

EXPLORING LINE INTENSITY CUBE PREDICTIONS WITH `limlam_mock`

PATRICK HORLAVILLE^{1,2}

Subject headings: cosmology: galaxy — dark matter — CO line intensity

1. INTRODUCTION

1.1. Background

The study of high redshift galaxies is increasingly important in the context of our understanding of galaxy formation and cosmology. While modern surveys still struggle to efficiently probe those populations, simulation techniques such as intensity mapping have proven to be able to infer the properties of those dear high redshift galaxies, *e.g.* intensity mapping of carbon monoxide and its ground-state transition (Li et al., 2016) [2].

Intensity mapping (IM), or line intensity mapping (LIM), is a technique that consists of probing the 3D sky (RA, Dec & ν_{obs}) at low angular resolution of specific molecular lines (Karkare, *n.d.* [1]). The strength of this analysis resides in 1) its potential to probe large cosmological volume quickly and 2) its ability to probe galaxies invisible to standard galaxy surveys.

1.2. Related to the Project

The goal of this project is to analyze how predicted galaxies' properties as found by Li et al. (2016) [2] compare to those of a Python program named `limlam_mock`. The two aim to perform a line intensity analysis using CO emission over dark matter halos, which allows to compute their CO brightness temperature. However, two main differences are to be considered. First, the average redshift of the probed halos are not the same for the two analyses. The population used by Li et al. (2016) [2] has $z \sim [2.4, 2.8]$. For the `limlam_mock` comparison, refer to Fig. 2. Second, the method used to identify those halos are not the same. This will be elaborated upon later.

Both analyses consider a minimum cutoff for halos' mass. Indeed, low-mass halos are not expected to be CO-bright as they are expected to be poor of systems rich in dust or metal (Li et al., 2016) [2]. Then, how does the CO brightness temperature depend on the chosen mass cutoff? The final objective of this project would be to use `limlam_mock` to determine this relationship and compare it to the result of Li et al. (2016).

2. PROCESS

2.1. Getting Started

To first get started with the `limlam_mock` program, I cloned it to my home repository in my CITA environment. From there, I looked into some of its objects. The first one is a lightcone simulation output within the `catalogues` directory. It notably contains the virial mass (in solar masses M_{\odot}) and the redshift z of every halo.

Fig. 1 and 2 display the $\text{Log}(\text{mass})$ and redshift distribution of those halos present in our catalogue.

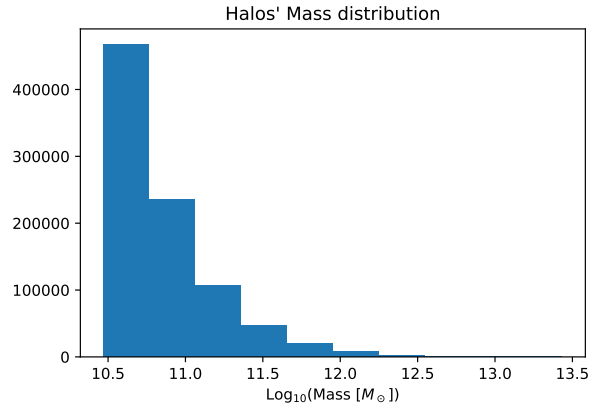


FIG. 1.— Here shown is the distribution of halos' $\text{Log}(\text{masses})$ from the catalog used in our `limlam_mock` analysis. As indicated by the exponential x -scale, we have $M \sim [10^{10}, 10^{13}] M_{\odot}$ for our catalog.

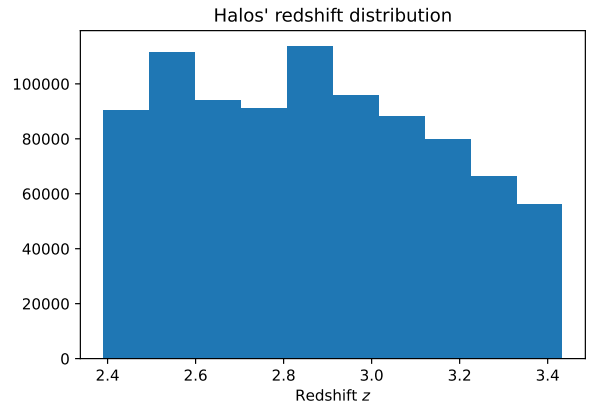


FIG. 2.— Here shown is the distribution of halos' redshifts from the catalog used in our `limlam_mock` analysis. This population of halos differs to the one of Li et al. (2016) as a significant part of it has $z \in [2.8, 3.4]$.

2.2. Tweaking the Code

Since the inception of `limlam_mock`, `numpy` and possibly other packages upon which the program depends upon have changed so that running the code as it is yields an error. This error was solved by adding the argument `allow_pickle = True` while loading the halo catalogue in the file `load_halos.py`.

