**Model file structure**

*\*\*Note: While file names refer to LHS (which refers to Latin Hypercube Sampling), the actual sampling scheme used here is Sobol sequences. The sampling scheme can be modified in LHS\_contactweights.m\*\**

**advance\_in\_time.m** : performs each time step of the simulation. Calls each model pathway in the pathways directory

**Age\_DerpParameters\_MR.csv** : age-structured parameter estimates

**agent.m** :defines the agent class and attributes

**define\_params\_sample.m** : defines all model parameters and names for save files and directories. Calls readparams.m to import values from Age\_DerpParameters\_MR.csv

**determine\_contacts.m** : returns a vector of length nr\_agents whose entries are contact weights between the current agent and all other agents.

**determine\_contacts\_hospitals.m** : returns a vector of contact weights for a hospitalized agent. A hospitalized agent has contact with params.nr\_hosp\_contacts hospital employees.

**define\_params\_sample\_batch\_test.m** : For testing the process for running on a cluster. Similar to define\_params\_sample\_batch\_test.m, except that the number of simulation days is set so only one model time step is performed and the number of replications is 2.

**example\_lhs\_settings\_file.m** : An example sample settings file, with ranges for the varied parameters, the number of samples to perform and the parameter definition file to use. An input to LHS\_contactweights.m or LHS\_perform\_sampling.m.

**example\_params\_file.m** : Parameter definition file similar to define\_params\_sample.m but with fixed parameter values instead of using a sample matrix. Can be used directly as an input for main\_nonspatial\_SEI.m to perform a single run.

**find\_hospital\_workers.m** : identifies agents that work in hospitals by matching data from Washtenaw\_hospitals.csv with the synthetic population/workplace data

**initialize\_agents.m** : defines agents and assigns initial attribute values

**LHS\_contactweights.m** : generates parameter samples from ranges and runs corresponding simulations. Uses LHS\_perform\_sampling.m to set up the parameter matrix and then performs all the runs on the local system.

**LHS\_perform\_sampling.m** : Creates the parameter sample matrix. Used by LHS\_contactweights.m or run directly to set up for batch runs on a cluster.

**main\_nonspatial\_SEI.m** : main function file for running a single simulation

**mich-med-data.csv** : contains testing data for Michigan Medicine employees, for use in comparing with the number of infected hospital workers.

**read\_LHS\_data.m** : function that reads in data\_dir, run\_dir, params\_file, and a logical variable generate\_plots and returns an output structure containing estimated reporting rates, numbers of reported cases, cost function, and PRCC results. If generate\_plots is true, it also plots simulation outputs and fits to data.

**readdata.m** : reads synthetic population data stored in the synthetic-populations directory

**readparams.m** : reads values from Age\_DerpParameters\_MR.csv and converts them to model rate parameters

**storedata\_network.m** : stores which agents belong to different agent classes (exposed, different infected classes, dead) at each time step

**us-counties.csv** : data on case and death counts per day for all US counties, sourced from <https://github.com/nytimes/covid-19-data>

**wash-county-data-from-web.csv** : contains daily data from the Washtenaw County webpage on cases, deaths, hospitalizations, and age and race distributions.

**pathways/**

**death.m** : determines which hospitalized patients will die

**hospitalize.m** : determines which severe infections will become hospitalized

**new\_exposures.m** : determines exposure for susceptible contacts of infected agents. Calls determine\_contacts.m.

**progress\_from\_exposed.m** : determines which exposed agents will progress to symptomatic infection, severe infection, or recovered

**recover\_from\_infected.m** : determines which agents will recover from symptomatic (not sought care), symptomatic (sought care), or hospitalized

**seek\_care.m** : determines which symptomatic (not sought care) agents will seek care

**synthetic-populations/**

**Washtenaw-county-MI/** : synthetic population data for Washtenaw County

**Washtenaw\_hospitals.csv** : contains latitude, longitude, and number of employees for hospitals in Washtenaw County. Source: American Hospital Association annual survey

**Batch Process Files**

**batch\_run.m** : The Matlab file that is used to perform the replications for one parameter set in a single batch job. Invoked from SLURM script job-array-uncompiled.slurm.

