



**HAL**  
open science

## Gas sensors data analysis system: A user-friendly interface for fast and reliable response-recovery analysis

B.S. de Lima, W.A.S. Silva, A.L. Ndiaye, V.R. Mastelaro, Jerome Brunet

### ► To cite this version:

B.S. de Lima, W.A.S. Silva, A.L. Ndiaye, V.R. Mastelaro, Jerome Brunet. Gas sensors data analysis system: A user-friendly interface for fast and reliable response-recovery analysis. *Chemometrics and Intelligent Laboratory Systems*, 2022, 220, pp.104460. <10.1016/j.chemolab.2021.104460>. <hal-03607535>

**HAL Id: hal-03607535**

**<https://hal.science/hal-03607535v1>**

Submitted on 5 Jan 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons CC BY-NC 4.0 - Attribution - Non-commercial use - International License

# Gas sensors data analysis system: A user-friendly interface for fast and reliable response-recovery analysis

B. S. de Lima<sup>\*1,2</sup>, W. A. S. Silva<sup>1</sup>, A. L. Ndiaye<sup>2</sup>, V. R. Mastelaro<sup>1</sup>, J. Brunet<sup>2</sup>

<sup>1</sup>Sao Carlos Institute of Physics, University of Sao Paulo, 565-905 São Carlos, SP, Brazil

<sup>2</sup>Université Clermont Auvergne, CNRS, SIGMA Clermont, Institut Pascal, F-63000 Clermont-Ferrand, France

**ABSTRACT:** Semiconductor-based gas sensors have been commercially available since the early seventies. Over the past decade, the development of nanotechnology and new carbon-nanomaterials has further increased both fundamental research and commercial innovations of such materials and devices. Each sensing element is expected to exhibit a signal for a given gas concentration that is described by three parameters: 1) the response or sensitivity, 2) the response time, and 3) the recovery time. A typical calibration or characterization procedure involves exposing several samples or devices simultaneously to different concentrations of a gas of interest. The response is then dynamically measured over time, and these three parameters can be calculated for each exposure cycle. Within this context, we present an open-source graphical user interface (GUI) that aims to facilitate the analysis procedure of dynamic response-recovery curves of resistive semiconductor-based gas sensors. The code was written in python, and it uses the open-source libraries *matplotlib*, *pandas*, *NumPy*, and *SciPy* for data visualization, handling, and fitting. PyQt is the library used for the graphical elements because it offers excellent flexibility and compatibility with different operating systems. Our software can analyze eight samples simultaneously that share the same time data, shortening the analysis process to a couple of minutes. Its source code is available at Github. This article describes its main features, the workflow, and we present three examples for data analysis whose data tables are available for user testing.

*Keywords:* GUI, gas sensor, data analysis

*\*Corresponding Author:* delima.bs@gmail.com, delima.bs@usp.br

## 1. INTRODUCTION

Gas sensors are devices that generate an electrical signal upon interaction with a gaseous species present in its surroundings. According to Liu *et al.* [1], there are two different classes of gas sensors. The first class is based on semiconductor materials with their electrical resistivity dependent on a surrounding gaseous species' concentration. Such devices are frequently addressed as *chemiresistors*, as the output signal is an electrical resistance variation upon the interaction with chemical species. The second class is based on the variation of other physical properties such as optical, acoustic, or even the material's heat capacity upon interaction with the target gas. According to the Market Size Report of 2020 from Grand View Research, the gas sensor market is projected to grow from US\$ 1 billion in 2018 to US\$ 4.1 billion by 2027 at a rate of 8.3% per year, which demonstrates the high demand for new technologies in this market [2].

Semiconductor metal oxides (SMOx) are a class of materials that are commonly used as chemiresistors [3] because they exhibit bandgaps in a range that yields considerable electrical resistance changes upon surface redox reactions with common reactive gases [4]. This effect has been known for decades. For instance, the first patent registered for a carbon monoxide detector was registered in 1971 by Naoyoshi Taguchi, and it was based on the SnO<sub>2</sub> [5], an important SMOx typically used as a gas sensing layer in modern gas sensing devices. Other SMOx with remarkable gas sensing properties are ZnO [6], In<sub>2</sub>O<sub>3</sub> [7], CuO [8], and WO<sub>3</sub> [9]. It has been empirically determined that the resistance (R) of a SMOx-based chemiresistor depends on the concentration (C) of a specific gaseous species according to a power-law described as  $R = aC^b$ , in which "a" and "b" are constants. The power-law exponent b depends on the gas and not on the sensing material. This universal behavior has been explained by Yamazoe and Shimano [10]. In their theory, the sensing mechanism is a consequence of charge transfer between the gas and the SMOx.

The pioneering work of R. Morrison in 1982 [11] first elaborated that the changes in electrical resistance of semiconductor materials during exposure to gaseous species were a consequence of redox reactions between the semiconductor's surface and the reactive gases. In his work, the charge on the semiconductor's surface  $V_s$  from ionic adsorbed gaseous species is the parameter that most affects the electrical current flowing between the semiconductor grains.

This parameter can be used to derive the resistance  $R_g$  of the semiconductor when exposed to a given gas as:

$$R_g = R_0 \exp(qV_s/k_B T) \quad (1)$$

Where  $R_0$  is the electrical resistance measured before the gas exposure,  $k_B$  is the Boltzmann constant, and  $T$  is the temperature [12, 13]. As a consequence of this charge transfer process, the gas sensing analysis of chemiresistors will always address a parameter often defined as “response”, which is the output resistance variation given by the ratio  $R_g/R_0$ , or by  $\Delta R/R_0$  (where  $\Delta R = |R_g - R_0|$ ) [14-19] or simply as  $\Delta R$ .

