

FACS: A geospatial agent-based simulator for analysing COVID-19 spread and public health measures on local regions

Imran Mahmood, Hamid Arabnejad, Diana Suleimenova, Isabel Sassoon, Alaa Marshan, Alan Serrano, Panos Louvieris, Anastasia Anagnostou, Simon J E Taylor, David Bell, Derek Groen

Department of Computer Science, College of Engineering Design and Physical Sciences, Brunel University, Kingston Lane, Uxbridge, London, UB8 3PH, UK

ARTICLE HISTORY

Compiled July 16, 2020

ABSTRACT

The recent Covid-19 outbreak has had a tremendous impact on the world, and many countries are struggling to help incoming patients and at the same time, rapidly enact new public health measures such as lock downs. Many of these decisions are guided by the outcomes of so-called Susceptible-Exposed-Infectious-Recovered (SEIR) models that operate on a national level. Here we introduce the Flu And coronavirus Simulator (FACS), a simulation tool that models the viral spread at the sub-national level, incorporating geospatial data sources to extract buildings and residential areas within a predefined region. Using FACS, we can model Covid-19 spread at the local level, and provide estimates of the spread of infections and hospital arrivals for different scenarios. We validate the simulation results with the ICU admissions numbers obtained from the local hospitals in the UK. Such validated models can be used to support local decision-making for an effective health and social care capability response to the epidemic. For the convenience of the reader, the source code and pre-processing infrastructure are publicly released ¹

KEYWORDS

Agent-based Simulation, COVID-19 Spread, Location Graph, lock down scenarios, Epidemiology, Model Validation

1. Introduction

Each year, millions of people are exposed to serious health risks due to emerging infectious and communicable diseases. This poses a severe threat to public health at local, regional or national and international level. Coronavirus disease (COVID-19) is an infectious disease caused by the severe acute respiratory syndrome (WHO, 2020). The SARS outbreak in 2003, caused by SARS-CoV-1, resulted in more than 8000 cases and 800 deaths. It was eventually contained as a result of surveillance, prompt isolation of patients and strict quarantine of all contacts (Chan & Xu, 2003). In December 2019 in Wuhan, China, the related virus strain of severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) was discovered and has since spread globally, resulting in an ongoing pandemic. As of 7th June 2020, more than 6.66 million cases have been

CONTACT Imran Mahmood Email: imran.mahmood@brunel.ac.uk

¹<https://github.com/djgroen/facs/blob/master/README.md>

reported across 188 countries and territories, resulting in 392,802 deaths. More than 3.1 million people have recovered (CSSE, 2020), (ECDC, 2020). On the 11th March 2020, the World Health Organisation declared COVID-19 as a controllable pandemic, and the countries at risk were advised to step up counter measures (WHO, 2020).

To effectively prepare for the emergency response and mitigate the challenges involved in the front line defence against the COVID-19 outbreak, decision-makers, public health teams, healthcare planners and hospital managers need reliable and reproducible forecasts for the spread of COVID-19 in the local regions (Wynants et al., 2020). To do this, we need a computational model that encompasses the characteristics of the virus (e.g., its transmissibility, means of the spread, temporal phases and infectivity), as well as the geographical aspects of the local region, the dynamical movements and interaction of people within it. It is computationally challenging to both realistically map the real-world characteristics of a large region (e.g., city scale, population demography and behavioural dynamics, urban infrastructures and population mobility) and account for uncertainties in a large number of key parameters.

We choose to address these challenges by combining the body of knowledge from three prominent domains: (a) Agent-based Simulation (ABS) and (b) High Performance Computing (HPC). ABS enables the abstraction of disease dynamics and captures the population behaviour at a greater level of detail. The rationale for using this approach in disease modelling is the fact that the real-world is made of complex social and natural systems (e.g., infectious disease epidemiology). These are multi-scale, multi-resolution and multi-dimensional, where entities (e.g., hosts and pathogens) are themselves complex in their actions, and are interconnected, inter-dependent and interactive among each other (Miller & Page, 2009). These adaptive systems are complex due to decentralisation of decision making, distributed autonomy and the heterogeneity of behaviour of individuals in a concurrent execution environment.

ABS helps in studying complex adaptive systems through a systematic abstraction of the system in a ‘bottom-up approach’. In which a single monolithic model is replaced with a population of individual agents that have their own characteristics and behaviours (Bonabeau, 2002; Macal & North, 2010; Taylor, 2014). Each agent individually assesses its situation and makes decisions on the basis of a set of rules. ABS is essentially decentralised, meant to simulate the actions and interactions of individuals or collective entities, in their environment (Taylor, 2014). As a result of this heterogeneity and stochasticity embedded in their individual behaviour, a meaningful emergent phenomenon appears from their local properties and interactions. Emergence is the way complex systems, patterns and higher-level system properties arise out of a multiplicity of relatively simple behavioural interactions (Railsback & Grimm, 2019). This emergence, although not explicitly programmed in the model, can have properties that are decoupled from those of the individual components and is a distinguishing feature of ABS. HPC provides an exascale run-time environment for running large number of ensembles or replications of simulations with large time periods and large spatial scales and populations. For shorter time periods, smaller regions and limited populations, the use of HPC can be considered optional.

In this paper we present the agent-based *Flu And Coronavirus Simulator* (FACS) (Groen, Mahmood, & Arabnejad, 2020), which incorporates geospatial data, epidemiological data, disease parameters, population dynamics and demography for a given region. FACS models the transmission of COVID-19 (or other viruses as defined in the input) and simulates the COVID-19 outbreak in a given region using computational disease dynamics at a local (e.g., city or borough) scale. It can be used to forecast the number of infectious cases spread, to identify spatial hot spots across the region, to

estimate the number of infected patients arriving at hospitals, to evaluate various lock down scenarios and counterfactual, and to generate early warnings for rapid emergency response management. With the development of FACS, we aim to address the following research questions:

- (1) Which establishments and areas are the main drivers of the spread of the epidemic within a single region?
- (2) How much of an increase in admissions to the Intensive Care Unit (ICU) of a hospital can we expect when specific individual lock down measures are lifted or applied?

We conduct a COVID-19 case study using FACS framework as a proof of concept, and demonstrate the steps required to obtain estimates for the above research questions. We validate the simulation results with the data obtained from the local hospitals in the UK. For interested readers we present a STRESS (Strengthening the Reporting of Empirical Simulation Studies) checklist report (Mahmood, Anagnostou, Taylor, Bell, & Groen, 2020) to describe the proposed ABS framework, to help improve the the understanding of our simulation model, and to capture relevant details of the COVID-19 simulation study. This allows simulation reproducibility and ease of extend/reuse of implementation (Monks et al., 2019).

The rest of the paper is organised as follows: Section 2 presents an overview of different methods of disease simulation and a comparison of conventional approaches with the selected agent-based approach. Section 3 covers the details of the proposed simulation framework as our main contribution. Section 4 illustrates the case study of COVID-19 simulation, the simulation results and model validation and Section 5 presents the conclusions and future work.

2. Methods of Disease Modelling and Simulation

In this section we review various approaches used for modelling and simulating infectious diseases, with a particular focus on COVID-19. We first review a selection of conventional modelling approaches and then examine related agent-based modelling approaches. To conclude this section, we present a brief comparison of different methods.

2.1. Conventional Disease Modelling Approaches

2.1.1. Mathematical Models

Complex systems such as disease epidemics are traditionally modelled using mathematical equations, and despite the recent nature of the pandemic many works have already emerged. The Susceptible-Infected-Recovered (SIR) disease model, and variations of it, is particularly widely applied. These models use Ordinary Differential Equations (ODEs) to estimate infection spread (Volz & Meyers, 2007), and have for instance been applied to model the spread of COVID-19 in China (Chen et al., 2020). One variation of the SIR model incorporates an additional Dead state (resulting in a SIRD model) or Exposed (resulting in a SEIR model). Anastassopoulou et al. (Anastassopoulou, Russo, Tsakris, & Siettos, 2020) used the SIRD model to estimate the basic reproduction number (R_0), and calibrated it using report data to forecast the evolution of the outbreak in China. Using a similar approach, Kucharski et al. (Kucharski et

al., 2020) used modelled SARS-CoV-2 transmission with four data sets from within and outside Wuhan, to estimate how transmission in Wuhan varied during the winter of 2020. Peng et al. (Peng, Yang, Zhang, Zhuge, & Hong, 2020) adopted a SEIR model, while Hellewell et al. (Hellewell et al., 2020) made a customized mathematical model to assess the effects of case isolation and contact tracing on the transmission of COVID-19.

2.1.2. System Dynamics Models

A class of models known as System Dynamics (SD) is widely used to understand and capture the nonlinear behaviour of complex systems over time. It relies on mechanisms such as stocks, flows, internal feedback loops, table functions and time delays (Forrester, 1994). SD is mainly used to implement compartmental models of infectious diseases (Bagni, Berchi, & Cariello, 2002) (M. Li et al., 2019) (Bordehore, Navarro, Herrador, & Fonfria, 2020).

2.1.3. Discrete Event Simulation

Discrete event simulations (DES) are commonly used to model specific locations such as hospitals (Jacobson, Hall, & Swisher, 2006), but there are also relevant examples that have been applied in a wider epidemiological setting. Perumalla et al. (Perumalla, 2020) developed a massively scalable discrete event simulator for the epidemiology of COVID-19, while Balcan et al. (Balcan et al., 2010) integrated empirical mobility networks in a computational epidemic model to analyse discrete stochastic epidemics worldwide. This tool was used by Chinazzi et al. (Chinazzi et al., 2020) to study the effect of travel restrictions on the spread of the COVID-19 outbreak.

2.1.4. Data Driven Modelling

Data-driven modelling approaches are another class of models that have been widely used to provide new insights into the COVID-19 pandemic. Huang et al. (Huang et al., 2020) present a range of treatment, and clinical outcomes of laboratory-confirmed cases infected with SARS-CoV2 using a data-driven model, while Verity et al. (Verity et al., 2020) used data from Macau and Hong Kong to estimate the fatality ratio across the spectrum of COVID-19 disease. Li et al. (Q. Li et al., 2020) used laboratory-confirmed cases reported in January 2020 to estimate key epidemiological time-delays, periods of exponential growth and the reproductive number. Prompetchara et al. (Prompetchara, Ketloy, & Palaga, 2020) performed a different type of study, reviewing aspects such as host-pathogen interaction, host immune responses and pathogen immune evasion strategies to highlight the design for an immune intervention for COVID-19.

