



Universiteit
Leiden
The Netherlands

UCLCHEMCMC: an MCMC inference tool for physical parameters of molecular clouds

Keil, M.; Viti, S.; Holdship, J.R.

Citation

Keil, M., Viti, S., & Holdship, J. R. (2022). UCLCHEMCMC: an MCMC inference tool for physical parameters of molecular clouds. *The Astrophysical Journal*, 927(2).
doi:10.3847/1538-4357/ac51d0

Version: Publisher's Version
License: [Creative Commons CC BY 4.0 license](#)
Downloaded from: <https://hdl.handle.net/1887/3561949>

Note: To cite this publication please use the final published version (if applicable).



UCLCHEMCMC: An MCMC Inference Tool for Physical Parameters of Molecular Clouds

Marcus Keil¹ , Serena Viti^{2,1} , and Jonathan Holdship^{2,1} ¹ University College London, Gower St, Bloomsbury, London WC1E 6BT, UK; marcus.keil.19@ucl.ac.uk² Leiden Observatory, Leiden University, P.O. Box 9513, 2300 RA Leiden, The Netherlands

Received 2021 August 4; revised 2022 January 31; accepted 2022 February 2; published 2022 March 17

Abstract

We present the publicly available open-source code UCLCHEMCMC, designed to estimate physical parameters of an observed cloud of gas by combining Markov chain Monte Carlo (MCMC) sampling with chemical and radiative transfer modeling. When given the observed values of different emission lines, UCLCHEMCMC runs a Bayesian parameter inference, using an MCMC algorithm to sample the likelihood and produce an estimate of the posterior probability distribution of the parameters. UCLCHEMCMC takes a full forward-modeling approach, generating model observables from the physical parameters via chemical and radiative transfer modeling. While running UCLCHEMCMC, the created chemical models and radiative transfer code results are stored in an SQL database, preventing redundant model calculations in future inferences. This means that the more UCLCHEMCMC is used, the more efficient it becomes. Using UCLCHEM and RADEX, the increase in efficiency is nearly two orders of magnitude, going from 5185.33 ± 1041.96 s for 10 walkers to take 1000 steps when the database is empty, to 68.89 ± 45.39 s when nearly all models requested are in the database. In order to demonstrate its usefulness, we provide an example inference of UCLCHEMCMC to estimate the physical parameters of mock data, and perform two inferences on the well-studied prestellar core, L1544, one of which shows that it is important to consider the substructures of an object when determining which emission lines to use.

Unified Astronomy Thesaurus concepts: [Astrochemistry \(75\)](#); [Interdisciplinary astronomy \(804\)](#); [Markov chain Monte Carlo \(1889\)](#); [Bayesian statistics \(1900\)](#); [Posterior distribution \(1926\)](#)

1. Introduction

Throughout the interstellar medium (ISM), chemical reactions impact the environments that we observe. In turn, the physics of a molecular cloud greatly affects the chemistry. For example, at high densities ($\gtrsim 10^5 \text{ cm}^{-3}$) and low temperatures ($\lesssim 30 \text{ K}$), atoms and molecules freeze out onto dust grains, where they can react through many pathways (for a review, see Allodi et al. 2013). On the other hand, shocks from protostellar outflows impacting the surrounding medium can lead to desorption of many molecular species stored on dust grains (Caselli et al. 1997). Hence, emission and absorption lines of different species allow us to study the physics of the objects we observe.

In order to interpret the observations that are made, radiative transfer codes can be used to calculate the expected intensities that should be observed with a given set of physical parameters. One example of such a code is RADEX (van der Tak et al. 2007), which focuses on nonlocal thermal equilibrium (non-LTE) analysis. These types of codes require parameters that describe the condition of the gas as well as the column density of a species. These are connected by chemistry, but are treated as free parameters in many radiative transfer models. Modeling tools such as UCLCHEM (Holdship et al. 2017) or GRAINOBLE (Taquet et al. 2012), among many others, provide the fractional abundances of species, which can be used to calculate the column density. The fractional abundances can be combined with estimates of the total gas column

density of an object in order to calculate the column density of an individual species.

Chemical models calculate fractional abundances by considering the rate of change of many species as they interact through a network of reactions. These reaction networks usually include a gas-phase database such as KIDA (Wakelam et al. 2012) or UMIST (McElroy et al. 2013), as well as gas-grain and grain surface processes such as freeze out, nonthermal desorption, and surface reactions. Additional processes such as thermal desorption or sputtering of ice mantles are often included, depending on the chemical code and its intended purpose. The complexity involved in determining what should be included in these models in order to maximize the accuracy while minimizing computational cost of creating a model is an aspect that requires significant expertise.

The best modeling approaches combine chemical modeling codes with radiative transfer codes. One benefit in doing this is that the column densities can be calculated with the chemical model, which can then be combined with the set of physical parameters calculated by the chemical code to use as parameters for the radiative transfer model. There are additional parameters for the radiative transfer codes, such as the line width, which are not directly calculated by a chemical code, but can be treated as free parameters or derived from observations. The outputs from the radiative transfer code can then be compared to spectroscopic observations (Viti 2017; Punanova et al. 2018; Harada et al. 2019).

In order to assist in the inference of physical parameters of an observation, we present the open-source Markov chain Monte Carlo (MCMC) inference tool UCLCHEMCMC.³ The



Original content from this work may be used under the terms of the [Creative Commons Attribution 4.0 licence](#). Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

³ <https://zenodo.org/badge/latestdoi/10.3847/1538-4357/ac51d0>

intended use of UCLCHEMCMC is to infer the probability distribution of key physical parameters given some observed data. The following section describes the code in detail, starting with the forward-modeling approach and the tools it uses in Section 2.1, followed by the workflow of the code and how it stores models to allow future inferences to be more efficient in Section 2.3, which in turn is followed by a brief description of the chosen interface in Section 2.4. After this, we examine an example case by providing UCLCHEMCMC with mock observations, and we then perform a stress-test inference using observations of the prestellar core L1544 in Section 3, before we proceed to discuss caveats for the use of this tool. We summarize in Section 4.

2. UCLCHEMCMC

UCLCHEMCMC infers physical parameter values from molecular observations using chemical and radiative transfer models. First, we use a chemical model, in order to obtain abundances for a user-defined list of species for which UCLCHEMCMC was configured. These abundances can then be used with a radiative transfer model to calculate the intensities for the emission lines of those species. For a single model, this process can take several minutes to be calculated on a standard computer.

In order to infer the physical parameter values, we use the affine-invariant MCMC ensemble algorithm of Goodman & Weare (2010) as implemented in the Python package *emcee* (Foreman-Mackey et al. 2013). This kind of sampling initiates walkers with a set of physical parameters, which are used to calculate the chemical and radiative transfer models as just described. During each step, the walkers calculate the likelihood value for that set of parameters using the likelihood function given by the user (see Section 2.2 for details of the likelihood). After calculating the likelihood of the current values, the walkers choose a new set of values in parameter space for which the likelihood is also evaluated. At this point, the walker must decide whether it remains stationary for this step, discarding the new set of values and keeping the one it already has, or if it discards the old values and keeps the new ones. This decision is dependent on the type of “move” function that is chosen (for details, we refer to Foreman-Mackey et al. 2013). The default function for UCLCHEMCMC is a combination of a differential evolution proposal (Nelson et al. 2013) and a snooker proposal using differential evolution (ter Braak & Vrugt 2008). Mixing like this is recommended by Foreman-Mackey et al. (2013) when dealing with multimodal problems because the differential evolution proposal can allow for large enough step sizes to cross between peaks in probability. However, the standard version strongly recommends that at least $N = 2d$ walkers are used, where d is the number of parameters over which to infer, and it performs better with higher N . The snooker proposal using differential evolution allows for an improved performance compared to the basic version, when a smaller number of walkers is used. The process of sampling parameter space in this way requires thousands of models to be calculated before a meaningful posterior is produced. The calculations of each model using UCLCHEM and RADEX can take a minute or more, which can result in several hours of computing time before the parameter space is sampled well enough to produce a usable posterior.

