

М. І. Мосольд, Є. В. Мелешко, М. С. Якименко, Д. В. Бащенко

Центральноукраїнський національний технічний університет, Кропивницький, Україна

ПРОГРАМНЕ ІМІТАЦІЙНЕ МОДЕЛЮВАННЯ ЕПІДЕМІЙ ЗАСОБАМИ МОВИ PYTHON ТА ПАКЕТУ ІНСТРУМЕНТІВ EPIMODEL ДЛЯ МОВИ R

Анотація. Метою даної роботи було створення програмного забезпечення для імітаційного моделювання епідемій та дослідження можливостей мови Python і пакету інструментів EpiModel для мови R при вирішенні даної задачі. Моделювання процесів поширення та прогнозування наслідків епідемій як інфекційних, так і інформаційно-психологічних є важливою науково-практичною задачею у наш час. Існують підходи та поради щодо моделювання епідемій, а також інструменти, за допомогою яких можна спростити шлях до досягнення потрібної мети. Епідемічні моделі поділяють на два основні види: стохастична модель та детермінована модель. Найбільш відомі математичні моделі епідемій – SI, SIR та SIS. Було розроблено програмну імітаційну модель епідемій засобами мови Python та пакету інструментів EpiModel для мови R. Тестувалася модель на прикладі поширення інфекційних захворювань, хоча також може бути використана для моделювання поширення інформаційно-психологічних впливів. Було досліджено можливості мови програмування Python та пакету інструментів EpiModel мови R для імітаційного моделювання епідемій. Дослідження показало, що дані мова програмування та пакет інструментів дають усі необхідні засоби для розробки програмної моделі епідемій. Python дозволяє швидко розробляти програми для аналізу даних, Qt Designer дає можливість створити зручний графічний інтерфейс користувача для застосунків на мові Python, а пакет інструментів EpiModel дозволяє реалізувати найбільш поширені математичні моделі епідемій, зокрема, SI, SIR та SIS. Структура соціальної мережі у даній моделі не генерувалася, а завантажувалася з частини віртуальної соціальної мережі, що прискорило розробку, а мова програмування Python дозволила за допомогою своїх інструментів та бібліотек створити парсер та взаємодію з API потрібного веб-ресурсу. Також було проведено серію експериментів з різними параметрами моделі для перевірки її працездатності та отримано очікувані результати моделювання.

Ключові слова: програмна модель, імітаційне моделювання, епідемії, вірусна епідемія, інформаційна епідемія, Python, EpiModel

Вступ

Моделювання процесів поширення та прогнозування наслідків епідемій як інфекційних, так і інформаційно-психологічних є важливою науково-практичною задачею у наш час.

Існують підходи та поради щодо моделювання епідемій, а також інструменти, за допомогою яких можна спростити шлях до досягнення потрібної мети. Епідемічні моделі поділяють на два основні види: стохастична модель та детермінована модель.

Стохастична модель – це математична модель, в якій параметри, характеристики стану об'єкта моделювання і умови його функціонування, представлені випадковими величинами і пов'язані випадковими, непостійними залежностями; або вихідна інформація також представлена стохастичними величинами [1].

Отже, характеристики стану об'єкта в моделі визначаються не однозначно, а через закони розподілу їх ймовірностей. Моделюються стохастичні процеси в теорії масового обслуговування, а застосовуються в мережевому плануванні та управлінні і в інших областях. При побудові такої моделі найчастіше застосовуються методи регресійного і кореляційного аналізу.

Детерміновані моделі, як правило, призначені для поглиблення в певний основний механізм або природний процес. Вони відрізняються від статистичних моделей (наприклад, лінійної регресії), метою яких є практично оцінити відносини між змінними. Детермінована модель розглядається як корисне наближення до дійсності, яку простіше побудувати і інтерпретувати, ніж стохастичну. Проте, такі моделі

можуть бути надзвичайно складними через велику кількість вхідних і вихідних даних, і, отже, часто незворотними; фіксований єдиний набір вихідних даних може бути отриманий за допомогою декількох наборів вхідних [2].

Таким чином, використання надійних параметрів і невизначеності моделі є вирішальними факторами, навіть більшою мірою, ніж для стандартних статистичних моделей.

