

A Membrane System Simulator for Covid-19 Infection Diffusion

Bachelor Thesis in Bioinformatics

Francesco Reiff

Supervised by:
Prof. Giuditta Franco

Department of Informatics
University of Verona

Contents

1	Introduction	3
1.1	Research Objectives	3
1.2	Thesis Structure	4
2	State of the Art	5
2.1	Membrane Computing	5
2.2	Epidemiological Modeling	6
2.3	LOIMOS	6
2.3.1	Membrane Hierarchy	6
2.3.2	Internal Objects	7
2.3.3	Infection Dynamics	7
2.3.4	Key Features	8
2.4	MVT	8
2.4.1	Membrane Hierarchy	9
2.4.2	Internal Objects	9
2.4.3	Infection Dynamics	9
2.4.4	Key features	10
3	Definition of the Model	11
3.1	Conflict Resolution	11
3.2	System of Membranes	13
3.3	System Architecture: Objects and Rules	15
3.4	Behavioural Model	17
3.4.1	The Prudence Parameter	17
3.4.2	The Caution Factor	18
3.4.3	Vaccination Willingness	19
4	Simulator	21
4.1	Workflow and Daily Routine	21
4.2	Graphical User Interface	22
4.3	Simulation Output Overview	26

4.3.1	Data Files and Automated Visualizations	26
4.3.2	Interactive Graphical User Interface	27
5	Results	29
5.1	Parameter Validation	29
5.1.1	The Prudence Parameter (PP)	29
5.1.2	The Caution Factor (CF)	31
5.1.3	Vaccination Willingness (F^*)	31
5.2	Dataset and Simulation Setup	33
5.2.1	Dataset Description	33
5.2.2	Simulation Setup	34
5.2.3	Prevalence and Mortality Trajectories	34
6	Conclusions	37
6.1	Future Work	40

Chapter 1

Introduction

This thesis presents a computational model for simulating disease spread, specifically focusing on a COVID-like viral dynamics. The work has been developed with a research collaboration among the University of Verona, the University of Milano Bicocca, the University of Valencia, the University of Trieste. While the formal definition and theoretical groundwork of the epidemiological model were primarily advanced in Sandro Erba's master thesis, in this manuscript the software development and testing is described, along with further refinements and validations of an extended model. By integrating and enhancing components from existing membrane computing frameworks (LOIMOS and MVT), a robust and adaptable simulator is here developed, that accurately captures complex epidemiological dynamics beside the intricate interaction between human behavior and disease progression.

1.1 Research Objectives

The thesis aims at integrating the LOIMOS and MVT membrane computing frameworks into a cohesive, unified simulator capable of modeling disease spread. This work requires identifying and resolving the differences between the two models, which range from the programming language to the identification of the membrane structure, infection and death probability, and behavioral components. Another critical aspect is the thorough validation of the resulting model using real-world data, ensuring its precision and reliability in simulating epidemiological scenarios. The ultimate goal is to build a robust and versatile simulation platform that merges LOIMOS's detailed biological interactions with MVT's behavioral flexibility, making it more realistic and adaptable across a wide range of epidemiological contexts.

1.2 Thesis Structure

The present thesis is structured into six chapters. Chapter 2 offers an in-depth review of the state of the art in membrane computing and epidemiological modeling, concluding with a summary of the foundational LOIMOS and MVT models from the recent literature. Chapter 3 defines a new unified model, detailing its membranes, objects, rules, and behavioral components, and outlines the strategies adopted to resolve conflicts that emerged during the integration of LOIMOS and MVT. Chapter 4 describes the development of the simulation tool, delving into its workflow, daily routines, and graphical user interface. Chapter 5 presents the model's validation, including parameter validation and its comparison with real-world data from the Veneto region. The concluding chapter summarizes the obtained results, draws conclusions, and outlines possible directions for future research. The last three chapters specifically report the original contribution of my work.

Chapter 2

State of the Art

2.1 Membrane Computing

Membrane computing, proposed by Gheorghe Păun in 1998, is a natural computing paradigm inspired by the compartmental structure and the function of living cells. It defines computational models called *P systems*, characterized by hierarchically ordered membranes that contain multisets of objects and evolution rules. These models operate in a highly parallel and nondeterministic fashion, differently from traditional sequential models such as Turing machines. [1]

Membrane computing models computation using a structure of nested membranes, like those in a living cell, where objects evolve via rules in parallel and can move between compartments. These rules mimic biological processes such as chemical reactions and trans-membrane transport. Computations occur as objects change or move. Those models balance simplicity, biological realism and computational power. Variants also exist, inspired by tissues, neural systems, and bacterial populations. [2]

Membrane computing began as a theoretical model focused on computation inspired by biology, not on directly modeling real biological processes. However, it later proved useful in biology and medicine for simulating cellular processes. Compared to differential equations, it offers advantages like modularity, scalability, transparency, and programmability. Besides applications in biology, membrane computing was considered also in other areas, such as computer graphics, cryptography, modelling, linguistics and economics.

2.2 Epidemiological Modeling

Epidemiological models are crucial to understand, predict, and control the spread of infectious diseases. Classic models such as SIR (Susceptible-Infectious-Recovered) and SEIR (Susceptible-Exposed-Infectious-Recovered) rely on systems of differential equations to track population-level transitions. While useful for macro-scale predictions, these models may not capture individual heterogeneity or spatial and social structures effectively. [3]

To address these gaps, membrane computing proves highly beneficial in this field by enabling the compartmentalization of the population and incorporating rules that reflect social norms and behaviors.

Membrane computing therefore offers a promising alternative for epidemiological modeling by integrating compartmental logic with parallel computation. In this context, individuals or population groups are modeled as objects or membranes, while rules simulate infection, recovery, movement, interaction processes and social behaviours.

P systems have proven particularly effective in modeling population clusters [4]. In these contexts, each membrane can represent a subpopulation, and communication rules allow for the modeling of disease transmission between them. Such models support parallel processing and can be scaled or adapted to simulate a variety of disease scenarios.

Real-world applications include simulations of COVID disease, exploring the effects of vaccination, travel restrictions, and behavioral changes [4].

2.3 LOIMOS

LOIMOS model [4] [5] [6] is a membrane computing (P system) framework that represents an epidemic as hierarchically nested membranes, enabling simultaneous simulation of spatial, temporal, and within-host dynamics. By combining active-membrane rules with stochastic transitions, LOIMOS captures individual movements, infection processes, and intervention effects in a unified formalism. A list of main features of LOIMOS follows:

2.3.1 Membrane Hierarchy

- **Skin (Echo):** global container.
- **Place membranes:** parallel compartments inside Echo representing Home (4 000 families), Primary and Secondary schools, Workplace, Common area, Leisure area, Residence, Elderly day-centre, Hospital, ICU, Hospital_aux and Post-ICU.

- **Host membranes (denoted by h_i):** each individual is a membrane containing objects for age group, role, clock, and infection state; hosts move between place membranes via stochastic movement rules.

2.3.2 Internal Objects

Each host membrane contains:

- *Clocks:* Hour_{0-23} and $\text{Day}_{\text{Mon-Sun}}$ to drive time-dependent movements.
- *Demographics:* age tags **ss1** (0–12), **ss1b** (13–19), **ss2** (20–59), **ss3** (60+); role tags (**worker**, **student**, **resident**).
- *Infection markers:* viral particles (v_k), immunity markers (**antiv**, **antivesp**), and course labels (asymptomatic, mild, severe, critical). See section 2.2.4 for more details.

2.3.3 Infection Dynamics

The model represents infection as a series of stochastic state transitions, updated on an hourly clock. Each host progresses through four main clinical states according to probabilistic rules that capture:

- **Latency and symptom onset:** hosts may remain asymptomatic or develop mild symptoms, during which they continue their normal schedule but contribute to transmission.
- **Progression and severity:** mild cases can worsen to severe illness, at which point hosts are confined at home or in hospital wards.
- **Critical care demand:** severe cases may progress to critical status, requiring ICU admission; if there is no space available, mortality risk rises.
- **Outcome and recovery:** survivors leave critical care and spend a convalescent period in a post-ICU compartment before rejoining their regular activities.

All transitions are governed by rules whose probabilities depend on within-host viral load, immune response markers, and healthcare capacity; this stochastic framework allows LOIMOS to capture both individual variability and the aggregate epidemic trajectory under different intervention scenarios.

2.3.4 Key Features

- **Virus and antibody objects:** `v1` denotes active virus (per mille units), `v1_ino` harmless virus; `antiv` (non-specialized antibodies) and `antivesp` (specialized antibodies) evolve via rules such as `antiv v1 -> antiv v1_ino sint` (non-specialized response) and `antivesp v1 -> antivesp v1_ino sint` (enhanced clearance), with `antivesp` appearing when 200 `antiv` and 200 `v1` accumulate, fully eliminating `v1` and curing the host.
- **Symptom/status objects:** Each host carries exactly one of `E1`–`E4` to represent clinical state: `E1` (no symptoms), `E2` (mild), `E3` (severe), `E4` (critical). A high-load mechanism—triggered when `sint` ≥ 700 creates `flag` then `cont`—drives transitions `E1->E2`, `E2->E3`, `E3->E4`, and the reverse when viral load subsides.
- **Immunity and reinfection:** Upon cure, hosts retain `antivesp` objects conferring partial protection; immunity wanes as `antivesp` and `v1_ino` are consumed by phagocyte (`phag`) rules, permitting possible reinfection under high-exposure events.
- **Control markers and capacity effects:** `sint` counts recent viral-particle events, guiding `flag/cont` for symptom onset; `phag` eliminates harmless virus (`v1_ino`); ICU capacity is tracked and reduced by each `E4` host, with overflow handled via auxiliary compartments.
- **Integration with interventions:** A global transmission-reduction parameter scales the probability of passing `v1` between hosts, interacting with `E2`–`E4` to model isolation, masking, and lockdown effects dynamically.

