

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Кваліфікаційна наукова
праця на правах рукопису

Ковалишин Олег Степанович



УДК 004.89 + 004.932

ДИСЕРТАЦІЯ
ПРОДУКЦІЙНА СИСТЕМА НЕЧІТКОЇ ЛОГІКИ ДЛЯ ОПТИМІЗАЦІЇ
ПЛАНІВ ВІДНОВЛЮВАЛЬНОЇ ТЕРАПІЇ

05.13.23 – Системи та засоби штучного інтелекту

Подається на здобуття наукового ступеня кандидата технічних наук

Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело


_____ О.С. Ковалишин

Науковий керівник – Ткаченко Роман Олексійович, доктор технічних наук,
професор

Львів – 2018

АНОТАЦІЯ

Ковалишин О.С. Продукційна система нечіткої логіки для оптимізації планів відновлюваної терапії. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня кандидата технічних наук за спеціальністю 05.13.23 – системи та засоби штучного інтелекту. – Національний університет «Львівська політехніка», Міністерство освіти і науки України, Львів, 2019.

Дисертаційна робота присвячена розв’язанню важливого наукового завдання – розробленню методів і засобів продукційної системи нечіткої логіки для багатокритеріальної оптимізації план-графіків відновлювальної терапії.

В першому розділі обґрунтовано необхідність складання ефективних графіків відновлювальної терапії, дано характеристику задачам побудови графіків, проведено аналіз сучасних методів їх вирішення та перспектив застосування. Означено причини багатокритеріальності задачі побудови план-графіків, зокрема відновлювальної терапії.

Багатокритеріальність задачі побудови план-графіків відновлювальної терапії впливає з необхідності виконання медичних, методичних та організаційних обмежень і вимог, наявності множини індивідуальних планів реабілітації, а також врахування побажань пацієнтів та медичного персоналу лікувальних установ.

Аналіз дає змогу стверджувати, що для вирішення задач побудови план-графіків розроблено велику кількість методів дискретної оптимізації. Особливе місце серед них займають методи, які можна розділити на дві групи: точні та наближені.

Дослідження свідчить, що загальним недоліком точних методів є те, що вони засновані на ітераційній процедурі пошуку або поліпшення деякого значення. Відповідно, отримане рішення має пряму залежність від деякого початкового наближення і природно виникає проблема вибору його значення.

Наближені методи, як правило, ґрунтуються використання різноманітних евристик або евристичних алгоритмів. Формування план-графіків з використанням евристичних правил дозволяє прискорити пошук "найкращого" рішення, та використання таких алгоритмів в більшості випадків гарантує знаходження лише наближеного до оптимального розв'язку (досягнення локального екстремуму). Перевагою евристичних методів є зручність їхньої реалізації на ЕОМ навіть за умови розв'язання громіздких задач. Основний недолік евристичних методів полягає в складності оцінки близькості одержаних графіків до оптимального. Крім того, для кожного з описаних евристичних алгоритмів існують задачі, для яких застосування означених алгоритмів дає погані результати.

Одним з найбільш перспективних, судячи по літературних даних, підходів для багатокритеріальної оптимізації план-графіків медичних закладів, що дає можливість уникнути локальних екстремумів та отримати множину глобальних субоптимальних рішень, є використання генетичних алгоритмів.

У **другому розділі** розкрито засади функціонування реабілітаційно-оздоровчих установ, дано характеристику індивідуальним планам реабілітації пацієнтів, формалізовано процес побудови графіку клініки, розроблено метод побудови опорного план-графіку медичних установ, на основі адаптації абстрактної моделі автомата Мілі.

Аналіз процесів функціонування реабілітаційно-оздоровчих установ свідчить, що одним з ключових етапів відновлювальної терапії є складання індивідуального плану відновлювального лікування хворого, ключовим елементом якого є перелік необхідних заходів та процедур, спрямованих на відновлення та компенсацію порушених або втрачених функцій організму, а також подальший розвиток здібностей пацієнта і поліпшення стану його життєдіяльності в цілому. Розробка індивідуальних планів реабілітації разом з їх впровадженням та наступний контролем і коректуванням вимагають значних зусиль, взаємного зв'язку виконавців, відповідальних за результати роботи на кожному етапі реабілітації. Тому доцільним є використання в процесі побудови автоматизованих інтелектуальних методів та моделей.

Беручи до уваги необхідність узгодження індивідуальних планів реабілітації пацієнтів між собою, графіків роботи процедурних кабінетів, лікарів, та використання медичного обладнання, запропоновано підхід до вирішення задачі багатокритеріальної оптимізації план-графіку роботи клініки, що складається з трьох основних етапів: побудова опорного план-графіку, що задовольняє жорстким обмеженням, оцінка якості план-графіку, послідовна оптимізація план-графіку з метою підвищення результатів оцінки.

Для побудови опорного план-графіку вперше розроблено універсальний незалежний від типу процедур метод, що забезпечує лінійну залежність часу вирішення відносно об'єму задачі. Він ґрунтується на визначенні пріоритетності включення процедур до план-графіку та дозволяє врахувати вимоги індивідуального плану реабілітації, наявність необхідного кваліфікованого персоналу для проведення процедур, доступність медичного інвентаря тощо.

Для роботи методу визначено чинники, що впливають на пріоритетність процедур. Розроблено механізми аналізу реалістичності побудови план-графіку за умови наявності певних множин планів реабілітації, засоби вирішення конфліктних ситуацій при включенні процедур в план-графік, а також засоби послаблення певних жорстких вимог при неможливості побудови план-графіку.

Метод базується на використанні розробленої в дисертаційній роботі функціональної моделі скінченного автомата Мілі, що дозволяє найточніше врахувати специфіку досліджуваної предметної області.

У **третьому розділі** проведено аналіз методик, за якими здійснюють порівняння різних варіантів план-графіків, адаптовано продукційний алгоритм нечіткої логіки для інтегральної оцінки план-графіків, проведено їх багатокритеріальну оптимізацію з використанням генетичного алгоритму.

Вперше розроблено метод побудови план-графіків в режимі реального часу функціонування медичних установ, на основі використання нечіткої логіки, який забезпечує врахування побажання пацієнтів в умовах жорстких і м'яких обмежень план-графіків роботи процедурних кабінетів лікувальних установ, що дозволило відмовитися від затратного в часі повного перебору можливих варіантів. Для цього

розроблено контролер нечіткої логіки на основі системи T-Controller, що уможливив трансформування лінгвістичних даних, отриманих в результаті збору відгуків пацієнтів та медичного персоналу, у кількісні змінні, провести оцінку відгуків про якість роботи клініки та встановити рівень їх задоволеності від наданих лікувальних послуг.

Для здійснення багатокритеріальної оптимізації план-графіків відновлювальної терапії розроблено метод на основі генетичних алгоритмів. Запропоновано спосіб представлення план-графіку функціонування клініки у вигляді хромосоми, що забезпечує підвищену швидкодію алгоритму за рахунок усунення необхідності кодування та декодування кожної особини популяції, метод генерації початкової популяції графіків на основі дослідження проведеного в розділі 2, а також механізм селекції хромосом для їх подальшої оптимізації. Проведено оптимізацію план-графіків відновлювальної терапії без відриву від середовища їх проведення – медичного закладу, з врахуванням вимог на процес лікування як жорстких обмежень, які визначаються планом лікування та забезпеченістю медичним обладнанням і персоналом, так і м'яких обмежень, що залежать від побажань пацієнтів.

У **четвертому розділі** розроблено загальну архітектуру, запропоновано принцип функціонування та реалізацію програмного комплексу оптимізації план-графіків відновлюваної терапії, охарактеризовано умови роботи з графічним користувацьким інтерфейсом. Наведено алгоритм його роботи в процесі експлуатації користувачами та розкрито особливості програмної реалізації, описано порядок функціонування, вимоги до комп'ютерних засобів та технічні характеристики.

Розроблено програмний комплекс, в якому використано архітектуру “тонкого клієнта”. Завдяки цьому забезпечується незалежність від апаратної архітектури клієнтів та їх операційних систем, простота додавання нового функціоналу, висока безпека та високий рівень стійкості проти відмов. Запропонований програмний комплекс є основою інформаційної технології автоматизованої побудови та оптимізації графіків відновлювальної

терапії, володіє адаптивним інтерфейсом, віддаленою базою даних та є багатокористувацькою програмною системою.

Застосування трирівневої архітектури забезпечує автоматичне централізоване оновлення програмного комплексу, незалежне конфігурування на кожному з рівнів архітектури, централізований контроль за рахунок використання механізмів авторизації та автентифікації.

Розроблений програмний комплекс надає можливість розгортання на більшості сучасних платформ та операційних систем за рахунок використання мови програмування високого рівня – Java.

Система надає дружній інтерфейс роботи з використанням низки мобільних та стаціонарних апаратних пристроїв. Користування програмним комплексом не вимагає спеціалізованих знань в галузі інформаційних технологій.

Ключові слова: лікувальні заклади, план-графік, контролер нечіткої логіки, генетичні алгоритми, оптимізація планів відновлювальної терапії, програмний комплекс.

ABSTRACT

Kovalyshyn O.S. Production fuzzy logic system for medical rehabilitation plans optimization. - Qualifying scientific work on the rights of manuscripts.

Thesis for obtaining the scientific degree of the candidate of technical sciences in the specialty 05.13.23 - systems and means of artificial intelligence. - National University "Lviv Polytechnic", Ministry of Education and Science of Ukraine, Lviv, 2019.

The dissertation is devoted to solving an important scientific problem - the development of methods and means of fuzzy logic production system for multicriteria optimization of rehabilitation therapy schedules.

