STEM Fellowship Big Data Challenge 2022

Using Genetic Algorithms to Optimize Energy Harvesting Relative to Budgets and Geolocation

Andrei Dumitrache[1] and Alex Rodriguez-Perez[1]

[1]Salmon Arm Secondary

February 26, 2022

Abstract

Replacing non-renewable energy sources with clean energy sources is one of the key steps in solving our world's greatest threat, climate change. Chapter 5 of the UN's Energy Progress Report[1] shows us that money is becoming significantly less of a problem as wealthier countries work together to support poverty-stricken countries in affording this transition to renewable energy. That being the case, a new, more challenging problem arises: how should a country make the most of a budget when purchasing energy harvesting technologies? This paper presents the solution to this problem through the application of genetic algorithms and reinforced learning to budgets and locations. In our research, we worked towards developing a method in the form of an algorithm that takes the name of a country along with a budget and returns the quantities of solar panels, wind turbines, and point absorbers that would generate the most kWh/day. With modified genetic algorithms, days of analyzing their performances along with results, and databases containing up to 70 years of data recorded hourly, our paper presents a method that can be used to bring about a new generation of data that could lead to possible improvements within government plans to include clean energy sources in their energy mix, should they be applied. An app has also been developed, giving anyone the ability to use our algorithm by simply inputting any country and budget, as climate change is a problem that affects us all, therefore our data should be available to everyone.

Keywords
Genetic Algorithm, Climate, Geolocation, Efficiency, Budget

1 Introduction

According to the United States' Office of Energy Efficiency and Renewable Energy[2], the power from a single ocean wave is enough to power an electric car for hundreds of miles. On top of that, 50% of the United States' population lives relatively close to the shoreline. Our question is, if the information stated previously is true, then why is the U.S. government not capitalising on this opportunity by investing in attenuators, point absorbers, and Pelamis-like systems?
Chapter 5 of the UN's Energy Progress Report[1] states that international financial flows (with the purpose of helping poorer countries afford renewable energy) have been increasing over the last 5 years. This helped with relieving the countries' financial strains and helped them focus on their economy while still investing in renewable energy sources, as shown in the following graphs with data from Morocco, where renewable energy sources increase and nonrenewable sources decrease while gdp continues to rise:

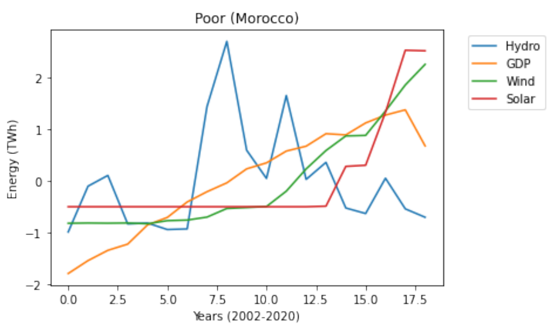


Figure 1: Morocco (Renewables)

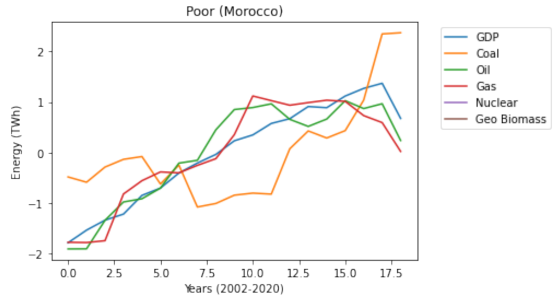


Figure 2: Morocco (Non-Renewables)

As financial restraints are gradually becoming less of a significant problem for poorer countries and more data is released showing that wealthier countries are not applying their budgets as strategically as they should, a new, more complicated question arises: in what (and how much) should a country invest in that would yield the highest profits? On the path to solving this issue, it is important to recognise early in the process that this problem is a combinatorial optimization problem, like The Knapsack Problem. Fortunately, problems like these have been studied for quite some time, with one of the most popular solutions for them being genetic algorithms. Theoretically, this problem could also be tackled by using a mathematical function, but like many others, this becomes a brute force method that fails to return a result in a reasonable amount of time as more variables are appended to our problem. Important variables that must be considered in order to receive optimal results are the climate characteristics of the area you are working on. For example, solar panels installed closer to Spain would produce more energy then those installed in Mongolia. Ideally, if this problem were to be solved, and the algorithm's results were to be applied to a country's energy infrastructure, major increases in the percentage of renewable energy sources that make up its energy mix would be observed. An interesting benefit of using a genetic algorithm to solve this problem is that if you can automate the process and add a user interface to it, the solution to this trivial-looking problem can be quickly created and accessible to anyone in the world.