2.4 MVT

The MVT model [7] is a membrane-based, stochastic epidemiological simulator built using P Systems with active membranes and object-oriented logic. It represents every region, place, and individual as nested membranes, encapsulates individual attributes and behaviors as objects, and integrates information-driven dynamics (e.g., caution and vaccination willingness) to capture realistic responses to rising case counts.

2.4.1 Membrane Hierarchy

- **Eco-Membrane (Skin):** The outermost membrane enclosing the entire model.
- **Province-Membranes:** Twelve membranes labeled by province codes (e.g., MI, MB, BS...), each containing:
- **Place-Membranes:** Schools ($SC_1 \dots SC_n$), workplaces ($WP_1 \dots WP_k$), hospitals ($HP_1 \dots HP_\ell$), and common areas ($CA_1 \dots CA_m$).
- **Individual-Membranes:** Each person is a membrane that enters and exits place-membranes.

2.4.2 Internal Objects

- **Temporal objects:** $Hour_i$ (0–23) and Day_j (1–7) regulate daily and weekly routines.
- **Host category objects:** g (Young), a (Adult), an (Elderly) tag age groups and guide movement patterns.
- **Infection status objects:** I_{inc} (incubating), I (infected), Imm (recovered with immunity), V (vaccinated).
- **Auxiliary objects:**
 - φ : infection-number counter for infected individuals in a place, used in contagion probability.
 - Counters for incubation days, infection days, hospitalization days, immunity duration, and vaccination efficacy/duration.

2.4.3 Infection Dynamics

1. **Contagion rules:** When individuals share a place-membrane, each susceptible may become infected with probability

$$P = \beta \times \frac{\varphi}{N} \times \psi(M), \quad \psi(M) = \frac{1}{1 + a M/N}$$

where β is a base rate depending on the disease, φ is the local infected count, N the local population, and $\psi(M)$ is the “caution” factor reducing risk as cases M rise.

2. State Progression:

- Incubation: $g_j \text{Inc}_1 \rightarrow g_j \text{I}$ (becomes infectious after last incubation day)
- Infection: $g_j \text{I}_x \rightarrow g_j \text{I}_{x-1}$ until $g_j \text{I}_1 \rightarrow g_j \text{Imm}$ (recovered with 180-day immunity)

These are modeled by probabilistic rewriting rules driving countdowns.

3. **Hospitalization & death:** Affected individuals may move into a hospital membrane with fixed rates and then either recover or die, represented by additional rewriting rules.

2.4.4 Key features

- **Disease progression and immunity:** Individuals pass through incubation and infection stages via object countdowns, leading to either temporary immunity or hospitalization. Recovery includes immunity of fixed duration, after which reinfection becomes possible.
- **Behavior and provinces:** The population varies its behavior based on changes in the epidemiological context. These behavioral adjustments, which are influenced by infection prevalence, impact both transmission rates and vaccination uptake. The model has a fixed number of provinces, and each holds an equal share of the total population. People move between these regions with probabilities tied to the infection level in the destination province, integrating behavior-driven mobility into the system's dynamics.
- **Information-driven infection control:** A caution factor $\psi(M)$ reduces transmission probability based on the number of infected in each province, simulating risk perception and precautionary behavior. This dynamic coupling allows infection pressure to modulate both movement and contagion likelihood.

Chapter 3

Definition of the Model

3.1 Conflict Resolution

To merge the LOIMOS and MVT frameworks into a single, coherent simulator, it was crucial to acknowledge and integrate their different implementation bases. LOIMOS was developed in Java, while MVT was written in Python. Therefore, in the joint work with Sandro Erba we decided to implement the existing MVT model, enhancing it with new features and translating the LOIMOS rules, which were originally defined in XML. Subsequently, the main points of divergence between the two frameworks were systematically identified and reconciled. This section adapts and integrates the conflict-resolution analysis from Erba, [8]. The analysis has been rephrased and extended to fit our unified framework. Table 3.1 summarizes each conflict and indicates which model served as the basis (“Taken from”) or whether a new approach was introduced.

1. **Programming language**

LOIMOS was implemented in Java with XML rule-management layer, making additions laborious. MVT, by contrast, leverages an object-oriented Python architecture that facilitates rapid extension. Consequently, the unified simulator adopts Python as its implementation language.

2. **Place membranes**

LOIMOS distinguishes 12 location types (e.g., houses, ICUs, secondary schools), whereas MVT originally modeled only four (schools, workplaces, hospitals, common areas). To balance realism and performance, we integrated LOIMOS’s most impactful membranes—houses, hospitals, ICUs, workplaces, schools, leisure centers, and the common area—while omitting lower-priority types to maintain computational efficiency.

Table 3.1: Summary of conflict resolution

Conflict	Taken from
Programming language	MVT
Place membranes	LOIMOS
Infection and death probability	New
Behavior	MVT
Infection Management	Both
Symptoms	LOIMOS
Movement and Quarantine	MVT
Weekend	LOIMOS
Immunity	MVT
Hospitalization	Both
Output	MVT
Recovery	Both

3. Infection and death probability

The two models used different base rates for transmission and fatality, reflecting distinct membrane definitions and disease assumptions. We introduce a unified calibration phase to harmonize infection and mortality rates across location-specific transmission weights and healthcare settings.

4. Behavior

MVT’s explicit modeling of decision-making—interprovincial travel, vaccination willingness, and a dynamic “caution” factor—was retained. We enhanced this module with a refined *Prudence Parameter* to modulate isolation decisions more granularly.

5. Infection Management

MVT tracks health status via discrete labels (Iinc, I, Imm), progressing deterministically each day. LOIMOS employs multiset objects (virus, antiv, antivesp) with stochastic rules for within-host dynamics. Our combined model embeds LOIMOS’s infection-management scheme into MVT’s state machinery to capture both discrete health states and viral-load processes.

6. Symptoms

Unlike MVT, LOIMOS encodes four symptom levels (E1–E4), each triggering specific isolation or hospitalization behaviors. These symptom objects are critical for modeling progression and care pathways,

so all levels and transition probabilities were adopted in the unified system.

7. Movement and Quarantine

MVT’s geography uses province membranes without quarantine rules; LOIMOS defines rule-based movement restrictions (curfews, house-only isolation). We preserve provincial mobility and introduce configurable quarantines that nullify non-house movement probabilities.

8. Weekend

LOIMOS implements differentiated weekend routines (e.g., leisure center visits), whereas MVT applies a uniform schedule. We integrated LOIMOS’s age- and location-specific weekend patterns to reflect higher weekend mixing.

9. Immunity

MVT enforces 180-day post-recovery immunity. LOIMOS’s complex immunity landscape (four infection types) was simplified: we retain a single-duration immunity for tractability.

10. Hospitalization

MVT applies a constant hourly chance of hospital admission with treatment effects. LOIMOS bases admission on symptom severity ($E3 \rightarrow$ hospital, $E4 \rightarrow$ ICU). The integrated model uses LOIMOS thresholds for admission, together with MVT’s treatment duration, mortality reduction, and movement restrictions.

11. Output

LOIMOS generated detailed hourly CSV logs (224+ columns); MVT produced daily summary metrics. We adopt MVT’s concise daily outputs (current infected, new cases, prevalence, deaths), plus automated plotting.

12. Recovery

Both models include recovery transitions, but LOIMOS’s antibody-driven clearance adds mechanistic depth. This mechanism is combined with MVT’s deterministic recovery timing to capture stochastic immune success alongside a fixed-duration convalescence.

3.2 System of Membranes

In our unified simulator, each entity—whether spatial compartment or individual—is modeled as a *membrane*, providing both containment and rule-

driven behavior. Building on a hierarchical compartmentalization, and the classes defined in `membrane.py`, the model is composed by eight membrane types. Table 3.2 lists these core membranes, encompassing geographic regions, places, and individuals.

Table 3.2: Membrane classes in the simulation

Membrane Class	Description
ProvinceMembrane	Geographic aggregation of places
HouseMembrane	Domestic household units
SchoolMembrane	Educational settings for children
WorkPlaceMembrane	Adult workplaces
LeisureCenterMembrane	Social venues and leisure activities
CommonAreaMembrane	General public spaces (markets, transit)
HospitalMembrane / ICUMembrane	Healthcare and critical care facilities
IndividualMembrane	Agent-level health state and behavior

The **ProvinceMembrane** forms the top-level compartment, encapsulating all place-based sub-membranes within its geographic boundary. It orchestrates other membrane initialization, vaccination campaigns, infection progression, and migration.

Each of the place membranes captures a different environment in which individuals interact. The **HouseMembrane** models family units, supporting realistic household size distributions and incorporating age-stratified infection probabilities to reflect the close-contact nature of domestic settings. **SchoolMembrane** and **WorkPlaceMembrane** simulate the structured environments of education and employment, with moderate base infection rates and temporal routines based on weekday activity patterns.

Social and informal mixing is handled by the **LeisureCenterMembrane** and the **CommonAreaMembrane**. The former emphasizes elevated transmission during evening and weekend periods, particularly among the young and middle-aged population, while the latter accounts for casual interactions in public spaces such as markets and transport hubs, with heightened attention to the elderly’s vulnerability.

