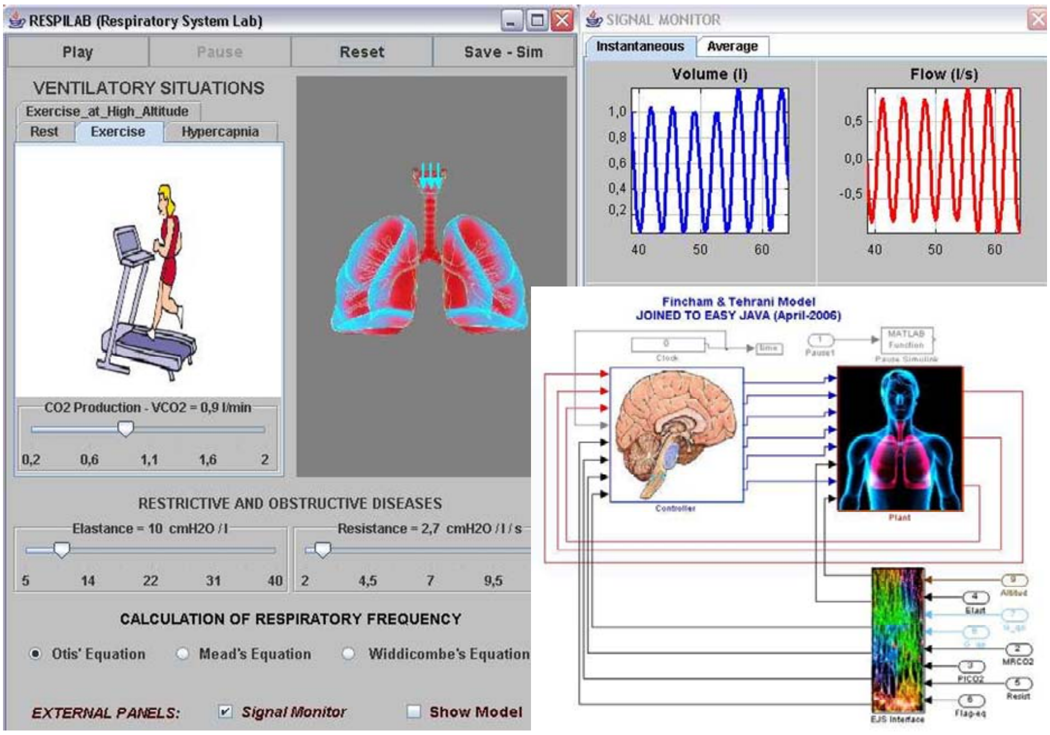
RESPILAB USER MANUAL

Virtual Laboratory for Analysis of Human Respiratory System

Version 0.3 – April 2006



Alher Mauricio Hernández

Miguel Ángel Mañanas

Ramón Costa

Table of Contents

1.About Respilab

Respilab is a virtual laboratory tool designed to analyze the human respiratory control system and to illustrate mathematical modeling of physiology processes. This laboratory is completely graphic and interactive, so it can be used to illustrate the behavior of human respiratory control system under certain circumstances or pathologies and the influence of relevant parameters in the system. This virtual lab allows the students and researchers obtain sensations and experience that would be very difficult otherwise because of the difficulties in performing experimental human studies.

The tool has been built combining MATLAB/Simulink and Easy Java Simulation (EJS) [1]. While MATLAB/Simulink allows the implementation of complex models in straightforward manner, EJS allows designing attractive views and introducing interactivity easily. This combination is quite suitable for virtual lab development.

2.Installing and running Respilab

Respilab can be run under any operating system that supports a Java Virtual Machine and MATLAB/Simulink, that is: Linux, Linux x86-64, Mac OS X, Solaris, Windows and Windows x64. This section describes the installation process assuming you are using the Microsoft Windows operating system, the explanations should be clear enough for users of different software platforms, with obvious changes.

2.1.Installation

First of all, you must know that Respilab needs Matlab/Simulink version 7.1/6.4 and Java Runtime Environment (JRE) installed previously in your computer. You can verify if Java software is installed in your machine in this web link:

http://java.com/en/download/installed.jsp

If it is not, you can download the software and install it.