The first chapter substantiates the necessity of compiling effective schedules of rehabilitation therapy schedules, describes the issues of scheduling tasks, analyzes modern methods of their solution and prospects of application. The reasons of the

multicraterality of the problem of scheduling, in particular, rehabilitation therapy, are indicated.

The multicraterality of the task of constructing rehabilitation therapy schedules follows from the necessity of fulfilling medical, methodological and organizational constraints and requirements, the availability of a plurality of individual rehabilitation plans, as well as need in taking into account the wishes of patients and medical staff of medical institutions.

The analysis allows us to assert that a large number of discrete optimization methods have been developed to solve the problems of constructing graphs. A special place among them is the methods that can be divided into two groups: precise and approximate.

The study shows that the general disadvantage of precise methods is that they are based on the iterative search procedure or improvement of some value. Thus, solution obtained has a direct dependence on some initial approximation and naturally there is a problem of choosing its value.

Approximate methods, in common, are based on the use of various heuristics or heuristic algorithms. Generating schedules using heuristic rules can accelerate the search for the "best" solution, and the use of such algorithms in most cases ensures that only an optimal solution is reached (the achievement of a local extremum). The advantage of heuristic methods is the convenience of their implementation on a computer, even if the bulky tasks are solved. The main disadvantage of heuristic methods is the complexity of assessing the proximity of the resulting graphs to the optimal. In addition, for each of the described heuristic algorithms there are problems for which the application of the algorithms gives poor results.

One of the most promising, judging by the literature data, approaches for multicriteria optimization of rehabilitation plan-schedules, which makes it possible to avoid local extremes and obtain a plurality of global suboptimal solutions, is the use of genetic algorithms.

The second chapter describes functioning principles of rehabilitation and health institutions, describes the individual plans of rehabilitation of patients, formalizes the

process of constructing the schedule of the clinic, developed the method of constructing a reference schedule of medical institutions, based on the adaptation of the abstract model of the Milly machine.

The analysis of the processes of the functioning of rehabilitation and health institutions shows that one of the key stages of rehabilitation therapy is the preparation of an individual plan for restorative treatment of the patient, a key element of which is a list of necessary measures and procedures aimed at restoring and compensating for the violated or lost functions of the body, as well as further development of abilities the patient and improve the state of his life as a whole. The development of individual plans for rehabilitation, along with their implementation and subsequent control and adjustment, requires considerable effort, mutual communication of performers responsible for the results of work at each stage of rehabilitation. Therefore, it is expedient to use in the process of constructing automated intelligent methods and models.

Taking into account the necessity of coordinating individual plans of rehabilitation of patients among themselves, schedules of work of procedural offices, doctors, and use of medical equipment, an approach to solving the problem of multicriteria optimization of the work schedule of the clinic, which consists of three main stages: the construction of a reference schedule, which satisfies the strict constraints, the assessment of the quality of the schedule, the consistent optimization of the schedule to improve the evaluation results.

In order to construct a reference plan, a universal independent type-of-procedure method is developed. It provides a linear dependence of the decision time with respect to the volume of the task. It is based on the prioritization of the inclusion of procedures in the schedule and allows for the consideration of the requirements of an individual rehabilitation plan, the availability of the necessary qualified personnel for procedures, availability of medical equipment, etc.

The method determines factors that influence the priority of procedures. Mechanisms of analysis of the feasibility of plan-schedule construction are developed, provided that there are certain sets of rehabilitation plans, means of resolving conflict

situations when procedures are included in the schedule, as well as means of relaxing certain strict requirements in case of impossibility of scheduling.

The method is based on the use of the developed functional model of a Finite milly automata, developed in the dissertation, which allows the most precise consideration of the specifics of the subject domain.

In the third section, an analysis of the methods used to compare the different variants of the schedules was performed, production algorithm of fuzzy logic for the integral estimation of the schedules quality was adapted, their multi-criteria optimization using the genetic algorithm was conducted.

For the first time, a method for constructing schedules in real-time operation of medical institutions was developed, based on the use of fuzzy logic, which takes into account the wishes of patients in the conditions of hard and soft constraints of work schedules of procedural offices of medical institutions, which allowed to abandon time-consuming a complete overview of possible options. For this purpose, a fuzzy logic controller based on the T-Controller system was developed, which made it possible to transform the linguistic data obtained from the collection of patient and medical personnel responses into quantitative variables, to evaluate the reviews of the quality of the clinic's work and to determine their level of satisfaction with the provided medical services.

For the implementation of multicriteria optimization of regimen therapy schedules, a method based on genetic algorithms has been developed. A method for presenting a schedule for the functioning of the clinic in the form of a chromosome is provided, which enables increased algorithm performance by eliminating the need for coding and decoding each individual population, the method of generating the initial population of the graphs based on the study conducted in section 2, as well as the mechanism of selection of chromosomes for their further optimization. The optimization of rehabilitation therapy schedules without interruption from the environment of their conduction - the medical institution, taking into account the requirements for the treatment process as strict restrictions, which are determined by the plan of treatment and

the provision of medical equipment and personnel, and soft restrictions – the wishes of patients. .

In the fourth section, the general architecture was developed, the principle of functioning and implementation of the program of optimization of the schedules of regenerative therapy was proposed, the conditions of work with the graphical user interface were described. An algorithm of its work is presented in the process of exploitation of users and the features of program realization are disclosed, the order of functioning, requirements to computer facilities and technical characteristics are described.

The software complex, which uses the "thin client" architecture, was developed. This ensures independence from the hardware architecture of clients and their operating systems, the ease of adding a new feature, high security and high level of failure resistance. The proposed software package is the basis of information technology for automated construction and optimization of regenerative therapy schedules, has an adaptive interface, a remote database and is a multi-user software system.

The application of the three-tier architecture provides automatic centralized update of the software complex, independent configuration at each level of the architecture, centralized control through the use of authorization and authentication mechanisms.

The developed software package provides the ability to deploy on most modern platforms and operating systems through the use of high-level programming language - Java.

The system provides a friendly interface for working with a number of mobile and stationary hardware devices. The use of the software does not require specialized knowledge in the field of information technology.

Key words: therapeutic establishments, schedule, fuzzy logic controller, genetic algorithms, optimization of rehabilitation therapy plans, program complex.

Список публікацій здобувача:

Наукові праці, в яких опубліковані основні наукові результати дисертації:

1. Tkachenko R.O., Kovalyshyn O.S. Multicriteria optimization of medical institutions schedules on the basis of neuro fuzzy models and evolutionary algorithms. *Econtechmod: An international quarterly journal on economics in technology, new technologies and modeling processes*. 2017. Vol.6. No.3. P. 53-60. (***Index Copernicus***).
2. Tkachenko R.O., Kovalyshyn O.S. A method of assessing of clinic patients feedback with fuzzy logic. *Вісник Львівського державного університету безпеки життєдіяльності: зб. наук. праць*. 2017. № 15. С. 82-88. (***Ulrich's Periodicals Directory***)
3. Tkachenko R.O., Kovalyshyn O.S. A method of making up a clinic schedule with use of finite state automaton. *Econtechmod: An international quarterly journal on economics in technology, new technologies and modeling processes*. 2016. Vol.5. No.3. P. 131-134. (***Index Copernicus***)
4. Kovalyshyn O.S., Gabriel Yu. I. Development of a management systems model of automatic control with using fuzzy logic. *Econtechmod: An international quarterly journal on economics in technology, new technologies and modeling processes*. 2014. Vol. 3. No.4. P. 87 -90. (***Index Copernicus***)
5. Ткаченко Р.О., Ковалишин О.С. Багатокритеріальна оцінка оптимальності розкладів медичних закладів з використанням нечіткої логіки. *Енергетика та автоматика: журнал. НУБіП*. 2018. №1. С. 79-98. (***Ulrich's Periodicals Directory, Index Copernicus, Agris, Base***)
6. Ковалишин О.С. Аналіз методів оптимізації розкладів в контексті відновлювальної терапії . *Науковий вісник НЛТУ України: зб. наук. праць*. 2018. Том 28. №8. С. 136-140. (***Index Copernicus***)
7. Ковалишин О.С. Нейронечіткий генетичний алгоритм оптимізації планів відновлювальної терапії. *Вісник Національного університету «Львівська Політехніка»*. зб. наук. праць. Серія: Інформаційні системи та мережі. № 901. 2018. С.1-10.
8. Ковалишин О.С. Інформаційна технологія оптимізації розкладів відновлювальної терапії. *Науковий вісник Луцького НТУ*. зб. наук. праць. Серія «Комп'ютерно-інтегровані технології: освіта, наука, виробництво». 2018. №33.

C.85-89. (*Index Copernicus, Universal Impact Factor, Open Academic Journals Index*).

Наукові праці, які засвідчують апробацію матеріалів дисертації:

9. Ковалишин О.С. Вирішення задачі багатокритеріальної оптимізації розкладів за допомогою систем нечіткої логіки. Інформаційне суспільство: технологічні, економічні та технічні аспекти становлення: тези доповідей міжнародної наукової інтернет-конференції, м. Тернопіль, 17 травня, 2017 р. Випуск 19. Частина 1. Тернопіль, 2017. С. 40–42.

10. Ковалишин О.С. Оптимізація розкладів лікувальних закладів методом генетичного алгоритму. Інтелектуальні системи прийняття рішень і проблеми обчислювального інтелекту: матеріали міжнародної наукової конференції, 22-26 травня 2017 р. ПП Вишемирський В. С., 2017. С.274-275.

11. Ковалишин О.С. Кінцевий автомат побулови розкладу медичних закладів. Інтегровані інтелектуальні робототехнічні комплекси (ІРТК-2016): матеріали дев'ятої міжнародної науково-практичної конференції, м. Київ, 17-18 травня 2016 р. Київ: НАУ, 2016. С. 22 – 24.