У математичному моделюванні, детерміновані моделі не містять випадкових величин і ступеню випадковості, і в основному складаються з рівнянь, наприклад, диференціальних. Ці розрахунки мають відомі вхідні дані і вони стають наслідком унікального набору вихідних, на протилежність стохастичному моделюванню, яке містить в собі випадкові змінні.

Детерміновані моделі використовують в наукових дослідженнях, які можна знайти в різних дослідженнях: в сфері демографічних досліджень, розвитку клімату, забруднення навколишнього середовища, а також в інших областях, як інженерія, хімія і ведення політики.

Детерміновані моделі отримали увагу в літературі по статистиці під загальною темою комп'ютерних експериментів. Комп'ютерні експерименти моделюють складні системи, що вимагають певної кількості вхідних даних. Використання стохастичної системи набагато дешевше, але є неточним і спрощеним способом.

Метою даної роботи було створення програмного забезпечення для імітаційного моделювання епідемій та дослідження можливостей мови Python і пакету інструментів EpiModel для мови R при вирішенні даної задачі.

Основна частина

У якості існуючого інструменту для моделювання епідемій у даній роботі було розглянуто програмне забезпечення EpiModel.

EpiModel – це набір програмних інструментів для моделювання математичних моделей, динаміки інфекційних захворювань. Класи епідемічних моделей включають детерміновані компартментні моделі, стохастичні моделі індивідуального контакту та стохастичні моделі мережі. Мережеві моделі використовують надійні статистичні методи моделей випадкових графіків експоненціальних сімей (ERGM) із набору програм Statnet. Стандартні шаблони для моделювання епідемій включають такі типи захворювань: SI, SIR та SIS. EpiModel має простий API для розширення цих шаблонів для вирішення нових цілей наукових досліджень [3].

Для моделювання поширення інфекційних захворювань може використовуватися класична модель SIR. Населення в моделі ділиться на три частини в залежності від статусу: сприйнятливий до захворювання, заражений і одужавший. Ця модель створена методом системної динаміки. Щоб відображати дійсність, модель повинна ґрунтуватися на реальних властивостях конкретної хвороби і враховувати зміни системи під зовнішніми впливами.

Для розробки програмної імітаційної моделі було вирішено обрати інтерпретовану мову програмування Python. У зв'язку із стрімким розвитком персональної обчислювальної техніки, відбувається поступова зміна вимог до мов програмування. Все більшу роль починають грати інтерпретовані мови, оскільки зростає потужність персональних комп'ютерів починає забезпечувати достатню швидкість виконання інтерпретованих програм. А чи не єдиною суттєвою перевагою компільованих мов програмування є створений ними високошвидкісний код. Інтерпретована мова є найбільш простим і гнучким інструментом програмування, що має дуже високий рівень портативності розроблених програмних засобів. Python – це універсальна сучасна мова програмування високого рівня, до переваг якої відносять високу продуктивність програмних рішень і структурований код. Ядро має дуже зручну структуру, а широкий перелік вбудованих бібліотек дозволяє застосовувати значний набір корисних функцій і можливостей. Python може підтримувати широкий перелік стилів розробки застосунків, в тому числі, дуже зручний для роботи з об'єктно-орієнтованою та функціональною парадигмами програмування [4].

Для створення графічного інтерфейсу користувача було використано Qt Designer [5]. Він дозволяє створювати графічні інтерфейси за допомогою ряду наступних інструментів. Існує панель інструментів, в якій доступні для використання багато різноманітних елементів інтерфейсу. Кожен віджет має свій набір властивостей, що визначається відповідним йому класом бібліотеки Qt. Характерною особливістю Qt Designer є підтримка візуального редагування сигналів і слотів. Qt Designer дозволяє швидко спроектувати віджети і діалоги, використовуючи екран-

ні форми з використанням тих же віджетів, які будуть використовуватися в застосунку. Форми, створені з Qt Designer, є повністю функціональними, а також можуть бути переглянуті в режимі реального часу. Розроблений інтерфейс зберігається в файл з розширенням ui, який підключається до створюваної програми за допомогою спеціальних методів бібліотеки Qt. Цей файл має xml-формат, і може, в разі необхідності, редагуватися в будь-якому текстовому редакторі.