2.1.5. Machine-learning Techniques

Machine-learning techniques are used by a range of groups to further improve the quality of data-driven models. For instance, Dandekar et al. (Dandekar & Barbastathis, 2020) used a machine learning algorithm to help predict when infections will slow down in each country, while Lorch et al. (Lorch et al., 2020) applied a spatio-temporal Bayesian optimisation-based epidemic model to quantify the effects of contact tracing, testing, and containment of COVID-19 in Germany. Liu et al. (Liu et al., 2020) focused on Exponential Growth (EG) and maximum likelihood estimation (MLE) methods to estimate the reproductive number (R) of SARS-CoV2 and SARS in China. Lastly,

Markov chain Monte Carlo (MCMC) methods for the analysis of infectious disease have also been frequently used (see e.g., O’Neill, Balding, Becker, Eerola, and Mollison (2000)).

Despite their customary usefulness, the conventional modelling approaches often have limitations because they treat the system components as homogeneous entities. For example, in differential equation models or System Dynamics, the individuals are represented only using numerical variables, quantities or stocks which aggregates the entity behaviour at a macro-level. Therefore the dynamics of the spread of disease can only be implemented on the cohort as a whole using a centralised mathematical formula or instruction set. Similarly, the data driven approaches such as Bayesian Inference, Regression and Machine learning are solely dependent on the data, and cannot perform well when only a sparse data with partial knowledge of underlying phenomenon of the real-system is available at hand. In addition, these set of models fail to express properties of complex systems, such as hierarchical forms, collective behaviour, evolution and adaptation, spatial or a-spatial networks, due to their top-down nature and monolithic logic (Bar-Yam, 2002).

2.2. Agent-based Disease Modelling Approaches

ABS is increasingly used to model emerging diseases, such as COVID-19, where the key disease parameters and assumptions are provisional or unknown (Mahmood, Jahan, Groen, Javed, & Shafait, 2020; Miksch, Urach, Einzinger, & Zauner, 2014; Moghadas et al., 2017; Tuomisto et al., 2020; Venkatramanan et al., 2018). ABS supports the simulation many agents and their interactions, allowing researchers to recreate and predict the emergence of complex phenomena from individual behaviours (Railsback & Grimm, 2019).

Because the agents are explicit, ABS provides a flexible environment where interventions can easily be introduced on the individual level. These models also make it easier to express heterogeneity in agents and non-linearity in their behaviours. For example, the use of ABS in the implementation of lock down scenarios and other control measures is straightforward because agents are able to ‘socially interact’ with each other and with their virtual environment, which is in contrast with the mean-field approaches used in many conventional models.

ABS works using rule sets defined in a bottom-up manner. These rules are usually derived from published literature, expert opinion, or empirical data (Moghadas et al., 2017). For example, the rules that govern the dynamics of the spread of an infection across an agent population can be based on interactions between susceptible and infectious individuals in the environment through agent contact. Because agents are autonomous decision making entities in the model, their decisions to interact determine the possibility of disease transmission. This preliminary logic can be extended by adding agent mobility to the environment, to study how their movement patterns influence the disease spread.

2.2.1. Existing Frameworks

Here we review several prominent works on COVID-19 simulation using ABS. Most notably in the UK, Ferguson et al.(Ferguson et al., 2020) published a report that used an individual-based micro-simulation epidemiological model of two countries: UK and USA, to examine different public health measures, referred to as ‘non-pharmaceutical interventions’ (NPIs). These NPIs aim to mitigate and/or suppress the spread of

COVID-19 by reducing contact rates in the population and thereby reducing transmission of the virus. Klepac et al. (Klepac et al., 2020) analyses a set of fine-scale, age-specific social mixing patterns extracted from the self-reported contact data of 36,000 volunteers in UK. They generated contact matrices by context (home, work, school, other) and type (conversational or physical) of contact, to analyse and evaluate strategies that reduce the amount of mixing in the population, such as school closures, social distancing, or working from home. Murray et al. (Murray, 2020) created a model that helps forecast the impact of the first wave of the COVID-19 pandemic on hospital demand and deaths for the USA and other countries. Chang et al. (Chang, Harding, Zachreson, Cliff, & Prokopenko, 2020) presents an agent-based model for a fine-grained computational simulation of the COVID-19 spread in Australia. The proposed model accounts for reproductive number, the length of incubation and generation periods, age-dependent attack rates, and the growth rate of cumulative incidence for the local transmission of the disease. As a last example, Candido (Candido, 2020) developed an open source ABS framework to simulate a population of agents with different mobility patterns and analyse the spread of COVID-19 under different control scenarios.

Most of these works use ABS to better express the properties of complex disease dynamics at a fine-grained resolution but ABS, too, has its limitations. For example, there is limited use of spatial data for the model construction, limited or no notions of demographic patterns, simplistic disease transmission, with minimal or no abstraction of lock down or any other control measures, and the execution performance bottlenecks can result in having to run simulations with unrealistically few agents.

2.2.2. Proposed Framework

In this paper, we use ABS to study the complex dynamics of the COVID-19 epidemic and demonstrate how our approach can help answer several key research questions. Our proposed framework is different to the aforementioned works in the following areas:

- FACS provides an open ended platform for the specification and implementation of the primary components of ABS: (i) Agents; (ii) Virtual environment and (iii) Rule-set using a systematic Simulation Development Approach.
- FACS inherits features of a comprehensive simulation framework from its ancestors: (i) FLEE (Groen & Arabnejad, 2015) and (ii) FabSim3 (Groen & Arabnejad, 2014). Where, FLEE mainly specialises in ABS complex dynamics e.g., agent movements; FabSim3 is provides the ability to simulate a large population of agents with microscopic details using remote supercomputers. The combination of this legacy code offers numerous benefits including: high performance, high scalability and greater re-usability through model coupling. Hence it provides an open-ended API for modellers and programmers to use it for further scientific research and development.
- FACS generalises the process of disease modelling and provides a template to model any infectious disease. Thus allowing non-programmers (e.g., epidemiologists and healthcare data scientists) to use the framework as a disease modelling suite. In this paper we present the case study of COVID-19 simulation as a proof of concept.
- FACS offers a built-in location graph construction tool that allows import of large spatial data-sets (e.g., Open Street Map), automated parsing and pre-processing of the spatial data and generating buildings of various types, thus allowing an ease in the synthesis of the virtual environment for the region under considera-

- tion. We plan to extend this tool with more features like public transport, roads network analysis and street-level routing.
- FACS provide a realistic disease transmission algorithm with the ability to capture population interactions and demographic patterns e.g., age diversity, daily-life activities, mobility patterns, exposure at the street-level or in public transportation, use or no use of face mask, assumptions of exposure within closed quarters

3. FACS - Simulation Framework

In this section, we describe our proposed simulation framework in detail. FACS is an ABS tool that focuses on COVID-19 transmission within the context of a town, small city, or borough. It combines disease properties, geospatial information and basic demographic information to attempt a forecast of expected infection and hospitalisation rates across the region. It also supports the implementation of, and a lifting of, a wide range of mitigation scenarios and lock down measures. The code is implemented in Python3, and relies on OpenStreetMaps (OSM) for the underlying geospatial data.

3.1. Simulation Development Approach

In this section, we describe the Simulation Development Approach (SDA) used to build the COVID-19 simulation, which is an adaptation of the generalised ABS development approach previously presented by Suleimenova (Suleimenova, Bell, & Groen, 2017). We propose the extension of the SDA presented in a series of following steps and as shown in Figure 1. This figure contains the road map to develop a disease simulation using the FACS framework.

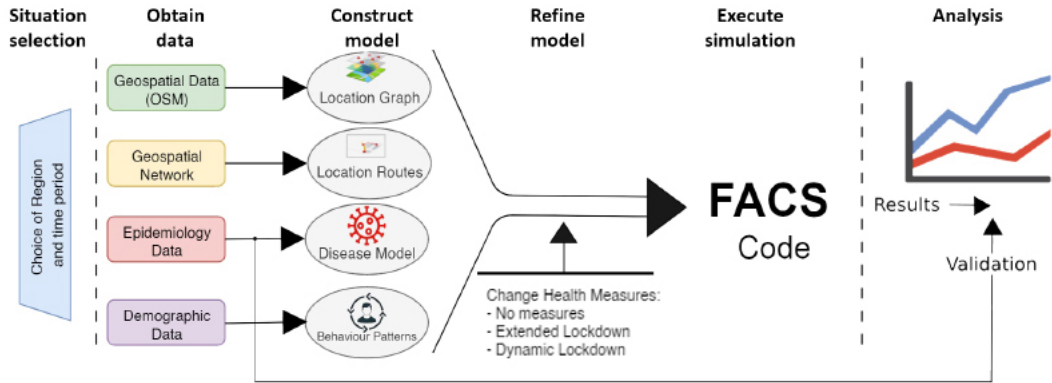


Figure 1. Simulation Development Approach

3.1.1. Step 1: Situation Selection

At first a modeler needs to define a spatial and temporal scale of the model. The modeler chooses a region usually a sub-national geographical area (e.g., a small city or a borough within London, UK) and a time-period (start date, end date) of the intended simulation. FACS uses a time step size of 1 day, and the typical simulated time is between 3 months and 2 years.

3.1.2. Step 2: Obtain Data

The modeler then needs to prepare the inputs. This concerns with the historical data in most cases. For the disease-related parameters it includes parameters in accordance with the state-of-the-art knowledge of the disease. This step mainly involves gathering and pre-processing the following input data:

- (1) Geo-spatial data of the selected region for the construction of Location Graph;
- (2) Geo-spatial network data of the selected area;
- (3) Epidemiology data of the disease under consideration (e.g., COVID-19) to formulate the disease model;
- (4) Demographic data of the local population for the initialization of behavioural patterns (e.g., mobility of people between different locations, time of the movement, length of stays).
- (5) Data specifying the disease model ² (details are given in Disease Model (see section 3.3.1).
- (6) Data detailing the type of interventions undertaken in the past, and those that are to be modelled as part of the forecast

We provide comprehensive documentation on how to perform these activities as part of the FACS documentation ³.

3.1.3. Step 3: Construct Model

Next, the modeller needs to use the inputs from step 2 to construct the model artifacts and initial configurations. These include:

- (1) Location Graph (LG): Here we collect an Open Street Map (OSM) data file, crop it using an area boundary polygon, and parse it to obtain different types of locations including: 'hospital', 'house', 'office', 'park', 'leisure', 'school', 'supermarket', 'shopping' (as shown in table 1). When generating the location graph, we rely on the following assumptions:
 - (a) Houses are spawned within the residential areas marked by OSM. Since there is insufficient tagging and OSM data does not resolve to the types of residences such as detached, semi-detached, terraced, flats, so all houses are assumed to have an identical structure, with two households well separated inside a single location. Therefore, each house building is assumed to contain two households by default.
 - (b) By default, offices are spawned at random locations. This is to mimic the fact that most people may work outside of the borough's boundaries. Office location affects who is likely to go there, but has no effect on adjacent buildings in the model.
 - (c) People spend on average a given number of minutes per week at different location types. We base this on their age as shown in figure 6. We also assume that this duration is spread across a number of 'visits', and have defined an average visit duration for each location type.
 - (d) All people have the same needs e.g., everyone currently require 40 minutes/week in the hospital, except when they are hospitalised or dead. Hospitals include all healthcare settings, and the 40 minutes per week is an average.

