User Manual

ODEKO

One Dimensional Equilibrium, Kinetics and Optimizations

Release v1.0

Author:
Nicolás de Jong Cantariño

November 12, 2023

Contents

1 Introduction

ODEKO is an application developed by Nicolás de Jong Cantariño meant to simulate the internal flow of a rocket engine. The application started as a bachelor thesis, which was supervised by Juan Manuel Tizón Pulido, for the degree of Aeronautical Engineering in Madrid's Technic University. The project was continued after the end of a thesis and currently remains in development.

The application simulates the internal flow of a rocket engine nozzle (using either liquid or solid propellant). The user is able to specify the propellant to be used, as well as the chemical species to be accounted for (and reactions in case of chemical kinetics). For solid rocket engines, the grain geometry may also be specified, along with Vielle's parameters. Finally, the user shall define the nozzle to be used or, in the case of solid propellant, said nozzle can be optimized for a given target parameter. Once all components are set, the user may perform several types of simulations. The values of all thermodynamic and transport variables along the nozzle can be obtained, as well as the rocket parameters (and their values over time if a combustion geometry is specified). The application is meant to be used as a tool for preliminary design, as teaching support in a classroom or as a simulation & optimization tool for amateur rocketry.

Throughout the user guide, all aspects of ODEKO will be explained in detail. Note that this explanation is limited to the interactions between the user and application and does not delve into the theoretical concepts behind the calculations. Nonetheless, it may be beneficial to the user to at least be aware of said principals, in order to use the application in a more adequate manner. An engineer should not take the output of a program as fact, they should have the sufficient criteria to question its results. For a detailed explanation of the concepts behind ODEKO refer to the the full documentation.

The user can also access from ODEKO an "About the application" text, where similar remarks are made. This can be found in the upper menu, under the "File" section. Furthermore, additional information as well as future updates of the software can be found at the ODEKO website and the author can be contacted at odeko@nickdejongc.com.

2 Acknowledgments

Firstly, the author would like to thank Juan Manuel Tizón Pulido for his role as bachelor thesis supervisor during the initial development of ODEKO and his continued help after the end of the bachelor thesis.

The author, as well as Juan Manuel Tizón Pulido would like to thank:

- MEC & UPM for the propulsion department collaboration scholarship which I benefited from and in which I worked on aspects of ODEKO.
- INTA & Jesús Sánchez, director of Millistic and Optoelectronic department for their material and non-material support to the department of propulsion of ETSIAE.

Furthermore, publicly available libraries and tools used internally by ODEKO should also be noted:

- Qt - Qt is a cross-platform C++ framework for GUI design. It simplifies development with a widget toolkit and signals and slots for communication. It's open source with both open-source and commercial licensing options, making it a versatile choice for efficient and polished application development across different platforms. The ODEKO GUI was created using Qt.
- QCustomPlot - QCustomPlot is a C++ library for Qt that enables easy creation of customizable and interactive plots and charts in Qt applications. It provides a simple interface for developers to generate high-quality plots with various options for customization, making it a valuable tool for visualizing data in Qt-based projects. All plots present in the ODEKO GUI employ this library.
- CVodes - CVodes is a numerical integration library for solving ordinary differential equations (ODEs) and differential-algebraic equations (DAEs). It is part of the SUNDIALS (Suite of Nonlinear and Differential/Algebraic Equation Solvers) package and is widely used in scientific and engineering applications for simulating dynamic systems. CVodes is designed to be efficient and scalable, making it suitable for a variety of numerical simulations. ODEKO employs CVodes for the integration of the differential equations required to compute the ODK model.
- Eigen - Eigen is a C++ template library for linear algebra that provides high-performance, versatile, and easy-to-use matrix and vector operations. It is header-only, meaning there is no need for separate compilation, making it convenient to integrate into projects. Eigen is widely used in scientific computing, machine learning, and computer graphics for its efficiency and expressive syntax. ODEKO employs Eigen for the resolution of linear equation systems, required to compute chemical equilibrium.

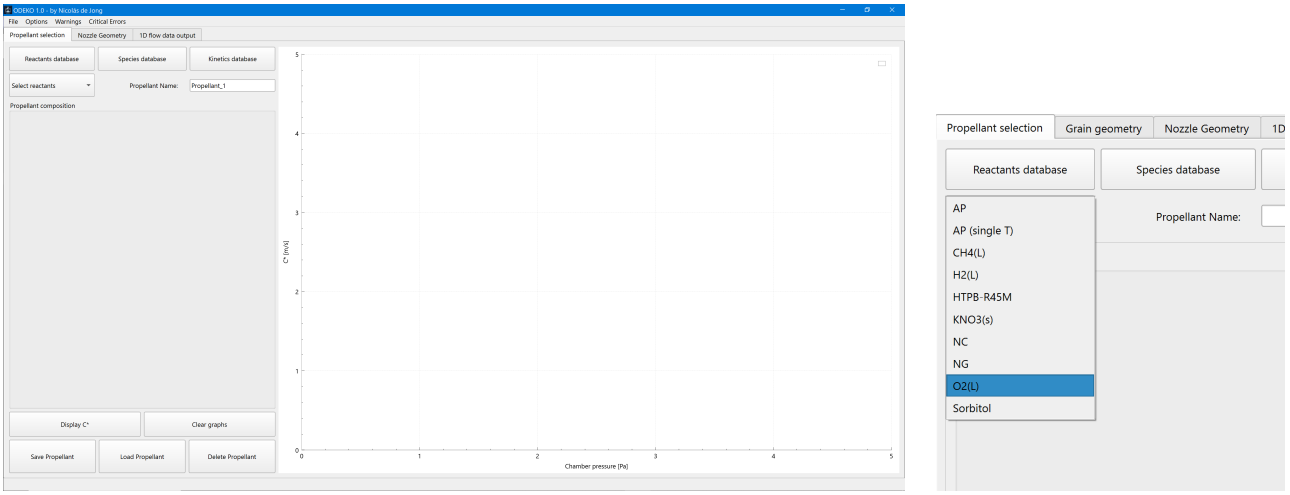
3 Propellant definition tab

This tab is in charge of the propellant definition. It allows the user to define their own propellants or modify already existing ones. It also offers the possibility to add information about Vielle's parameters or the species and chemical reactions to be considered during calculations (as well as modifying their respective databases). The propellants defined in this tab will be used in the calculations tab (and optionally in the nozzle tab in order to perform nozzle optimizations for solid propellants).

3.1 Definition of a new propellant

Right after installation, ODEKO comes packaged with a basic database that includes the most common species and reactions and example reactants and propellants. As will be subsequently explained in subsection 3.2, the user may alter the databases or add new elements to them. For the propellant definition, it will be assumed that the user has already added all necessary components to the databases or is using the default settings.

In Figure 1b the user has clicked on 'select reactant' opening a dropdown list with reactants present in the database. The user can select any set of reactants, which will appear on the space below, as seen in Figure 2. The program will not allow the user to add twice the same reactant. No other restrictions are applied. As shown in Figure 2, each reactant has a 'delete' button, allowing the user to remove it from the propellant. Each reactant also comes with an empty line box which allows the user to specify its relative mass in the propellant. The sum of all reactants relative mass does not need to add up to any specific number, each reactant will be assigned the ratio between its number and the total sum (calculations are performed per unit mass of total propellant, only the ratio is relevant).



(a) Empty propellant tab

(b) Selection of reactants

Figure 1: Selection of reactants in propellant tab

Once all reactants have been selected and their relative masses specified, the user may save the propellant. Note that if the definition is not properly done, the application will give out an error message and not save the propellant (this includes specifying a name for the propellant). If the propellant name is already taken, the application will let the user know that this data will be overwritten and allow them to cancel the operation. Furthermore, the software will also not allow the propellant to be saved if the selected reactions contain a specie that has not been selected itself (see subsubsection 3.2.1 and subsubsection 3.2.3).

It is important to remark that, when a propellant is saved it does not only include the information about its reactants (and optionally Vielle's parameters as explained in subsection 3.4) but it also includes information about the species and reactions to be considered during calculations involving said propellant. This information can be specified in the species and kinetics databases as explained in subsubsection 3.2.1 and subsubsection 3.2.3 respectively. As explained in subsection 3.5, this information is also loaded with the propellant both on this tab and on the calculations tab (see section 6).

Once the propellant has been saved it is immediately available on the calculations tab (see section 6) and on the optimizations section of the nozzle tab (see subsection 5.3).

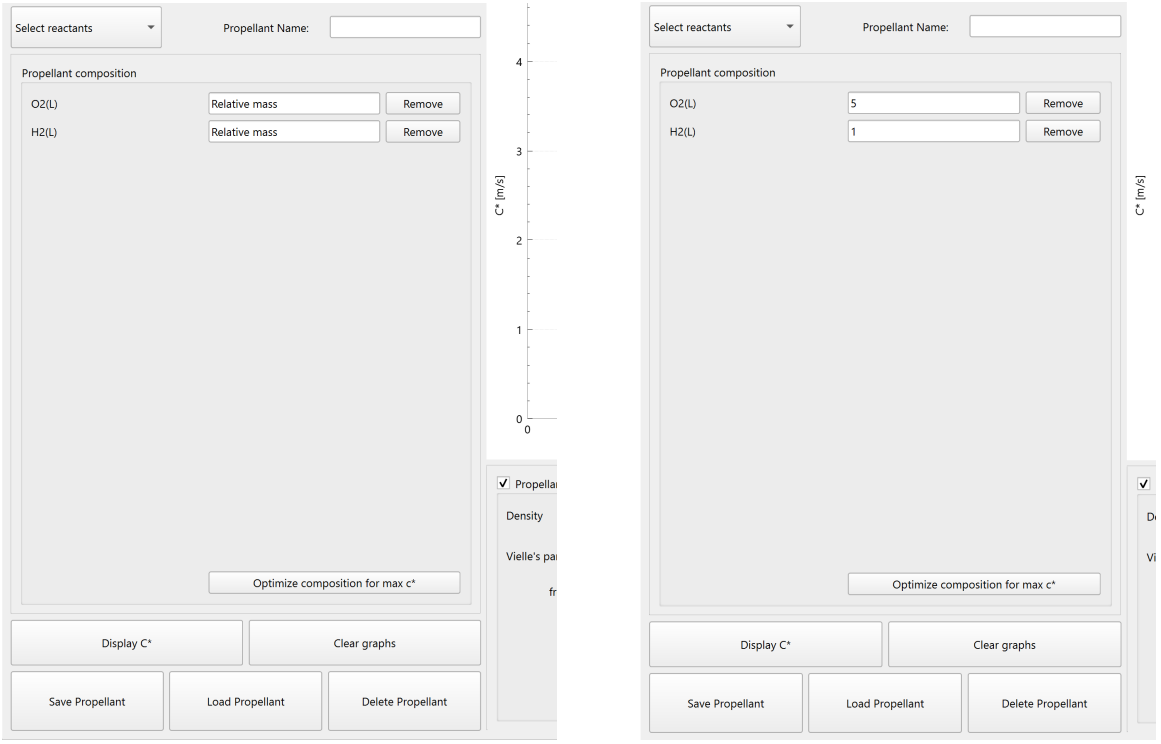


Figure 2: Two selected reactants in the composition list

3.2 Data Bases

The propellant tab has three buttons that allow the user to access the databases. It is recommended to the inexperienced user to not access this databases nor modify their content. On the other hand, the more experienced user is more than welcome to add new items or modify which species/kinetics are to be considered in the calculations. Note that ODEKO comes with default content for the databases which allows the user to simulate the most common types of propellants. Nonetheless, the user is able to add manually or through external files new items to the databases. The user is responsible for the correctness of the data that is input into the databases, since the simulation results are directly dependant on said information. ODEKO performs several checks when inputting data into the database (invalid values, coherence...) but the ultimate responsibility lies on the user's side.

3.2.1 Specie Database

This database contains the information about all the chemical species that could be considered by ODEKO. The user may edit them or add new ones. In order to add new species the user may either add them manually or import a thermodynamic data file, which follows NASA's CEA format (as explained in subsubsection 10.2.1). The user may also import transport data files to complement this information in already defined species (also in NASA's CEA format), but never as a means to define a new one (format explained in subsubsection 10.2.2). More input formats are expected to be made available in future releases.

If the user edits a specie (or manually adds one) a window with all the information about the specie will be opened, where the user can edit it. This window is shown in Figure 3. Note that transport properties are not required to define the specie, but if the user wants to obtain accurate transport properties for the internal flow it is crucial to have transport properties defined for as many species as possible. An in depth-explanation of the required data for the transport properties (least-squares fitting) can be found at the full documentation of ODEKO.

Relevant parameters in specie definition (see Figure 3):

- **Name:** It is just used for the user to identify the specie, it can be any character string.
- **Composition:** It should be written as atom identifier (e.g. H for Hydrogen), followed by number of said atoms (it can be relative amount to the other atoms), if more atoms are present an inclined bar should be added to separate them (/). Example: *C3/H2/Cl* . For ions, the electron can be written as 'E' which should have a positive index for negative ions an negative index for positive ions. Note that ODEKO performs checks on this syntax and will let the user know if there are any errors present in the composition.
- **Molecular Weight:** Note the units, $\frac{g}{mol} = \frac{kg}{kg-mol}$ & $\frac{kg}{mol} = 10^{-3}\frac{kg}{kg-mol}$

- **Formation enthalpy:** Currently this value is not used in calculations (it is used for reactant calculations, as explained in subsubsection 3.2.2).
- **Cp:** The user may define the Cp of the specie in as many intervals as he wishes, which any number of terms for the minimum squares fitting in each interval. Note the presence of two integration constants (regarding enthalpy and entropy calculations). For more detailed information consult the theory manual of the application.

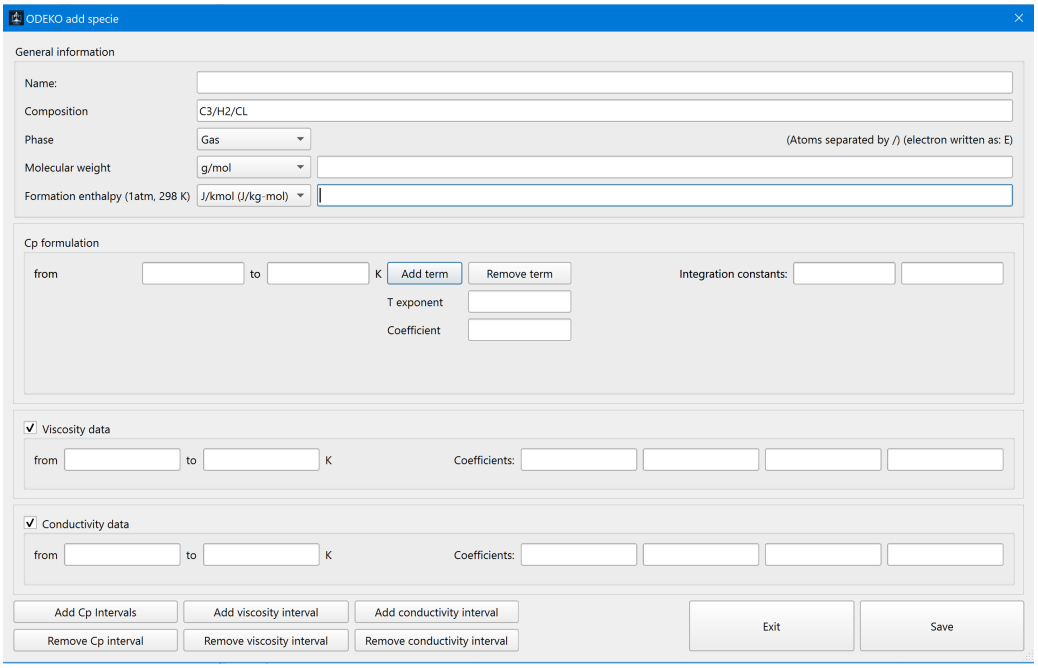


Figure 3: Manual edit/addition of specie

Note that, by default, only the species compatible with the currently selected reactants are shown in the database. 'Compatible' means that said species are made exclusively out of atoms present in the reactants. The user may uncheck the box at the bottom of the window (see Figure 4) or modify the preferences (see section 8) in order to show all species present in the database instead. By default, incompatible species will be shown as 'disabled' which will not allow the user to unmark them or edit them, this behaviour can also be modified in the preferences window. Note that if the preferences are changed so that all species appear as enabled, even though incompatible species can be checked or unchecked that will not have any effect on the current propellant (although those selection will appear when a future propellant is compatible with said species).