During the exposure of  $SMO_x$  to the target gas, a transient period occurs corresponding to the evolution of resistance from  $R_0$  to  $R_g$ . This period corresponds to the adsorption process of ions at surface if the gas is oxidizing (for example,  $O_{2(gas)} + e^-_{(surface)} = 2O^-_{(surface)}$ ) or to the consumption of already adsorbed  $O^-$  ions for reducing gases ( $CO_{(gas)} + O^-_{(surface)} = CO_{2(gas)} + e^-$ ). The kinetics of charge  $q$  adsorption is described by the so-called Elovich equation [20, 21]:

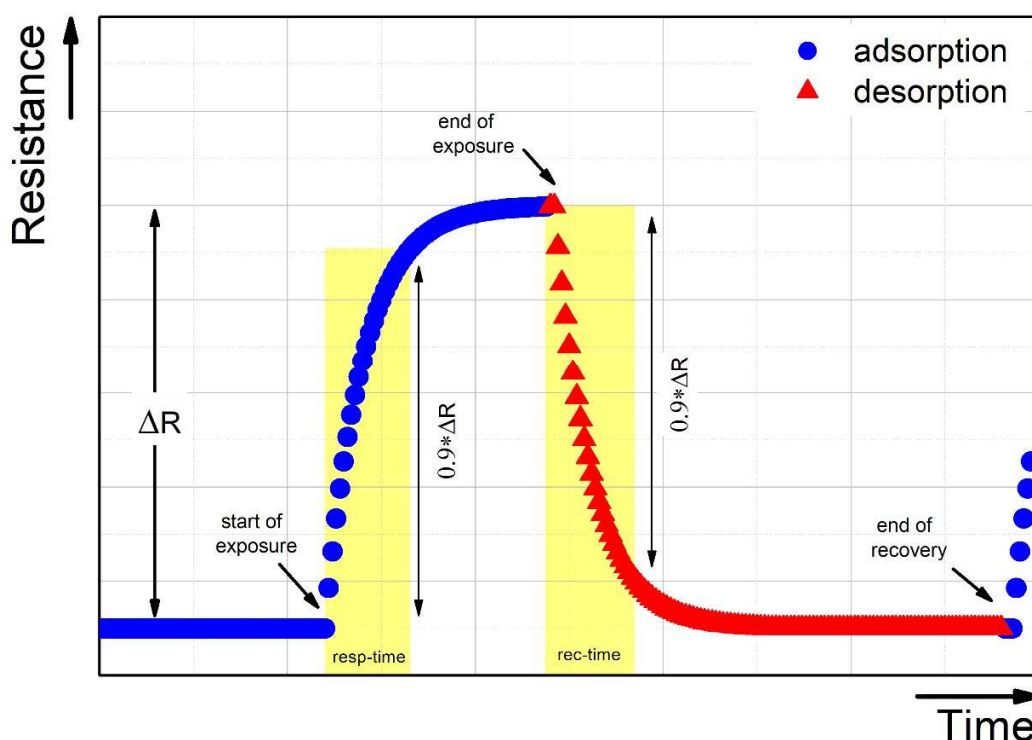
$$\frac{dq}{dt} = a e^{-bq} \quad (2)$$

Where  $q$  is the amount of charge(s) of the absorbed species,  $a$  and  $b$  are constants. By considering that the electrical resistance of a semiconductor is proportional to  $q$  due to the Morrison relation [11], it is possible to demonstrate that the  $R_g$  depends on time according to the following equation [19]:

$$R_g(t) = C + \Delta R \exp(-t/\tau) \quad (3)$$

Where the adsorption/desorption time constant  $\tau$ . By considering these parameters, the typical dynamic-response recovery curve of a *chemiresistor* upon a given gas exposure can be calculated. An example of such a response-recovery curve is shown in **Figure 1** based on an n-type semiconductor material upon adsorption of oxidizing species.

**Figure 1.** Typical exponential behavior of dynamic response-recovery curve for a gas sensor based on a semiconductor. Three parameters can be extracted from this curve, the sensor response ( $\Delta R/R_0$ ), the response-time, and the recovery-time. Arrows indicate the beginning and end of exposure and the end of the recovery.



Clifford and Tuma further investigated this process in which a semiconductor interacts with a target gas and creates a measurable resistance signal [22, 23]. Their seminal work on this subject demonstrated that this change in electrical resistivity depends on the concentration of oxygen and other reactive species. Furthermore, they proposed a general equation in which the power-law nature of the relation between the electrical resistance change and gas concentration was also valid for gases mixtures at equilibrium [22]. Regarding the transient regime between the two steady states, their model complemented the Elovich relation by proposing that the surface reaction between the semiconductor and the gas depends on the diffusion of oxygen vacancies and the potential intergranular barrier from the adsorbed species. This approach successfully

explained the strong dependence of the SMOx's response to the operating temperature temperature [23].

Two parameters known as response-time and recovery-time are usually used to estimate the charge transfer rate between the gas and the semiconductor more frequently than the time constant from equation 3. These parameters are defined as the time necessary to reach 90% of  $\Delta R$  during the adsorption/desorption regimes, and they can be depicted as an indirect measure of the surface redox reaction kinetics. Another critical parameter often discussed in the context of gas sensing performance is known as sensitivity, which is defined as the ratio between response and concentration [24]. This could be important to compare data from different studies or devices, as responses are frequently measured and presented differently. In other words, if two authors have not presented their responses in the same manner (resistance ratio, resistance variation, or percentage), the response data are often not comparable.

The calibration of gas sensors based on semiconductor materials is carried out by continuously measuring its electrical resistance upon controlled exposure to a given gas at different concentration levels. Commercial semiconductor gas sensors are calibrated to several gases, and the calibration curves are available on the device's datasheet. These curves are logarithmic plots in which the y-axis is the sensor response, and the x-axis is the gas concentrations. These log-log plots are helpful because the power-law nature of the interaction between the gas and the semiconductor [10] can be readily read. In other words, one calibration curve is determined after measuring the device's electrical resistance upon several gas exposures in which the response, response time, and recovery time should be calculated for each dynamic response-recovery curve.

