illustris_elt Documentation

Kieran Leschinski

Jul 28, 2020

Contents:

The main goal of this project is to connect the Illustris cosmological simulation database to the ELT/MICADO instrument simulator, in order to determine the observational limits of the ELT/MICADO system with respect to star formation before cosmic noon (z>2).

The ELT will have sensitivity similar to the next generation of space-based telescopes, but with a resolution 50x better. This will enable the study of the spatial characteristics of the earliest galaxies on scales of ~100pc. This in turn will enable the scale and spatial extent of the star formation regions to be understood, and should shed light on just how the first stars and galaxies formed in the universe.

CHAPTER 1

Scope and Outcomes

1.1 Introduction

Illustris is a large cosmological simulation suite that has simulated the history of a "universe in a box" over a time frame of 13 Gyr. The primary building block of the simulations is the "dark-matter particle", which has an initial mass of 6 million solar masses (6x10^6 Msun) and an initial gas mass of 1.3x10^6 Msun (http://www.illustris-project.org/data/). Each particle also contains various sub-grid parameters, such as stellar mass, star formation rate, number of black holes, etc.

Todo: Continue this explanation

As time progresses in the the simulation, particles group together, and galaxies form in the dense centres of these groups (the so-called Sub-halos. Halos are the larger overarching groups that surround galaxy clusters). The snapshots of this universe-in-a-box allow us to view the evolution history of a certain galaxy all the way back to its beginning.

Simulations are one thing, reality is often different. Hence out main goal of this project is to determine whether the predictions made through this cosmological simulation will actually be testable in real life - i.e. will the next generation of telescopes be able to observe the phenomena seen in the simulations.

The current generation of telescopes, through simple physics, will never be able to resolve galaxies at distances larger than z>3, simply because the galaxies are two small (D<1 kpc), the diffraction limit of an 8m class telescope (VLT) is around 0.05" in K-band (lam/D), and the cosmological angular size scale is ~8kpc/" (0.125"/kpc) at 1<z<3 (http://www.astro.ucla.edu/~wright/CosmoCalc.html). Even the JWST, with its 6.5m mirror, will struggle to get any spatial information from high-z galaxies.

The ELT however, with a 40m mirror and a corresponding diffraction limit of 0.007" in J (0.012" in K), will have a spatial resolution of ~60pc (100pc) in J (K), and will thus be able to provide information on the spatial variations of star formation in the very first galaxies. Unfortunately the ELT is a ground-based telescope, and consequently suffers from a sensitivity limit due to atmospheric extinction.

The main goals of this project revolve around the central theme:

```
What will actually be bright enough for the ELT to observe?
```

1.2 Scope and Method

- We will concentrate on using 3D particle data form the illustris API
- We will generate realistic 2D on-sky projections of the particle data using the hyperion radiative transfer code
- We will create simulated observations of particle data using the SimCADO instrument data simulation software
- We will determine what star formation rates will be visible for a redshift and volume density

CHAPTER 2

Project Description

2.1 Background

- ELT
- Star formation
- Universal in all environments?
- Constant over time / outside the Milky Way?
- High-z galaxies
- What is different to today's galaxies

2.2 Tools

- **Illustris**
  - What is Illustris?
  - Why is this interesting to use?
- **SimCADO**
  - What is SimCADO?
  - What can SimCADO deliver?

2.3 Questions

- **Using SmCADO to observe galaxies generated by Illustris**
  - Why? - To determine the observational limits of key observational quantities?

    - **Why? -**
        * Because time on the ELT (TMT etc) will be very expensive
        * Because we want to find out which fundamental questions the ELT will help us answer
- **What properties of the illustris galaxies do we want to look at?**
    - Star formation rate
    - Star formation distribution at different z
    - General galaxy structure in high-z galaxies
    - Consequences of Interactions between galaxies at high-z

2.4 Success Criteria

- **We have a pipeline which**
    - queries Illustris galaxies
    - downloads particle information that belong to the galaxies
    - **projects the galaxies in any orientation**
        * also determines which

CHAPTER 3

Tools and software to use in this package

This project will make use of the following software packages and tools:

Todo: Describe the reason why we are using these tools, and how

3.1 Python packages

- **IllustrisAPI**
  - https://github.com/zpenoyre/illustrisAPI
- **Hyperion**
  - http://docs.hyperion-rt.org/en/stable/index.html
- **SimCADO**
  - https://simcado.readthedocs.io/en/latest/index.html
  - https://github.com/astronomyk/SimCADO

3.2 Testing framework

- pytest
- **TravisCI:**
  - https://travis-ci.org/astronomyk/illustris_elt

3.3 Documentation framework

- **ReStructured Text (RST) - for writing the documentation**
  - http://docutils.sourceforge.net/docs/user/rst/quickref.html
  - https://docs.typo3.org/m/typo3/docs-how-to-document/master/en-us/WritingReST/CheatSheet.html
- **ReadTheDocs (RTD) - for displaying the documentation**
  - https://illustris-elt.readthedocs.io/en/latest/