Our aim is to improve the efficiency without decreasing the accuracy. To do this, UCLCHEMCMC manages a database of

previously calculated models from which it can retrieve values when required. The curated database contains the input and output from both the chemical and radiative transfer models and is used to perform an inference of the physical parameter space of an observed object without repeating any calculation. Anytime a new step is taken, it can check if this combination of parameters has been used before, and if it has, use the old output rather than perform a new calculation. A full flowchart of the processes done can be found in Figure 1. In this section, we start by briefly describing the forward-modeling method we use that the software UCLCHEMCMC requires in order to create the models that it stores, followed by the details of the MCMC inference and how UCLCHEMCMC manages the database, before detailing the interface it has.

2.1. Forward Modeling

To create the simulations that are stored in the database, we use UCLCHEM combined with RADEX as the chemical model and radiative transfer code, respectively. Physical parameters describing the gas conditions are passed to UCLCHEM to generate abundances, and then a subset of these values is passed to RADEX in order to obtain a list of transition lines. Beyond the outputs from UCLCHEM, additional free parameters are required for RADEX, such as the line width. A detailed flowchart of how the modeling tools interact with each other can be found in Figure 2 for clarity. The inputs for UCLCHEM and RADEX listed in the flowchart can be changed according to the needs of the inference to be run, but this requires changes to be made to configuration files.

By default, UCLCHEMCMC is configured to use RADEX for the radiative transfer calculations; we use the fractional abundance calculated by UCLCHEM to approximate the column density by using the visual extinction calculated from the given R_{out} and gas volume density. This value is used alongside the other inputs UCLCHEM was given, which RADEX needs as well, such as gas volume density and kinetic temperature, in order to run the radiative transfer model to produce observables that can be compared to the data. The observable values produced by RADEX are the radiation peak temperature (T_{R}) in K, which is comparable to the measured line intensity; the integrated surface brightness in K km s^{-1} ; and the isotopic flux emitted in all directions in $\text{ergs}(\text{s cm}^2)^{-1}$ (van der Tak et al. 2007). Any of these can be sent to the likelihood function depending on the user’s data.

Both UCLCHEM and RADEX can be replaced because UCLCHEMCMC is only designed to perform an MCMC inference and manage an SQL database. As long as inputs and outputs are carefully tailored to a given project, the code could be simply modified to be used with any chemical modeling or radiative transfer codes. To begin, we use a limited list of chemical species whose collisional data are available in the Leiden Atomic and Molecular Database (LAMDA; Schöier et al. 2005), as RADEX requires such data.

2.2. MCMC Inference

For the MCMC inference, UCLCHEMCMC uses the Python package *emcee* (Foreman-Mackey et al. 2013) in order to calculate the posterior probability density function (PDF) of the physical parameters for the desired observation. We assume the errors on the data are Gaussian and that our model provides the true intensities for any given parameters. The initially

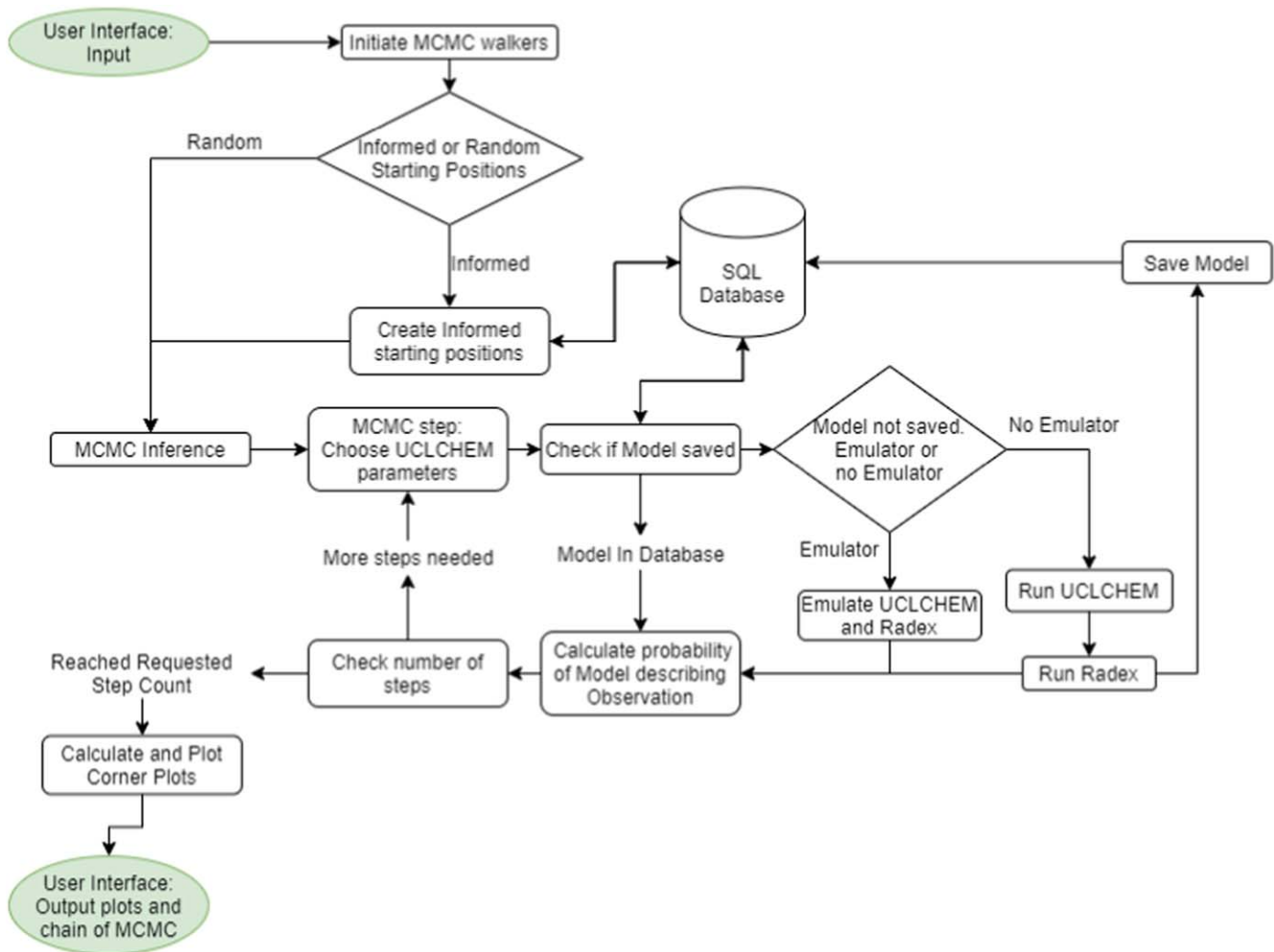


Figure 1. Flowchart of the various processes that happen in UCLCHEMCMC. Green ovals indicate parts with which the user interfaces. Diamonds indicate the parts of UCLCHEMCMC in which the next step is dependent on options specified by the user. The SQL database is represented with a cylinder and has been labeled this way for clarity. Arrows to and from the database represent a query of the SQL that then returns the models that match the query.

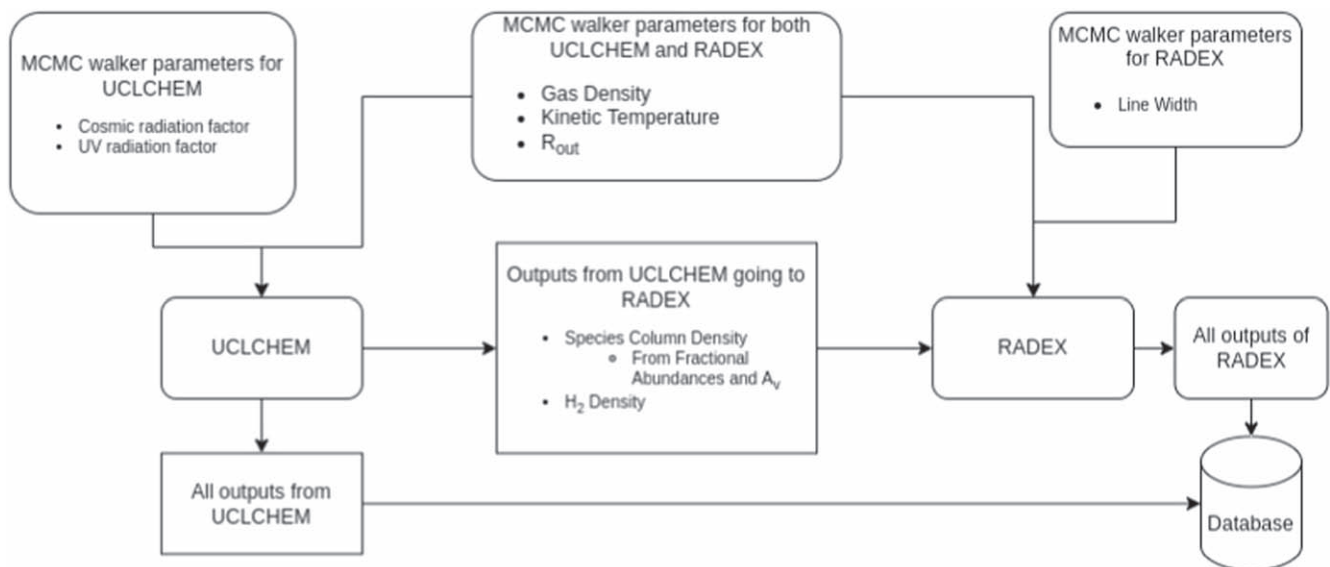


Figure 2. Small flowchart showing the parameters that enter UCLCHEM and the parameters that are taken from UCLCHEM and given to RADEX.