Не можливо уявити великі сучасні застосунки без зручного та дружнього до звичайного користувача інтерфейсу. Хороший користувацький інтерфейс охарактеризувати нескладно: такий інтерфейс простий у використанні. Для створення графічного інтерфейсу користувача на мові програмування Python, найбільш доцільним є використання Qt Designer в парі з PyQt 5. А для того, щоб швидко та гарно зображувати графіки, кращим вибором є PyQtGraph.

Для зберігання даних моделювання та прогнозування наслідків епідемії можна використовувати SQLite. Вона не вимагає великої потужності персонального комп'ютера.

Для проведення наукових розрахунків з використанням багатовимірних масивів обрано стандартний пакет NumPy. Розроблена система моделювання та прогнозування наслідків епідемії складається з двох програмних застосунків з графічними інтерфейсами. Перший застосунок має такі дві основні функції:

а) завантаження у базу даних структури зв'язків між користувачами деякої частини існуючої віртуальної соціальної мережі (для одержання реалістичної структури соціальної мережі);

б) підготовка бази даних до моделювання поширення епідемії по завантаженому графу віртуальної соціальної мережі.

Другий застосунок, оперуючи багатьма факторами, моделює епідемію, прогнозує її наслідки та відображає статистичну інформацію про хід моделювання та стан користувачів соціальної мережі. В подальшому ці застосунки можуть покращуватись за рахунок закладеної в них масштабованості, завдяки чому вони зможуть оперувати ще більшою кількістю факторів, які можуть впливати на розвиток епідемії.

Для зберігання даних розробленої системи застосунків використовується SQLite 3. SQLite – це вбудована кросплатформна СУБД, яка підтримує досить повний набір команд SQL і має досить зручний програмний інтерфейс для мови програмування Python [6]. Вона є дуже надійною та простою у використанні. SQLite ідеально підходить для використання в даній системі застосунків, тому що ця СУБД швидка, не вимагає спеціальної установки, а база даних зберігається на диску у вигляді одного файлу.

Щоб переглянути структуру та дані, що записані до бази даних, можна використовувати безкоштовний застосунок DB Browser. Цей застосунок являє собою візуальний інструмент з відкритим сирцевим кодом для створення, розробки і редагування файлів баз даних, сумісних з SQLite.

База даних, що створюється та використовується під час виконання системи моделювання та прогнозування наслідків епідемії, має 5 таблиць:

а) users – основна таблиця, що зберігає всіх опрацьованих учасників соціальної мережі;

б) before_modeling – учасники соціальної мережі потрапляють до цієї таблиці під час процедури підготовки бази даних до моделювання. Також виконує функцію резервного сховища даних, яке можна повторно використовувати, щоб моделювати декілька епідемій без повторного запуску процедури підготовки бази даних;

в) healthy – до цієї таблиці вносяться здорові учасники соцмережі під час моделювання епідемії;

г) not_healthy – до цієї таблиці вносяться хворі учасники соцмережі під час моделювання епідемії;

г) dead – до цієї таблиці вносяться мертві учасники соціальної мережі під час моделювання епідемії.

Наповнення бази даних відбувається через запити до серверів віртуальної соціальної мережі через API відповідного веб-ресурсу.

У деяких користувачів віртуальних соціальних мереж число друзів набагато перевищує число Данбара, що являє собою обмеження на кількість постійних соціальних зв'язків, які людина може підтримувати [7]. Підтримка таких зв'язків передбачає знання особливостей конкретного індивіда, зв'язок з яким необхідно підтримувати, його характеру, а також соціального стану, що вимагає наявності певних розумових здібностей і витрати значних інтелектуальних ресурсів. За різними оцінками, число Данбара в людських співтовариствах лежить в діапазоні від 100 до 230, найчастіше умовно число приймається рівним 150. Тому щоб мати в базі даних більш рівномірні дані, розроблений застосунок ігнорує людей з надто великим та надто малим числом соціальних зв'язків. Вікно налаштувань застосунку для підготовки бази даних можна побачити на рис. 1.