Copy the file “Respilab_installer.exe” to a temporary folder and install it (*You do not need special permission*).

If you are EJS user and install Respilab in EJS folder (Do not worry for copying the files inside “Simulations” folder, the installer will do it for you). If you want to try the Respilab application temporarily we suggest you install Respilab in a different location of EJS folder.

The installer shows step by step the installation process (see *Figure 1*).

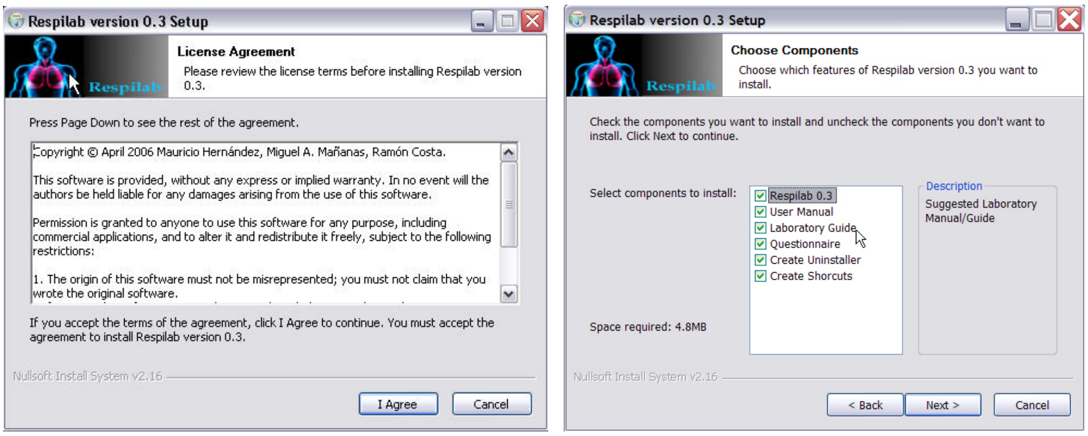


Figure 1: Respilab installer. GNU license agreement and step by step installation

All in all, if you followed the installation instructions provided and cannot get Respilab to run, please send a message to respilab@gmail.com with a simple description of the problem, including any error message you may have gotten in the process. We'll try to help you.

2.2. Running Respilab

When Respilab is installed, it will appear the windows message about there is new software installed in your computer. In this case it is true; one Respilab Shortcut will appear in your program list (see *Figure 2*).

