

Article

OpenToolFlux: An Open-Source Software for Estimating Gas Fluxes from Automatic Chamber Data

Carmen Galea ^{1,2,*} , Alberto Sanz-Cobeña ^{1,2} , Luis Lassaletta ^{1,2}, Alba Monistrol ^{1,2}, Antonio Vallejo ^{1,2} and Rasmus Einarsson ^{1,3,*} 

¹ Departamento de Química y Tecnología de Alimentos, Escuela Técnica Superior de Ingeniería Agronómica, Alimentaria y de Biosistemas, Universidad Politécnica de Madrid, Ciudad Universitaria, 28040 Madrid, Spain; a.sanz@upm.es (A.S.-C.); luis.lassaletta@upm.es (L.L.); a.marcas@upm.es (A.M.); antonio.vallejo@upm.es (A.V.)

² Centro de Estudios e Investigación Para la Gestión de Riesgos Agrarios y Medioambientales (CEIGRAM), Universidad Politécnica de Madrid, 28040 Madrid, Spain

³ Department of Energy and Technology, Swedish University of Agricultural Science, Lennart Hjelm's väg 9, 750 07 Uppsala, Sweden

* Correspondence: authors: c.galea@upm.es (C.G.); rasmus.einarsson@slu.se (R.E.)

Abstract: OpenToolFlux 1.0 is an open-source software designed to estimate soil gas fluxes from gas concentration time-series data generated by automatic chamber systems. This paper describes the physical equipment used as well as the software design and workflow. The software is a command-line application that imports tabular time-series data from the analyzer following the instructions specified in a configuration file by the user, performs configurable data-cleaning operations, and outputs a data file with volumetric flux estimates as well as diagnostic plots. The software can be configured according to the specifics of the physical equipment and experimental setups, and it is, therefore, applicable to a wide range of studies.

Keywords: software; gas flux estimation; data processing; automatic chamber; greenhouse gases



Citation: Galea, C.; Sanz-Cobeña, A.; Lassaletta, L.; Monistrol, A.; Vallejo, A.; Einarsson, R. OpenToolFlux: An Open-Source Software for Estimating Gas Fluxes from Automatic Chamber Data. *Agronomy* **2023**, *13*, 2842. <https://doi.org/10.3390/agronomy13112842>

Academic Editor: Xiaobo Qin

Received: 22 September 2023

Revised: 18 October 2023

Accepted: 15 November 2023

Published: 18 November 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Emissions of greenhouse gases (GHG) from soils are an important contributor to climate change [1–3], with the food system responsible for a third of the total anthropogenic GHG emissions [4] in the form of N₂O, CH₄, and CO₂. Despite considerable empirical research on soil GHG emissions, there are still major uncertainties about the dynamics and processes that occur in the soil between its different components. These emissions are the result of microbiological processes strongly dependent on soil biogeochemical conditions [5,6], such as N and C availability [7,8] and soil water content [9,10], as well as other environmental and anthropogenic factors such as weather conditions, topography, land use, and vegetation [11,12]. The combination of these factors results in high temporal and spatial variability, usually with hot spots and hot moments regarding fluxes, with minimum and maximum values strongly fluctuating during the day depending on the area [13,14]. This spatial and temporal variability complicates accurate field-scale estimations and adds uncertainties depending on the measurement techniques used or the design of the experiment. This leads to imprecise empirical estimations when upscaled to a global scale (e.g., global inventories) and could lead to ineffective policy strategies [15].

Automatic chambers are a useful technology to measure GHG fluxes that help reduce some of the uncertainties compared to low-frequency methods such as static manual chambers. Automatic chamber systems take measurements continuously over a given period of time, making it possible to capture temporal variability and fluctuations, resolving an important source of uncertainty and improving data collection [13,16]. Although static manual chambers are the most common methodology with economic and logistic

advantages (since they are cost-effective and easy to use) compared to automatic systems, they also present limitations that can lead to under- or overestimations when measuring GHG. Unlike automatic chambers, manual chambers typically are used to sample once a day at most [15–17]. In contrast, automated systems with a higher frequency sampling allow one to address the response of GHG emissions to weather conditions, rainfall events, irrigation, and N fertilization [13,18], and to capture diurnal fluctuations [19], which is not practically possible with manual chambers. The use of automatic systems could, therefore, optimize the estimation of GHG emissions, thereby contributing to improving mitigation strategies and offering greater and more precise insight into the dynamics that rule GHG emissions [20,21]. Currently, a wide range of technologies that measure automatically and on a continuous basis have been developed and implemented and are widely used for exhaustive studies. Some of the most commonly used and tested in the literature are infrared gas analyzers and trace gas analyzers [17], which are useful when measuring CO₂ emissions and are usually also combined with a cavity ring-down spectrometer [22,23]. Another type of approach not covered here is micrometeorological technologies such as eddy covariance stations, which are also convenient for estimating, e.g., CO₂ in open fields and at higher spatial resolution [24].

One of the challenges in estimating gas fluxes using automatic chambers is the conversion of large time series of concentration values into estimated fluxes [17,22,25]. The raw data outputs from the gas analyzers in automatic systems are concentration time series that need to be transformed in several steps. Before the flux can be estimated, these time series may need to be cleaned, filtered, and cut into segments corresponding to single measurements and excluding failed measurements. The conversion to gas flux depends on the equipment characteristics, which can be very different depending on the user and will determine the complexity of the transformation. Chamber size, shape, and number; gas flow rate to the analyzer; and proximity of the chambers to the analyzer all directly influence the conversion. To address this challenge, we built flexible software comprising the key steps required to transform such raw data into numerical and graphical gas flux estimations.

In this paper, we present OpenToolFlux, open-source software created to process data generated by automatic chamber systems and transform them into gas fluxes (<https://github.com/rasmuse/opentoolflux> accessed on 10 September 2023). In the following sections, we introduce OpenToolFlux, present the type of physical equipment for which we designed the software, and describe its main components and workflow. To provide a better picture of how the software functions, an experimental example is shown to illustrate its potential uses for high-frequency GHG measurements and to underline the potential adaptations to similar systems as well as the limitations it may present. In addition, a complete example of configuration files, input data, and output data is published along with the software.

In summary, this article aims to describe a methodology for processing the data series of gas concentrations in automated chambers and transforming them into volumetric flows. It provides a purely illustrative description of the equipment used, without intending to delve into the details of the equipment or experimental design.

2. Materials and Methods

OpenToolFlux is an open-source software designed to estimate gas fluxes from soil using time-series data from automatic chambers. It is built in the Python 3.8 programming language and works in Python 3.8 and later versions. It was initially constructed to analyze data from the Picarro brand of equipment (details below) but is designed with flexibility so as to accept data from other equipment producing similar raw data.