Налаштування

Список початкових id, з якого почне наповнюватися БД:
(розділити пробілом)

132646860 243109925 245135939

Нижня межа друзів для 1 людини:

10

Верхня межа друзів для 1 людини

500

Відображати лише успішні записи до БД

Кількість ітерацій для перевірки друзів користувачів:

3

Обмеження кількості записів для моделювання:

1000000

Для моделювання відбирати користувачів лише з України

OK Cancel

Рис. 1. Вікно налаштувань для підготовки бази даних розробленого застосунку

Наступним кроком, після наповнення бази даних, є її підготовка до моделювання (приклад зображено на рис. 2). Інформація фільтрується та структурується в потрібних таблицях. Кожний учасник мережі має зв'язки з іншими учасниками, але не всі вони містяться в базі даних, тому відсутні зв'язки видаляються. Такий процес видалення зв'язків займає досить багато часу, в залежності від розмірів бази даних. На виході, перед моделюванням, база даних не має учасників мережі без зв'язків з іншими учасниками.

Продуктивність масової вставки SQLite може варіюватися від 85 вставок в секунду до більш 96000 вставок в секунду. Виявляється, продуктивність SQLite може значно відрізнятись (як для масових вставок, так і для вибірок) в залежності від того, як налаштована база даних і як використовується API. При найгіршому сценарії операції занадто повільні, тому що SQL буде скомпільовано в код VDBE для кожної вставки, і кожна вставка буде відбуватися у своїй власній транзакції. За замовчуванням SQLite буде оцінювати кожен оператор INSERT / UPDATE в рамках унікальної транзакції. Якщо виконується велика кількість вставок, рекомендується вкласти операцію в транзакцію. Використання транзакції є величезним поліпшенням, але повторна компіляція оператора SQL для кожної вставки не має сенсу, якщо використовується один і той же SQL багаторазово. Скомпільовавши оператор SQL один раз, а потім зв'язавши параметри з цим оператором, використовуючи `sqlite3_bind_text`, можна подвоїти продуктивність. Графічний інтерфейс створено через застосунок Qt Designer. Шаблон зберігається в *.ui. Звичайно, можна використовувати ui-файли безпосередньо з Python-коду, проте є й інші шляхи. Код ui-файлу конвертується в Python-файл, який потім імпортується і використовується як звичайний модуль. В цьому може допомогти інструмент розробника Pyuic5, який полегшує перетворення файлів *.ui PyQt5 Designer у файли *.py [8].

Налаштування

Обрати main.db

Нульовий пацієнт обирається випадково

Кількість нульових пацієнтів, що буде обрано випадково:

1

Введіть id користувача, з якого почнеться моделювання:

3973283 9259454 12373354

Тривалість хвороби

Мінімальна кількість ітерацій: 6

Максимальна кількість ітерацій: 16

Кількість рядків в топ (детальна інформація):

10

Ймовірність передати хворобу:

5

Ймовірність летального випадку:

15

Кількість ітерацій моделювання:

Обмежити кількість ітерацій

45

Пауза між кроками моделювання:

0.5

OK Cancel

Рис. 2. Налаштування розробленої імітаційної моделі для експерименту №1

Для відображення статистичної інформації роботи застосунка, що наповнює базу даних, створено клас MyErrorCounter, що рахує час виконання застосунка та кількість записів в БД, кількість деактивованих та закритих профілів, кількість колізій, перевищень кількості друзів, а за запитом повертає всю необхідну інформацію. Також застосунок для моделювання та прогнозування наслідків епідемії зображує на графіку кількість здорових, хворих, одужавших та померлих учасників соціальної мережі на кожному кроці моделювання. Приклади такого графіку можна побачити на рис. 3, 5, 7.

Щоб дослідити як система імітаційного моделювання епідемії працює при різних початкових параметрах, було проведено серію експериментів, результати яких наведено на рис. 2-7.

З запропонованих прикладів виконання застосунку моделювання та прогнозування наслідків епідемії видно, що все працює так, як було задумано. В залежності від різних наборів початкових параметрів, система видає різні результати, що відповідають очікуваним результатам. Також були протестовані всі налаштування даного застосунку та наглядно показана зручність розробленого графічного інтерфейсу.