²https://github.com/djgroen/facs/blob/master/covid_data/disease_covid19.yml

³<https://facs.readthedocs.io/en/latest/preparation.html>

- (e) People always visit the amenities/facilities closest to them.
- (f) People at home are assumed to be within 2m of each other 5% of the time.
- (g) The population age distribution is sampled from a borough- or country specific age distribution data.
- (h) Household size is randomly generated between 1 and 4, using a uniform distribution.

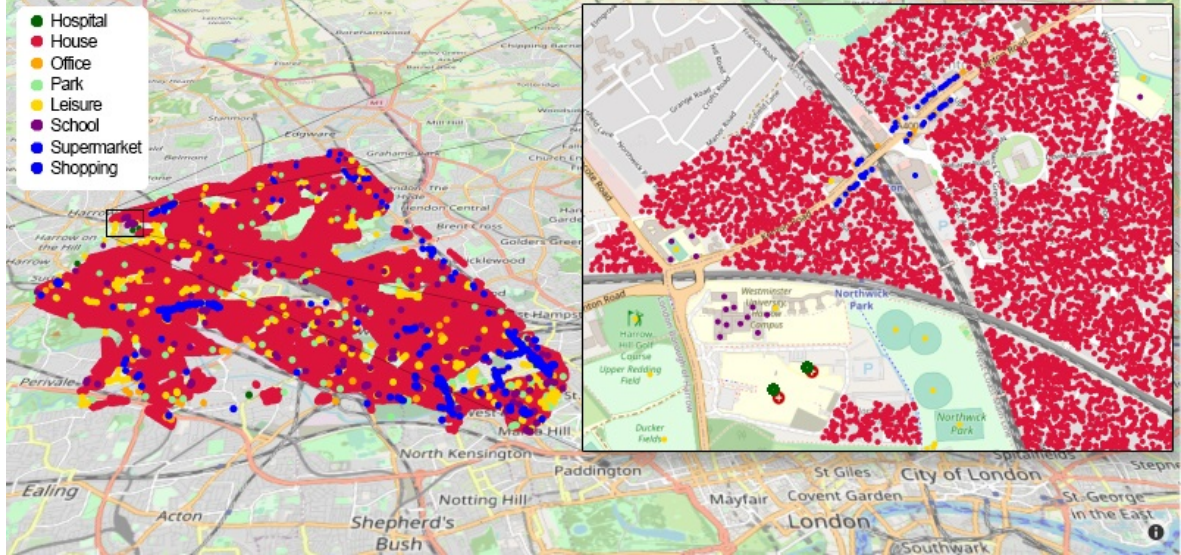


Figure 2. Location Graph (visualised using Plotly API)

- (2) Disease Model : In this step we specify the key characteristics of the disease using YAML ⁴ file. This file contains parameters and assumptions used to implement the logic for epidemic spread. In the context of infectious disease dynamics, there are usually three important time periods: (i) the pre-infectious or latent period, defined as the time from the exposure to when a host is able to transmit the agent on to another host; (ii) the incubation period, defined as the time from infection to the onset of clinical disease; and (iii) the infectious period, defined as the period from the end of the pre-infectious period until the time when a host is no longer able to transmit the infection to others (Vynnycky & White, 2010).

The following parameters and key assumptions are used for COVID-19 disease model:

- (a) Infection rate = 0.07
When one infectious and one susceptible person spend 24 hours in a 4m² area, the likelihood of the susceptible person to become infected is equal to the infection rate 0.07, or 7% (W. Li et al., 2020), (Bi et al., 2020), (Mizumoto & Chowell, 2020).
- (b) When a susceptible person resides at a location where infectious persons also reside on the same day, then the probability of the infection is given by 4.1

⁴YAML is a human friendly data serialisation standard for all programming languages <https://yaml.org/>

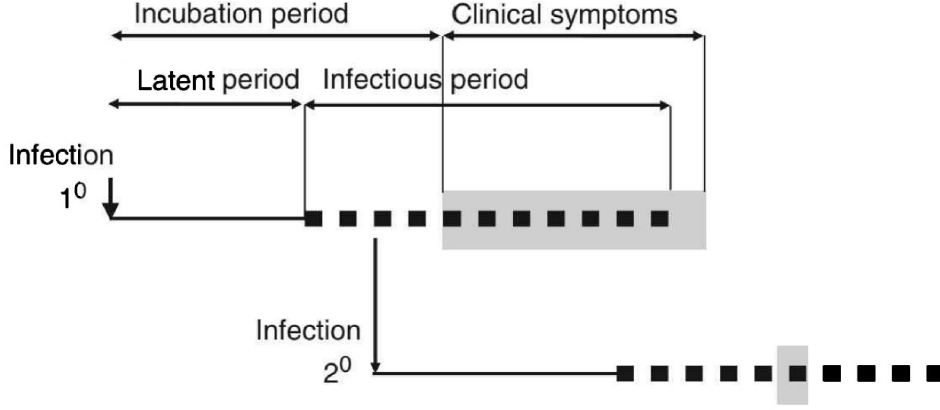


Figure 3. Different periods of infectious disease (Venkatramanan et al., 2018)

$$\mathbb{P}_{inf} = \left(\frac{LSs}{OD_{loc}} \times \frac{LSi}{A_{con}} \right) \times \frac{IR}{360} \times M \quad (4.1)$$

where:

\mathbb{P}_{inf} is the probability of the susceptible person become infected.

LSs is the length of stay of the susceptible person [minutes]. LSi is the length of stay of the infectious person [minutes].

OD_{loc} is the opening duration of the location on that given day.

A_{con} is area of the location [in m^2].

IR is the infection rate (i.e., 0.07)

M is a static contact rate multiplier that has a value of 1 by default, but that is modified when certain public health interventions are triggered ⁵

- (c) Incubation Period = 3.5 days.

We assume that patients are not infectious in the incubation period. 5 days of incubation, infectiousness starts 36 hours before the end of incubation (Zhou et al., 2020).

- (d) Time period to hospitalisation = 12 days (average)

- (e) Mortality rate = 1.6% (this is provided as a rough indicator only, and we expect other models to predict these rates more accurately in due course)

- (f) Mortality per Age = 0 - 0.134 as shown in figure 4

- (g) Period to hospitalisation for severe cases: 12 days

- (h) Period to either mortality or recovery for severe cases: 8 days starting from the day of hospitalisation (ICU)

- (i) Period to recovery for mild cases: 8.5 days (This is within the 7–10 day range specified by the NCID (NCID, 2020))

- (j) The transitions between different states of the COVID-19 disease are presented using SEIRD state chart as shown in figure 5.

- (3) Demographic data of the local population for the initialisation of behavioural patterns (e.g., mobility of people between different locations, time of the move-

⁵Note that hospitals have a reduced infection rate, due to the safety precautions taken there (e.g., the use of personal protective equipment (PPE) and effective quarantining of patients). We reflect this protective aspect using a multiplier that initially reduces the infection rate in hospitals by 50%, and that increases in value as the simulation progresses and hospitals adopt improved practices and have more PPE, to a final value of 92%.

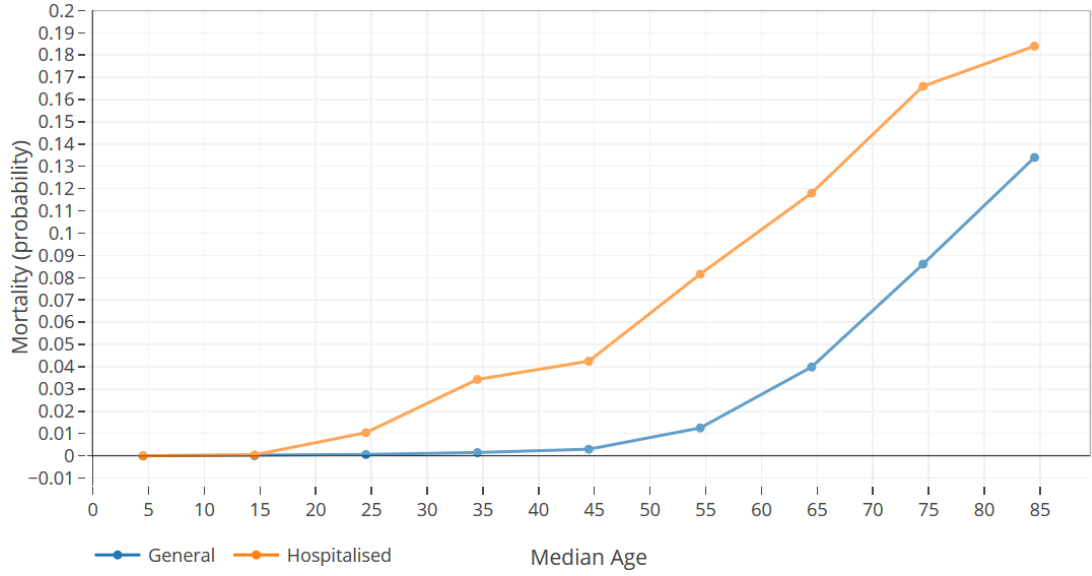


Figure 4. Mortality per age (Verity et al., 2020)

Location Type	Duration (min)
House	-
Park	90
Hospital	60
supermarket	60
Office	360
School	360
Leisure	60
Shopping	60

Table 1.: Location Types

ment, length of stays). Different types of locations and the average length of stays are shown in table 1. Figure 6 shows the radar chart of the average length of stays at a particular building type per age. The behavioural patterns of the population are stored in YAML files for the simulation input.