**do-batch-runs** : A Bash shell script that simulates a set of batch runs on the local system. Performs the runs by invoking batch\_run.m once for each varied parameter set created by LHS\_perform\_sampling.m. Used for testing the batch process without having to wait on the input queue of a cluster batch system.

**job-array-uncompiled.slurm** : The SLURM script submitted to a batch system. This will be a separate batch job of this script for each varied parameter set. This script invokes batch\_run.m.

**submit-job-array** : Used to submit on a cluster a set of batch runs for a large sample. It requires a separate run directory has been created and which has been setup by running LHS\_perform\_sampling.m to create the parameter samples.

**How to run the model**

The model can be run in the COVID source directory on a local system from within the Matlab GUI interface or on a cluster using the cluster’s batch system (only the SLURM batch system is currently supported).

When run on a local system from within Matlab the output directory can be a sub-directory of the COVID source directory or another directory. All the run functions require an output directory name.

A sub-directory of the COVID source directory can be used for small test runs during model development. This should not be done for production runs.

For production runs, another directory should be used, in a location typically used for production runs of models (ex. ~/ModelRuns). This will make it easier to preserve each production run set for future reference.

Script make-lhs-dir (*\*Kirschner lab only\**) is useful for making run directories for production runs.

When running on a local system from within Matlab, be sure to add sub-directory “pathways” to the Matlab path, either via “addpath(‘pathways’)” in the Matlab command window or selecting the “pathways” sub-directory in the Matlab file browser window and using the right click addpath menu option.

This is not necessary when doing batch runs on a cluster. The batch scripts do this internally.

Note: Storing the full network data takes up a lot of space, so if only summary information is needed then set params.savestates = false in the settings file. If detailed information about which agents belong to each infection state and who they were infected by, then set params.savestates = true.

There are 3 ways to run the model, as described in the following separate sections: single run locally in the Matlab GUI, sample run locally in the Matlab GUI, sample run on a cluster batch system.

System requirements:

* Unix system, either MacOS or Linux.
* A licensed copy of Matlab on all systems where it will be run. On clusters this means a licensed copy of Matlab on each compute node that will be used.

**For a single run on the local system in the Matlab GUI**

1. Create a run directory to store the parameter definition file and the run result files.
2. Copy a parameter definition file to the run directory, ex. example\_params\_file.m.  
     
   Give it a different name, so any that exists in the COVID source directory is not used (since the COVID source directory will be in the Matlab path).  
     
   Rename the function name in the function header line to match the new file name.
3. Define parameter values and file and directory names in the parameter definition file.
4. Run main\_nonspatial\_SEI('run\_dir', 'parameter\_definition\_file', '.')  
   run\_dir is the directory created in step 1 above.  
     
   parameter\_definition\_file is the name given to the parameter definition file in run\_dir (do not include the path, just the file name, main\_nonspatial\_SEI will handle the path). **DO NOT INCLUDE “.m” AT THE END OF THE NAME.**  
     
   '.' is for the data directory, the directory containing the csv files and sub-directories with addition csv files used by the model. '.' means the current directory, which is the COVID source directory. This argument is needed for cluster batch runs, which are not run in the COVID source directory, but need to know the location of the data files.