configured likelihood of observing our data given some parameters that were used for the example cases in Section 3 is therefore

$$\mathcal{L}(d|\theta) = \exp \left[-\frac{1}{2} \sum_i \frac{(d_i - \theta_i)^2}{\sigma_{d_i}^2} \right], \quad (1)$$

where d_i represents the observed input, θ_i is the output from RADEX, using physical parameters from UCLCHEM, and σ_{d_i} is the error in the observed input, each for line i . This is then combined with a prior on the physical parameters, $P(\theta)$ and the Bayesian evidence, $P(d)$, in the Bayes theorem to derive the full PDF in the form of

$$P(\theta|d) = \frac{P(\theta) \mathcal{L}(d|\theta)}{P(d)}. \quad (2)$$

By default, the prior is a uniform top-hat function in grid space on the ranges designated by the end user, but it can be altered by end users. As the prior is applied in grid space, it will be a log uniform prior if the physical parameter has a log-spaced grid applied to it. The evidence is treated as a normalization factor, and the values that are used for our example application on a prestellar core are discussed in Section 3. This approach to parameter inference has been used before (Holdship et al. 2019), and UCLCHEMCMC makes such inference problems simple.

2.3. Database

The core of UCLCHEMCMC is managing a database of models and running an MCMC inference that is supplemented by this database. After giving the inputs, which are detailed in Section 2.4, UCLCHEMCMC initiates a string of operations in order to start calculating the posterior PDF of the physical parameter values of an object using an MCMC sampler. A detailed flowchart of the processes can be seen in Figure 1. The code starts by initiating the walkers that will be used for the MCMC inference. If informed starting positions were requested, the SQL database will be searched for models that are similar to the given observations based on a simple top-hat function with a configurable distance on either side of the observed intensities. By searching and retrieving the parameters of models that have intensities similar to the given observation, we can construct a function by calculating the mean and standard deviation for each parameter to produce a normal distribution from which we can sample starting positions. Each parameter will have its own distribution from which the starting positions for the MCMC walkers are sampled. If random starting positions are chosen, then a uniform distribution of the parameter space is sampled in order to create the starting positions for each walker. Upon creating the starting positions, we then invoke the MCMC inference that will need to be able to access the SQL database, as described in Section 2.3.

For the sake of storing the inputs and outputs from UCLCHEM and RADEX, we use an SQL database using the SQLite implementation. This is chosen as it is a light-weight, widely used, and easy to implement solution for storing large volumes of data in such a way that they can easily be queried. The main advantage of having access to an SQL search method is that it can quickly check if the combination of parameters to be calculated has been previously stored. If it has, then the

program goes directly to evaluating the likelihood using Equation (1), which takes an almost negligible amount of time to perform. If the combination of parameters is not in the database, then a model can be created and stored for that set of parameters. This means that the calculation speed of the MCMC inference is dependent not only on the number of walkers and desired steps, but also on how many models are stored in the database.

Based on this, UCLCHEMCMC will become faster as more inferences are performed. The calculation of a single model can take around one minute when using UCLCHEM and RADEX. While this can be parallelized, it is still a limiting factor when thousands of models have to be calculated to obtain a reasonable estimation of a PDF. In contrast, the action of submitting a query to an SQL database and evaluating the probability of the stored models matching the observations takes less than a second. Improving efficiency in this way has the advantage over techniques such as emulation (de Mijolla et al. 2019) because no approximation is made. To quantify the improvement, we measure the time it took 10 walkers to perform 100 steps at three different times: (i) when no database is being used, (ii) when about half of the models that the inference wishes to use are retrieved from the database, and (iii) when nearly all models used by the inference are retrieved from the database. We use this type of measurement for the performance because the minimum time, the time when every model can be retrieved from the database, should be identical regardless of the chemical and radiative transfer model that is used. For the three cases, the mean time and standard deviation are (i) $5185.33 \text{ s} \pm 1041.96 \text{ s}$, (ii) $4834.67 \text{ s} \pm 843.24 \text{ s}$, and (iii) $68.89 \text{ s} \pm 45.39 \text{ s}$. We emphasize that the times found for cases (i) and (ii) are strongly dependent on the chemical and radiative transfer models that are used, while case (iii) should only be weakly dependent on which models are used, with the dependence disappearing if all models the inference wishes to use are within the database.

The database can be accessed both by the code and by a user who wishes to use the models stored within it for other purposes. At the time of the first release, we store all inputs that are given to the chemical model when it is run, as well as all outputs that are produced by it. This can include the output of intermediate time steps that UCLCHEMCMC can be set up to store if requested. UCLCHEMCMC then takes the given line width, either as a free parameter of the MCMC or as a constant value given by the user, as well as the kinetic temperature, volume density of H_2 , and the fractional abundances of the atomic or molecular species from the chemical code in order to run the radiative transfer code. The outputs from this code are then stored in the SQL database. From here, the emission lines given by the radiative transfer code can be compared to the observations to evaluate the likelihood, as discussed previously.

2.4. Interface

The user interface (UI) for UCLCHEMCMC is browser based, in order to give a simple usable interface that should be compatible with most operating systems. A further advantage of this is that an online, publicly available version can be more easily created in the future, such that end users will not need to change the workflow.

The inputs that are requested from a user of UCLCHEMCMC are separated onto three pages within the UI, a

Table 1
Inputs and Options per Page

Page	Input/Option	Description
Parameter input (Page1)	Final volume density [cm^3]	Hydrogen volume density at which the model stops collapsing
	Kinetic temperature [K]	Kinetic temperature of the gas
	CR ionization rate	Multiplicative factor of the galactic rate of ionisation caused by CR ($1.3 \times 10^{-17} \text{ s}^{-1}$)
	UV radiation field strength	Strength of the external UV radiation field strength acting on the cloud
	R_{out} [pc]	Radius of the modeled cloud
	Line width [km s^{-1}]	RADEX line width of observation
Observation input (Page2)	Species list	List of the species that have been configured
	Transition list	List of transition lines that have been configured for a given species
	Observation inputs	Space to fill in the observations, errors, and choice of units for the observation
Options (Page3)	Grid type	Choice on whether to use coarse or fine grid for the parameters being inferred
	Informed starting position	Choice on whether to use informed starting positions or random starting positions
	Session name	Back end session name to allow the session to be reloaded later
	Number of walkers	Number of walkers the MCMC should use (It is recommended by the package emcee to use twice as many walkers as parameters being inferred)
	Number of steps	choice of the number of steps an inference should take before stopping and loading evaluation corner plots
Inference (Page4)	Start inference	to start the MCMC inference or continue it if a previous session is loaded
	MCMC corner plot	MCMC corner plots of previous steps, if previous steps exist, or the starting positions if a new inference is being started

Note. Options of inputs and outputs per page for UCLCHEMCMC.

brief summary of each page can be found in Table 1. The first page requests the ranges of the physical parameters over which the inference should be performed. At the time of the first release, the configured parameters are (i) the volume density of the gas in cm^{-3} for the point at which the model should stop collapsing, (ii) the kinetic temperature of the gas in K, (iii) the cosmic ray (CR) ionization rate in units of the galactic CR ionization rate (ζ_0), (iv) the UV radiation field strength in units of Habing, (v) the radius of the assumed spherical cloud being modeled in parsec (R_{out}), and (vi) the line width to be used with RADEX in units of km s^{-1} . Upon supplying the desired ranges and the parameters that should be kept constant, the next set of inputs is the observations. Here, a list of species can be selected to be added to the current inference. Once a species has been selected, the compatible lines will be shown and can be selected, after which it is possible to add the values of the observations, errors, and the observed quantity. As of the first

Table 2
Parameter Ranges

Parameter [Units]	Lower Bound	Upper Bound
Volume density [cm^{-3}]	5.0×10^4	1.0×10^7
Kinetic temperature [K]	5	20
UV radiation field [Habing]	0.1	10
R_{out} [pc]	0.0001	0.1
CR ionization rate [$1.3 \times 10^{-17} \text{ s}^{-1}$]	0.1	10

Note. Physical parameter range the inference is allowed to explore for both the mock and observational inference.

release, UCLCHEMCMC is configured to allow for the units that RADEX has as outputs, detailed in Section 2.1.

