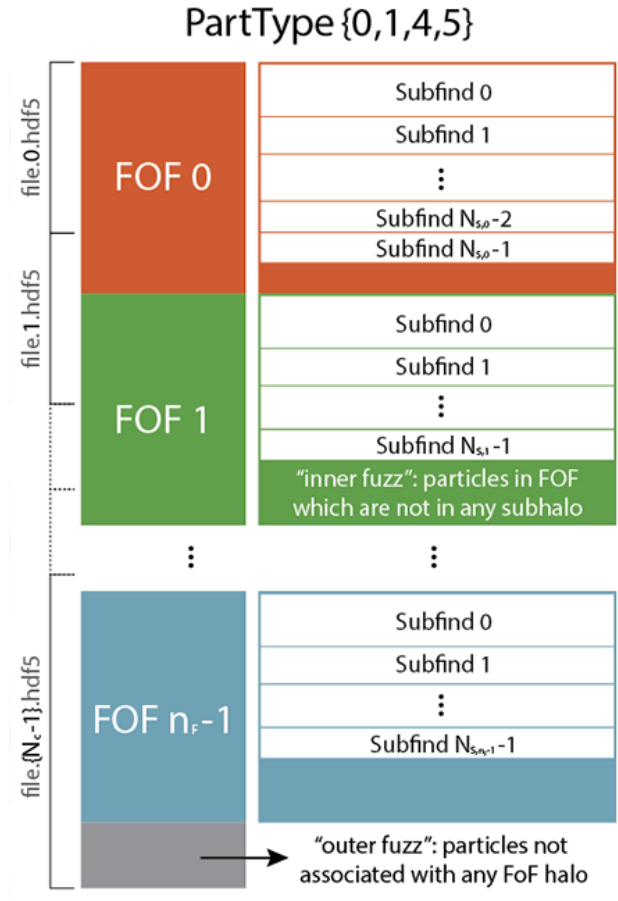
As this project is focused only on the stellar component of the subhalos, we will be using only the `PartType4` subfile for each snapshot.

Some of the most important elements in snapshots that we will be using in the analysis are:

- **Header**
  - `BoxSize`: Size of the simulation box, used to center subhalo coordinates and avoid periodicity.
  - `HubbleParam`: Hubble Parameter value, used for unit scaling.
  - `NumFilesPerSnapshot`: Total number of files per snapshot, used to loop over all files to input data
  - `NumPart_ThisFile`: Total number of particles in this file per type, used to add that many particles to the particle magnitude array (mass, coordinates, velocities, etc).
  - `NumPart_Total`: Total number of particles, used to initialize and allocate each particle array to the the total number of particles of that type.
- **PartType4**
  - `Coordinates`: Coordinates of each particle cell.
  - `GFM_StellarFormationTime`: Time of formation, if  $t \leq 0$  then they are wind particles. Used to separate stellar particles from wind particles.
  - `GFM_StellarPhotometrics`: Stellar magnitude in 8 bands (U, B, V, K, g, r, i, z). In our case, we are using SDSS g band to calculate  $R_{29}$ . These "synthetic" magnitudes are calculated by emulating the g-band filter in SDSS from [Stoughton et al. \(2002\)](#).
  - `Masses`: Mass of each particle cell.

### 2.1.2 Friends-of-friends files and subhalo data

In IllustrisTNG, unlike in other simulation projects, every particle array that contains the information is sorted by its group number and subhalo number, this means the most massive or central subhalos for each group will be placed first in the array, followed by the less massive subhalos and then by the particles (in our case stellar and wind particles) that aren't bound to a subhalo, this structure is shown in Figure 2.1.<sup>1</sup>



**Fig. 2.1.** Organization of snapshot files in Illustris TNG simulations, from most massive to less massive subhalos in each group, having free particles (not bound to any subhalo) at the end. Source: IllustrisTNG website (<https://www.tng-project.org/data/docs/background/>)

The information about the first elements for each subhalo and group is in the friends-of-friends files, which we will need in order to select and crop the PartTypeX array into each subhalo array, so we can make calculations only on a given subhalo data. These friends-of-friends files are built by the sub-find algorithms named in chapter 1.

These files are shaped similarly to the snapshot files, but they instead have the information for halos and subhalos instead of PartTypeX arrays. This means they have initial "Header", "Config" and "Parameters" files and then two data groups, "Group" and "Subhalo".

<sup>1</sup>All this information can also be found at the project website: <https://www.tng-project.org/data/docs/specifications/#sec2a>



- **Header**

- **Ngroups\_ThisFile**: Number of groups in the current file, used to add that number of elements to group arrays.

- **Ngroups\_Total**: Total number of groups, used to initialize any allocate each group array to the total number of groups.

- **Nsubgroups\_ThisFile**: Number of subgroups in this file, used to add that number of elements to group arrays.

- **Nsubgroups\_Total**: Total number of subgroups, used to initialize any allocate each subhalo array to the total number of subhalos.

- **Group**

- **GroupLenType**: Number of particles of each type per group, used to copy group information to particle arrays, such as group number.

- **GroupFirstSub**: First subhalo of each group, and also de central and most massive one, used to locate central subhalos and plot the stellar mass of each central subhalo vs  $M_{200}$  of the group.

- **Group\_M\_Crit200**: Total mass of the group enclosed in a sphere of  $\Delta = 200$ , used in post-processing.

- **Group\_R\_Crit200**: Total radius of the group that encloses a sphere of  $\Delta = 200$ , used in post-processing.

- **Subhalo**

- **SubhaloLenType**: Number of particles of each type per subhalo, used to copy subhalo information to particle arrays, such as Subhalo Number.

- **SubhaloGrNr**: Group number of each subhalo, used to assign group values to subhalo arrays.

- **SubhaloHalfMassRadType**: Radius that holds half the mass from each particle type for each subhalo, used to calculate  $R_{50}$ .

- **SubhaloMassType**: Total mass of the subhalo for a given particle type, used for total stellar mass.

- **SubhaloPos**: Position of the particle with minimum potential energy in the subhalo, used to center subhalo coordinates.

- **SubhaloSFR**: Sum of all SFR of gas cells in the subhalo, used in post-processing.

- **SubhaloSpin**: Total spin per axis, used to rotate galaxy to face-on perspective.

### 2.1.3 The offsets file

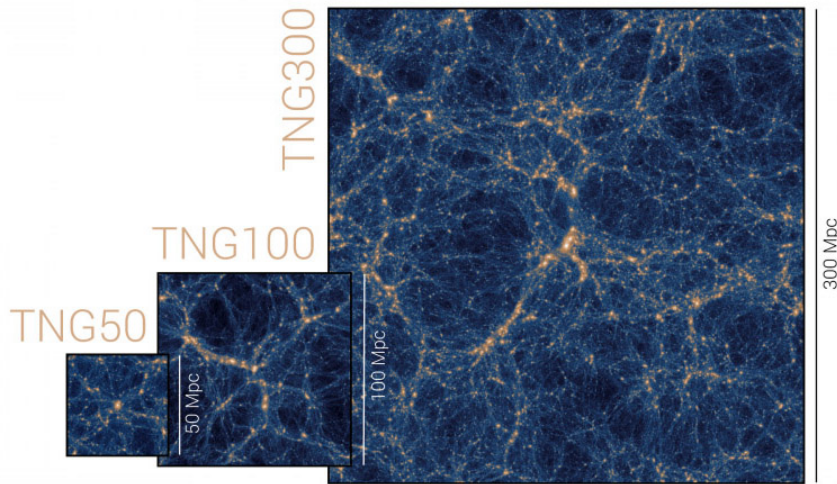
The last file type we will be using is the `offsets` file. This file is made to ease the process of importing data. It contains the first elements for each file, group, subhalo, and more so navigating through the snapshot and friends-of-friends files is an easier task. The elements we will be using are the following:

- **Subhalo/SnapByType**: Gives the location in the particle array of the first particle for each subhalo, used for loading data into subhalo arrays.