**For running a sample on the local system in the Matlab GUI**

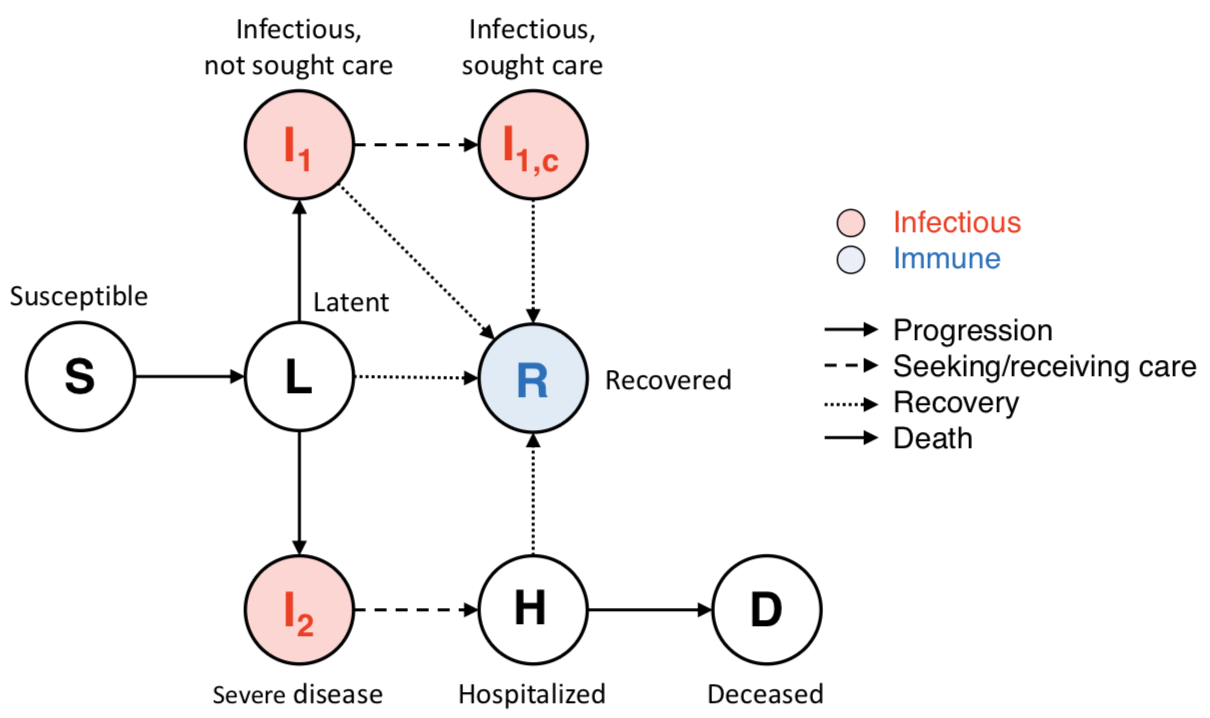
1. Create a run directory to store the sample settings file, parameter definition file and the run result files.
2. Copy a parameter definition file to the run directory, ex. define\_params\_sample.m.  
     
   Give it a different name, so any that exists in the COVID source directory is not used (since the COVID source directory will be in the Matlab path).  
     
   Rename the function name in the function header line to match the new file name.
3. Define parameter values and file and directory names in the parameter definition file.
4. Copy a sample settings file to the run directory, ex. example\_lhs\_settings\_file.m.  
     
   Give it a different name (for the same reason as for the parameter definition file) so any that exists in the COVID source directory is not used (since the COVID source directory will be in the Matlab path).  
     
   Rename the function name in the function header line to match the new file name.
5. Define the sample settings as needed.  
     
   Define number of samples to run and the parameter ranges for the varied parameters.  
     
   Make sure to update the setting for which parameter definition file to use (settings.parameterFunctionName) to correspond to the renamed parameter definition file. **DO NOT INCLUDE “.m” AT THE END OF THE NAME.**  
     
   Define settings.data\_dir = '.', since the data directory (with the csv files) will be the current directory when run from within the Matlab GUI.  
     
   If you want to sample different parameters than are currently being sampled, edit define\_params\_sample.m so that those parameters read in values from sample\_mat.
6. Run LHS\_contactweights('run\_dir', 'settings\_file\_name').  
     
   run\_dir is the directory created in step 1 above.  
     
   settings\_file\_name is the name given to the sample settings in run\_dir (do not include the path, just the file name, LHS\_contactweights will handle the path).  
     
   LHS\_contactweights runs LHS\_perform\_sampling. LHS\_perform\_sampling is also used when doing batch runs on a cluster. It writes some text files (.txt) files needed for batch runs but not needed by interactive runs within the Matlab GUI. These can be ignored.