2 Materials & Methods

2.1 Sources and Python Libraries

For our pre-analysis, we retrieved and used two datasets from 'Our World in Data'[3][4]. The main Python libraries our algorithm uses include Random, NumPy, Matplotlib, pandas, Selenium, OS, pathlib, shutil, and math. We also created our own module (Data_Getters) that is called upon by our genetic algorithm when data needs to be retrieved. Here is the repository: Repository. There are four functions in Data-Getters: getWeather(), solarOutput(), windOutput(), and waveOutput().

2.1.1 solarOutput()

The solarOuput() function takes in the name of a country and returns how much electricity the solar panel that we chose (Solaria PowerXT 400 R-PM 400W) would output per day (on average) based on the hours of sunlight in that area for a solar panel, along with the price for each panel. This is done by using the following formula:

$$c = 0.75wh$$

Where w is solar panel watts, h is average hours of sunlight, and e is daily watt-hours. The function uses daylight-time data from 186 different government sites.

2.1.2 getWeather()

The getWeather() function mainly takes in one argument (a location by name) and uses the OikoLab weather API[5] to retrieve meteorological data for that location.

2.1.3 windOutput()

The windOutput() function takes in a weather dataset (usually that returned by getWeather()) and calculates the air density for that area by using the temperature, air pressure, and dew point data from the dataset inputted into the function. It then uses the following formula to calculate the energy that would be theoretically produced by a Vestas (V90-2.0MW) wind turbine:

$$P = \frac{1}{2}pAv^2$$

Where P is equal to power, A is swept area of blades, p is air density, and v is wind speed. With this new number, windOutput() will perform a couple more simple operations and return the daily kWh average along with the price of one turbine. Information about our chosen wind turbine was retrieved from their(distributor) official website[6].

2.1.4 waveOutput()

The waveOutput() function returns the amount of energy that a point absorber would theoretically produce daily should it be installed in the selected country's EEZ (Exclusive Economic Zone) designated by the United Nations.

We split Earth's main bodies of water into 11 groups and placed "classification markers" (recorded as coordinates) in each group. We then handpicked the average wave height and wind speed of each group from the National Data Buoy Center[6] and the Global Wind Atlas[7].

The function works by first taking in a pair of coordinates (which could be included in the dataset returned by getWeather()) and applying the Haversine Formula to them, returning a list of distances from your location to each marker. Whatever group the closest marker belonged to would also be the body of water that the chosen country bordered.

We decided to follow the trends for the energy production rate of the point absorbers set by the University of Strathclyde in "Analysis of Cost Reduction Opportunities in the Wave Energy Industry"[8]. They used the following formulas(Figure 4) to bring forth the following graph(Figure 3):

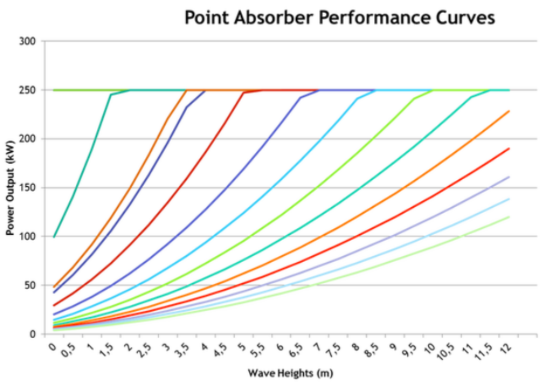


Figure 3: Point Absorber Trends with Each Line Representing a Different Power Curve

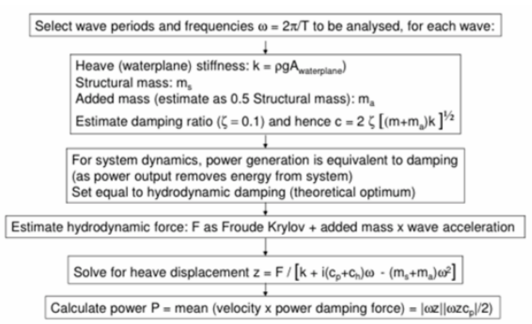


Figure 4: Strathclyde Calculations

By mimicking their trends, we created a simple mathematical function that can be used to calculate a rough estimate for how much energy a point absorber produces. Finally, waveOutput() will calculate how much energy would be harvested in a day for that location in kWh and return it along with the overall cost of the point absorber. This price was also taken from the University of Strathclyde's report[8] mentioned above.