The penultimate page contains the options for the MCMC inference. The options are the MCMC details, walker starting positions, and grid type. There are three options for the MCMC

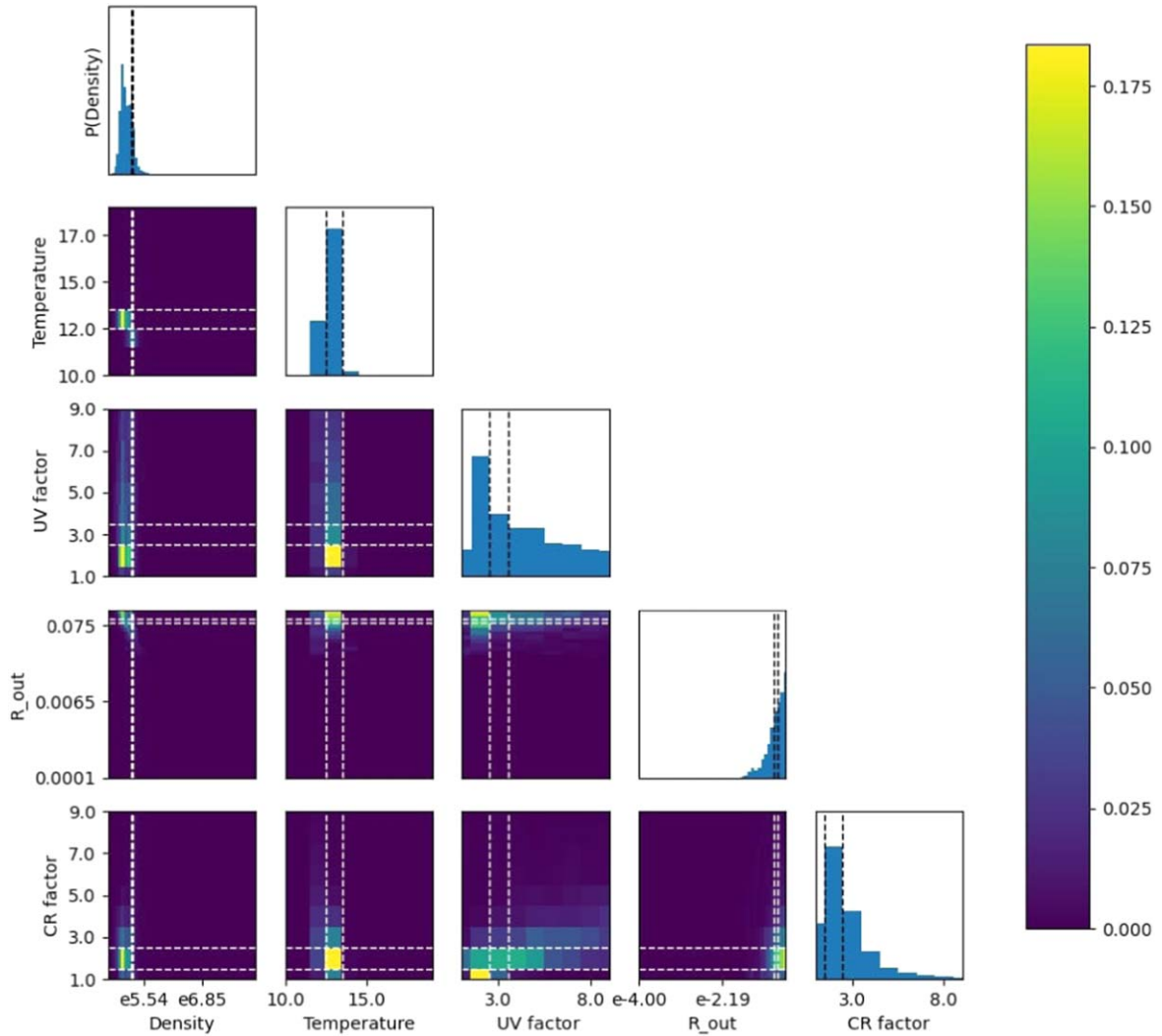


Figure 3. Posterior distribution function of the evaluation run performed on mock data. The histograms represent the PDF of volume density, kinetic temperature, UV field factor, R_{out} and the CR ionization rate factor. The color bar shows the value ranges of the joint distribution functions. The dashed white lines in the joint distributions and the dashed black lines in the PDFs represent the true value used to create the mock data.

Table 3
Mock Data Used for the Evaluation

Species	Transition	Frequency (GHz)	Line Width (km s^{-1})	RADEX Value	Mock Data (T_{MB})	Units
CS	2, 0–1, 0	97.98095	1.0	2790.9	2412.4 ± 558.2	mK
SO	2, 2–1, 1	86.09395	1.0	1918.6	2553.0 ± 383.7	mK
	2, 3–1, 2	99.29987	1.0	450.3	367.6 ± 90.0	mK
o-H ₂ CS	3, 1–2, 1	109.2522	1.0	185.5	222.3 ± 37.1	mK
	3 _{1,3} – 2 _{1,2}	101.4778	1.0	130.6	151.6 ± 26.1	mK
	3 _{1,2} – 2 _{1,1}	104.6170	1.0	85.5	83.8 ± 17	mK

Note. Values of the mock data created for evaluation of UCLCHEMCMC using UCLCHEM and RADEX. The RADEX value column contains the value given by RADEX, while the mock data column contains the same values with added Gaussian noise and the corresponding error values.

algorithm that an end user can easily change. They are (i) the number of walkers that the inference should have, (ii) the number of steps the inference should perform before saving, and (iii) the name of the session. Naming the session allows for an inference to be started again at a later time without having to reenter all the previous parameters and observations. This was added in case the code crashes, or if after evaluating the results it was determined that the MCMC walkers could benefit from more steps to ensure

that the walkers converged. The starting positions of the MCMC walkers can either be randomly determined or set to inform starting positions, depending on the preference of the end users. The details of how informed starting positions are calculated are given in Section 2.3. The grid type option allows an end user to choose the physical parameter space grid they wish to use for the inference they are going to run, and the options are intended to be created and managed by the end user. By default, a coarse and a