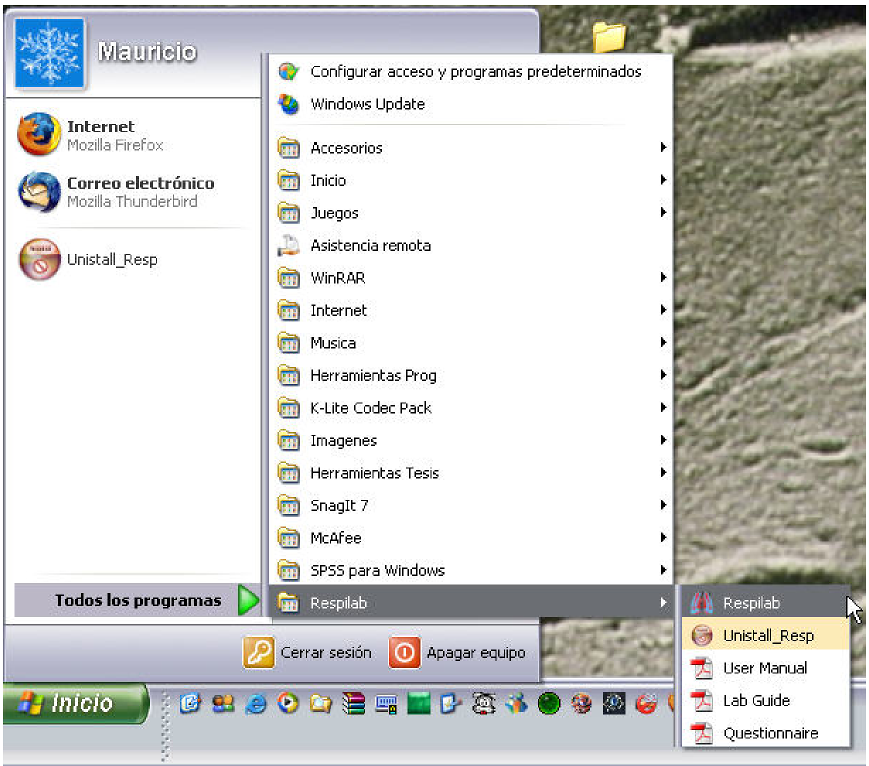


Figure 2: Start Menu, the Shortcut to Respilab, the Uninstaller and documents

Double clicking on “Respilab_03.jar” the application will run. Do not worry if the virtual laboratory console does not appear quickly, this is a normal situation because Respilab needs time to open a Matlab/Simulink session. ***Please do not close the Matlab window during Respilab use.***

The Respilab installer will copy four documents to your hard drive (if you selected them during the installation): This user manual, one suggested laboratory guide/manual, laboratory report and one questionnaire that we will be glad if you return it answered.

The other files and folders copied to your hard drive are essential for success work of Respilab. Users of Ejs only need to copy the “Respilab_0.3” folder into “Simulations” folder inside “Ejs” location to modify or run the application from the Ejs Console.

3.Interactive panel

Interface of the Virtual Laboratory is shown in *Figure 3*. The interactive module is on the left side where interesting parameters can be changed by means of sliders and tabs in order to simulate different ventilatory conditions. A multisignal scope can be seen in the right side of the interface when the user selects <Signal Monitor> as external panel.