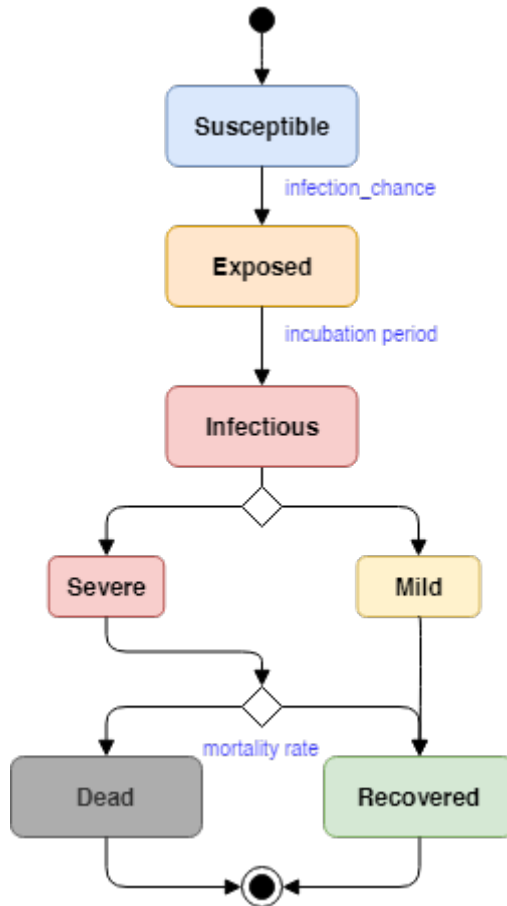


Figure 5. Susceptible - Exposed - Infected - Recovered - Dead (SEIRD) State-chart

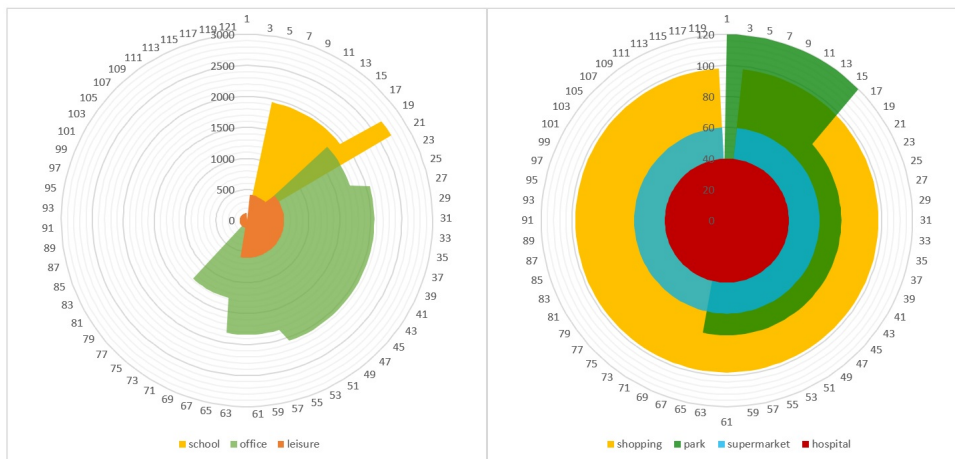


Figure 6. Radar Chart showing the average length of stay at a building type by age

3.1.4. Step 4: Refine Model

In this step, the modeller is able to define and manage different types of public health interventions for the simulation runs. A wide range of interventions are possible, in-

cluding:

- (1) Close all schools and leisure locations.
- (2) Close 80% of all shops.
- (3) Enact Social Distancing (using the assumptions from (Ferguson et al., 2020))
- (4) Implement Case isolation (using the assumptions from (Ferguson et al., 2020)).
- (5) Implement Work from Home directive (causes 75% of the workforce to no longer go to the office).
- (6) Encourage the public to wear masks (expected compliance can be set manually).
- (7) Household isolation (assuming 80% of non-infected people in an infected household reduce their non-house contacts by 75%. We took a higher compliance rate than Imperial Report that used in (Ferguson et al., 2020), because the household isolation is imposed by the government, and not really voluntary)
- (8) Lifting and (re-)enacting any of these measures at given points in the simulation.

3.1.5. Step 5: Execute Simulation

The next step is to run the simulations. This can be done using either a local machine or a remote supercomputer. Building a simulation execution workflow is normally time-consuming. Steps such as exploration of different parameters, construction of network maps, and execution of multiple runs require careful planning and scrutiny. To simplify these steps, we developed an automation toolkit, namely FabCovid19 (Arabnejad, Groen, & Mahmood, 2020), which is a plugin of FabSim3 toolkit (Groen et al., 2016). FabSim3 is a Python-based automation toolkit that enables users to create, manage, execute, curate, analyse and modify these complex and dynamic ensemble workflows. FabSim3 integrates with a range of super-computing platforms, and features a flexible plugin system that helps users to easily automate simulations tasks in diverse domains. FabCovid19 is a partially automated implementation of our COVID-19 simulation, and provides an environment for researchers to construct and modify COVID-19 simulations, instantiate and execute multiple runs for different rules, parameters and assumptions. It also automates some aspects of the post-processing.

FACS Simulator is a platform independent software that can easily be installed on Windows, Mac OS or Linux with latest Python run-time environment. A user can choose to either run it on a desktop PC for local execution or on a remote super computing platform. For local execution a desktop PC with Corei7 processor, 16GB RAM and 256GB SSD storage will be an optimal setting. For HPC we utilised resources on the Eagle supercomputer at the Poznan Super-computing and Networking Centre (PSNC). The PSNC Eagle machine hardware characteristics can be found here: <https://wiki.man.poznan.pl/hpc/index.php?title=Eagle>. According to our observation, the execution time for each simulation run is between 40 minutes to 1 hour depends on the size of the selected area (we aim to improve the execution time with our parallel algorithm currently being developed). Interested users with basic knowledge of operating python executable scripts can easily follow the detailed step-by-step tutorial at: <https://facs.readthedocs.io/en/latest/> to install and execute FACS simulator with example models.

3.1.6. Step 6: Analysis

In this step, the modeller performs output analysis of the simulation runs. The 'Post-Processing module' allows the visualisation and validation of the results.

3.2. FACS - Architecture and Design

In this section, we present the architecture of the FACS framework and the design of its core components as shown in figure 7. FACS Core is the main component that takes inputs from the Pre-Processing Module. This module helps in the preparation of input data as shown in figure 1. The Core component can either be used to prepare an ensemble run or multiple-runs (replicas) to simulate on the supercomputer using the FabCovid-19 module (as discussed in section 3.1.5) or it can be used to simulate a single run in the local machine. The outputs from FACS Core single run or FabCovid-19 batch runs are post-processed for visualisation using Post-Processing Module.

The design of the FACS-Core module is illustrated using a class diagram shown in figure 7. The Core consists of five based classes:

- (1) *Person*: This class is used to model the structure, behaviour and the key characteristics of a Person agent.
- (2) *Household*: This class models a collection of Person agents as households.
- (3) *Location*: The location class models a typical building in an area. 'House' inherits from this class.
- (4) *Disease*: This class models the characteristics of a disease under consideration.
- (5) *Needs*: This class models the daily life needs of person agents and is used to model their mobility patterns.

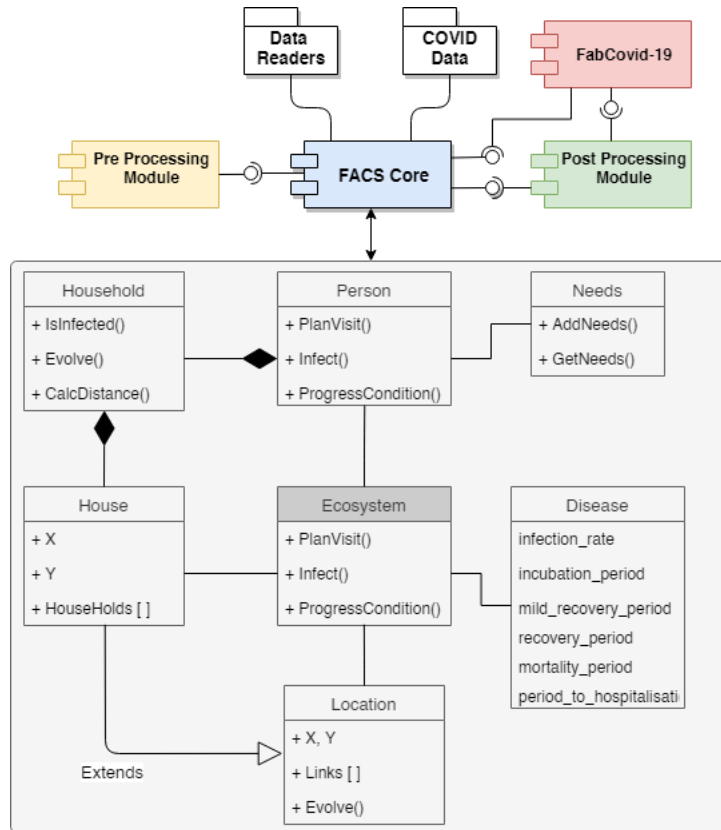


Figure 7. FACS - System Architecture

3.3. FACS - Algorithm

In this section we present the proposed ABS algorithm of disease spread under different public health measures. Algorithm 1 shows the single step of the simulation i.e., the evolution of the dynamics of the model in one day. It takes the following inputs:

- (1) *End time*: This is used to set the length of the simulation
- (2) *Transition Day*: It specifies the day when the transition of the public health measure is intended to take effect.
- (3) *Transition Scenario*: This is the keyword used to specify the public health measure to be evaluated during the simulation run. Table 2 shows a list of different scenarios available for COVID-19 simulation and their description.
- (4) *Needs*: This is a list of locations, time to visit and length of stays for people of different ages as shown in Figure 6.
- (5) *Age Distribution*: This is the data of age distribution in a population under consideration.
- (6) *Disease Parameters*: This is a list of disease parameters
- (7) *Location Graph*: This is a geospatial data of the latitude, longitude and the type of buildings. The location graph file is taken as input into the simulation for instantiating the location objects.

The output of this algorithm is a CSV with a row representing a day in the simulation and the aggregate number of people at different states (i.e., susceptible, exposed, infectious, recovered and dead).

Algorithm 1: Simulation Step

Input: EndTime:*Integer*, TransitionDay:*Integer*, TransitionScenario:*String*,
Needs:*list*, AgeDistribution:*list*, DiseaseParameters:*list*,
LocationGraph:*file*

Output: Results:*file*

```
1 Ecosystem ← InitializeEcosystem(SimulationLength = EndTime)
2 Ecosystem ← Read(AgeDistribution)
3 Ecosystem ← Read(DiseaseParameters)
4 Ecosystem ← Read(LocationGraph)
5
6 for t ← 0 to EndTime n do
7   | if t == TransitionDay then
8   |   | Ecosystem ← ApplyScenario(TransitionScenario)
9   |   end
10  | Ecosystem ← Evolve
11  | t ← t + 1
12 end
13 return Results
```

In Algorithm 1, lines 1-4 initialise the Ecosystem object of the Simulation Core, using the input data. Lines 6-11 show the loop for the length of the simulation, where each iteration evolves the ecosystem on a day basis. Algorithm 2 presents the daily evolution of the simulation. If the day reaches a *transition day*, the corresponding *transition scenario* will take into effect.