Note that in the database window there are two separate behaviours on exit. If the user saves the loaded species, the check-state of the species will be saved, else it will be discarded and returned to its previous state. The checkboxes indicate which species are to be considered during calculations (out

of the compatible ones). This information will be saved along with the propellant (and also loaded along a propellant). Considering all species will offer the most precise results, but will also be more computationally expensive. Therefore, the experienced user may unselect species which are known to not be present in the mixture at any point, in order to speed up the process. Note that not considering a specie present in the mixture will fundamentally alter the obtained results.

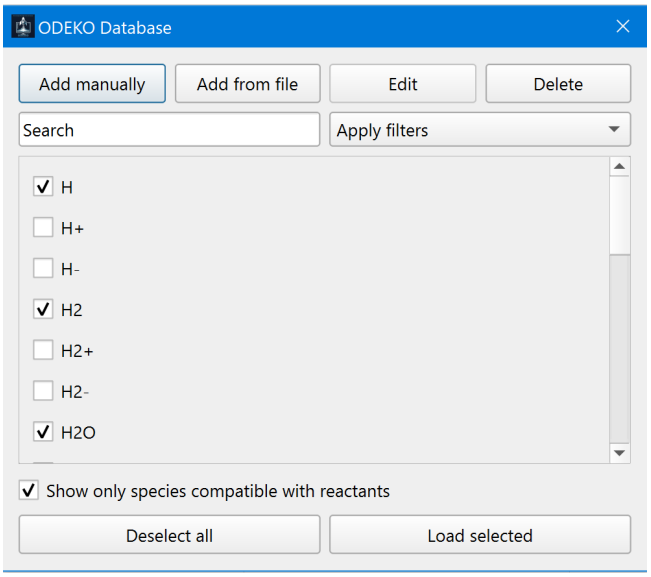


Figure 4: Specie database

As seen in Figure 4, the user is able to perform a search in the database, or apply a set of filters. The search option will only show species whose name contains the specified character string. The filters will allow the user to choose which types of species are to be shown (all, gaseous, ions, solid, liquid, with transport properties). Note that if a search option or filter is applied, the buttons 'deselect all' and 'select all' (which after 'deselect all' has been pressed appears in its place) only affect the shown species (after the filter).

3.2.2 Reactant Database

The database for reactants is quite similar to that for species, with an important distinction. Reactants do not require the user to specify a function for its C_p, it is possible to exclusively define their enthalpy of formation (along with the temperature of formation). The reasons behind this are further explained in the theory manual. Nonetheless, if the user wants to study the impact of initial propellant temperature on the rocket behaviour it is necessary to have C_p defined for the reactants, or else the same results will be given for all initial temperatures (the user will be given a warning message letting them know that the reactants are defined for a single temperature).

Moreover, the reactant database has no check/uncheck options, since reactants for the propellant are chosen as explained in subsection 3.1. Nonetheless, reactants may be selected to be edited. Therefore, it only has one behaviour on exit, since it saves no information about selected items. Furthermore, the reactant database also has search and filter options (all, gas, solid, liquid). Adding reactants manually and from a file function in the same fashion as species, except for the already mentioned differences.

3.2.3 Kinetics Database

ODK calculations (One Dimensional Kinetics) require a set of possible reactions between the species in the mixture to be specified, so does the calculation of equilibrium conductivity of the mixture as well as its Prandlt number (both properties require a set of reactions even wqhen computing through ODE, for details refer to the full documentation). The reactions of ODEKO's database can be accessed here. The behaviour of this database is quite similar to that of the species database. The user may check or uncheck reactions in order to indicate if they are going to be considered in the saved propellant or not. Note that said selections will also be loaded along with the propellant. The inclusion of reactions severely impacts the speed of computations of ODK. Therefore, it is recommended to find in the literature a suitable kinetic scheme that offers precise results for the given propellant without requiring a great amount of reactions to be considered.

Reactions may be added manually or through an import file which follows ANSYS CHEMKIN format, detailed in subsubsection 10.2.3. Adding (or editing) a reaction manually will open a window were all the relevant data can be specified (shown in Figure 5). Note that the definition of 'third bodies' is optional. It is important to keep in mind that, when writing the stoichiometry in a manually added reaction, only species that are defined in the specie database should be used. Same principle applies to the species considered as 'third bodies'.

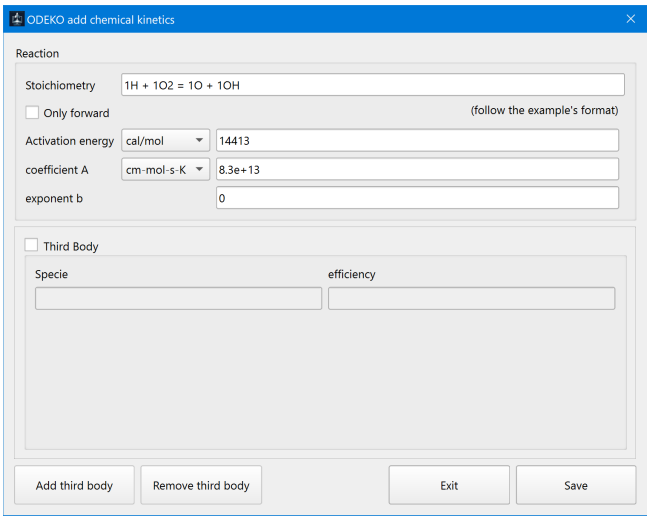


Figure 5: Manual addition of a chemical reaction

It is important to highlight that, as also happened in the species database, by default only compatible reactions are shown (reactions whose involved species themselves are also compatible). This behaviour can be changed in preferences (refer to section 8). Note that, if a reaction were to be selected but one of its species was not, a problem would arise. Therefore, ODEKO gives out a warning informing of this and does not allow the propellant to be saved until this problem is fixed (either deselecting the reaction or selecting the appropriate species). On the other hand, species present as 'third bodies' are not required to be included in the mixture. When using the propellant a warning will show up indicating that some considered third bodies are not present in the mixture. A the inclusion of third bodies not present in the mixture does not change the result of the calculations (but offers more versatility, so the reaction can be used in several different mixtures).

3.3 c^* graphs

The user has the option to graph the value of the c^* of the given propellant with respect to pressure or mixture ratio. Note that this is performed on the same tab and does not require a grain or nozzle definition (it only accounts for combustion chamber values).

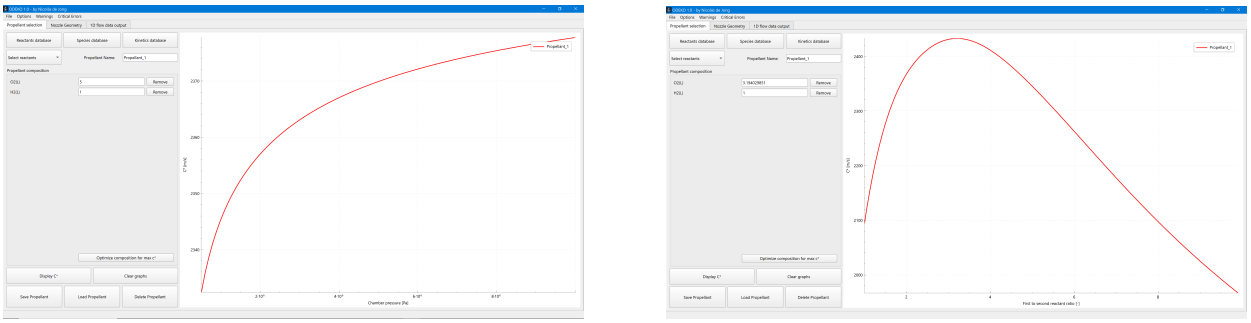
3.3.1 c^* with respect to chamber pressure

At any time, if at least one reactant is present the user may plot the c^* of the propellant with respect to pressure, by clicking on the button on the lower left of the screen (Figure 6a). As many plots as desired can be represented in the same graph (the user may change the composition and re-plot the graph). Note that the name for each graph present in the legend is the name that was given to the propellant. If there are two or more plots with the same name, a numeric index will be appended in order to differentiate them. If a different type of curve were to be plotted (c^* with respect to mixture ratio) all current plots of the graph would be cleared (since the x axis is changing). The upper and lower limits of the x axis may be modified in the preferences menu (refer to 8).

3.3.2 c^* with respect to mixture ratio

This type of curves may only be plotted if exactly two reactants are present. When this condition is satisfied, a button will appear giving the user this option (Figure 6b). Note that the mixture ratio in the x axis is the ratio between the first reactant added and the second. Therefore, it would be wise to have the oxidizer as the first reactant for most common propellant combinations. Nonetheless, the user may change the lower and upper limits of the x axis at any time in the preferences menu (refer to section 8). The plot is also performed for a constant chamber pressure, which also may be modified in the preferences menu. This plot also performs an "optimization" where the highest value of the propellant's c^* is found in the given mixture ratio range; afterwards, said ratio is set as the reactants relative masses. As was mentioned in subsubsection 3.3.1, the user may plot as many curves as they

want as long as they are of the same type (mixture ratio in this case), plotting c^* over pressure would change the x axis and in consequence clear the previous plots.



(a) c^* with respect to pressure

(b) c^* with respect to mixture ratio

Figure 6: c^* plots in ODEKO for HLOX propellant

3.4 Vielle's parameters

The user also has the option (although it is not required) to specify the propellant's Vielle's parameters in the lower right section of the propellant tab. Note that this region is hidden by default (as all options related to grain geometries and solid motor simulation are hidden in a default installatioin). Nevertheless, the user can toggle through the options menu (subsubsection 7.2.1), as well as set the solid grain options enabled by default in preferences (subsubsection 7.2.6). On Figure 1a the propellant tab with hidden solid motor information was shown, whilst on Figure 7 the user can see the same tab when said option was untoggled in the options menu.

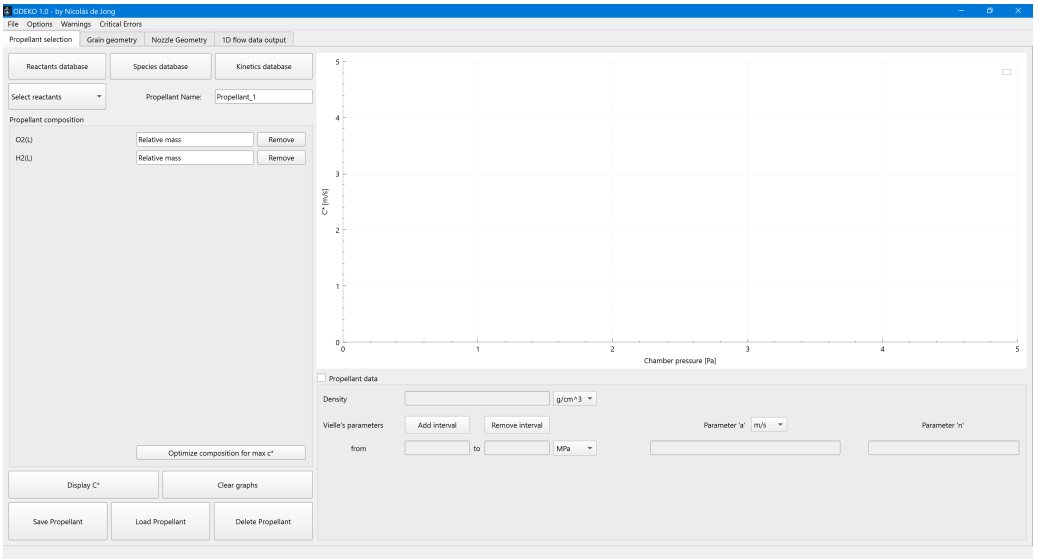


Figure 7: Propellant tab with shown solid motor information

Specifying Vielle's parameters as well as propellant density is required if the user wants to simulate a grain combustion with this propellant (although it is not necessary for simulations with constant chamber pressure). Vielle's parameters can be defined for different pressure intervals of the propellant (as many as the user is willing to add). This definition also requires the definition of the propellant density (to be used in grain combustion calculations).

It is important for the user to be aware of the units employed during the definition. The user may select out of a variety of units, both for speed and pressure. Note that, when the a and n constants are given, they must be coherent with the units of pressure and speed selected. The value for n does not depend on the units. On the other hand, the value of a depends not only on the units used for speed but also on the pressure units used during the linear fitting, changing said units will also change the value of a. It is recommended to the inexperienced user to not try and perform a unit conversion and write the values in the units in which they were found in the literature. The user may also choose if an upper pressure limit is to be specified or not in the preferences menu (refer to section 8). The only difference is that, if an upper limit is specified a warning will be set if a pressure above this limit is used in calculations (but the parameters for the last interval will be used anyway).

3.5 Advanced use of this tab

The fact that the considered species and reactions during calculations are saved along with the propellant offers many possibilities to the user. Firstly, it allows for quick comparison of different kinetic schemes (or species considerations) for the same propellant. The user may just save two copies of the propellant each with a different scheme and perform calculations with both in the calculations tab.

Moreover, since the kinetic scheme and considered species are dependant on the propellant, it is very helpful to have them tied to each other. Therefore, if a propellant is to be used, the user does not need to stop and think about the kinetic scheme or try to find in their notes which scheme to use (as long as it was adequately chosen when defining the propellant). Therefore, a proper definition allows the user to quickly change propellants during calculations without worrying about kinetic schemes.

Still, even though it is useful to only worry about the schemes during the propellant definition, it may seem tedious to some to have to define the scheme for each propellant (specially if they are similar in nature and the same scheme can be applied to all of them). ODEKO easily allows to avoid this hassle. When new reactants are added, the default scheme is that which was used for the last propellant. Moreover, if a propellant is loaded the default scheme will change to the one of that propellant. Consequently, if the user wants to define a new propellant which uses the same scheme as a previously defined one, the user may just load it to get its scheme and then change the composition in order to obtain the new propellant. Note that, the only species and reactions whose consideration is changed when loading a propellant are those who are compatible with said propellant. Therefore, if a O-C-H-N propellant is loaded, the inclusion of species with Cl will not change, it will remain as it

was (corresponding to the selection of the last propellant that included Cl). On the contrary, species containing exclusively O-C-H-N will be modified accordingly to the new propellant (either including them or excluding them). Therefore, if a propellant is being used as a scheme base for a new one, it is important that they have the same elements present (or to be aware that it is needed to manually modify or attend to the elements in which they differ).

3.6 Export

As explained in subsubsection 7.1.2, in every tab the user has the ability to export the present data, via a button in the file menu. Note that only one .csv file will be created, which includes all selected information. In the propellant tab, the user is able to choose which of the following sets of data are to be exported:

- Propellant composition: Exports to a .csv file information about which reactants form the propellant and their relative masses.
- Propellant data: Exports to a .cvs file additional information about the propellant: density and Vielle's parameters definition.
- Considered species: Exports to a .csv file a full list of all the species that are considered in calculations performed with this propellant.
- Considered reactions: Exports to a .csv file a full list of all the reactions that are considered in calculations performed with this propellant.
- Cstar curve: Exports as a .png file the current c^* graph present in the propellant tab (including the legend).

4 Grain geometry definition tab

By default, this tab is completely hidden from a new user. Nevertheless, it can be easily toggled on in the options menu (subsubsection 7.2.1), by deselecting the option "hide solid grain simulations". The user can disable this option by default in preferences (subsubsection 7.2.6), in order to have this tab visible every time the software is launched. Enabling this option also enables aspects related to solid grain combustion on the other three tabs, as explained in subsection 3.4, subsection 5.3 and subsubsection 6.2.2.

This tab is in charge of the definition of grain geometries for solid propellant combustion. Note that, in contrary to propellant and nozzle definition, grain definition is not required to perform calculations. Grain geometry is only used for nozzle optimizations or to study the time evolution of solid propellant rocket engines. The default grain geometry tab can be seen in Figure 8.