CHAPTER 4

Roadmap for the Project

4.1 Get initial galaxy data

1. **Choose 2 galaxies to follow their history**
    - massive elliptical
    - beautiful spiral
2. **Find their merger history**
    - Get sub_halo_id number for each galaxy for each of the 135 snapshots
    - Assume that the merger history continues along the line of largest merger participant
3. **Decide what data will be interesting**
    - Star formation rate
    - Filter magnitudes
    - **Particle positions**
        - Particle sizes too (if available)
    - **Mass of particle**
        - Total mass
        - Dark matter mass
        - Gas mass
        - Star mass
4. **Decide which particles in the sub-halo "belong" to the galaxy at the centre of the sub-halo**
    - Do we decide on a star-mass density limit? Or combined star+gas mass density limit
    - Do we use a luminosity (density) limit?

4.2 Pull data from illustris database

1. Write function to get the above mentioned data for a galaxy at a particular snapshot
2. Save the data somewhere in a coherent naming scheme
3. **Decide, based on query/download speed whether to download data again every** time the scripts are run, or whether to save the data in cache and load from disk (i.e. `astropy.download_file(url=..., cache=True)`)

4.3 Make "ideal" mock 2D maps/images of galaxy

"Ideal" maps/images mean what the galaxy would look like before a telescope was used to observe it. This could mean we display the stellar mass of the galaxy in an image, or actually use the intensity of a galaxy in a given filter band (UVIKgriz). We also want to be able to view the galaxy from any angle

1. Function to display an integrated 2D "column-density"-style map of the galaxy for any set of rotation angles
2. Function to automaticaly determine the semi-major and -minor axes of the sub-halo ellipsoid
3. **Function to rotate particle positions to the following orientations:**
    - Face on - view of largest and 2nd largest axes
    - Edge on - view of largest and smallest axes
    - End on - view of smallest and 2nd smallest axes

    This could be done with a principle component analysis
4. **Function to make maps of the following of each galaxy for each of the three main orientations:**
    - Flux intensity in each of the UVRIgriz filters
    - Star formation rate
    - Stellar mass
    - Gas + stellar mass
    - Dark matter
5. **Function to generate these maps on-the-fly and return a fits image at a giving spatial angular resolution, given a snapshot**
    - See Ned Wright's cosmological calculator for angular distances vs redshift
6. **Combine this funtionality into a python package**
    - Write the appropriate tests, and documentation

Time for the fun stuff:

7. **Generate these maps for each of the snapshots of a galaxy**
    - Create GIFs of the evolution of the galaxy over cosmic time
    - Create these maps for a fixed viewing angle
    - Create these maps oriented along one of the principle axes
8. **Plot integrated parameters over the whole galaxies vs cosmic time (snapshot and lookback time)**
    - Each of the parameters listed in 4.

CHAPTER 5

The idea for Illustris ELT

The main idea:

```
What is the faintest and smallest galaxy in the Illustris sample that will be visible␣
↪to the ELT
```