In Algorithm 2, lines 1-20 present a hierarchical loop to iterate each house in the region, for each household in the house and for each person in the household. Line 4

accumulates a list of visits planned during the day as per the agent's needs. Line 5 processes the condition of the agent based on its current state and the time elapsed, as shown in Algorithm 3. Line 6-13 show the process of agent movement and possible exposure during the visit. In Algorithm 3, the person's condition is progressed in time as explained in figure 5.

Algorithm 2: Evolve

```

1 for house  $\leftarrow$  0 to NumofHouses do
2   for household  $\leftarrow$  0 to NumofHouseholds do
3     for person  $\leftarrow$  0 to NumofOccupants do
4       Visits  $\leftarrow$  PlanVisits(Needs)
5       Person.State  $\leftarrow$  ProgressCondition(CurrentState)
6       for visit  $\leftarrow$  0 to Visits do
7         if State == Susceptible then
8            $P_{\text{Infection}} \leftarrow \text{InfectionProbability}$ 
9           if Random() < InfectionProbability then
10             $\text{Person.State} \leftarrow \text{Exposed}$ 
11          end
12        end
13        visit  $\leftarrow$  visit + 1
14      end
15      person  $\leftarrow$  person + 1
16    end
17    household  $\leftarrow$  household + 1
18  end
19  house  $\leftarrow$  house + 1
20 end

```

Algorithm 3: Progress Condition

```

1 if CurrentState == Exposed and  $t \geq \text{PhaseDuration}$  then
2    $\text{Person.State} \leftarrow \text{Infectious}$ 
3 end
4 else if CurrentState == Infectious and  $t \geq \text{PhaseDuration}$  then
5   if Infection == Mild then
6      $\text{Person.State} \leftarrow \text{Recovered}$ 
7   end
8   else
9     if random() < MortalityRate then
10       $\text{Person.State} \leftarrow \text{Dead}$ 
11    end
12    else
13       $\text{Person.State} \leftarrow \text{Recovered}$ 
14    end
15  end
16 end

```

No	Transition Scenario	Description
1	NO-MEASURES	No interventions at all.
2	EXTEND-LOCKDOWN	Extend London lockdown as of 15-05-2020 infinitely.
3	DYNAMIC-LOCKDOWN	50% of the population is sent to work (and associated facilities opened) when fewer than 100 ICU beds are occupied. This is checked once a week.
4	PERIODIC-LOCKDOWN	Switch between the 15-05-2020 lock down situation and 50% sent to work every 90 days.
5	OPEN-ALL	Lift the lockdown entirely at a pre-set date.
6	OPEN-SCHOOLS	Open all schools from a pre-set date.
7	OPEN-SHOPPING	Open all shops from a pre-set date.
8	OPEN-LEISURE	Open all leisure locations from a pre-set date.
9	WORK50	50% sent to work 40% of the shops are opened.
10	WORK75	75% sent to work schools and shops are open. 50% of the leisure locations are open.
11	WORK100	All measures lifted except for social distancing isolation of infected cases and household quarantine.

Table 2.: List of example transition scenarios supported by FACS

4. COVID-19 Case Study

The aim of the COVID-19 case study is to demonstrate the functionality of the FACS framework and give estimates to the research questions posed earlier in this paper. In this case study we simulate the spread of COVID-19 across three boroughs of London namely: (i) London Borough of Brent; (ii) London Borough of Ealing; and (iii) London Borough of Harrow. We use four variants of the lock down scenarios: (a) Extended Lock down; (b) Dynamic lock down; (c) Periodic lock down and (d) No measures, taking into account the London lock down as it has been imposed in March 2020 (except for the no-measures run). These scenarios are described in table 2. As our understanding of what the most realistic scenarios could be is evolving rapidly, these scenarios are selected primarily for illustrative purposes.

We extract geospatial data for each borough from the Open Street Map and constructed a location graph (LG). Each LG contains hospitals, houses, offices, schools, supermarkets, leisure locations and parks. We use COVID-19 disease model as discussed earlier in section 4.1.3. We gather different type of demographic data for each borough e.g., we use the age distributions as shown in figure 8.

After the pre-processing steps, we execute the simulation using FabCOVID-19 module. We simulate an ensemble of 25 replicas for each four scenario for each borough. The results are intended to be illustrative, therefore the ensemble size was limited to 25 in part to constrain the computational footprint. The simulation has a warm-up period of 30 days (not included in the plots). This batch of simulation runs are executed on a remote supercomputer. After the execution we use the Post-processing module for visualization and model validation.

4.1. Simulation Results

The simulation results for the COVID-19 case study are discussed in this section as shown in figure 9 - 10. The plots illustrate the number of susceptible, exposed, infectious, recovered and dead in each borough under different lock down scenarios. Here, days are plotted on the x axis: day 1 is the 1st of March, and the length of the simulation is 180 days for 'No measures' and 'Extend lock down' scenarios and 730 days for 'Dynamic' and 'Periodic' lock down scenarios. The y axis indicates the magnitude of population under each SEIRD state. The solid line shows the mean whereas the shaded area shows the stochasticity obtained from the 25 replica runs. The shaded area spans the minimum and maximum values of the 25 runs. We also calculated the 95% Confidence Interval of these run that lies within this shaded area.

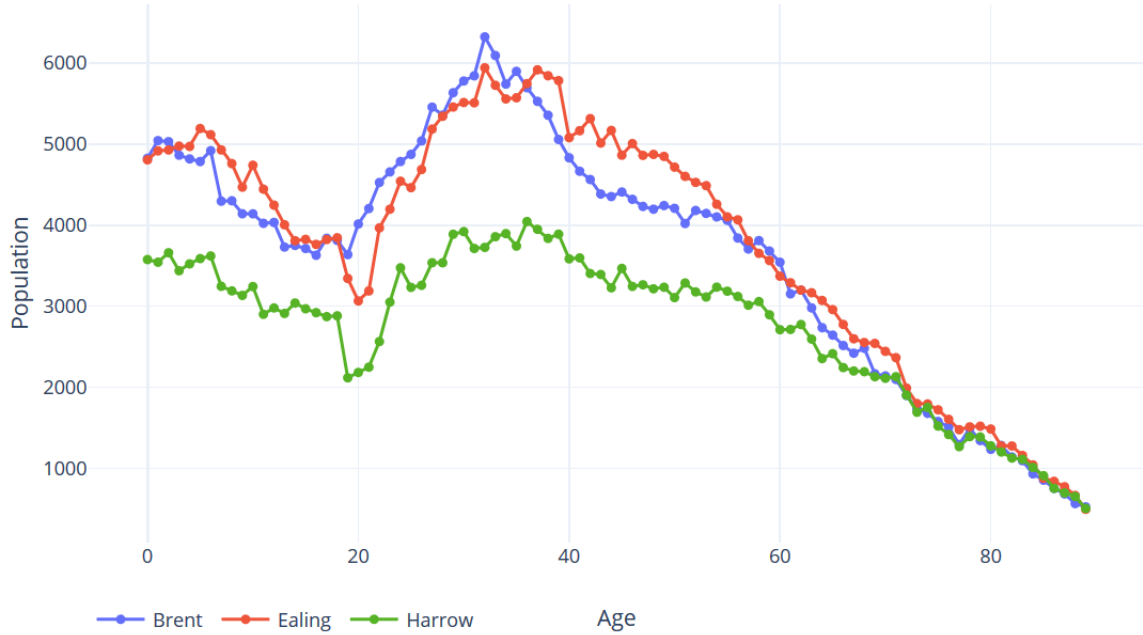


Figure 8. Age Distributions in different boroughs of London

For the London borough of Brent (with a population size of 330,795), no-measures scenario peaks at 12,000 exposed and 50,000 infected cases (and about 300k recovered cases) on day=75. After this peak the majority of the population has already had Covid-19, so the rate of new infections rapidly drops. Extend-lock down exhibits a rise of 700 exposed and 6000 infected cases on day=23, before it declines and continues to rise gradually (figure 9). Dynamic lock down shows a relatively lower peak at day=23 but it rises up to 300 exposed and 3000 infected cases before reaching a steady state (figure 10). Periodic lock down initially resembles dynamic lock down but later shows a more volatile harmonic trend due to its very nature (figure 10). The susceptible, recovered and dead states are directly dependent on both exposed and infected states but inversely proportional.

The London borough of Ealing (with a population size of 341,982) shows a high spread of infection as compared to other boroughs. In the case of no-measures scenario, it peaks at 15000 exposed and 60,000 infected cases on day=60 (figure 9). Extend-lock down exhibits a rise of 900 exposed and 8000 infected cases on day=23, and continues to rise gradually up to 5000 exposed and 7000 infected cases, before a gradual descent (figure 9). Dynamic lock down is similar to extended lock down (figure 10). But in the case of Periodic lock down there is an initial peak of 1000 exposed and 8000 infected cases before a rise of the relatively lesser volatile harmonic trend (figure 10). The susceptible, recovered and dead states are directly dependent on both exposed and infected states but inversely proportional.

For the London borough of Harrow (with a population size of 250,149), no-measures scenario peaks at 10,000 exposed and 40,000 infected cases on day=60 (figure 9). Extend-lock down exhibits a rise of 700 exposed and 6000 infected cases on day=23, and continues to rise gradually up to 4000 exposed and 3000 infected cases, before a gradual descent (figure 9). Dynamic lock down shows a peak of 300 exposed and 3300 infected cases at day=23 (figure 10). In the case of Periodic lock down there is an initial peak of 500 exposed and 5000 infected cases before a rise of the relatively

lesser volatile harmonic trend (figure 10). The susceptible, recovered and dead states are directly dependent on both exposed and infected states but inversely proportional.

