

SLIRD Modeling On COVID-19 Data

Robert Jacobs

Department of Electrical and Systems Engineering
Washington University in St. Louis
St. Louis, MO, United States
robert.jacobs@wustl.edu

Max Cusick

Department of Electrical and Systems Engineering
Washington University in St. Louis
St. Louis, MO, United States
c.marguerite@wustl.edu

Ben Ko

Department of Electrical and Systems Engineering
Washington University in St. Louis
St. Louis, MO, United States
ben.k@wustl.edu

Abstract—This case study is on modeling COVID-19 and comparing said models to real world data. We work with the Susceptible, Infected, Recovered, Dead, or SIRD model, and expand it into the SLIRD model when we incorporate lockdowns into our calculations. Then based off of that model we propose a solution for decreasing the spread of COVID-19 which is to increase the use of lockdowns.

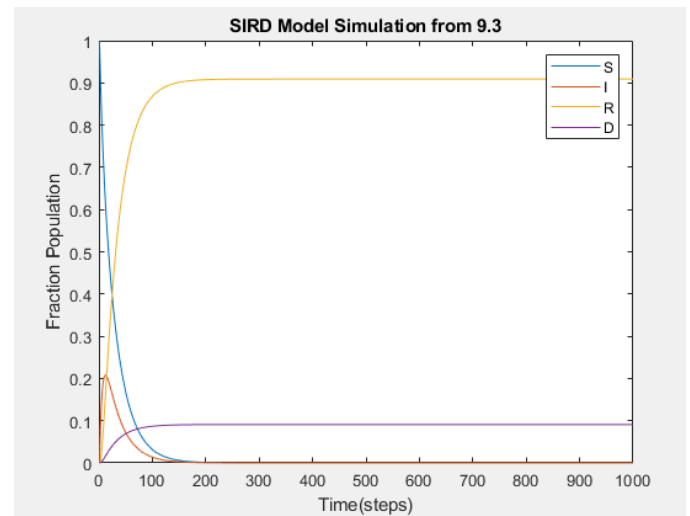
I. INTRODUCTION

This case study focuses around the use of models for tracking the effects of COVID on specific Missouri counties. We use and implement both the SIRD model and the SLIRD model in multiple ways to define a model to represent the results of the COVID-19 pandemic. We first implemented a basic SIRD model based off of the textbook, then expanded that SIRD model into fitting with the St. Louis COVID data. Then we expand our SIRD model into the SLIRD model and fit it with the St. Louis data as well as data from Jefferson City and Springfield, MO. Next, we looked at the models we were able to generate and then worked on ways to implement policies to limit infections and deaths while not being completely totalitarian.

II. METHODS

A. Part I, the SIRD Model from 9.3 of the Textbook

Part i of the case study focuses on first implementing a basic SIRD model in MATLAB as given by the textbook. This model takes in a matrix A and vector x_t the initial SIRD values to iterate over a given number of time steps. For each step the matrix A is multiplied with the column vector x_{t+1} which represents the SIRD values at the end of the time step. This is done with simple matrix multiplication in MATLAB and then plotted to produce these results:



The results were then compared against a provided SIRD model function to confirm accuracy.

B. Part ii, the SIRD Model on COVID Data

Part ii of the case study introduces the COVID data to the SIRD model. Using `base_sir_fit.m` and its associated functions in

siroutput.m and siroutput_full.m we compared the real number of infections and deaths to those predicted by our SIRD model. Our siroutput.m and siroutput_full.m files were made to take in data from base_sir_fit and run it through a simulation then calculate the cost of the simulated results based off of the real data.