2.1. Overview of Physical Equipment and Measurement Data

OpenToolFlux was developed to process data from an automatic system formed by three main components: (1) chambers that open and close; (2) a two-part multiplexer

sequentially opening and closing chambers while redirecting a gas stream from the currently closed chamber to the gas analyzer; and (3) the gas analyzer, connected to a computer, continuously measuring gas concentrations and saving the output.

OpenToolFlux assumes a setup without recirculation of the sampled gas. Sampled gas is drawn from the chamber to the analyzer while the air from the outside enters passively through a small inlet tube that maintains pressure equilibrium between the chamber and ambient air.

In our specific setup, the gas analyzer is a Picarro G2308 (Picarro, Santa Clara, CA, USA) Gas Concentration Analyzer, a cavity ring-down spectroscopy (CDRS) instrument, with a sampling frequency of around 1 Hz. This analyzer measures concentrations of N_2O , CH_4 , and CO_2 in volumetric parts per million (ppmv). The analyzer has built-in water correction software so that concentrations are automatically reported on a dry gas basis. In our case, the chambers are opened and closed using pneumatic actuators connected to an air pump through the multiplexer. Figure 1 shows all the equipment and how all the parts are assembled.

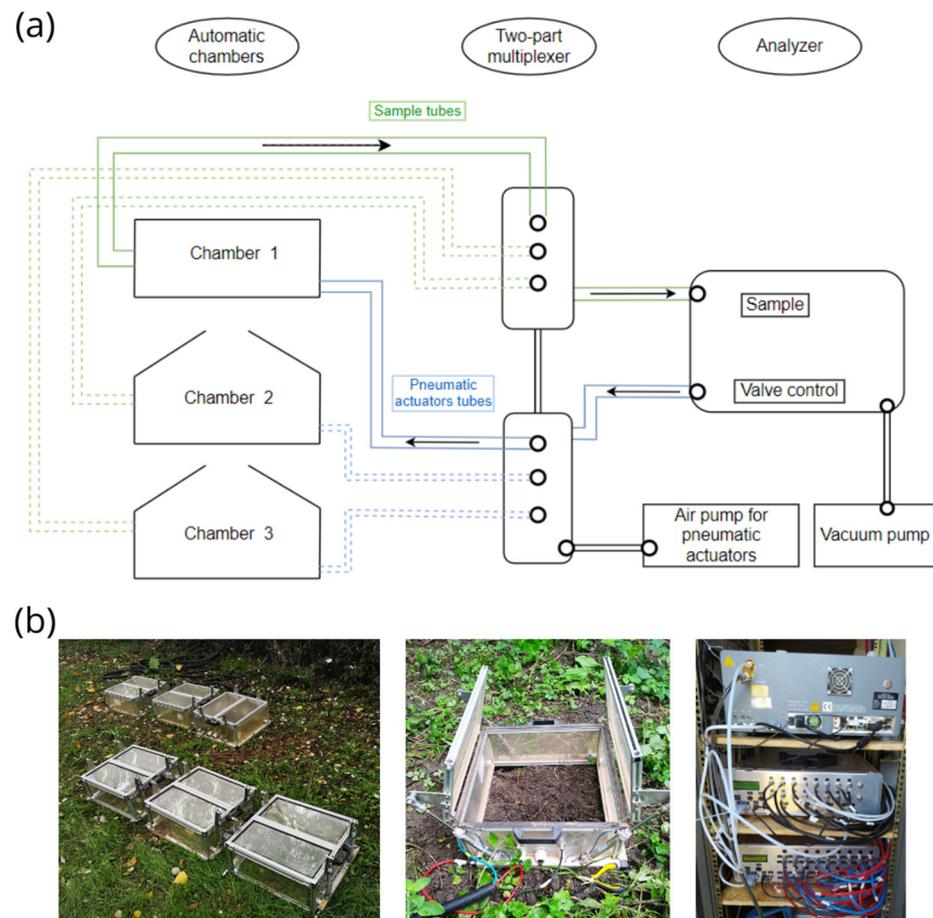


Figure 1. (a) Schematic view of the setup. Chambers are connected to a two-part multiplexer that carries the sample to the analyzer and opens and closes the chambers. A vacuum pump provides the analyzer with the pressure and temperature required for it to start measuring. Arrows indicate the direction of the air flow. (b) Photograph of the chambers closed and open, and the analyzer with the two-part multiplexer and all the chambers connected. Source: own elaboration.

The sampling rate and gas species are in principle irrelevant, as long as the gas analyzer has a sufficiently high sampling frequency to allow fitting a concentration curve over time during the closure. OpenToolFlux also assumes that the concentrations are reported in volumetric fractions (e.g., ppmv) and that water content corrections or similar corrections

have been made if necessary. The software further assumes a multiple-chamber system with the chambers opening and closing one at a time, in any order, and that one column of the tabular input data to OpenToolFlux indicates the currently active chamber, i.e., the chamber currently closed and from which gas is sampled. Moreover, OpenToolFlux is based on the assumption of the no-recirculation configuration as described above. The following sections define a mathematical model of the system and specify in greater detail the input data needed by the software.

2.2. Flux Calculation (from Concentrations to Fluxes)

2.2.1. Assumptions and Known Parameters

The automatic system used and the conditions of the experiment determine the method used to estimate gas fluxes from soil. This section describes a method based on the assumptions determined by our context.

Since N_2O is rather nonreactive in the presence of other air constituents and with the chamber materials, and the time scale of the measurement is short (in our case ca. 20 min), the gases are assumed to be chemically unreactive. Consequently, the concentration of the gas that goes into the analyzer equals the concentration in the chamber, adjusted only for the time delay introduced by the tube between the chamber and the analyzer.

Depending on the dimensions of the chambers, and the dimension and depth of the frames inserted in the ground, we calculate a known total volume (V) and a known area that the chamber occupies on the soil (A). The unknown soil gas flux being estimated (N_2O in our example) is assumed to be a constant volumetric net flux from the soil into the chamber (F). A negative flux is possible and would mean that the soil is a net sink for the given gas.

During chamber closure, a constant and known volumetric gas flow (Q) is pumped from the chamber to the analyzer. While the chamber is closed, we assume a constant volumetric flow (Q') of ambient air into the chamber through the inlet vent, replacing the sampled outflow. The soil may also have a net volumetric exchange with the chamber, but the net size of these exchanges is very small compared to Q , so that $Q' \approx Q$. In other words, almost all the air drawn into the chamber to replace the sample comes from the ambient atmosphere.

When the chamber is open, ambient air mixes into the chamber until the concentration of the gas being analyzed (e.g., N_2O) inside the chamber equals the constant ambient air concentration (c_0). When the chamber closes the next time, the initial concentration is therefore c_0 . During chamber closure, the concentration starts to change as a function of time, $c(t)$, due to the mixing of ambient air inflow and the net flux from the soil. We assume here that the air in the chamber is mixed well during closure so that the concentration $c(t)$ is uniform in the chamber (Figure 2). The parameters that need to be known in the estimation (see below) are A , Q , and V . To later convert volumetric fluxes to molar fluxes, the pressure P and temperature T are needed, but the software does not make this transformation.