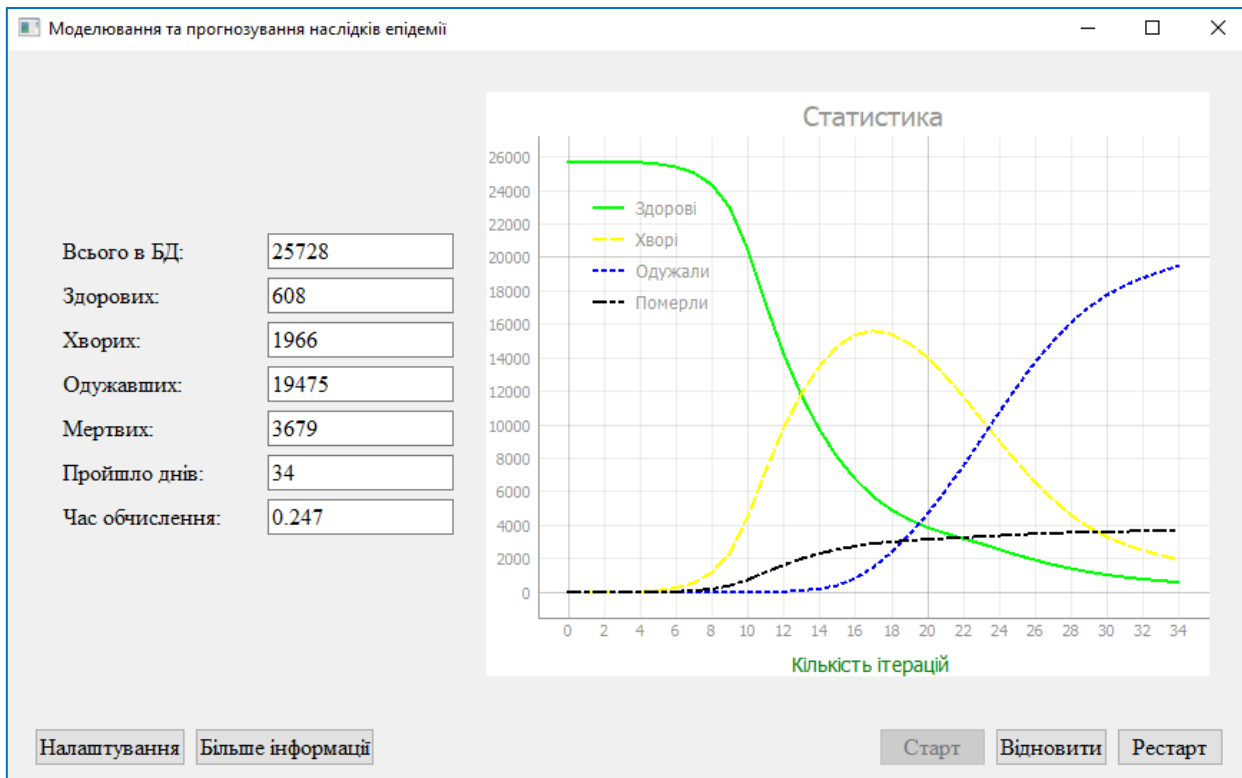


Рис. 3. Результат проведення експерименту №1 у розробленій програмній імітаційній моделі