Modern calibration systems in industry or academia measure several sensors' dynamic response recovery curves simultaneously, leading to massive resistance-time data tables that should be carefully analyzed to determine each device's calibration curves correctly. This task can be time-consuming and can lead to imprecisions to the correct determination of these parameters. In industry, this problem can lead to slow pace production, and in academia, it can lead to erroneous data interpretation. Recent developments on new materials for gas sensing technologies and the growing gas sensors market create a great demand for data analysis solutions. Modern gas analyzers based on multi-sensor arrays are also on the rise, and the

possibility to analyze the data from each sensing element simultaneously could benefit from smart data analysis solutions designed for such applications [25-27].

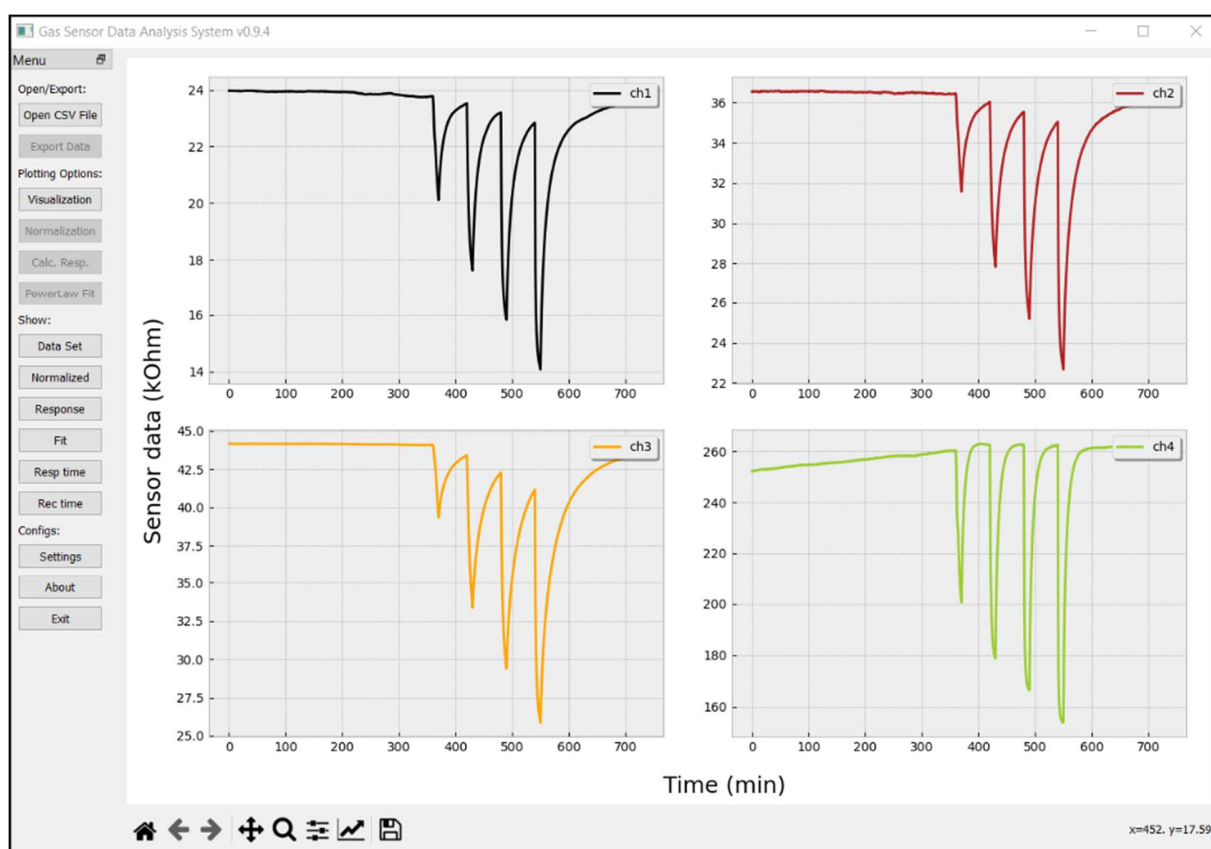
Nowadays, several tools are available that can be used to analyze data in gas sensing applications. Some are general tools for statistical analysis and machine learning that have been successfully applied to analyze the signal of multiple sensing arrays, such as Orange or the Eigenvector PLS ToolBox. Commercial gas sensors are often available with an interface that allows the user to monitor the concentration of a target gas, such as the GasLab data logging package or the SGX Sensortech Evaluation kit. Other tools are designed to analyze the data from sensors in general, not specifically for gas sensors as the Vernier Graphical Analysis software. To the best of our knowledge, there is only one analysis tool that was specifically designed for gas sensing applications, the DAV<sup>3</sup>E [28]. This software allows a complete toolbox to analyze and calibrate multiple gas sensors that operate in temperature cycling mode, generating complex patterns and requires a rigorous approach to its proper analysis.

Within this context, we report an open-source, user-friendly graphical interface (GUI) that provides a simple path to calculate the response, response time, and recovery time of 8 sensors measured simultaneously. This GUI was written in python, and it uses the open-source libraries *matplotlib*, *pandas*, *Numpy*, and *Scipy* for data handling, visualization, calculation, and fitting. The graphical elements were imported from the Quick-time library for python, known as PyQT5. This report also shows two examples of use by calculating two data tables' calibration curves, one from a single sensor measurement and another table in which four sensors were measured simultaneously.

## 2. DISTRIBUTION

This software is open-source, and the source code can be accessed on GitHub on the following repository (<https://github.com/delimabs/Gas-Sensor-Data-Analysis-System>). A windows installer is also available, not requiring any previous installation of python or the libraries. Bug reports and suggestions can be given at <https://github.com/delimabs/Gas-Sensor-Data-Analysis-System/issues>. The main layout of the software is shown in **Figure 2**. *Figure S2* shows all dialog boxes that are used in the analysis process.

**Figure 2.** Graphical User Interface showing the main menu and plot area.



### 3. ALGORITHM AND METHODOLOGY

As soon as the software is running, it creates six *pandas* Data Frames (DFs) addressed as *rawDF*, *previewDF*, *visualizationDF*, *normalizationDF*, *propertiesDF*, *fitDF* within the code. All of these data frames are initially empty and will be populated according to the user's workflow.