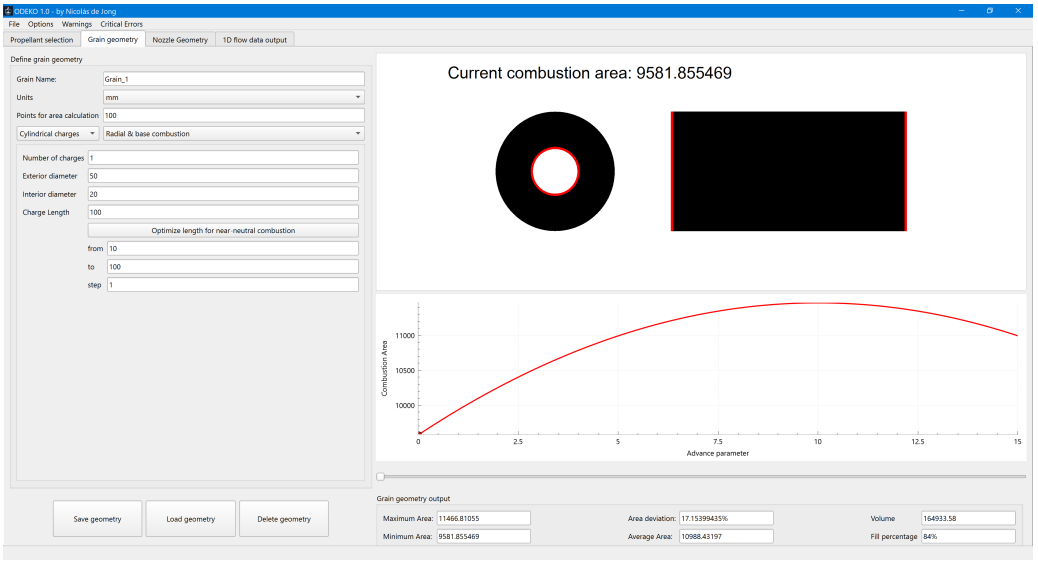
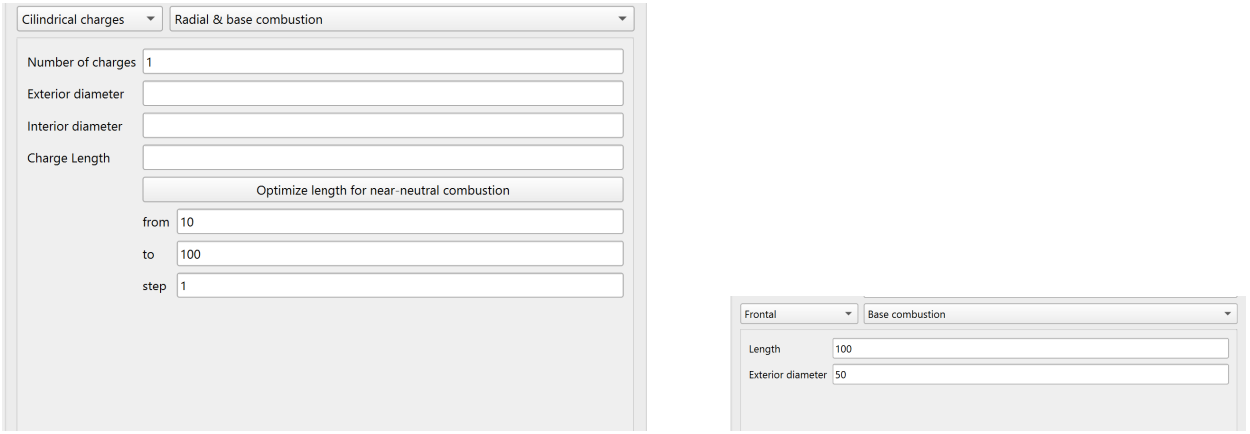


Figure 8: Base grain geometry definition tab

4.1 Grain definition

In order to define the grain geometry the user must specify a name, the number of area points to be used during computations, the units in which the values will be inputted and the type of geometry and the geometry parameters. In the subsequent subsections each type of geometry definition will be explained. The type of combustion must also be specified, either radial, through the bases or both types simultaneously. The type of combustion affects both the shape and values of the combustion area curve. Note that the number of charges does not change the shape of the curve (they all have the same combustion) but it does multiply its value.



(a) Definition of Cylindrical charge

(b) Definition of frontal combustion charge

Figure 9: Two grain geometrz options currently available

4.1.1 Cylindrical charges

In Figure 9a the parameters used in cylindrical charge definition are shown. Note that, for cylindrical charges, as seen in the figure, an optimization of the charge length is possible. Said optimization searches for the charge length that results in the least difference between maximum and minimum areas of combustion, standardized with the average area. Therefore, the closest to neutral combustion is obtained for said length. The optimization allows the user to set the minimum and maximum lengths between which the search is performed and the step to be used in the search.

4.1.2 Frontal combustion

The parameters from frontal combustion are: number of charges, length and exterior diameter. This type of geometry only allows for frontal combustion (it makes no sense to have a radial combustion without an interior hollow space). Furthermore, frontal combustion assumes that combustion only takes place in one of the bases of the charge. On the other hand, cylindrical charges with base combustion have both bases as active combustion surfaces.

4.1.3 Star charges

Currently star charges are not supported, but they will be implemented in version 1.1.

4.1.4 Custom charges

Currently custom charges are not supported, but it will be implemented in the future.

4.2 Output

The grain geometry tab has three output components. Firstly, on the bottom right of the screen a set a numerical values are displayed (Figure 10) for the following parameters: maximum area, minimum area, area deviation, mean area, volume occupied by propellant and percentage of chamber volume occupied by propellant. The area deviation is computed as the difference between maximum and minimum combustion area divided by the average combustion area.

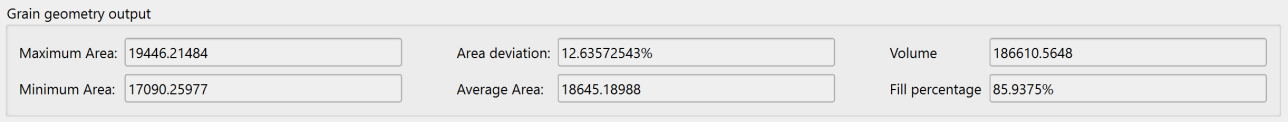


Figure 10: Grain geometry output parameters

Additionally, two connected visual aids are also given, as seen in Figure 8. The plot represent the evolution of the area relative to the advance parameter from the start of the combustion until the entirety of the grain has been combusted. The image on top represents the shape of the grain, as seen from the top and from the side. There is a scrollbar which the user may move, this will change the "current" advance parameter, which will be marked in the graph with a dot. Moreover, the grain representation will change in order to show its state for the current advance parameter (the combustion surfaces, marked red, will advance). The value of the combustion area for said advance parameter is also displayed.

4.3 Export

As explained in subsubsection 7.1.2, in every tab the user has the ability to export the present data, via a button in the file menu. Note that only one .csv file will be created, which includes all selected information. In the grain geometry tab, the user is able to choose which of the following sets of data are to be exported:

- Grain parameters: Exports to a .csv file all parameters related to the definition of the grain geometry.
- Grain image: Exports to a .png file the current state of the grain image of the tab (upper right).
- Combustion area curve: Exports to a .png file the current state of the combustion area plot of the tab (lower right).
- Combustion area values: Exports to a .csv file a table containing the combustion area value for each value of the advance parameter (for the number of total points specified in the grain definition).

5 Nozzle definition tab

This tab is in charge of the nozzle geometry definition. Along with the propellant definition, the nozzle definition is required to perform any calculations. On launching the application a default nozzle will be shown in the window, but the user may make any modifications to it. The user might choose between two nozzle types and input/modify their parameters. Moreover, if the menu option "hide solid motor simulation" is toggled off (see subsubsection 7.2.1), the optimization section might be accessed, in which the user can select a grain geometry, as well as an engine characteristic in order to optimize the nozzle shape.

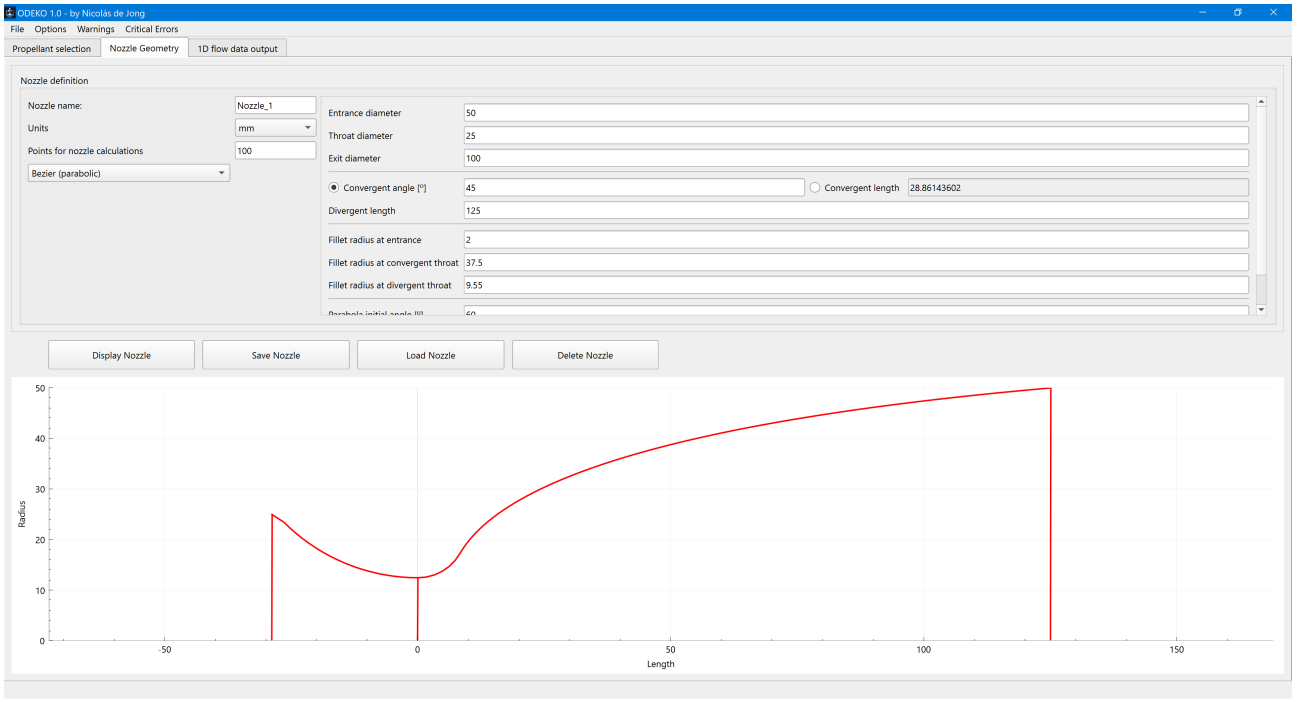


Figure 11: Empty nozzle geometry definition tab

5.1 Nozzle definition

In the upper left corner of the screen (as seen in Figure 11) the user must define the name of the nozzle, the units to be used for the parameters input, the number of points to be considered during calculations and the type of nozzle to be defined. The user might select between two nozzle types: conical or parabolic. The specific parameters that need to be inputted for each are shown on the upper right section of the screen and will be explained next.

5.1.1 Conical Nozzle

The parameters for the conical nozzle are shown in Figure 12. Note that the red lines are generated by ODEKO (nozzle shape) whilst the green and blue lines have been manually added to aid the user understand the various parameters. Fillets are not required to define the conical nozzle but can be optionally specified. Length and angle (for convergent and divergent regions respectively) are mutually exclusive. The user may define either length or angle and ODEKO will compute the other and display its value to the user.

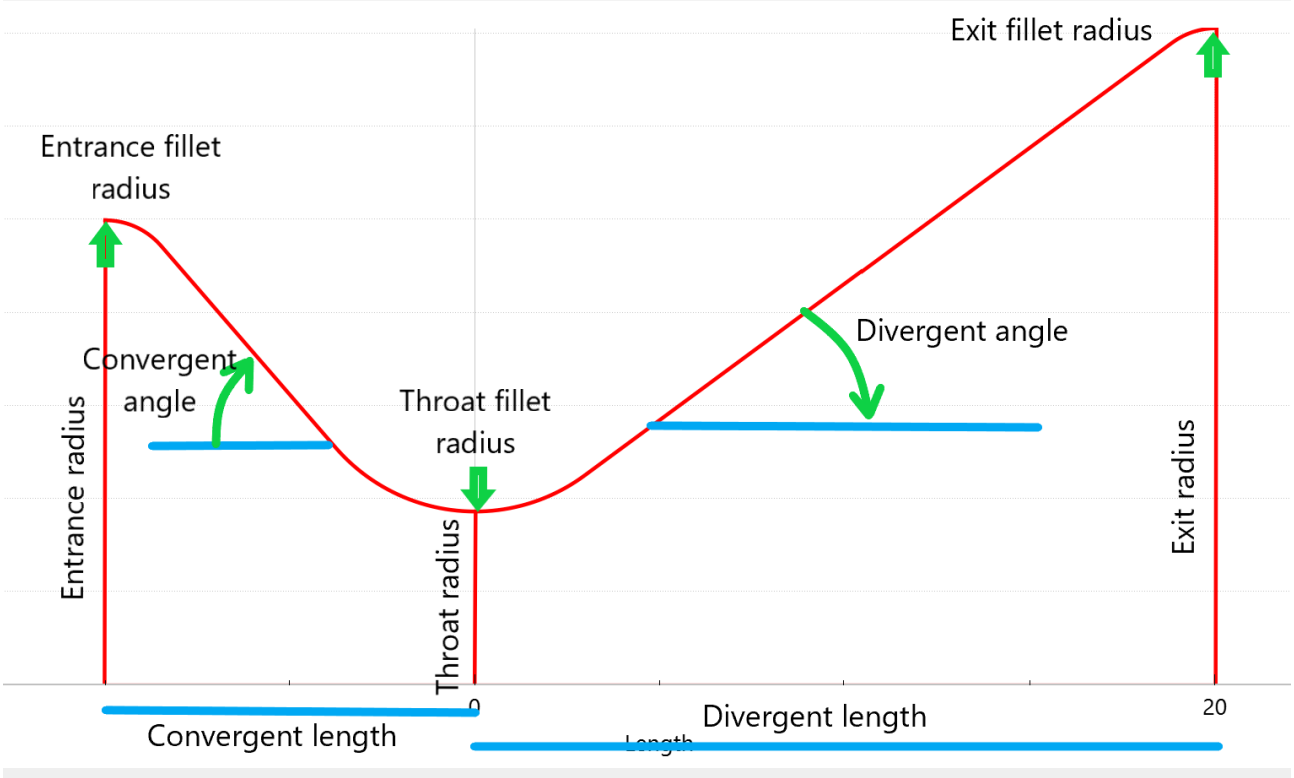


Figure 12: Example for conical nozzle

5.1.2 Parabolic Nozzle

The parameters for the parabolic nozzle are shown in Figure 13. Where the red lines are generated by ODEKO (nozzle shape) and the green and blue lines have been manually added to indicate the meaning of all parameters that need to be specified for the geometry definition. Note that convergent length and convergent angle are mutually exclusive, only one of them can be set by the user (the value for the other one will be computed by the application and displayed for the user).

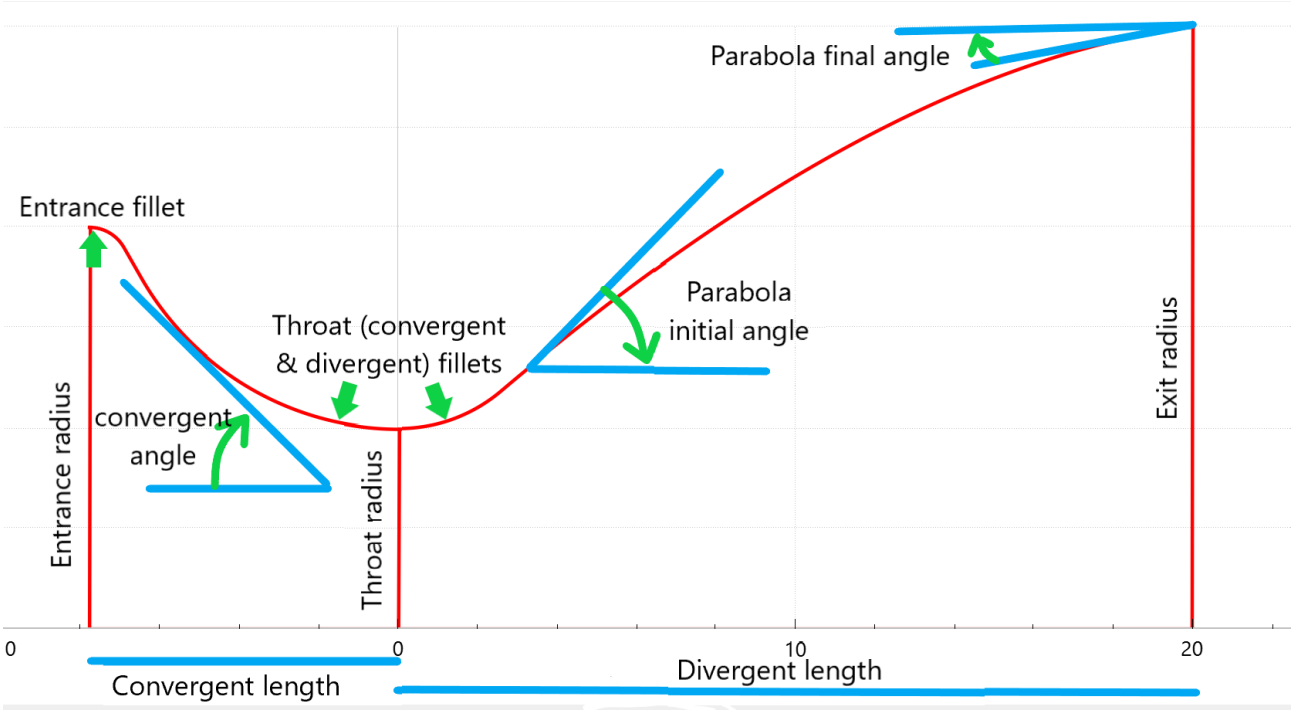


Figure 13: Example for bezier (parabolic) nozzle

5.2 Display options

In the preferences menu (see section 8) the user may select how the nozzle is represented. If only the curve is plotted, the throat will be tangent to the x axis. If the axis is also represented, the nozzle will be seen as shown in Figure 12 and Figure 13. Note that, in these figures, vertical lines in the entrance, throat and exit are present, this behaviour can be toggled off in the preferences menu as well. The user can also choose to represent the entire nozzle, which will show the section of the nozzle instead of just a quarter section. Finally, the user may also specify if they want the nozzle to be colorfilled or not. Note that all these options are merely aesthetic and have no impact on calculations.