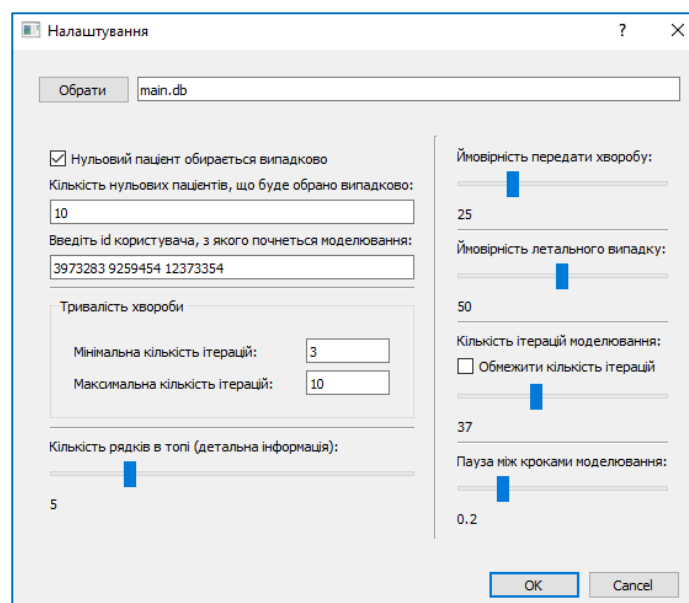


Рис. 4. Налаштування розробленої імітаційної моделі для експерименту №2

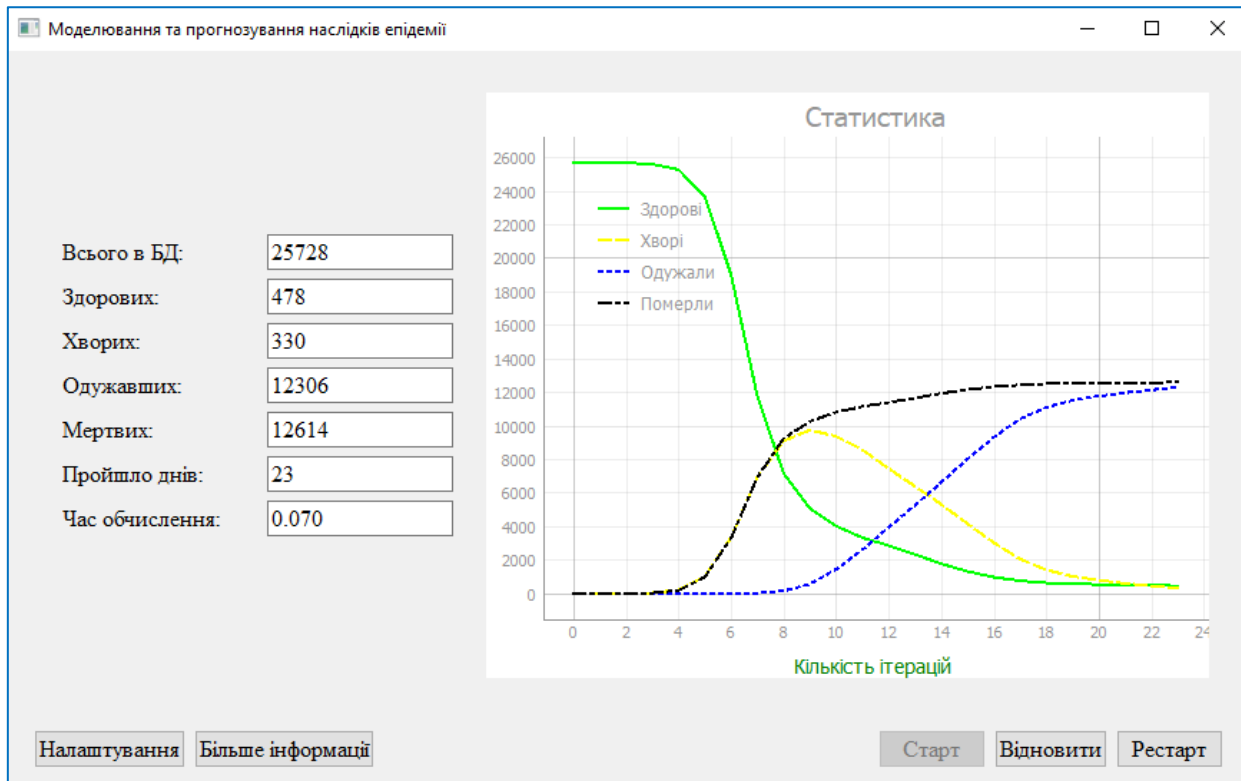


Рис. 5. Результат проведення експерименту №2 у розробленій програмній імітаційній моделі