2.2 Our Genetic Algorithm

We coded the algorithm ourselves, adding a reinforced machine learning technique to the algorithm conducive to optimising our results. The algorithm follows the standard steps that a genetic algorithm should follow; it creates the initial population (with randomised traits in the chromosomes), sends the chromosomes through a selection process to determine the "fitness level" of each individual, performs gene crossovers between the best individuals, mutates parts of the chromosomes at random (later changed when Reinforced Learning is applied), evaluates the newly formed chromosomes, penalises the "weak" and eliminates them, checks if the new generation has reached a better result, and restarts the entire process. The algorithm was modified to not stop after a certain number of generations, but when the results obtained cease to change and vary instead, implying that the algorithm has found the best possible mutated combination for the generation it started off with. Based on the research done in "Using Reinforcement Learning for Tuning Genetic Algorithms"[9], we followed a similar concept that Q-Learning uses, but catered it to our program. For example, a probability matrix is set to 50% for all parts of the chromosome, as seen below:

[50%, 50%], [50%, 50%], [50%, 50%]

Figure 5: Mutation Probabilities (Before)

The percentages are the probability rates for change in each section. In this improved version of the algorithm, our code checks to see if previous mutations were beneficial to the individual and modifies the probability matrix by increasing the rate in one section of the gene (the one that may become beneficial) and decreasing it in another. Here is a demonstration of the process applied to the matrix presented previously, along with the flow chart included in 'Using Reinforcement Learning for Tuning Genetic Algorithms'[9] that partly illustrates our application of Reinforced Learning to the algorithm:

[50%, 50%], [20%, 80%], [65%, 35%]

Figure 6: Mutation Probabilities (After)

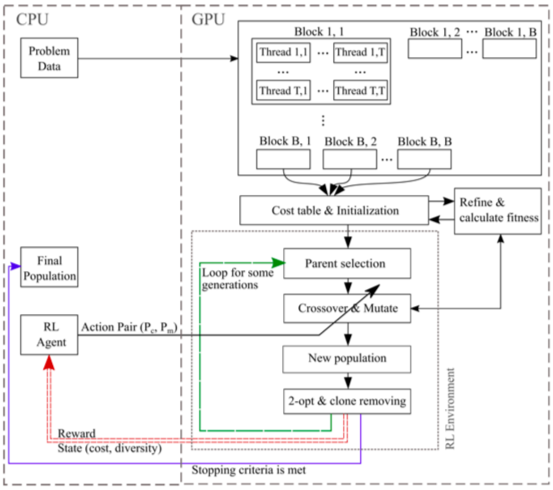


Figure 7: Algorithm Flow Chart

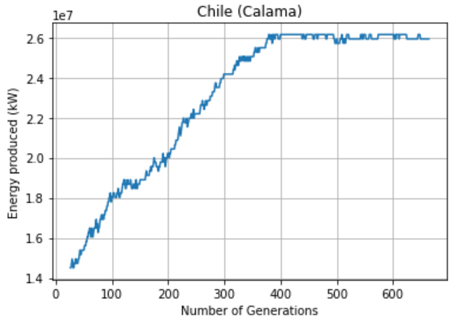


Figure 8: Current Genetic Algorithm Trend

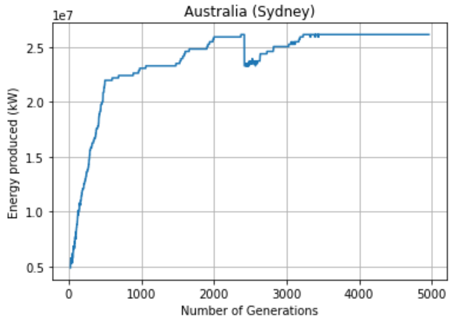


Figure 9: Current Genetic Algorithm Trend

Due to the fact that every time a genetic algorithm is run, a different result is produced, we have decided to make the program run the algorithm multiple times and pick the best result out of them all.