5.3 Optimization

If the user is employing a solid propellant with known Vielle's parameters, combustion geometry and propellant density, the nozzle can be optimized for this configuration attending to one out of five target parameters. Note that this optimization required toggling on the solid motor simulation in the options menu (subsubsection 7.2.1). On Figure 11 the nozzle tab was shown with the solid motor options hidden, on Figure 14 this option has been untoggled, so the user can access the optimization menu.

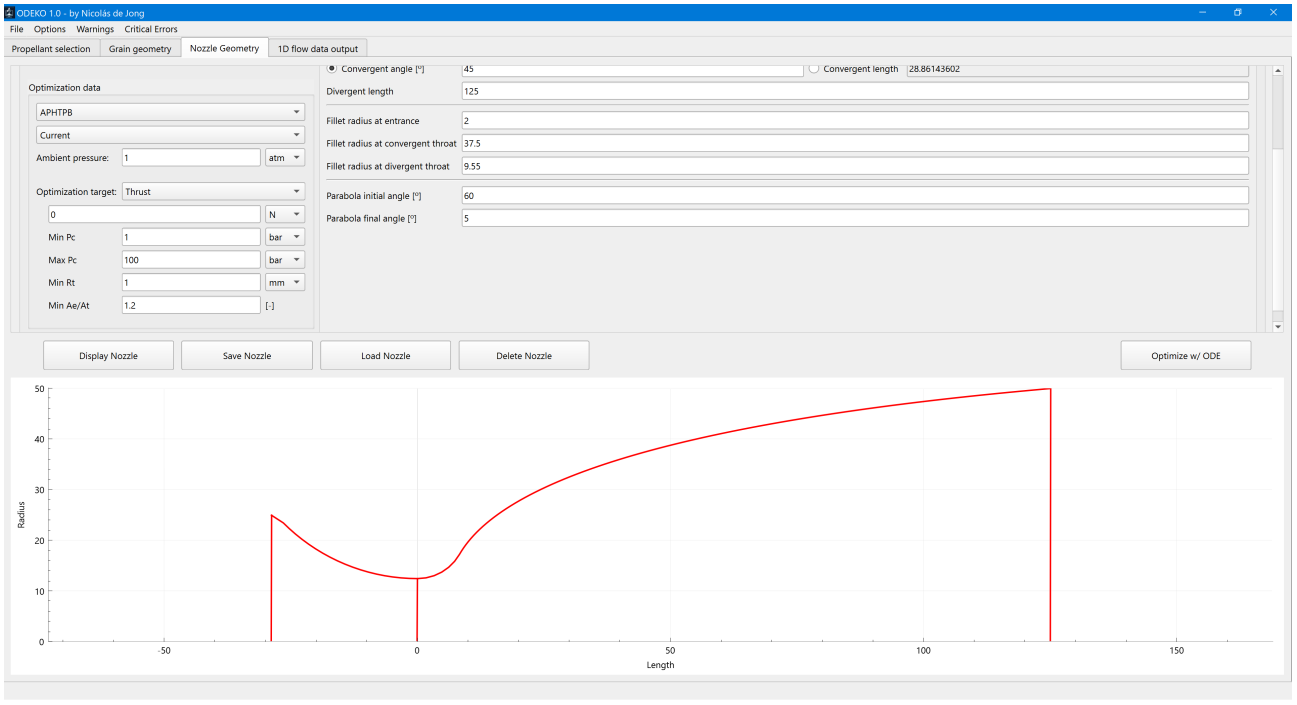


Figure 14: Nozzle tab with solid motor options shown

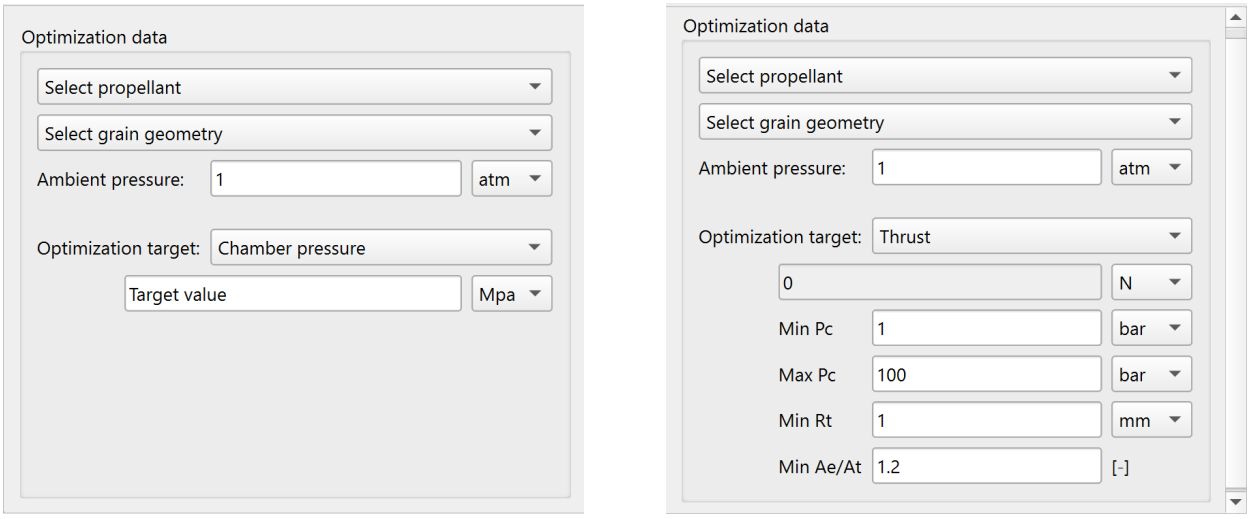
5.3.1 Chamber pressure & Combustion time

The user may specify a given chamber pressure or combustion time for the rocket engine. The application will compute the throat area that achieves said target value. The user might also choose if the nozzle is to be adapted or not. If not, the exit area is not modified since its not relevant for these two parameters and the value of the atmospheric pressure will have no effect in these optimizations. On the other hand, if the nozzle is adapted the ambient pressure will be used to dictate the exit area required (once the throat has been computed).

Note that the optimization for a given chamber pressure does not require any iteration, resulting in a very fast process. On the other hand, optimization for a given combustion time does require some iteration, but it still is a relatively fast procedure.

5.3.2 Thrust, specific impulse & total impulse

The user can optimize the nozzle so that it maximizes the thrust, specific impulse or total impulse value. ODEKO will compute the values for the throat area and exit area that achieve said maximum values whilst complying with the restrictions set by the user (minimum and maximum chamber pressure, minimum throat radius and minimum area relation). Note that it is necessary to specify the atmospheric pressure for which the optimization is taking place. It should be noted that all these optimization require many iterations and can be quite slow (specially if optimizing for total impulse).



(a) Parameters for thrust optimization

(b) Parameters for chamber pressure optimization

Figure 15: Nozzle optimization parameters

5.3.3 Abort optimization

After the optimization is started the "Optimize" button will be disabled. After a given time, if the optimization has not yet finished, the button will be substituted by an "Abort optimization" button. If the user presses the button the optimization will be halted and all its progress will be lost. The time until the abort button appears can be selected in the preferences menu (see subsubsection 7.2.6) and defaults to 15s.

The use of the abort button might be specially useful if the optimization is stuck (or specially slow) and the maximum number of allowable iterations was set to a large number (which can also be modified from the preferences menu). Nevertheless, for expensive optimizations such as thrust, specific impulse or specially total impulse, it is not strange to have to wait even a few minutes for the optimization to succeed. The abort method is not implemented for pressure optimizations, as they require no iterations and are almost instant. Note that the abort signal is only checked for at the start of every iteration, so it might take a few seconds from the push of the button until the process is fully stopped. If the process is not stopped even after pressing the abort button the user should report this as a bug.

5.4 Export

As explained in subsubsection 7.1.2, in every tab the user has the ability to export the present data, via a button in the file menu. Note that only one .csv file will be created, which includes all selected information. In the nozzle geometry tab, the user is able to choose which of the following sets of data are to be exported:

- Nozzle parameters: Exports to a .csv file all parameters related to the definition of the nozzle geometry.
- Nozzle points coordinates: Exports to a .csv file a table with the axial and radial coordinate of all points in the nozzle.
- Nozzle points areas: Exports to a .csv file a table with the area values for all points in the nozzle.
- Nozzle image: Exports to a .png file an image of the currently defined nozzle.

6 Calculations tab

The calculations tab allows the user to simulate the internal flow of a rocket engine nozzle, employing the propellant and nozzle (and optionally grain geometry) that were defined in the previous tabs. The optimization has multiple parameters that the user is able to modify, in order to best understand them it is recommended to refer to the full documentation. The main two simulation processes are chemical equilibrium and chemical kinetics.

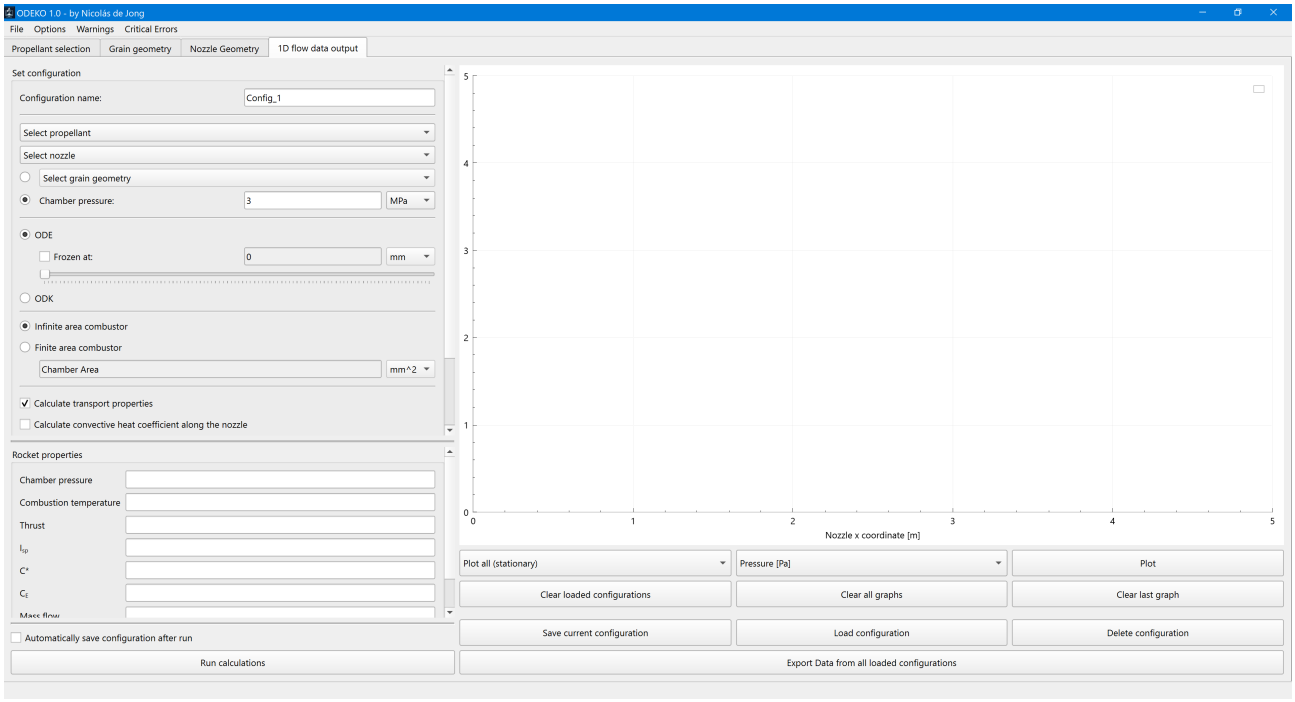


Figure 16: Empty calculations tab

6.1 Input information

A short explanation of the different input parameters of the rocket engine simulation is given in this section. Nonetheless, it may prove beneficial to consult the full documentation in order to properly understand their implications in the simulation process. All these parameters are inputted in the left section of the data tab, as can be seen in Figure 16. Note that the option to select a grain geometry will be hidden by default, as the options menu "hide solid motor simulation" is toggled on startup (see subsubsection 7.2.1).

- Propellant: Propellant to be used, as defined in the Propellant tab. Note that to use a defined propellant it needs to be saved first in its respective tab. Loading the propellant will also load the selected set of species and chemical reactions to be used in the calculations.

- Nozzle: Nozzle to be used, it can either be a previously saved nozzle or the nozzle currently displayed in the nozzle tab (this means the last nozzle that was plotted without errors).
- Chamber pressure / grain geometry: Here the user may select a constant chamber pressure for which to show results along the nozzle's axial coordinate or a grain geometry in order to compute the time evolution of the rocket engine's properties. The grain geometry to be selected can either be a previously saved one or the one currently displayed in the grain geometry tab (this means the last grain that was computed without errors). This is further explained in subsubsection 6.2.1 and subsubsection 6.2.2 respectively.
- ODE/ODK: Model to be used, either chemical equilibrium (which also allows to freeze the flow at a given point of the nozzle) or chemical kinetics. Note that ODK is much slower than ODE, although more precise (as long as the chosen kinetic scheme is precise itself). For more in-depth information about the differences and inner-workings of both computation methods please consult the full documentation.
- IAC/FAC: Indicates how to model the combustor, either with infinite area or finite area. If the user is not aware of these concepts it is recommended to always use IAC (FAC works best for liquid rocket engines with a well known injection velocity which needs to be inputted through the preferences menu).
- Calculate transport properties: Whether transport properties for the flow should be computed. Not performing this computation will result in a slightly faster simulation. Note that, in order to obtain valid transport properties, most of the mixture species need to have their transport properties defined in the database (refer to subsubsection 3.2.1).
- Calculate convective heat transfer coefficient: Whether the convective heat coefficient should be computed. Not performing this computation will affect very little the overall simulation time. In the preferences menu (section 8) many parameters for this computation can be edited by the user. Note as well that computing the convective heat transfer coefficient requires the computation of transport properties.
- Ambient pressure: The ambient pressure for which the various rocket parameters (such as thrust) are computed.
- Set uniform temperature for reactants: The temperature at which the propellant is before the combustion. This value will affect slightly the overall performance. Note that, in order for this parameter to have an effect on the simulation, the reactants must have their C_p defined (refer to subsubsection 3.2.2).

6.2 Computation

As mentioned in the prior section, the user may either choose a constant chamber pressure at which to run the simulation or a grain geometry. If a constant chamber pressure is chosen the simulation is considered 'stationary' and axial variables are shown. If a grain geometry is used the simulation is considered 'time dependant' and the time evolution of variables is presented.

6.2.1 Stationary

When a given chamber pressure is selected, the output is 'stationary'. This means that the chamber pressure is assumed constant and output is given for that specific scenario. The evolution along the nozzle axis of different variables is presented. The variables that can be plotted are: pressure, temperature, density, enthalpy, entropy, molecular weight, gamma, C_p, C_V, velocity of sound, fluid velocity, mach number, mass flow and the molar fraction of the various present species. Moreover, if transport properties are computed, the user may also plot conductivity, viscosity and Prandtl number. The user has also the option to calculate the convective heat transfer coefficient. Values in the combustion chamber are not present in the plots but are shown if the table display is selected (see subsubsection 6.3.2).