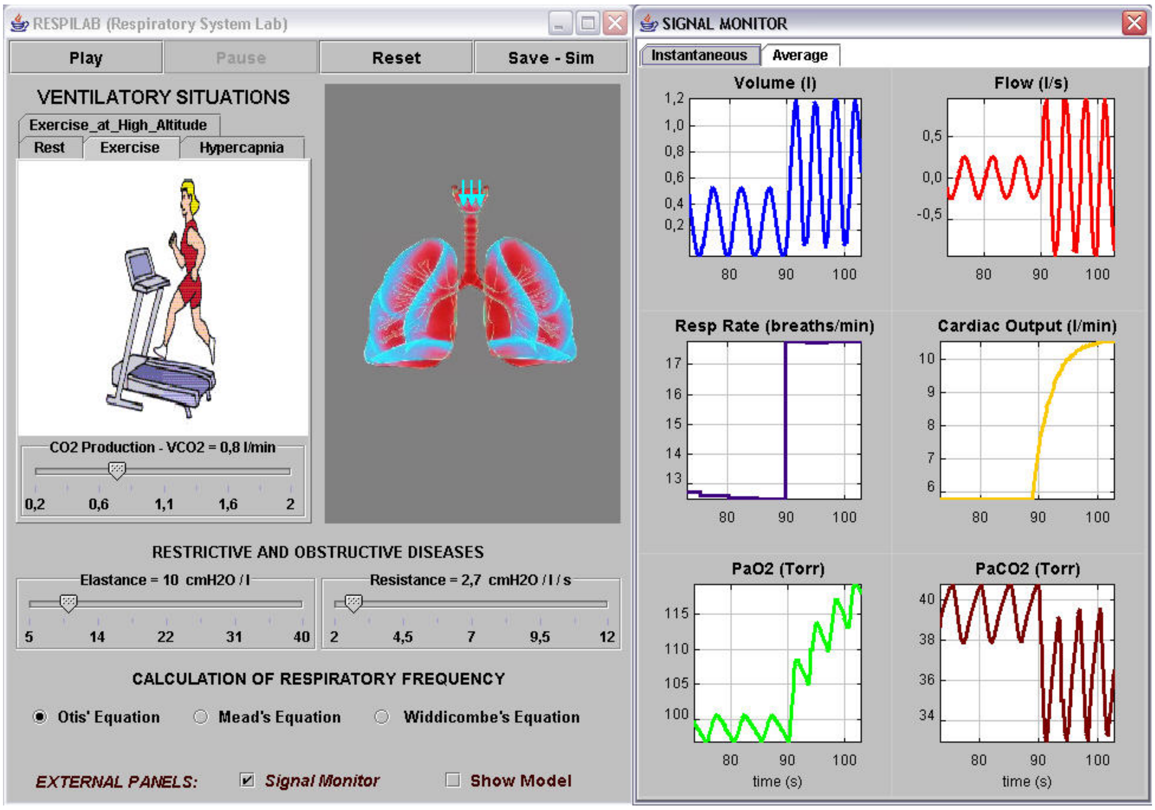
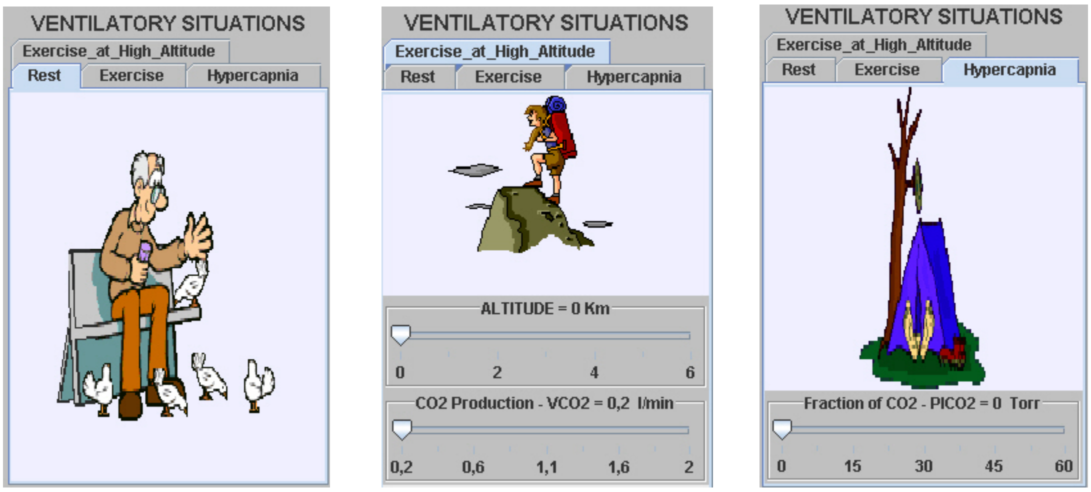


Figure 3: Interface of the Virtual Laboratory. Interactive Module and Signals Monitor are shown on the left and right side, respectively. Note external panel selected is “signal monitor”

The interactive module is composed by three important types of simulations:

- Ventilatory situations such as rest, exercise, hypercapnia and a combination of exercise and hypoxia.
- Restrictive and Obstructive Diseases.
- Calculation of respiratory frequency.

In the first one, user can select the kind of stimulus clicking one of the four tabs available. A representative animated picture is shown in each tab: a person seated in a bench when a resting condition is simulated (*Figure 4*a), a climber at the top of a mountain for exercise at high altitude (*Figure 4*b), a person breathing deeply inside a closed tent for hypercapnia (*Figure 4*c) and a runner on a treadmill in order to show exercise as it can be seen in *Figure 3*.



(a) (b) (c)

Figure 4: Tabs of Ventilatory Situations: The animated pictures represent rest in (a), Exercise at high altitude in (b), and Hypercapnia in (c).

User can changes the level of stimulus modifying the value of a respiratory parameter by means of a slider:

- Exercise: Variable $\dot{V}CO_2$ can be changed from 0.2 l/min at rest to a maximum value of 2 l/min.
- Altitude: User can simulate exercise at an altitude from sea level until 6 Km (which corresponds to a specific level of hypoxia).
- Hypercapnia: Variable P_ICO_2 is modified from 0 torr in normal conditions until 60 torr.