The siroutput files take in seven values: three rates and SIRD. They get the transmission constraints made up of $k_{\text{infections}}$, k_{fatality} , and k_{recover} as well as the initial conditions ic_{susc} , ic_{inf} , ic_{rec} , and ic_{fatality} . These conditions result from using the fmincon function in base_sir_fit, and then put into matrices in the siroutput files so they can be used in our calculations. We then used these values to set up our A and x0 matrices in a similar way to the textbook model. Our A matrix used the transmission constraints to outline how SIRD would be applied in our model. We set it up row by row using 1's where we wanted values to carry over and 0's where we wanted no transmission effects. Our S row set the initial susceptible to be $1 - k_{\text{infections}}$ since we wanted the initial susceptible to be the entirety of the population excluding the ones who started infected. Our I row set the initial infected to be $k_{\text{infections}}$ then set the amount who stayed infected to be $1 - k_{\text{fatality}} - k_{\text{recovered}}$ since the infected could only stay infected, recover, or die. Our R row set k_{recover} as the amount who would recover and then kept the ones who had already recovered as recovered. Our D row was set up similarly to our R row where we kept track of how many died and how many had already died. Our x0 vector of initial conditions took the initial conditions we had already set up and put them in a form our model could interpret.

After setting up the transmission and initial matrices we ran them through the simulation with ss and lsim, then returned both the results of the simulation and the cost of the simulation. The cost was found in siroutput.m by taking the cumulative sum of the results of our model and comparing it to the real world data with the norm function. This returned the difference between the two sets of data as our cost. All of this information was then taken and put into base_sir_fit where our simulated data

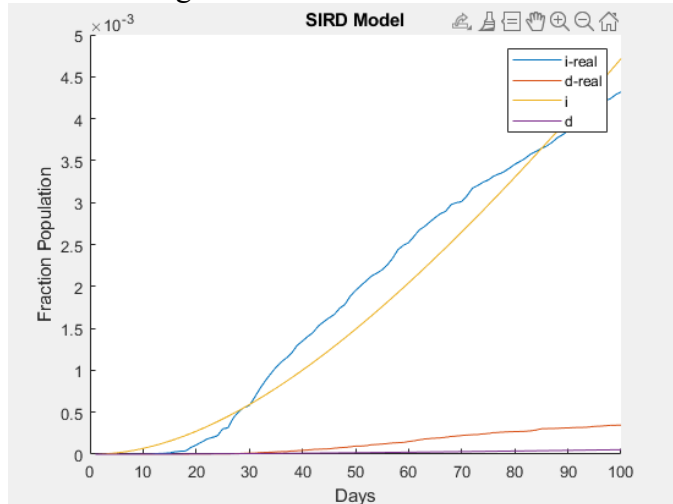
was plotted against the real world data to further check the accuracy of our fit.

We used base_sir_fit to set up our constraints for our data and how our transmission and initial values should be made. We made six initial parameter inequality constraints starting with constraining the death rate, x_2 , to less than 0.16 or 16% this was to make sure an appropriate amount of people in our model died. Then we constrained the recovery rate, x_3 , to being no larger than 0.92 or 92% to make sure not everyone recovered and some of the model population would still die. Next we constrained the infection rate, x_1 , to being no less than 0.7 to cause an appropriate amount of the population to get infected. We felt that at no point should more than 70% of the population be infected. This intuition is based on what we have seen so far with covid. Next, we constrained the total recovery x_6 , to less than 0.85 since not everyone would recover from the virus. Then we restrained the total deaths x_7 , to be less than 0.15 since most people who get COVID recover. Lastly, we make a general constraint for the susceptible population x_4 , to constrain x_4 to a max of 1 since the population will never go over 100%.

Furthermore, we set up a one-parameter constraint for the form $Ax = b$. This constraint was simply forcing S, I, R, and D to have a total sum of 1. This was a basic requirement for the SIRD model.