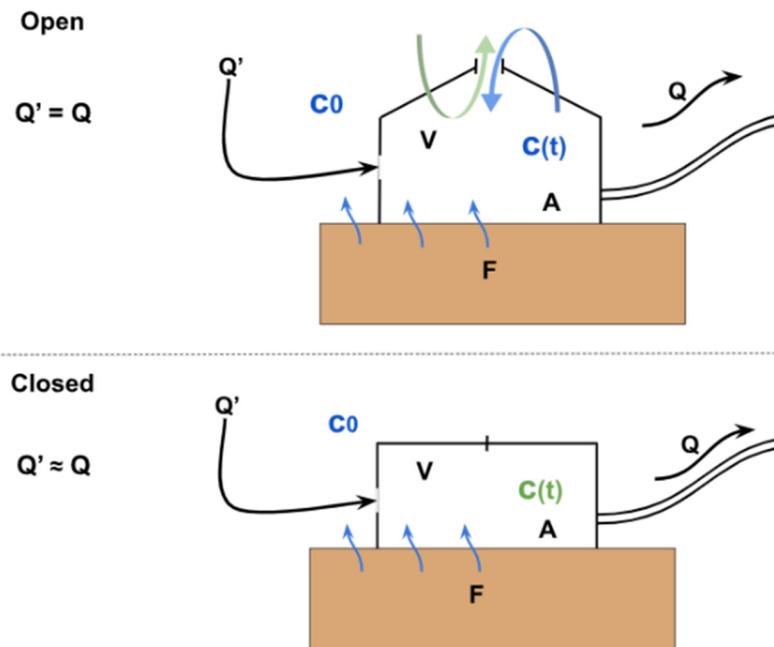


Figure 2. Schematic representation of the dynamics in the chambers showing the main parameters that are used to calculate the fluxes. V = volume of the chamber, A = area of the soil occupied by the chamber, F = constant volumetric net flux from the soil into the chamber, Q = volumetric gas flow pumped from the chamber to the analyzer, Q' = volumetric gas flow entering the chamber, c_0 = constant ambient gas concentration, $c(t)$ = change of the gas concentration inside the chamber in time. Green arrow represents the air leaving the chamber and blue arrows represent the air entering the chamber.

2.2.2. Estimating the Flux from Concentration Time Series

The flux estimation method used in OpenToolFlux is the accumulation method, widely used for non-steady state methodology [26,27]. The principle of the calculation is to derive and solve a differential equation based on the above description and the mass conservation criterion input = output + accumulation, and then fit the parameters of the solution to the concentration time series.

The solution to the differential equation (see Supplementary Materials) gives the time evolution of the concentration $c(t)$ inside the chamber as follows:

$$c(t) = c_0 + F \frac{A}{Q} \left(1 - e^{-\frac{Q}{V}(t-t_0)} \right),$$

where t_0 is the time of closure. Note that at closure, the gas concentration inside the chamber is the same as in ambient air: $c(t_0) = c_0$.

Note that this can also be written as follows:

$$c(t) = b_0 + b_1 g(t)$$

where $b_0 = c_0$, $b_1 = F$, and

$$g(t) = \frac{A}{Q} \left(1 - e^{-\frac{Q}{V}(t-t_0)} \right)$$

Since $c(t)$ is the measurement data and $g(t)$ is calculated from known parameters, c_0 and F can now be estimated using a linear regression with $c(t)$ as the dependent variable and $g(t)$ as the independent variable. The fit is linear with respect to $g(t)$, but when results are back-transformed, the solution shows a slight curvature as we obtain an exponential model (Figure 3).

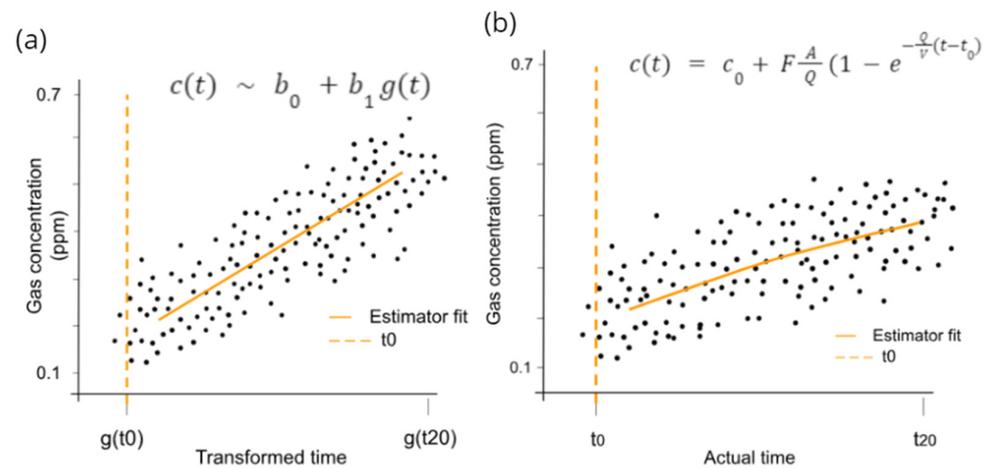


Figure 3. The flux F to be estimated can be found using a linear regression of concentration data $c(t)$ against a transformed time variable $g(t)$. Panel (a) shows the linear regression with transformed time and panel (b) shows the same result back-transformed to the time domain.

2.2.3. Adjusting Concentration Data for Tube Delay

So far, we have assumed that the time t_0 is a known parameter. However, knowing when the chamber is closed is not sufficient information since the tube between the chamber and the gas analyzer introduces a delay between chamber closure and gas arrival to the analyzer. The analyzer takes the sample from the chamber from the moment it closes, but due to the length of the tube that carries the sample, the beginning of a data segment labeled as belonging to a given chamber is still analyzing gas concentrations from gas that has been stagnant in the tube system and thus should be discarded in the data analysis. This delay can be substantial depending on tube length. After this tube delay time, the concentration values rapidly change, corresponding to the first gas sampled from the present chamber just after closure. This is what should be taken as t_0 in the data analysis when analyzing a time series collected during the closure of one chamber; therefore, OpenToolFlux discards data corresponding to the tube delay as estimated by the user (6 min in the example), plus a margin (e.g., 2 min) in case the delay is underestimated (Figure 4). The software allows setting the tube delay equal for all chambers or different for each one, corresponding to different tube lengths. The remaining part of the data must be long enough to fit the regression model corresponding to the chamber studied.