- `Group/SnapByType`: Gives the location in the particle array of the first particle for each group, used for loading data into group arrays.

## 2.2 Simulation volumes and resolutions

IllustrisTNG holds different types of simulations classified in volume and resolution. There are TNG50, TNG100, and TNG300 volumes, which correspond to box sizes of 50, 100, and 300 Mpc each. For every volume, we have 3 different resolutions, which are numbered from 1 (highest resolution) to 3 (lowest resolution). In the case of TNG50, there is a fourth simulation with even less resolution. This gives a total of 13 different simulations.



**Fig. 2.2.** The three different IllustrisTNG simulation volumes, TNG50, TNG100, and TNG300. From small box size and high resolution, so better details for smaller systems can be examined to big box size and low resolution so clusters and groups of galaxies can be studied. Source: IllustrisTNG website (<https://www.tng-project.org/data/docs/background/>)

Also, for each of the 13 simulations, there's a gravitation-only alternative simulation that emulates the size and resolution of the hydrodynamical simulation but only using DM. These have the suffix *-Dark*.

In our case, we will use TNG50 and TNG100. We will be using TNG50-3, which is the second lowest resolution one, to perform different tests on the code but the real post-processing analysis will be performed on TNG100-1 and TNG50-1. The need to use a combination of TNG50 and TNG100 will be discussed in detail in chapter 3. Briefly explained, we can achieve more resolution in the lower mass range of subhalos if we use TNG50, so it is handy to combine different resolutions if one wants to have information from a wider range of subhalo masses and sizes.

The exact resolution and length of the co-moving simulation volume along with other information about each simulation can be found in Table 2.1.

**Table 2.1.** Length of the simulation volume cube ( $L_{\text{box}}$ ), number of Dark Matter total particles ( $N_{\text{DM}}$ ), the minimum DM mass of a resolution cell ( $m_{\text{DM}}$ ), and the minimum gas mass of a resolution cell ( $m_{\text{gas}}$ ) for each simulation used in the project.

	$L_{\text{box}}$ [Mpc]	$M_{\text{DM}}$	$m_{\text{DM}}$ [ $M_{\odot}$ ]	$m_{\text{gas}}$ [ $M_{\odot}$ ]
<b>TNG100-1</b>	110.7	1820 <sup>3</sup>	$7.5 \times 10^6$	$1.4 \times 10^6$
<b>TNG50-1</b>	51.7	2160 <sup>3</sup>	$4.5 \times 10^5$	$8.5 \times 10^4$
<b>TNG50-3</b>	51.7	540 <sup>3</sup>	$2.9 \times 10^7$	$5.4 \times 10^6$
<b>TNG50-4</b>	51.7	270 <sup>3</sup>	$2.3 \times 10^8$	$4.3 \times 10^7$

## 2.3 Cosmology and units

Finally, it is important to know that all the simulations used are at redshift  $\sim 0$ , with a Hubble parameter of  $h = 0.6774$ , and cosmological parameters for matter and dark energy  $\Omega_0 = 0.31$ ,  $\Omega_{\Lambda} = 0.69$ , (Pillepich et al. 2017; Planck Collaboration 2016).

The units in simulations are commonly dependent on the Hubble parameter, as this makes lengths and masses redshift dependent. In this framework, IllustrisTNG units are kpc/h for lengths and  $10^{10}M_{\odot}/h$  for masses and km/s for velocities. Other variables used in the project and its units are the star formation rate ( $M_{\odot}/\text{yr}$ ), and stellar magnitudes (mag). The comoving quantities can be converted to physical quantities using the scale factor  $a$ . Note that for our case, at redshift  $z=0$ , the scale factor is  $a=1$ , so the numerical values of comoving quantities are equal to their physical equivalents.

# Chapter 3

## Methodology

*This chapter serves as an in-depth description of the methodology used in the project. First, we present the tools used, both hardware and software and then we start to dive into the code. For this purpose, we introduce code maps that facilitate the task of understanding and then, describe in detail the different scripts used and their functions. Then, we explain in depth the process of obtaining the size of galaxies, giving details of the main processes and routines present in the main pipeline of the code. Finally, we address the file organization carried out to store the information in an orderly way.*

---

### 3.1 Software and hardware tools

#### Software


All the code has been written using Python3, as it is the standard high-level programming language used to analyze simulations. For the input section of the code, we used the library `h5py`, which makes the work easier when working on loading data from `.hdf5` binary files. For the analysis and main pipeline we also used: `numpy`, `matplotlib`, `scipy`, and `datetime`.

All the code is version-controlled using `git`<sup>1</sup>. This is very helpful in case one loses some files or makes some changes that wants to revert. It is also very useful when working remotely, as one can edit scripts locally and push changes to `git`, then in the remote computer edits are pulled and the different scripts executed.

#### Hardware

The code has run using different IAC machines. Tests with the lightest datasets were run in `virus`. When the largest datasets were used for production analysis, the code was run on `diva`, using the parallel jobs queue system via the login node `deimos`.

---

<sup>1</sup>  Project repository: [https://github.com/andygarciaserra/tfm-galaxy\\_sizes/tree/main](https://github.com/andygarciaserra/tfm-galaxy_sizes/tree/main)

## 3.2 The code

### 3.2.1 Original EAGLE analysis code

The initial analysis code, written by Claudio Dalla-Vecchia, was made to input data from EAGLE simulations and compute the galaxy radius at stellar density  $1 M_{\odot}/pc^2$  ( $R_1$ ), as well as the radius at a surface brightness of  $29 \text{ mag}/pc^2$  ( $R_{29}$ ). This code was split into two different pipelines that imported the data and computed the two different radii. The pipelines themselves had some routines that were able to make the post-processing and analysis of the data. The code also had a `config.py` file that served as a quick parameter setter for things like the data directory, the mass, density, and surface brightness thresholds, or the different data files.

The only thing we have reused from the original code are the two routines that compute  $R_1$  and  $R_{29}$ . These are not in their original form, as the code has shifted into an object-based code and the input, output, and use of the variables have had to be adapted.

The following sections of this chapter will be used to describe the new code and its features. In the explanation of these sections, the computing radius routines from the original code will be addressed, but keep in mind that this is only to describe how the code works, the coding of these routines has not been performed by me.