Next, we set up our upper and lower bounds constraints. We made these constraints by broadly figuring out what each SIRD value shouldn't go above or below while still leaving a large amount of room for the simulation to optimize. For our upper bounds, we set the first four values to one since the rates for infection, recovery, and death should never go above one as well as the susceptible population. We also set the maximum for our infection and recovery as .7 since not all of the population will get infected and not everyone who gets infected will recover. We also set the maximum for our death as .3 since COVID is not a very lethal disease although deaths are occasional. For our lower bounds, we set the lowest infection to .015 since at minimum we know at least 1.5% of the population should become infected. We set the recovered to .01 following the same logic that at the bare minimum,

1% of people who are infected with Covid should recover. We set the lower bound for death rate a bit higher at .3 since, through testing, it helped to fit our model closer to the real world values. We then set our minimum susceptible to .1 since not all of the population would get COVID. Then we set our recovery rate, death rate, and infection rate all to .01 so that at least 1% of the population regarding each would never go to or below zero in the model.



c. Part III, Modeling the Data for Each Region of Missouri

For part iii we chose to model the data for each of the three regions in Missouri separately. We decided that a few individuals transmitting the disease between regions would most likely not stay long and most likely not have a significant impact on the region's population. To account for interregional transmission, we instead chose to increase the infection rate slightly. To account for vaccinations, we made a constant vaccination rate of .3 in our updated sliroutput files. This rate seemed reasonable since it took a fair bit of time to get most of the population vaccinated, and for the most part, once vaccines were available, people became vaccinated. Our models for each of the three regions were constructed similarly. All three regions utilized the sliroutput.m and sliroutput_full.m files which were files from part 2 modified to accept SLIRD. We then built a base_slir_fit file for each region to facilitate the creation of three different location base models.

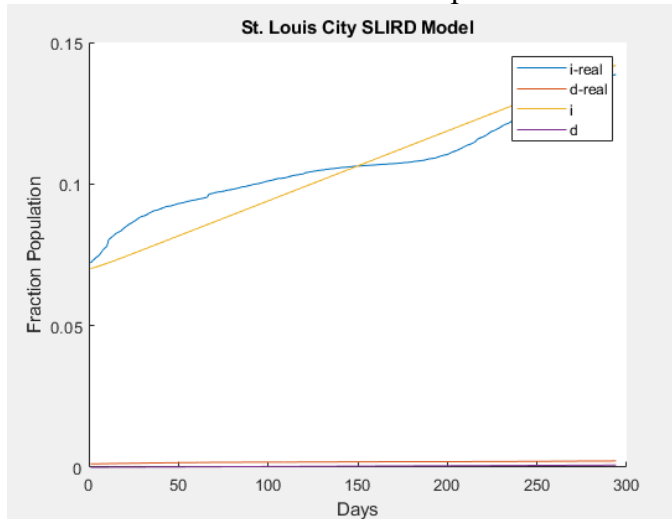
The sliroutput.m and sliroutput_full.m files were based on the siroutput.m and siroutput_full.m

files from part ii. We added lockdown as a rate and initial condition, changing the A matrix and initial conditions vector. We expanded the A matrix into a 5x5 matrix to represent the SLIRD model. Our first row changed to have those in lockdown and the vaccinations subtracted from the rest of the susceptible population as well as setting .4 in lockdown. Our second row set up how many were in lockdown and how many could move from lockdown to infected. Our third row was similar to our infections from part ii with minor changes to account for the vaccinations. Our fourth row kept track of both those getting vaccinated and those who naturally recovered and categorized both as recovered. Our final row was the same as in part ii and recorded the fatalities. We also set up our initial conditions similar to in part ii with the modification of adding lockdown.

To model each individual region, we made the files base_slir_fit_stlcity.m, base_slir_fit_jeffcity.m, and base_slir_fit_springfield.m. Each one called upon the sliroutput.m and sliroutput_full.m files the same way base_slir_fit.m called upon siroutput.m and siroutput_full.m. The files all ran the simulation similarly and plotted the cumulative data of each model against its region's real-world COVID data.