Besides, one of three optimization basis for the calculation of respiratory frequency can be selected by the user. The equation used by default in Respilab is Otis' equation [2]:

$$f_{Otis} = \frac{-E_{rs}V_D + \sqrt{\left(E_{rs}V_D\right)^2 + 4E_{rs}R_{rs}\pi^2 V_D \dot{V}_A}}{2\pi^2 R_{rs}V_D} \qquad (1.1)$$

This equation is calculated on a minimum respiratory work basis. Turbulence forces and inertia of the lungs are neglected considering all the work performed during inspiration interval. Respiratory system elastance (E_{rs}), airway resistance (R_{rs}), dead space volume (V_D) and sinusoidal flow waveform are considered in the mathematical expression.

Other available equations in Respilab are Mead's and Widdicombe's equation. Mead's equation [3] is formulated in the basis of optimal inspiratory pressure-time integral as a measure of the energy cost of breathing developed by the respiratory muscles:

$$f_{Mead} = \sqrt[3]{\frac{E_{rs}^2 \dot{V}_A}{4\pi^2 R_{rs}^2 V_D}} \tag{1.2}$$

Similarly, Widdicombe's equation [4] is based in the optimization principle applied to moderate and severe exercise:

$$f_{Widdicombe} = \sqrt{\frac{E_{rs} \dot{V}_A}{4 R_{rs} V_D}} \tag{1.3}$$

There are two main types of lung disease, obstructive and restrictive, which are related to changes in Resistance (R_{rs}) and Elastance (E_{rs}) respectively. Remember that both mechanical parameters are included in respiratory frequency equations: (1.1), (1.2), (1.3). This two mechanical parameters can be modified simultaneously: Elastance of respiratory system (E_{rs}) between 5 and 40 cmH_2O/l and Resistance of respiratory system (R_{rs}) in the range of 2 to 12 $cmH_2O/l/s$.

A big picture of two lungs is shown in the middle of the Virtual Laboratory and their sizes change according to the air volume inhaled or exhaled during the simulation. Their increment and reduction during inspiration and expiration, respectively, is proportional to the respiratory volume. Furthermore, three arrows over the upper airway are moved upwards and downwards indicating the entry and exit of the air. The size of the arrows is proportional to the respiratory flow.

Finally, standard options in Virtual Laboratories are provided such as to <play>, to <pause> to <reset> and to <save> the simulation at any moment. The <Save-Sim> button allows saving the simulation for future analysis or comparison with real data. Furthermore, external windows appear when the user clicks the options <Show Model> (the MATLAB/Simulink model is shown) and/or <Signal Monitor>.

4.Signal monitor

One of two kinds of plots is shown when the corresponding tab of <Signal Monitor> is selected by the user: Instantaneous and Average.

In the former, following variables are shown during the respiratory cycles corresponding to the last 30 seconds: respiratory volume, flow and frequency, cardiac output, PaO_2 and

$PaCO_2$ (see *Figure 3*). Inspiration and expiration intervals during each respiratory cycle are clearly observed by means of the sinus waveform.

In the Average option, changes of variables inside the respiratory cycle are not shown but their values from the beginning of simulation: tidal volume, total ventilation, respiratory rate, cardiac output, PaO_2 and $PaCO_2$ (see *Figure 5*). **Clicking over the trace it is possible to see the exact value of whichever variable.**

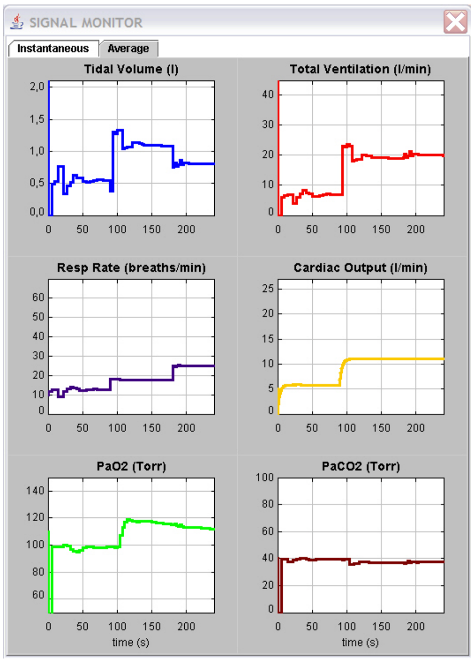


Figure 5: Tab of Average values in the Signal Monitor panel

5.Simulink model

When the external panel <Show Model> is selected, the Simulink model will appear, without difference if the model is running or not (see *Figure 6*).