**Figure S1** illustrates the main tasks possible within the workflow of this software. After the software is started, the user can access the open file dialog box by clicking on the “Open CSV file” button. In this box, **Figure S2(a)**, the user should select the appropriate column separator character before opening the CSV file. Here, the system will check if the dataset has at least two columns, the first one is assumed as the time data, and all others will be assumed as the sensor signal from each sample. The algorithm will count the number of columns in the dataset, and if the number is one, it will warn the user with an error message pointing out that the column separator is not correct. The user can also fill the input boxes with the time and channel units and their respective converting factors. This feature was designed for convenience since experimental datasets often present time in seconds or minutes. The sensor signal is frequently measured as electrical resistance, typically varying from  $k\Omega$  to  $G\Omega$ . In addition, some measurement setups can measure electrical current or voltage drop as the sensor signal. The preview button allows the user to check the dataset before plotting it. After choosing these parameters and verifying if everything is correct, the system is suitable to plot data from up to eight samples (maximum columns allowed is 8, and another one from the time). The data will be inserted in the *previewDF* and plotted separately in the plot area by clicking the accept button.

Once the data to be analyzed is shown in the plot area, the user can set specific visualization parameters such as the time interval and channels of interest by clicking on the Visualization button in the dock menu initially set on the left. This button will open the dialog box shown in **Figure S2(b)**. All further analysis will depend on the parameters chosen in this dialog. Here, the user can select the time interval by setting the data frame's start and endpoints. During the evaluation of real samples/devices' gas sensing properties, it is common to let the system reach thermal equilibrium before exposing the samples to the gases, and this process can take several hours. Thus, this function was designed to allow the user to determine the region of interest for analysis. This box is designed to handle possible invalid parameters, such as writing characters when choosing the time interval or leaving all channel checkboxes unchecked. The error-handling routine in these situations consists of popping up a window to warn the invalid parameters chosen and plotting the data on the *reviewDF*, respectively. By setting these parameters and clicking the plot button, the system fills the *visualizationDF* and plots it.

After the visualization process is complete, the user can choose to compare the dataset by normalizing. For example, if the data set consists of the electrical resistance data from four

different samples, it is expected that each sample will exhibit its own electrical resistance range. Therefore, to compare, the user can choose a specific time in which the system will divide each column by its own value at that specific time. This feature allows fast comparison between different samples because all normalized data will then be plotted together. The normalization option is available by clicking on the normalization button in the dock menu that opens the normalization dialog box, shown in **Figure S2(c)**.

After setting the visualization parameters, it is also possible to calculate the response, response-time, and recovery-time for each exposure-recovery cycle from the analyzed dataset. To perform this calculation, the user should click on the dock widget's "Calc. Resp." button to access the response dialog box, shown in Figure S2(d). Each sample dataset comprises multiple exposure cycles in a typical sensor calibration routine, similar to the one pointed out in **Figure 1**. Each cycle yields one response value, one response-time, and one recovery-time. **Figure 2** presents the software interface, and it shows a dataset of four samples composed of four different exposure cycles. If the user manually calculates the response, response-time, and recovery-time for each sample cycle, that would result in 48 operations. The response dialog box will automatically perform these calculations after entering the concentration value, exposure time, end of exposure time, and end of recovery time. These values are subjective, depending on the experimental setup and configuration, but a good estimative is pointed out by arrows in **Figure 1**. To calculate the response, response time, and recovery time, the user has to click on the calculate button after filling the input box with the time of start and end of exposure and the time of the end of recovery.

The calculation of these three parameters is the core of the algorithm of this software. By entering the start of exposure time, end of exposure time, and end of recovery time, the system will first find the closest time value in the timetable for each of them, addressed here as  $t_0$ ,  $t_{f1}$ , and  $t_{f2}$ , respectively. For each sensor's data table (one column in the DF), the system will find its correspondent values to  $t_0$ ,  $t_{f1}$ , and  $t_{f2}$  and assign them to the correspondent signal variables  $S_0$ ,  $S_{f1}$ , and  $S_{f2}$ . Now, the system will calculate two variations and the response as  $\Delta S_1$  and  $\Delta S_2$ , defined as:

$$\Delta S_1 = S_{f1} - S_0 \quad (3)$$

$$\Delta S_2 = S_{f2} - S_{f1} \quad (4)$$

$$response = \frac{100x\Delta S_1}{S_0} \quad (5)$$

Once these values are calculated, the system will find in the sensor column its value corresponding to 90% variation of value  $\Delta S_1$  and  $\Delta S_2$ , as the result of the following operations:

$$S_{f1-90\%} = S_0 + 0.9\Delta S_1 \quad (6)$$

$$S_{f2-90\%} = S_{f1} - 0.9\Delta S_2 \quad (7)$$

Then, the corresponding time values will be used to calculate the response and recovery times as:

$$resp - time = t_{Sf1} - t_{S0} \quad (8)$$

$$rec - time = t_{Sf2} - t_{rS1} \quad (9)$$

A preview of response, response time, and recovery time calculated for the first channel will be shown in the dialog box. If the values are correct, the user can click the append button to append the values to a table shown in the text area to the right. This data will be used to fill the *propertiesDF*. This data frame is then displayed in the text box on the right side of the dialog box. The button “clear last” will erase the last data point in case of any errors, and the button plot will generate three different plots, one for the response, one for the response time, and one for recovery time.