**For running a sample on a cluster batch system**

1. Perform steps 1 to 5 in the section above “For running a sample on the local system in the Matlab GUI”.  
     
   One difference is that in the sample settings file, settings.data\_dir should be set to the full path name of the COVID source directory. Ex.:  
     
   settings.data\_dir = '/home/renardy/immunology/COVID';
2. Copy file job-array-uncompiled.slurm from the COVID source directory to the run directory.  
     
   The submission script expects the SLURM batch script to be in the run directory.  
     
   Do not edit job-array-uncompiled.slurm.
3. Run LHS\_perform\_sampling('run\_dir', 'settings\_file\_name').  
     
   Do this from the COVID source directory. You can do it on a local system from within Matlab and then copy the run directory to the cluster or you can run Matlab from the command line on the cluster (matlab -r “addpath('pathways');LHS\_perform\_sampling('run\_dir', 'settings\_file\_name');exit”  
     
   run\_dir is the directory created in step 1 above.  
     
   settings\_file\_name is the name given to the sample settings in run\_dir (do not include the path, just the file name, LHS\_perform\_sampling will handle the path).
4. cd to the run directory.  
     
   The batch jobs must be submitted from the run directory.
5. In the run directory, run script submit-job-array.  
     
   Use the full path to invoke the submission script. The submission script uses that to determine the name of the COVID source directory, to use as the data directory (where the csv files are) when submitting the runs.  
     
   ~/immunology/COVID/submit-job-array wall-time-limit  
     
   On lab cluster mycobacteria (where there is no effective wall time limit, so a very large one can be used):  
   ~/immunology/COVID/submit-job-array 999:00:00  
     
   On other clusters use a more realistic wall time limit. Make sure to make it big enough, as the COVID model does not have check point/restart capability. If a batch job hits a wall time limit, it will need to be rerun from the beginning, with a longer wall time limit.

**Model assumptions and parameters**

Model structure:



Parameter values from the University of Michigan COVID-19 Modeling group:

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Description** | **Value** | **Units** |
| fReport | Fraction of symptomatic individuals who are lab confirmed | 0.1 | Unitless |
| R0 | Basic reproduction number | 2 | People |
| IncPer | Incubation period | 5 | Days |
| InfPer | Infectious period | 7 | Days |
| fMort | Mortality fraction among infected individuals | Age-dependent | Unitless |
| TimetoDeath | Time from symptom onset to death | 18.5 | Days |
| fAsymp | Fraction who are asymptomatic or only experience mild symptoms | 0.18 | Unitless |
| fDocVisit | Fraction of symptomatic who will visit the doctor | 0.5 | Unitless |
| TimetoSeekCare | Time to visit doctor (non-hospital) | 2.5 | Days |
| fHosp | Fraction of symptomatic who will be hospitalized | Age-dependent | Unitless |
| TimetoHosp | Time from symptom onset to hospitalization | 11 | Days |
| StayLengthHosp | Duration of hospital stay | 11 | Days |
| RelativeDeathRep | Relative reporting of deaths vs cases | 2 | Unitless |
| L0 | Initial proportion of population who is latent (exposed) as of March 8, 2020 | 1e-4 | Fraction |
| I0 | Initial proportion of population who has symptomatic infection as of March 8, 2020 | 1e-5 | Fraction |

Calculating model parameters from parameter table:

Initial fraction in I1 = (1-fHosp(a))\*I0

Initial fraction in I1c = 0;

Initial fraction in I2 = fHosp(a)\*I0;

Initial fraction in H = 0

Initial fraction in exposed (L) = L0

Transmission rate = R0/(InfPer\*TimetoHosp)/[(-1+fAsymp)\*((-1+fHosp(a))/TimetoHosp – fHosp(a)/InfPer)]

Rate of progression from L to I1 = (1-fHosp(a))\*(1-fAsymp)/IncPer;

Rate of progression from L to I2 = fHosp(a)\*(1-fAsymp)/IncPer;

Rate of progression from I1 to I1c = 1/TimetoSeekCare;

Rate of progression from I2 to H = 1/TimetoHosp;