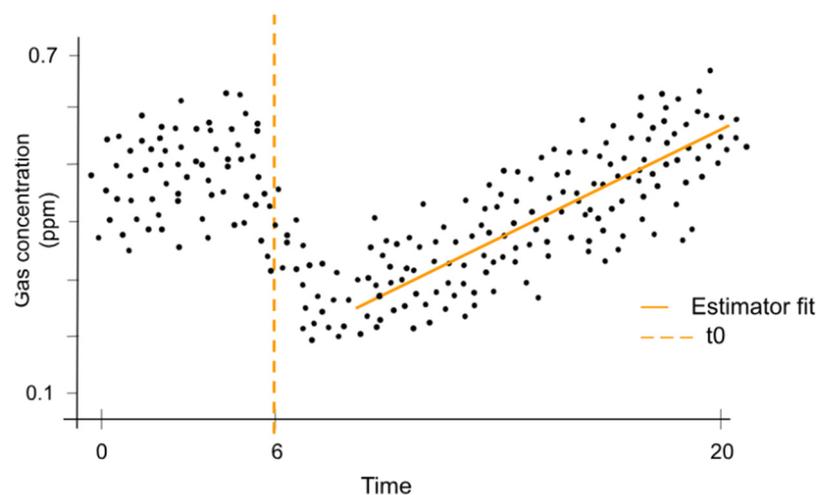


Figure 4. Due to the delay caused by the length of the tubes, the data collected during one chamber's closure corresponds to data from the previous measurement. The software discards these few initial minutes and estimates the flux with the remaining data.

2.3. Overview of the OpenToolFlux Software

This section describes how the OpenToolFlux software is used to carry out the calculation steps mentioned above. The software is a command-line application with several commands for different steps. The workflow can be divided into four main steps: (1) configuration of the software's input data and parameters in a configuration file, (2) data import from raw data files using the import command; (3) partitioning of time-series data into data segments corresponding to chamber closures and flux estimation from concentration time series using the fluxes command; (4) visualization of the results using the plot command. There is also a complementary command info that gives information on the input data. The software also has built-in documentation for each subcommand that can be accessed using the flag `--help`.

Instructions for installation and use, and a full example including input data, configuration, and output data, are distributed along with the software (<https://github.com/rasmuse/flux-estimation-example> accessed on 10 September 2023). In this section, we provide an overview of the functioning of the three main commands needed, as summarized in Figure 5.

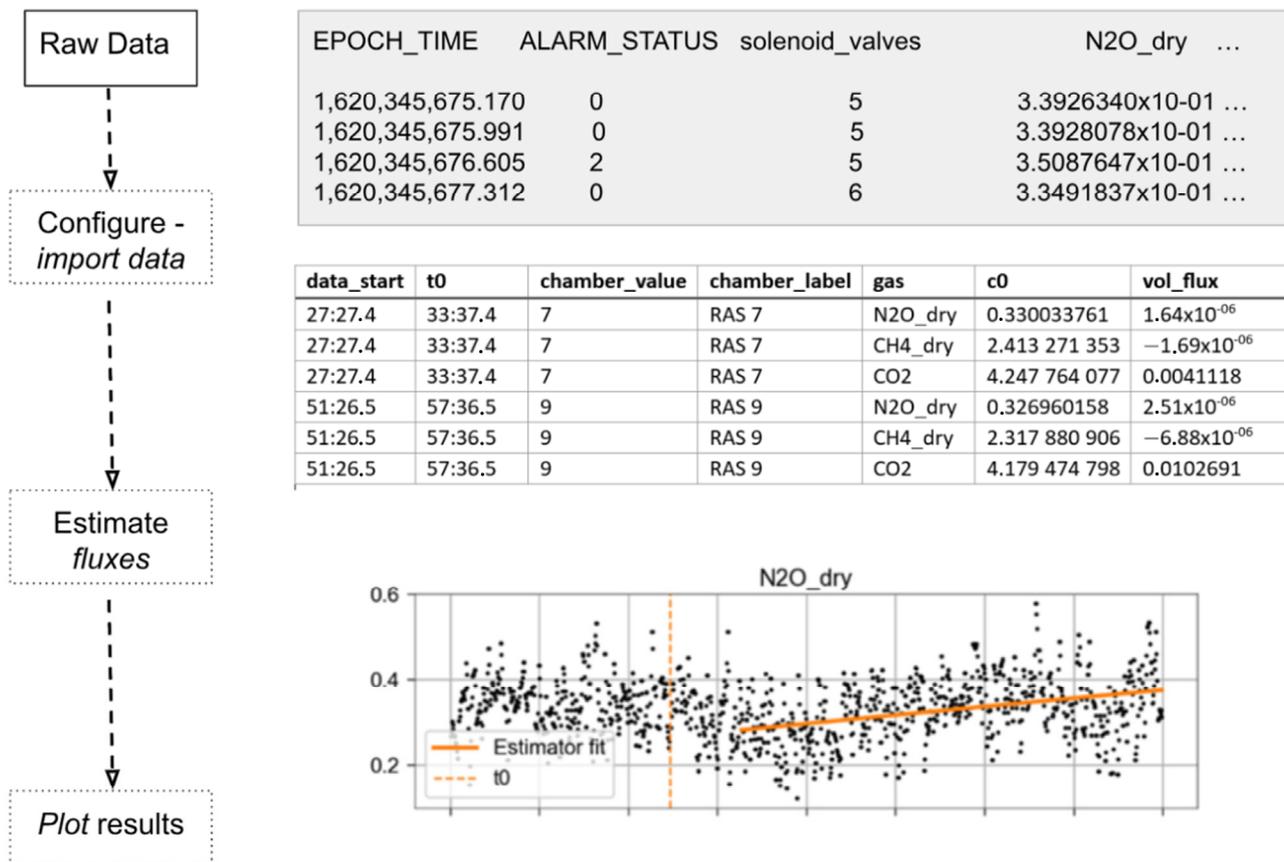


Figure 5. Schematic figure of the steps to follow when using the software. First, the configuration file is adapted to the input data and physical equipment. Second, the import command incorporates the input data into an internal database. Third, fluxes are estimated using the fluxes command. Fourth, visualizations of each curve fit can be generated using the plot command. There is an additional command, info, which provides a summary of the information about the internal database for diagnostics.

2.3.1. Configuration

The software is configured in a configuration file in TOML language. This file allows the user to specify the particular features of the input data and physical equipment used,

including the path to raw data, data format details, chamber volume and area, gas sample flow rate, tube delay time, and several settings for how to filter data to deal with occasional failures or incomplete measurements (details below). The software looks for this TOML file in the working directory and then the use of the other commands will be available.

