# E7 FINAL PROJECT WRITE UP

**Jacob Haddad, Ishaan Srivastava, Daniel Paulo-Wach**

**<u>Algorithm Overview:</u>**

Our app's functionality is fundamentally determined by the user's inputs, in terms of both button selection and the files themselves. The app's startup screen includes a graph of historical data and projected "business as usual" carbon emissions vs. time. For the app to function properly, it is **necessary** to first specify a wedges file. If a strategy isn't specified, implementation year must be specified, and specifying this year automatically generates a flat line projection of emissions from that year onwards. Pressing the "Upload" button for a given file opens up a dialogue box, allowing the user to select a file from any given directory. The user input file must be a .txt file, otherwise the app (not MATLAB) will generate an error and request that the correct file type be uploaded.

Once the wedges file is selected, the data is stored using the index of each wedge in the file, and each wedge becomes accessible in the dropdown menu. Each wedge's identity and relevant information is determined by its index, and hence the indices are stored as properties, making them accessible throughout the app. We graphed the wedges using exponential fitting after solving for the coefficients using "\", using the knowledge that each wedge corresponds to a reduction in emissions of 1GtC/yr after 50 years of implementation. We shaded the wedges via the area and fill functions, and determined the colour based on the wedge's category, which was also associated with its index. The colour and name of the wedge is provided on the graph's legend for easy identification. In addition to strategies and wedges, a constraints file may be uploaded, but is not required. If the file is uploaded, the app prevents selection of wedges that violate the constraints in the file. Regardless of whether a constraints file is uploaded, the app prevents a single wedge accounting for more than half of the total wedges used.

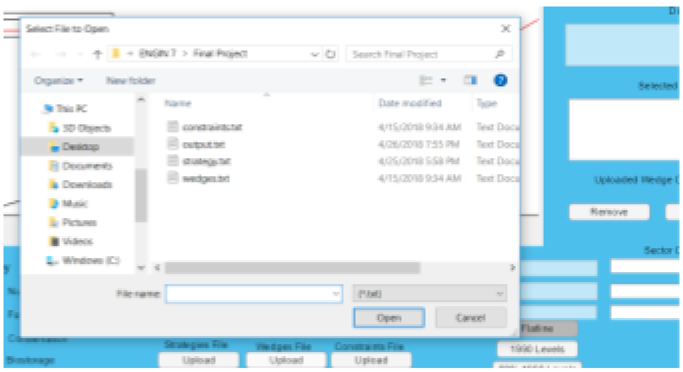
If the user attempts to input invalid files or data, or interact with the app's elements in an unexpected manner, the relevant error message is displayed in the dialogue box with suggested solutions. In the event of an error, the user has the option of using the "Reset" button to clear their wedges and implementation year and start over. As per the basic and intermediate objectives, the user is able to individually add and remove wedges as desired. The wedges all start from the same year, but this year is user determined. We have included a series of counters so that the user can track their current number of wedges, the total cost associated with these wedges, and how close they are to violating any possible constraints. We also have a counter that displays the number of wedges required to flatline. The intermediate objectives require that the user can toggle between flatlining, 1990 levels of $CO_2$ emission, and 80% of 1990 $CO_2$ emissions. We have successfully coded this functionality into our app, but the user must toggle to the desired category and select it before the counter displays the correct answer. Until then, the counter displays 0. As mentioned earlier, a pre-uploaded/pre-selected wedges file and a strategy file/implementation year are both necessary for this functionality.

We have also included the relevant $CO_2$ and global warming graphs, which are available in the secondary tab. These graphs are appropriately labeled and dynamically shift as per the addition or removal of wedges by the user. Additionally, the user can input their own value of the contested constant λ. In terms of the advanced features, we include a short summary of the mitigation strategy as well as its needs and challenges to employ it in a description box for each wedge. We also proposed a wedge, the details of which can be found on page 3, and we included a random wedge generator as our unique feature, because we wanted to be able give the user greater autonomy in how they generate wedges, beyond just manually selecting a wedge or uploading a strategy file. Clicking the random wedge generator uses randi to generate a pseudo-random index for a wedge

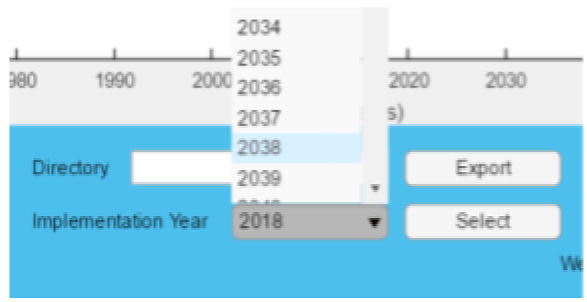
Finally, the output file follows the format specified in the project instructions, with the data being extracted using the wedges' respective indices in the original wedges file. By default, the export file is exported to the current working directory, but the user is able to input their own export destination in the provided directory text box.