After setting the correct values for each exposure cycle, the user can then fit the response data to a power-law function. This operation is carried out using the function *curve\_fit* from the *scipy.optimize*, and it considers the following equation:

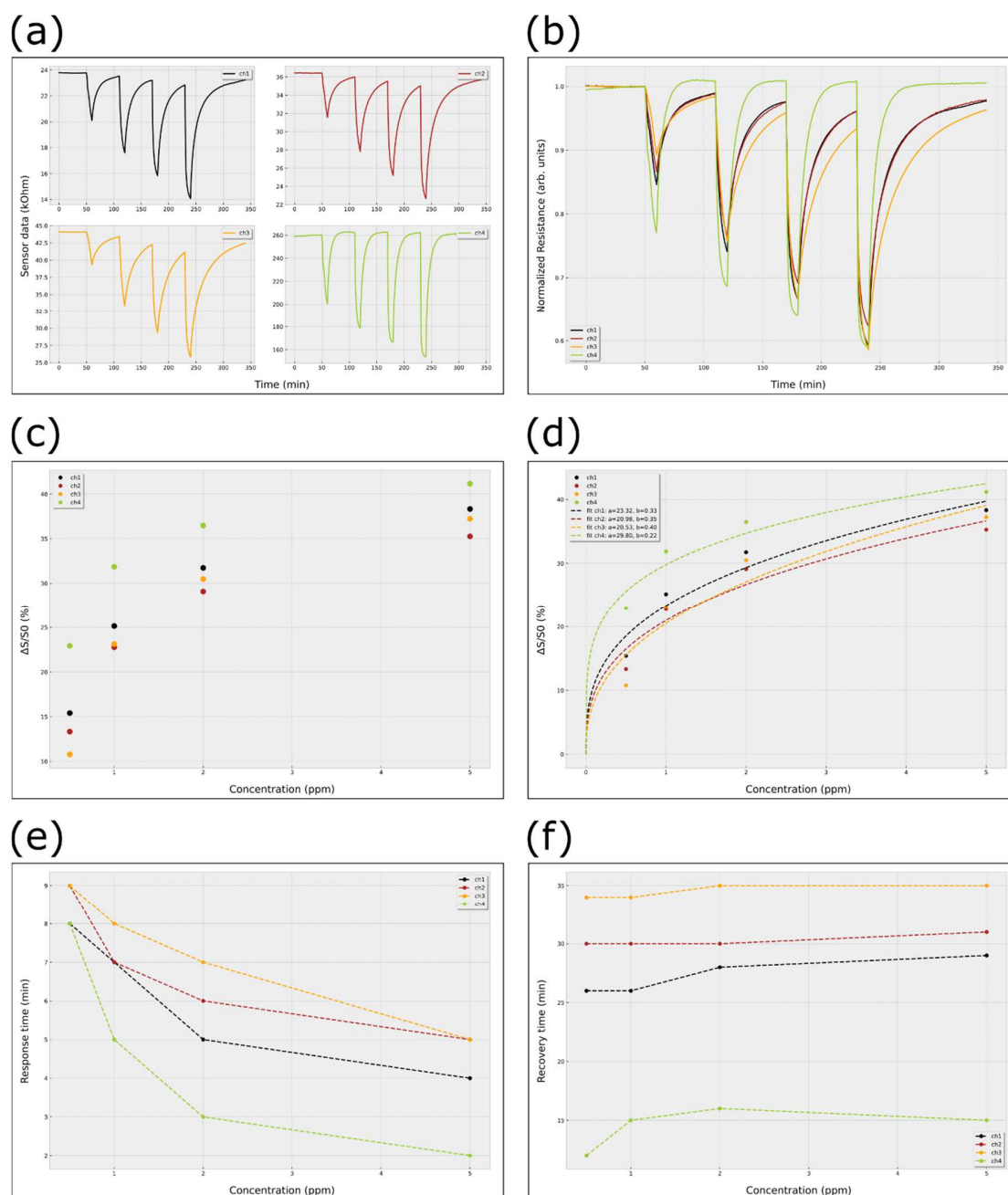
$$R = a * C^b$$

Where  $R$  is the sensor's response (equation 5),  $a$  and  $b$  are the coefficients, and  $C$  is the concentration value. Finally, after the analysis is complete for all exposures, the user can then export the data from the *visualizationDF*, *normalizationDF*, *propertiesDF*, the fit information by clicking the export button, entering the desired file name, and selecting the data available to export. Another minor feature available is related to the settings menu accessible in the main menu bar. In this dialog box, the user can choose the response form ( $\Delta S/S_0$ ,  $S_g/S_{air}$ , or simply  $\Delta S$ ), the number of points generated by the fitting procedure, different plotting styles (from *matplotlib.style*), and different color pallets for visualization. Finally, by selecting the sensitivity option in the settings menu, **Figure S2(f)**, the system will perform a linear regression of the concentration values and the chosen response, and it will show in the legend of the fitting the slope of this curve as the sensitivity and the resulting R-squared value [24]. It is also available to show the ratio between response and concentration, which can be helpful to visualize how the sensitivity is changing with concentration, in a case where the R-squared value deviates from one, which can suggest that the response is not following a linear trend.

#### 4. RESPONSE ANALYSIS EXAMPLES

In our GitHub repository, it is possible to download three ZIP files named “dataSample\_1”, “dataSample\_2”, and “dataSample\_3” to test the analysis process. Each of these files contains a folder with the analysis results, the experimental information, and the data. For these files, the time data (first column) is given in seconds. All other columns represent the electrical resistance in ohms from different samples. By the first raw data file, the user can set the column separator to Tab, the number of channels to 4, time factor to 60, channel factors to 1000, time units will be given in minutes, and electrical resistance  $k\Omega$ . These unit values could also be provided by the user and preview the data table. By accepting it, the data will be plotted in the plot area. To choose the visualization parameters for this analysis, the user can select four channels, set the time interval from 310 to 650 minutes, and select the time to zero check box. By clicking the plot button, the user should see the plot area like in **Figure 3(a)**. To compare the data from the samples quickly, the user can access the normalization menu, enter 50 in the input dialog box, and click the normalize button. Now, the plot area should be similar to the one shown in **Figure 3(b)**.

**Figure 3.** Data analysis example 1. Response-recovery curve of rGO-based sensors upon NO<sub>2</sub> exposure between 0.5 and 5 ppm. This figure shows the output of the four main processes of the data analysis. (a) Visualization, (b) Normalization, (c) Response data, (d) power-law fit, (e) response time and (f) recovery time.