Simulink model is composed by three subsystem block masked to prevent the non intentional modification.

The respiratory system model selected to build Respilab is described by Fincham & Tehrani [5], [6] and have been studied and validated in recent studies [7], [8]. In the plant, there are blocks and variables indicating physiological processes: gas concentrations in veins and arteries, ($CvCO_2/CvO_2$ and $CaCO_2/CaO_2$, respectively), gas exchange in body tissue and brain, circulatory mixing and circulation time delay from tissues to chemoreceptors. Central and peripheral chemoreceptors get the $PaCO_2$ and PaO_2 and send this information to the controller located in the medulla. The model includes important variables in the respiratory pattern generation as tidal volume, V_T, and respiratory frequency, *f*.

The model described considers a controller whose parameters of respiratory pattern are calculated every cycle as it happens physiologically. Additionally, lung volume changes following a sinus during the respiratory cycle to simulate more properly inspiration and expiration intervals. In other models, this evolution during the respiratory cycle is not considered.

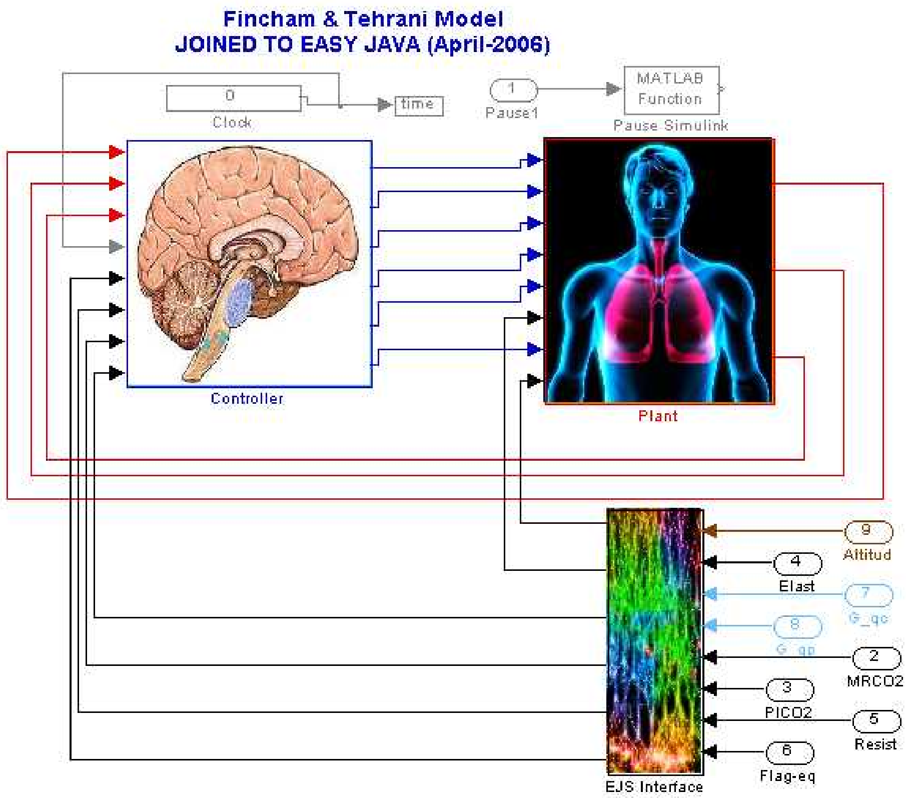


Figure 6: Simulink model of respiratory system: Controller, chemical and mechanical plant and Ejs interface.