2.3.2. Data Import

Once the configuration file is in place, the input data can be imported using the import command in the command line. The software reads tabular input data from delimited text files, such as .csv files, into a Feather file. In the tabular input files, each row corresponds to one sample. One or more columns specify the concentration of each gas measured, one column contains time stamps for each sample, and another column specifies the chamber that is currently taking the sample. The columns the user wants to import are set in the configuration file. When using the import command, a new database is created, or new data are added to a previous one, and stored in the same directory as the configuration file. The software will not modify or remove the source data, but the data imported can be replaced or changed by overwriting the Feather file. The data import step can optionally be replaced by providing a finished database in Feather format (e.g., prepared using a custom script or copying a pre-existing database).

2.3.3. Flux Estimation

Once the database is imported, the software uses the imported database and the configuration to perform the following steps: filtering the database (discards data following filter settings configured in step 1 above), cutting data into segments for each chamber closure, discarding segments that are too long or short (configured in step 1 above), estimating the flux for each of them, and finally writing all the volumetric flux estimates to a .csv file that can be opened using other software (e.g., Excel or R) for further processing.

The filtering of data, cutting into segments, and optional discarding of segments that are too long or too short are key data-cleaning operations that are, in our experience, almost always necessary to deal with occasional equipment failures and partial measurement segments generated during equipment startup and shutdown. While more complicated error modes are conceivable and may require specialized pre-processing, our experience is that the data-cleaning operations built into OpenToolFlux are very useful for typical use cases.

2.3.4. Plotting the Results

An optional output of the software is a visualization of the curve fit used for each flux estimate. This visualization shows the estimated curve fit calculated for each gas and chamber during the closing period, with additional marks to show the tube delay setting. These figures are useful diagnostic tools to identify potential errors at a glance and to possibly discard the measurements in further calculations.

3. Results

A field experiment to estimate gas fluxes from agricultural soil carried out in 2020–2021 where OpenToolFlux was applied serves as a full-scale example to illustrate the functioning of the software and the results it provides.

In addition, a smaller example including the input data, configuration files, and output data is provided at <https://github.com/rasmuse/flux-estimation-example> (accessed on 10 September 2023).

The experiment began in December 2020 at the Centro Nacional de Tecnología de Regadíos in Madrid (Spanish Ministry of Agriculture, MAPA, Madrid, Spain). It consisted of a split-plot experiment with three different cover crops (vetch, barley, and fallow) and two different soil treatments (tillage and no tillage). Fifteen automatic chambers were displayed on the field, each of them closed for 24 min with three measurements per day for each chamber, in a serial sequence, with the farthest chamber 55 m away from the analyzer.

The closure time here is only indicative; in this experiment, it was designed this way so that each chamber would take measurements at the same time every day, depending on the requirements of each experiment and the operator's needs. The experiment lasted 1 year with GHG measurements taken daily and every second, ending with approximately 34 GB of output files from the analyzer in files corresponding to 1 h blocks.

To illustrate the process when calculating fluxes with OpenToolFlux, a month of the resulting data (about 3 GB) is used in this section and the detailed configuration and command outputs are included in the Supporting Information files.

3.1. Configuration

Once the data from the analyzer is in the computer (Figure 6), it is possible to launch the software. The first step is to specify the settings in the configuration file: (1) the file paths for the input data, (2) selected columns and names (Table 1), (3) timestamp filters (range of dates from the input data, from 1 January 2021 to 31 January 2021 in this case), (4) alarm filters (filter excluding all the alarms except the water alarm, which is already corrected by the Picarro analyzer software (Picarro, Santa Clara, CA, USA)), (5) separation into measurements with a given duration (*chamber_col* identifies the number of the current valve, and the maximum and minimum duration of what is selected for measurement is 24 and 30 min), and (6) chamber labels (from RAS 1 to RAS 15).

DATE	TIME	FRAC_DAYS_SINCE_JAN1
2021-01-01	16:22:54.628	0.68257672
2021-01-01	16:22:54.967	0.68258065
2021-01-01	16:22:56.046	0.68259314
2021-01-01	16:22:56.862	0.68260258
2021-01-01	16:22:57.469	0.68260959
2021-01-01	16:22:58.187	0.68261792
2021-01-01	16:22:59.269	0.68263043
2021-01-01	16:23:00.063	0.68263963
2021-01-01	16:23:00.415	0.68264370
2021-01-01	16:23:01.503	0.68265630
2021-01-01	16:23:02.328	0.68266584
2021-01-01	16:23:02.931	0.68267282
2021-01-01	16:23:03.664	0.68268130
2021-01-01	16:23:04.769	0.68269410
2021-01-01	16:23:05.592	0.68270362
2021-01-01	16:23:05.943	0.68270767

Figure 6. Fragment from one file of raw data from the analyzer containing about 37 columns in total. Each of these files contains 1 h of data, 24 files per day.

Table 1. Imported data format and selected columns with the original column names in the raw data. We discarded most of the data that were not useful for the calculations, and we kept six columns from the previous 37. These settings are configured in the .toml file.

Selected Data	Data Format
A timestamp of the sample: EPOCH_TIME	float64
Current measuring chamber: SOLENOID_VALVE	float16
Type of alarm if raised: ALARM_STATUS	int8
CO ₂ Concentration: CO ₂	float32
N ₂ O Concentration: N ₂ O_dry	float32
CH ₄ Concentration: CH ₄ _dry	float32

3.2. Data Import

Once the configuration file is ready, we can import the data selected to create our database. The software filters the data and stores it in the Feather file (101.5 MiB), which can be exported and transferred to any other computer. Table 1 shows the columns imported into the software with the default name from the raw data.

3.3. Flux Estimation

The software now performs the calculations on the data to obtain the fluxes using the defined parameters from the configuration file. In this case, the values for the flux calculation parameters were those specified in Table 2. These values depend on the physical equipment the user has. The units of the concentration input data will determine the units of the resulting volumetric flux values. In our case, the volumetric flux becomes $\mu\text{m}/\text{s}$ because the concentration input data is in parts per million in volume (ppmv). We recommend setting the values of A, V, and Q in SI units, although technically the only requirement is that V/Q has a unit second (see the software README file for further details).

Table 2. Parameters, units, and values used in the calculations of the fluxes for this example. Chamber size, volume, and air flow of the gas from the chamber to the analyzer are required and may vary depending on the equipment.

Parameter	Value and Units	Meaning
A	0.25 m ²	Area occupied by the chamber on the soil.
Q	4.17×10^{-6} m ³ /s (0.25 L/min)	Air flow through the tube carrying the sample.
V	50×10^{-3} m ³ (50 L)	Chamber volume

The output is a .csv file with the volumetric flux values per chamber, gas, and measurement time. Table 3 shows the columns contained in the output file, with a timestamp, number and label of the chambers, the specific starting time of the estimation fit, the name of the gas, and the value for each measurement.

Table 3. Selected columns for the tabulated data.