Healthcare environments are simulated by the **HospitalMembrane** and its specialized subclass, the **ICUMembrane**. These membranes operate under limited capacity and face elevated transmission risks. The ICU membrane is based upon the hospital model by incorporating even stricter capacity constraints, which mimics the controlled access and critical care functions characteristic of intensive medical units.

At the most granular level, each individual in the system is represented by an **IndividualMembrane**. These agent-level membranes encapsulate personal health states, behavior profiles, and disease progression trajectories. By treating individuals as membranes, the simulator achieves consistency with the P-system metaphor, enabling concurrent execution of individual rules such as infection progression, symptom emergence, recovery, vaccination response, and decision-based movement.

Membranes interact through a set of explicitly defined operations: provinces coordinate updates to their internal places; these evaluate infection and movement outcomes for their occupants; and individuals trigger state transitions in response to environmental exposure. A central infection rules module applies uniform stochastic procedures across all membrane types, ensuring that disease spread remains coherent across both spatial and agent-level dynamics. This membrane-oriented architecture preserves modular clarity while faithfully reproducing the complex behaviors of a heterogeneous population during an epidemic.

Spatial structure is established by the **ProvinceMembrane** and its various child classes. Within each province, there will be schools, workplaces, hospitals, leisure centers, common areas, and houses. These membranes track local infection statistics and apply infection rules specific to their respective scopes.

Together, these class-based objects and coherent infection dynamics provide a modular simulation platform capable of representing complex epidemic scenarios with both biological fidelity and behavioral adaptability.

3.3 System Architecture: Objects and Rules

The simulation’s architecture relies on a tightly coupled set of Python class-based objects and rule systems. These components collectively manage every aspect of temporal progression, individual behavior, infection dynamics, and system-wide epidemiological trends. Far from being mere data containers, these objects embody the logic and transitions of the entire model, making possible rich and configurable simulations of disease spread.

At the heart of the simulation are the temporal objects **days** and **hours** which regulate the main loop of the simulation. Each simulated day is divided into 24 hours, allowing for the modeling of daily routines, behavioral schedules, and rule applications. This structure ensures that events such as school attendance, work shifts, and curfews are executed with a sensible order.

The simulation’s orchestration is distributed procedurally within the sim-

ulation.py script, which serves as the main controller. It initializes provinces and places, seeds the infection, and updates all relevant system states over time. Outputs such as current infection counts, deaths, recoveries, and prevalence are tracked and logged through this module.

Individual agents are modeled by the **Individual** class, which encapsulates key attributes such as **age_group**, **symptoms**, **infection_status**, **hospitalization**, and **vaccination**. These attributes evolve stochastically over time as the simulation invokes movement and infection rules, by allowing agents to interact realistically with their environment.

Infection transmission is governed by two modular systems that can be toggled via the GUI: a biologically detailed virus-based system and a more classical state-transition model. In the virus-based model, infection is tracked using symbolic entities—**v1** objects—that represent viral particles. As individuals accumulate these viral agents, they become infected. Then, their immune systems respond by generating **antiv** and **antivesp** tokens, representing antibodies. Once the level of **antivesp** reaches a threshold, the infection is cleared and the individual recovers. This stochastic immune response is expressed as:

$$P(\text{generate antivesp}) = p_{a,s} \cdot 3$$

where $p_{a,s}$ is the probability of antibody production based on age and symptom level. The value 3 was chosen as a calibration parameter, after performing parameter tuning.

Alternatively, the state-based system models infection using a timed progression through discrete epidemiological stages. Individual transition from susceptible to exposed state, then to infectious and symptomatic state, based on incubation delays and symptom escalation probabilities. Symptom progression is defined by:

$$P(E2 \rightarrow E3) = 0.0015, \quad P(E3 \rightarrow E4) = 0.001$$

These values align with those calibrated in Erba’s thesis [8], reflecting gradual disease worsening for a minority of symptomatic individuals.

Contagion across individuals occurs within shared membranes. The probability of infection is determined by the infection density in a given place and is modulated by behavioral caution:

$$P(\text{infection}) = \frac{\beta \cdot I}{N} \cdot \frac{1}{1 + \frac{M}{f^*}}$$

where I is the number of infected individuals in the location, N is the total number of individuals, β is the base transmission rate of the membrane, and M/f^* encodes a behavioral caution term (defined in section 3.4.2).

Recovery can occur through either symptom resolution (in the state-based model) or via immune clearance (in the virus-based model). In the virus-based model, recovery is triggered once:

$$\text{antivesp} \geq 40$$

This hard threshold captures the point at which accumulated immunity is sufficient to eliminate the viral load.

Immunity is implemented as a temporary state, with individuals transitioning to susceptible again after a configurable number of days.

Mortality is modeled probabilistically and linked to symptom severity. Infected individuals at E3 and E4 levels are evaluated hourly for potential death. The probabilities used are:

$$\begin{aligned} P(\text{death}_{E3}) &= 1 \cdot 10^{-4} \\ P(\text{death}_{E4, ICU}) &= 6 \cdot 10^{-4} \\ P(\text{death}_{E4, \neg ICU}) &= 12 \cdot 10^{-4} \end{aligned}$$

These calibrated values ensure that critical cases are rare but represent meaningful epidemiological risk.

While infection, recovery, and mortality are modeled independently of individual behavior, several parameters—such as exposure, movement, and response to symptoms—are influenced by behavioral models. These are described in detail in the next section.

3.4 Behavioural Model

Human behavior plays an important role in shaping the trajectory of epidemic spread, particularly when it comes to precautionary measures and vaccine uptake. To reflect this, our simulator integrates a behavioral response model that modulates individual decisions based on current epidemiological context.

3.4.1 The Prudence Parameter

A central component of the behavior model is the *prudence parameter*, denoted as ρ , which governs how symptom severity influences individual movement decisions. It serves as a behavioral modifier that adjusts the probability of leaving one’s residence based on health status and perceived epidemiological risk.

When an individual shows moderate symptoms (E2), their movement probability is adjusted by ρ . This adjustment’s impact depends on both how severe their symptoms are and the infection pressure at their planned destination.

A higher value of ρ reflects more cautious behavior: symptomatic individuals are significantly more likely to stay at home. Conversely, a lower ρ suggests a more risk-tolerant or less compliant population. The value of this parameter ranges from 0 to 1.

This design is consistent with real-world observations. Block et al.[9] demonstrated that behaviorally adaptive distancing—where individuals strategically reduce social contact based on perceived local infection risk—can effectively mitigate epidemic spread. Their findings underscore that individual behavioral responses, especially within community structures or repeated interaction networks, can substantially flatten the infection curve without the need for blanket restrictions.

In our simulator, the prudence parameter encapsulates this behavioral feedback mechanism. It allows agents to make decisions that realistically reflect how individuals weigh personal health, social norms, and environmental cues when navigating a pandemic. This adjustable parameter therefore connects micro-level behavior with macro-level policy outcomes, making the model more responsive to diverse human dynamics.

3.4.2 The Caution Factor

The caution factor is a dynamic behavioral component that modulates the probability of contagion based on the overall infection level in the population. It reflects an adaptive risk-response mechanism: as the epidemic worsens, individuals are assumed to become more cautious—through actions such as spontaneous distancing, improved hygiene, or reduced social contact—even in the absence of formal mandates.

This effect is encoded directly into the contagion formula through a saturation function that scales down the effective transmission rate in shared spaces. The probability that a susceptible individual becomes infected in a given membrane is given by:

$$P(\text{infection}) = \frac{\beta \cdot I}{N} \cdot \psi(M)$$

$$\psi(M) = \frac{1}{1 + \frac{M}{f^*}}$$

where:

- I is the number of infected individuals in the membrane,
- N is the total population of that membrane,
- β is the membrane-specific base transmission rate,
- M is the number of currently infected individuals in the entire population,
- f^* is a normalization constant defining the sensitivity of behavioral caution.

The function $\psi(M)$ decreases as M increases, which diminishes contagion. A lower value of f^* means that even a small rise in infection levels will lead to a major behavioral response, which models a very vigilant population. On the flip side, a higher value of f^* results in a slower decline of $\psi(M)$, indicating that individuals are less reactive and more tolerant to rising infections before adjusting their behavior.

This mechanism enables the model to simulate varying levels of societal responsiveness and provides a flexible structure for testing how behavioral inertia or vigilance influences epidemic outcomes.

3.4.3 Vaccination Willingness

Vaccination willingness is modeled as a dynamic behavioral response to rising infection pressure. Rather than assuming constant or uniform compliance, the model allows each individual's probability of accepting vaccination to vary with the perceived severity of the epidemic.

This behavior is captured by a sigmoid-like response function that increases non-linearly with the proportion of infected individuals in the population. The willingness function is defined as:

$$W(M, N) = 1 + \frac{\left(\frac{f}{F^*}\right)^2}{1 + \left(\frac{f}{F^*}\right)^2}, \quad \text{with} \quad f = \frac{M}{N}$$

where:

- M is the total number of currently infected individuals,
- N is the total population size,
- f is the current infection ratio,

- F^* is a tunable threshold that controls the steepness of the willingness curve. It represents a parameter that dictates how reactive the population is to rising infection levels when it comes to vaccine uptake

This formulation ensures that when the infection ratio is low, willingness remains close to 1 (minimal urgency). As infections rise and f approaches F^* , the function increases rapidly, approaching 2 in the limit. This models the population's tendency to become significantly more receptive to vaccination as perceived risk increases.