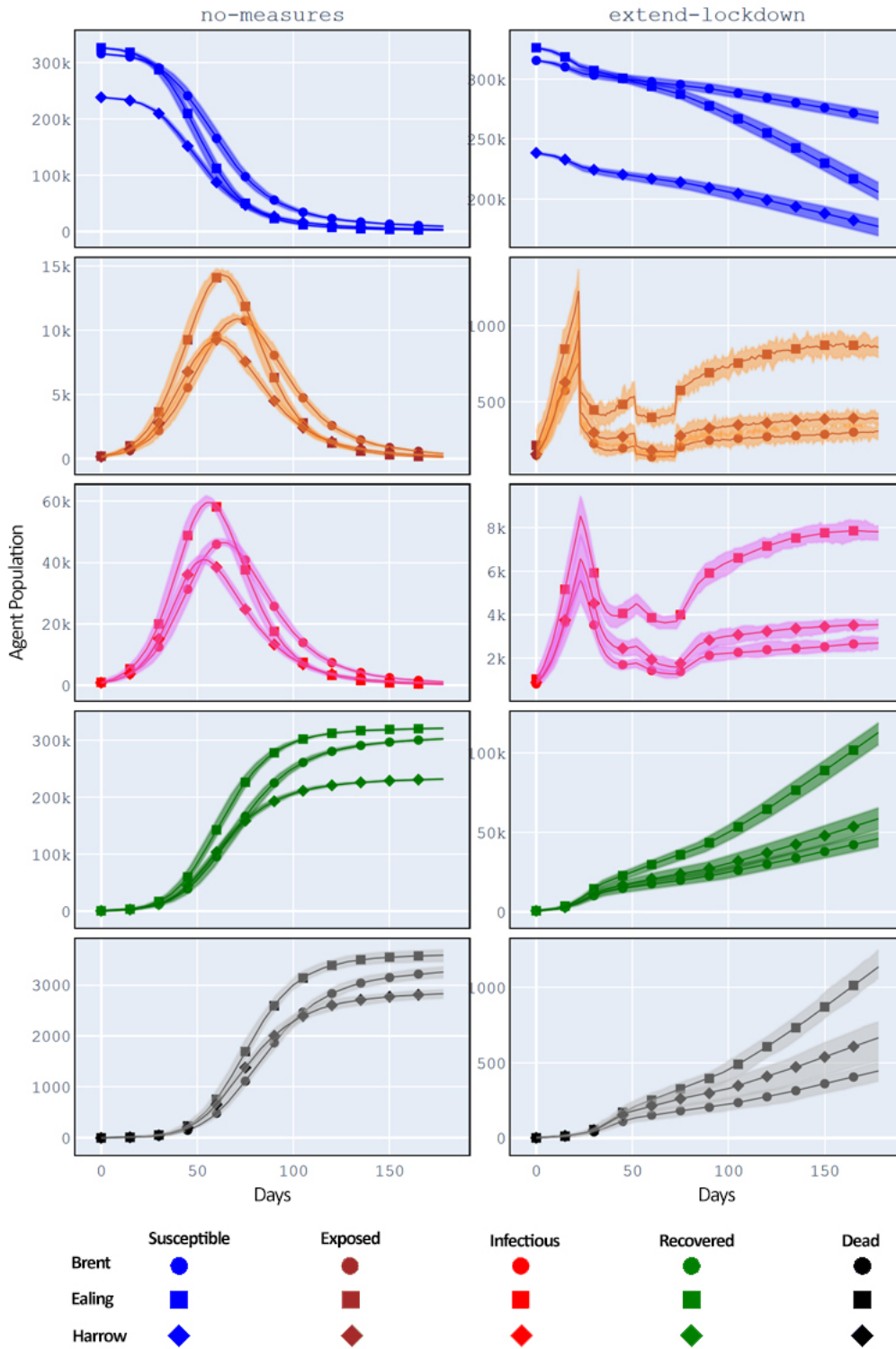


Figure 9. Simulation Results (a) No measures (b) Extended Lock down

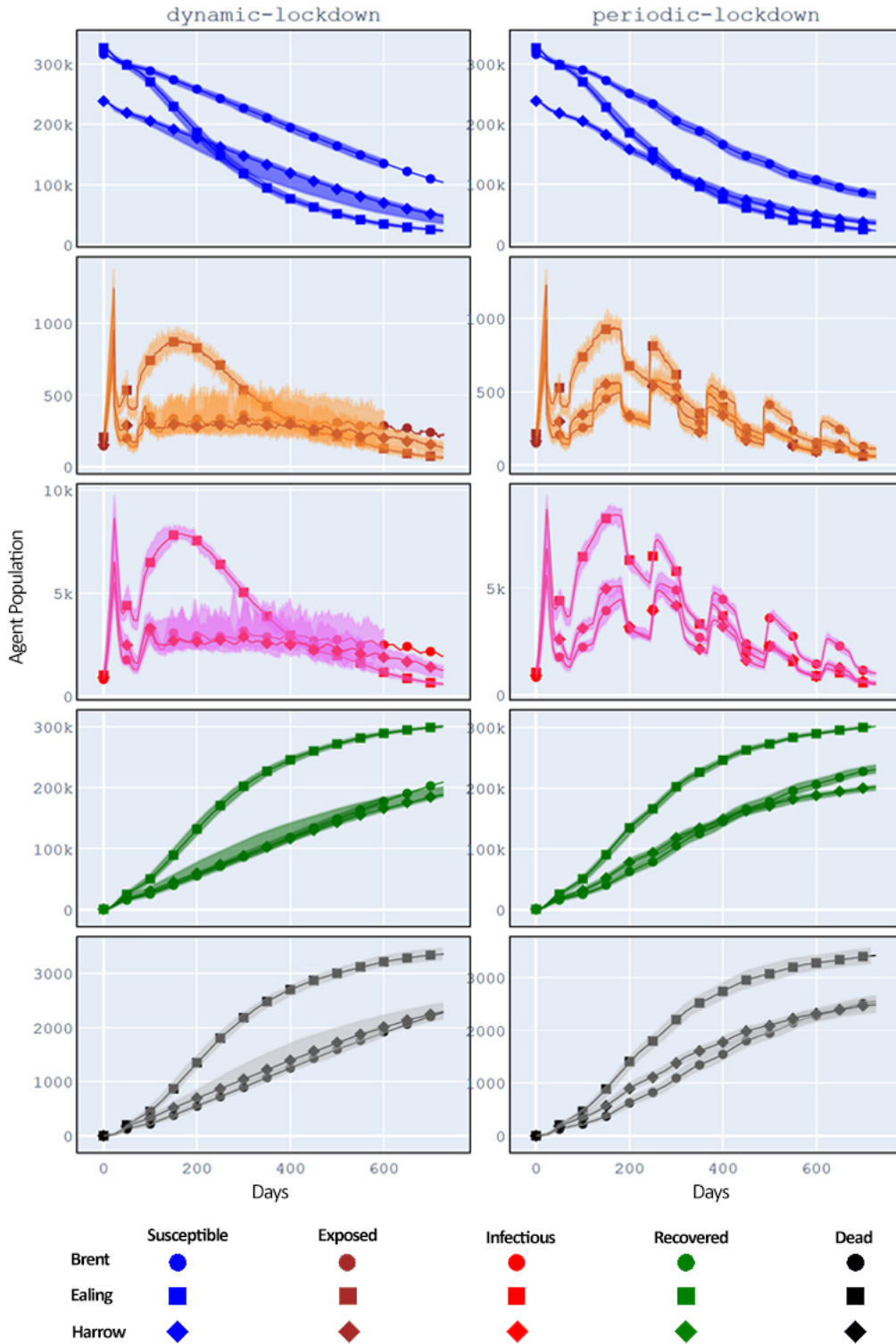


Figure 10. Simulation Results (a) Dynamic Lock down (b) Periodic Lock down

The simulation results of the the above scenarios in different borough indicate that the spread of the disease is dependent on the factors such as: (a) Location Graphs; (b) population demographics; (c) mobility patterns; (d) social distancing; (e) rate of

contact; and (f) COVID-19 infectivity.

Figure 11 shows a ward level spatial distribution of the infected cases (simulated) on a single day (day=20) for the three boroughs using a choropleth chart. It is evident that Ealing as well some adjoining parts of Harrow have greater accumulation of cases, as compared to the borough of Brent. This visualization is helpful to analyze the concentration of the infection and its spread and can be used as a tool to identify and narrow down the hot spots and their movements and direction over time. Based on the availability of spatial data, this chart can further be refined into more fine-grained administrative divisions, thus aid in the more accurate spatial analysis of the disease spread. .

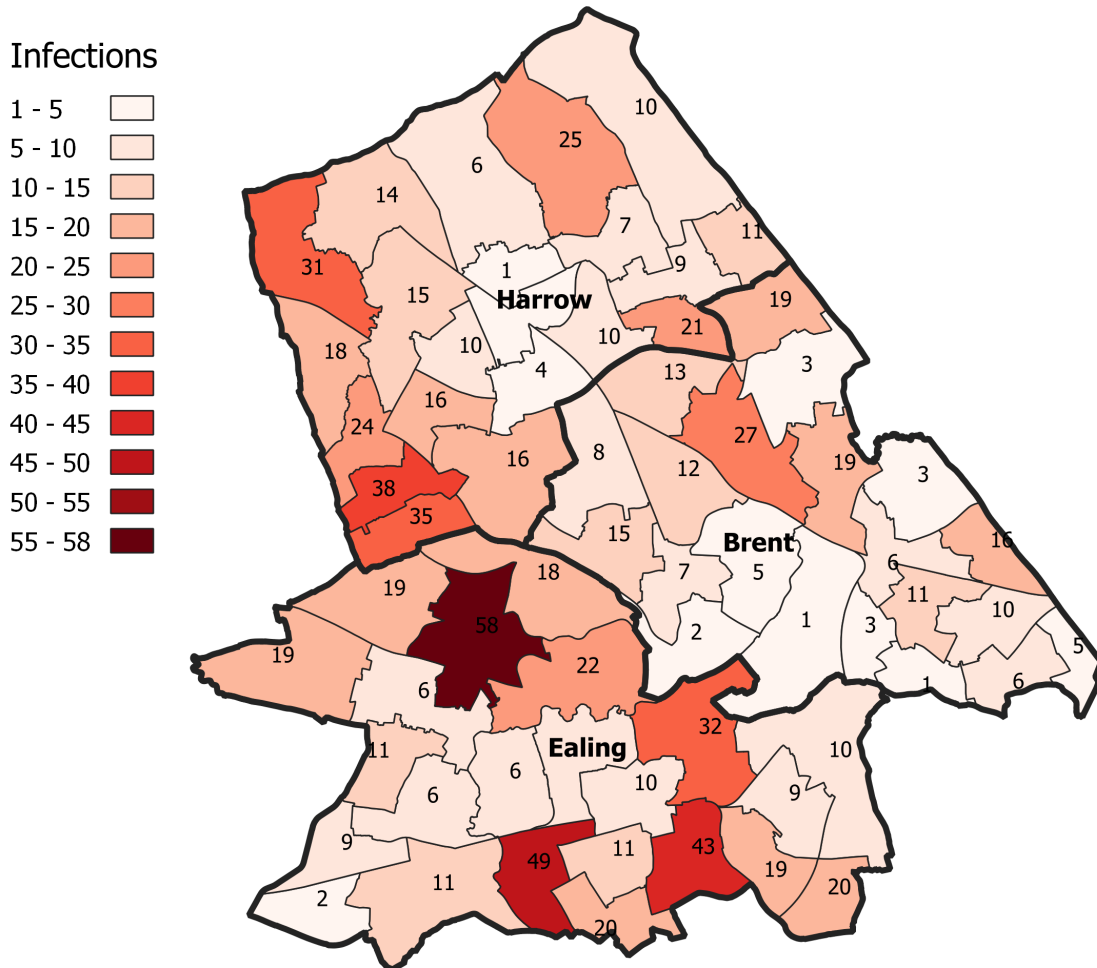


Figure 11. Choropleth Map of the simulated infections (day=20)

The simulation results of the case study show that the current lock down measures are not strict enough to eradicate the disease altogether. Note however, that the model assumes that 100% of the people are susceptible to the disease at the start of the simulation. Users may change this initial condition if there is evidence/data to support. FACS simulator is the product of applied research and our goal is to build software tools to help epidemiologists, public health practitioners and government officials in the exploration/analysis of disease such as COVID-19. While we attempt to use COVID-19 case study as a concept of proof of the functionality of simulating the disease

dynamics, we are making continuous improvements through the ongoing collaboration with epidemiologists and public healthcare professionals to fine tune its functionalities and provide descriptive and perspective insights of the system.