Column Name	Description
data_start	The timestamp of the measurement's first sample
chamber_value	Current chamber being measured
chamber_label	Name of the sampling point measuring
t0	Initial time for the estimation fit
gas	Name of the gas measured
volumetric_flux	Value of the gas flux measured at the time

3.4. Plotting the Results

Finally, the *plot* command creates image files with visualizations of flux estimations as described above. Figure 7 shows N₂O emissions from 1 January. The first three measurements were taken the same day and show a normal pattern with accumulation of the gases. The last one is from September and shows an atypical distribution, probably due to a technical error during the sampling. These types of results, clearly visible, should be reviewed and discarded manually as the software does not do it automatically.

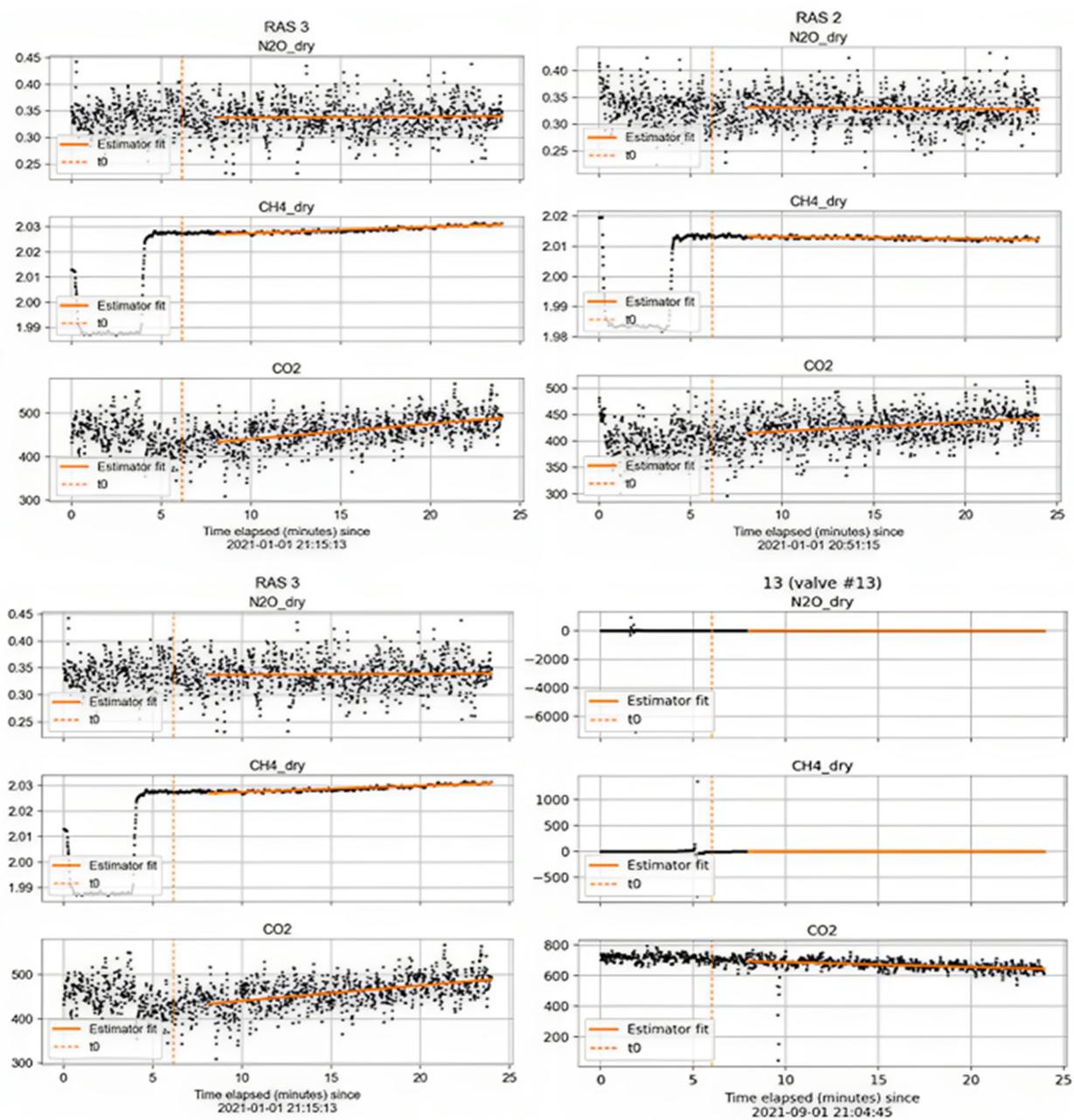


Figure 7. Three consecutive measurements in chamber number 1, 2, and 3 with the estimation of the curve for each gas. At first glance, these three measurements seem to present a normal dynamic, but the last image shows a different pattern indicating a potential error during the measurement.

3.5. Representation of the Resulting Molar Fluxes

The transformation from volumetric to molar flux, for which temperature and pressure data are required, is out of the scope of the software and this paper. The intention of this section is to show a possible use of the results obtained from the software, which in this experiment were useful to evaluate the different crops and treatments described above. Figure 8 shows the mean N_2O emissions per day for each chamber during the month processed. These results allow us to study the influence of different variables on GHG emissions. In this experiment, factors such as the soil treatments applied, cover crops selected, fertilization events, and meteorology may have effects on the emissions obtained.

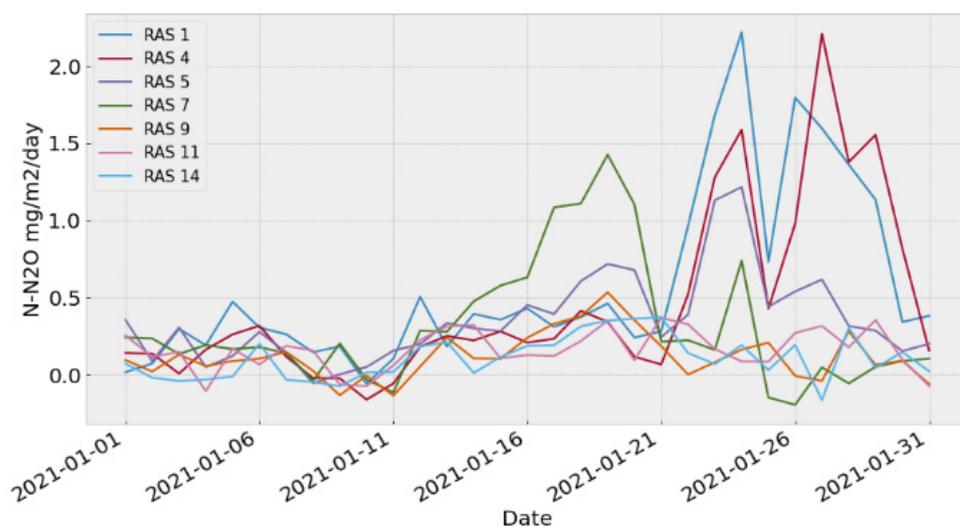


Figure 8. Daily mean N₂O emissions per chamber during January 2021. To easily distinguish the trends, only one repetition of each treatment is represented with the name “RAS” and its corresponding number. The emissions over the month show a significant increase at the end, coinciding with a rewetting event.