3 Results

3.1 Data Retrieved from Algorithm

We selected 11 countries from varying demographics and ran them through our algorithm with the budget set at $248,000,000 for all. The following table contains the results returned to us:

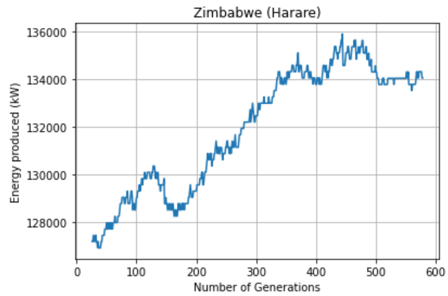


Figure 10: Current Genetic Algorithm Trend

Country	City	Possible Energy Output	Cost (CAD)	Wind Turbines	Solar Panels	Wave Point Absorbers
Canada	Vancouver	19345111.66653916 kW	$246,492,136.24	24	20	28
Australia	Sydney	26149337.480000004 kW	$247,638,272.87	0	12	119
Mexico	Villahermosa	8906354.710219435 kW	$247,776,323.13	30	42	81
Zimbabwe	Harare	142656.57523168437 kW	$243,713,700.00	83	48786	0
Saudi Arabia	Abha	9639503.352000002 kW	$243,486,733.41	0	36	115
Japan	Kanazawa	26149418.730000004 kW	$247,667,522.87	0	77	120
Russia	Yakutsk	35694036.72 kW	$245,583,878.14	0	71	117
Spain	Valencia	29422015.368 kW	$247,634,222.87	0	4	119
Romania	Bucharest	6371373.381600002 kW	$241,427,838.68	0	85	116
Chile	Calama	25929788.620000005 kW	$245,582,078.14	0	67	118
U.S.A.	Los Angeles	12962185.524000002 kW	$245,575,778.14	0	54	118

As our algorithm was generating the values displayed in the table above, we also recorded the results of each generation and plotted them. These graphs (Figures 8, 9, and 10) represent the trend that our algorithm follows when searching for the solution.

3.2 Recorded Data of the Algorithms' Performances

The data given to us by the current algorithm and previous versions of the algorithm was analysed as well, and proved to be useful. The performance of previous versions of the algorithm (without the implementation of reinforced learning) was recorded and compared to the most recent version that we used to obtain the results listed in the previous subsection. Here is Figure 11 displaying the results of an earlier algorithm where reinforced learning was not applied:

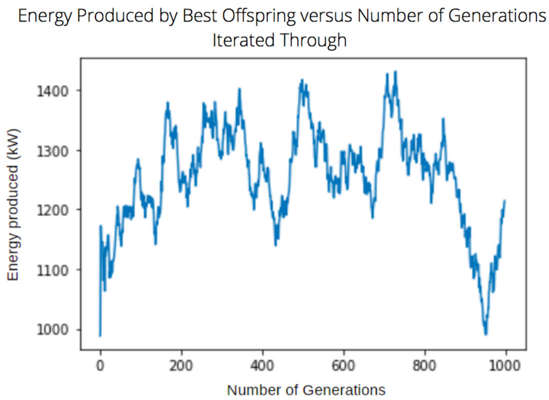


Figure 11: Previous Version of the Genetic Algorithm without Reinforced Learning Applied

We also recorded the performance of a previous version that was programmed to stop after a specific number of generations, unlike the current version that stops when the result ceases to vary.

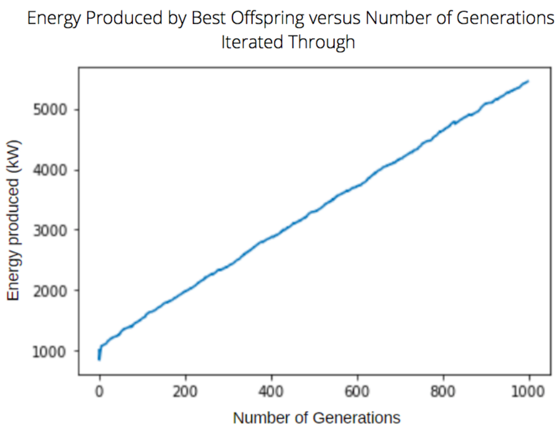


Figure 12: Previous Version of the Genetic Algorithm with a Fixed Number of Generations to Iterate Through and Reinforcement Learning Applied

Another important piece of data that was collected and graphed were the recommendations of the algorithm relative to the country (Figure 13).