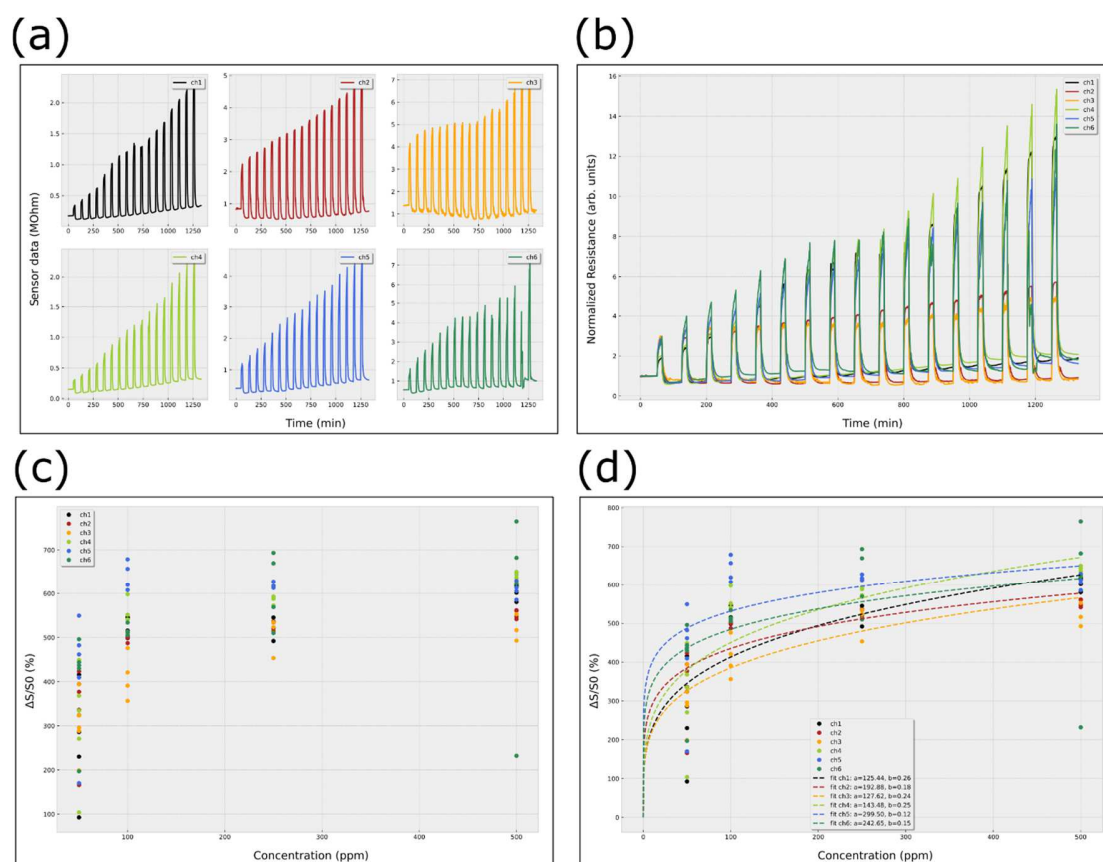
The user can access the response analysis menu by clicking the “calc resp” button in the dock menu. It is important to note that this dataset consists of four cycles of exposure to NO<sub>2</sub> during 10 minutes, with the concentration of each cycle of 0.5, 1, 2, and 5 ppm, respectively. This measurement recorded the electrical resistance over time of four different rGO-based sensors, which exhibited a p-type behavior towards this strong oxidizing gas. The recovery

period of each cycle was set to 50 minutes. For each cycle, the user can enter the correspondent start of exposure time, end time of exposure time, and end time of recovery. For instance, in the first cycle, these values are 50, 60, and 110 minutes (10 min of exposure, 50 minutes of recovery). For the second cycle, these values are 110, 120, and 170 min. For the third cycle, 170, 180, and 230. For the last fourth cycle, it was 230, 240, and 290 min. After entering the exposure cycles information, calculating the response parameters for each cycle, and appending each of these values to a table, the user can then plot the response as shown in **Figure 3(c)**. The response versus concentration data can be fitted to the power-law, and the system will return a plot as shown in **Figure 3(d)**. The response time and recovery time versus concentration are already in the memory at this point and can be accessed and visualized by clicking the “Resp time” and “Rec time” buttons on the Show menu, **Figure 3(e)** and **(f)**. All data and fit information can be exported to a specific location. The system will generate “.dat” files that will use the same column separator as the initial data file. These files are also available in our GitHub repository for comparison purposes in the results folder within the zip file.

The second zip file, “dataSample\_2”, consists of data set measured to evaluate the stability of ZnO-based sensors for O<sub>3</sub> detection. This data set consists of measuring the response-recovery cycle of two sensors prepared together in a sputtering chamber, and they were kept at 200°C for one month. Each sensor was submitted to several sequences of O<sub>3</sub> exposures in three different samples. In this data set, channels 1, 2, and 3 correspond to the data from one sensor measured on weeks 1, 2, and 3, respectively. Channels 4, 5, and 6, corresponds to the data of another sensor measured on weeks 1, 2, and 3. This dataset can also be downloaded from our GitHub repository. In this case, the stabilization period lasted 12 hours. Each sample was exposed to O<sub>3</sub> for 15 minutes and recovered for 60 minutes.

**Figure 4.** Data analysis example 2. (a) Visualization, (b) Normalization, (c) Response data, (d) power-law fit. This dataset corresponds to a stability test carried out on two ZnO-based sensors upon controlled O<sub>3</sub> exposure between 50 and 500 ppb during one month. Channels 1, 2, and 3

are the data from one sensor measured on three different weeks, while data from channels 4, 5, and 6 are the data from another similar sample measured at the same three weeks.