6.Saving simulation data

Respilab has a button to save the simulation data. The button is available when the simulation is paused. Clicking on the button a window to save the simulation will appear (see *Figure 7*). The variables saved in the location you choose are:

- Time → *time*
- Respiratory frequency → F
- Tidal Volume → V_T
- Cardiac output → Q
- Arterial partial pressures of O_2 and CO_2 → PaO_2 & $PaCO_2$

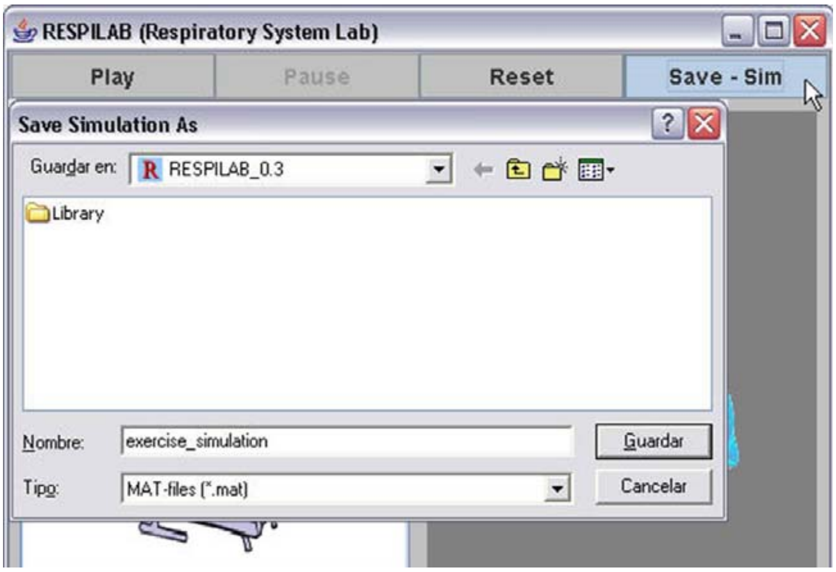


Figure 7: Saving simulation data: clicking on <Save-Sim> button, the window "Save Simulation as" will appear. In this case the name of file selected is "exercise_simulation.mat "

Data saved in the file "exercise_simulation" can be loaded from Matlab (***version 6 included***). To obtain a complete description of how to compare simulated data with real data please read the laboratory guide.

7.References

[1] Esquembre F. *Creación de Simulaciones Interactivas en Java*, Pearson Education, Prentice Hall, 2005.

[2] Otis, A. B., Fenn, W. O., and Rahn, H., Mechanics of Breathing in Man *J Appl Physiol*, vol. 2, no. 11, pp. 592-607, May 1, 1950.

[3] Mead, J., Control of respiratory frequency *J Appl Physiol*, vol. 15, no. 3, pp. 325-336, May 1, 1960.

[4] Widdicombe, J. G. and Nadel, J. A., Airway volume, airway resistance, and work and force of breathing: theory *J Appl Physiol*, vol. 18, no. 5, pp. 863-868, Sep 1, 1963.

[5] Fincham W. F. and Tehrani F. T., A mathematical model of the human respiratory system *Journal of Biomedical Engineering*, vol. 5
, pp. 125-133, 1983.

[6] Tehrani, F. T., Mathematical analysis and computer simulation of the respiratory *Biomedical Engineering, IEEE Transactions on*, vol. 40, no. 5, pp. 475-481, 1993. 0018-9294.

[7] Mananas M.A. , Navarro C., Romero S., Grinó R., Rabinovich R., Benito S., and Caminal P., "Control system response of different respiratory models under ventilatory stimuli and pathologies," *Proceedings of the 15th IFAC World Congress on Automatic Control,* pp. 2317-2322, 2002.

[8] Mananas, M. A., Hernandez, A. M., Romero, S., Grino, R., Rabinovich, R., Benito, S., and Caminal, P., "Analysis of respiratory models at different levels of exercise, hypercapnia and hypoxia," *Engineering in Medicine and Biology Society, 2003. Proceedings of the 25th Annual International Conference of the IEEE,* pp. 2754-2757 Vol.3, 2003.