Lower values of F^* cause the willingness to increase sharply even with minor rises in infection, indicating a highly risk-averse or responsive population. Higher values of F^* suggest that vaccination willingness only grows under extreme epidemic pressure, characterizing populations that may exhibit hesitancy or denial.

This adaptive behavioral feature lets users explore how varying levels of risk perception affect vaccination rates and the subsequent success of epidemic containment efforts.

Chapter 4

Simulator

The simulator was developed as part of a joint effort with Sandro Erba (University of Milano Bicocca), whose main focus was on the formal definition of the epidemiological model. His work laid the theoretical groundwork that guided many of the design choices. While I focused on the software development and testing, Sandro contributed actively to the coding process as well, especially during the early implementation stages and in fine-tuning certain components.

Our collaboration proceeded with an iterative approach. We reviewed design options, discussed potential improvements, and mutually made decisions to balance model accuracy with computational efficiency.

Input from Claudio Zandron (University of Milano Bicocca), who co-supervised the work, helping align the technical aspects of the simulator with the research objectives. Overall, the collaboration strengthened the project on both a theoretical and practical level.

4.1 Workflow and Daily Routine

The core of the `Simulation` tool is a discrete-time, agent-based loop that advances in hourly steps over a sequence of simulated days. Each simulation run proceeds through three main stages: initialization, hourly time-stepping, and periodic updates to individual health states. The following describes the internal mechanics of the simulation and how individual routines are defined by temporal and demographic patterns, shown in Tables 4.1 and 4.2.

At initialization, a fixed number of `ProvinceMembrane` objects is created (decided by the user via GUI), each containing a set of spatial membranes (`House`, `School`, `WorkPlace`, `LeisureCenter`, `CommonArea`, `Hospital`, `ICU`). These represent the main locations where agents interact. Individual agents

(`IndividualMembrane`) are instantiated with assigned age groups (Young, Adult, Elderly), home residences, and an initial infection status—usually healthy, with a seeded minority of infected cases. Global parameters such as transmission rates, quarantine rules, and behavioral caution factors are set before the simulation begins.

Each simulated day is broken into 24 hourly iterations. In each hour, agents may decide to move from their current location to another (e.g., from `House` to `WorkPlace` or `School`) based on their age group, the time of day, whether it is a weekday or weekend, and the current epidemiological state (e.g., quarantine in effect). These movements are governed by rules encoded in the `Simulation` class and are derived directly from the patterns in the tables below.

Once the population has been relocated, infection dynamics uncoil within each location at varying rates. Then, health state progression is updated: exposed individuals advance through disease stages (E1–E4), potentially necessitating hospitalization or ICU care, and ultimately either recover or die, depending upon probabilistic outcomes.

Daily routines within the simulation vary based on age group and the type of day. The majority of movements are percentage-based, meaning not all individuals adhere to an identical routine. On weekdays, younger agents typically attend school during morning and afternoon hours, returning home for lunch, and may utilize leisure centers or common areas in the evening. Adults follow a comparable schedule focused on their workplaces, often including a morning stop at a common area. Elderly individuals predominantly remain at home or make brief visits to public spaces. During weekends, all age groups transition to more adaptable patterns, characterized by increased time spent in leisure and public areas.

Tables 4.1 and 4.2 summarize the implemented movement routines for weekdays and weekends, respectively. These behaviors are applied uniformly across agents within the same category and hour.

4.2 Graphical User Interface

The simulator provides a user-friendly interface implemented by using Python Tkinter library. This graphical user interface (GUI) allows users to visually configure epidemic scenarios by adjusting a wide range of parameters without the need to write or modify code. The interface is organized into a tabbed layout using a `ttk.Notebook` widget, where each tab groups related inputs and controls. A high-level screenshot of the GUI is shown in Figure 4.1 (place `gui_overview.png` here).

Table 4.1: Weekday routine per age group and hour. LC stands for Leisure Center and CA stands for Common Area

Hour	Young	Adult	Elderly
1	House \leftarrow LC	House \leftarrow LC	House
2–5	House	House	House
6	House	CA	House
7	CA	Work	House
8–16	School	Work	House/CA/LC
17–18	School \rightarrow House	Work \rightarrow House	House/CA
19	House	House	House
20	House	House	House
21–23	House \leftrightarrow LC	House \leftrightarrow LC	House
24	House \leftarrow LC	House \leftarrow LC	House

Table 4.2: Weekend routine per age group and hour. LC stands for Leisure Center and CA stands for Common Area

Hour	Young	Adult	Elderly
1–4	House \leftarrow LC	House \leftarrow LC	House
5–7	House	House	House
8–12	House \leftrightarrow CA \leftrightarrow LC	House \leftrightarrow CA \leftrightarrow LC	House \leftrightarrow CA \leftrightarrow LC
13–14	House	House	House
15–19	House \leftrightarrow CA \leftrightarrow LC	House \leftrightarrow CA \leftrightarrow LC	House \leftrightarrow CA \leftrightarrow LC
20–21	House	House	House
22–24	House \leftrightarrow LC	House \leftrightarrow LC	House

The GUI comprises several tabs that segment the input fields into logically coherent categories.

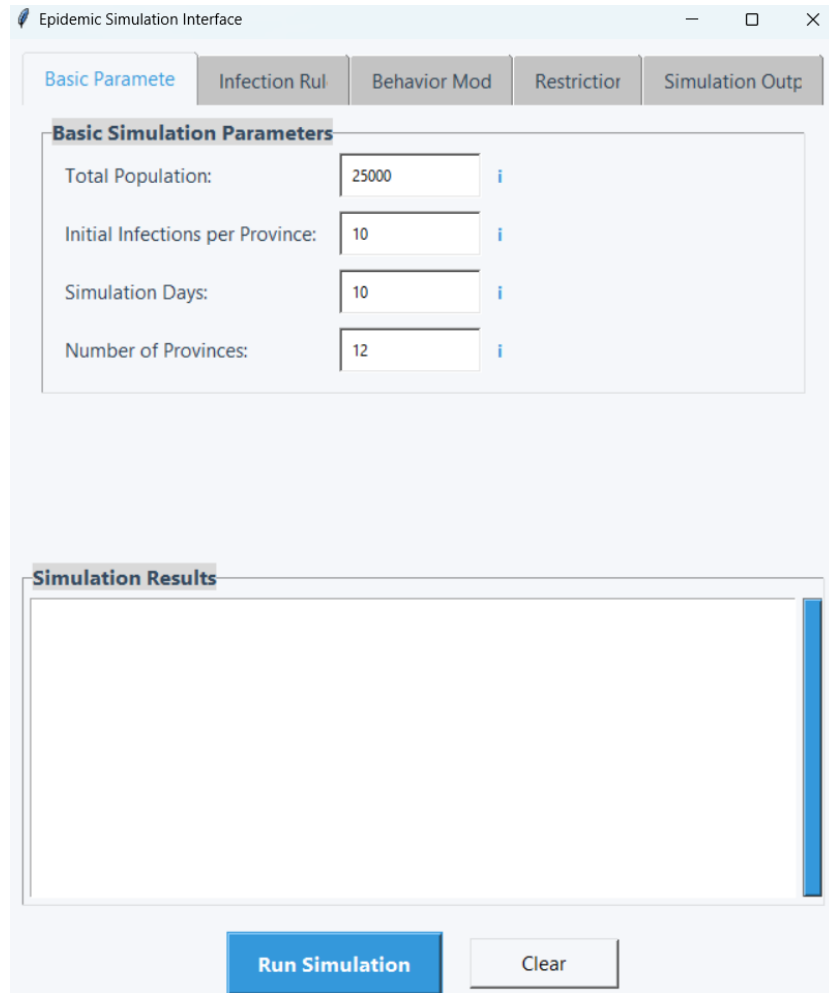
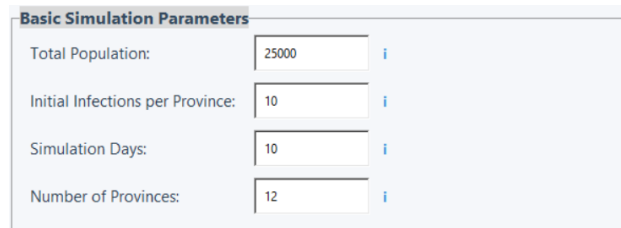


Figure 4.1: Main window of the epidemic simulation interface.

The first tab, labelled **Basic Parameters**, includes fields for setting the total population, the number of initially infected individuals per province, the total number of provinces, and the overall simulation duration. A visual representation is provided in Figure 4.2. The next tab, **Infection Rules**, allows users to configure epidemiological parameters such as the incubation period, caution factor, and prudence parameter. It also includes toggle buttons to enable or disable aspects like behavioral modeling, vaccination strategies, the effect of viral load, and ICU presence, as shown in Figure 4.3. The **Behavior Model** tab permits finer control over behavioral dynamics through settings

for vaccine coverage, a duration–correlation factor, and the F_{star} parameter (Figure 4.4). A further tab named **Restrictions** includes options to simulate quarantine policies, specifying whether quarantine is enabled, its duration, and when it begins.



The 'Basic Simulation Parameters' tab contains four input fields, each with an information icon (i) to its right:

Parameter	Value
Total Population:	25000
Initial Infections per Province:	10
Simulation Days:	10
Number of Provinces:	12

Figure 4.2: Basic parameters tab.