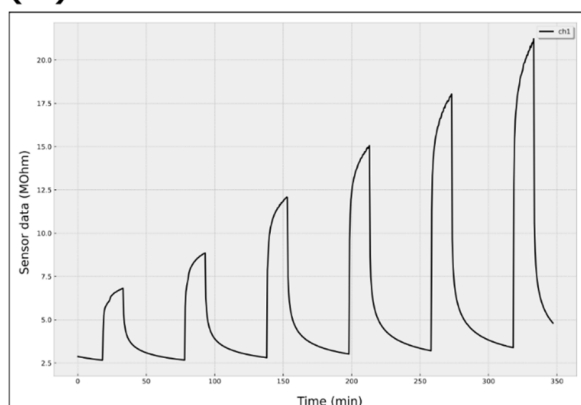
In order to perform the analysis of this dataset, the user can open the “dataSample\_2”, set column separator to tab, the number of channels 6, divide the time data by 60, and the channels by 1000, then the time will be given in minutes and the sensor data in  $k\Omega$ . By clicking the visualization button, the user can set the visualization parameters to start at 670 and to end at 2000 min. These values were chosen to get the start time of the first exposure time at 50 minutes (12 hours of stabilization = 720 min – 670 min). By setting these parameters and clicking the plot button, the user should see the plot shown in **Figure 4(a)**. For fast comparison, the data can be normalized by clicking the normalization button and setting the normalization time to 50, which will result in the plot shown in **Figure 4(b)**. To calculate the response, response time, and recovery time, the user has to enter the concentration, start of exposure time, end of exposure time, and end of recovery time for each cycle, click calculate and append. This data set’s first five exposure sequences correspond to an  $O_3$  concentration of 50 ppb, the following four

exposure cycles of 100 ppb, the subsequent four cycles of 250 ppb, and the last four of 500 ppb. The first cycle's exposure starts at 50, ends at 65, and the recovery ends at 125 (15 minutes of exposure, 60 minutes of recovery). The second exposure cycle starts at 125, ends at 140, and the recovery ends at 200. This dataset has 17 exposure cycles, and the resulting curve of response versus concentration is shown in **Figure 4(c)**. By fitting this dataset, the user should see the curve shown in **Figure 4(d)**. As these samples were prepared in the same batch and exposed to O<sub>3</sub> in the same conditions, it is expected that they would exhibit similar sensing behavior.

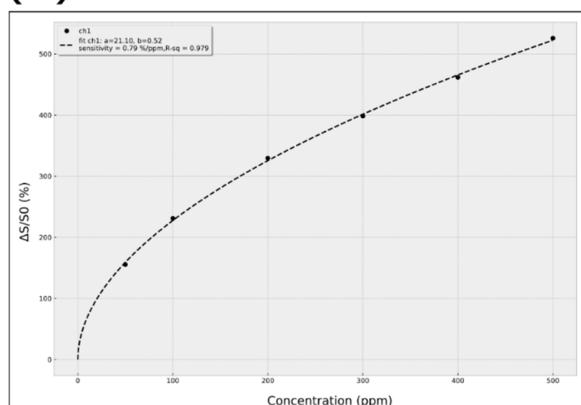
For the third example, **Figure 5**, the user can open the "dataSample\_3.dat", set the time factor to 60, unit to min, channels' factor to 1e6 (1x10<sup>6</sup>), and channel units to MΩ. This dataset was obtained after measuring the dynamic resistance of a polyaniline film upon controlled ammonia exposures ranging from 50 to 500 ppm. In this dataset, the user set the visualization to start at 3 minutes and check the box "set initial time" to zero. By entering these parameters, the user's plot should be similar to the one shown in **Figure 5(a)**. Here, each exposure cycle lasted 15 minutes, and the recovery cycle lasted 45 minutes. By filling the response parameters in the dialog box shown in **Figure S2(d)** (accessed by clicking the Calc. Resp. button), the user can then fit the data with a power-law function. Finally, in the settings menu, the user can check the sensitivity checkbox, which will also perform a linear regression between the concentration values and the response values when the power-law fit button is clicked. The sensitivity will then be the slope of this fit, and it will be shown in the legend of the figure, as shown in **Figure 5(b)**. This information will also be available to export in the Fit Info data frame.

**Figure 5.** Data analysis example 3. (a) Visualization and (b) fitting also considering a linear fit whose slope is considered the sensitivity in response/concentration units.

(a)



(b)



## 5. CONCLUSIONS

This article presents a simple and friendly graphical user interface that can analyze the data from multiple SMOx-based gas sensors that share the same data table. This system can analyze the dynamic data from 8 samples simultaneously, and it yields a fast and straightforward path to data comparison, visualization, and fitting. The system extracts the response, response-time, and recovery-time from each dynamic response-recovery cycle. Then, each of these properties can be plotted. The system can perform two fits, one related to the power-law equation from the gas-material interaction, which can yield information about the interaction mechanism between the gas/solid interaction, and another linear fit that can be useful to estimate the sensitivity of the devices.

## SUPPLEMENTARY MATERIAL

The supplementary material presents the workflow algorithm to carry out the gas sensor data analysis routine (**Figure S1**). Also, it shows the main dialog boxes for the input of the main parameters to carry out the analysis (**Figure S2**).

## **ACKNOWLEDGMENTS**

This work was based upon financial support from the Sao Paulo Research Foundation (grants 2018/07517-2, 2019/22899-1). The authors also wish to acknowledge two other Brazilian national agencies, Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) and Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES).

## **DATA AVAILABILITY STATEMENT**

The data used in this work is openly available in our GitHub repository (<https://github.com/delimabs/Gas-Sensor-Data-Analysis-System>).

## **REFERENCES**

- [1] X. Liu, S. Cheng, H. Liu, S. Hu, D. Zhang, H. Ning, A Survey on Gas Sensing Technology, *Sensors* 12(7) (2012) 9635-9665.
- [2] Gas Sensor Market Size, Share & Trends Analysis Report By Product (NO<sub>x</sub>, CO<sub>2</sub>, O<sub>2</sub>, CO), By Type, By Technology (Electrochemical, Infrared), By End-use (Environmental, Industrial), By Region, And Segment Forecasts, 2020 - 2027, 2020. 2021).
- [3] A. Dey, Semiconductor metal oxide gas sensors: A review, *Materials Science and Engineering B-Advanced Functional Solid-State Materials* 229 (2018) 206-217.
- [4] K. Grossmann, U. Weimar, N. Barsan, Semiconducting Metal Oxides Based Gas Sensors, *Oxide Semiconductors* 88 (2013) 261-282.
- [5] N. Taguchi, Gas detecting device (US3695848A), United States of America, 1971.
- [6] L. Zhu, W. Zeng, Room-temperature gas sensing of ZnO-based gas sensor: A review, *Sensors and Actuators a-Physical* 267 (2017) 242-261.