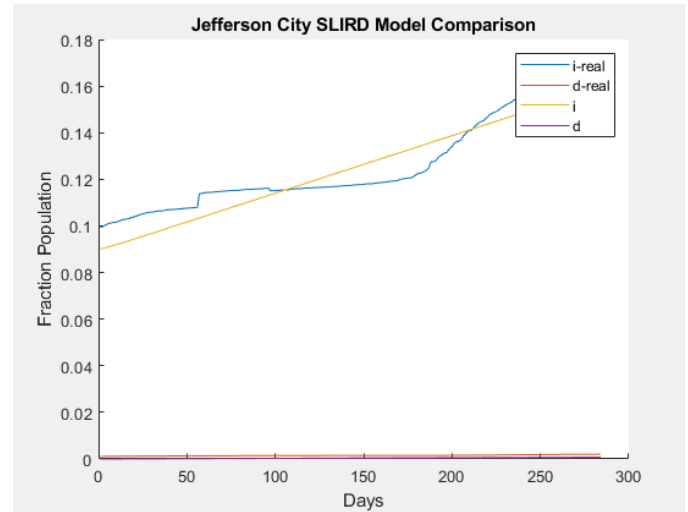
For base_slir_fit_stlcity.m we updated all of our constraints to account for the addition of lockdown to the model. Our rate and initial condition constraints were changed to make it so that all the rates were less than one and recovery and fatality didn't go above .5 when combined. These constraints were determined through testing our model multiple times and finding that they provided the best fit. Our fixed constraints were left the same as in part ii to make sure the combined rates equaled 1 so that the entirety of the population would always be affected. Our upper and lower bounds were also left mostly the same with minor changes to account for the extra people in lockdown and to make sure at least .3 of the population was in lockdown. Our initial conditions were set based on experimental testing and what we thought was reasonable. We set x_0_1 , the initial infection rate, to .3 since we know the disease will still spread moderately well even with lockdowns. We set the initial death rate, x_0_2 , to .03 since we reasoned the

combination of lockdowns and vaccinations would result in reduced deaths. We set the initial recovery rate, x_{03} , to .4 since time has shown most people with the disease eventually recover, but recovery wouldn't usually occur in one day. We set our initial lockdown rate, x_{04} , to .27 since most people would obey the lockdown, but it would not happen immediately. We set our initial susceptible and lockdown values, x_{05} and x_{06} , to .5 since we thought the population would be mostly split between them, then set the initial infection, recovery, and death values to 0 so that the model would balance them out for us. Running this model we got a moderate fit, with low deaths and linearly growing infections. Oddly, our model was outputting values that were a tad too large, so we had to implement some artificial fitting into our final graphs. Additionally, since these models only account for 2021 data we placed the starting value of the models at values near the expected levels.



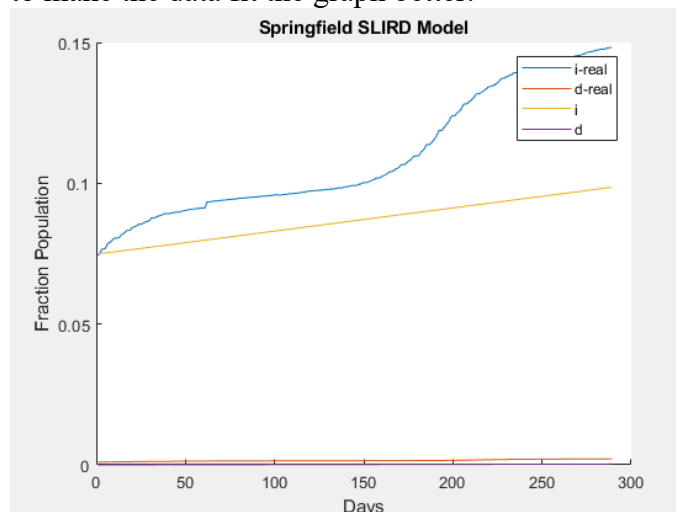
For `base_slir_fit_jeffcity.m`, we kept most of what we did similar to the St. Louis model. Our constraints mostly stayed the same with minor adjustments and our model produced similar fitting results. Our initial conditions were set based on our previous testing and what we thought was reasonable for the Jefferson City area. We set x_{01} the initial infection rate to .4 since we thought there would be slightly fewer people in Jefferson City getting the vaccine and going into lockdown than in St. Louis. We left the initial recovery and death rates the same since the disease would have the same recovery and fatality rate no matter the region (assuming similar medical care). We set our initial