The 'Infection Parameters' tab contains six input fields, each with an information icon (i) to its right:

Parameter	Value
Incubation Period:	5
Caution Factor:	0.001
Prudence Parameter:	0.9
Behavior Trigger:	<input checked="" type="checkbox"/>
Vaccination Trigger:	<input type="checkbox"/>
Viral Load:	<input checked="" type="checkbox"/>
ICU Presence:	<input checked="" type="checkbox"/>

Figure 4.3: Infection rules tab.



The 'Behavior Model Parameters' tab contains three input fields, each with an information icon (i) to its right:

Parameter	Value
Vaccine Coverage:	0
Duration Correlation:	0.8
F Star:	0.01

Figure 4.4: Behavior model tab.

A key usability feature of the GUI is the presence of information icons beside each input field. When the user hovers over one of these icons, a tooltip appears to provide a short, descriptive explanation of the parameter's role or its expected value range. For instance, hovering over the enable quarantine input displays all the important information about this parameter. Figure 4.5 illustrates this feature.

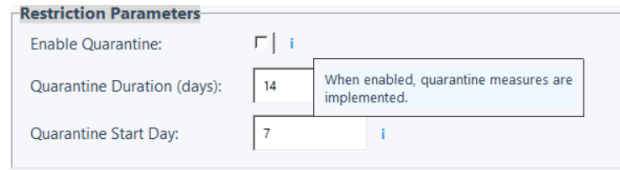


Figure 4.5: Tooltip providing parameter guidance.

Beneath the tabbed area, two main control buttons are available. The first, labeled **Run Simulation**, validates the entered values, generates the necessary input files, and initiates the simulation process. The second button, **Clear**, resets all fields to their default state. On the right-hand side of the main window, a scrollable text area logs messages related to simulation progress. Part of the interface, the contents and output of this log are discussed separately in Section 5.3, which is dedicated to simulation results.

To ensure usability, the Graphical User Interface (GUI) features a simple and accessible layout. Selected tabs transition to a contrasting color, and input fields are uniformly styled to enhance clarity. The labels follow a clear visual hierarchy, and interactive components like scrollbars and buttons share a unified design approach. This design pattern significantly improves the user experience, rendering the interface both intuitive and visually appealing.

4.3 Simulation Output Overview

The simulator produces comprehensive outputs in both machine-readable (CSV file) and visual formats (graphical and table formats) as seen in Figure 4.6 and Figure 4.7, facilitating a thorough analysis of simulated epidemic dynamics. These outputs encompass structured data files detailing daily statistics and automatically generate visualizations that aid the interpretation.

4.3.1 Data Files and Automated Visualizations

Upon completion of each simulation, a dedicated output directory is created within the project's root folder, typically under `simulations/`. This directory contains the following components:

- **CSV Summary File:** A structured data file following the naming convention `simulation.<population>.<provinces>.<days>.<timestamp>.csv`. This file provides a comprehensive tabular summary of daily simulation metrics, including:

- Daily incidence (number of new infections per day)
 - Disease prevalence (total active cases)
 - Cumulative mortality count
 - Computational performance metrics (execution time in seconds per simulated day)
- **Automated Graph Generation:** Within the `graphs/` subdirectory, three key epidemiological plots are automatically generated and saved as high-resolution PNG files:
 - Epidemic curve showing daily new infections over time
 - Prevalence trajectory displaying total active cases throughout the simulation period
 - Cumulative mortality curve tracking deaths over time

These visualizations enable rapid assessment of epidemic patterns and facilitate identification of key epidemiological phases such as growth, peak, and decline periods.

4.3.2 Interactive Graphical User Interface

Beyond file-based outputs, the simulator offers immediate visual feedback through an integrated graphical user interface (GUI). This feature significantly enhances the user experience and enables real-time monitoring of simulation progress.

- **Integrated Visualization Panel:** Upon simulation completion, the GUI automatically presents the three generated plots within a dedicated "Simulation Output" tab. This embedded visualization capability obviates the need for external file navigation, providing instant access to key epidemiological indicators
- **Real-time Simulation Monitor:** During simulation execution, a live console displays continuous updates including daily statistics such as new case counts, mortality figures and prevalence levels. The system also provides (on the terminal) automated alerts for significant epidemiological events, including quarantine implementations and epidemic stabilization indicators (e.g., "No new infections detected").

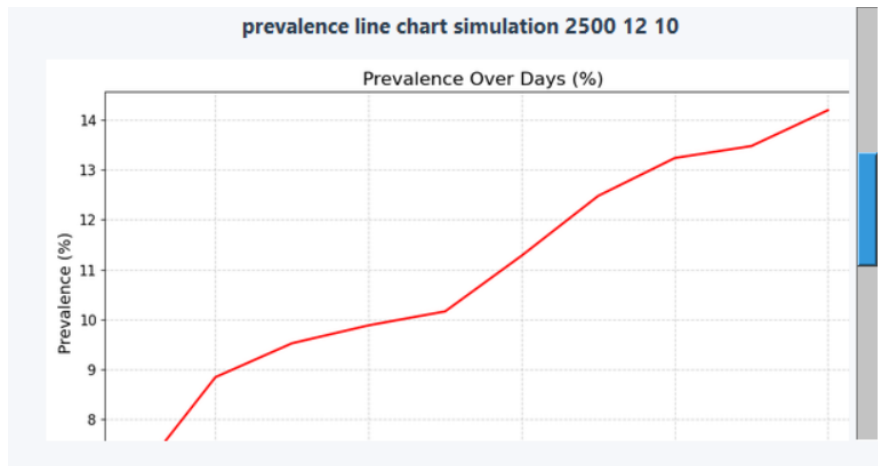


Figure 4.6: The prevalence chart tracks how the epidemic unfolds over time, showing what percentage of the population gets affected. It captures the typical pattern of disease spread - starting small, building momentum, and eventually reaching its peak before declining.

Day	New Cases	Prevalence	Deaths	Seconds
1	52	172	0	0.81
2	49	221	0	0.95
3	17	238	0	1.10
4	9	247	0	1.71
5	7	254	0	1.06
6	28	282	0	2.24
7	30	312	0	2.27
8	19	331	0	2.39
9	6	337	0	2.79
10	18	355	0	1.39

Simulation completed!

Figure 4.7: The simulation results table presents daily epidemiological metrics, which include new cases, prevalence, deaths, and computational performance. This tabular format furnishes precise numerical data for each simulation day, thereby facilitating detailed quantitative analysis.

Chapter 5

Results

This chapter presents the findings from the computational simulations, focusing on the validation of the model’s behavioral parameters and its overall performance in simulating disease spread. All simulations and validations discussed herein are designed and interpreted within the context of COVID-like viral dynamics, reflecting characteristics such as airborne transmission, varied symptom severity, and the influence of population behavior on epidemic trajectories.

5.1 Parameter Validation

This section demonstrates the operational effectiveness of the behavioral components newly incorporated within the model framework, confirming their appropriate influence on simulated epidemic dynamics. These parameters were implemented to authentically represent behavioral adaptation patterns within the population. The findings reveal how integrated behavioral mechanisms, particularly prudence and caution responses, modify pathogen transmission dynamics, thereby providing valuable insights into their implications for public health outcomes.

5.1.1 The Prudence Parameter (PP)

The Prudence Parameter (ρ) is a central component of the behavioral model, governing how symptom severity influences an individual’s movement decisions. It acts as a behavioral modifier, adjusting the probability of individuals leaving their residence based on their health status and the perceived epidemiological risk.

In the context of a COVID-like disease simulation, the Prudence Parameter directly reflects the cautious behavior of individuals. A higher value of ρ indicates a more cautious population where symptomatic individuals are significantly more likely to stay at home. Conversely, a lower ρ suggests a population that is more risk-tolerant or less compliant with self-isolation measures. This mechanism enables agents to make endogenous decisions that mirror real-world balancing of personal health, social norms, and environmental cues during a pandemic. The model captures the concept that individuals strategically reduce social contact based on perceived local infection risk, which is known to effectively mitigate epidemic spread.

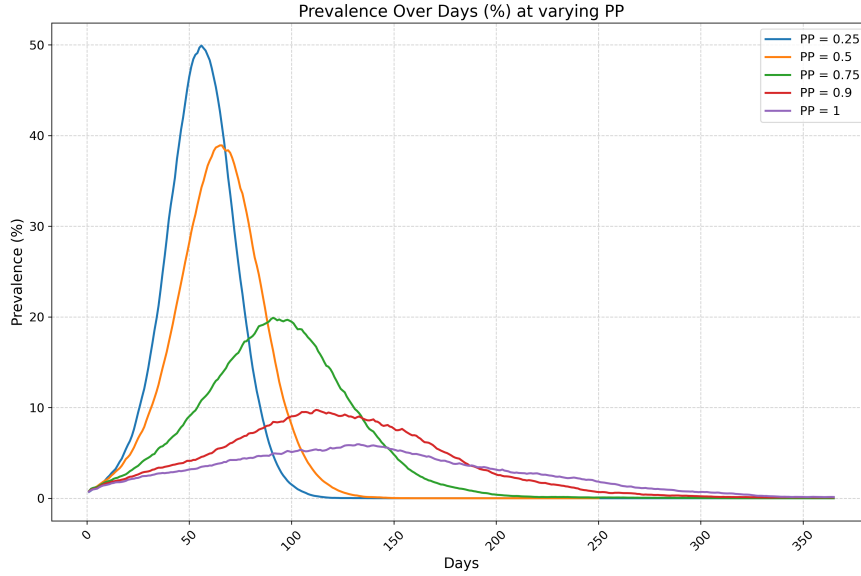


Figure 5.1: This graph illustrates the prevalence of infection over time by varying Prudence Parameter (PP) values, from 0.25 to 1.0.