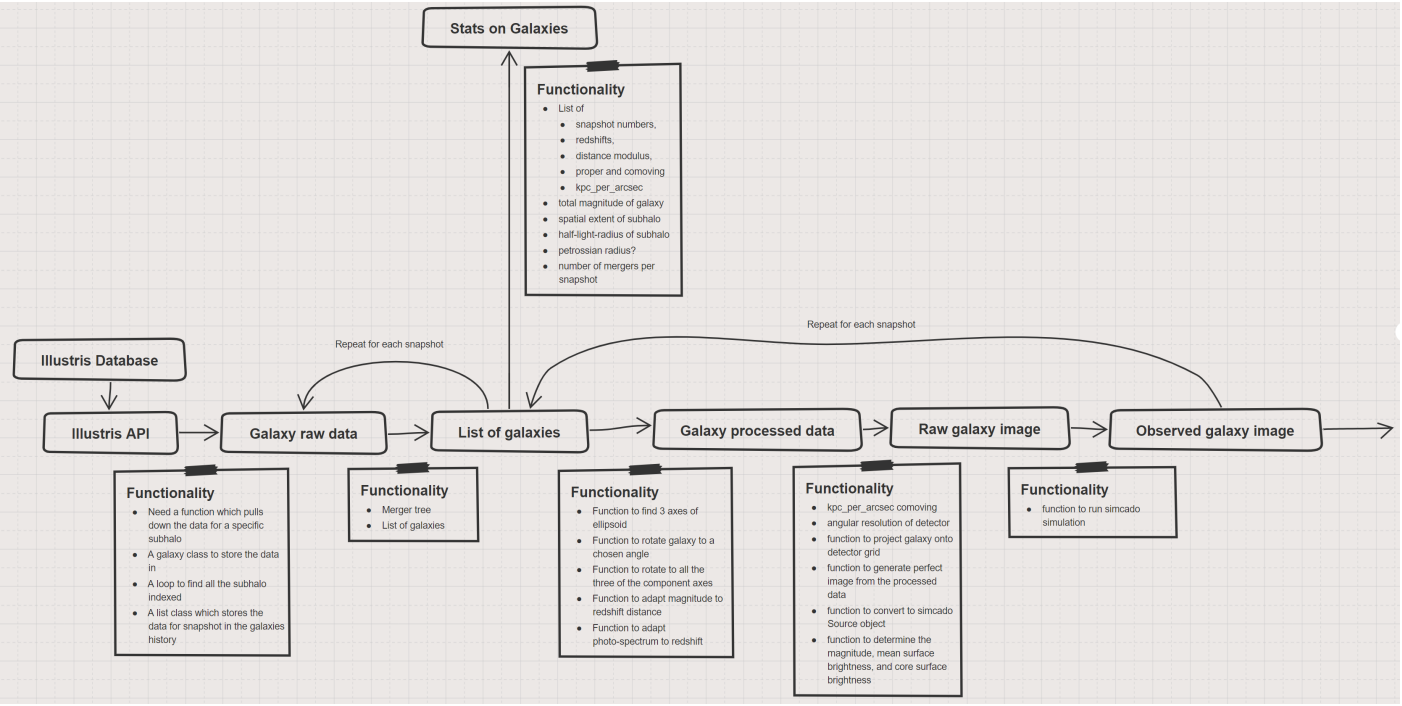

Fig. 1: Workflow for Illustris-ELT

Steps involved:

Doctests:

```
>>> 1 == 1
True
```

CHAPTER 6

Literature list

6.1 High-z galaxies

- **Förster-Schreiber (PPT pres)**
  - Star Formation in high z Galaxies: Star Formation in high z Galaxies: Insights from Resolved Observations Insights from Resolved Observations of the Ionized and Molecular Gas of the Ionized and Molecular Gas
  - https://www.noao.edu/meetings/first-light/Sunday0314/Morning/ForsterSchreiber_NOAO50th.pdf
- **Madau+ 2014 (Review)**
  - Cosmic Star Formation History
  - https://arxiv.org/abs/1403.0007

6.2 Star formation

- **Bastian+ 2010 (Review)**
  - https://arxiv.org/abs/1001.2965
  - A Universal Stellar Initial Mass Function? A Critical Look at Variations

6.3 Illustris

- **Vogelsberger+ 2014**
  - https://arxiv.org/abs/1405.2921
  - Introducing the Illustris Project: Simulating the coevolution of dark and visible matter in the Universe
- **Genel+ 2014**

  - https://arxiv.org/abs/1405.3749
  - Introducing the Illustris Project: the evolution of galaxy populations across cosmic time

- **Sparre+ 2014**
  - https://arxiv.org/abs/1409.0009
  - The star formation main sequence and stellar mass assembly of galaxies in the Illustris simulation

- **Asa+ 2019**
  - https://arxiv.org/abs/1902.01665
  - What shapes a galaxy? - Unraveling the role of mass, environment and star formation in forming galactic structure

CHAPTER 7

The “Science as Software” methodology

There are many similarities between scientific research and software development. Today many scientists develop their own software tools for their scientific research, and as such use many of the development tools and methodologies from software engineering. The question we pose here is:

```
Is it possible to use software engineering methodologies to run a
scientific project – essentially doing "science engineering"?
```

Science and Software development have a significant overlap between their respective discipline methodologies:

Software	Science
Defining scope	Background research
Requirements	Formulate hypothesis
Design	Experimental method planning
Coding	Data preparation / reduction
Testing	Data analysis / Hypothesis tests
Documentation	Write / publish paper
Deployment	Present work at conferences

Over the last couple of decades the whole software industry has used these steps as a guide to delivering software on schedule to meet a certain goal. Scientific research on the other hand follows a much more fluid format, where scientists “play” with data to prove or disprove a certain idea.

Unfortunately the code writen to “play” with data is too often writen ad-hoc and then thrown away after the project is complete. More often that not the code is messy and undocumented, such that when the scientist returns to the code 6 months later, they can no longer make sense of their own constructions. Cumulatively this leads to mountains of code being thrown away and results in days and weeks of “wasted” time, as the results of any such study are difficult to replicate or repeat.

Software developers on the other hand have long realised that their software will most likely out-live them in their current position, and as such the code *should* be writen in a way that the next developer on the project can read and (more importantly) maintain the code base.

Given that both fields are essentially writing code to solve problems, might it not be prudent for scientists then to also adopt such a mindset and the associated practices and methodologies? Here I argue: yes they should!

In this project we will try to follow the workflow of a software developer to attempt to answer a series of scientific questions in a coherent, repeatable, documented, and above all orderly manner. Whether or not this methodology is indeed applicable to science remains to be seen, but given the state of current scientific coding techniques, it is definitely worth the attempt:

```
Leave no code behind!
```

CHAPTER 8

Goals of this study

Target z>2, goal z>5

- Determine surface brightness observation limits for individual star forming regions in z>2 galaxies in all the major NIR filters
  - Calculate what star formation rate this translates to
- Determine whether the star formation mechanism (triggered vs jeans collapse) will be observable in z>2 galaxies by observing
- See whether this technique of "science as software" is a viable methodology for scientific research