It is easy to see, for example, how the low temperatures and snow during a storm occurring at the beginning of the month may have lowered the microbial activity in the soil, and how the later melting of this snow probably caused a great increase in the N₂O emissions caused by a rewetting event. There is also a visible differentiation between the treatments represented, some of them showing a considerable increase in the emissions compared to others in which the soil treatment or cover crop applied seems to lower them.

The data analysis and further transformations of the results obtained from the software can be performed in any other software preferred by the user.

4. Discussion: Potential Uses and Limitations

4.1. Potential Uses

The software was created to perform recurring data analysis tasks encountered while working with data from our automatic chamber system. This loading, filtering, and transformation of data follows a common pattern but needs to be adapted to specific settings (e.g., timing and chamber configurations) that may vary with different experimental setups. Streamlining and documenting this process is important to facilitate reproducible, high-quality, and time-efficient analysis of experimental data.

The software is adaptable to other automatic systems (e.g., a different analyzer, number of chambers, closing time, etc.) assuming that (1) there is no recirculation in the system and (2) the raw data from the analyzer are tabular data files with a timestamp for each row.

Since OpenToolFlux is an open-source software, the users of Python programming are free to make any further development to their liking.

4.2. Limitations and Sources of Uncertainty

Although OpenToolFlux can handle several input data formats and different experimental setups, there are some limitations to the flexibility. As mentioned above, the software assumes a no-recirculation system and the raw data files are tabular data files where each row provides a timestamp, the current chamber, and the concentration of one or more gases.

OpenToolFlux has a flexible mechanism to discard input data with measurement values or error indicators outside specified bounds, as well as entire measurements that are too long or short compared to specification, but these mechanisms cannot reliably detect all kinds of problems or interruptions during the measurements, for example, if the chamber

was not closed while the analyzer was measuring. Additional data quality checks should be made by the user on raw data and/or program output.

Finally, it is necessary to point out that the accuracy of the model's assumptions used in the derivation of the flux estimation is outside the scope of this paper.

5. Conclusions and Further Development

Our aim herein was to develop a tool to streamline and document the process of loading, cleaning, and transforming time-series data into fluxes. The software is broadly applicable in GHG emissions research using automatic chambers, an essential technology to obtain more precise estimations of global dynamics.

This paper documents the software and its underlying assumptions and is moreover intended as a first guide when using the software, in addition to the more technical user guide distributed with the software. In addition to the example discussed in this paper, the software is provided along with a complete example of input data, configuration file, and output data as a demonstration and a concrete guide to get started (<https://github.com/rasmuse/flux-estimation-example>, accessed on 10 September 2023).

As open-source software, it is open for extension and refinement by us and/or by other users who may have different needs. We hope that the software will be useful as is and can also function as a base for further development in diverse scientific contexts.

Supplementary Materials: The following supporting information can be downloaded at: <https://www.mdpi.com/article/10.3390/agronomy13112842/s1>. Supplementary Materials S1: Solution to the differential equation. Figure S1: Fragment of the output data after calculating the volumetric flux with the fluxes command; S1 Code section: Configuration file settings used in the example with one month of data (January 2021); S2 Code section: The commands in the command prompt for importing and transforming data into fluxes in OpenToolFlux.