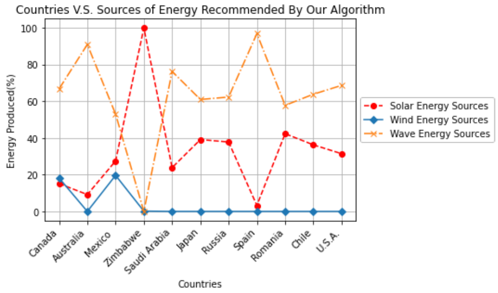


Figure 13: Energy Source Recommendations in Percentage for each Country

In addition, we created a flowchart(Figure 14) to illustrate the process our current algorithm follows in the Figures 8, 9, and 10 to display those results:

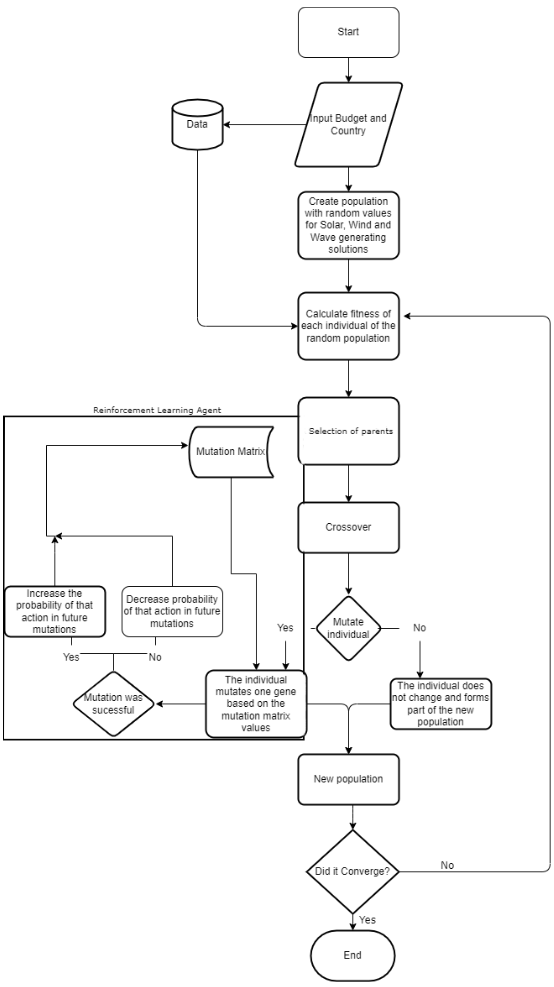


Figure 14: The Process our Graphs of the Algorithms' Performances Display

4 Discussion

4.1 Results and Connections

Comparing Figures 8, 9, and 10 to 11 and 12, the application of reinforced learning and iterative modification is proven to be crucial in obtaining accurate and satisfactory results. Our modification to the generation iterator ensures that the best values will be returned. A perfect example of this is the difference between Figure 9 and Figure 12. Figure 12 is the version of our algorithm that was set to stop creating generations after reaching generation #1000. This resulted in the algorithm stopping before it had the chance to reach the best selection of genes. Figure 9, on the other hand, is the current version of the algorithm we used, and it iterated through generations until the gene sequence ceased to mutate, reaching a total of roughly 5000 generations. Figure 9's results are also displayed in our results table, and if analyzed, prove that the algorithm is quite efficient in working towards finding a good gene sequence. Due to the lack of targeted mutation in the version of the algorithm displayed in Figure 11, the mean slope is significantly less steep than that of Figure 8's. This is because reinforced learning gives the algorithm a stronger "sense of direction" when it comes to working towards finding the best solution. It's also important to note that Figure 11 shows that the older version of the algorithm iterated through more generations than that of Figure 8's, and still did not receive a better result. This insinuates that the application of reinforced learning also shortens the processing time of the computer and improves the accuracy of the results.