12. Ковалишин О.С. Задача багатокритеріальної оцінки та оптимізації планів відновлювальної терапії пацієнтів медичних закладів. Информационные системы и технологии: материалы 4-ой международной научно-технической конференции, г. Харьков, 21-27 сентября 2015 г. Харьков, НТМТ, 2015. С. 70-72.

13. Ковалишин О.С. Прикладні аспекти використання систем нечіткого логічного висновку в задачах багатокритеріальної оптимізації. Інформаційна безпека та комп'ютерні технології: тези доповідей III Міжнародної науково-практичної конференції, м. Кропивницький, 19-20 квітня 2018 року. Кропивницький: ЦНТУ, 2018. С. 300-302.

Наукові праці, які додатково відображають наукові результати дисертації:

14. Р.О.Ткаченко, І.В. Ізонін, Н.В. Веретеннікова, О.Л. Березко, О.С. Спосіб зміни роздільної здатності набору сценарних зображень. пат. 124680 Україна: МПК G 06 K 9/36. №u201706713; заявл. 29.06.2017; опубл. 25.04.2018, Бюл. № 8. 6 с.

15. Ковалишин О.С., Ткаченко Р.О., І.В. Ізонін І.В. Комп'ютерна програма "Програмний комплекс побудови та оптимізації розкладів відновлювальної терапії": свід. про реєстр. автор. права на твір № 75244 від 06.12.2017. Зареєстр. в Міністерстві економічного розвитку і торгівлі України, заявка 10.10.2017, № 76059.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ	17
ВСТУП.....	18
РОЗДІЛ 1. АНАЛІЗ МЕТОДІВ РОЗВ’ЯЗКУ ЗАДАЧІ ОПТИМІЗАЦІЇ , ТЕОРІЇ СКЛАДАННЯ КАЛЕНДАРНИХ ГРАФІКІВ, ОЦІНКА АЛЬТЕРНАТИВ.	25
1.1 АКТУАЛЬНІСТЬ СКЛАДАННЯ КАЛЕНДАРНИХ ГРАФІКІВ У ЗАДАЧАХ ВІДНОВЛЮВАЛЬНОЇ ТЕРАПІЇ	25
1.2 ЗАГАЛЬНА ХАРАКТЕРИСТИКА ЗАДАЧІ СКЛАДАННЯ ТА ОПТИМІЗАЦІЇ КАЛЕНДАРНИХ ГРАФІКІВ.....	27
1.3 ЗАГАЛЬНА КЛАСИФІКАЦІЯ МЕТОДІВ ПОБУДОВИ КАЛЕНДАРНИХ ГРАФІКІВ	28
1.3.1 Точні методи побудови календарних графіків	29
1.3.2 Наближені методи побудови календарних графіків	36
1.4 ОЦІНКА МЕТОДІВ ПОБУДОВИ КАЛЕНДАРНИХ ГРАФІКІВ.....	41
1.5 БАГАТОКРИТЕРІАЛЬНІСТЬ ЗАДАЧІ ПОБУДОВИ КАЛЕНДАРНИХ ГРАФІКІВ.....	43
1.5.1 Основні поняття і визначення	43
1.5.2 Методи згортання векторного критерію	44
1.5.3. Багатокритеріальність задачі формування план-графіку відновлювальної терапії	48
ВИСНОВКИ ДО РОЗДІЛУ 1	51
РОЗДІЛ 2. РОЗРОБКА МЕТОДІВ І АЛГОРИТМІВ ОЦІНКИ ТА ОПТИМІЗАЦІЇ ПЛАНІВ ВІДНОВЛЮВАЛЬНОЇ ТЕРАПІЇ НА ОСНОВІ МОДЕЛЕЙ НЕЧІТКОЇ ЛОГІКИ І НЕЙРОНЕЧІТКИХ ЗАСОБІВ.	53
2.1 ЗАСАДИ ФУНКЦІОНУВАННЯ РЕАБІЛІТАЦІЙНО-ОЗДОРОВЧИХ УСТАНОВ.....	53
2.2 ІНДИВІДУАЛЬНІ ПРОГРАМИ РЕАБІЛІТАЦІЇ.....	56
2.3. ФОРМАЛІЗАЦІЯ ПРОЦЕСУ ПОБУДОВИ ПЛАН-ГРАФІКУ ВІДНОВЛЮВАЛЬНОЇ ТЕРАПІЇ	60

2.4 МЕТОД ПОБУДОВИ ОПОРНОГО ПЛАН-ГРАФІКУ	62
2.4.1 Загальна схема методу	62
2.4.2 Пріоритезація процедур.....	64
2.4.3 Призначення процедури	65
2.4.4 Вирішення конфліктів.....	66
2.4.5 Підбір часу для проведення процедури	68
2.5 СКІНЧЕННИЙ АВТОМАТ ПОБУДОВИ ПЛАН-ГРАФІКУ	68
2.5.1 Аналіз та вибір типу абстрактного автомата.....	69
2.5.2 Побудова функціональної моделі скінченного автомата Мілі	71
ВИСНОВКИ ДО РОЗДІЛУ 2	76
РОЗДІЛ 3. РОЗРОБКА АРХІТЕКТУРИ СИСТЕМИ ОПТИМІЗАЦІЇ ПЛАНІВ	
ВІДНОВЛЮВАЛЬНОЇ ТЕРАПІЇ НА ОСНОВІ НЕЙРОНЕЧІТКИХ	
МОДЕЛЕЙ.....	77
3.1. МЕТОДИКА ОЦІНКИ ТА ПОРІВНЯННЯ РІЗНИХ ВАРІАНТІВ ПЛАН-ГРАФІКІВ З	
ВИКОРИСТАННЯМ НЕЧІТКОЇ ЛОГІКИ	77
3.1.1. Принципи побудови систем нечіткої логіки.....	77
3.1.2. Аналіз існуючих підходів до побудови систем нечіткого логічного	
висновку	82
3.1.3. Розробка контролера нечіткої логіки оцінки оптимальності план-	
графіків лікувальних закладів.....	84
3.2 МЕТОД ОПТИМІЗАЦІЇ ПЛАН-ГРАФІКІВ	90
3.2.1 Застосування генетичних алгоритмів для оптимізації план-графіків	90
3.2.2 Розробка генетичного алгоритму оптимізації план-графіків.....	92
ВИСНОВКИ ДО РОЗДІЛУ 3	101
РОЗДІЛ 4. РОЗРОБКА І АПРОБАЦІЯ ПРОГРАМНОГО КОМПЛЕКСУ	
ОПТИМІЗАЦІЇ ПЛАНІВ ВІДНОВЛЮВАЛЬНОЇ ТЕРАПІЇ.....	103
4.1 ЗАГАЛЬНА АРХІТЕКТУРА ТА ПРИНЦИПИ ФУНКЦІОНУВАННЯ ПРОГРАМНОГО	
КОМПЛЕКСУ	103
4.2 РЕАЛІЗАЦІЯ ПРОГРАМНОГО КОМПЛЕКСУ.....	110

4.2.1	Опис основних модулів програмного комплексу	110
4.2.2	Опис структури бази даних програмного комплексу	115
4.3	РОБОТА З ВХІДНИМИ ДАНИМИ ПРОГРАМНОГО КОМПЛЕКСУ	119
4.3.1	Зчитування та лінгвістичний аналіз вхідних даних	121
4.3.2	Усунення суперечностей вхідних даних	121
4.4.	ОБҐРУНТУВАННЯ ВИБОРУ ТЕХНОЛОГІЇ РОЗРОБКИ ПРОГРАМНОГО КОМПЛЕКСУ	122
4.5	РОБОТА З ГРАФІЧНИМ КОРИСТУВАЦЬКИМ ІНТЕРФЕЙСОМ ПРОГРАМНОГО КОМПЛЕКСУ	124
4.5.1	Загальні компоненти графічного користувацького інтерфейсу	124
4.5.2.	Управління пацієнтами та планами реабілітації	125
4.5.3	Побудова та оптимізація план-графіку	127
4.6.	ПРАКТИЧНА АПРОБАЦІЯ ПРОГРАМНОГО КОМПЛЕКСУ ТА ПОРІВНЯННЯ З ІСНЮЮЧИМИ СИСТЕМАМИ ВІДНОВЛЮВАЛЬНОЇ ТЕРАПІЇ	129
	ВИСНОВКИ ДО РОЗДІЛУ 4	133
	СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	137
	ДОДАТКИ.....	149
	ДОДАТОК 1. СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА ЗА ТЕМОЮ ДИСЕРТАЦІЇ ТА ВІДОМОСТІ ПРО АПРОБАЦІЮ РЕЗУЛЬТАТІВ ДИСЕРТАЦІЇ.....	149
	ДОДАТОК 2. ДІАГРАМИ КЛАСІВ РОЗРОБЛЕНОГО ПРОГРАМНОГО КОМПЛЕКСУ	152
	ДОДАТОК 3. РЕЗУЛЬТАТИ АПРОБАЦІЇ РОЗРОБЛЕНОГО МЕТОДУ ОПТИМІЗАЦІЇ ПЛАН- ГРАФІКУ КЛІНІКИ	157
	ДОДАТОК 4. АКТИ ВПРОВАДЖЕНЬ РЕЗУЛЬТАТІВ ДИСЕРТАЦІЙНИХ ДОСЛІДЖЕНЬ.....	161

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

API – Application Programming Interface
BADD – basic defuzzification distributions
COA – center of area
COG – center of gravity
GIT – global information tracker
JSON – JSON
MOM – middle of maximum
MVC – Model View Controller
REST – Representational state transfer
SLIDE – semi-linear defuzzification
SPT/LPT – Shortest/Longest processing time
SQL – Structured Query Language
SRT/LRT – Shortest/Longest remaining time
XML – eXtensible Markup Language
ГА – генетичні алгоритми
ІПР – індивідуальна програма реабілітації
ОС – Операційна система
ПЗ – Програмне забезпечення

ВСТУП

Актуальність теми. Задачі календарного планування, які активізувалися у середині ХХ століття, були і є предметом широких наукових досліджень. Область їх використання охоплює різноманітні сфери людської діяльності.