6.2.2 Time-dependant

If a grain geometry is specified (along with a solid propellant and its corresponding Vielle's parameters), the problem no longer is stationary. Due to the grain geometry information, an evolution of the chamber pressure relative to time can be computed. Therefore, the plots represent the evolution of key variables with time. Those variables are: Chamber pressure, combustion temperature, thrust, specific impulse, c^*, C_E and mass flow. In this case no variables along the nozzle axis are presented. Nevertheless, the user always has the option to select a given point in time (for which the chamber pressure is given by this calculation) and simulate that specific point to obtain the spatial evolution in the nozzle of the various variables. Note that in this case the values present in the rocket paremeters section (subsubsection 6.3.3) represent the average values over time.

6.2.3 Abort Calculations

After the calculation is started the "Run calculations" button will be disabled. After a given time (which can be selected in preferences menu and defaults to 15s), if the optimization has not yet finished, the button will be substituted by an "Abort calculations" button. If the user presses the button the calculation will be halted and all its progress will be lost. This might be specially useful if the computation is stuck (or specially slow) and the maximum number of allowable iterations was set to a large number (which can also be modified from the preferences menu).

The abort procedure is currently only implemented for ODK calculations, as complex kinetics schemes can be very computationally expensive. On the other hand, stationary ODE simulations are almost instant, so the abort button has not been implemented for them. For time-dependant simulations the abort behaviour is also present, as depending on the grain geometry they can be quite computationally taxing. Note that the abort signal is only checked for at the start of every iteration, so it might take a few seconds from the push of the button until the process is fully stopped.

6.3 Output

The output for the calculations tab can either be represented in a table or in a graph (the user may change this in the preferences window, refer to section 8). Moreover, the main characteristics of the engine are displayed in the lower left section of the screen.

6.3.1 Graph display

The graph is the option set by default and it allows the user to plot any of the given variables along the nozzle axis or over time (depending on the type of calculation performed). The user can plot as many variables as desired and either plot different configurations (different calculations made that session). This allows the user to compare the same variable for different calculation methods (such as between ODE and ODK) or compare how a change in the propellant or nozzle affects the results. A legend is automatically added, using the name of the configuration and the name of the plotted variable. The user can plot simultaneously different variable types, but their different scales might make the graph unusable. Nevertheless, the preferences menu (section 8) has an option to normalize all variables, making mixed plots more readable. On the preferences menu the user can also find an option to always shown the $y = 0$ point of the axis, instead of just the region where the variable is located.

When a calculation is finished, the currently selected variables is automatically plotted (either along the length of the nozzle or respect to time, depending on the calculation type). If a plot is already present, the new one will be added to the graph. Moreover, this calculation will be added to the list of current configurations, so it can be plotted at any time without the need to perform the calculation again (saving valuable computing time).

In order to plot a variable from an already computed configuration, said configuration needs to be selected, as shown in Figure 17. Note that the possibility to choose 'all stationary' or 'all time dependant' is also present, which will perform the plotting for all configurations that fit said category. Afterwards, the user will select the variable to be plotted at the drop-down list at the right of the previous one. Finally, if the plot button is pressed, the variable will be represented for all the selected configurations and the corresponding items will be added to the legend. This can be done any number of times and the new plots will be added to the existing ones.

The user may clear the last plot or all plots in the graph at any point in time. This clear will not eliminate the calculations made, allowing the data to be used again for plots (without needing to redo calculations that might be computationally expensive). The user can clear the session's calculations at any time as well, but in this case it will not be possible to plot them unless computed again. Clearing calculations is not required but might be useful if too many calculations are made and the list is becoming too large.

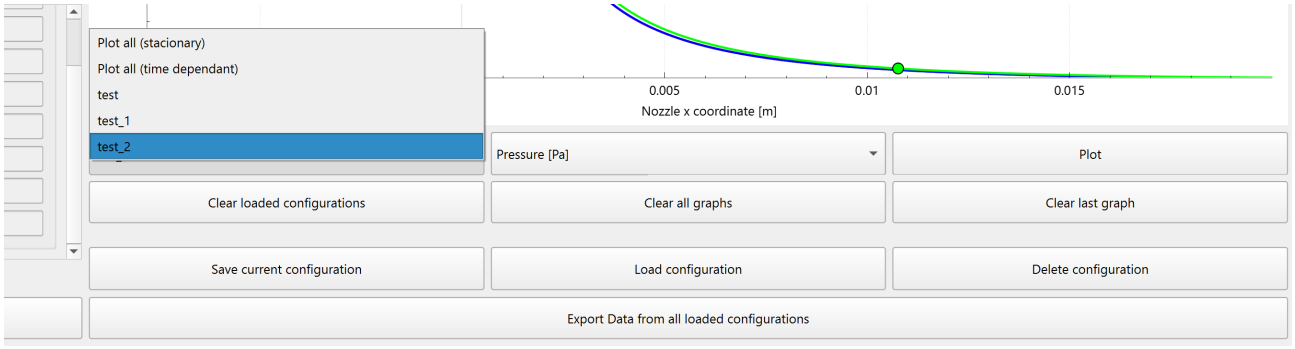


Figure 17: Selection of an already computed configuration

6.3.2 Table display

By default, the display tab is the aforementioned graph display. Nevertheless, the user may change at any time the display type to a table through the preferences menu (see section 8). This will change the look of the calculations tab from the one shown in Figure 16 to the one presented in Figure 18.

When using the tab display, all variables are shown simultaneously; therefore, the variable selector used for the graph display is disabled. As can be seen in the image, the table display also includes information about the combustion chamber, which was not present in the graph display. The table display only allows the data of one configuration to be shown, only the graph display can show multiple configurations at once. Nevertheless, in the table display the user can quickly select a different configuration and show its data on the table.

Note that, for variables that have not been computed (for example transport variables if the option is not selected) their respective cells will show N/A. This also applies to the velocity, mach number and mass flow in the combustion chamber, as these variables are not computed there for the model chosen in the image example. For more details on the computation method refer to the full documentation.

By default, only the variables at the combustion chamber, nozzle entrance, nozzle throat and nozzle exit are shown in the table. If the flow is frozen at a given point, the variables at said point will also be shown in the table. For time dependant computations the points at $t = 0, t = t_b/2, t = t_b$ will be shown, with t_b being the combustion time. If the user wishes to have the data at all computed points this can be achieved by exporting the current configuration, as explained in subsection 6.5.

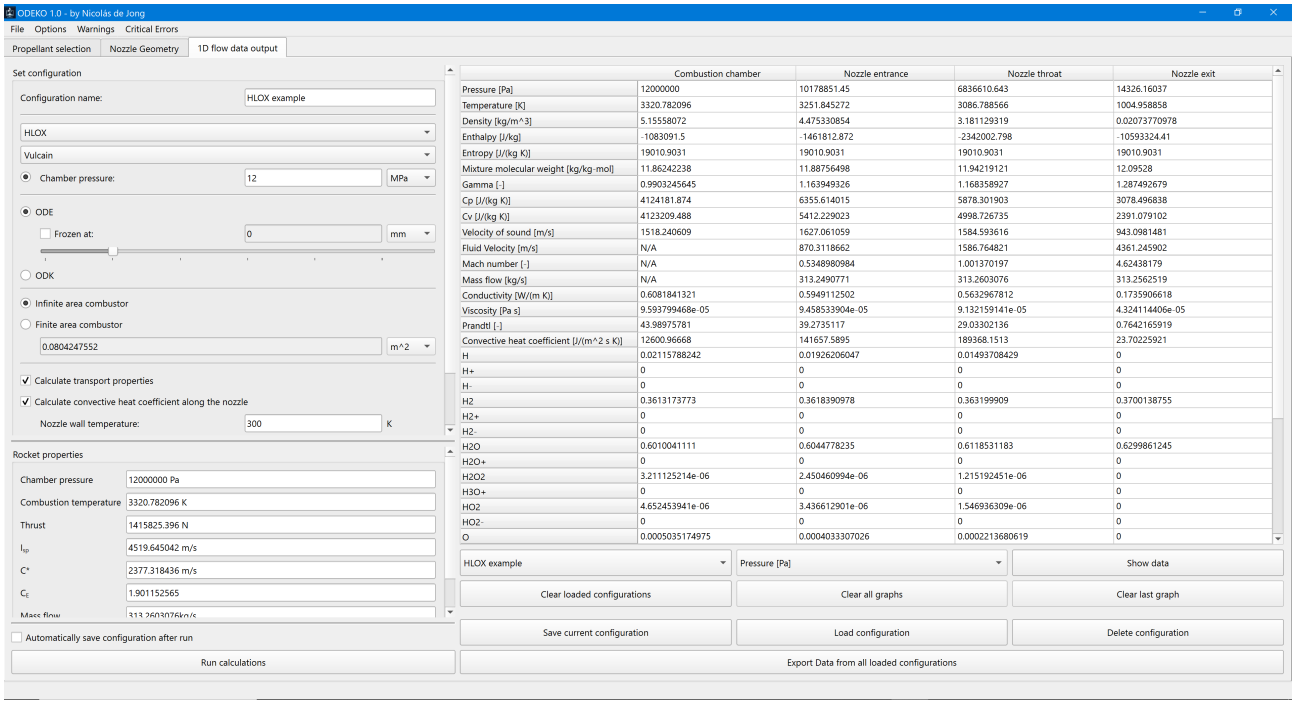


Figure 18: Table display in Calculations tab

6.3.3 Rocket parameters

The rocket parameters, shown in the lower left of the screen (Figure 16), are: chamber pressure, combustion temperature, thrust, specific impulse, c^*, C_E, mass flow and combustion time. For the stationary case, the combustion time is not given since there is no way to compute this value. For the time-dependant case, the combustion time can be computed using the specified grain geometry. Moreover, in this case, the average values for the rocket parameters considering the entire combustion are given.

The rocket parameters shown by the application correspond to the latest calculation performed or the latest selected configuration (whichever happened last). This means that the user can quickly change between the values of different configurations just be selecting them as shown in Figure 17.

6.4 Configuration save & load

If the current configuration is saved, only the input parameters will be saved (not the output nor results of the calculation). In the same fashion, if a configuration is loaded, it needs to be calculated afterwards in order to plot or use its results. The intention behind saving or loading a configuration is to be able to have a 'benchmark' with which to compare results that can be quickly loaded (so the user does not have to go through every input parameter). This is also useful if the user wants to perform small modifications to an already existing configuration.

6.5 Export

As explained in subsubsection 7.1.2, in every tab the user has the ability to export the present data, via a button in the file menu. For this tab, an export button is also present on the lower right corner. Note that only one .csv file will be created, which includes all selected information. In the calculations tab, the user is able to choose which of the following sets of data are to be exported:

- Parameters used for calculations: Exports to a .csv file all the parameters used in the calculation (this also includes propellant, nozzle and optionally grain geometry names).
- Configurations to be exported: The user must choose if they want only the currently selected configuration data to be exported or the data of all configurations. This will affect all other options present in the export.
- Graphs: The user may choose to export to a .png file the currently plotted graph or multiple ones. In the second case, the user can specify which types of graphs they want to export (thermodynamic, transport, composition, time dependant). Note that, if multiple configurations are being considered, only one graph of each type will be exported and it will contain the plots for all the considered configurations.
- Tables: The user may choose to export the output data in table format to a .csv file. The user may choose to export thermodynamic, transport and/or composition data. By default this only exports data for the key points of the nozzle (combustion chamber, entrance, throat, exit and frozen point if present), but the user can specify that the data should be exported for all points of the nozzle instead. For time dependant calculations the user only needs to specify that all points should be exported, the other options will be ignored.
- Rocket parameters: Exports to a .csv file the rocket parameters of the current/all configuration/s (as selected). If the configuration is time dependant, the export takes the mean values.

7 Menu

The menu can be accessed at the top of the screen, no matter in which tab the user is at, since it is always present on top of the tap content. There are four main sections of the menu each with their own subsections, which are explained in the subsequent sections of the manual.

7.1 File

7.1.1 File changes

In the file menu, the user may change which files are used to store the data of the different components of the application (known as save-files, as explained in subsection 10.1). The user may select any of the save file types and change its associated file, which will from there on be used to retrieve and save its corresponding data. This is of special use when exchanging component information with a colleague or if the user wishes to have two differentiated sets of component data. It is also very useful since the user can give their entire configuration data along with a bug report, to ensure replicability.

Note that, keeping the configurations file but changing other may result in inconsistencies, where a configuration is not able to be loaded due to missing information (said components where located in the default save files but not in the new selected ones).

The preferences file might also be changed in the same fashion. This way the user will be able to share specific set of preferences or quickly change between two configurations.

7.1.2 Export

The Export button, inside the File Menu, allows the user to export data from the current tab to an output file. The data that the user may choose to export is different for each tab and it is explained in each corresponding section: subsection 3.6 subsection 4.3 subsection 5.4 subsection 6.5. Specific information for the output files can be found on those sections, whilst a more general overview is given in subsection 10.3.

7.1.3 Input information

There are three buttons in the File menu, regarding the formats for the three input data files (thermo, transport & kinetics). These formats are further explained in section subsection 10.2.

7.1.4 About ODEKO

If the user clicks on it, a window with a short description of the application will be shown. This information is very similar to the introduction of this manual.

7.1.5 Open Documentation

There are two buttons on the file menu which open the User Manual and the Full documentation respectively. The User Manual is this one, whilst the full documentation contains the entire User Manual as well as the Theory documentation and some other relevant information about ODEKO.

7.2 Options

7.2.1 Hide solid motor options

As mentioned in section 4, by default all options related to solid motor simulations and grain geometry are hidden. The user might untoggle this checkbox in the options menu in order to enable these additional features. Furthermore, the user might change the default behaviour in the preferences menu (subsubsection 7.2.6) in order to have this option toggled off by default. The hidden features related to solid motor simulation include:

- The entire grain geometry tab, which allows the user to define and save grain geometries.
- The propellant properties section in the propellant tab, which allows the user to specify vielle's parameters and the propellant density (subsection 3.4).
- The optimizations in NozzleTab, which allow the user to optimize the nozzle using a given grain geometry for the desired pressure or combustion time or to maximize thrust or impulse (subsection 5.3).
- The time-dependant simulations in the Calculations tab, which show the time evolution of the rocket parameters for a given grain geometry (subsubsection 6.2.2).

the entire grain geometry tab, the option to specify vielle's parameters and

7.2.2 Baby mode

Referred to in-application as 'Baby mode'. It is a simplified mode for the application that can be toggled through the options menu. It severely limits the capabilities of the program, hiding many of its functions (such as ODK calculations). Its main objective is to provide a simpler and less confusing interface for students using this program. In preferences the user may select if 'baby mode' is active by default and if the preferences are accessible in this mode. Moreover, the user may also lock the application in 'baby mode' which would disable the ability to toggle it off. This may be useful if a teacher is installing the program in a students computer and does not want this mode to be toggled off. The only way to undo the lock is to manually edit the preferences file and set it to false. A comparison between normal mode and baby mode for the propellant tab is shown in Figure 19.

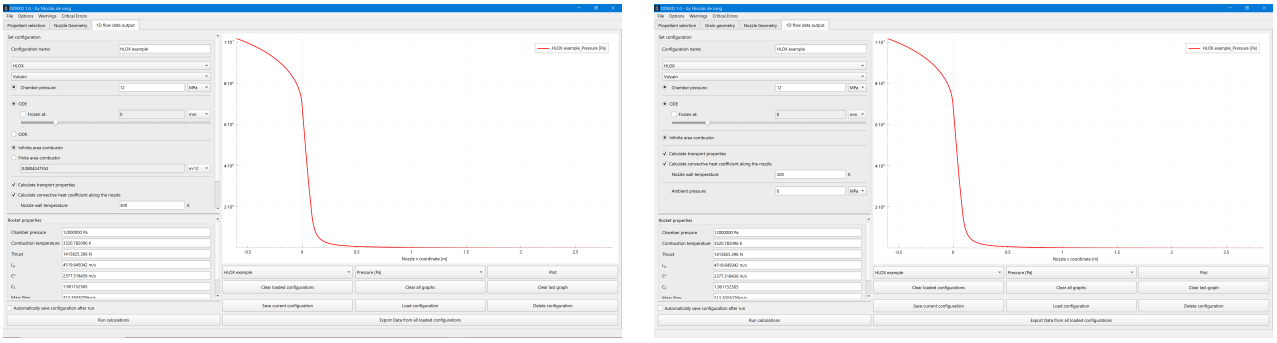


Figure 19: Comparison between normal mode and baby mode for the calculations tab

7.2.3 Visual changes

There are two visual changes available in the options menu. Firstly, the user may toggle on or off the visibility of the status bar. The status bar provides tips about the functionality of certain buttons or parts of the application when the user hovers over them. Therefore, for a new user it is useful to have the status bar visible, whilst for a more experienced one it might become redundant.

On the other hand, the user may also turn on or off a 'night mode' for the application. Said mode changes the color palette to a darker tone, in place of the default light one. This change is only aesthetic and has no other impact, is up to user preference. A comparison of both palettes can be seen in image 20.