The observed trends in Figure 5.1 clearly demonstrate the impact of the Prudence Parameter on the epidemic trajectory. Lower PP values (e.g., 0.25 and 0.5) lead to sharp, early peaks of infection, indicating a rapid spread through a less cautious population. These peaks are notably higher, reaching approximately 50% prevalence for PP=0.25 and 40% for PP=0.5. Following these rapid ascents, the infection prevalence declines to zero. In contrast, higher PP values (e.g., 0.75, 0.9, and 1.0) produce broader, lower curves. This signifies a more controlled and mitigated epidemic spread, characterized by lower peak prevalence (e.g., approximately 20% for PP=0.75 and 10% for PP=0.9) and more gradual declines. The shift from sharp, high peaks to flatter, lower curves with increasing PP values highlights that greater

individual prudence can substantially flatten the infection curve, reducing the overall burden of the epidemic and delaying its peak. This outcome aligns with real-world observations where behavioral adaptations, such as distancing, contribute to mitigating disease transmission [9] [10].

5.1.2 The Caution Factor (CF)

The Caution Factor functions as a dynamic behavioral element that adjusts infection probability according to population-level disease prevalence. This represents an adaptive risk-response system where individuals naturally become more cautious as epidemic severity increases. This heightened awareness manifests through behaviors like voluntary social distancing, improved hygiene protocols, and reduced interpersonal contact, occurring independently of official public health directives.

In the context of a COVID-like disease simulation, the **Caution Factor** models how a population’s **risk perception** dynamically influences **transmission rates**. A lower value of f^* —which defines the **sensitivity of behavioral caution** within the $\psi(M)$ function, where $\psi(M)$ decreases as the total infected individuals (M) increases—signifies that even a modest rise in **infection levels** will elicit a robust behavioral response, leading to a steep decline in transmission. Conversely, a higher value of f^* indicates that individuals exhibit reduced reactivity and greater tolerance to escalating infection pressure before modifying their behavior, resulting in a slower decrease of $\psi(M)$. This mechanism enables the model to simulate diverse levels of **societal responsiveness**.

The trends observed in Figure 5.2 illustrate the impact of the Caution Factor (CF) on the epidemic spread. Specifically, lower CF values (e.g., 0.01 and 0.001) diminish the peak’s height. This indicates that a more vigilant population, one that adopts cautious behaviors faster, can effectively flatten the epidemic curve and attain a reduced peak prevalence. Contrarily, higher CF values (e.g., 1 and 0.1) elevate the peak’s height, resulting in a more widespread prevalence of infections overall. These findings indicate that populations with lower sensitivity to rising infection rates will experience more severe epidemic peaks.

5.1.3 Vaccination Willingness (F^*)

Vaccination willingness is modeled as a dynamic behavioral response to rising infection pressure, allowing each individual’s probability of accepting vaccination to vary with the perceived severity of the epidemic. This behavior is

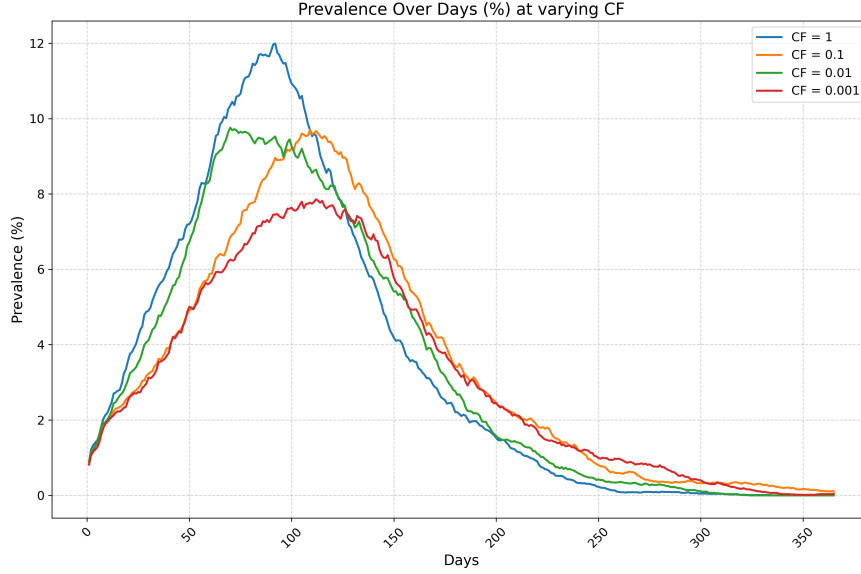


Figure 5.2: The graph illustrates the prevalence of infection over time for different Caution Factor (CF) values, ranging from 1 to 0.001.

captured by a sigmoid-like response function that increases non-linearly with the proportion of infected individuals in the population.

In the context of a COVID-like disease simulation, the parameter F^* dictates how reactive the population is to rising infection levels when it comes to vaccine uptake. Lower values of F^* cause vaccination willingness to rise steeply with even small increases in infection, reflecting a highly risk-averse or responsive community. Conversely, higher values of F^* indicate that vaccination willingness increases only under extreme epidemic pressure, capturing populations that may exhibit hesitancy or denial. This behaviorally adaptive mechanism allows the simulator to test how different risk sensitivities impact vaccine uptake rates and, consequently, overall epidemic control.

For COVID-like diseases, the primary objective of vaccination campaigns is often not to completely prevent infection, but rather to reduce the severity of illness, hospitalization rates, and crucially, mortality. Therefore, while vaccination might impact prevalence to some extent, its most significant effect is typically observed in the reduction of severe outcomes and deaths. This aligns with findings that emphasize the role of vaccines in preventing severe disease, hospitalization, and death, even if they do not entirely block transmission [11] [12].

The observed trends in Figure 5.3 demonstrate this critical outcome. Lower F^* values, representing a population highly responsive to increasing

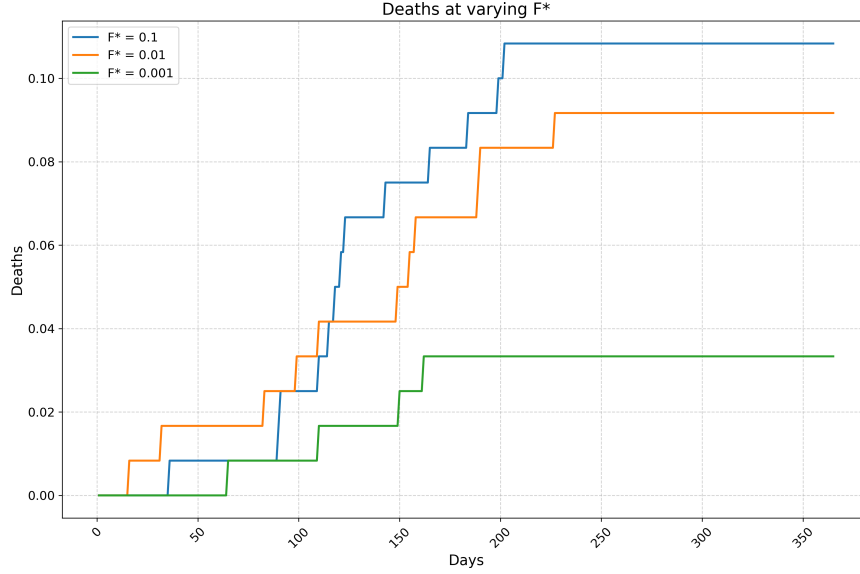


Figure 5.3: The graph depicts cumulative mortality over time across different F^* values, where F^* represents the adjustable threshold parameter that determines the slope of the vaccination acceptance curve.

infection rates and thus more willing to vaccinate quickly, lead to a significantly lower number of cumulative deaths. This is evident as the curves for lower F^* (e.g., $F^* = 0.001$) show a much flatter trajectory and reach a considerably lower final death toll compared to higher F^* values. Conversely, higher F^* values, reflecting a less responsive population with delayed or reduced vaccine uptake, result in a higher cumulative death count. This clearly highlights that a population's willingness to vaccinate in response to perceived risk plays a direct and substantial role in mitigating the most severe outcomes of an epidemic.

5.2 Dataset and Simulation Setup

5.2.1 Dataset Description

To validate the model against real-world data for COVID-like pandemics, a dataset from the Italian region of Veneto was utilized. This real-world data was sourced from the official GitHub repository of the Italian Civil Protection Department: <https://github.com/pcm-dpc/COVID-19/tree/master/dati-regioni>. The raw data underwent a filtering and manipulation process via custom Python scripts to extract only the most relevant metrics for

comparison, specifically daily cases, deaths, and prevalence. Processed data were then graphed to facilitate future analysis and comparison with the simulation outputs. For the purpose of simulating a 365-day period, the real dataset for Veneto was specifically aligned to cover the days ranging from day 570 to day 935 of the pandemic, providing a concrete reference period for validation. This time period also represents the highest peak of infection.