**<u>Example Output and Key Features:</u>**

The program's GUI is incredibly user-friendly and aesthetically pleasing, with the program's unique features become readily available upon the user uploading a file that describes all possible wedge choices and specifying the implementation year. The user can easily complete both these preliminary tasks by selecting the 'Upload' button under the label which reads 'Wedges File', then selecting a year from the drop down item box and pressing the 'Select' button right beside it.



Users upload the appropriate files via clearly labeled buttons and a convenient dialogue box.




The user has the choice of defining their own starting year for the implementation of the wedges.

After this, a wide range of possible functionalities present themselves. For one, the mitigation strategies all become available for selection via a drop down box. Users are able to select whatever strategy they please as mitigation wedges are plotted on the primary graph of Emissions vs. Time. These wedges are filled in and color coded based on what major mitigation category they belong to - as described by a useful color key in the bottom left corner. We have also shaded, in light blue, the remaining amount of emissions that must be overcome after selecting a wedge to reach 80% of 1990s emission. While selecting wedges, a user can toggle a counter between 'Flatline', '1990 Levels', and '80% 1990 Levels' to display in a text box above how many wedges are required to achieve specific goals by the target year of 2068. These numbers adjust according to the selected implementation year. Additionally, information on each wedge is displayed upon selection, as well as a tally of the total cost.

The software engineers behind this project understand that in the midst of difficult international negotiations regarding a worldwide carbon mitigation initiative, it may be difficult for foreign representatives to come to a consensus on what the best thing that can be done for the planet is. That is why we included an advanced, custom-made feature called the 'random' button that randomly selects which wedge to be implemented as to break any stalemates that may arise in negotiation. It supports all features of the select button, and obeys any possible constraints specified by the user.

**Wedge Proposal:**

Our new proposed wedge is based on technology developed by Climeworks, which brands its products as "the world's first commercial carbon removal technology"[1]. The primary difference between this proposed wedge and the pre-existing wedges is that rather than prevent carbon or carbon dioxide emission in the first place, this wedge will actively remove carbon dioxide from the atmosphere. Since there is no shortage of carbon dioxide in the atmosphere, this shouldn't be a problem. To be as effective as the other wedges, the Climeworks technology must be able to remove 1 GtC/yr in 50 years.

Assuming a linear model, to reach this goal of 1 GtC/yr in 50 years, total amount of carbon that must be extracted in this time period $= \frac{1}{2} \times 50 \times 1 \; = \; 25 \; GtC$ (area of triangle)

Since the aforementioned technology extracts carbon dioxide from the atmosphere instead of pure carbon, we must convert accordingly.

1 GtC = 3.664 $GtCO_2$,[2] therefore 25 GtC = 91.6 $GtCO_2$

Thus, amount of $CO_2$ to be extracted per day over 50 years $= \frac{91.6\times10^{12} kg}{365\times50 \; days} = 5.02 \times 10^9 kg/day$ (approx.)

The most effective Climeworks unit can extract approximately 4920 kg of $CO_2$ per day[3]

Therefore total number of units required $= \frac{5.02\times10^9}{4920}$ = 1020000 (approx.)

Assuming a cost of $100,000 per unit, this brings total cost to 1020000*100000 = $102 billion. Initially, this seems too expensive, yet this goal is being achieved over 50 years

Therefore, annual expenditure = $2.04 billion. In comparison, the annual budget of USA's EPA (Environmental Protection Agency) was $8.2 billion in 2017[4]

Thus, expressed as a percentage, the cost of this proposed wedge is $\frac{2.04}{8.20} \times 100 \; = \; 24.88\%$ of the EPA's budget. However, the EPA is only listed as an example, and assuming the whole world is committed to reaching this goal of reducing CO2 emissions, then this wedge becomes both

[1] http://www.climeworks.com/about/

[2] http://www.globalcarbonproject.org/carbonbudget/archive.htm (conversion listed in global budget for 2016)

[3] http://www.climeworks.com/our-products/

[4] https://www.epa.gov/sites/production/files/2017-05/documents/fy-2018-budget-in-brief.pdf

technologically and financially viable. As such, we are happy to propose this Climeworks technology as another wedge for governments, policymakers, and even the public, to consider.