¹ CITA, University of Toronto

² McGill University

2.3. Running the Code

The file performing the line intensity mapping, `lim_mock.py`, can now be ran. Now that this is solved, we can investigate how the program works in its great lines.

First, a mock map is being generated out of parameters specified in the file `params.py`. Then, `load_peakpatch_catalogue` allows to load the halos from the catalogue. After being loaded, the halos undergo the minimum mass cutoff with `cull_peakpatch_catalogue`. From there, multiple routes are possible to retrieve the CO brightness temperature of our halos. Here we follow the procedure prescribed by Li et al., 2016 [2], which is the default method in the `limlam_mock` package. The halo mass densities can be translated to stellar mass functions, which are then related to star formation rates (SFR). Given SFRs are not directly observable, they can be inferred from spectral tracers such as the total infrared luminosity L_{IR} as given by Equation 1:

$$\text{SFR} = \delta_{\text{MF}} \times 10^{-10} L_{\text{IR}}. \quad (1)$$

The infrared luminosity can then be converted to CO luminosity as given by Equation 2, where the relationship between the two is assumed to be a power law with parameters α and β :

$$\log L_{\text{IR}} = \alpha \log L'_{\text{CO}} + \beta, \quad (2)$$

where the units of L_{IR} are L_{\odot} and those of L'_{CO} are $\text{K km s}^{-1} \text{ pc}^{-2}$. Li et al., 2016 [2] use values of $\alpha = 1.37$ and $\beta = -1.74$ from a census of high-redshift galaxies.

L'_{CO} can be converted into L_{CO} (with units of L_{\odot}) as per Equation 3:

$$L_{\text{CO}} = A \times L_{\odot} \left(\frac{\nu_{\text{CO,rest}}}{115.27 \text{ GHz}} \right)^3 \left(\frac{L'_{\text{CO}}}{\text{K km s}^{-1} \text{ pc}^{-2}} \right), \quad (3)$$

where A is a fudge factor of 4.9×10^{-5} and $\nu_{\text{CO,rest}}$ is the rest-frame frequency of the CO transition (115.27 GHz). The brightness temperature, finally, can be found through the Rayleigh-Jeans relation as shown in Equation 4:

$$T = \frac{c^2 I_{\nu, \text{obs}}}{2 k_b \nu_{\text{obs}}^2} \quad (4)$$

where k_b is the Boltzmann constant and $I_{\nu, \text{obs}} = \frac{L_{\text{CO}}}{4\pi D_L^2 \delta_\nu}$, inside which D_L is the luminosity distant to the source, δ_ν is the width of the spectral resolution element, a parameter determined by the prospective resolution of future instruments that will carry on those observations.

One of the outputs of the program, an `.npz` file, contains the sought-after CO brightness temperature for every point in our grid.

2.4. Loading Results

Each run of `limlam_mock.py` corresponds to finding the CO brightness temperature across all of our map for one specified threshold of minimum mass. If we run it three times with $\text{min. mass} = 3 \times 10^{10}, 10^{11}$ and $10^{12} M_{\odot}$,

we can take the average over the cube of the CO brightness temperature at each time and compare this preliminary result with the results found by Li et al. (2016) [2] between the minimum mass threshold chosen and the resulting average CO brightness temperature. The results are shown in Fig. 3

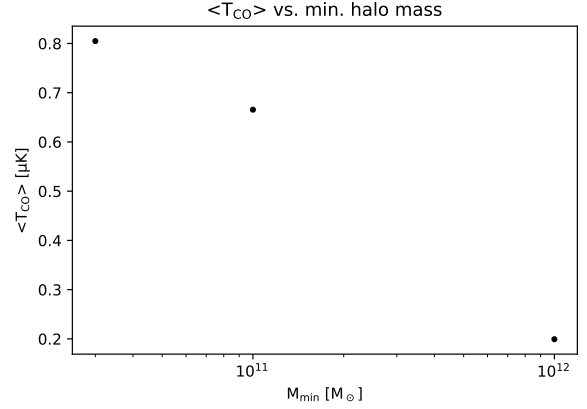


FIG. 3.— Here shown are the results from 3 runs of the `limlam_mock` analysis using different minimum halo mass thresholds. The values found are compared with their equivalent from Li et al. (2016) [2] in Table 2.4.

TABLE 1
COMPARISON OF RESULTS BETWEEN LI ET AL. (2016) [2]
AND `LIMLAM_MOCKER`.

	Min. Mass [M_{\odot}]	$\langle T_{\text{CO}} \rangle [\mu\text{K}]$
<code>limlam_mock</code>	3×10^{10}	0.80
Li et al. (2016) [2]	1×10^{10}	0.92
<code>limlam_mock</code>	1×10^{11}	0.67
Li et al. (2016) [2]	1×10^{11}	0.67
<code>limlam_mock</code>	1×10^{12}	0.20
Li et al. (2016) [2]	1×10^{12}	0.26

Notice that both methods seem to give results of the same order. This comparison is investigated further in the next section.