lockdown rate, x_{04} , to .2 since we reasoned fewer people would be going into lockdown. We set our initial susceptible value, x_{05} , to .6 since we thought more people would initially be susceptible than in lockdown, x_{06} , which we set to .4. Then we again set the initial infection, recovery, and death values to 0 so that the model would balance them out for us. Running this model, we got a moderate fit which looked similar to the fit we got for our St. Louis city model. Like the Stl model, we had to do a little error correction on the scale of the model.

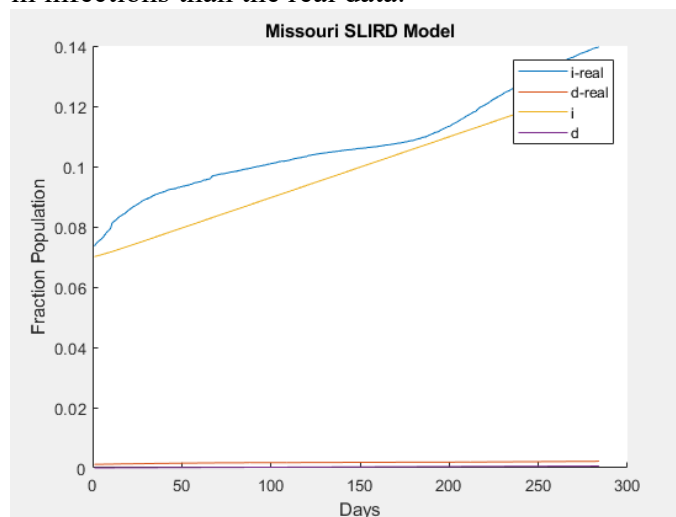


For `base_slir_fit_springfield.m`, we kept most of what we did like St. Louis and Jefferson City models. We decided that the constraints we used for our Jefferson City model would also be reasonable for Springfield since we expected those regions to have similar vaccination rates and compliance towards lockdown. This led to us having a model with fit slightly below the actual data. We most likely overestimated both the number of people who would be willing to get the vaccine and the number of people who would be willing to stay in lockdown. This would make sense since Springfield was one of the areas hit hardest in the last wave of COVID to go through Missouri. Just like the previous two models, we fixed the scaling

to make the data fit the graph better.



We then took our smaller region models and factored in inter region transmission. When making our new A matrix for inter-region transmission we took our values from our three individual regions and combined them into a large 15x15 transmission matrix. This led us to getting a model slightly lower in infections than the real data.



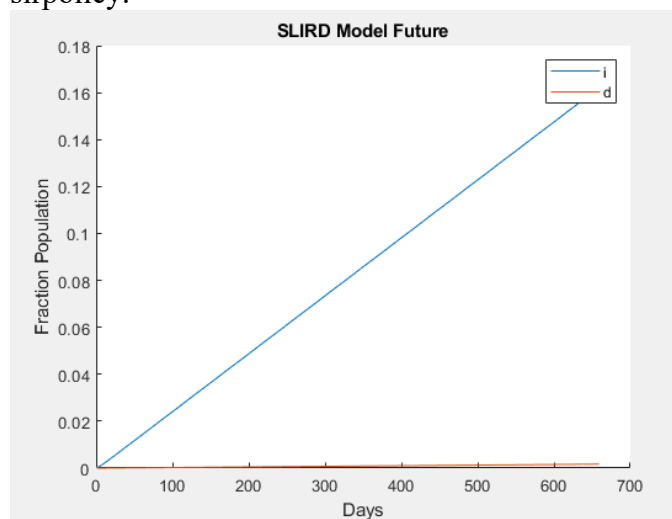
D. Part IV, Model Interpretation

Our models show some inconsistencies with real-world data. We assume this is because they do not account for real-world events such as the delta variant, various lockdowns beginning and ending, and the exact start date of vaccinations. For example, actual data showed an increase in covid cases around August of 2020, where we suspect the beginning of the school semester would have resulted in the small bump in number of covid cases. Our model did not entirely match this increase. Our model seems valid