### 3.2.2 The new IllustrisTNG input and analysis code

#### Scripts and general code map

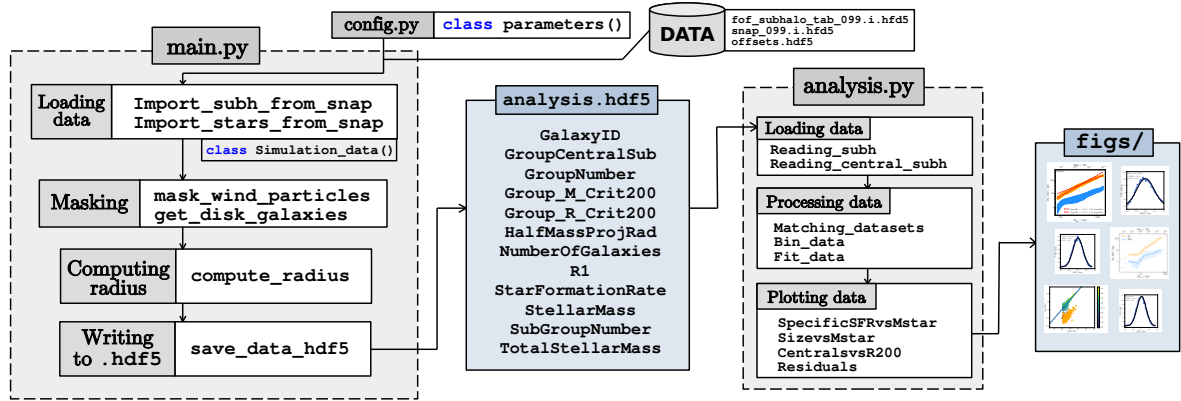
As explained above, the new code only keeps the `compute_radius` function developed for EAGLE simulations. We will explain later on the behind-the-scenes of the  $R_1$  and  $R_{29}$  calculation.

In Figure 3.1 we present the code map. Firstly, data is input into the pipeline using the parameter class present in the config file. This file holds different important parameters and paths. Then, the data is masked using the thresholds present in the config file, also wind particles are masked out. Then, the `compute radius` routine is called with the necessary input. After calculations are made, the results are written in an analysis file in binary format. This analysis binary file is loaded from the `analysis.py` script that loads the data and processes it to finally plot the results. In the following sections, we will explain the different scripts and routines in detail. The order will be following the process shown in the code map.

#### Config script

The goal of having this file is to have quick access to certain variables and avoid having to change parameters and variables in the pipeline. The original code had a simpler version `config.py` that held the different paths (for data and analysis written files) and the thresholds associated with mass and magnitude.

This script was improved by making it object-based. We defined a parameter class that initializes depending on the volume of data one wants to load. This data loading depends on a string variable. This makes it so convenient because when calling from the `main.py`



**Fig. 3.1.** Code map of the different scripts used and files written. Scripts in grey color and dashed line. Written files are in blue with solid lines. Data files are in grey or white with solid lines.

pipeline the parameter class is already initialized to the volume data, having all the paths, threshold variables, and simulation parameters within easy reach.

These are the most important elements of the `config.py` script:

- `DATADIR` and `snapNAME`: The result of initializing parameters with different input names. They provide the path to data, very useful to load it, and the name of the simulation volume, used in figure comments and written file names.
- `snap_prefix` and `fof_prefix`: Useful when looping over the different files to load or search for certain data as files only differ in one number.
- `minMass`, `maxMass` and `minStellarParts`: Mass and total particle number thresholds for subhalo masking.
- `densThreshold` and `magThreshold`: Density and surface brightness thresholds for computing subhalo radius.
- `FIGDIR` and `analysisDIR`: Paths to save or load the different figures and binary analysis files.
- `UnitMass` and `UnitLength`: Units to transform everything to  $M_{\odot}$  and kpc.

certain sections that do not need to load each snapshot or friends-of-friends file. This is done for each piece of data that is not bound to a certain snapshot. Some of these parameters are `HubbleParam`, `BoxSize`, and the different numbers of total halos, subhalos, and the total number of stars.

## Main pipeline and considerations

### Simulation data class

As the config file, the main pipeline of the code is also modular. It works with the class object `Simulation_data`, which holds the arrays of data needed for computing the radius. These

arrays are initialized when starting the code, which uses the parameter file also initialized by `config.py` to allocate the different length arrays. By allocating the memory used we can make sure the code will not crash in case there is a lack of memory.

Not only these arrays are a part of the class object, but also the routines. The code calls these routines that update the values from the initialized arrays for loading the different stellar or subhalo data. This is far more optimal and convenient with close to zero difference in computing time.

This will be the first step, the initialization of the data arrays that form the `Simulation_data` class object. Then, the data gets loaded from the different binary files using internal class routines.

### Loading of data

Once the class arrays are initialized, we can jump to the loading process. This implementation was the longest to code, as many different elements and array positions had to be taken into account. Also, the difficult part of this routine is that snapshots are divided into different files and the code must be made to load looping for each snapshot file. In Figure 3.2 we are presenting the data load steps and functioning.

For the stellar data, we initialize a position counter (`s`) at zero so we can load data from the first element into each stellar array. We use the total number of stellar particles for each file and load that number of array elements, then starting at the last index loaded we loop over each file doing the same, this is done first for all snapshots and friends-of-friends files. It is important to check whether or not files have stellar particles to avoid the code crashing

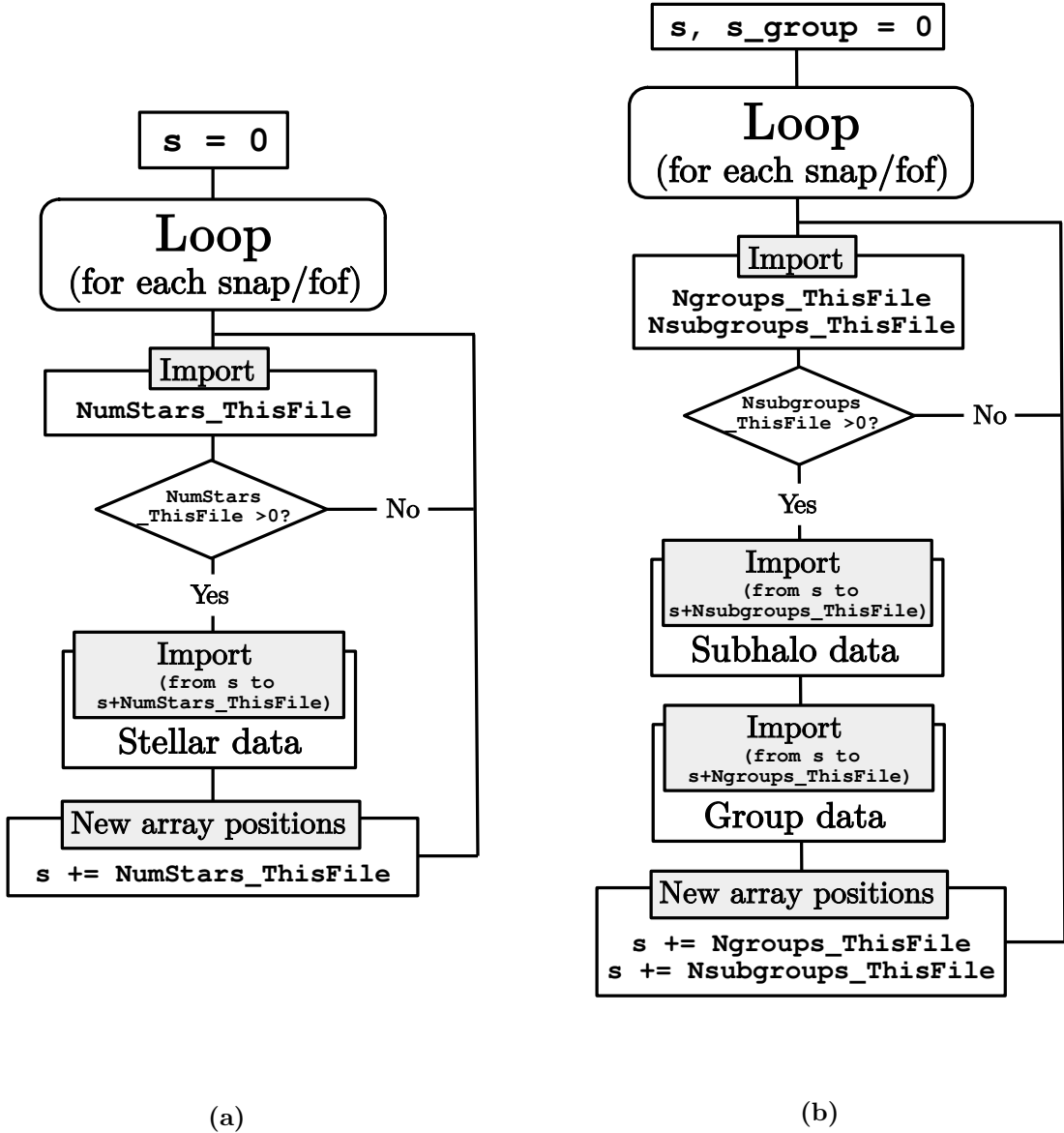
For subhalo data, the situation is close to being the same. We loop over each snapshot and friends-of-friends to load the data starting at the initialized indexes `s` and `s_group`, which belong to the subhalo data and the group data. In this case, as we have data from these two, we first load the total number of subhalos and subgroups for each file and load both the subhalo information and the group information separately, resetting the starting indexes at the end, as before.