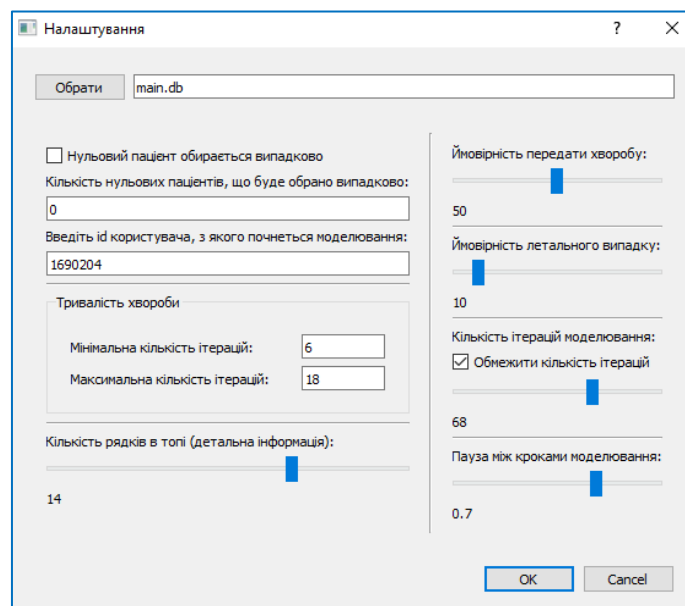


Рис. 6. Налаштування розробленої імітаційної моделі для експерименту №3

Висновки

Було розроблено програмну імітаційну модель епідемії засобами мови Python та пакету інструментів EpiModel для мови R. Тестувалася модель на прикладі поширення інфекційних захворювань, хоча також може бути використана для моделювання поширення інформаційно-психологічних впливів.

Було досліджено можливості мови програмування Python та пакету інструментів EpiModel для мови R для імітаційного моделювання епідемії. Дослідження показало, що дані мова програмування та пакет інструментів

дають усі необхідні засоби для розробки програмної моделі епідемії. Python дозволяє швидко розробляти програми для аналізу даних, Qt Designer дає можливість розробити зручний графічний інтерфейс користувача для застосунків на мові Python, а пакет інструментів EpiModel дає можливість реалізувати найбільш поширені математичні моделі епідемії, зокрема, SI, SIR та SIS. Структура соціальної мережі у даній моделі не генерувалася, а завантажувалася з частини віртуальної соціальної мережі, що прискорило розробку, а мова програмування Python дозволила за допомогою своїх інструментів та бібліотек створити парсер та взаємодію з

API потрібного веб-ресурсу. Також було проведено серію експериментів з різними параметрами моделі для

перевірки її працездатності та отримано очікувані результати моделювання.