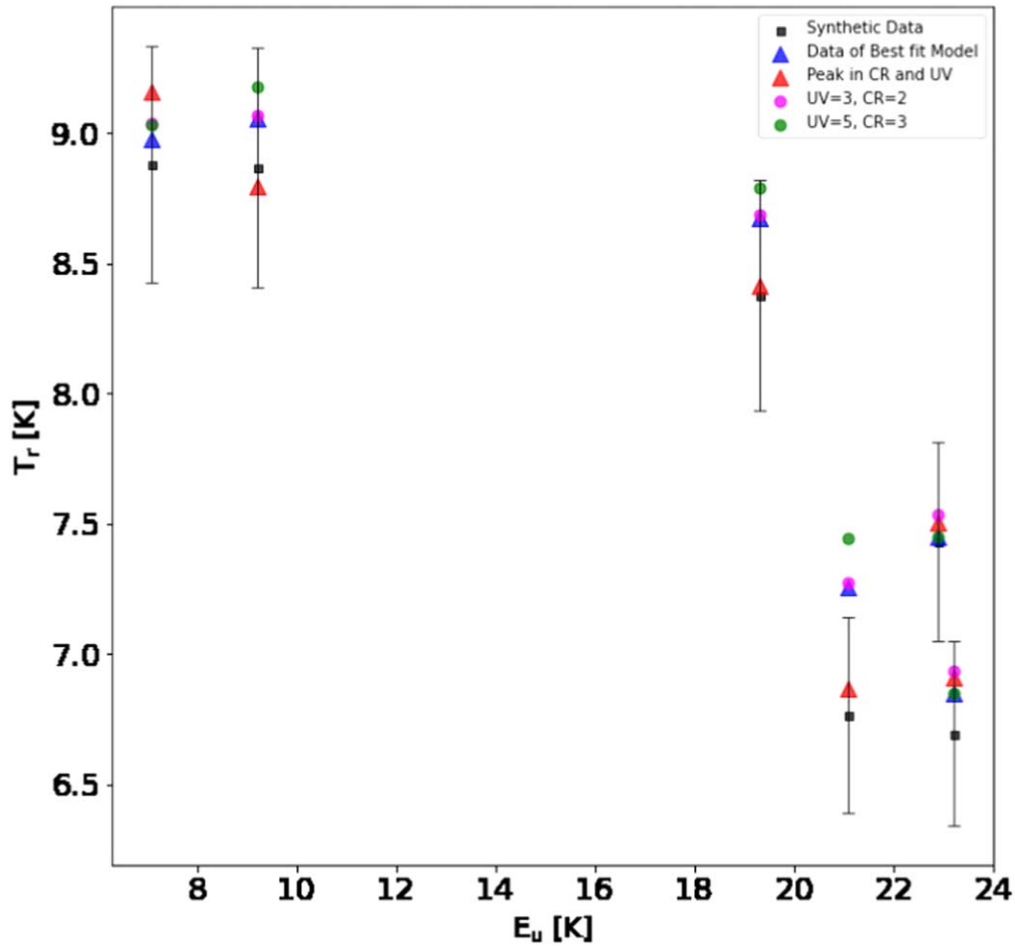


Figure 4. The radiation temperature, T_R , calculated by RADEX against the energy of the upper state for the mock data and errors given to UCLCHEMCMC are shown in black, and the data created when using the most likely parameter values from the 1D distributions from the inference of the mock data are plotted in blue. Red represents the peak in the joint distribution of the CR ionization rate and UV radiation field while keeping the remaining parameters as they are for the previous model, while green and fuchsia represent two additional points with values for the CR ionization rate and UV radiation field values in the elongated distribution of likely values to show why the inference still gave some importance to these values.

Table 4
Observations Used for the Evaluation

Species	Transition	Frequency (GHz)	Line Width (km s^{-1})	Observation (T_{mb})	Units
CS	2, 0–1, 0	97.98095	0.64 ± 0.07	1226.5 ± 0.1	mK
SO	2, 2–1, 1	86.09395	0.42 ± 0.01	223.7 ± 5.9	mK
	2, 3–1, 2	99.29987	0.45 ± 0.01	1422.5 ± 40.6	mK
	3, 1–2, 1	109.2522	0.39 ± 0.01	176.1 ± 5.2	mK
HCS+	2–1	85.34789	0.43 ± 0.01	246.8 ± 6.1	mK
OCS	6–5	72.97678	0.36 ± 0.01	106.3 ± 6.8	mK
	7–6	85.13910	0.38 ± 0.01	87.4 ± 7.2	mK
	8–7	97.30121	0.37 ± 0.01	70.5 ± 5.4	mK
	9–8	109.4631	0.34 ± 0.04	49.4 ± 6.4	mK
o-H ₂ CS	$3_{1,3} - 2_{1,2}$	101.4778	0.44 ± 0.01	558.4 ± 11.7	mK
	$3_{1,2} - 2_{1,1}$	104.6170	0.44 ± 0.01	514.3 ± 12.0	mK
p-H ₂ CS	$3_{0,3} - 2_{0,2}$	103.0405	0.45 ± 0.02	536.9 ± 16.5	mK

Note. Observations collected from Vastel et al. (2014) and Vastel et al. (2018). o- and p- represent the ortho- and para-version of species, respectively.

fine grid are provided to give an example of how they are meant to be created. The discretization of the parameter space to grids was implemented because chemical models with physical parameters that differ only to a small degree would produce nearly indistinguishable outputs, but would be considered separate models by the code, which retrieves and stores models in the SQL database. This would lead to models with minor differences in

parameters space being calculated despite producing indistinguishable outputs.

3. Application

In order to give an example of UCLCHEMCMC, we run three inferences. One inference is on mock data that were

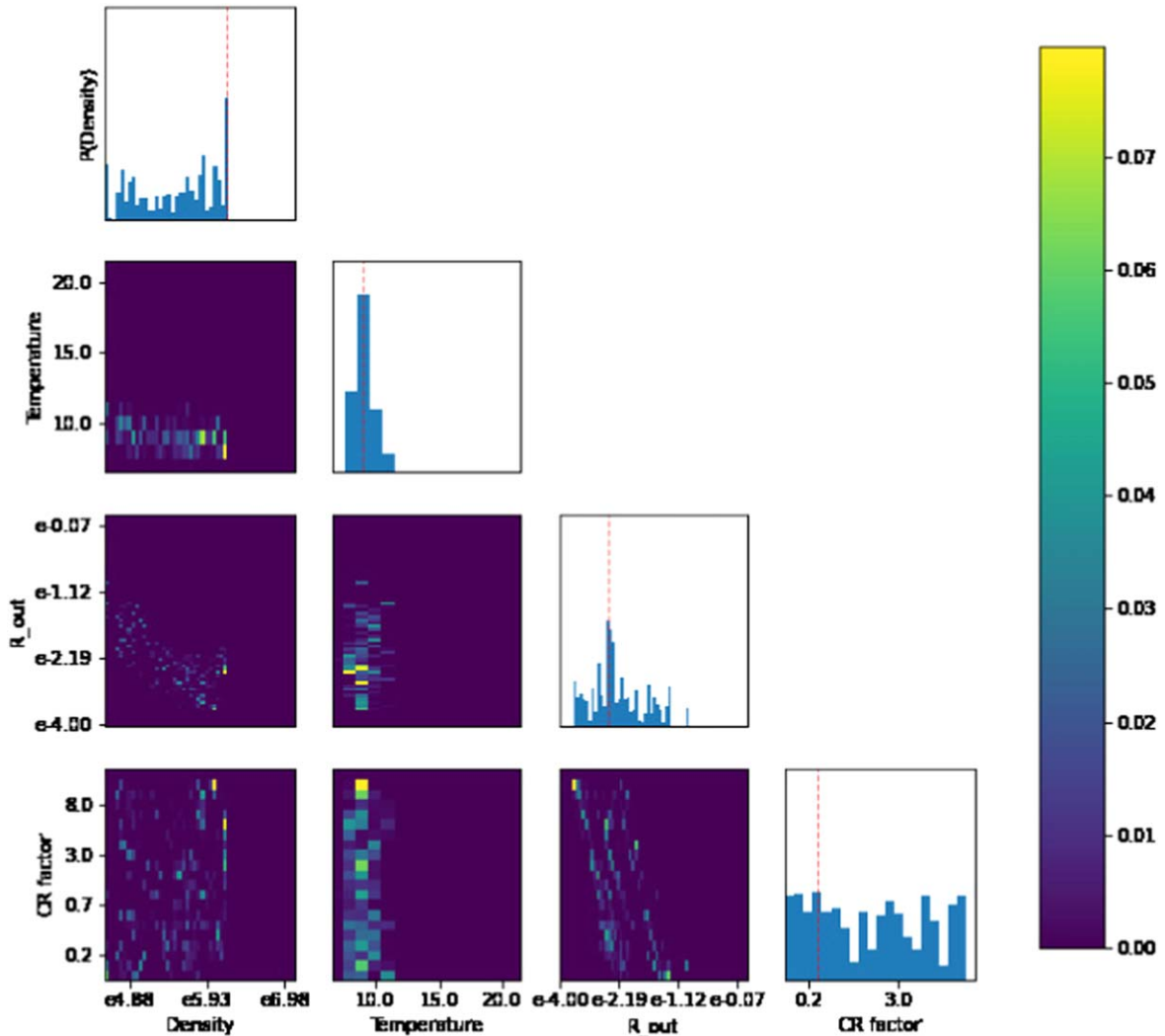


Figure 5. Posterior distribution function of the evaluation run performed on the emission lines from OCS only. The histograms represent the PDF of the volume density, kinetic temperature, R_{out} , and the CR ionization rate factor, the color bar shows the value ranges of the joint distribution functions, and the dashed red line in the PDF is the value with the highest probability.

created by using UCLCHEM and RADEX because these two codes are used in UCLCHEMCMC to perform the inference. The second and third inference are for the prestellar core L1544 (Caselli et al. 2002), once considering the emission from only one molecular species, and once with all sulfur-bearing species. The data we used for L1544 can be found in Table 4 and were used to infer the kinetic temperature, volume density, CR ionization rate, and R_{out} . This object is a very well-studied prestellar core located at R. A. = $05^{\text{h}}01^{\text{m}}11^{\text{s}}.0$, decl. = $25^{\circ}07'00''$ (Caselli et al. 2002; Vastel et al. 2014; Puanova et al. 2018 and Vastel et al. 2018).