As loading the data to new defined `numpy array` objects will double the memory used by the routine, we use a, `h5py` routine called `read_direct`, which fills the initially allocated array element by element directly without the need to create a bridge array.

### Masking

Once we load all data we need to apply a masking routine to the subhalos so the sample meets our requirements. In our case, we are using the threshold values explained above in the `config` file, which are limiting the mass range and a minimum of stellar particles per subhalo.

These thresholds were chosen considering the volume unit resolution for each simulation and, of course, considering a mass range that will include all galaxy types. We are considering galaxies from  $10^{8.5}$  to  $10^{14} M_{\odot}$  in stellar mass. We are aware that this upper threshold is bigger than a giant spiral or elliptical galaxy, which can reach about  $10^{13} M_{\odot}$ . The upper



**Fig. 3.2.** Data loading routines for the stellar particles data (a) and subhalo data (b).

mass limit is set to include every galaxy and the lower mass limit is set to limit the galaxy masses in case we want to avoid resolution limitations.

Also, the threshold for a minimum number of stellar particles per subhalo is set to 3000. This is so that the algorithm that computes the radius has a large enough sample to estimate stellar density at different radii. We have discerned that this lower threshold for total stellar particles tends to adjust the minimal mass present in our samples, as we know the stellar particle mass. This will be addressed in chapter 4, where we will further explain the lower cut in mass for each plot.

These two initial maskings are very straightforward. We just selected in the subhalo stellar mass array the positions where the thresholds were being satisfied and the same for



the subhalo stellar particles array, then, we matched both criteria to discern which of these array indexes were common to both standards and shortened every array to only the indexes that satisfied both. Keep in mind each of the loading and masking processes is updating the values of the `Simulation_data` class arrays, which are initialized at the start of the main pipeline.

Finally, the last mask is to remove wind particles from the sample. As they are artificially modeled to form from stellar particles, masking them is as easy as keeping only particles that have a formation time above zero, as wind particles are set to have negative to zero stellar formation time.

As this last masking is using much longer arrays (there are much more stellar particles than total subhalos) we were having excessive memory crashes when running the code, so we had to do some adjustments to this subsection of the masking process. To avoid this excess memory we defined temporary arrays that held only the stellar particles, then shortened the class arrays to the temporary arrays and deleted the temporary arrays. Doing this for each array gives some room for memory usage.

### Computing radius

Once our data is correctly loaded and masked we can compute the radius. It is important to mention that this routine is the only one which is external to the class definition, as the architecture of the previous version was not easily adaptive to the current script class definition. Remember this is the only section of the code that was not coded for this project. The code map for the main pipeline is presented in Figure 3.3.

The first step in the routine is to allocate the subhalo arrays that will hold the different data corresponding to the selected subhalos, after the masking of the data. This allocation creates for each data array an empty array that has the length of the number of subhalos, which is a parameter passed to the routine. All this data is stored as a numpy `recarray`, these are data structures that gather groups of arrays, which are very useful for navigating data and writing the result in a final binary file, as we need.

After allocating the data, the process of computing the radius starts, and for that, the code loops over subhalos. The first step of this loop is to load the subhalo stars. For this, the code uses the `Stars_SubhaloNr` array, which holds the subhalo number for each star. Knowing the number of the subhalo to use, the code extracts the positions of the stars within that subhalo.

Once stellar coordinates are loaded, these need to be preprocessed. The first step is centering them and avoiding periodic boundary conditions which is handled by subtracting the center of potential for each subhalo from each coordinate in three dimensions. Then the code checks whether the coordinates are further from the periodic volume, in which case they are re-scaled to be inside and in their real not periodic position.

The coordinates are then rotated so the galaxies that are not face-on become flat in the 2D view. This makes it easier to calculate the radial distance to the center of the galaxy. To perform this rotation the code uses the subhalo total spin from the simulation data. A normal rotation matrix is applied to the coordinates so that the spin vector is orthogonal to