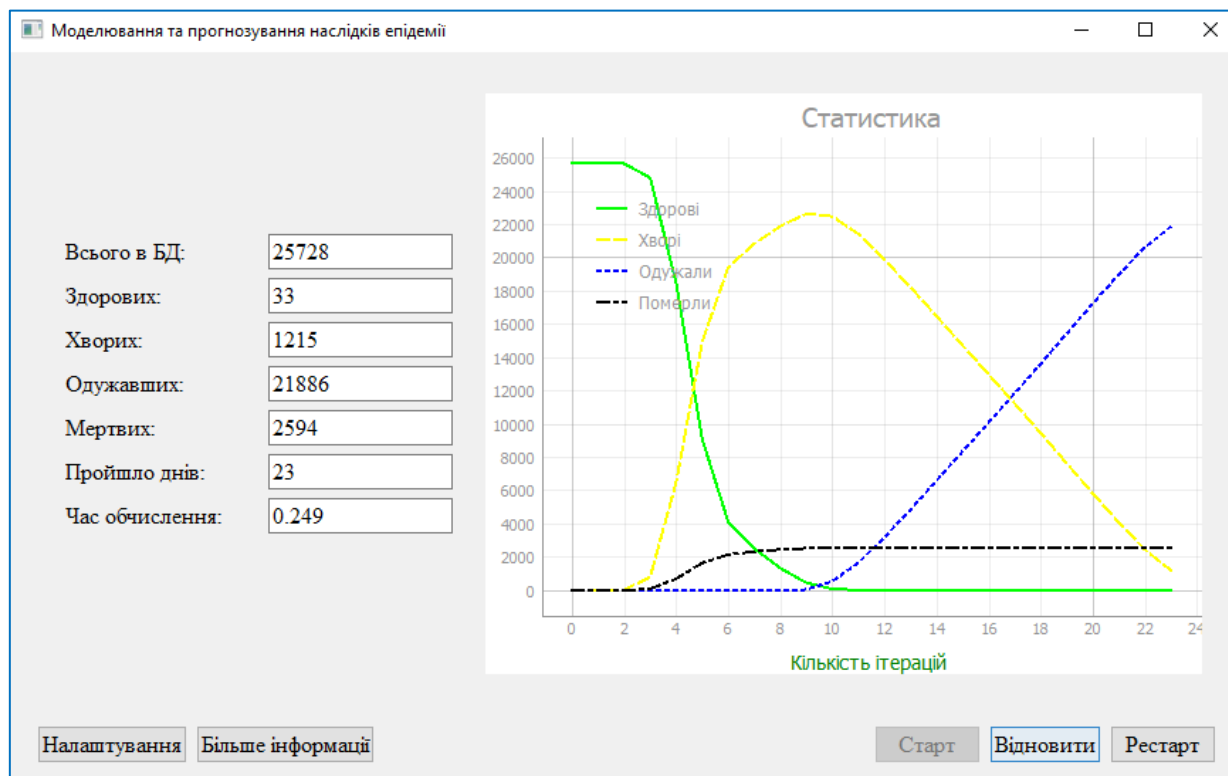


Рис. 7. Результат проведення експерименту №3 у розробленій програмній імітаційній моделі

СПИСОК ЛІТЕРАТУРИ

1. Brzezniak Z. "Basic Stochastic Processes: A Course Through Exercises (Springer Undergraduate Mathematics Series)", 2000, с. 236.
2. David P., Raftery A.E., "Inference for Deterministic Simulation Models: The Bayesian Melding Approach", Journal of the American Statistical Association 95, 2000, с. 1244–1255.
3. EpiModel [Електронний ресурс] – Режим доступу до ресурсу: <https://www.epimodel.org/>
4. Budd T., "An Introduction to Object-Oriented Programming", Addison-Wesley; Subsequent edition, 1996, 452 p.
5. Blanchette J., Summerfield M., "C++ GUI Programming With Qt 4", Prentice Hall Ptr, 2006, 537 p.
6. SQLite [Електронний ресурс] – Режим доступу до ресурсу: <http://sqlite.org/>
7. Число Данбара [Електронний ресурс] – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Число_Данбара
8. Інструмент розробника PyQt5 [Електронний ресурс] – Режим доступу до ресурсу: <https://github.com/Abdelatif/Pyuic5-Tool>

Received (Надійшла) 12.09.2022

Accepted for publication (Прийнята до друку) 23.11.2022

Computer simulation model of epidemics using the python language and the epimodel tool package for the r language

M. Mosold, Ye. Meleshko, M. Yakymenko, D. Bashchenko

Abstract. The goal of this work was to create software for simulating epidemics and to research the capabilities of the Python language and the EpiModel toolkit for the R language in solving this problem. Modeling the processes of spreading and forecasting the consequences of epidemics, both infectious and informational and psychological, is an important scientific and practical task in our time. There are approaches and tips for modeling epidemics, as well as tools that can simplify the path to achieving the desired goal. Epidemic models are divided into two main types: stochastic model and deterministic model. The most famous mathematical models of epidemics are SI, SIR and SIS. A computer simulation model of epidemics was developed using the Python language and the EpiModel tool package for the R language. The model was tested on the example of the spread of infectious diseases, although it can also be used to model the spread of informational and psychological influences. The possibilities of the Python programming language and the R language EpiModel toolkit for simulating epidemics were investigated. The research showed that the given programming language and toolkit provide all the necessary tools for developing a computer model of epidemics. Python allows you to quickly develop programs for data analysis, Qt Designer allows you to create a convenient graphical user interface for Python applications, and the EpiModel toolkit allows you to implement the most common mathematical models of epidemics, in particular, SI, SIR and SIS. The structure of the social network in this model was not generated, but loaded from a part of the virtual social network, which accelerated the development, and the Python programming language allowed using its tools and libraries to create a parser and interaction with the API of the desired web-resource. A series of experiments with various parameters of the model was also conducted to check its efficiency and expected simulation results were obtained.

Keywords: computer model, simulation modeling, epidemics, virus epidemic, information epidemic, Python, EpiModel.