Для розв'язання задач побудови графіків виконання робіт розроблено низку методів за результатами досліджень українських та закордонних учених. Практичну реалізацію вони отримали в сферах освіти, виробництва перевезень та масового обслуговування. Проте через потребу врахування специфіки функціонування лікувальних установ та закладів відновлюваної терапії (зокрема множину планів реабілітації та їх типів, низку медичних та організаційних вимог, необхідність врахування побажань пацієнтів відповідно до графіків роботи процедурних кабінетів) можливості побудови на їх основі оптимальних план-графіків є обмеженими.

Існує низка критеріїв, що визначають якість план-графіків відновлювальної терапії. Їх можна розділити на об'єктивні (вимоги до черговості лікувальних заходів, наявність спеціалізованого медичного обладнання для проведення процедур тощо) та суб'єктивні (побажання пацієнтів та медичного персоналу). Ефективність різних варіантів план-графіків в такому контексті визначається ступенем врахування як об'єктивних, так і суб'єктивних критеріїв. За таких умов варто застосовувати методи багатокритеріальної оцінки та оптимізації під час побудови план-графіків відновлювальної терапії.

Оскільки значну частку вимог, що формують критерії ефективності, подають у лінгвістичній формі відповідні експерти, для побудови інтегрального критерію оцінки доцільно використати апарат нечіткої логіки.

Значний вклад у розвиток теорії і практики алгоритмів нечіткої логіки зробили відомі українські та зарубіжні вчені, зокрема, Л. Заде (L. Zadeh), Е. Мамдані (E. Mamdani), М. Сугено (M. Sugeno) Х. Беренджі (H. Berenji), Р.О. Ткаченко, Р. Ягер (R. Yager), В. Целларі (W. Cellary). Прикладні аспекти застосування методів нечіткої логіки досліджували Дж. Баклі (J. Bakley), Х. Циммерман (H. Zimmerman) (нечітка оцінка та моделювання), О. О. Недосекін,

К.І. Воронов (нечіткий аналіз в управлінні), Т. Сааті (T. Saaty) (числова оцінка суб'єктивних суджень) тощо. Вони запропонували низку методів, які ґрунтуються на нечіткій логіці та придатні для розв'язання задачі побудови план-графіків медичних установ.

Серед методів дискретної багатокритеріальної оптимізації суттєві переваги має метод генетичних алгоритмів. Він є ефективним оптимізаційним алгоритмом, дає можливість уникнути локальних оптимумів та отримати множину глобальних оптимальних розв'язків. Проте окремі компоненти (кросовер, мутація, селекція, функція пристосованості) подаються узагальнено, що потребує адаптації до умов діяльності медичних закладів, для яких важливими є дискретність варіантів рішень і багатокритеріальність цільової функції.

Очевидним є те, що якість планів відновлювальної терапії суттєво впливає на рівень надання медичних послуг і ефективність роботи медичних установ, а їх оптимізація з використанням нових методів і засобів штучного інтелекту є актуальною і важливою задачею, що потребує окремого розв'язання.

Зв'язок роботи з науковими програмами, планами, темами. Дисертаційна робота виконана згідно з тематикою наукових досліджень кафедр Інформаційних технологій видавничої справи та Автоматизованих систем управління Національного університету “Львівська політехніка”, пов'язаних з інтелектуальною обробкою даних та паралельними обчисленнями, яка реалізована в межах держбюджетної науково-дослідної роботи “Відслідковування рухомих об'єктів у відеопотоках реального часу” (державний реєстраційний номер 0115U000432).

Мета і завдання дослідження. Метою дисертаційної роботи є розроблення методів і засобів продукційної системи нечіткої логіки для багатокритеріальної оптимізації план-графіків відновлювальної терапії.

Для досягнення поставленої мети необхідно було вирішити такі завдання:

– проаналізувати методи розв'язку задач побудови та оцінювання якості план-графіків, здійснити порівняння альтернатив;

- розробити метод автоматичної побудови опорних план-графіків відновлювальної терапії для подальшої оптимізації генетичним алгоритмом на основі нечітких моделей;
- розробити метод багатокритеріальної оптимізації генетичним алгоритмом з використанням бази продукційних правил для врахування побажань пацієнтів, що проходять реабілітацію;
- удосконалити продукційну систему нечіткої логіки для перетворення векторних критеріїв оптимізації план-графіків у скалярні;
- розвинути функціональну модель автомата Мілі для задач автоматичної генерації опорних план-графіків;
- розробити програмно-алгоритмічні засоби інтелектуальної системи автоматизованого складання план-графіків медичних установ та провести їх практичну апробацію.

Об’єктом дослідження є процеси формування план-графіків надання медичних послуг з урахуванням побажань пацієнтів, що проходять відновлювальну терапію, та наявних обмежень медичних установ.

Предмет дослідження – методи, моделі та алгоритми нечіткої логіки для формування інтегральних критеріїв оптимальності; генетичні алгоритми багатопараметричної, багатокритеріальної оптимізації для підвищення відповідності план-графіків сформованим критеріям оптимальності; теорія кінцевих автоматів для формування множини план-графіків надання медичних послуг за наявних можливостей і обмежень лікувальних установ.

Методи дослідження. Основні результати дисертаційної роботи отримані з використанням методів теорії нечіткої логіки для формування інтегрального критерію оцінки план-графіків на основі розробленої бази продукційних правил, еволюційного моделювання для побудови план-графіків функціонування медичних установ, методів теорії розкладів та теорії автоматів для побудови опорних план-графіків, теорії алгоритмів і програмування для розробки архітектури та програмних засобів.

Наукова новизна одержаних результатів полягає в тому, що

вперше розроблено:

- метод автоматичної побудови план-графіків функціонування медичних установ з урахуванням наявних обмежень на основі цифрового автомата Мілі, що дає можливість застосувати такі плани як опорні для наступної оптимізації генетичним алгоритмом.
- метод багатокритеріальної оптимізації план-графіків генетичним алгоритмом з використанням бази продукційних правил, розроблених на основі побажань пацієнтів, які проходять реабілітацію, що зменшує вірогідність потрапляння розв'язків у локальні екстремуми;

набули подальшого розвитку:

- цифровий автомат на основі абстрактної моделі Мілі для задач побудови план-графіків відновлювальної терапії, який шляхом врахування сукупності обмежень медичних установ забезпечує автоматичну генерацію опорних план-графіків.

удосконалено:

- продукційну систему нечіткої логіки шляхом використання в ній конфігурованого програмно-алгоритмічного T-Controller'а, що забезпечило підвищення якості багатокритеріальної оптимізації в порівнянні з існуючими методами внаслідок перетворення векторного критерію в скалярну форму.

Практичне значення одержаних результатів.

Отримані методи, моделі та алгоритми забезпечують формування біжучих план-графіків відновлювальної терапії за 2–3 хвилини з урахуванням побажань пацієнтів в умовах жорстких і м'яких обмежень лікувальних закладів.

Застосування розробленого методу багатокритеріальної оптимізації з використанням алгоритмічного контролера нечіткої логіки та генетичного алгоритму поліпшує якість результуючих план-графіків відновлювальної терапії за оцінками пацієнтів на 21-24%, та медичного персоналу – на 17-23%.

Запропонована інтегральна оцінка план-графіків роботи медичних установ,

яка ґрунтується на удосконаленій продукційній системі нечіткої логіки з використанням алгоритмічного T-Controller'a, забезпечує підвищення якості багатокритеріальної оптимізації порівняно з існуючими методами, оскільки враховує сукупність побажань пацієнтів та медичного персоналу, сформульованих у лінгвістичній формі.

Розроблена функціональна модель автомата Мілі забезпечила пришвидшене формування опорних план-графіків з можливістю внесення додаткових змін, що перманентно можуть виникати в процесі функціонування лікувального закладу.

Розроблений програмний комплекс пришвидшеного формування план-графіків відновлювальної терапії впроваджений і апробований у лікувально-діагностичному центрі, санаторії «Київ плюс» санаторно-курортного комплексу «Моршинкурорт», а також у Львівському обласному державному клінічному лікувально-діагностичному ендокринологічному центрі.

Отримані під час досліджень результати, зокрема методи, моделі та алгоритми складання план-графіків медичних установ з використанням нейронечіткого контролера, є основою підвищення ефективності їх діяльності.

Особистий внесок здобувача. Усі наукові результати дисертаційної роботи отримані автором самостійно. Одноосібно опубліковано праці [6, 7, 8]. У друкованих працях, опублікованих у співавторстві, здобувачеві належать: метод багатокритеріальної оптимізації план-графіків медичних закладів на основі удосконаленої продукційної системи нечіткої логіки та еволюційних алгоритмів [1, 6, 7, 10, 12]; характеристика нечіткого контролера для трансформації лінгвістичних даних, отриманих у результаті збору відгуків пацієнтів, у числові змінні [2]; метод побудови опорного план-графіка закладів лікувальної терапії на основі скінченного автомата, де проаналізовано тип та запропоновано механізм його функціонування [3, 11]; обґрунтування доцільності використання контролерів нечіткої логіки в автоматизованих системах оцінки та управління [4, 9, 13]; характеристика структурних компонентів інтелектуальної системи побудови та оптимізації план-графіків медичних установ [8, 14, 15]; визначення та

аналіз критеріїв, за якими можна інтегрально оцінити якість план-графіка медичного закладу з використанням контролера нечіткої логіки [5].