5.2.2 Simulation Setup

The simulation setup for the Veneto region was designed to align with previous validation efforts conducted by Sandro Erba, who successfully validated the model using data from the Lombardy region. To maintain consistency and allow for comparable results, the simulation parameters were scaled according to the population differences between Lombardy and Veneto. Given that Lombardy has an approximate population of 10 million and Veneto has approximately 4.8 million, the total simulated population was adjusted proportionally. Sandro Erba’s simulation for Lombardy utilized a population of 25,000 individuals; therefore, for Veneto, a simulated population of 12,000 individuals was used. Furthermore, Veneto comprises 7 provinces, and thus the simulation was configured to include 7 distinct regions. Consistent with the established methodology, the initial infection was set to infect less than 1% of the total population. Other less critical parameters were also set to match those used in Sandro Erba’s validation to ensure methodological consistency across the model’s applications.

5.2.3 Prevalence and Mortality Trajectories

This section presents the direct comparison between the simulation outputs and the real-world epidemiological data from the Veneto region, serving as a validation of our model predictive capabilities for COVID-like viral dynamics.

The model demonstrates successful performance in reproducing fundamental epidemiological patterns found in empirical observations. The initial figure, "Prevalence Over Days (%)", presents a 365-day simulation period. The model’s strongest characteristic is its ability to accurately recognize the peak of infection, a feature clearly illustrated in the image below. Throughout this extended timeframe, the model successfully forecasts the primary infection peak, with the simulated prevalence curve’s maximum closely corresponding to that documented in real-world data from the Veneto region. This alignment indicates the model’s effectiveness in capturing the essential dynamics of sustained epidemic transmission. The model was designed to capture the initial peak of the pandemic, and therefore struggles to detect

the subsequent peaks. This issue will be discussed in more detail in Section 6.1.

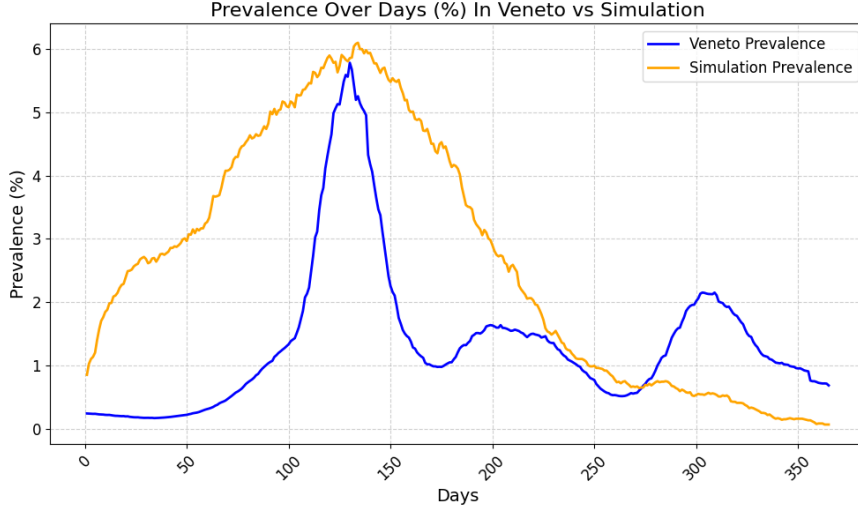


Figure 5.4: Prevalence over 365 days simulation

For a more granular validation, Figure 5.5, "Prevalence Over Days (%) In Veneto vs Simulation", illustrates a focused 90-day simulation. For this comparison, real data from Veneto covering days 1010 to 1100 of the pandemic were utilized. Within this shorter period, the simulation accurately predicts both the peak prevalence and the general trajectory of the pandemic, closely mirroring the rise and final prevalence observed in the real dataset. This detailed alignment over a shorter term further confirms the model's fidelity in capturing short-to-medium term epidemic behavior.

Beyond infection prevalence, the model ability to predict mortality trends is also critical. Figure 5.6, "Deaths Over Days (%) In Veneto vs Simulation", presents the cumulative deaths from the 365-day simulation. The graph demonstrates that the simulation correctly predicts the overall trend of deaths, showing a consistent accumulation of fatalities that closely follows the pattern seen in the real data. While the exact numerical values may vary slightly, the concordance in the cumulative death curve shape and progression reinforces the model robustness in simulating severe disease outcomes alongside infection dynamics. Collectively, these validations confirm that the developed model is capable of reproducing significant epidemiological patterns, including peak infection times and overall mortality trends, for COVID-like viral outbreaks.

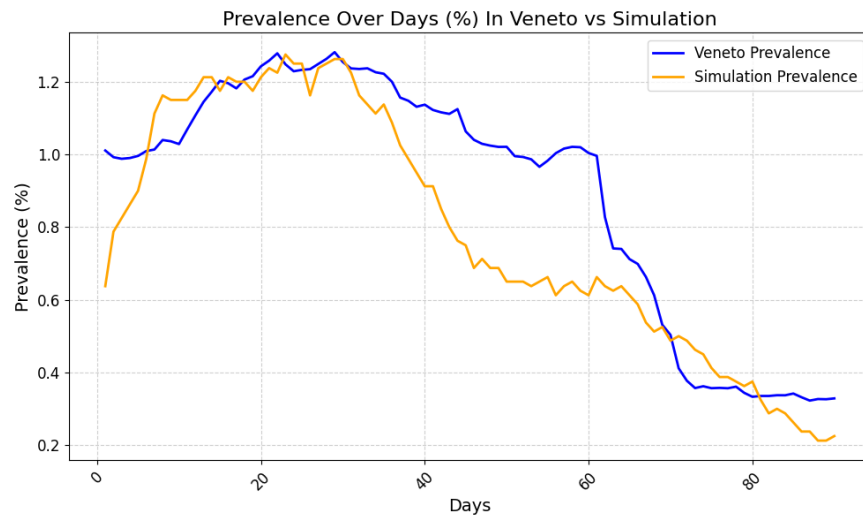


Figure 5.5: Prevalence over 90 days simulation

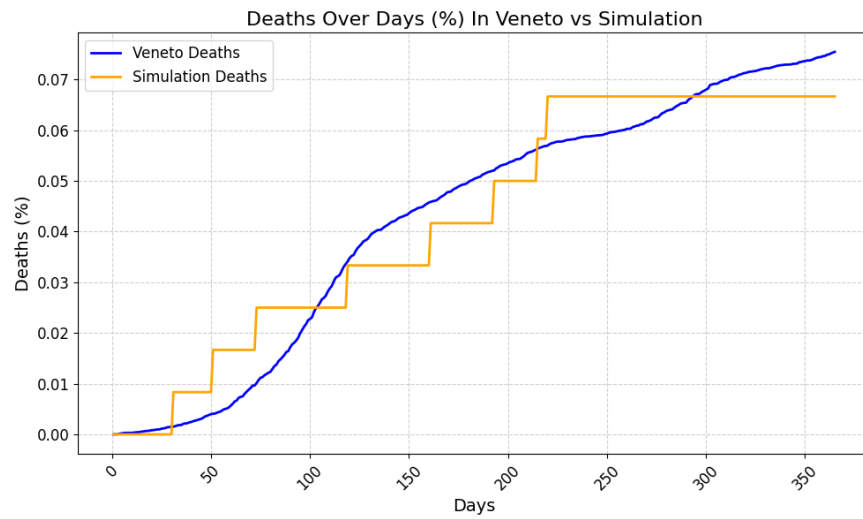


Figure 5.6: Deaths over 365 days simulation

Chapter 6

Conclusions

This thesis presented a new computational model for simulating disease spread within a population, leveraging the biologically-inspired framework of P-systems. By integrating and enhancing components from the LOIMOS and MVT models, we have developed a robust and adaptable simulator capable of capturing complex epidemiological dynamics, particularly those akin to COVID-19. The model architecture, rooted in hierarchical membranes and object-oriented logic, facilitates a high degree of modularity and scalability, allowing for nuanced representations of individual behaviors and spatial interactions.

A significant contribution of this work lies in the detailed validation of key behavioral parameters, namely the Prudence Parameter (PP), the Caution Factor (CF), and Vaccination Willingness. The simulations consistently demonstrated that these parameters, which modulate individual movement decisions, contagion probabilities, and vaccine uptake based on perceived epidemiological risk, profoundly influence epidemic trajectories. For instance, higher PP values led to a significant flattening of the infection curve and reduced peak prevalence, aligning with real-world observations on the efficacy of adaptive distancing. Similarly, lower CF values, indicative of a more vigilant population, resulted in a less pronounced epidemic peak. Crucially, the analysis of Vaccination Willingness confirmed that a population's responsiveness to rising infection pressure directly correlates with a substantial reduction in cumulative deaths, even if transmission is not entirely blocked. This empirical validation underscores the model ability to realistically simulate the intricate interplay between human behavior and disease progression.

Furthermore, the model predictive capabilities were rigorously assessed against real-world COVID-19 data from the Veneto region of Italy. The alignment of simulated prevalence and mortality trajectories with the observed data, particularly in predicting the timing of peak infections and

overall death trends, provides compelling evidence of the model’s fidelity. This real data based validation, observed across both long-term (365-day) and short-term (90-day) simulations, highlights the model accuracy in replicating complex epidemic patterns and its potential as a valuable tool for epidemiological forecasting and intervention analysis. The adoption of Python as the implementation language also enhances its applicability and ease of extension for future research.

In conclusion, this thesis successfully developed a biologically-inspired computational model that not only accurately simulates disease spread under varying behavioral responses but also provides a flexible platform for exploring public health interventions. Its capacity for realistic prediction make it a valuable asset for informing epidemiological strategies in the face of future outbreaks.