We use the same input parameter space for all inferences. The exception is the UV radiation field, which we hold constant for the inferences on L1544 as we expect the visual extinction to be sufficiently high for changes in the UV to be negligible. The ranges for the physical parameters can be found in Table 2.

3.1. Mock Data Inference

First, we verify that UCLCHEMCMC performs as intended by creating mock data using the same modeling codes that

UCLCHEMCMC uses to perform an inference. We add Gaussian noise to the data with a standard deviation of 5% for each emission line because this would make the mock data errors equal to but slightly higher than the average of the uncertainties on the L1544 observational data. We do this because running an inference where the true values are known and the data are model generated allows us to test whether UCLCHEMCMC performs as intended when the models are appropriate for the data. As this is just an example case and many different combinations of chemicals and transition lines could be picked, we choose a subset of emission lines from the observations we use for the inferences on L1544. In order to create the mock data, we randomly chose physical parameters that resulted in all emission lines having an observable flux. The parameters we chose are as follows: a final volume density $1.0 \times 10^5 \text{ [cm}^{-3}\text{]}$, a kinetic gas temperature of 13 [K], a radiation field of 3 Habing, a cloud radius of 0.08 pc, and a CR ionization rate value of $2.6 \times 10^{17} \text{ s}^{-1}$. The emission lines and corresponding mock data values are found in Table 3, which contains the exact values of each line given by UCLCHEM and RADEX prior to adding noise, as well as the data with Gaussian noise added and the corresponding uncertainties.

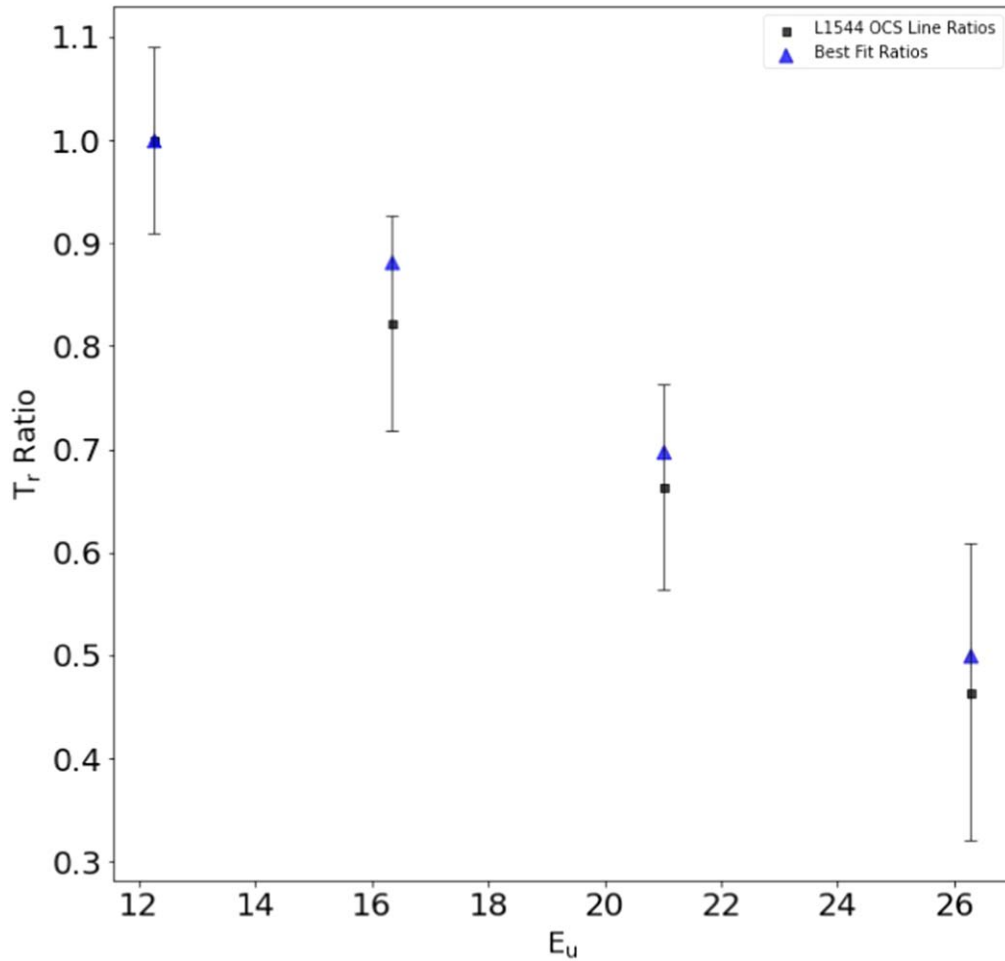


Figure 6. T_R over T_R of OCS 6-5 against the upper state energy for emission lines of OCS found in Table 4, compared to the data of the best-fit model after running an inference using only the OCS lines. All of the lines fit the observed line ratios quite well. Black represents the real data with error bars, and blue shows the best-fit model.

We run the inference, monitoring the chain of steps that each walker has taken. The likelihood of accepting a new set of parameters decreases as a function of the difference between the likelihood of the current model and the new model. This means that over time, the parameter space that is being traversed by all walkers will decrease as the walkers find areas of parameter space where the set of parameters produces models with a higher likelihood. Once the walkers stop reducing this parameter space, we stop the inference. Using these chains from the inference, we then create the posterior, shown in Figure 3. The distributions contain the true values, which indicates that UCLCHEMCMC works as expected. In order to validate this, we plot all mock observation lines against the upper state energy, seen in Figure 4, and do the same for the model values that are produced from UCLCHEMCMC’s parameters with the highest likelihoods in the 1D distributions. When we do this, we see that all lines but one lie within the uncertainties of the mock data.

We note that in the posteriors, there is a considerable degeneracy between the UV field and the CR ionization rates. Additionally, there is a clear peak in the joint distribution of the CR ionization rate and UV radiation field. This peak is at a CR ionization rate equal to the galactic value ($CR = 1$) and at a UV radiation field strength of 2 Habing, but this peak does not have the same value of the CR ionization rate as the parameter set

used to generate Figure 4 because this takes the most likely value from the 1D marginalized distributions rather than the overall most likely parameter set. To see how the emission lines change along this extended distribution and how the observations look at this peak in the joint distribution, we include the emission lines of this peak and two additional combinations of CR ionization rate and UV radiation field strength in Figure 4, while holding all other parameters constant. In looking at how the antenna temperature of the emission lines compare between the models and the mock data, it becomes quite clear that the values of the observations in this distribution all show significant agreement with the mock data, but that the peak in the CR ionization rate and UV radiation field strength distribution produces observations that fit better than the model created using the most likely parameter values in the 1D distribution.