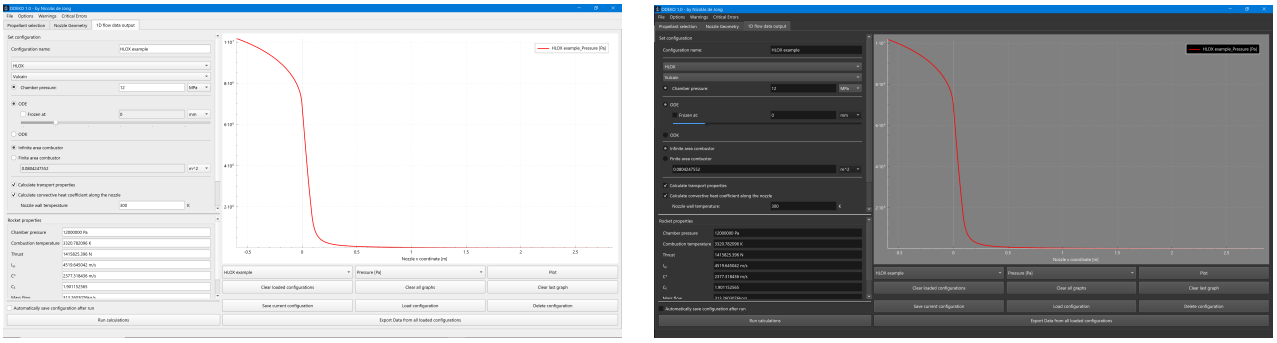


Figure 20: Comparison between light mode and dark mode for the calculations tab

7.2.4 Clear Warnings & Errors

In the options menu the user may clear the current warnings or the current critical errors. Note that this is an option that comes with a certain risk. Clearing the critical errors will allow the user to perform certain actions or calculations that would regularly be forbidden due to the errors occurred; therefore, it is only advised to clear them if the user knows for a fact that said errors will not result in any malfunction of the program. It is best to try to fix the errors rather than clear them. In the case

of warnings, clearing them does not have much of an impact, and it is done automatically in between calculations.

7.2.5 Update Warnings & Errors

Generally speaking, warnings and errors are updated automatically after events that might trigger them. Nevertheless, the user has the option to trigger a manual update of warnings and errors through the options menu. This might be of use if the user suspects that an error/warning happen which was not displayed.

7.2.6 Preferences

The preferences window can be accessed through this button, further information can be found in section 8.

7.3 Warning & critical erros

This two sections of the menu show the current warnings and critical errors respectively (those that occurred in the last calculation or action). Clicking on them will show a complete list of all warnings/errors along with longer descriptions, this list can also be seen in section 9.

8 Preferences

The preferences window may be accessed from the Options Menu (section 7). Here the user may modify different aspects of the application. Each tab of the preferences window groups items related to the same aspect of the application. The user may change any of the settings, changes will not be applied until the 'save' button is clicked. The user always has the option to discard the changes just made or to restore all preferences to their default value (as seen in Figure 21). Closing the window through the 'x' button will result in all changes being discarded.

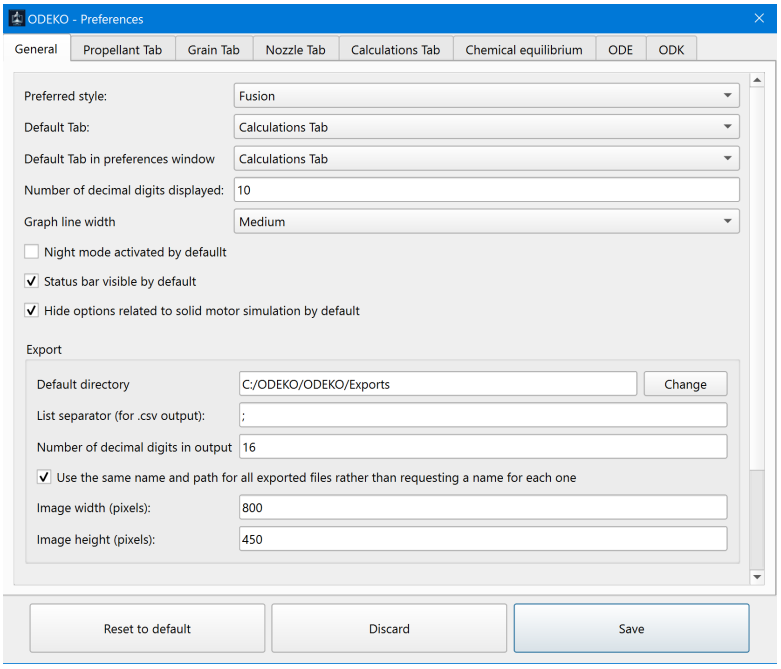


Figure 21: Preferences window

The default settings is what the creator of this application believes to be best for the average user. Therefore, it is advised to not change them unless the user has proper knowledge of its implications. Some of the preferences relate to visual aspects of the application, which can be modified without much regard. On the other hand, some preferences affect key aspects of the calculation process, and changing them may result in catastrophic consequences. In the following subsections, preferences marked with '*' are best not changed unless the user has a deep understanding of the theoretical aspects behind the calculations performed. It is recommended to thoroughly read the theory manual of ODEKO before modifying any preferences related to the calculations of the application.

8.1 General

- Preferred style: The application currently offers two visual styles: 'classic windows' or 'Fusion'. This change is only visual and will not have any other implications. Note that currently, dark mode is only supported for 'Fusion' style. Defaults to "Fusion".
- Default tab: Whenever the application is opened this will be the tab that is shown by default. This may proof specially useful top users that already have all their components designed (propellant, grain & nozzle) and only wish to perform calculations. Defaults to "Calculations Tab".
- Default tab in preferences window: Whenever the preferences window is opened, this will be the tab shown by default. This may proof useful to users that change a certain preference very often. Defaults to "General".
- Number of decimal digits displayed: This will affect all numeric displays present throughout the application. It is only a visual change and does not alter the numeric precision of the calculations performed by the program. Defaults to 10.
- Graph line width: This will affect all graphs throughout the application. It is only a visual change. Defaults to "Medium".
- Night mode activated by default: Whether the application should start on night mode by default. Note that this does not forbid the user from toggling it on/off during the session. Unchecked by default.
- Status bar visible by default: Whether the status bar should be visible by default. Note that this does not forbid the user from toggling it on/off during the session. Checked by default.
- Hide options related to solid motor simulation by default: As mentioned in section 4, the grain geometry tab is hidden by default when starting the application. Toggling this option off will result in the grain tab being visible by default when the application is launched. This also affects the other features related to grain combustion, such as Vielle's parameters in propellant tab (subsection 3.4), optimizations in nozzle tab (subsection 5.3) and time dependant simulations in the data tab (subsubsection 6.2.2). Note that, independently of whether this preference is toggled or not, the options related to solid motors can be toggled on or off at any time from the options menu (subsubsection 7.2.1), this preference only affects the startup behaviour. Checked by default.
- Default directory: The directory to be opened by default when an export file needs to be selected. Note that this only affects the first export of the session, as after it the folder that will be opened is the folder in which the last export was saved. Restarting the software will result in this default directory being opened when making an export again. Defaults to "INSTAL_PATH/Exports".

- *List separator: As explained in 10.3, this character is used to separate columns in the '.csv' output files. Said character should coincide with the one used by the application that reads said file (for example Excel). Defaults to ";".
- Number of decimal digits in output: Here the user might specify how many decimal digits are written in the output file when performing an export. Defaults to 16.
- Use the same name and path for all exported files rather than requesting a name for each one: As explained in 10.3, if this is chequed the program will only ask once for a output file name (even if various files are outputted). If it is unchecked, every output file will require the user to name it. Checked by default.
- Image width & height: Size (in pixels) of figures outputted by ODEKO. Note that all figures are exported in '.png' format. Defaults to 800p width and 450p height.
- Baby mode active by default: Whether 'baby mode' is active by default when the application is run. This does not forbid the user from toggling it on/off during the session. Unchecked by default.
- Block access to preferences in babymode: If checked, it will prevent the user from accessing the preferences menu if 'baby mode' is toggled. Unchecked by default.
- *Lock this application in babymode: If this item is selected the application will be locked in 'baby mode'. It will always start by default in 'baby mode' and said mode can not be untoggled from the menu nor from the preferences menu. The only way to disable 'baby mode' once this item is toggled is to manually change the preferences file. Unchecked by default.

8.2 Propellant tab

- Default reactant temperature: Temperature at which the reactants are before the combustion to be used by default in calculations. Note that, if a reactant is defined only for a single temperature this value will not have any effect on it (since its enthalpy is only defined for a single value). Defaults to 298.15 K (recommended, as most enthalpies of formation are given for said temperature).
- Allow limit for top pressure range in vielle's parameters: Whether the last range of Vielle's parameters should have an upper pressure limit or not. This may be useful if the definition is only valid for a given interval since, if the upper limit is set, the user will get a warning if it is surpassed during calculations (see 9.1). Checked by default
- Minimum & maximum pressure in c^* graph: Upper and lower limits for the x-axis in the c^* vs pressure graphs present in the Propellant tab. Defaults to 5 and 100 bar respectively.

- Number of points in c^* graph: This will impact the time it takes to compute and represent the c^* vs pressure graphs present in the Propellant tab. Defaults to 100.
- First to second reactant minimum & maximum ratio: Upper and lower limits for the x-axis in the c^* vs mixture ratio graphs present in the Propellant tab. Defaults to 1 and 10.
- Number of mixture ratios to be tested: his will impact the time it takes to compute and represent the c^* vs mixture ratio graphs present in the Propellant tab. Defaults to 200.
- Chamber pressure for optimization: The optimal mixture ratio given is valid for a specific chamber pressure, which is here specified. Note that the optimal point does not tend to vary much with pressure (the value for the c^* will change, but the optimum mixture ratio will hardly do so generally speaking). Defaults to 10 bar.
- By default only show species/reactions that are compatible with current reactants: If it is checked the user will only see species/reactions that are compatible with the reactants when the database is opened. Note that the database has a checkbox itself to change this behaviour, this preference only affects the default state. Checked by default.
- When all species/reactions are shown, disable those which are incompatible with current reactants: If checked, incompatible species will shown as disable, this not only offers a visual difference between compatible/incompatible species but it also prevents the user from selecting incompatible ones. If the user wishes to edit any specie/reaction without worrying about the current reactants, this item is best left unchecked. Checked by default.

8.3 Grain tab

- Combustion front evolution of the charge: Specifies which combustion fronts are shown in the upper right image in the grain tab. This change is only visual, has no effect on computation. By default shows radial and axial perspectives.
- Maximum number of iterations: Maximum number of iterations performed in order to find the chamber pressure when the computation involves a grain geometry (and therefore a propellant with properly defined Vielle's parameters). Defaults to 50.
- *Tolerance for convergence: Tolerance applied for the convergence towards a chamber pressure compatible with the grain and nozzle geometries and propellant characteristics. Note that altering this value will impact the results. A low tolerance (¿¿0) will give erroneous results, whilst a too fine tolerance will imply long computation times (or it might not even converge). Defaults to 0.0004.
- Pressure initial estimate: Initial estimate for the chamber pressure in grain combustion problems. Defaults to 20 bar.

8.4 Nozzle tab

- Nozzle display: Offers different ways to perform the nozzle display. It can be fully displayed (half section); only a quarter section; or only the curve (the throat will be contained in the x axis). This change is only visual, but this also decides how the nozzle figures are exported. Defaults to "curve + axis".
- Add vertical lines: If checked, vertical lines at the entrance, throat and exit of the nozzle will be painted. This change is only visual but it will be present in exported figures. Checked by default.
- Colorfill nozzle: Whether the nozzle interior is colorfilled or not. This change is only visual, but it will be present in exported figures. Unchecked by default.
- Optimize with: Method employed in optimization, currently only IAC (Infinite Area Combustor) is supported. Refer to the theoretical background documentation for more information about these methods. Defaults to IAC.
- Area to be used in optimizations: Employed in thrust, specific impulse and total impulse optimizations. Initial combustion area is recommended if a thrust target is required for take off. Final combustion area is recommended if a thrust target is needed at the end of propulsion. Maximum area will maximize the maximum thrust value, whilst minimum area will maximize the minimum thrust value. Average combustion area is generally the best choice, since it will offer a good optimization for the entire combustion (given a near-neutral combustion area). Defaults to "Average combustion".
- Initial temperature for reactants in optimization: Temperature at which the reactants are considered to be prior to the combustion for the optimization process. This only affects reactants that are not single defined. Defaults to 298.15 K (recommended value as most enthalpies of formation are defined for this temperature).
- Maximum number of iterations for optimization: Some optimizations may require a high number of iterations or a long computation time (specially total impulse optimization). This setting allows the user to limit it. Defaults to 200.
- *Tolerance for target value in optimization: This should be change depending on the precision with which the user wants to achieve the specified chamber pressure or combustion time. Defaults to 0.01.
- *Number of points used in particle swarm optimization: particle swarm is used for thrust, specific impulse and total impulse optimizations. A greater number will offer more reliable results, but it will increase computation time for the same number of iterations. Defaults to 10.

- Freeze flow at the throat in optimizations: Whether the optimizations should consider that the flow is frozen at the throat of the nozzle. This setting is recommended for solid propellants, since it generally offers a better approximation than chemical equilibrium throughout the entire nozzle. Checked by default.

8.5 Calculations tab

- Type of display: Whether the data is displayed in a graph or in a table. This change is only visual and does not affect the export options. Note that graphs may represent more than one configurations, whilst tables cannot. On the other hand, tables clearly represent all variables simultaneously, as well as show the values in the combustion chamber. Changing this option does not remove any calculations performed, so it can be changed to view the same simulation (or set of simulations) in both ways. Defaults to Graph.
- Number of seconds before "Abort calculations" button appears: If calculations take too long to finish the "Running..." button is substituted with an "Abort calculations" button, which allows the user to stop the calculation if desired. This preference allows the user to modify how many seconds it takes the abort button to appear. This button is also present when performing optimizations in the Nozzle tab. More information on the abort behaviour can be found in subsubsection 5.3.3 and subsubsection 6.2.3. Defaults to 15s.
- Normalize values plotted in graph: If checked all plotted values will be of order one. Generally speaking values will be normalized using the maximum value of the plot or a characteristic value (such as the chamber pressure). Some values will not be affected (such as the Mach number or mass fractions) as they already are of order 1. Unchecked by default.
- Always start graph y-axis at 0: Whether the 0 value should always be visible in the y-axis. Note that this may render some graphs useless (they will appear as straight lines due to the scaling that would need to be applied). Unchecked by default.
- Show freeze point in graphs: Whether the point at which the flow was frozen should be marked in the graph or not. It is recommended to do so, since it offers more understandable graphs & output. Checked by default.
- Show points at which ODK changes from pressure to area: Whether to mark the two points in between which the throat iteration is performed for ODK problems. For more information on why these points are relevant refer to the theory documentation. Checked by default.
- Only plot frozen variables even if the flow is in equilibrium: Generally, the variables of interest for the user are the equilibrium variables. Nevertheless, this depend on which equations are these variables being used in, for some application frozen variables might be needed instead of

their equilibrium counterparts. For a more in-depth explanation on the difference between both types refer to the theory manual. Unchecked by default.

- Derivative & Interpolation precision: A higher precision increases the number of interpolation points used, each step increases the number of points by two. The 'Low' setting only uses two points. Defaults to "High".
- Integration method: Method to be employed. Further information about these methods can be found in the documentation for the theoretical background of the application. Defaults to "Trapezoidal rule".
- Flow velocity in combustion chamber: Only used to compute the convective heat transfer coefficient in the combustion chamber. Defaults to 30m/S. For more information on how the convective heat transfer coefficient is computed and the relevance of the parameters that will be presented subsequently, refer to the theory manual.
- *Coefficient for Prandtl-Reynolds product: The Nusselt number is approximated as seen in equation 1, as a basis to calculate the convective heat transfer coefficient. This coefficient is represented as 'c'. Defaults to 0.026.
- *Exponents for Reynolds number: Exponent 'm' in equation 1. Defaults to 0.8.
- *Exponent for Prandtl number: Exponent 'n' in equation 1. Defaults to 0.4.
- *Viscosity exponent for temperature: In the limit layer, viscosity is considered proportional to an exponent of temperature as seen in equation 2. Defaults to 0.6.
- *Conductivity exponent for temperature: In the limit layer, conductivity is considered proportional to an exponent of temperature as seen in equation 2. Defaults to 0.6.
- *Cp exponent for temperature: In the limit layer, cp is considered proportional to an exponent of temperature as seen in equation 2. Defaults to 0.
- *Curvature correction exponent: Exponent for the curvature correction term in Bartz equation, used to calculate the convective heat transfer coefficient. Defaults to 0.1.

$$Nu = c \cdot Re^m Pr^n \tag{1}$$

$$\mu \propto T^{\omega_\mu} \quad ; \quad \lambda \propto T^{\omega_\lambda} \quad ; \quad C_p \propto T^{\omega_c} \tag{2}$$

8.6 Chemical equilibrium

This section can be better understood if the user has carefully read the documentation on the theoretical background of ODEKO.