4.2. Model Validation

We conducted the model validation exercise using anonymized data from London North West University Healthcare NHS Trust. The validation results are shown in figure 12. Here, results of the London borough of Brent are plotted for a period of 80 days on the x axis (day 1 = 01-March). The y axis indicates the magnitude. The top plot contains the number of new infections per day as a reference. The second plot contains a comparison of the number of actual and simulated hospital admissions. The simulated hospital admissions are obtained from the simulation runs, where the infected agents lead to severe illness. The third plot contains a comparison of actual and simulated ICU admissions. Although the agreement of the magnitude on y-axis is aligned, however the simulated hospitalisation (in blue) is overestimated than the actual observations. This may be resolved when we perform co-simulation/model coupling of multiple boroughs and handle boundary conditions in future.

For verification, we consulted NHS officials, epidemiologists and healthcare experts for: (a) input data validity; (b) parameters and assumption correctness and (c) behavioural logic of the model. We intend to perform a more formal verification, validation and sensitivity analysis in the future. Moreover, the urgency of constantly

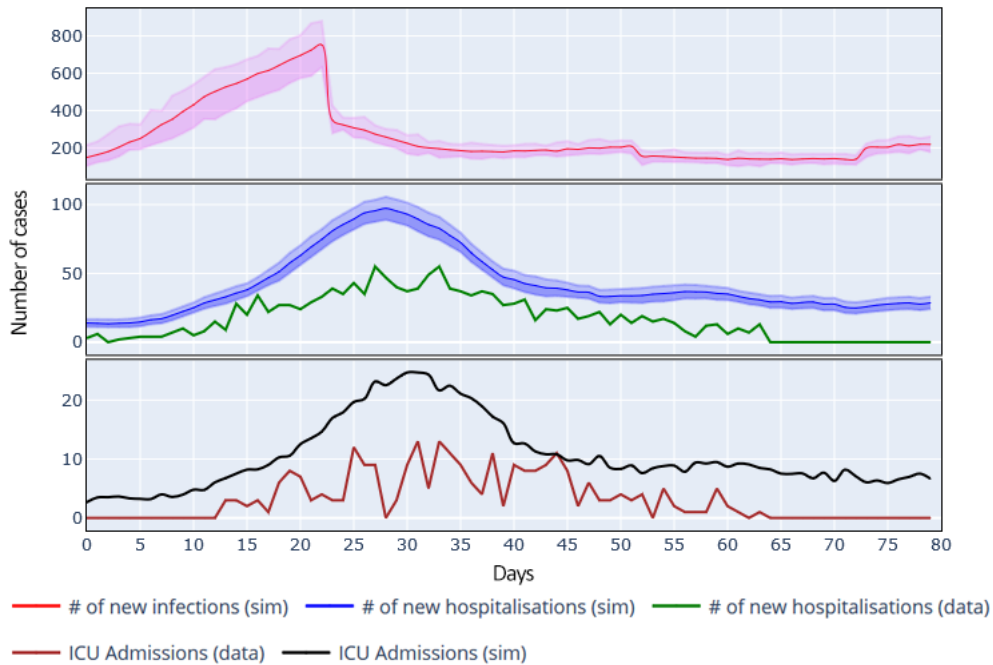


Figure 12. Model Validation (a) Infections in Brent borough - Simulated (b) Hospitalizations - Actual vs Simulated (c) ICU Admissions - Actual vs Simulated

refining and revising COVID-19 related simulation and forecasting models to leverage the ever-growing body of scientific research in this area necessitates the need to automatically shift through the relevant literature (corpus of knowledge) effectively. With such focus in mind, we aim to employ Natural Language Processing (NLP) techniques to develop and implement an approach that facilitates the extraction of and reasoning with assumptions from a large and growing corpus of relevant scientific publications in support of simulation models.

5. Summary and Conclusion

In this paper, we presented the *Flu And Coronavirus Simulator (FACS)*: an ABS tool that models the viral spread of a specific infectious disease on the *sub-national* level. We described the proposed simulation development approach as *modus operandi* for the researchers to adapt and use FACS in their local context. We demonstrated the functionality of our framework using COVID-19 simulation on the sub-national region and show the validation of our simulation results. Based on the results, the use of agent-based approach has proven to be quite productive in modeling complex systems like epidemic spread in a large region due to ever changing model requirements, multi-resolution abstraction, non-linear system dynamics, rule-based heuristics and above all large-scale computing requirements. During the development of this framework, we learned that the real-world abstraction changes more rapidly than in other circumstances. For instance, the concept of social distancing and lock down scenarios have evolved significantly since early March. Therefore, rapid changes in the ABS model were necessary. Model building in these cases benefits more using a bottom-up approach like ABS, as opposed to any centralized analytical solution.

A validated agent-based epidemic simulation helps in the accurate estimations of short-term and long-term forecasts of a specific infectious disease, answers different 'what-if' research questions and eventually helps in effective decision making. Our ABS-based simulation framework estimates the spread of infectious disease and identifies key hot spots within the given region. It also estimates the arrivals of infected cases from various parts of the region and predicts the admissions to hospitals, under different lock down scenarios. Result from all scenarios were shared with health service planning and transformation directors and used in their formulation of COVID response strategies.

This work is in progress and new features are being added in the framework to incorporate emerging needs of the epidemic researchers and to aid in effective decision support. We intend to improve the mobility of agents using the routes defined between houses and different location types using actual streets or roads imported from the open street map data. This will be used to resolve the street-level infection risks. Moreover, the urgency of constantly refining and revising COVID-19 related simulation and forecasting models to leverage the ever-growing body of scientific research in this area necessitates the need to automatically shift through the relevant literature (corpus of knowledge) effectively. With such focus in mind, we aim to employ Natural Language Processing (NLP) techniques to develop and implement an approach that facilitates the extraction of and reasoning with assumptions from a large and growing corpus of relevant scientific publications in support of simulation models.

Acknowledgement(s)

We are grateful to the London North West University Healthcare NHS Trust for providing us with validation data, as well as valuable feedback during the development and validation stages of this project. The calculations were performed in Poznan Supercomputing and Networking Center.

Disclosure statement

The simulation results presented in this paper are for educational and research purpose only.

Funding

This work was supported by the HiDALGO and VECMA projects, which has received funding from the European Union Horizon 2020 research and innovation programme under grant agreement No 824115 and 800925.

References

- Anastassopoulou, C., Russo, L., Tsakris, A., & Siettos, C. (2020). Data-based analysis, modelling and forecasting of the covid-19 outbreak. *PloS one*, *15*(3), e0230405.
- Arabnejad, H., Groen, D., & Mahmood, I. (2020). *FabCovid19: a fabsim3 plugin for flu and coronavirus simulator (flacs)*. <https://github.com/djgroen/FabCovid19>. GitHub.
- Bagni, R., Berchi, R., & Cariello, P. (2002). A comparison of simulation models applied to epidemics. *Journal of Artificial Societies and Social Simulation*, *5*(3).
- Balcan, D., Gonçalves, B., Hu, H., Ramasco, J. J., Colizza, V., & Vespignani, A. (2010). Modeling the spatial spread of infectious diseases: The global epidemic and mobility computational model. *Journal of Computational Science*, *1*(3), 132 - 145. Retrieved from <http://www.sciencedirect.com/science/article/pii/S1877750310000438> doi: <http://www.sciencedirect.com/science/article/pii/S1877750310000438>
- Bar-Yam, Y. (2002). General features of complex systems. *Encyclopedia of Life Support Systems (EOLSS)*, UNESCO, EOLSS Publishers, Oxford, UK, 1.
- Bi, Q., Wu, Y., Mei, S., Ye, C., Zou, X., Zhang, Z., . . . Zhang, T. (2020). Epidemiology and transmission of covid-19 in 391 cases and 1286 of their close contacts in shenzhen, china: a retrospective cohort study. *The Lancet Infectious Diseases*.
- Bonabeau, E. (2002). Agent-based modeling: Methods and techniques for simulating human systems. *Proceedings of the national academy of sciences*, *99*(suppl 3), 7280–7287.
- Bordehore, C., Navarro, M., Herrador, Z., & Fonfria, E. S. (2020). Understanding covid-19 spreading through simulation modeling and scenarios comparison: preliminary results. *medRxiv*.
- Candido, P. (2020). *Agent based simulation of covid-19 health and economical effects*. <https://github.com/petroniocandido>. GitHub.
- Chan, M., yeung, & Xu, R.-H. (2003). Sars: epidemiology. *Respirology*, *8*(s1), S9-S14. Retrieved from <https://onlinelibrary.wiley.com/doi/abs/10.1046/j.1440-1843.2003.00518.x> doi: <https://onlinelibrary.wiley.com/doi/abs/10.1046/j.1440-1843.2003.00518.x>
- Chang, S. L., Harding, N., Zachreson, C., Cliff, O. M., & Prokopenko, M. (2020). Modelling transmission and control of the covid-19 pandemic in australia. *arXiv preprint arXiv:2003.10218*.