Апробація результатів дисертації. Основні наукові теоретичні положення та практичні результати дисертаційної роботи доповідалися, обговорювалися і отримали позитивну оцінку на: IV Міжнародній науково-практичній конференції «Информационные системы и технологии» (Харківський національний університет радіоелектроніки, 21-27 вересня 2015р.); Міжнародній науково-практичній конференції «Інтегровані інтелектуальні робото-технічні комплекси» (Навчально-науковий інститут інформаційно-діагностичних систем, м. Київ, 17-18 травня 2016 р.); Міжнародній науково-практичній конференції «Аналіз та моделювання складних систем і процесів. Теоретичні і прикладні аспекти систем прийняття рішень. Обчислювальний інтелект та індуктивне моделювання» (м. Залізний Порт, 22-26 травня 2017 р.); Міжнародній науковій інтернет-конференції «Інформаційне суспільство: технічні економічні та технічні аспекти становлення»(м. Тернопіль, 17 травня 2017 р.); III Міжнародній науково-практичній конференції «Інформаційна безпека та комп'ютерні технології» (Центральноукраїнський національний технічний університет, м. Кропивницький, 19-20 квітня 2018 р.);

Матеріали дисертації доповідалися та обговорювалися впродовж 2014 – 2018 років на наукових семінарах кафедри інформаційних технологій видавничої справи Національного університету «Львівська Політехніка».

Публікації. За результатами досліджень, викладених у дисертації, опубліковано 15 наукових праць, у тому числі 3 статті у міжнародних наукометричних виданнях, 5 статей у наукових фахових виданнях України, 5 публікацій у збірниках праць наукових конференцій, 1 свідоцтво про авторське право та 1 патент на корисну модель.

Структура та обсяг дисертації. Дисертаційна робота складається зі вступу, чотирьох розділів, висновків, списку використаних джерел та додатків. Загальний обсяг роботи складає 165 сторінок, включаючи 138 сторінок основного тексту, 44

рисунки, 14 таблиць. Список використаних джерел містить 182 найменування і викладений на 11 сторінках. Чотири додатки розміщено на 16 сторінках.

РОЗДІЛ 1. АНАЛІЗ МЕТОДІВ РОЗВ'ЯЗКУ ЗАДАЧІ ОПТИМІЗАЦІЇ , ТЕОРІЇ СКЛАДАННЯ КАЛЕНДАРНИХ ГРАФІКІВ, ОЦІНКА АЛЬТЕРНАТИВ.

1.1 Актуальність складання календарних графіків у задачах відновлювальної терапії

Основний напрямок роботи реабілітаційних закладів різного профілю полягає у відновленні здоров'я пацієнтів і є важливим етапом після надання невідкладної допомоги та інтенсивного лікування в стаціонарі.

Завдання створення та роботи сучасних центрів реабілітації протягом останніх років піднімалися в програмах соціальної, фізичної та професійної реабілітації людей з обмеженими можливостями.

Тому проблема відсутності ефективного процесу реабілітації є важливою та потребує першочергового вирішення.

Розширення реабілітаційних центрів, та створення нових закладів реабілітації в Україні є комплексним завданням Міністерства охорони здоров'я, Міністерства освіти і науки, Міністерства соціальної політики та інших державних та громадських організацій.

Як зазначено в Законі України «Про реабілітацію інвалідів в Україні» від 06.10.2005 № 2961-IV, «реабілітаційна установа - установа, підприємство, заклад, у тому числі їх відділення, структурні підрозділи, незалежно від форми власності, що здійснює реабілітацію інвалідів та дітей-інвалідів у відповідності з державними соціальними нормативами у сфері реабілітації» [1].

До спеціалізованих реабілітаційних установ стаціонарного типу відносяться [2]:

- відділення реабілітації, що знаходяться в структурі стаціонару, де хворий проходив лікування в гострій фазі захворювання;
- спеціалізовані реабілітаційні стаціонари (однопрофільні і багатопрофільні);
- міжрегіональні центри реабілітації.

Перед кожною з цих структур стоїть мета закінчення процесу реабілітації, відновлення здоров'я пацієнта, його соціального та професійного статусу до максимально можливого в конкретній ситуації рівня. Впродовж даного етапу здійснюється поступовий перехід до використання немедикаментозних реабілітаційних заходів (фізіотерапія, засоби фізичної реабілітації, методи психотерапії тощо).

До основних проблем відносяться: етапність відновлення здоров'я хворих та інвалідів (стаціонар – поліклініка – реабілітаційний центр – санаторій – диспансер), профілактика можливих ускладнень, а також ефективність лікувальних та реабілітаційних процедур та заходів [3, 4].

До активів реабілітаційних центрів відноситься не лише спеціалізована медична апаратура, але й кваліфіковані фахівці з різних галузей медицини, медико-соціальної та фізичної реабілітації, тому до його задач входить також організаційно-методична робота. Окрім цього, центри віграють значну роль в забезпеченні взаємозв'язків між структурами медико-соціальної реабілітації з іншими лікувальними та профілактичними установами, а також з громадськими інститутами, що беруть участь в забезпеченні процесу реабілітації. Аналіз літератури свідчить, що багатопрофільні реабілітаційні центри повинні включати наступні відділення: медичної реабілітації, фізичної реабілітації, соціальної реабілітації, психологічної реабілітації, професійної реабілітації, трудової реабілітації, інші підрозділи [5].

Підвищення ефективності лікувально-профілактичних і оздоровчо-реабілітаційних заходів, ціллю яких є усунення негативних наслідків захворювань є неможливим без раціонального та цілеспрямованого використання наявних методів і засобів фізичної реабілітації [6].

Проблема раціоналізації використання доступних ресурсів набирає ваги в зв'язку з розбудовою нових та укрупненням існуючих лікувальних закладів, оскільки ручна побудова графіків використання обладнання та роботи персоналу є трудомістким процесом та вимагає значних часових затрат, що тим не менш не гарантує оптимальності отриманого результату [7, 8].

Також в процесі відновлення особливе місце відводиться питанням надання індивідуалізованих реабілітаційних послуг, оскільки на даний момент графіки реабілітаційних терапій дуже обмежено враховують інтереси самих пацієнтів, що на пряму впливає на показники якості лікування [9, 10, 11].

Впродовж організації відновлювальної терапії до уваги слід брати множину обмежень які накладаються на процес - необхідність узгодження графіків використання медичного обладнання, наявність кваліфікованого медичного персоналу для проведення кожної конкретної процедури в певний момент часу, доступність відповідного устаткування тощо.

Фактично для вирішення даної задачі необхідно побудувати повноцінний графік функціонування медичного закладу, на який накладаються множини жорстких обмежень – умов що повинні обов'язково виконуватися для забезпечення коректності графіку, та м'яких – виконання яких є бажаним, власне індивідуалізації відносно інтересів пацієнтів.

1.2 Загальна характеристика задачі складання та оптимізації календарних графіків

Задачі складання графіків є частиною теорії розкладів. Даний напрямок в науці бере свій початок з відомої роботи Генрі Гантта 1903 г. [12], який запропонував тип діаграм, що використовується для ілюстрації плану, графіка робіт за будь-яким проектом. Ці діаграми в даний час називають діаграмами Гантта, та використовують в багатьох роботах по теорії розкладів. Термін "теорія розкладів" запропонував Р. Беллман в 1956 році [13]. Методи та алгоритми розв'язання задач теорії розкладів застосовуються для вирішення завдань комбінаторної оптимізації в різних сферах – в виробництві, управлінні, в галузі медицини, тощо.

Активне теоретичне дослідження задач теорії розкладів почалося з 50-х років 20-го століття. Цим роботам присвячені роботи Джонсона (Johnson [14]), Джексона (Jackson [15]) і Сміта (Smith [10]), Танаєва [11] та Конвея (Conway [16]).

Задачі організації календарних графіків виникають там, де виникає необхідність узгодження порядку виконання завдань із використанням певних

доступних ресурсів. Рациональне їх розподілення для виконання множини завдань на пряму впливає на розмір різного виду затрат, час необхідний для завершення комплексу завдань, а також на інші показники ефективності графіку.

У найбільш загальному формулюванні задача складання графіку полягає в наступному. З використанням деякої множини наявних ресурсів повинна бути виконана деяка фіксована система завдань із заданим набором характеристик. Такі характеристики можуть розглядатися як: обмеження на послідовність виконання операцій кожного завдання й часткові послідовності строків їх завершення, а також строки виконання завдань; ресурси, які необхідні для виконання кожного з завдань; час виконання робіт кожної із операцій завдань при використанні різних видів ресурсів; директивні терміни початку та завершення виконання кожного завдання; перелік та характеристики ресурсів, необхідних для виконання всіх операцій кожного із завдань [17].

Більшість практично змістовних задач складання розкладів допускають довільні комбінації обмежень і дисциплін обслуговування і є NP-повними задачами [18, 8].

1.3 Загальна класифікація методів побудови календарних графіків

Для вирішення задач побудови графіків розроблено велику кількість методів дискретної оптимізації. Особливе місце серед них займають методи, які можна розділити на дві групи: точні та наближені. Проте цей поділ є умовним, оскільки багато точних методів можуть застосовуватися як наближені. З іншого боку наближені методи за певних умов можуть виступати як точні або їх складова частина [19].