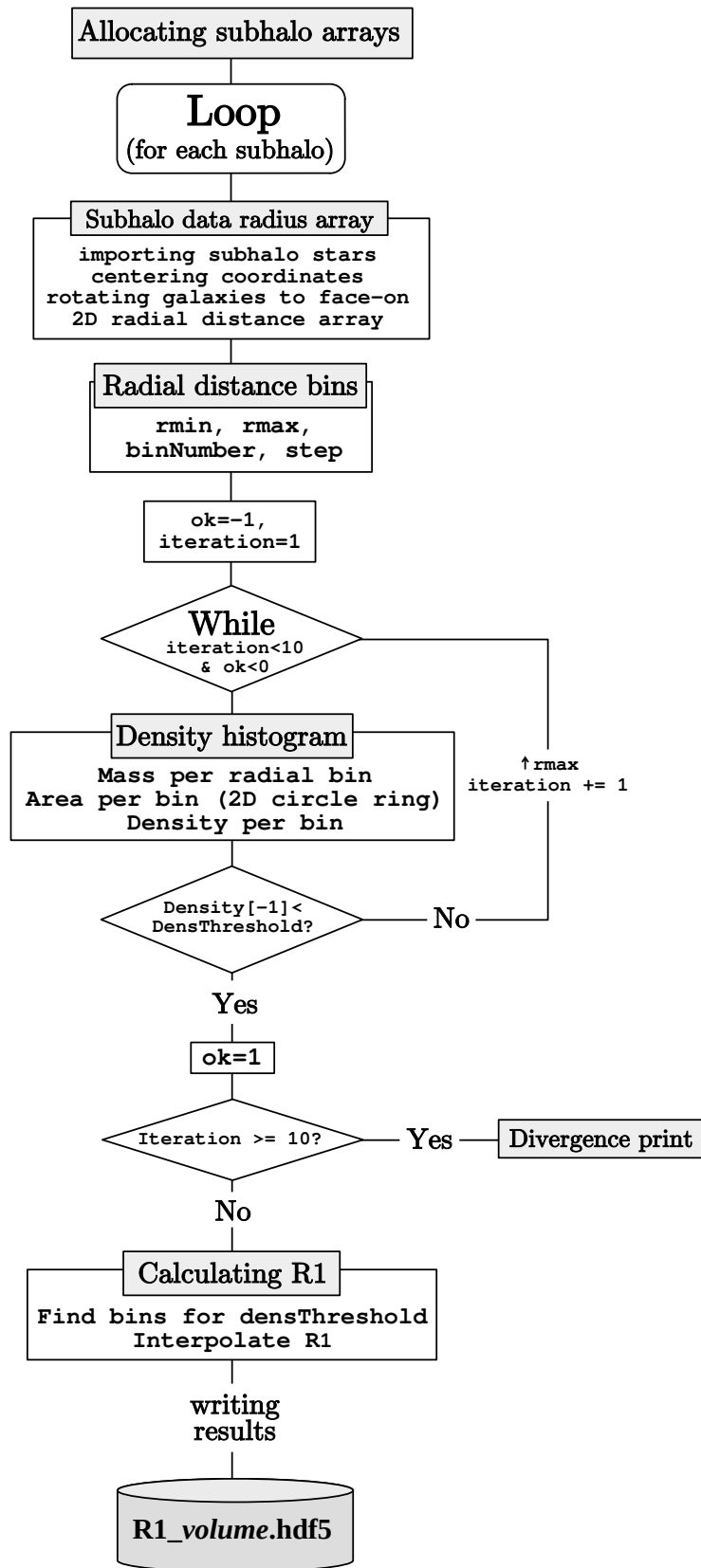


Fig. 3.3. Code map for the  $R_1$  computation routine.

the XY plane. Then radial distance is calculated as  $r = \sqrt{x^2 + y^2}$ .

Once the radius array for each particle is ready the code prepares the range of radius and the bins in which to perform a density profile of the subhalo. The minimum radius to start the calculation is the tenth part of the half-mass radius, given by the simulation data. The maximum radius is the minimum value between 10 times the half-mass radius and the maximum radial distance of all stellar particles within the subhalo. For the bin data, the code prepares a fixed number of bins, which is set to 50, and a step, which is set to 1.15. Also, the variables to control the iterations for the density profile convergence are set, which are `ok=-1` and `iteration=0`.

The code then starts a while loop (with the conditions of less than 10 iterations and no convergence) which consists in making a histogram of the number of stars weighted by mass for each radial bin in a logarithmic scale. Then, it calculates the area for each bin as the annular region of the bin with which the density of each bin gets computed. As a final check for each while loop the code checks whether or not the density of the last bin is less than the needed density threshold. If the density threshold is reached the density profile has converged, and the code then exits the loop by setting `ok=1`. If the convergence is not achieved, the code updates the number of iterations and iterates again but this time the radius range is multiplied by the step parameter, which makes the computing area bigger to reach for lower density. Once the code fails to converge in 10 iterations, the density profile of that subhalo is skipped and a convergence error message is printed on the screen.

After the density profile is computed for each subhalo, the code extracts the value of the radius at which the density equals the threshold by interpolating the radial values at the above and below edges. Finally, the values for the radius and parameters of the subhalo are written into the `recarray` structure.

### Writing data

Once the loop has looped over every subhalo, the data in the `recarray` is written to a binary analysis file whose filename starts with "R1\_" followed by the number of the volume of data, such as "TNG50-1" or "TNG100-1".

### R29 pipeline

It is important to mention that the pipeline regarding the computation of the radius at a surface brightness of 29 mag/pc<sup>2</sup> has the exact same structure and functioning, the only difference is that the histograms are weighted by magnitude in the g-band instead of by mass. This last script is very important because it computes the total stellar mass of a subhalo as the mass inside  $R_{29}$ .

## 3.3 File organization

The file system is split up into 3 folders, the one holding the different python scripts, `python-scripts/`, the one with the analysis files written from the pipeline and ready to

be analyzed, `analysis/`, and a third folder that contains the figures, `figs/`. The different resulting figures present in this last folder are the ones we will be addressing in the next chapter of the text.

# Chapter 4

## Results and discussion

*In this chapter we present the final results of the project along with some discussions. We compare this results with the ones obtained in EAGLE simulations previously in the literature as well as with other references. Firstly, we study the mass-size relation for different simulation volumes, comparing the different power law slopes found and discussing the impact of the use of different resolutions for Illustris and EAGLE. The scatters present in the different simulations are also addressed and discussed as the source of this scattering can be debatable.*

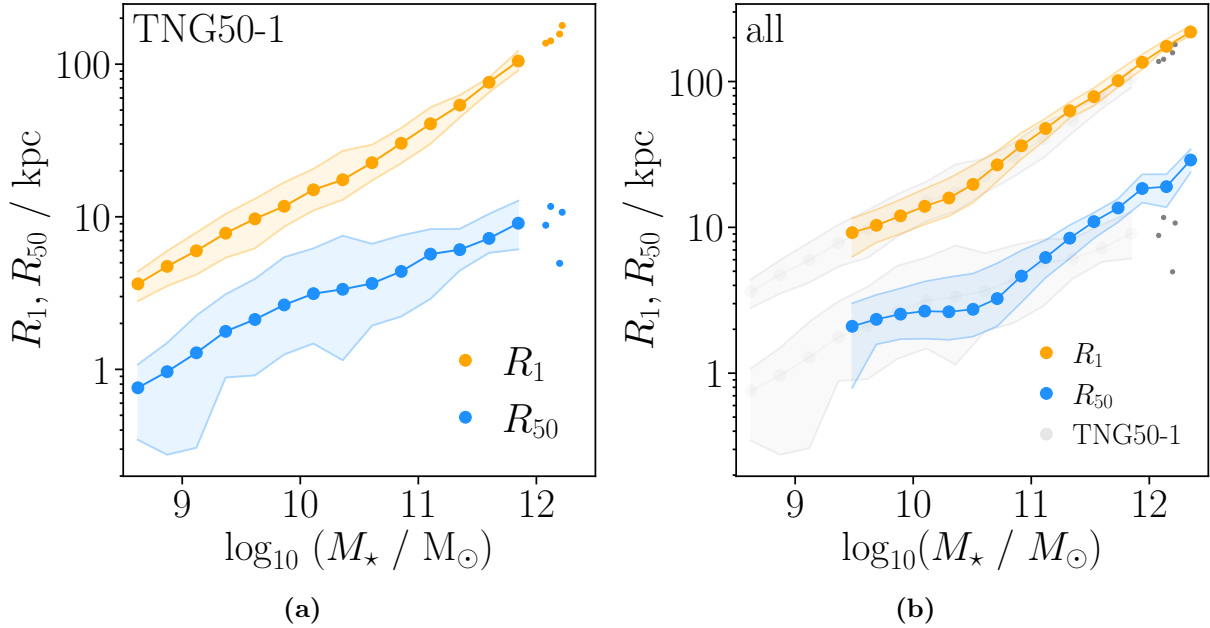
---

For the analysis of the results obtained by the pipeline, we focus on the mass-size relation of the subhalos. This analysis was performed using the `analysis.py` script, which is also modular and based on flags that hold different booleans to choose the different plotting options. These plots are made to compare them to the ones obtained by the original study performed in EAGLE simulations.