3.2. Inferring the Physical Parameters of the Prestellar Core L1544

With verification of how well UCLCHEMCMC can perform when using mock data, we now use real observations in order to run another inference. The observations we use are from Vastel et al. (2018), who present observations for two distinct regions in the L1544 object. One region is a methanol shell that is around 8000 au (Vastel et al. 2014) from the core. In this

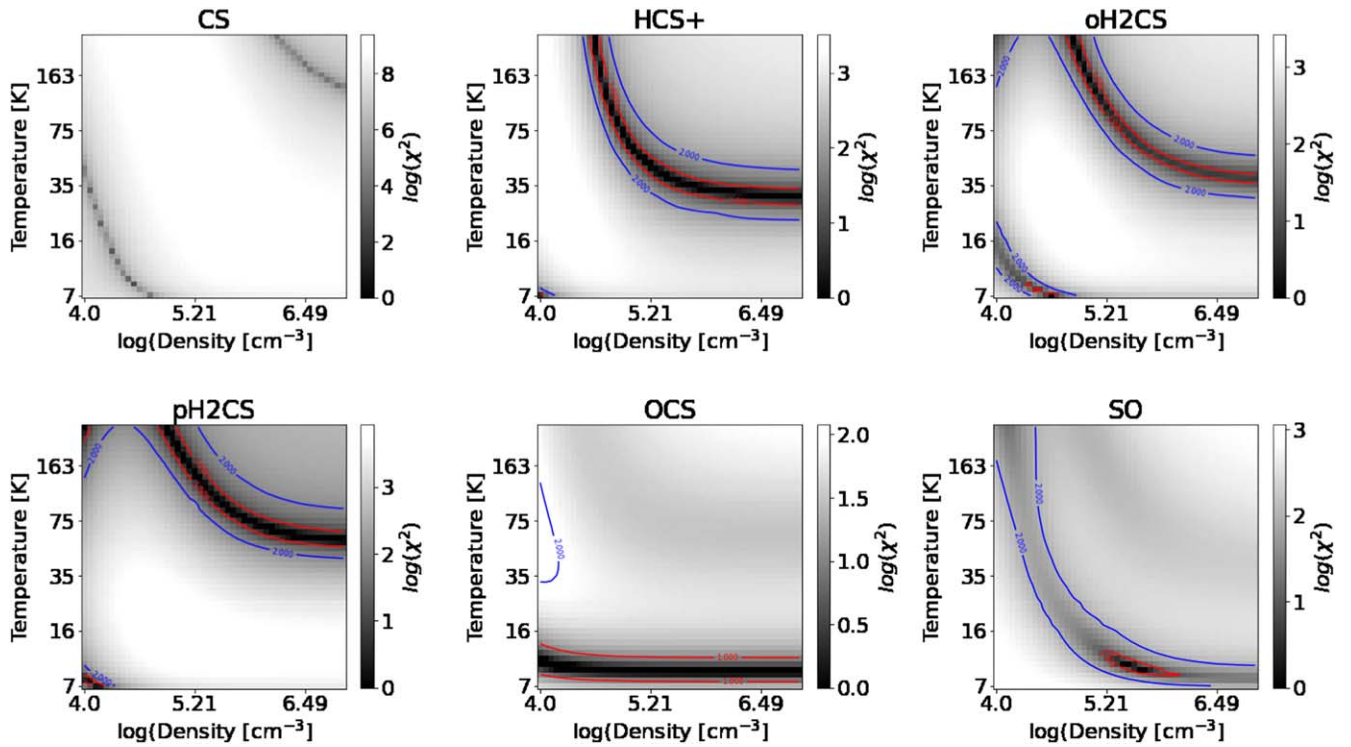


Figure 7. $\log(\chi^2)$ grid for the kinetic temperature and volume density using the column density from Vastel et al. (2018) for the six species that are used for the MCMC inference. The lower value of the $\log(\chi^2)$ is fixed at 0 to allow a better comparison between each species while allowing a flexible upper end, as the large ranges of $\log(\chi^2)$ values make it difficult to create an informative figure with a single range of values.

shell, UV photons can desorb methanol from the dust. The other region is a dust peak situated at the center of the object. We perform two inferences on the dust peak as there is a collection of sulfur-bearing species, where we start by using only the emission lines of OCS. In this first inference, we start with very broad priors and do not include additional information on the priors to show how UCLCHEMCMC can perform. In an inference that is not designed to showcase the performance, any prior knowledge would be used to inform the ranges of the priors. We then follow this with an inference using all sulfur-bearing species to serve as a test to inform us of how well of an inference can be performed when UCLCHEMCMC is given unfavorable conditions. More chemical species and emission lines are potentially unfavorable conditions because UCLCHEMCMC may struggle to find a single set of parameters that fits all lines at once. For a reference for the values we expect the inference to estimate, we turn to Vastel et al. (2018), who model the kinetic temperature and the volume density of the dust peak to be around 7 K and $2 \times 10^6 \text{ cm}^{-3}$, respectively.

As this is an example case of how to use UCLCHEMCMC, we leave a large physical parameter range for the volume density, R_{out} , and CR ionization rate value. We limit the kinetic temperature to be between 5 and 30 K as at this temperature, a single degree can make a difference to the diffusion and desorption rates of various species, impacting the fractional abundances of different species. The only two exceptions we make on limiting parameters is that we left the UV radiation field strength at the default value for UCLCHEM, as it is not a parameter of interest for the dust peak of a prestellar core, and we set the line width to the error-weighted mean of 0.37 km s^{-1} for the OCS-only inference.

We follow this by using all of the chemical lines found in Table 4 for a second inference of the dust peak. We intentionally use all of these lines as a stress test. For the second inference, we make the assumption that these species trace the same substructure, which we emphasize in the inference by setting all line widths to 1.0 km s^{-1} in UCLCHEMCMC. This could lead to an inference that is unable to fit the observations as UCLCHEMCMC could struggle to find one set of parameters that leads to emission lines that match the observations.

The posterior of the limited inference can be found in Figure 5. The distribution has a very broad range in volume density and in CR ionization rate value and a strong peak in kinetic temperature and a peaked area in R_{out} . This suggests that there is a wide range of possible volume density and CR ionization rate values that can describe the observations well.

In order to choose a good fit, we use previously modeled values of the total column density along with the fractional abundance of hydrogen to constrain the gas volume density. Caselli et al. (2002) modeled a total column density of $4.4 \times 10^{22} \text{ [cm}^{-2}\text{]}$, which we combine with the peak value of the R_{out} posterior to obtain a likely volume density of $10^6 \text{ [cm}^{-3}\text{]}$.

To validate that this is a good fit, we plot the emission line ratios with respect to the most intense line OCS 6-5 against the upper state energy to get Figure 6 to create a diagram analogous to a rotation diagram. This diagram shows that the modeled line ratios are within the estimated error bars of the observed line ratios, which supports the accuracy of the inference performed. A useful next step would be to remove the volume density as a free parameter, and use the measured column density to calculate it from R_{out} during another round