- Maximum number of iterations for equilibrium calculation: Generally convergence is achieved in very few iterations. Nonetheless, for combustion containing condensed phases, a greater number of iterations may be required. Therefore this value is advised to be kept high. Although it should not be too big in case the program gets stuck and is unable to find convergence (it would imply a great amount of time before it is forced to exit calculations). Defaults to 8000.
- Maximum iterations for ion balance convergence: After convergence is achieved, a number of iterations are performed until ions are balanced. Should take a low amount of iterations. Defaults to 100.
- Minimum composition for specie to be considered: Species with compositions lower than this value will be considered to not be present in the mixture (therefore, computation time is not wasted on them). Defaults to 1e-10.
- Minimum composition for ion to be considered: Ions with compositions lower than this value will be considered to not be present in the mixture (therefore, computation time is not wasted on them). Defaults to 1e-10.
- *Tolerance for temperature convergence: The user should not change this unless a proper understanding of the theoretical background is possessed. Defaults to 0.0001.
- *Tolerance for entropy convergence: The user should not change this unless a proper understanding of the theoretical background is possessed. Defaults to 5e-5.
- *Tolerance for molecular weight convergence: The user should not change this unless a proper understanding of the theoretical background is possessed. Defaults to 5e-6.
- *Tolerance for species convergence: The user should not change this unless a proper understanding of the theoretical background is possessed. Defaults to 1e-6.
- *Tolerance for ion convergence: The user should not change this unless a proper understanding of the theoretical background is possessed. Defaults to 0.0001.
- *Tolerance for species with extremely low composition: The user should not change this unless a proper understanding of the theoretical background is possessed. Defaults to 0.001.
- Seed temperature for combustion chemical equilibrium: Temperature used as initial estimate of combustion temperature. If calculations converge correctly, all seed temperatures should give the same result, it only affects the number of iterations it takes to achieve it. Defaults to 3800K.

- Initial estimate for mixture composition: Initial estimate for the inverse of the molecular weight of the mixture. Defaults to 0.1.
- *Initial value for introduced condensed species: When a condensed specie is introduced, it will be considered to have the composition here specified. Defaults to 1e-5.
- *Specie composition at forced reset: When a reset is forced (due to a singularity during calculations) this is the composition to which species will be reset. Defaults to 1e-6.
- *Ion composition for ion reset: When an ion reset is forced this is the composition to which species will be reset. Defaults to 1e-5.
- *Collision cross-section ratio (A*): Used for transport properties computation, user should not need to change it. Defaults to 1.1.

8.7 ODE

This section can be better understood if the user has carefully read the documentation on the theoretical background of ODEKO.

- *Tolerance for throat convergence: The user should not change this unless a proper understanding of the theoretical background is possessed. Defaults to 0.004.
- *Tolerance for pressure convergence: The user should not change this unless a proper understanding of the theoretical background is possessed. Defaults to 4e-5.
- *Tolerance for temperature convergence (frozen flow): The user should not change this unless a proper understanding of the theoretical background is possessed. Defaults to 5e-5
- Maximum iterations for throat convergence: Generally should only require a small number of iterations. Defaults to 20.
- Maximum iterations for pressure convergence: Generally should only require a small number of iterations. Defaults to 10.
- Maximum iterations for temperature convergence: Generally should only require a small number of iterations. Defaults to 10.
- Maximum iterations for frozen throat convergence: it may take a greater number of iterations than the other computations. Nonetheless, it it is prematurely terminated, the precision lost can be generally disregarded. Defaults to 20.
- Flow velocity at injector plane: Specifying this value accurately will also improve the accuracy of the FAC simulations. Defaults to 0 m/S (injector velocity disregarded).

- *Pinf/Pi forced reset for FAC problem: In very rare circumstances the FAC problem may need to force a rest for the pressure relation at the throat, this value is given here. Defaults to 1.0001 (value should be kept very slightly over 1).
- *Minimum area difference to consider new initial point in FAC: If the given combustor area differs too much from the nozzle entrance's area, a new point is created for the computation. Defaults to 1e-5.
- *X percentage offset for new initial point in FAC: The offset of the new point mentioned in the previous item. Defaults to 1.05.
- *Tolerance for FAC problem convergence: This should not be changed by the user unless a deep theoretical understanding is possessed. Defaults to 2e-5.
- Maximum number of iterations for FAC Pinf calculation: Depending on the problem this may take some iterations to converge. Defaults to 20.

8.8 ODK

This section can be better understood if the user has carefully read the documentation on the theoretical background of ODEKO.

- *Differential equation solver relative tolerance: This value greatly impacts the maximum tolerances that the ODK calculations are able to handle. it is not recommended to be outside the range of 1e-10 to 1e-13. It does not have a significant impact on computation time. Defaults to 1e-10.
- *Differential equation solver absolute tolerance: It is not an specially important tolerance value. Nonetheless, it is not recommended to change its default value. Defaults to 1e-6.
- Maximum iterations for global convergence: This number should be big, since it encompasses all other iterations during ODK calculations. It is also very common for ODK computations to take a great amount of iterations, specially for certain propellants. Defaults to 400.
- Maximum iterations for throat convergence: This iteration is performed at the end, therefore, it should not have a high maximum amounts or it could significantly impact computation time. Defaults to 20.
- Maximum iterations for mass flow convergence: This is the most basic iteration, it is required to converge many times; therefore its maximum iterations should not be too high in case it gets stuck by an unforeseen issue. Defaults to 20.
- Maximum iterations for near-throat points convergence: The maximum iterations should be similar to those set in the previous two items. Defaults to 40.

- *Area relation for throat iteration start/end: It is recommended for the throat iteration to have a significant number of points, although not too many (it slows computation). For a nozzle of about 200-800 points, the default values are more than sufficient. Increments to those values should not change the obtained results. If it does, an error must have occurred during computation. Defaults to 1.08 for iteration start and 1.12 for iteration end.
- *Tolerance for throat convergence: Finer tolerances than the default values are not generally possible to be achieved. If the solver tolerance is worsened, this tolerance should be changed accordingly. Defaults to 0.0001.
- *Tolerance for near-throat points convergence: Finer tolerances than the default values are not generally possible to be achieved. If the solver tolerance is worsened, this tolerance should be changed accordingly. Defaults to 0.0001.
- *Tolerance for mass flow convergence: Finer tolerances than the default values are not generally possible to be achieved. If the solver tolerance is worsened, this tolerance should be changed accordingly. Defaults to 0.0001.
- *Tolerance for throat position convergence: Finer tolerances than the default values are not generally possible to be achieved. If the solver tolerance is worsened, this tolerance should be changed accordingly. This tolerance is generally the hardest one to achieve out of the four. Defaults to 0.0001.
- *Area relevance for mass flow correction: The effect of this value can be seen in the theoretical background. This values are not fully optimized, so the intrepid user may change them to try to find better ones. Note that this may completely break convergence. Defaults to 150.
- *Maximum pressure correction for near-throat points: The effect of this value can be seen in the theoretical background. This values are not fully optimized, so the intrepid user may change them to try to find better ones. Note that this may completely break convergence. Defaults to 0.01.
- *Throat position relevance for pressure correction: The effect of this value can be seen in the theoretical background. This values are not fully optimized, so the intrepid user may change them to try to find better ones. Note that this may completely break convergence. Defaults to 0.1.

9 Warnings & critical errors

9.1 Warnings

This subsection contains a complete list of all possible warnings given by the program, along with a short explanation of what may have caused them and possible solutions. Warnings let the user now about a situation that occurred during calculations. It does not need to be a problem, although the user should check them in order to know if the results obtained may have been compromised.

- SP00: Unknown error. Its occurrence is generally due to a programming mistake, it should never happen.
- SP01: The temperature of the mixture was below the range of definition of C_p for one or more species. The value of C_p is assumed to be that of the smallest temperature in the range of definition.
- SP02: The temperature of the mixture was over the range of definition of C_p for one or more species. The value of C_p is assumed to be that of the biggest temperature in the range of definition.
- SP03: Equivalent to SP01 but related to viscosity range of definition.
- SP04: Equivalent to SP02 but related to viscosity range of definition.
- SP05: Equivalent to SP01 but related to conductivity range of definition.
- SP06: Equivalent to SP02 but related to conductivity range of definition.
- SP07: Computing of viscosity was attempted for a specie without a viscosity definition. This warning occurs when transport properties are computed and one or more species only have conductivity defined.
- SP08: Computing of conductivity was attempted for a specie without a viscosity definition. This warning occurs when transport properties are computed and one or more species only have viscosity defined.
- SP09: One of the species used was defined only for a single temperature value. The definition for said value is used for all values. The most likely outcome is that the calculations performed with this specie are invalid.
- SP10: Reached maximum number of iterations for equilibrium calculation. This may happen if the maximum iterations are limited to a very low amount or if the mixture is bad-behaved and has convergence problems (specially if a lot of condensed species are involved).
- SP11: Transport properties calculation was chequed in the configuration, but no species present

in the mixture have transport properties defined in their database. This supposes no error, but is recommended to uncheck the transport properties calculation in order to gain computation speed.

- SP12: Reached maximum number of iterations for ion balance in equilibrium calculation. This means that the mixture has converged but ion balanced could not be obtained, generally due to a very low number for the maximum of iterations.
- SP13: One or more reactions had efficiencies for a third body not present in the mixture. This supposes no problem, said species will simply not be considered since they are not present in the mixture.
- SP14: Species compositions were reset during equilibrium calculations in order to reintroduce an atom in the mixture. This may be caused by the removal of a no longer considered condensed specie. If no other warnings/errors are present, the user should disregard this event since the problem was fixed.
- SP15: Computation of equilibrium conductivity failed. Frozen value given instead. This is probably due to the selected reaction (kinetics) scheme, as it is required to compute equilibrium conductivity (even when using ODE).
- SP16: Computation of equilibrium gaseous cp failed. Frozen value of Pr given. Gaseous cp is not a direct output, but it is used to compute Pr. This is probably due to the selected reaction (kinetics) scheme, as it is required to compute equilibrium conductivity (even when using ODE).
- ODE00: Maximum number of iterations for throat calculation reached. This is extremely rare and should only occur if the maximum number of iterations permitted is extremely low.
- ODE01: Maximum number of iterations for nozzle point pressure convergence reached. Generally this is due to a extremely low limit to the maximum iterations or problems with the equilibrium calculations.
- ODE02: Maximum number of iterations for frozen temperature calculation reached. Extremely rare, probably due to bad convergence if frozen point is before the throat or to a low limit to the maximum iterations.
- ODE03: Maximum number of iterations for frozen throat calculation reached. Can be a common warning since it is a difficult to achieve convergence. Nonetheless it is not very significant if it is not achieved as long as the number of iterations performed is sufficiently large. For a sufficiently large maximum number of iterations this warning will never occur, although it may impact computing time slightly.
- ODE04: Maximum number of iterations for FAC initial conditions reached. Not common, may

be solved increasing maximum number of iterations.

- ODK03: It occurs when the area relation criteria to switch from given area to given pressure calculations in ODK in the convergent nozzle is invalid. It is usually caused due to no point in the nozzle being in between the throat an the area relation given. The user should change the criteria or increase the number of points in the nozzle.
- ODK04: Equivalent to OPDK03 but for the divergent nozzle, when calculations swithc from given pressure to given area.
- ODK05: Reached maximum number of iterations for global convergence of ODK problem. Generally this limit should be set very high, since a lot of iterations are required to accomodate all the sub-iterations present in the ODK problem.
- ODK06: Reached maximum number of iterations for throat convergence of ODK problem. Generally this is caused by a very fine tolerance for the throat position, which is a very hard convergence to achieve. It is recommended to reduce this tolerance. Alternatively the iteration limit may be increased but this will probably not avoid the convergence problem.
- ODK07: Reached maximum number of iterations for near-throat points convergence in ODK. The most common cause for this warning is that the amount of points calculated with pressure is to big. It is recommended to reduce the area relations at which the switch happens.
- ODK08: Reached maximum number of iterations for mass-flow convergence in ODK. This error occurs because the tolerance set is to fine for the number of points in the nozzle. Said number of points greatly impacts how many significant figures of the mass flow are maintained constant throughout the nozzle. Therefore, if the tolerance is too fine, convergence may never be achieved.
- V00: Pressure in the combustion chamber is beneath the lower limit for Vielle's parameters definition. This will have an effect in the validity of the results obtained. It is recommended to try and re-define the intervals or to change the nozzle geometry.
- V01: Equivalent to V00 but the pressure exceeds the upper limit.
- V02: Maximum number of iterations for pressure calculations reached. Generally due to the iteration procedure being stuck near an interval change in Vielle's parameters definition. There is no way to solve this problem, but the solution offered should be very close to the correct solution.
- OPT00: Maximum number iterations for optimization reached. This can be caused either by an invalid target, too strict restrictions or a bad-behaving Vielle's intervals definition.
- OPT01: Early exit from optimization procedure. This error is caused by an unexpected behaviour during the optimization procedure, it may be caused by an unusual interval definition for Vielle's

parameters. A bug report should be filed.

- F00: Species file could not be found at the location given in preferences, probably it was deleted or moved to a different location.
- F02: Reactants file could not be found at the location given in preferences, probably it was deleted or moved to a different location.
- F04: Kinetics file could not be found at the location given in preferences, probably it was deleted or moved to a different location.
- F06: Propellants file could not be found at the location given in preferences, probably it was deleted or moved to a different location.
- F08: Nozzle file could not be found at the location given in preferences, probably it was deleted or moved to a different location.
- F10: Grain file could not be found at the location given in preferences, probably it was deleted or moved to a different location.
- F12: Configurations file could not be found at the location given in preferences, probably it was deleted or moved to a different location.
- F14: Preferences file could not be found at the location given in preferences, probably it was deleted or moved to a different location.
- F16: Species file at the location given in preferences was corrupted, probably due to manually modifying the file. If the user chose not to delete the affected items on startup, the file should be corrected manually or it may cause malfunctions in the software.
- F18: Reactants file at the location given in preferences was corrupted, probably due to manually modifying the file. If the user chose not to delete the affected items on startup, the file should be corrected manually or it may cause malfunctions in the software.
- F20: Kinetics file at the location given in preferences was corrupted, probably due to manually modifying the file. If the user chose not to delete the affected items on startup, the file should be corrected manually or it may cause malfunctions in the software.
- F22: Propellants file at the location given in preferences was corrupted, probably due to manually modifying the file. If the user chose not to delete the affected items on startup, the file should be corrected manually or it may cause malfunctions in the software.
- F24: Nozzle file at the location given in preferences was corrupted, probably due to manually modifying the file. If the user chose not to delete the affected items on startup, the file should be corrected manually or it may cause malfunctions in the software.

- F26: Grain file at the location given in preferences was corrupted, probably due to manually modifying the file. If the user chose not to delete the affected items on startup, the file should be corrected manually or it may cause malfunctions in the software.
- F28: Configuirations file at the location given in preferences was corrupted, due to manually modifying the file. If the user chose not to delete the affected items on startup, the file should be corrected manually or it may cause malfunctions in the software.
- F30: Preferences file at the location given in preferences was corrupted, affected items will be returned to their default values. This may happen due to manually modifying the file or using a file from a previous version (in this case old values are unaffected and new preferences items are added to the file). All affected items were reset to its default values.

9.2 Critical errors

This subsection contains a complete list of all possible critical errors given by the program, along with a short explanation of what may have caused them and possible solutions. Critical errors forbid the program from continuing its calculations and may even prevent the user from starting them. They represent the occurrence of a serious problem and should be solved immediately.

- ODK00: Failure during initialization of the differential equations solver (CVodes). This prevents the solver from functioning and the ODK calculation is aborted. There is no way to solve it, a bug report must be sent to the developer.
- ODK01: Failure during calculations of the differential equations solver (CVodes). ODK calculation is aborted. There is no way to solve it, a bug report must be sent to the developer.
- ODK02: Failure during closure of the differential equations solver (CVodes). The calculations are finished, but a memory leak might have happened. There is no way to solve it, a bug report must be sent to the developer.
- U: A unit conversion failed, unit not defined in program. This error only happens if the user has tampered with the save files and manually changed a value. If it is not the case, please file a bug report to the developer.
- L00: All considered species have the same element ratio as that of the reactants. A soft-rest is set in place. This problem may arise if to little species are being considered in the mixture.
- L01: At least one condensed specie's elements are a linear combination of the others. One of the condensed species is automatically removed, but this may cause the solution to not be valid or prevent convergence.
- L02: No gaseous species were included in the calculation. Therefore, the mixture is invalid and

equilibrium calculations can not be performed.