Аналізуючи [20, 21, 22, 23, 24] можна стверджувати, що до точних методів відносять: методи цілочисельного програмування, логічне програмування з обмеженнями, послідовні алгоритми оптимізації. В свою чергу методи послідовної оптимізації включають в себе метод гілок та меж, динамічне програмування з обмеженнями та метод аналізу та відсіву варіантів. Найбільш поширеними з

наближених методів є евристичні методи, методи що імітують фізичні процеси – метод відпалу металу, імітаційне моделювання, генетичні алгоритми та еволюційні стратегії. Графічно розподіл та класифікацію методів зображено на рис 1.1.

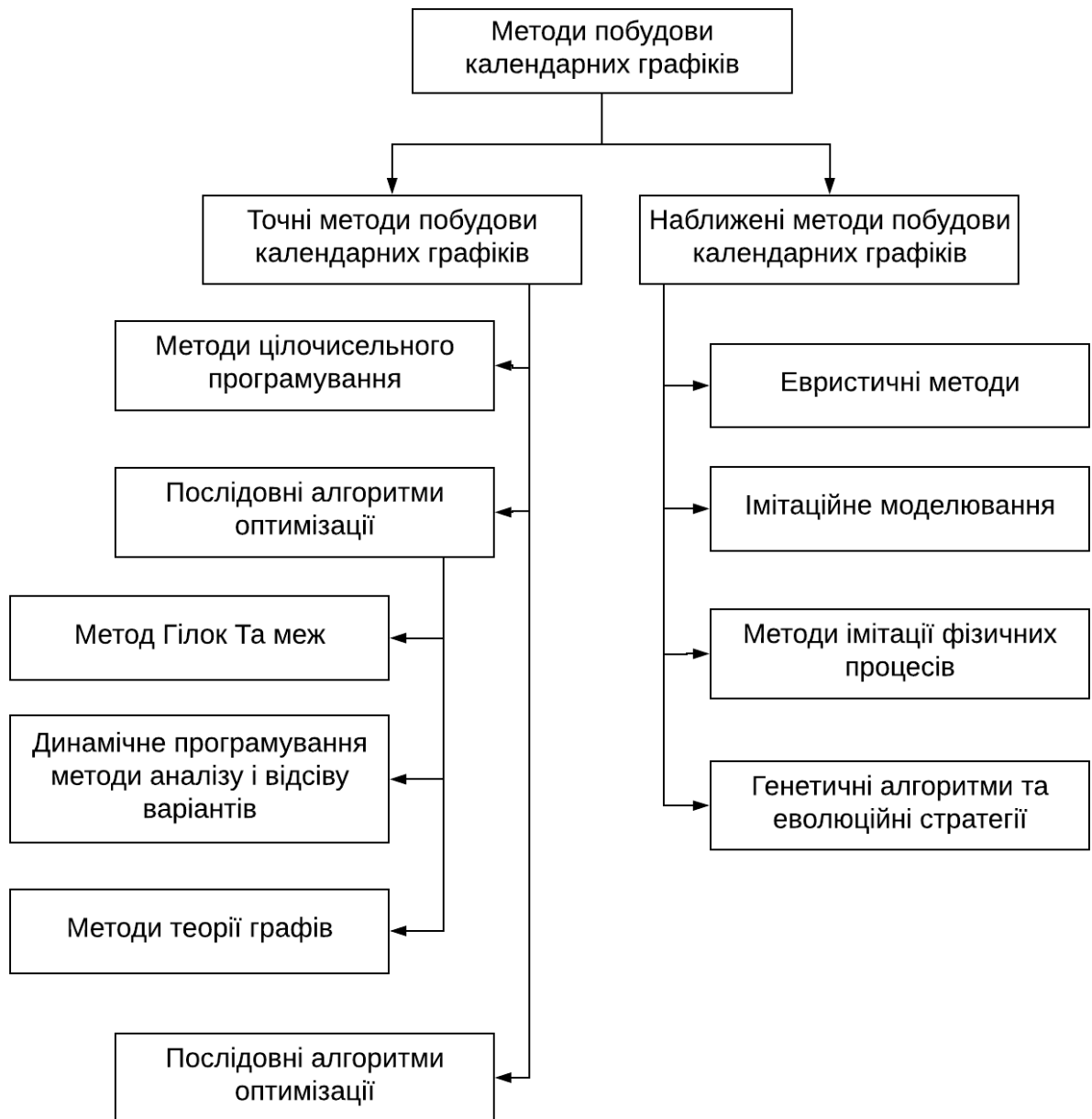


Рис. 1.1. Пропонована класифікація методів розв'язання задачі побудови графіків

1.3.1 Точні методи побудови календарних графіків

Методи цілочисельного програмування. Розвиток основ теорії розкладів (ТР) відбувався в період, коли моделі математичного програмування (МП) почали

застосовуватися для вирішення економічних задач. Тоді ж виникли спроби побудувати моделі МП і для задач ТР. Проте виникли певні труднощі. В задачах математичного програмування множина обмежень описує ситуацію, коли деяка сукупність умов повинна виконуватись одночасно. В теорії розкладів низка умов повинні виконуватись альтернативно (наприклад, або i -та активність виконується після завершення j -ї, або навпаки) [25].

Для знаходження оптимального розв'язку цілочисельних задач застосовуються спеціальні методи. Одним з найпростіших серед них, є знаходження її оптимального розв'язку як задачі, що має лише неперервні змінні, з подальшим округленням останніх [26]. Такий підхід часто є виправданим.

Для знаходження оптимальних планів задач цілочисельного програмування застосовують дві основні групи методів: методи відтинання та комбінаторні методи.

В основі методів відтинання лежить ідея поступового "звуження" області допустимих розв'язків вирішуваної задачі. Пошук цілочислового оптимумального рішення починається з розв'язування задачі з послабленими обмеженнями, тобто без врахування вимог цілочисловості змінних. Далі у модель вводяться додаткові обмеження, що враховують цілочисловість змінних. Багатокутник допустимих розв'язків задачі з послабленнями поступово зменшуємо до моменту, коли змінні оптимального розв'язку не набудуть цілочислових значень. До цієї групи належать наступні методи [27, 28, 29, 30] :

- методи розв'язування повністю цілочислових задач (дробовий алгоритм Гоморі);
- методи розв'язування частково цілочислових задач (другий алгоритм Гоморі, або змішаний алгоритм цілочислового програмування).

Комбінаторний підхід зводиться до цілеспрямованої перестановки пар робіт в деякій вихідній послідовності, доки не буде одержано оптимальний (або близький до оптимального) розв'язок.

Потреба у розробці алгоритмів побудови графіків, що здатні забезпечити високу якість отримуваних результатів і в той же час не потребуватимуть значних

обчислювальних ресурсів вимагає застосування методології побудови ПДС-алгоритмів для важкорозв'язуваних задач комбінаторної оптимізації [31]. Зміст цієї методології полягає у наступному. Спочатку на основі теоретичного аналізу досліджуваної задачі виявляються умови оптимальності допустимих розв'язків, потім розробляється алгоритм розв'язання задачі, що має дві складові: поліноміальну і експоненційну.

Поліноміальна складова породжується логіко-аналітичними умовами (p – умовами), виконання яких гарантує оптимальність отриманого розв'язку і синтезується таким чином, щоб послідовна процедура конструювання допустимих розв'язків була найефективнішою з точки зору реалізації p – умов. Коли допустимий розв'язок, отриманий поліноміальною складовою, не задовольняє p умовам, то розв'язання задачі може продовжуватись експоненційною складовою алгоритму [32].

Метод логічного програмування з обмеженнями. Побудову графіку можна представити у вигляді задачі задоволення обмежень. Програмування з обмеженнями тісно пов'язано з традиційним логічним програмуванням, в рамках якого воно сформувалося. Більшість систем програмування з обмеженнями представляють собою інтерпретатори мови Prolog з вбудованими механізмами для розв'язку певного класу задач задоволення обмежень. Програмування в рамках таких систем називають логічним програмуванням з обмеженнями (Constraint Logic Programming) [33, 34, 35].

Вирішення такої задачі полягає в визначенні деякої множини змінних x_1, \dots, x_n та областей їх значень. Область їх значень X_1, \dots, X_n , що описує обмеження, яким повинні задовольняти змінні. Враховуючи це, необхідно знайти такі значення змінних, які б задовольняли всім заданим обмеженням одночасно. В результаті реалізації цього методу визначається множина значень кожної змінної, що не суперечить вказаним обмеженням. Основна перевага даного методу полягає в зменшенні кількості варіантів пошуку. Це досягається за рахунок виключення з розгляду неперспективних варіантів [36, 37].

Метод гілок та меж. Метод гілок та меж є загальним алгоритмічним методом вирішення завдань оптимізації, зокрема він набув широкого застосування в таких NP-повних задачах як, задача комівояжера та задача про ранець. Даний метод є варіацією повного перебору. При його використанні відкидаються підмножини допустими хрешень, для яких не існує оптимальних значень цільової функції [38, 39].

В методі гілок та меж використовуються дві індивідуальні процедури: розгалуження та знаходження оцінок (меж) і універсальні процедури методу гілок та меж – процедури відсіву та зупинки.

Процедура оцінок включає в себе визначення верхньої V і нижньої H оцінки. Якщо для знайденого вузла дерева пошуку верхня межа співпадає з нижньою, то це значення є оптимальним і досягається на відповідній підмножині допустимих рішень. Оскільки будь яка перестановка з n чисел визначає допустиме рішення задачі, то в якості верхньої (досяжної) оцінки можна розглядати значення критерію, що визначається деякою перестановкою [40].

Процедура розгалуження розглядає всі допустимі варіанти побудови перестановок. За допомогою процедури розгалуження множина допустимих рішень на кожному етапі розділяється на підмножини меншої розмірності. Ці множини стають вузлами дерева пошуку.