Rate of recovery from L = fAsymp/IncPer;

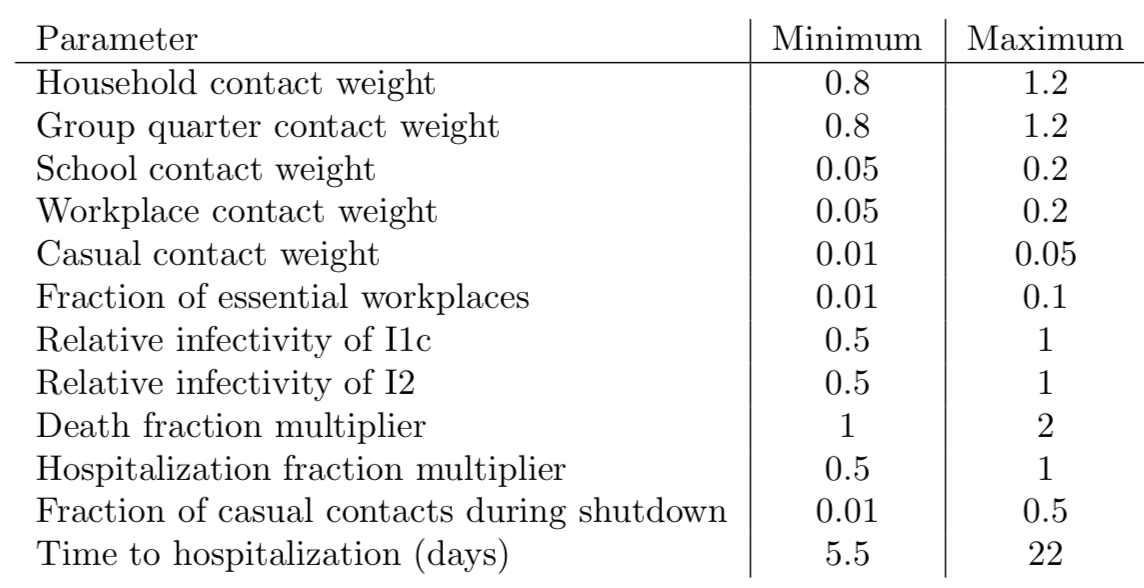
Rate of recovery from I1 = 1/InfPer;

Rate of recovery from I1c = 1/InfPer;

Rate of recovery from H = [1-fMort(a)/fHosp(a)]/StayLengthHosp;

Rate of death from H = [fMort(a)/fHosp(a)]/StayLengthHosp;

Ranges for parameters that are varied:



Key assumptions:

* Everyone has 10-50 casual contacts (randomly assigned)
* Individuals at large workplaces/schools have 10-50 workplace/school contacts (random)
* If hospitals are utilized, hospitalized patients have contact with 10 healthcare workers
* Hospitals are identified based on their latitude/longitude coordinates and number of workers living in the county. This data is obtained from the American Hospital Association annual survey.
* Healthcare workers are identified as all agents who work at a hospital, regardless of infection status
* School contact weight goes to 0 on March 16, the date of county-wide school closures.
* Workplace contact weight decreases linearly to 0 over 7 days prior to closure of non-essential businesses (March 23)
* Casual contact weight decreases linearly to a fraction of its initial value over 7 days prior to the stay at home order (March 24)
* A fraction of workplaces are “essential” and thus their employees keep all their workplace contacts. These are randomly chosen among all workplaces. (This fraction can be varied to simulate reopening businesses gradually, can also be made a function of time)
* College dorms are removed entirely. Individuals in other types of group quarters (e.g. prison, nursing homes) have no casual contacts since there are no visitors, but do mix among themselves.
* Model parameters are age-dependent for hospitalization and mortality rates. Other parameters are constant and do not vary from individual to individual.
* “Observed” cases are estimated from taking the sum of cases in compartments I1c and H (i.e., people who have sought medical care), multiplied by respective reporting rates.
* Reporting rates are constant over time
* We assume the latent period is equal to the incubation period.