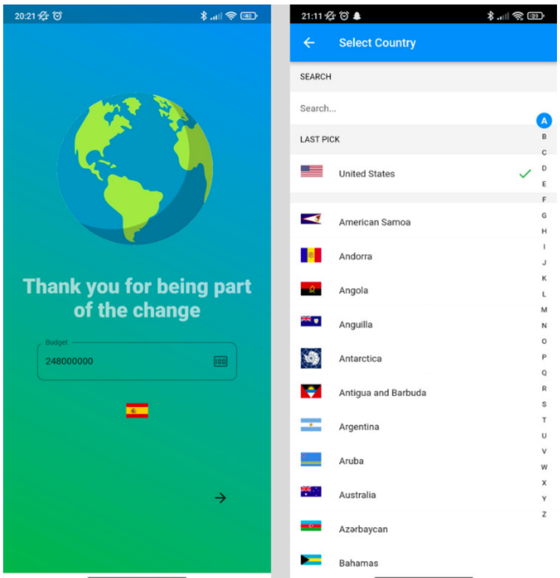
Our table gives us a significant amount of insight into how our algorithm works and how countries might be inadvertently investing in the wrong energy harvesting technologies. Considering that the budget was set at $248,000,000 CAD, the algorithm used as much as it could for the majority of its runs, using $247,776,323.10 (99.9%) of the budget on Villahermosa, Mexico, and $241,427,838.6 (97.3%) of the budget on Bucharest, Romania. It is important to note that the amount spent in Bucharest was the lowest relative to the other locations, meaning that our algorithm only leaves a small percentage of the budget out, even in the worst cases. The algorithm tends to capitalise on investing in point absorbers, as seen in our table and in Figure 13. This suggests that point absorbers are much more profitable relative to other energy harvesting methods like solar panels or wind turbines. Wind turbines are on the other end of the spectrum, with few countries having wind turbines recommended to them. Even in locations that tend to have higher wind speeds, the wind turbines still prove to be inefficient. Regardless of whether the country is landlocked or not, solar panels are preferred as substitutes for wind turbines when point absorbers are not an option, as displayed in our table for Harare, Zimbabwe. This led us to deduce that out of the three energy harvesting systems we chose, point absorbers are the most efficient, wind turbines are the least efficient, and solar panels are in between the two. This method could be extrapolated to other energy harvesting systems so as to analyse their efficiency and profit returns relative to each other. Other interesting results are the values in the Possible Energy Output column of our table. Theoretically, they should have been relatively close to each other, with minor deviations(for those cities). Surprisingly, all the values varied quite a bit. For example, Calama (Chile) was estimated to have 25,929,788.62 kW produced, and Bucharest (Romania) was estimated to only produce 6,371,373.38 kW (19,558,415.24 kW less). By more closely analysing Zimbabwe's possible energy output, we discovered that money is not the problem when attempting to maximise energy production. Instead, geolocations are the variables that have to be closer examined. Zimbabwe's energy output is the smallest of all 11 countries, even though it was given the exact same budget and resources as all the other countries. The only starting point that separates Zimbabwe from the other countries is its location. Being a landlocked country, Zimbabwe cannot take advantage of point absorbers, the most profitable energy harvesting technology out of the three. Regardless of the inclusion and application of hydro dams, geothermal plants, and other technologies, the majority of landlocked countries still have the disadvantage of not having access to large bodies of water where point absorbers could be implemented. Therefore, the problem is not the money, but the access to coastlines and technologies that limit the production of clean energy. Figure 13 displays an interesting correlation between solar panel recommendations and point absorber recommendations. Starting from Mexico on the x-axis and continuing until the United States, the linear models of solar energy sources and wave energy sources are reflectively symmetrical to each other. This occurs because the algorithm recognises point absorbers as the most efficient source of energy, with solar panels as the "runner-up" solution. This correlation is displayed by Figure 13 at Zimbabwe, where wave energy cannot be used. Therefore, the wave energy sources line drops to zero and the solar energy sources line climbs to what was supposed to be the wave energy sources line's y-coordinate.