Процедура відсіву передбачає, що якщо наступне значення верхньої (досяжної) оцінки в одній з вершин дерева розгалужень не перевищує значення нижньої оцінки в іншій вершині, то друга вершина виключається з розгляду.

Процедура зупинки визначає закінчення процесу обчислень. Якщо залишилася невідкинутаю лише одна вершина, в якій значення оцінок збігаються, то знайдено оптимальне рішення задачі, яке визначається перестановкою, що відповідає верхній оцінці.

Формальний опис методу для розв'язання задач на мінімум [41, 42] можна подати у такому вигляді:

$$f(x) \rightarrow \min, x \in D \quad (1.1)$$

де D – множина допустимих рішень; $d \in D$.

Функцію $b(d)$, яка співвідносить множині d розбиття його на підмножини $d_1, \dots, d_n, N > 1$, називають розгалуженням.

Функція $H(d)$ є нижньою межею для d , якщо:

1. $H(d) \leq \min f(x); x \in d$
2. на одноелементній множині $\{x\}$ вірна рівність $H(\{x\}) = f(x)$.

Реалізація методу гілок та меж, складається з послідовності подібних етапів.

На кожному етапі визначається запис x^0 і підмножини t_1, t_2, \dots, T_L не розглянутих розв'язків. На початку роботи методу $L = 1, t_1 = D, x^0$ – довільний елемент множини D , чи порожня множина.

На кожному етапі робота методу розпочинається з перевірки елементів розбиття. Хай здійснюється перевірка множини t_j . Множина t_j відсікається в одному з двох випадків, перевірка яких виконується послідовно :

а) якщо $H(t_j) \geq f(x^0)$;

б) якщо $H(t_j) < f(x^0)$ і знайдено такий елемент $y \in t_j$, що $f(y_j) = \min f(x) = H(t_j), x \in t_j$

В другому випадку відбувається заміна запису $x^0 = y_j$.

Нехай t_1, t_2, \dots, T_M ($M \leq L$) – невідсічені множини. Якщо $M = 0$, метод завершує роботу, і розв'язком задачі приймається запис x^0 . У випадку, коли $M \geq 1$ з множин t_1, \dots, t_M вибирається множина для виконання наступного розгалуження. Нехай такою є множина t_1 . Тоді проводиться розгалуження $b(t_1) = (d_1, \dots, d_N)$, в результаті якого отримуємо список множин $d_1, \dots, d_N, t_2, \dots, T_M$. Для цих множин проводиться індексація числами від 1 до L , і починається новий етап роботи методу.

В ході розв'язання конкретної задачі необхідно визначити методи побудови нижньої та верхньої оцінок, спосіб виконання розгалуження та правило вибору перспективної множини для розбиття.

Для визначення «перспективного» елемента розбиття в більшості випадків використовують одночасне и одностороннє розгалуження [43]. При виконанні одночасного розгалуження функція b може бути застосована до довільного

елементу розбиття. Часто в якості такого елемента вибирається підмножина t_k з мінімальною нижньою межею:

$$f(y_j) = \min_{x \in t_j} f(x) = H(t_j), x \in t_j \quad (1.2)$$

При односторонньому розгалуженні номер підмножини, що підлягає розбиттю, відомий заздалегідь.

Динамічне програмування та аналіз і відсів варіантів. Методи динамічного програмування отримали велике поширення під час вирішення деяких завдань дискретної оптимізації, зокрема задачі побудови графіків.

З точки зору формальної логіки схема методу зводиться до такої послідовності повторення процедур [44]:

- розбиття безлічі варіантів вирішення завдання на кілька підмножин, кожне з яких має специфічні властивості;
- використання цих властивостей для пошуку логічних протиріч в описі окремих підмножин;
- виключення з подальшого розгляду тих підмножин варіантів рішення, в описі яких є логічні суперечності.

Методика послідовного розвитку, аналізу і відсіву варіантів полягає в такому способі розвитку варіантів і побудови операторів їх аналізу, які дозволяють відсівати безперспективні початкові частини варіантів до їх повної побудови [45].

Задачу послідовного аналізу і відсіву варіантів можна означити наступним чином:

$$\min f(x), x \in D(X), \quad (1.3)$$

Обмеження які накладаються на $g_i(x_1, \dots, x_n) \leq 0, i = \overline{1, m},$

$$x_j = Q_j, j = \overline{1, n}, \quad (1.4)$$

де $Q_1 = (q_{1j}, \dots, q_{nj})$ – задані скінченні множини,

$x = (x_1, \dots, x_n)$, $p < n$ – часткове рішення, якщо $x_j \in Q_j$.

Множину всіх допустимих рішень схематично можна відобразити в вигляді дерева рішень H вершинам якого взаємо однозначно відповідають часткові рішення, висячим вершинам – повні рішення $p = n$.

Процес розв’язання задачі за допомогою методу послідовного аналізу і відсіву варіантів можна інтерпретувати як просування на дереві H , починаючи з його кореня, що відповідає частковому вирішенню $x_{(0)}$.

Щоб задати алгоритм вирішення конкретної задачі необхідно вказати правило вибору часткових рішень, що підлягають розвитку на кожному кроці, тобто стратегію послідовного конструювання рішень. Наступним кроком є побудова набору елімінувальних тестів, які здійснюють відсів часткових рішень, що не можуть бути добудовані ні до допустимих, ні до оптимальних рішень [46, 47, 48].

Метод розфарбування графу. Складання графіку можна представити як процес розфарбовування графу – пошук мінімального числа кольорів, необхідних для розфарбовування вершин деякого графу таким чином, щоб для кожної пари сусідніх вершин використовувалися різні кольори [49, 50, 51].

Така задача є NP- повною. Для її розв’язання найчастіше використовують різні жадібні алгоритми.

Під час складання графіку як задачі розфарбовування графу будується деякий граф, кожна вершина якого представляє собою заплановану активність. У разі виникнення конфлікту між вершинами вони з’єднуються ребром, що еквівалентно забороні одночасного проведення даних активностей [52]. Відповідно, складання графіку представляється як процес мінімізації числа кольорів, необхідних для розфарбування графа. Кожен період проведення активностей відповідає конкретному кольору. Застосування даного методу для розв’язку практичних реальних задач, очевидно, є малоефективним. Проте задача розфарбування графу в процесі складання графіку може бути корисною у випадку її комбінації з іншими алгоритмами [53, 54].

1.3.2 Наближені методи побудови календарних графіків

Обмежені перспективи застосування точних методів розв'язку задач теорії розкладів спричинили розробку наближених методів, які дозволяють одержувати задовільні розв'язки за порівняно невеликий час. Умовно наближені методи розділяються на евристичні та ймовірнісні. Реальні практичні задачі теорії розкладів відносяться до класу NP-повних задач [55]. Практично, це складно розв'язувані задачі й для алгоритму, який вирішує NP-повну задачу. У найгіршому випадку буде необхідна, експоненційна кількість часу для отримання точного рішення. Відповідно, такий алгоритм може використовуватися на практиці, тільки для отримання точних рішень задач невеликої розмірності [56]. Через великий обсяг практичних задач ні один із розглянутих вище підходів не гарантує отримання точного їх вирішення у прийнятний час [57]. Ця обставина зумовила розробку наближених методів, які дозволяють отримати прийнятне рішення при порівняно не великих затратах часу.

Евристичні алгоритми в своїй основі використовують так званий прийом зниження вимог. При цьому застосовуються деякі раціональні міркування, що не мають чіткого обґрунтування та гарантії, що знайдений графік є оптимальним.

Характерним представником групи евристичних алгоритмів є метод локального пошуку. За його використання заздалегідь визначена множина перестановок використовується для послідовного покращення початкового графіку поки таке покращення є можливим. Таким чином знаходиться локальний оптимум в даному класі перестановок [58].

Іншим напрямком евристичних методів в теорії розкладів є формування функцій переваг (пріоритетів). Для кожної i -ї роботи обчислюється значення функції переваги f_i та вибирається та робота, для якої f_i досягає мінімуму чи максимуму. Розглянемо деякі правила переваг [59]:

- Правила SPT/LPT (Shortest/Longest processing time). Перевага віддається роботі (операції) з множини готових до виконання, для якої час виконання є мінімальним/ максимальним.

- Правило SRT/LRT (Shortest/Longest remaining time). Перевага віддається роботі, для якої сума часів виконання залишкових робіт є найменшою/найбільшою.

Серед евристичних важливе місце займають так звані жадібні алгоритми. На кожному кроці їх роботи здійснюється вибір з усіх можливих альтернатив тої, що забезпечує найменше значення деякого критерію. Цей принцип може бути застосований як при знаходженні початкового графіку, так і на етапі його подальшого покращення (наприклад, з використанням вищезазначеного шляхом локального пошуку) [60, 61].

Іншим типом евристичних підходів є так звані генетичні алгоритми [62], суть яких полягає у наступному. Нехай є два коректні графіки. На їх основі за деякими визначеними правилами будується декілька наступних графіків, кожен з яких є “схожим” деякою мірою на ці два вихідні. З одержаних нащадків вибираються найкращі на основі певного заданого критерію та здійснюється перевірка, чи є вони кращими за “батьків”. Якщо так, то за цими графіками знову будуються нащадки і т.д.

Метод імітації відпалу. Цей метод імітує фізичний процес нагрівання та наступного контрольованого охолодження субстанції [63]. Він є одним із прикладів методів Монте-Карло. У ході пошуку оптимального результату допускається перехід з деякою ймовірністю в стан, що покращує значення цільової функції.

Алгоритм ґрунтується на імітації фізичного процесу, який відбувається при кристалізації речовини (при поступовому зниженні температури), в тому числі при відпалі металів [64].