### 4.1 Mass-size relation

The calculations regarding  $R_1$  were performed for two different simulation volumes, TNG100-1 and TNG50-1. This gives a wider range of subhalo masses present in the analysis, as the TNG100-1 volume mass resolution is lower, meaning the subhalo masking threshold of a minimum number of stars sets this minimum mass above the actual masking threshold. In the case of TNG50-1, the volume mass resolution is higher enough that the minimum mass is the mass of the mass masking threshold. This difference in minimum mass can be seen in Figure 4.1b, where the distribution of galaxy sizes with respect to the stellar mass for the two simulation volumes are presented. After applying the lower and upper subhalo stellar mass thresholds to the data, the total number of subhalos for each simulation volume are 5678 for TNG50-1 and 12996 for TNG100-1.

As we can see in the case of TNG50-1, the minimum mass is somewhere around  $10^{8.5} M_{\odot}$ , which is the minimum mass threshold, whereas in TNG100-1 the minimum mass is around  $10^{9.4} M_{\odot}$ , which is set by the minimum number of stellar particles masking threshold multiplied by the mass resolution volume. There is approximately 1 order of magnitude difference in the minimum mass for EAGLE simulations, this is caused by the minimum number of stellar particles set for each subhalo in this analysis. For EAGLE simulations, the relation



**Fig. 4.1.** Mass-size plot of galaxy size measurements  $R_1$  in orange and  $R_{50}$  in soft blue against total stellar mass ( $M_\star$ ) for TNG50-1 (a) and both simulation volumes (b). Each point presents the mean value of galaxy size for all subhalos within each mass bin, with a total of 14 bins, where the horizontal axis coordinate of each point is the mid-bin value. The filled area around each size relation corresponds to the 16th and 84th percentile of each data point.

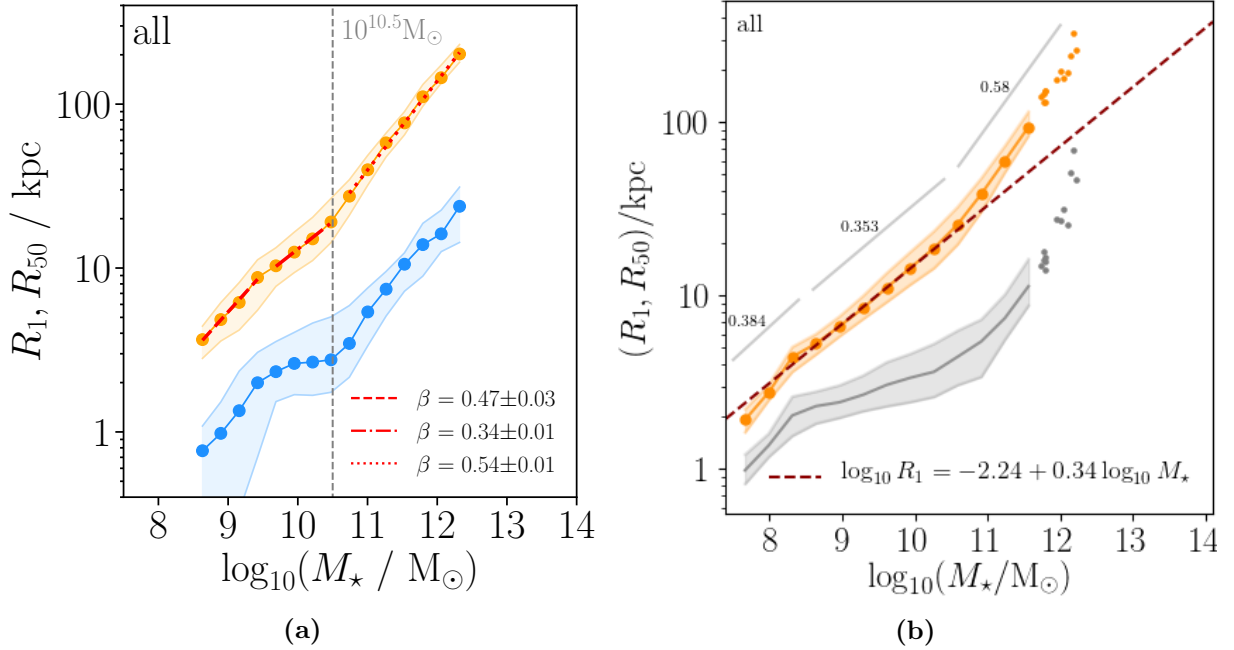
was performed using both RecalL0025N0752 and RecalL0100N1504 volumes, which have respectively  $2.26 \times 10^5 M_\odot$  and  $1.81 \times 10^6 M_\odot$  in gass cell resolution mass.

By matching both volume data and plotting the mass-size relation of the whole set, we can fit the 3 different slopes for mass ranges. We clearly differentiate a three component power law, limited firstly for stellar masses below  $10^{9.5} M_\odot$ , which might be related to a resolution effect, as it is also present in the EAGLE mass-size relation. Then the central and most important slope for spirals, with stellar masses between  $10^{9.5} M_\odot$  and  $10^{10.5} M_\odot$  and finally the last most massive galaxies above  $10^{10.5} M_\odot$ . This plot can be seen in Figure 4.2a, where the different slopes follow the data trend nicely.

The slope for the lower mass range is  $\beta = 0.47 \pm 0.03$  ( $\log_{10} R_1 = -3.51 + 0.47 \log_{10} M_\star$ ), in the intermediate-mass region  $\beta = 0.34 \pm 0.01$  ( $\log_{10} R_1 = -2.24 + 0.34 \log_{10} M_\star$ ) and for masses above  $10^{10.5} M_\odot$  we calculated a slope of  $\beta = 0.54 \pm 0.01$  ( $\log_{10} R_1 = -4.38 + 0.54 \log_{10} M_\star$ ). These last two values are consistent with  $\beta = 0.35 \pm 0.01$  and  $\beta = 0.58 \pm 0.02$  obtained by Trujillo et al. (2020).

The change of slope in  $10^{10.5} M_\odot$  can be clearly identified as the artificial burst of AGN feedback present in Illustris simulations, which is related to the quenching of star formation rate due to a change in the feedback model (explained in chapter 1). On the other hand, the EAGLE model presents a very gentle slope change, proper of a more natural model, as there is not an AGN feedback burst set as a simulation parameter.