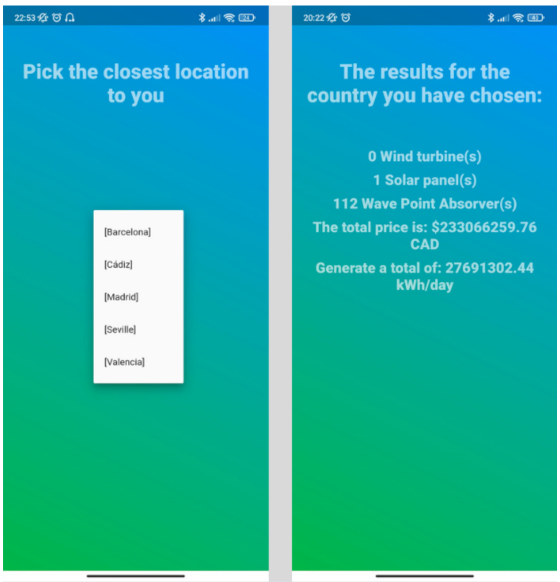
4.2 Recognising Sources of Error and Possible Solutions

Any sources of error in our study could only come from weather data inputted into the algorithm by functions in our Data_Getter module. The getWeather() function (one of the most important functions in the program) uses a weather API to retrieve data for us. This API is trusted and used by the Government of Canada. Therefore, we decided to exclude this function from our list of possible sources of error. The windOutput() function is also excluded as a possible source of error because it calls upon the getWeather() function to provide it with data to perform its calculations. The waveOutput() function could introduce an error into our results due to the fact that there are no public datasets that record the wave height for specific sections in various bodies of water. This led us to collecting averages by hand from weather tracking buoys stationed throughout the ocean. The identification of what bodies of water border the selected country in the waveOutput() function can also lead our calculations into error. This occurs as the function only takes into consideration the statistics from the closest ocean relative to the coordinates of that city or country, ignoring other possible coasts, thus not taking advantage of other bodies of water (for countries with multiple coasts), which could return more energy should a point absorber be installed there. This can be corrected by modifying the section in our code that handles the execution of the Haversine Formula. What's important to note is that all of these errors can be solved by providing the algorithm with more accurate datasets and modifying code. If we were given the opportunity to access other records that are not available to the public, we would be able to drastically increase the performance (in terms of accuracy) of our genetic algorithm.

Conclusions

In conclusion, it's important to recognise that our algorithm does not only return the solution to maximising energy production by applying the budget in the smartest way, but also provides a new generation of data such as efficiency of renewable energy harvesting technologies by location, possible improvements in government plans to include particular energy sources into their energy mix, and the time as well as cost it would take to switch the entire world to clean energy. We have made many interesting observations, such as money not being the problem in maximizing energy return, but the location, and that wind turbines are significantly less efficient then point absorbers or solar panels. In addition, we have worked hard towards bringing to light the immense potential genetic algorithms have in researching solutions for climate change through energy infrastructure optimization. We have also created an app that adds a user interface to our algorithm, allowing the user easy access to our algorithm by simply inputting a country and a budget.





We recognise that climate change is a problem that can only be solved if everyone does their part, so we thought it was only fair to make our algorithm available to anyone from their phones. In the future, we would like to modify the app so it may return results for someone's house, granting people the opportunity of utilizing our app to switch their household's energy source to renewable energy sources.

Acknowledgements

We would like to thank everyone at the STEM Fellowship who helped set up this event, as well as our teacher, Mrs. Thomson, our mentor, Mr. Michael Thomson, and, of course, our parents.

References

[1] United Nations. Energy - united nations sustainable development, 2021.

[2] Steven Bushong. How are ocean waves converted to electricity?, Apr 2017.

[3] Bernard Looney and Spencer Dale. Statistical review of world energy: Energy economics: Home, 2021.

[4] Bolt Lutien van Zanden, Jutta Lutien van Zanden, and Jan Luiten van Zanden. Maddison project database 2020, Dec 2021.

[5] Joseph Yang. Weather data downloader, 2020.

[6] V90-2.0 mw™, May 2002.

[7] Jake Badger, Neil Davis, Ray Drummond, Andrea Hahmann, Duncan Heathfield, Niels Gylling Mortensen, and Marko Onninen. Global wind atlas, 2015.

[8] Amaia Diaz Santamaria, Christopher John Casser, and Eva Topham Gonzalez. Analysis of cost reduction opportunities in the wave energy industry.

[9] Jose Quevedo, Marwan Abdelatti, Farhad Imani, and Manbir Sodhi. Using reinforcement learning for tuning genetic algorithms. 2021.