- L03: At least one positive and one negative ion must be considered (or none at all). Having only positive or only negative ions would make impossible charge equilibrium, which is required.
- L04: One of the species present in the propellant definition was not found in the database. This error happens if the user has deleted species that a propellant was considering.
- L05: One of the reactants present in the propellant definition was not found in the database. This error happens if the user has deleted a reactant that forms part of a propellant.
- L06: One of the reactions present in the propellant definition was not found in the database. This error happens if the user has deleted a reaction that was considered by a propellant.
- L07: Propellant load failed for an undetermined reason (probably a file could not be accessed or was corrupted).
- L08: Nozzle load failed for an undetermined reason (probably a file could not be accessed or was corrupted).
- L09: Grain load failed for an undetermined reason (probably a file could not be accessed or was corrupted).
- F01: Species file could not be found at its default location, probably it was deleted or moved to a different location.
- F03: Reactants file could not be found at its default location, probably it was deleted or moved to a different location.
- F05: Kinetics file could not be found at its default location, probably it was deleted or moved to a different location.
- F07: Propellants file could not be found at its default location, probably it was deleted or moved to a different location.
- F09: Nozzle file could not be found at its default location, probably it was deleted or moved to a different location.
- F11: Grain file could not be found at its default location, probably it was deleted or moved to a different location.
- F13: Configurations file could not be found at its default location, probably it was deleted or moved to a different location.
- F15: Preferences file could not be found at its default location, probably it was deleted or moved to a different location.

- F17: Species file at its default location was corrupted, probably due to manually modifying the file. If the user chose not to delete the affected items on startup, the file should be corrected manually or it may cause malfunctions in the software.
- F19: Reactants file at its default location was corrupted, probably due to manually modifying the file. If the user chose not to delete the affected items on startup, the file should be corrected manually or it may cause malfunctions in the software.
- F21: Kinetics file at its default location was corrupted, probably due to manually modifying the file. If the user chose not to delete the affected items on startup, the file should be corrected manually or it may cause malfunctions in the software.
- F23: Propellants file at its default location was corrupted, probably due to manually modifying the file. If the user chose not to delete the affected items on startup, the file should be corrected manually or it may cause malfunctions in the software.
- F25: Nozzle file at its default location was corrupted, probably due to manually modifying the file. If the user chose not to delete the affected items on startup, the file should be corrected manually or it may cause malfunctions in the software.
- F27: Grain file at its default location was corrupted, probably due to manually modifying the file. If the user chose not to delete the affected items on startup, the file should be corrected manually or it may cause malfunctions in the software.
- F29: Configuirations file at its default location was corrupted, due to manually modifying the file. If the user chose not to delete the affected items on startup, the file should be corrected manually or it may cause malfunctions in the software.
- F31: Preferences file at its default location was corrupted, affected items will be returned to their default values. This may happen due to manually modifying the file or using a file from a previous version (in this case old values are unaffected and new preferences items are added to the file). All affected items were reset to its default values.

10 File structure

10.1 Save Files

Save files contain the information of the different components employed by the application. All save files are '.json' files. The files are both written from and read by the application in order to use the data of said components. It is highly recommended to not open nor tamper with these files in any way. The default names for the files are:

- Reactants: Contains information about the reactants of the application. This information can be modified in the Propellant Tab, clicking on the 'Reactant Database' button.
- Species: Contains information about the species of the application. This information can be modified in the Propellant Tab, clicking on the 'Specie Database' button. Note that it also contains the information about the last loaded species (further explained in subsection 3.5).
- Kinetics: Contains information about the chemical reactions of the application. This information can be modified in the Propellant Tab, clicking on the 'Kinetics Database' button. Note that it also contains the information about the last loaded reactions (further explained in subsection 3.5).
- Propellants: Contains the information necessary to describe the various propellants created by the user.
- Grains: Contains the information necessary to describe the various grain geometries created by the user.
- Nozzles: Contains the information necessary to describe the various nozzles created by the user.
- Configurations: Contains information of the different configurations for engine simulation in the 'Calculations tab' saved by the user.
- Preferences: Contains all the information regarding the users preferences. This is the only file that could be needed to be manually modified, further explained in subsubsection 7.2.2.

By default, after installation, the program comes with all these files (and a set of basic components and information). Nonetheless, the option to use different compatible files is given, as is explained in subsubsection 7.1.1. This may be useful in order to exchange save files with a colleague, in order to test the components created by them. Note that using configurations saved with a different file set may result in errors.

10.2 Input files

10.2.1 Thermodynamic data files

Currently, the only supported format for thermodynamic data input files is the one used by NASA's CEA, although other format types are planned to be implemented in future releases. Figure 22 shows an example of such a file, where different parts are highlighted in order to show the format to be followed. Note that all this information can also be input through the application (as seen in subsubsection 3.2.1).

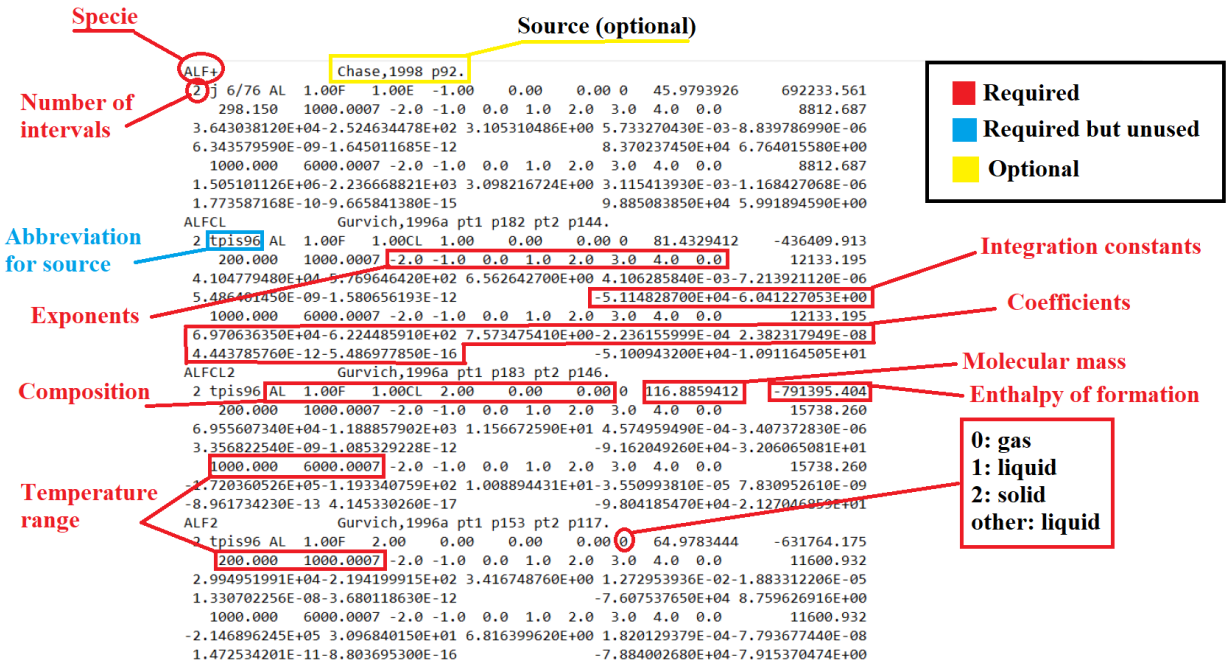


Figure 22: Format for thermodynamic data input files

10.2.2 Transport properties data files

Currently, the only supported format for transport properties data input files is the one used by NASA's CEA, although other format types are planned to be implemented in future releases. Figure 23 shows an example of such a file, where different parts are highlighted in order to show the format to be followed. Note that all this information can also be input through the application (as seen in subsubsection 3.2.1).

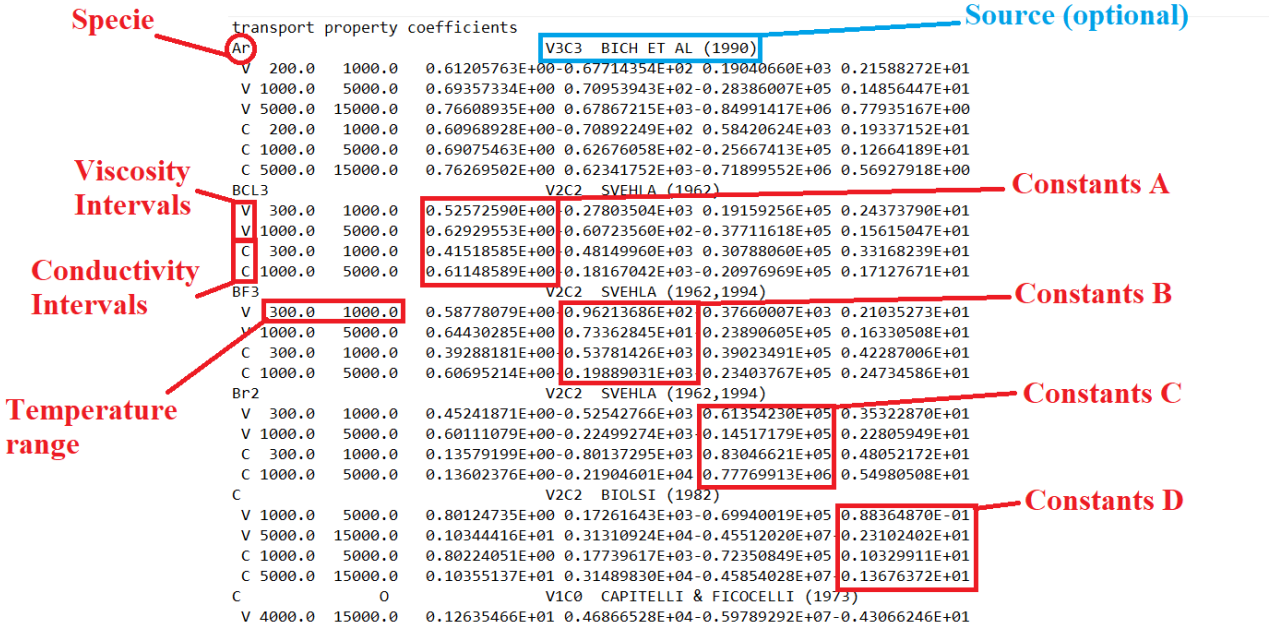


Figure 23: Format for transport properties data input files

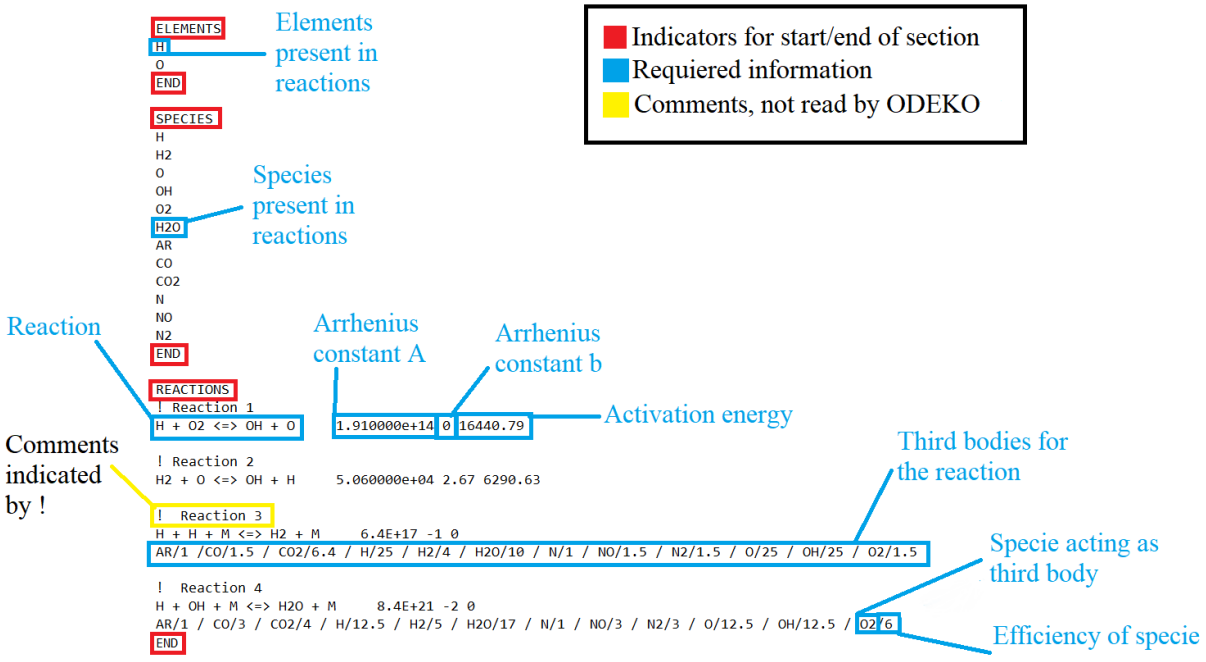


Figure 24: Format for chemical reactions data input files

10.2.3 Chemical reactions data files

Currently, the only supported format for chemical reactions data input files is the one used by Ansys CHEMKIN, although other format types are planned to be implemented in future releases. Figure 24 shows an example of such a file, where different parts are highlighted in order to show the format to be followed. Note that all this information can also be input through the application (as seen in subsubsection 3.2.3).

10.3 Output files

Output files are generated by the application when a data export is conducted by the user. It generates '.csv' files, which can be read using Microsoft's Excel. Note that in order for the files to be properly read, the delimiter character used by the ODEKO must match that used by the reading application (e.g. Excel). In the preferences menu, said character may be changed (see subsection 8.1). Note that this character can also be changed in Excel. The name for the csv file is chosen by the user.

The export structure is different for each tab and also differs based on the selected data from that tab to be exported. Specific information can be found in the section regarding each tab (3.6 4.3 5.4 6.5).

Figures may also be exported in several tabs. The image size can be modified in the preferences menu (subsection 8.1). By default, the image will have the same name as the csv file. In case of multiple figures, they will have the same base name, followed by a differentiating character string. Note that this may be changed in the preferences menu (subsection 8.1), which would force the user to choose a name for each generated file.

11 Bug reports & future releases

11.1 Where & how to file a bug report

Bug reports can be sent to odeko@nickdejongc.com and a dedicated bug report page will soon be made available at the ODEKO website.

It is appreciated that bugs are reported with the most amount of information possible. The most important pieces of information are the error caused by the program (let it be a crash, a freeze, a malfunction...) and the setup needed to replicate the error (parameters used in the calculations, preference settings...). It is always appreciated to include the preferences and database files used by the user, so it is easier to replicate the issue.

Bug reports will be ordered by relevance, prioritizing the most critical bugs first. A new release will not be launched for each solved bug. Whenever a sufficient amount of changes or bug fixes are implemented a new release will come out. If a major bug is resolved a new release with that patch will also be put out.

11.2 Where to find information about new releases

Information about new releases will be posted on the ODEKO website. Here the latest version of the software as well as of the documentation will be available. Only the latest version of the software and documentation will be available, but the user will also be able to find the patch notes for all releases, which contain information on bug fixes and added features for each release with respect to the prior one.

11.3 Plans for version 1.1

Release 1.1 is already on the works and various changes are planned for it, the following lists contains some of the changes that are meant to be implemented. More likely than not, a decent amount of bug fixes will also be included in v1.1. Any suggestions are more than welcome, and may be sent to odeko@nickdejongc.com.

Installer upgrades:

- Improve the visuals of the installer.
- Offer the option to choose which shortcuts the user wants in the installer.
- Allow to install updates of the application instead of having to uninstall and reinstall again. This will allow the user to update the software directly from their installation instead of having to download the new version.

Cosmetic or QOL additions:

- Include an in-application bug report option.
- Add keyboard shortcuts to the application.
- Addition of progress bars to calculations and optimizations.
- Capability to delete multiple saved items simultaneously.
- More customization options!

New features:

- Improve optimization procedure (speed and accuracy).
- Allow to define star-shaped grains.
- Allow to define custom grains.
- Offer to define custom nozzles.
- Add ODK FAC options.
- Add IAC FAC with grain geometry.
- Auto-correction of corrupt save files.
- Better load checks to avoid inconsistencies between loaded files.
- Improved support for external input of thermal and transport properties as well as kinetics.