6.1 Model Strengths and Limitations

The unified computational model developed in this thesis demonstrates several key strengths while also exhibiting certain limitations that present opportunities for future enhancements.

Strengths

- **Integration of Diverse Frameworks:** A primary strength of the model is its successful integration of the LOIMOS and MVT frameworks. This merge combines LOIMOS’s detailed biological interactions and multi-scale dynamics (spatial, temporal, within-host) with MVT’s robust behavioral flexibility and object-oriented logic. This integration creates a more comprehensive and realistic simulator.
- **User-Friendly Interface and Output:** The developed graphical user interface (GUI) and comprehensive output mechanisms (CSV files, automated plots, interactive console) enhance usability and facilitate thorough analysis and real-time monitoring of simulations.
- **Behavioral Adaptability:** As a novelty the model rigorously incorporates human behavioral responses to epidemiological context through parameters like the Prudence Parameter (PP), Caution Factor (CF), and Vaccination Willingness. These elements enable the simulation of realistic adaptive behaviors such as self-isolation, social distancing, and vaccine uptake, profoundly influencing epidemic trajectories. The validation section consistently demonstrates how these parameters effec-

tively modify pathogen transmission dynamics and public health outcomes.

- **Accurate Prediction of Initial Epidemic Peak:** As demonstrated by the comparison with real-world data from the Veneto region, the model exhibits strong performance in reproducing fundamental epidemiological patterns, particularly in successfully forecasting the primary infection peak.
- **Validation against Real-World Data:** The model has undergone rigorous validation against real-world COVID-19 data from the Veneto and Lombardy region, showing alignment between simulated prevalence and mortality trajectories with observed data. This confirms its fidelity and reliability for COVID-like viral dynamics.

Limitations

- **Computational Intensity and Dataset Scaling:** The simulation, being agent-based and operating on hourly iterations, is computationally onerous. This intensity limits the ability to simulate populations as large as real-world datasets (e.g., comparing a 12,000-individual simulation to a 4.8 million population in Veneto). While proportional scaling is applied, the difference in scale can introduce discrepancies or reduce the granularity of population-wide effects, particularly for highly localized phenomena or rare events.
- **Fixed Population Size:** The current simulation operates on a fixed population size, focusing solely on disease-related mortality. For long-term epidemiological studies or diseases with lower mortality rates, the absence of natural birth and death rates limits its realism and comprehensive demographic representation.
- **Limited Prediction of Subsequent Peaks:** While the model excels at forecasting the primary infection peak, it is not currently capable of consistently predicting subsequent peaks in the epidemiological curve. This limitation suggests that the model's current mechanisms may not fully capture all the complex factors (e.g., variant emergence, waning immunity, staggered behavioral responses over extended periods) that contribute to multi-wave epidemic patterns observed in real-world data.

6.1 Future Work

The current model provides a solid foundation, yet several lines for future work can further enhance its computational efficiency and applicability. This list is arranged from the easiest to implement to the most challenging:

1. **Testing on Diverse Pandemics:** The model has been rigorously validated against COVID-19 data. A crucial next step would be to test its adaptability and predictive power against historical or hypothetical pandemics with different transmission characteristics (e.g., influenza, measles). This would involve recalibrating parameters such as base transmission rates and disease progression probabilities to match the specific pathogen. Such cross-validation would solidify the model’s generalizability and demonstrate its utility as a versatile tool for infectious disease modeling beyond the COVID-19 context.
2. **Integration of Natural Births and Deaths:** The current simulation operates on a fixed population size, focusing solely on disease-related mortality. For long-term epidemiological studies or simulations of diseases with lower mortality rates, incorporating natural birth and death rates would add a crucial layer of realism. This would involve introducing mechanisms for age-dependent mortality, as well as birth events that introduce new susceptible individuals into the population, providing a more comprehensive demographic representation over extended simulation periods.
3. **Enhanced Vaccination Modeling:** The current vaccination willingness is responsive to perceived risk, yet the mechanism for vaccine efficacy and re-vaccination can be expanded. Future work could incorporate more granular details such as varying vaccine efficacies against different variants, waning immunity requiring booster doses, and age-stratified vaccination priorities. This would allow for a more nuanced simulation of real-world vaccination campaigns, especially considering the dynamic nature of viral evolution and public health recommendations. The current model’s integration of MVT’s fixed-duration immunity and LOIMOS’s simplified antibody landscape presents an opportunity to develop a more complex, biologically-driven immunity model. This would allow for simulating scenarios where immunity wanes at different rates for different individuals, or where partial immunity offers varied levels of protection against.
4. **Addressing Dataset Scaling for Comparison:** The computational

intensity of the simulation currently limits the size of the simulated population, making direct comparison with much larger real-world datasets challenging. Future efforts could explore advanced techniques for creating highly representative sub-datasets from larger empirical data, potentially leveraging machine learning algorithms. These algorithms could identify key epidemiological patterns and demographic features in vast datasets, allowing for the generation of smaller, statistically similar datasets that are more computationally tractable for direct comparison with the simulated output, thereby enabling more robust validation without needing to run the simulation on the full scale of real-world populations.

5. **Code Parallelization for Optimization:** Given the agent-based nature and hourly iterations of the simulation, computational efficiency is a critical consideration. The current Python implementation, while flexible, could greatly benefit from parallelization techniques. Exploring libraries such as ‘multiprocessing’ or ‘Dask’ for parallelizing operations across provinces or individual agents would significantly reduce simulation run times, especially for larger populations and extended simulation periods. This optimization would enable users to conduct more extensive parameter sweeps and explore a wider range of epidemiological scenarios in a shorter timeframe.
6. **Improved Prediction of Subsequent Peaks:** While the model currently excels at forecasting the primary infection peak, it struggles to consistently predict subsequent epidemic waves. Future work could address this by incorporating more complex, dynamic factors such as the emergence of new viral variants with altered transmissibility or severity. Additionally, integrating adaptive behavioral responses that evolve beyond the immediate aftermath of an initial peak (e.g., changes in risk perception due to prolonged exposure or policy fatigue) could help capture the drivers of multi-wave dynamics.

Bibliography

- [1] Santhosh Raghavan and Krishnan Chandrasekaran. Tools and simulators for membrane computing—a literature review. *Bio-inspired Computing—Theories and Applications, 11th International Conference, BIC-TA 2016, Revised Selected Papers, Part I*, 10049:81–93, 2016.
- [2] Gheorghe Păun and Mario J Pérez-Jiménez. Membrane computing: brief introduction, recent results and applications. *Biosystems*, 85(1):11–22, 2006.
- [3] Giulia Giordano, Franco Blanchini, Raffaele Bruno, Patrizio Colaneri, Alessandro Di Filippo, Angela Di Matteo, and Marta Colaneri. Modelling the covid-19 epidemic and implementation of population-wide interventions in italy. *Nature medicine*, 26(6):855–860, 2020.
- [4] Marcelino Campos, José M Sempere, Juan Carlos Galán, Andrés Moya, Carlos Llorens, C de Los-Angeles, F Baquero-Artigao, Rafael Cantón, and Fernando Baquero. Simulating the impact of non-pharmaceutical interventions limiting transmission in covid-19 epidemics using a membrane computing model. *MicroLife*, 2:uqab011, 2021.
- [5] Fernando Baquero, Marcelino Campos, Carlos Llorens, and José M Sempere. P systems in the time of covid-19. *Journal of Membrane Computing*, 3:246–257, 2021.
- [6] Marcelino Campos, José M Sempere, Juan C Galán, Andrés Moya, Rafael Cantón, Carlos Llorens, and Fernando Baquero. Simulating the efficacy of vaccines on the epidemiological dynamics of sars-cov-2 in a membrane computing model. *MicroLife*, 3:uqac018, 2022.
- [7] Davide Valcamonica, Alberto d’Onofrio, Muhammad Mazhar Fareed, Giuditta Franco, and Claudio Zandron. A dynamic behavior epidemiological model by membrane systems. *Journal of Membrane Computing*, 6:23–38, 2025.

- [8] Sandro Erba. Covid-19 infection inspired diffusion dynamics: Generalization of a membrane-based simulator. Master’s thesis, University of Milano - Bicocca, Department of Informatics, Systems and Communication, Milan, Italy, 2025. Supervisor: Prof. Claudio Zandron; Co-supervisor: Prof. Giuditta Franco.
- [9] Per Block, Marion Hoffman, Isabel J Raabe, Jennifer Beam Dowd, Charles Rahal, Ridhi Kashyap, and Melinda C Mills. Social network-based distancing strategies to flatten the covid-19 curve in a post-lockdown world. *Nature human behaviour*, 4(6):588–596, 2020.
- [10] Mark Jit and Marc Brisson. Modelling the epidemiology of infectious diseases for decision analysis: a primer. *Pharmacoeconomics*, 29:371–386, 2011.
- [11] Mark W Tenforde, Wesley H Self, Manjusha Gaglani, Adit A Ginde, William B Stubblefield, H Keipp Talbot, Jonathan D Casey, David N Hager, Christopher J Lindsell, et al. Effectiveness of mRNA vaccination in preventing COVID-19–associated invasive mechanical ventilation and death – united states, march 2021–january 2022. *MMWR. Morbidity and Mortality Weekly Report*, 71(12):429–435, 2022.
- [12] Li-Lin Liang, Hsu-Sung Kuo, Hsiu J Ho, and Chun-Ying Wu. Covid-19 vaccinations are associated with reduced fatality rates: Evidence from cross-county quasi-experiments. *Journal of Global Health*, 11:05019, 2021.