Comparing with the results obtained with the EAGLE simulations (see Figure 4.2b), we can see the fitted slopes are very similar to the ones obtained by Trujillo et al. (2020). For



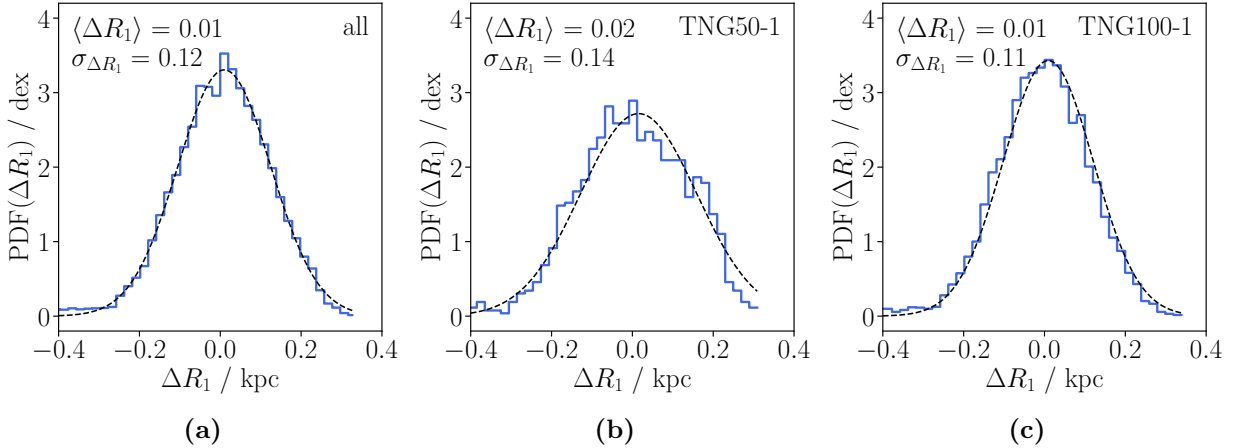
**Fig. 4.2.** Mass-size plot of galaxy size measurements  $R_1$  in orange and  $R_{50}$  in blue for IllustrisTNG (a) and grey for EAGLE (b). Both simulations combine different simulation volumes (TNG100-1 and TNG50-1 for Illustris and EAGLE RecalL0025N0752 and RefL0100N1504 for EAGLE). Each point presents the mean value of galaxy size for all subhalos within each mass bin, with a total of 14 bins for Illustris and 16 for EAGLE, where the horizontal axis coordinate of each point is the mid-bin value. The filled area around each size relation corresponds to the 16th and 84th percentile of each data point.

lower mass subhalos the relation slope is steeper than the actual literature value of  $\beta = 0.38$  and closer to our fitted value of  $\beta = 0.47 \pm 0.03$ , this might be due to the fact that computational models generate an excessive degree of stellar activity for that mass range when compared to empirical observations, as dwarf galaxies have much more physical processes to consider when modeling their star formation physics. For the central region, EAGLE data trend is consistent with the observations, with  $\beta=0.34$ .

## 4.2 Mass-size residual distributions and scatter

Another important aspect to mention is the scatter of  $R_1$  and  $R_{50}$  present in Illustris data is much larger than that in the literature and EAGLE simulations. In order to study this scatter we calculated the residual of each subhalo for Illustris data. Residuals are calculated by subtracting the value of each subhalo size indicator  $R_n$  to the value of the size indicator power law fit,  $\Delta R_n = R_n(\text{subh}) - R_n^{\text{fit}}(m_{\text{subh}})$ . In Figure 4.3 we can see the histogram of residuals fitted to a gaussian distribution to obtain the proper width of the distribution as an indicator of the mass-size relation scatter.

In the case of EAGLE simulations, the width of the gaussian fitting is  $\sigma_{R_1} = 0.06$  dex, consistent with the value reported by Trujillo et al. (2020),  $\sigma_{R_1} = 0.061 \pm 0.005$  dex. However, in the case of Illustris, we obtained a value of  $\sigma_{R_1}^{\text{all}} = 0.115 \pm 0.001$  dex, which is twice as large as the comparative values.



**Fig. 4.3.** Residual distribution of  $R_1$  values for TNG50-1 and TNG100-1 combined volumes (a), and for TNG50-1 (b) and TNG100-1 (c) isolated volumes. The histograms present the distribution of the residuals ( $\Delta R_1$ ) normalized to the area, presenting a PDF of the residuals in the vertical axis.

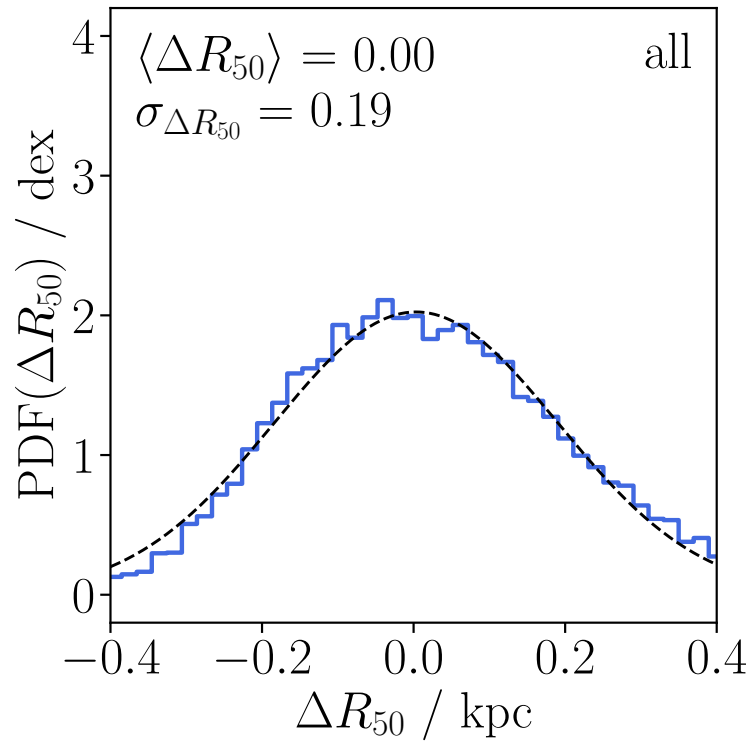
One might think that by matching different simulation volumes the scatter of the mass-size distribution gets larger. This problem is addressed in EAGLE simulations by performing a calibration of the data so the different simulation volumes have similar properties (Crain et al. 2015), which leads to a smaller scatter in the size-mass relation. This calibration step is not performed in Illustris simulations, this is why in Figure 4.3 we made a comparison between the distributions of residuals for each volume independently to check for resolution dependence.

We could associate the width ( $\sigma_{R_1}^{\text{TNG50-1}} = 0.143 \pm 0.004$  dex) and bad fitting of the TNG50-1 residual distribution to be the cause of the large scatter in the combined volumes residual distribution, but the TNG100-1 width ( $\sigma_{R_1}^{\text{TNG100-1}} = 0.111 \pm 0.001$  dex) is very close to the width of the combined volumes itself. We can then claim that the residual scatter has no dependence on the simulation volume size and resolution.

This scatter of data could also be associated with a bad computation of  $R_1$ . That is why in Figure 4.4 we present the distribution of  $R_{50}$  residuals, to serve as a comparison with a size measurement not computed by the code. The width in this case is observed to be  $\sigma_{R_{50}}^{\text{all}} = 0.188 \pm 0.002$ , indicating that the scatter of  $R_1$  is not due to a computation error, but rather it must be attributed to the scatter of subhalo masses in the Illustris simulations compared to EAGLE or empirical observations.

It is also important to mention that the scatter obtained for  $R_1$  residuals is about 1.6 times smaller than the one of  $R_{50}$ , whereas the relation between these two measured by Trujillo et al. (2020) is about 2.5. Also, the scatter is clearly larger for dwarfs and spirals but smaller for elliptical galaxies, this might be caused by the mechanism of these to grow, as merging and other physical mechanisms get a larger influence in the formation and growth of ellipticals than stellar activity. Also, as the literature says, the difference in star formation activity is larger for lower mass galaxies than for ellipticals (Li et al. 2010).