Author Contributions: Conceptualization, C.G.R., A.S.-C., L.L. and R.E.; Methodology, C.G.R., A.V. and R.E.; Software, C.G.R., A.M. and R.E.; Validation, R.E.; Resources, A.V.; Data curation, C.G.R., A.M. and R.E.; Writing—original draft, C.G.R. and R.E.; Writing—review & editing, A.S.-C., L.L., A.M. and A.V.; Visualization, C.G.R.; Supervision, A.S.-C., L.L. and R.E.; Project administration, A.S.-C. and A.V.; Funding acquisition, A.S.-C. and A.V. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the following sources: AgroGreen-SUDOE project (SOE4/P5/E1059) the Ministry of Science, Innovation and Universities and AgroScena-UP (PID2019-107972RB-I00) in Spain, and the Comunidad de Madrid, Spain (AGRISOST-CM S2018/BAA-4330 project), the Coordinated Research Project (No. CRP D15020) of the Soil and Water Management and Crop Nutrition Section, Joint FAO/IAEA Division of Nuclear Techniques in Food and Agriculture, through the Technical Contract "Development, Validation and Refining of New Ammonia Emission Method on Field Scale Using Nuclear" (No. 24236), the Project APOYO-JOVENES-NFW8ZQ-42-XE8B5K funded by the Comunidad de Madrid the Spanish Ministry of Economy and Competitiveness (MINECO) and a European Commission ERDF Ramón y Cajal grant (RYC-2016-20269), Programa Propio from UPM.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article and the software repository. Name: OpenToolFlux; developer and contact info: R.E., Carmen Galea, and A.M.; year first available: 2023; hardware required: personal computer; software required: Python 3.8+ (tested on Windows and Linux); availability: open source, MIT license; program language: Python 3.8+; program size: 108 kB Python code (commented, uncompressed); URL: <https://github.com/rasmuse/opentoolflux>, accessed on 10 September 2023.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. IPCC. *Climate Change 2022: Impacts, Adaptation and Vulnerability*; Cambridge University Press: New York, NY, USA, 2022; 3050p.
2. Gupta, A. Greenhouse Gas Emission Flux from Forest Ecosystem. In *Greenhouse Gases: Sources, Sinks and Mitigation*; Sonwani, S., Saxena, P., Eds.; Springer Nature Singapore: Singapore, 2022; pp. 63–84.
3. Oertel, C.; Matschullat, J.; Zurba, K.; Zimmermann, F.; Erasmi, S. Greenhouse gas emissions from soils—A review. *Geochemistry* **2016**, *76*, 327–352. [[CrossRef](#)]
4. Crippa, M.; Solazzo, E.; Guizzardi, D.; Monforti-Ferrario, F.; Tubiello, F.N.; Leip, A. Food systems are responsible for a third of global anthropogenic GHG emissions. *Nat. Food*. **2021**, *2*, 198–209. [[CrossRef](#)] [[PubMed](#)]
5. Butterbach-Bahl, K.; Baggs, E.M.; Dannenmann, M.; Kiese, R.; Zechmeister-Boltenstern, S. Nitrous oxide emissions from soils: How well do we understand the processes and their controls? *Phil. Trans. R. Soc. B* **2013**, *368*, 20130122. [[CrossRef](#)] [[PubMed](#)]
6. Giles, M.; Morley, N.; Baggs, E.M.; Daniell, T.J. Soil nitrate reducing processes—Drivers, mechanisms for spatial variation, and significance for nitrous oxide production. *Front. Microbiol.* **2012**, *3*, 407. [[CrossRef](#)]
7. Deng, L.; Huang, C.; Kim, D.; Shangguan, Z.; Wang, K.; Song, X.; Peng, C. Soil GHG fluxes are altered by N deposition: New data indicate lower N stimulation of the N₂O flux and greater stimulation of the calculated C pools. *Glob. Chang. Biol.* **2020**, *26*, 2613–2629. [[CrossRef](#)]
8. Dlamini, J.C.; Chadwick, D.; Hawkins, J.M.B.; Martinez, J.; Scholefield, D.; Ma, Y.; Cárdenas, L.M. Evaluating the potential of different carbon sources to promote denitrification. *J. Agric. Sci.* **2020**, *158*, 194–205. [[CrossRef](#)]
9. Le Mer, J.; Roger, P. Production, oxidation, emission and consumption of methane by soils: A review. *Eur. J. Soil Biol.* **2001**, *37*, 25–50. [[CrossRef](#)]
10. Poblador, S.; Lupon, A.; Sabate, S.; Sabater, F. Soil water content drives spatiotemporal patterns of CO₂ and N₂O emissions from a Mediterranean riparian forest soil. *Biogeosciences* **2017**, *14*, 4195–4208. [[CrossRef](#)]
11. Abalos, D.; De Deyn, G.B.; Kuyper, T.W.; van Groenigen, J.W. Plant species identity surpasses species richness as a key driver of N₂O emissions from grassland. *Glob. Chang. Biol.* **2014**, *20*, 265–275. [[CrossRef](#)]
12. Silverthorn, T.K.; Richardson, J.S. Forest Management Impacts on Greenhouse Gas Fluxes from Riparian Soils Along Headwater Streams. *Ecosystems* **2021**, *24*, 1810–1822. [[CrossRef](#)]
13. Grace, P.R.; Weerden, T.J.; Rowlings, D.W.; Scheer, C.; Brunk, C.; Kiese, R.; Butterbach-Bahl, K.; Rees, R.M.; Robertson, G.P.; Skiba, U.M. Global Research Alliance N₂O chamber methodology guidelines: Considerations for automated flux measurement. *J. Environ. Qual.* **2020**, *49*, 1126–1140. [[CrossRef](#)] [[PubMed](#)]
14. Hawkins, E.; Sutton, R. The Potential to Narrow Uncertainty in Regional Climate Predictions. *Bull. Am. Meteorol. Soc.* **2009**, *90*, 1095–1108. [[CrossRef](#)]
15. Charteris, A.F.; Chadwick, D.R.; Thorman, R.E.; Vallejo, A.; Klein, C.A.M.; Rochette, P.; Cárdenas, L.M. Global Research Alliance N₂O chamber methodology guidelines: Recommendations for deployment and accounting for sources of variability. *J. Environ. Qual.* **2020**, *49*, 1092–1109. [[CrossRef](#)] [[PubMed](#)]
16. Görres, C.M.; Kammann, C.; Ceulemans, R. Automation of soil flux chamber measurements: Potentials and pitfalls. *Biogeosciences* **2016**, *13*, 1949–1966. [[CrossRef](#)]
17. Kostyanovsky, K.I.; Huggins, D.R.; Stockle, C.O.; Waldo, S.; Lamb, B. Developing a flow through chamber system for automated measurements of soil N₂O and CO₂ emissions. *Measurement* **2018**, *113*, 172–180. [[CrossRef](#)]
18. Fassbinder, J.J.; Schultz, N.M.; Baker, J.M.; Griffis, T.J. Automated, Low-Power Chamber System for Measuring Nitrous Oxide Emissions. *J. Environ. Qual.* **2013**, *42*, 606–614. [[CrossRef](#)]
19. Wu, Y.; Whitaker, J.; Toet, S.; Bradley, A.; Davies, C.A.; McNamara, N.P. Diurnal variability in soil nitrous oxide emissions is a widespread phenomenon. *Glob. Chang. Biol.* **2021**, *27*, 4950–4966. [[CrossRef](#)]
20. Giltrap, D.L.; Li, C.; Sagar, S. DNDC: A process-based model of greenhouse gas fluxes from agricultural soils. *Agric. Ecosyst. Environ.* **2010**, *136*, 292–300. [[CrossRef](#)]
21. Savage, K.; Phillips, R.; Davidson, E. High temporal frequency measurements of greenhouse gas emissions from soils. *Biogeosciences* **2014**, *11*, 2709–2720. [[CrossRef](#)]
22. Courtois, E.A.; Stahl, C.; Burban, B.; Van den Berge, J.; Berveiller, D.; Bréchet, L.; Soong, J.L.; Arriga, N.; Peñuelas, J.; Janssens, I.A. Automatic high-frequency measurements of full soil greenhouse gas fluxes in a tropical forest. *Biogeosciences* **2019**, *16*, 785–796. [[CrossRef](#)]
23. Petrakis, S.; Seyfferth, A.; Kan, J.; Inamdar, S.; Vargas, R. Influence of experimental extreme water pulses on greenhouse gas emissions from soils. *Biogeochemistry* **2017**, *133*, 147–164. [[CrossRef](#)]
24. Yu, K.; Hiscox, A.; DeLaune, R.D. *Greenhouse Gas Emission by Static Chamber and Eddy Flux Methods*; DeLaune, R.D., Reddy, K.R., Richardson, C.J., Megonigal, J.P., Eds.; SSSA Book Series; American Society of Agronomy and Soil Science Society of America: Madison, WI, USA, 2015; pp. 427–437.
25. Venterea, R.T.; Petersen, S.O.; Klein, C.A.M.; Pedersen, A.R.; Noble, A.D.L.; Rees, R.M.; Gamble, J.D.; Parkin, T.B. Global Research Alliance N₂O chamber methodology guidelines: Flux calculations. *J. Environ. Qual.* **2020**, *49*, 1141–1155. [[CrossRef](#)] [[PubMed](#)]

26. Laville, P.; Lehuger, S.; Loubet, B.; Chaumartin, F.; Cellier, P. Effect of management, climate and soil conditions on N₂O and NO emissions from an arable crop rotation using high temporal resolution measurements. *Agric. For. Meteorol.* **2011**, *151*, 228–240. [[CrossRef](#)]
27. Venterea, R.T.; Parkin, T.B. Quantifying Biases in Non-Steady-State Chamber Measurements of Soil–Atmosphere Gas Exchange. In *Managing Agricultural Greenhouse Gases*; Elsevier: Amsterdam, The Netherlands, 2012; pp. 327–343.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.