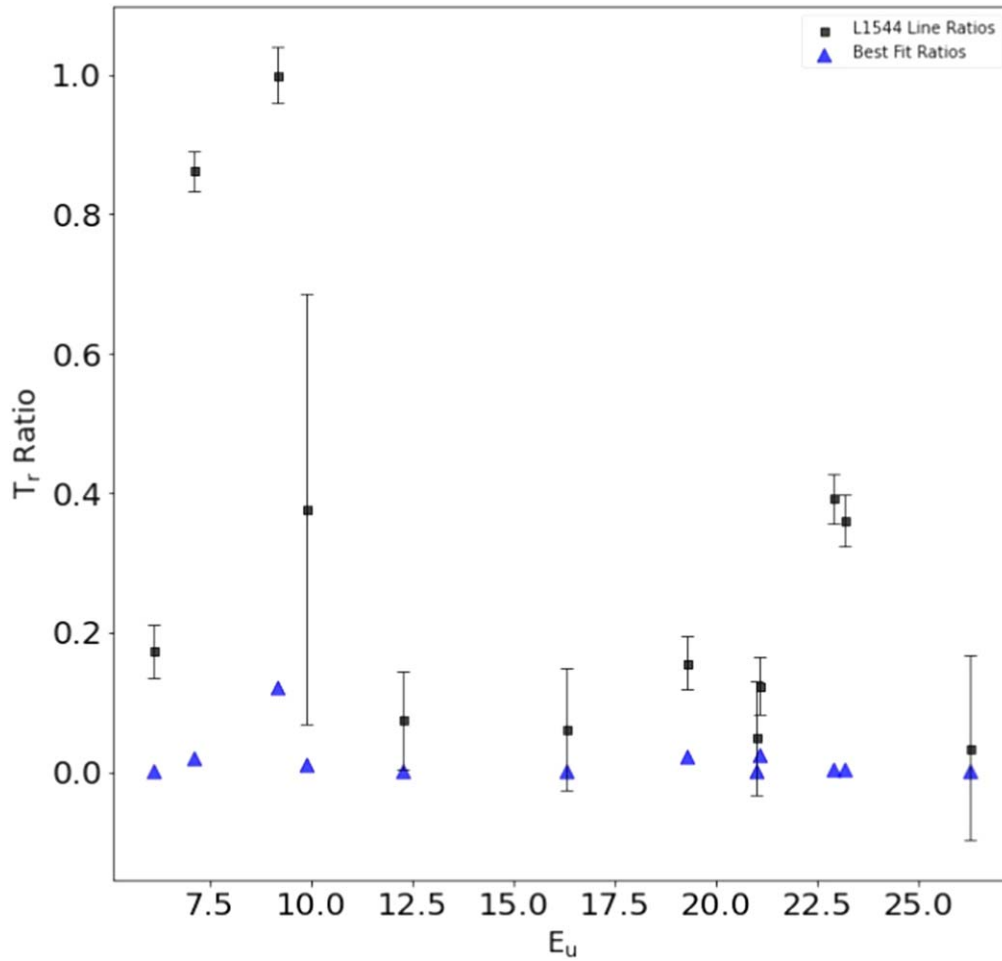


Figure 8. T_r against the energy of the upper state for all emission lines in Table 4, compared to the data of the best-fit model after running the stress-test inference. While some lines almost fit their observed counterparts, it is clear that UCLCHEMCMC is unable to match all lines at once, which caused it to settle for a set of parameters that allow each line to at least get somewhat close to the observations. Black represents the real data with error bars, and blue shows the best-fit model.

of inference, with a finer grid in parameter space. Since this is just an example case, we will instead move on to the stress test of UCLCHEMCMC.

Prior to running this test, we calculate a χ^2 fit of the volume density and kinetic temperature using only the radiative transfer code RADEX to serve as a baseline comparison to the performance of UCLCHEMCMC on the observed data, as this is a more common approach. To perform this fit, we take the column densities determined by Vastel et al. (2018) through radiative transfer modeling for each species in Table 4, and run RADEX on a grid of kinetic temperatures and volume densities. The results of this fit can be seen in Figure 7. The χ^2 fit is unable to find one set of parameters that agree with each other for all lines. It also accepts a very large area of parameter space as potential fits to each individual species, severely limiting how helpful this fit is to any modeling effort. In addition, this method requires that we either provide a column density estimate or that we include a grid of column densities over which to calculate, which would significantly increase the calculation time as it would be adding an additional dimension to the parameter space. We note, however, that the speed at which these calculations were performed is at least three orders of magnitude lower than a traditional MCMC inference that does not use an SQL database.

We perform the stress-test inference with the observational data by using all species and emission lines found in Table 4. As is the case with the χ^2 fit, the stress-test inference is unable to match all lines at once. The area onto which the MCMC inference converges is a delta-like distribution on a single set of parameters that we then use to model the emission lines. We again plot the emission line ratios against the upper state energy, this time with respect to the SO (2, 3)–(1, 2) line as it is the strongest line, resulting in Figure 8. In this figure it is clear that while one or two of the line ratios fit the observed ratios, the vast majority of lines from the best-fit model do not match the observations at all. This failure is expected as UCLCHEMCMC assumes that a simple homogeneous model should fit the observations. As more species and transitions are added, the assumption of a simple homogeneous model will be broken. We include this example of a failed fit to assist users of UCLCHEMCMC in understanding some of the limitations and more importantly, as a cautionary note when trying to fit multiple molecular transitions with one single gas component. In addition, it is important to analyze the best-fit models and not just assume that the posterior must be a good fit.

4. Summary



The publicly available MCMC inference and SQL database managing tool UCLCHEMCMC is capable of inferring

physical parameters of astrochemical observations. This paper presents the details necessary to understand the use, strengths, and shortcomings of this tool. The management of the database, using SQLite, increases the efficiency of parameter inference as the tool is used. Using the MCMC inference package emcee as well as decoupling the chemical code and radiative transfer code from the inference also makes UCLCHEMCMC capable of handling any other chemical modeling or radiative transfer modeling tool.

We showed the outputs of UCLCHEMCMC when inferring the physical parameters of mock data created using UCLCHEM and RADEX, detailing just how well this recovered the physical parameters of the mock data. The use of the SQL database in the inference has shown that when most of the models that the inference searches for are in the database, UCLCHEMCMC goes from taking 5185.33 ± 1041.96 s for 10 walkers to take 1000 steps to needing 68.89 ± 45.39 s, which is a significant decrease in computational time. When inferring the physical parameters of actual observations, we detailed some of the issues that must be taken into consideration when running this tool. Users should be aware that if they keep the physical parameter ranges too small, then the inference may not be able to find matching parameters, resulting in nonphysical answers. We intend to add a fast prior predictive checking functionality to UCLCHEMCMC. Additionally, giving too many emission lines without taking into consideration that they may arise from separate structures or that the combination of emission lines requires physical parameters outside of the inference range can cause UCLCHEMCMC to become unable to find physical parameters that match all lines. Because of this, we advise caution about using a long list of emission lines without first studying whether these lines come from regions within the object that have similar physical parameters, as this tool will assume that they all come from one structure with one set of physical parameters. When taking these factors into consideration, UCLCHEMCMC can be a great asset in inferring physical parameter ranges in which to start modeling astrochemical environments.

This project has received funding from the European Union’s Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No 811312 for the project “Astro-Chemical Origins” (ACO) as well as from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation program MOPPEX 833460.

ORCID iDs

Marcus Keil  <https://orcid.org/0000-0002-4948-7468>
 Serena Viti  <https://orcid.org/0000-0001-8504-8844>
 Jonathan Holdship  <https://orcid.org/0000-0003-4025-1552>

References

- Allodi, M., Baragiola, R., Baratta, G., et al. 2013, *SSRv*, **180**, 101
 Caselli, P., Hartquist, T. W., & Havnes, O. 1997, *A&A*, **322**, 296
 Caselli, P., Walmsley, C. M., Zucconi, A., et al. 2002, *ApJ*, **565**, 331
 de Mijolla, D., Viti, S., Holdship, J., Manolopoulou, I., & Yates, J. 2019, *A&A*, **630**, A117
 Foreman-Mackey, D., Hogg, D. W., Lang, D., & Goodman, J. 2013, *PASP*, **125**, 306
 Goodman, J., & Weare, J. 2010, *CAMCS*, **5**, 65
 Harada, N., Nishimura, Y., Watanabe, Y., et al. 2019, *ApJ*, **871**, 238
 Holdship, J., Viti, S., Codella, C., et al. 2019, *ApJ*, **880**, 138
 Holdship, J., Viti, S., Jiménez-Serra, I., Makrymallis, A., & Priestley, F. 2017, *AJ*, **154**, 38
 McElroy, D., Walsh, C., Markwick, A. J., et al. 2013, *A&A*, **550**, A36
 Nelson, B., Ford, E. B., & Payne, M. J. 2013, *ApJS*, **210**, 11
 Punanova, A., Caselli, P., Feng, S., et al. 2018, *ApJ*, **855**, 112
 Schöier, van der Tak, F. F. S., van Dishoeck, E. F., & Black, J. H. 2005, *A&A*, **432**, 369
 Taquet, V., Ceccarelli, C., & Kahane, C. 2012, *A&A*, **538**, A42
 ter Braak, C., & Vrugt, J. 2008, *Stat. Comp.*, **18**, 435
 van der Tak, F. F. S., Black, J. H., Schöier, F. L., Jansen, D. J., & van Dishoeck, E. F. 2007, *A&A*, **468**, 627
 Vastel, C., Ceccarelli, C., Lefloch, B., & Bachiller, R. 2014, *ApJL*, **795**, L2
 Vastel, C., Quénard, D., Le Gal, R., et al. 2018, *MNRAS*, **478**, 5514
 Viti, S. 2017, *A&A*, **607**, A118
 Wakelam, V., Herbst, E., Loison, J.-C., et al. 2012, *ApJS*, **199**, 21