**Fig. 4.4.** Residual distribution of  $R_{50}$  values for TNG50-1 and TNG100-1 combined volumes. The histogram presents the distribution of the residuals ( $\Delta R_{50}$ ) normalized to the area, presenting a PDF of the residuals in the vertical axis.

# Chapter 5

## Conclusion

*This chapter of the text includes the conclusions and key concepts drawn from the different chapters of the text taking into account the initial considerations motivated in the first section of the paper. First, we present a brief summary of the process we have followed to arrive at the results, followed by the most important conclusions and key ideas that can be drawn from the mass-size relation results. Some insights are also drawn on the origin of the scatter of the residuals from the relation fit. Finally, some improvements in the code and future ideas for development beyond the current state of the project are presented.*

---

In this project, we apply a galaxy size definition based on stellar density  $R_1$  to combined sets of IllustrisTNG simulation volumes, studying the behavior of the mass-size relation of these galaxies and comparing them with EAGLE simulation results, which helps us understand how different AGN feedback models affect the galaxy size.

### Mass-Size relation

The mass size relation of IllustrisTNG using  $R_1$  as size indicator gives very similar values to the literature, especially in the spiral galaxy range, where the slope of  $\beta = 0.34 \pm 0.01$  is compatible with other studies (Trujillo et al. 2020; Mowla et al. 2019).

We detected an abrupt change in the mass-size relation for stellar masses around  $10^{10.5} M_{\odot}$  which can be related to the IllustrisTNG feedback model that forces a change in the feedback mode for Black Hole masses that correspond to those subhalo stellar masses, an effect also detected in Zinger et al. (2020).

After this slope change, we understand the relation for the most massive galaxies is steeper than the spiral galaxy mass range, as these larger galaxies have a growing mechanism related to merging processes, whereas the size evolution of spiral galaxies is driven by stellar formation activity.

As a final conclusion, we can state that  $R_1$  size definition is independent of AGN feedback, as simulations with very different feedback models give similar results. This positions  $R_1$  as a good galaxy size indicator when wanting to study galaxy size without taking feedback into

account. As a downside, this definition of galaxy size is not a good indicator for constraining feedback models.

## The scatter of the mass-size relation

When studying the scatter of the mass-size relation fitting residuals we found that the value of the scatter present in IllustrisTNG data ( $\sigma_{R_1}^{\text{all}} = 0.115 \pm 0.001$  dex) is larger than the value obtained in EAGLE ( $\sigma_{R_1} = 0.06$  dex) or by Trujillo et al. (2020) ( $\sigma_{R_1} = 0.061 \pm 0.005$  dex). We studied the source of this scatter, finding that it is not dependent on the simulation volume (numerical resolution), as the scatter for different volumes is systematically larger than observed. It is not related to a bad computation of  $R_1$ , as the scatter of the distribution for  $R_{50}$  was also larger than the one present in the literature and, on top of that, the ratio between  $R_1$  and  $R_{50}$  scatter values for the literature and IllustrisTNG results were approximately the same.

We also address the fact that the scatter values tend to get smaller for the most massive galaxies, which can be related to the growing mechanism being dominated by mergers.

This leads us to the conclusion that the scatter of the mass-size relation is not related either with the calculation of  $R_1$  or with the combination of different simulation volumes, but with the intrinsic scatter of galaxy stellar mass present in IllustrisTNG simulations, which could be an indicator that the source of this scatter is related to the difference in the feedback model.

## Future improvements

In the making of this thesis, we addressed many problems that were not finally explained and included in the text. As we had limited time there are some ideas on how to continue developing the project that have not been done yet.

Firstly, the pipelines and more specifically the data loading of the code should be parallelized to give a better overall performance. It is very important that large amounts of data are loaded through lighter versions of the code to speed up the computing process, as time and computing power are very important resources.

Studying the impact of AGN feedback in  $R_1$  for different redshift values will be interesting, as this can give us a better understanding on the impact of AGN feedback in the evolution of galaxy sizes in time.

Exploring different simulation volumes and threshold parameters for galaxy masking is also a very interesting approach, as the scatter of the mass-size relation residuals is not yet understood with clarity.

# Bibliography

- Begelman, M. C. 2014, *Accreting Black Holes*
- Chabrier, G. 2003, *PASP*, 115, 763
- Clavijo-Bohórquez, W. E., de Gouveia Dal Pino, E. M., & Melioli, C. 2023, *AGN and Star Formation feedback in the evolution of galaxy outflows*
- Crain, R. A., Schaye, J., Bower, R. G., et al. 2015, *Monthly Notices of the Royal Astronomical Society*, 450, 1937
- Dalla Vecchia, C. & Schaye, J. 2012, *MNRAS*, 426, 140
- Dolag, K., Borgani, S., Murante, G., & Springel, V. 2009, *MNRAS*, 399, 497
- Katsianis, A., Xu, H., Yang, X., et al. 2020, *Monthly Notices of the Royal Astronomical Society*, 500, 2036
- Kennedy, R., Bamford, S. P., Baldry, I., et al. 2015, *Monthly Notices of the Royal Astronomical Society*, 454, 806
- Kormendy, J. & Ho, L. C. 2013, *Annual Review of Astronomy and Astrophysics*, 51, 511
- Li, I. H., Glazebrook, K., Gilbank, D., et al. 2010, *Monthly Notices of the Royal Astronomical Society*, 411, 1869
- Libeskind, N. I., van de Weygaert, R., Cautun, M., et al. 2017, *Monthly Notices of the Royal Astronomical Society*, 473, 1195
- McNamara, B. R. & Nulsen, P. E. J. 2012, *New Journal of Physics*, 14, 055023
- Mowla, L., van der Wel, A., van Dokkum, P., & Miller, T. B. 2019, *The Astrophysical Journal Letters*, 872, L13
- Nesvadba, N. P. H., Boulanger, F., Salomé, P., et al. 2010, *A&A*, 521, A65
- Pillepich, A., Springel, V., Nelson, D., et al. 2017, *Monthly Notices of the Royal Astronomical Society*, 473, 4077
- Planck Collaboration. 2016, *Astronomy & Astrophysics*, 594, A13
- Rodriguez-Gomez, V., Genel, S., Vogelsberger, M., et al. 2015, *MNRAS*, 449, 49
- Rosito, M. S., Pedrosa, S. E., Tissera, P. B., et al. 2021, *A&A*, 652, A44
- Schaye, J., Crain, R. A., Bower, R. G., et al. 2014, *Monthly Notices of the Royal Astronomical Society*, 446, 521

- Springel, V. 2005, Monthly Notices of the Royal Astronomical Society, 364, 1105
- Springel, V. 2010, Monthly Notices of the Royal Astronomical Society, 401, 791
- Springel, V., White, S. D. M., Jenkins, A., et al. 2005, Nature, 435, 629
- Stoughton, C., Lupton, R. H., Bernardi, M., et al. 2002, The Astrophysical Journal, 123, 485
- Trujillo, I., Chamba, N., & Knapen, J. H. 2020, Monthly Notices of the Royal Astronomical Society, 493, 87
- Wylezalek, D. & Zakamska, N. L. 2016, MNRAS, 461, 3724
- Zinger, E., Pillepich, A., Nelson, D., et al. 2020, Monthly Notices of the Royal Astronomical Society, 499, 768