- [7] P. Xu, Z. Cheng, Q. Pan, J. Xu, Q. Xiang, W. Yu, Y. Chu, High aspect ratio In<sub>2</sub>O<sub>3</sub> nanowires: Synthesis, mechanism and NO<sub>2</sub> gas-sensing properties, *Sensors and Actuators B-Chemical* 130(2) (2008) 802-808.
- [8] K. Akhtar, I. Ul Haq, K. Malook, Gas sensing properties of semiconducting copper oxide nanospheroids, *Powder Technology* 283 (2015) 505-511.
- [9] Y. Shen, T. Yamazaki, Z. Liu, D. Meng, T. Kikuta, N. Nakatani, Influence of effective surface area on gas sensing properties of WO<sub>3</sub> sputtered thin films, *Thin Solid Films* 517(6) (2009) 2069-2072.
- [10] N. Yamazoe, K. Shimano, Theory of power laws for semiconductor gas sensors, *Sensors and Actuators B-Chemical* 128(2) (2008) 566-573.
- [11] S.R. Morrison, Semiconductor Gas sensors, *Sensors and Actuators* 2(4) (1982) 329-341.
- [12] Z. Hua, Y. Li, Y. Zeng, Y. Wu, A theoretical investigation of the power-law response of metal oxide semiconductor gas sensors I: Schottky barrier control, *Sensors and Actuators B-Chemical* 255 (2018) 1911-1919.
- [13] Z. Hua, Z. Qiu, Y. Li, Y. Zeng, Y. Wu, X. Tian, M. Wang, E. Li, A theoretical investigation of the power-law response of metal oxide semiconductor gas sensors II: Size and shape effects, *Sensors and Actuators B-Chemical* 255 (2018) 3541-3549.
- [14] Q. Zhou, W. Zeng, W.G. Chen, L.N. Xu, R. Kumar, A. Umar, High sensitive and low-concentration sulfur dioxide (SO<sub>2</sub>) gas sensor application of heterostructure NiO-ZnO nanodisks, *Sensors and Actuators B-Chemical* 298 (2019).
- [15] Y.X. Li, N. Chen, D.Y. Deng, X.X. Xing, X.C. Xiao, Y.D. Wang, Formaldehyde detection: SnO<sub>2</sub> microspheres for formaldehyde gas sensor with high sensitivity, fast response/recovery and good selectivity, *Sensors and Actuators B-Chemical* 238 (2017) 264-273.
- [16] R. Kumar, N. Goel, M. Kumar, UV-Activated MoS<sub>2</sub> Based Fast and Reversible NO<sub>2</sub> Sensor at Room Temperature, *Acs Sensors* 2(11) (2017) 1744-1752.
- [17] K.R. Park, H.B. Cho, J. Lee, Y. Song, W.B. Kim, Y.H. Choa, Design of highly porous SnO<sub>2</sub>-CuO nanotubes for enhancing H<sub>2</sub>S gas sensor performance, *Sensors and Actuators B-Chemical* 302 (2020).
- [18] W.Y. Chen, X.F. Jiang, S.N. Lai, D. Peroulis, L. Stanciu, Nanohybrids of a MXene and transition metal dichalcogenide for selective detection of volatile organic compounds, *Nature Communications* 11(1) (2020).
- [19] C.H. Wu, G.J. Jiang, K.W. Chang, Z.Y. Deng, Y.N. Li, K.L. Chen, C.C. Jeng, Analysis of the Sensing Properties of a Highly Stable and Reproducible Ozone Gas Sensor Based on Amorphous In-Ga-Zn-O Thin Film, *Sensors* 18(1) (2018).
- [20] F.C. Wu, R.L. Tseng, R.S. Juang, Characteristics of Elovich equation used for the analysis of adsorption kinetics in dye-chitosan systems, *Chemical Engineering Journal* 150(2-3) (2009) 366-373.

- [21] I.S. McIntock, The Elovich Equation in Chemisorption Kinetics, *Nature* 216 (1967) 1204–1205.
- [22] P.K. Clifford, D.T. Tuma, Characteristics of semiconductor gas sensors 1. Steady-state gas response, *Sensors and Actuators* 3(3) (1983) 233-254.
- [23] P.K. Clifford, D.T. Tuma, Characteristics of semiconductor gas sensors 2. Transient-response to temperature change, *Sensors and Actuators* 3(3) (1983) 255-281.
- [24] F.-G. Banica, *Chemical Sensors and Biosensors. Fundamentals and Applications*, John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, West Sussex, PO19 8SQ, United Kingdom 2012.
- [25] M. Wang, F. Gao, Q. Wu, J.Y. Zhang, Y.Y. Xue, H. Wan, P. Wang, Real-time assessment of food freshness in refrigerators based on a miniaturized electronic nose, *Analytical Methods* 10(39) (2018).
- [26] M.A. Ali, L. Dong, J. Dhau, A. Khosla, A. Kaushik, Perspective-Electrochemical Sensors for Soil Quality Assessment, *Journal of the Electrochemical Society* 167(3) (2020).
- [27] Z. Yuan, M. Bariya, H.M. Fahad, J.B. Wu, R. Han, N. Gupta, A. Javey, Trace-Level, Multi-Gas Detection for Food Quality Assessment Based on Decorated Silicon Transistor Arrays, *Advanced Materials* 32(21) (2020).
- [28] M. Bastuck, T. Baur, A. Schutze, DAV(3)E - a MATLAB toolbox for multivariate sensor data evaluation, *Journal of Sensors and Sensor Systems* 7(2) (2018) 489-506.