2.5. Iterating Over a Lot of Minimum Masses

Rather than being stuck with three data points, we would like to have a continuous-like array of minimum masses and corresponding average CO brightness temperature values to better establish the relationship between those two quantities. To do so, I modified the scripts from the `limlam_mock` repo so as to integrate notably `N`, the number of desired iterations as well as `ubound` and `lbound`, the upper and lower bound on the log space of values over which the iterations are being performed. A few functions have to be adapted here and there, but it's mainly a matter of integrating a `for` loop over all the process. The resulting relationship is shown in Fig. ???. The equivalent result from Li et al. (2016) [2] is shown in Fig. ???.

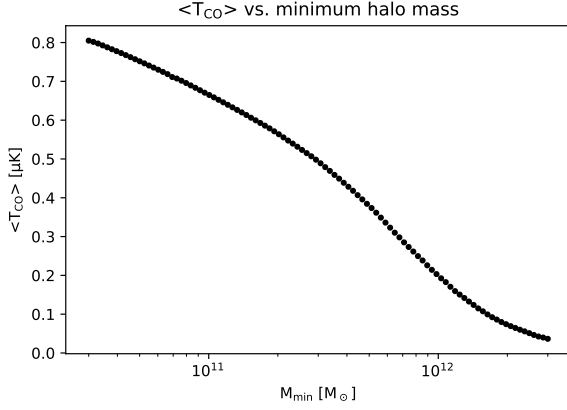


FIG. 4.— Here shown are the results from 100 runs of the new `limlam_mock` program, namely `new_llm`. The values chosen for the iteration were `lbound` = 3×10^{10} , `ubound` = 3×10^{12} [in M_{\odot}] and $N = 100$. We can compare this graph to Figure 11 from Li et al. (2016) [2], as shown in Fig. ??.

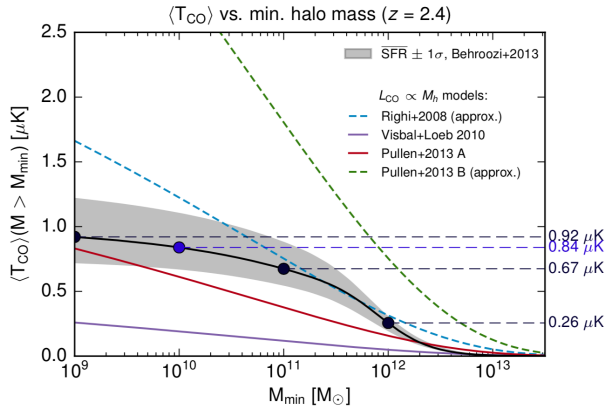


FIG. 5.— Here shown is Figure 11 of Li et al. (2016) [2], displaying the relationship between the chosen minimum halo mass and the corresponding average CO brightness temperature [in black].

3. DISCUSSION & CONCLUSION

Fig. ?? and Fig. ?? show a qualitative match between the `limlam_mock` and the Li et al. (2016) [2] relationship between M_{\min} and $\langle T_{CO} \rangle$.

This is already remarkable given the different methods used. Li et al. (2016) [2] had recourse to a cosmological N -body dark matter simulation to retrieve the dark matter halos, which is computationally very expensive compared to the method used by `limlam_mock`: the Peak Patch method. The mass-Peak Patch method allows to look into very large volumes of halo catalogues without having to go into the numerical dread of N -body simulations. Despite being a more approximate method, the mass-Peak Patch method has proven to be efficient for creating large volumes of halo catalogues (Stein et al., 2018 [3]). Here, it seems to have predicted the relationship between halo minimum mass cutoff and average CO brightness temperature just as an N -body simulation would have.

The functionality of the mass-Peak Patch method promises to be of help in the simulation of large cosmological dark matter halo populations. I hope I can get to make interesting use of this tool in the prospect of my research work this summer, notably in other contexts like C II line intensity mapping.

References

- [1] Kirit S. Karkare. *Line Intensity Mapping*. URL: <https://scholar.harvard.edu/kkarkare/line-intensity-mapping> (visited on 05/16/2022).
- [2] Tony Y. Li et al. “CONNECTING CO INTENSITY MAPPING TO MOLECULAR GAS AND STAR FORMATION IN THE EPOCH OF GALAXY ASSEMBLY”. In: *The Astrophysical Journal* 817.2 (Jan. 2016), p. 169. DOI: 10.3847/0004-637x/817/2/169. URL: <https://doi.org/10.3847/0004-637x/817/2/169>.
- [3] George Stein, Marcelo A Alvarez, and J Richard Bond. “The mass-Peak Patch algorithm for fast generation of deep all-sky dark matter halo catalogues and its N -body validation”. In: *Monthly Notices of the Royal Astronomical Society* 483.2 (Nov. 2018), pp. 2236–2250. DOI: 10.1093/mnras/sty3226. URL: <https://doi.org/10.1093/mnras/sty3226>.