- Chen, T.-M., Rui, J., Wang, Q.-P., Zhao, Z.-Y., Cui, J.-A., & Yin, L. (2020). A mathematical model for simulating the phase-based transmissibility of a novel coronavirus. *Infectious diseases of poverty*, 9(1), 1–8.
- Chinazzi, M., Davis, J. T., Ajelli, M., Gioannini, C., Litvinova, M., Merler, S., ... Vespignani, A. (2020). The effect of travel restrictions on the spread of the 2019 novel coronavirus (covid-19) outbreak. *Science*, 368(6489), 395–400. Retrieved from <https://science.sciencemag.org/content/368/6489/395> doi:
- CSSE, J. (2020, Feb). *Covid-19 dashboard by the center for systems science and engineering (csse) at johns hopkins university*. JHU. Retrieved from <https://gisanddata.maps.arcgis.com/apps/opsdashboard/index.html>
- Dandekar, R., & Barbastathis, G. (2020). Quantifying the effect of quarantine control in covid-19 infectious spread using machine learning. *medRxiv*. Retrieved from <https://www.medrxiv.org/content/early/2020/04/06/2020.04.03.20052084> doi:
- ECDC. (2020, June). *Covid-19 situation update worldwide, as of 7 june 2020*. Author. Retrieved from <https://www.ecdc.europa.eu/en/geographical-distribution-2019-ncov-cases>
- Ferguson, N., Laydon, D., Nedjati Gilani, G., Imai, N., Ainslie, K., Baguelin, M., ... Cuomo-Dannenburg, G. (2020). Report 9: Impact of non-pharmaceutical interventions (npis) to reduce covid19 mortality and healthcare demand.
- Forrester, J. W. (1994). System dynamics, systems thinking, and soft or. *System dynamics review*, 10(2-3), 245–256.
- Groen, D., & Arabnejad, H. (2014). *Fabsim3*. <https://github.com/djgroen/FabSim3>. GitHub.
- Groen, D., & Arabnejad, H. (2015). *Flee*. <https://github.com/djgroen/flee>. GitHub.
- Groen, D., Bhati, A. P., Suter, J., Hetherington, J., Zasada, S. J., & Coveney, P. V. (2016). Fabsim: Facilitating computational research through automation on large-scale and distributed e-infrastructures. *Computer Physics Communications*, 207(Supplement C), 375 - 385. Retrieved from <http://www.sciencedirect.com/science/article/pii/S0010465516301448> doi:
- Groen, D., Mahmood, I., & Arabnejad, H. (2020). *Flacs: Flu and coronavirus simulator*. <https://github.com/djgroen/flacs>. GitHub.
- Hellewell, J., Abbott, S., Gimma, A., Bosse, N. I., Jarvis, C. I., Russell, T. W., ... Eggo, R. M. (2020). Feasibility of controlling covid-19 outbreaks by isolation of cases and contacts. *The Lancet Global Health*, 8(4), e488 - e496. Retrieved from <http://www.sciencedirect.com/science/article/pii/S2214109X20300747> doi:
- Huang, C., Wang, Y., Li, X., Ren, L., Zhao, J., Hu, Y., ... Cao, B. (2020). Clinical features of patients infected with 2019 novel coronavirus in wuhan, china. *The Lancet*, 395(10223), 497 - 506. Retrieved from <http://www.sciencedirect.com/science/article/pii/S0140673620301835> doi:
- Jacobson, S. H., Hall, S. N., & Swisher, J. R. (2006). Discrete-event simulation of health care systems. In *Patient flow: Reducing delay in healthcare delivery* (pp. 211–252). Springer.
- Klepac, P., Kucharski, A. J., Conlan, A. J., Kissler, S., Tang, M., Fry, H., & Gog, J. R. (2020). Contacts in context: large-scale setting-specific social mixing matrices from the bbc pandemic project. *medRxiv*. Retrieved from <https://www.medrxiv.org/content/early/2020/03/05/2020.02.16.20023754> doi:
- Kucharski, A. J., Russell, T. W., Diamond, C., Liu, Y., Edmunds, J., Funk, S., ... Munday, J. D. (2020). Early dynamics of transmission and control of covid-19: a mathematical modelling study. *The lancet infectious diseases*.
- Li, M., Yu, W., Tian, W., Ge, Y., Liu, Y., Ding, T., & Zhang, L. (2019). system dynamics modeling of public health services provided by china cdc to control infectious and endemic diseases in china. *Infection and drug resistance*, 12, 613.
- Li, Q., Guan, X., Wu, P., Wang, X., Zhou, L., Tong, Y., ... Wong, J. Y. (2020). Early transmission dynamics in wuhan, china, of novel coronavirus-infected pneumonia. *New England Journal of Medicine*.

- Li, W., Zhang, B., Lu, J., Liu, S., Chang, Z., Peng, C., ... Chen, J. (2020, 04). Characteristics of Household Transmission of COVID-19. *Clinical Infectious Diseases*. Retrieved from <https://doi.org/10.1093/cid/ciaa450> (ciaa450) doi:
- Liu, T., Hu, J., Kang, M., Lin, L., Zhong, H., Xiao, J., ... Rong, Z. (2020). Transmission dynamics of 2019 novel coronavirus (2019-ncov).
- Lorch, L., Trouleau, W., Tsirtsis, S., Szanto, A., Schölkopf, B., & Gomez-Rodriguez, M. (2020). A spatiotemporal epidemic model to quantify the effects of contact tracing, testing, and containment. *arXiv preprint arXiv:2004.07641*.
- Macal, C. M., & North, M. J. (2010). Tutorial on agent-based modeling and simulation. *Journal of Simulation*, 4(3), 151-162.
- Mahmood, I., Anagnostou, A., Taylor, S., Bell, D., & Groen, D. (2020, Jun). *Facs framework stress report*. Brunel University London. Retrieved from https://brunel.figshare.com/articles/online_resource/FACS_Framework_STRESS_Report_pdf/12520763/1
- Mahmood, I., Jahan, M., Groen, D., Javed, A., & Shafait, F. (2020). An agent-based simulation of the spread of dengue fever. In *International conference on computational science*.
- Miksch, F., Urach, C., Einzinger, P., & Zauner, G. (2014). A flexible agent-based framework for infectious disease modeling. In Linawati, M. S. Mahendra, E. J. Neuhold, A. M. Tjoa, & I. You (Eds.), *Information and communication technology* (pp. 36–45). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Miller, J. H., & Page, S. E. (2009). *Complex adaptive systems: An introduction to computational models of social life*. Princeton university press.
- Mizumoto, K., & Chowell, G. (2020). Transmission potential of the novel coronavirus (covid-19) onboard the diamond princess cruises ship, 2020. *Infectious Disease Modelling*, 5, 264 - 270. Retrieved from <http://www.sciencedirect.com/science/article/pii/S2468042720300063> doi:
- Moghadas, S. M., Shoukat, A., Espindola, A. L., Pereira, R. S., Abdirizak, F., Laskowski, M., ... Chowell, G. (2017). Asymptomatic transmission and the dynamics of zika infection. *Scientific reports*, 7(1), 1–8.
- Monks, T., Currie, C. S. M., Onggo, B. S., Robinson, S., Kunc, M., & Taylor, S. J. E. (2019). Strengthening the reporting of empirical simulation studies: Introducing the stress guidelines. *Journal of Simulation*, 13(1), 55-67.
- Murray, C. J. (2020). Forecasting the impact of the first wave of the covid-19 pandemic on hospital demand and deaths for the usa and european economic area countries. *medRxiv*.
- NCID. (2020). *Position statement from the national centre for infectious diseases and the chapter of infectious disease physicians, academy of medicine, singapore*. <https://www.ncid.sg/Documents/Period%20of%20Infectivity%20Position%20Statementv2.pdf>.
- O'Neill, P. D., Balding, D. J., Becker, N. G., Eerola, M., & Mollison, D. (2000). Analyses of infectious disease data from household outbreaks by markov chain monte carlo methods. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 49(4), 517–542.
- Peng, L., Yang, W., Zhang, D., Zhuge, C., & Hong, L. (2020). Epidemic analysis of covid-19 in china by dynamical modeling. *arXiv preprint arXiv:2002.06563*.
- Perumalla, K. (2020). *Exacorona*. <https://github.com/perumallaks/ExaCorona>. GitHub.
- Promptchara, E., Ketloy, C., & Palaga, T. (2020). Immune responses in covid-19 and potential vaccines: Lessons learned from sars and mers epidemic. *Asian Pac J Allergy Immunol*, 38(1), 1–9.
- Railsback, S. F., & Grimm, V. (2019). *Agent-based and individual-based modeling: a practical introduction*. Princeton university press.
- Suleimenova, D., Bell, D., & Groen, D. (2017). A generalized simulation development approach for predicting refugee destinations. *Scientific reports*, 7(1), 1–13.
- Taylor, S. (2014). *Agent-based modeling and simulation*. Springer.
- Tuomisto, J. T., Yrjölä, J., Kolehmainen, M., Bonsdorff, J., Pekkanen, J., & Tikkanen, T. (2020). An agent-based epidemic model reina for covid-19 to identify destructive policies. *medRxiv*. Retrieved from <https://www.medrxiv.org/content/early/2020/04/17/2020.04.09.20047498> doi:

- Venkatramanan, S., Lewis, B., Chen, J., Higdon, D., Vullikanti, A., & Marathe, M. (2018). Using data-driven agent-based models for forecasting emerging infectious diseases. *Epidemics*, *22*, 43–49.
- Verity, R., Okell, L. C., Dorigatti, I., Winskill, P., Whittaker, C., Imai, N., . . . Fu, H. (2020). Estimates of the severity of coronavirus disease 2019: a model-based analysis. *The Lancet Infectious Diseases*.
- Volz, E., & Meyers, L. A. (2007). Susceptible–infected–recovered epidemics in dynamic contact networks. *Proceedings of the Royal Society B: Biological Sciences*, *274*(1628), 2925–2934.
- Vynnycky, E., & White, R. (2010). *An introduction to infectious disease modelling*. OUP oxford.
- WHO. (2020, Feb). *Rolling updates on coronavirus disease (covid-19)*. World Health Organization. Retrieved from <https://www.who.int/emergencies/diseases/novel-coronavirus-2019/events-as-they-happen>
- Wynants, L., Van Calster, B., Collins, G. S., Riley, R. D., Heinze, G., Schuit, E., . . . van Smeden, M. (2020). Prediction models for diagnosis and prognosis of covid-19: systematic review and critical appraisal. *BMJ*, *369*. Retrieved from <https://www.bmj.com/content/369/bmj.m1328> doi:
- Zhou, F., Yu, T., Du, R., Fan, G., Liu, Y., Liu, Z., . . . Gu, X. (2020). Clinical course and risk factors for mortality of adult inpatients with covid-19 in wuhan, china: a retrospective cohort study. *The lancet*.