VULCAN - MAWS Software Improvement Team

Antoine Baize, Hannah Collins, Mariafernanda Hernandez, Nicholas Simone

Project Description

Aptamers are short, single-stranded DNA or RNA oligonucleotides that can bind with high affinity and specificity to proteins and other molecules of interest. The current process for discovering or designing aptamers, known as systematic evolution of ligands by exponential enrichment (SELEX), has led to many advances in the synthetic biology field, especially in terms of therapeutics, but is not an ideal process. In 2004, the first aptamer for therapeutic applications was approved by the FDA and used to treat macular degeneration, and many additional aptamers are going through clinical evaluations to be used as hematology, oncology, ocular, and inflammation indicators. However, the inefficiency and shortcomings of SELEX limit the creation of new aptamers and its applications to therapeutics. SELEX is costly, time consuming, requires laboratory work, and does not sample all possible nucleotide sequences.The Heidelberg Team from the 2015 iGEM competition tried to address this issue by creating MAWS (Making Aptamers Without SELEX). MAWS is not easily accessible due to the complexity of the installation process, the problems with running the program, and poor coding practice. Our team aimed to create a new software tool using MAWS as a model. It was our intention to fix the shortcomings MAWS had and create an improved, functioning tool. We felt that a viable in-silico aptamer selection platform could be a potent tool for including aptamer parts in genetic circuit designs.

Our project attempted to replicate MAWS' functionalities and make it more accessible to synthetic biologists. However, we were unsuccessful in our attempts to do

so. We managed to extract the information MAWS uses for their calculations via an alternative and more general Amber calling approach. This approach was using the pytraj module. Yet even here we ran into troubles when it came to extracting vital energy information from the molecule. It was unfortunate that we couldn't successfully run our program, but we described what our code aimed to accomplish in a separate "Failure Analysis" document.

Software Components

The main components of our project were the implementation of the pytraj and numpy modules.

Our Python script contained the pytraj library - this is an Amber library that allows users to pull molecular information from protein data bank (.pdb) files. We use this to load in pdb files and can perform operations to extract relevant molecular information to run the software. We implemented the numpy module to handle the nested array format used throughout MAWS. Also when we used a pytraj function to load in .pdb files, the format of the file was able to be manipulated using numpy functions, as it matches the format of MAWS' load commands and is able to be manipulated in the same manner.