for some of the time periods we compare with and invalid for longer term scaling. Our SIRD model for the first 110 days of the pandemic in St. Louis seems to fit the data for infections moderately well, but doesn't fit the data for deaths well. This trend is also shown in our SLIRD models for St. Louis, Jefferson City, and Springfield. Our SLIRD models are also all linear which doesn't always represent the waves of COVID well. Overall our models make some realistic predictions about infections on a limited time scale, but become less reasonable when predicting deaths and on larger time scales.

E. Part V, Policy Implementation

For part V we looked at all our models so far with specific focus on St. Louis. We used the rates developed in our `base_slir_fit_stl_future.m` file which predicts an additional year of data after the existing covid data as the baseline for the sirpolicy.



This model showed infections increase steadily so we decided that the best way to go about making policy changes was to increase the amount of people going into lockdown. We recognized that vaccines were limited in supply, and it would be impossible to vaccinate the entire city quickly. Instead, we decided we would investigate increasing the rate of lockdown so fewer people overall became infected. Ethically this seemed to be a good idea since although lockdowns were poorly received, they had already been used with success. Ideally, lockdowns wouldn't last for long, but the length of lockdowns is dependent on factors such as vaccination rate, economic strength, and other real-world factors we could not model.

To enact our policy, we used a new matrix in sirpolicy with slightly updated multipliers. In testing we found that to maximize the J_benefit, increasing lockdowns weren't as fruitful and an increase in vaccines actually benefited the J_benefit more than increasing the lockdowns. The resulting matrix multiplier is as follows:

```
[ 1 0 0 0 0;
  0 .95 0 0 0;
  0 .05 1 0 0;
  0 0 0 1 0;
  0 0 0 0 1];
```

This makes sense though as during lockdowns, people can still become infected, and we know from the past year that there are people who don't abide by the lockdown mandate and spread the virus. A vaccine prevents any gray area in the effectiveness of viral spread prevention (assuming the vaccine works as intended).

III. RESULTS AND DISCUSSION

We found that our models could be somewhat accurate when constrained correctly and put on the right timescale. Our SIRD model seemed to marginally fit the data better than our SLIRD models, though our SLIRD models also had some success. However, our models cannot account for real world events and deviations. We have found that the largest flaw in our models is that they rely on consistency in

the rates that the virus spreads. This consistency is not found in the real world where COVID came in waves and many factors affected the spread of the disease. We are also unable to properly account for the true scale of inter-regional transmission. We think that our policy decision of increasing lockdowns is easy to implement and not absurdly unethical. It was also shown by our models to have a moderate effect on overall infections.

IV. CONCLUSION

In conclusion, we found that our models were somewhat accurate. We can fit infections reasonably well but come short in accurately representing deaths. We also fall short in the long-term accuracy of our model's predictions and keep our model consistent with the various waves of COVID. In the future, we would like to explore making models with more real-world data to base our models on, as well as developing models which are better equipped to handle the nuances of real-world data. Our policy recommendation of more lockdowns found moderate success. It could most likely be made better by also using vaccination or by making lockdowns stricter. Overall, our models are imperfect, but still somewhat reasonable. In the end, lockdowns are the most difficult to implement but could be the most fruitful. In the current climate, vaccines are probably the least issue-inflicting path as we've seen many people have not warmly accepted the concept of a lockdown. With a "perfect" vaccine there is much less reliance on the actions of the public.