За допомогою моделювання такого процесу шукається така точка або безліч точок, на якій досягається мінімум деякої числової функції $F(x)$, де $x=(x_1, \dots, x_m) \in X$.

Вводиться послідовність точок x_0, x_1, \dots, x_n простору X . Алгоритм послідовно знаходить наступну точку по попередній, починаючи з точки x_0 , яка є початковим наближенням. Алгоритм зупиняється після досягнення точки x_n . У контексті задачі

складання графіку даний метод можна представити наступним чином. Перед початком ітеративного оптимізаційного процесу генерується деякий початковий графік X_0 , що задовольняє жорсткі обмеження, та інтерпретується як поточний розв'язок задачі $X = X_0$. Також задається початкове високе значення температури T_0 деякий критерій зупинки роботи алгоритму та операція перетворення графіку.

Процес оптимізації складається з наступних етапів:

1. На основі введених операцій перетворення та поточного розв'язку генерується новий коректний графік X' , який дещо відрізняється від поточного.

2. Визначається зміна цільової функції $\Delta f = f(X) - f(X')$. Якщо $\Delta f < 0$ (розв'язок не погіршився), то новий варіант графіку стає поточним ($X = X'$). Якщо $\Delta f > 0$ (розв'язок погіршився), то новий графік стає поточним лише з ймовірністю p . Відповідно, з ймовірністю $(1 - p)$ попередній графік зберігається як поточний.

3. Виконується функція зміни поточної температури. Температура, і, відповідно, ймовірність прийняти поточного графіку з більшим значенням цільової функції зменшується на кожній ітерації.

4. Якщо не виконується критерій зупинки оптимізаційного процесу, виконується перехід на п.1.

Алгоритми імітації відпалу можуть бути використані для побудови графіків та виконання прикладних програм на всіх етапах проектування обчислювальних систем реального часу [65, 66]. Однак для їх практичного впровадження актуальною задачею залишається зменшення обчислювальної складності алгоритмів.

Метод імітаційного моделювання. Беручи до уваги NP-повноту задачі, для її розв'язання застосовують імітацію роботи експерта, що складає графіки. У цьому випадку метод оперує безпосередньо зі списком активностей, які необхідно включити до графіку [67].

Процес починається з порожнього графіку, коли всі активності розташовані в списку незапланованих. Далі алгоритм здійснює перебір для розподілення в часі всіх існуючих незапланованих активностей, найоптимальнішим чином. Процес триває доти, поки не буде сформований повний графік, або виконається фіксоване

число ітерацій. За детальнішого розгляду метод можна розділити на три основні етапи [68, 69]:

1. Обирається активність зі списку незапланованих. Вибір робиться в результаті аналізу найбільш пріоритетних ресурсів.

2. Для обраної активності визначаються всі можливі варіанти розміщення в графіку, які задовольняють жорстким обмеженням. Кожна альтернатива оцінюється за допомогою цільової функції та процедура розміщується в найкращій із знайдених позицій.

3. У випадку суперечливої ситуації проводиться видалення конфліктуючих активностей з графіку та їх внесення назад у список незапланованих.

Особлива увага під час розробки методу, заснованого на імітаційній моделі, приділяється евристичним правилам вибору активностей, визначенні найкращої для неї альтернативи, її розміщенні в графіку та його оцінці. До позитивних сторін такого підходу можна віднести можливість детального врахування специфіки задачі, що розв'язується. Проте слід зазначити, що навіть при незначних змінах у процесах побудови графіку доведеться вносити суттєві зміни в метод. Оскільки алгоритм ґрунтується на діях, які виконує експерт під час складання графіку, то існує можливість організувати дієвий діалог між користувачем і системою в процесі пошуку найкращої альтернативи. Проте це справедливо лише для відносно невеликих задач, оскільки в протилежному випадку значна залежність алгоритму від дій користувача може зробити такий діалог малоефективним.

Метод генетичного алгоритму. Генетичні алгоритми – стохастичні евристичні оптимізаційні методи, основна ідея яких взята з теорії еволюційного розвитку видів, та використання еволюційних механізмів для пошуку рішення [70]. Основними концепціями теорії еволюції є спадковість та природний відбір. Ці ж механізми використовуються генетичними алгоритмами.

На першому кроці розробки генетичного алгоритму здійснюється кодування структури задачі у вигляді певного вектору значень (“хромосоми”), що буде представляти розв’язок.

Наступним кроком є генерація деякої кількості початкових елементів (“початкової популяції”). У випадку задачі побудови та оптимізації графіків необхідно розбудувати такі їх множини, які б задовольняли жорстким обмеженням, що накладаються предметною областю [71].

Для роботи алгоритму оптимізації розробляється метод оцінки розвитку в популяціях. Таким методом може виступати вимога максимізації цільової функції або, в термінах генетичних алгоритмів, фітнес-функції. Тут доцільно використати інтегральний показник оптимальності графіку, заснований на штрафах за невдоволення поставлених до розв’язку вимог [72, 73].

Перетворення графіку здійснюється за рахунок використання еволюційних операторів мутації та кросоверу. В процесі їх реалізації також беруться до уваги жорсткі обмеження [74].

Для закінчення роботи алгоритму будується критерій зупинки. Таким критерієм може бути [75]:

- досягнення глобального оптимального рішення;
- вичерпання числа ітерацій генетичного алгоритму;
- вичерпання часу роботи генетичного алгоритму.

Таким чином, після розміщення початкової популяції в розробленому середовищі та реалізації еволюційних процесів, отримується ітераційний метод пошуку оптимального рішення. На кожному його етапі виконуються наступні дії [76, 77, 78]:

1. Кожна особина поточної популяції оцінюється за допомогою фітнес-функції.
2. Кращі рішення копіюються до нової популяції без змін.
3. На основі пропорційного відбору з поточної популяції вибираються хромосоми, які піддаються рекомбінації.
4. В випадку, коли нова популяція сформована, стара видаляється. Після цього переходимо до п. 5, а у протилежному випадку – до п. 3.

5. Якщо не виконується критерій зупинки оптимізаційного процесу, виконується перехід на п.1.

1.4 Оцінка методів побудови календарних графіків

Загальним недоліком точних методів є те, що вони засновані на ітераційній процедурі пошуку або поліпшення деякого початкового наближення. Відповідно, отримане рішення має пряму залежність від деякого початкового наближення і природно виникає проблема вибору його значення.

Застосування методів цілочисельного математичного програмування, навіть для задач відносно малої розмірності, вимагає значних часових затрат [79]. Для дотримання основних вимог до графіку, в математичну модель необхідно включити низку обмежень та змінних. Відповідно не лише цілочисельна, а й проста задача лінійного програмування великої розмірності не має перспектив на оптимізацію точними методами [80].

Основною перевагою методу гілок і меж є те, що в будь-який момент часу, припинивши обчислення, краще рішення, відповідне верхньої (досяжній) оцінці, може бути прийнято за наближене рішення задачі [81].

Основним недоліком методу гілок і меж є відсутність критерію оптимальності і, в результаті, потреба повністю вирішувати задачі лінійного програмування, асоційовані з кожною вершиною множин допустимих рішень. Оптимальне рішення може бути знайдено задовго до зупинки алгоритму, але виявити це в загальному випадку не можна, оскільки оптимальність з'ясовується лише після вичерпання списку завдань. Для задач великої розмірності це вимагає значних і, певною мірою, невиправданих з практичної точки зору витрат часу. Окрім цього, для конкретних типів задач проблематичним може бути отримання оцінок для здійснення процедури розгалуження.

Наближені методи, як правило, ґрунтуються на використанні різноманітних евристик або евристичних алгоритмів. Формування графіків за допомогою деяких правил забезпечує прискорення пошук найкращої альтернативи графіку. Але

використання таких методів часто гарантує знаходження лише наближеного до оптимального рішення (досягнення локального екстремуму).

Перевагою евристичних методів є зручність їхньої реалізації на обчислювальній техніці навіть для розв'язання громіздких задач.

Основний недолік евристичних методів полягає в складності оцінки близькості одержаних графіків до оптимального. Крім того, для кожного з описаних евристичних алгоритмів існують задачі, для яких застосування означених алгоритмів дає погані результати.

Для багатьох NP-важких задач найкращі рішення були отримані алгоритмами імітації відпалу. Однак їх недоліком є висока обчислювальна складність [82]. Це зумовлюється тим, що для отримання необхідного рішення необхідним є досить повільне зниження ймовірності переходу в стан із більш високим значенням цільової функції, що в свою чергу призводить до великої кількості ітерацій алгоритму. Алгоритми імітації відпалу можуть бути застосовані для побудови графіків на всіх стадіях проектування обчислювальних систем реального часу [83]. Однак для практичного використання методів відпалу важливою залишається задача зменшення обчислювальної складності алгоритмів.

Генетичні алгоритми можуть бути застосовані в ході вирішення широкого класу задач, у тому числі великорозмірних. Вони характеризуються стійкістю до потрапляння у локальні оптимуми поверхні можливих рішень [84, 85]. Ще однією важливою особливістю генетичних алгоритмів є можливість використання в задачах зі змінним середовищем. Проблематично використовувати ГА у випадках [86, 87]:

- коли необхідно знайти точний глобальний оптимум;
- час виконання функції оцінки великий;
- необхідно знайти всі можливі вирішення задачі, а не одне з них;
- кодування рішень надто складне.

1.5 БАГАТОКРИТЕРІАЛЬНІСТЬ ЗАДАЧІ ПОБУДОВИ КАЛЕНДАРНИХ ГРАФІКІВ

1.5.1 Основні поняття і визначення

Багатокритеріальна задача - математична модель прийняття оптимального рішення одночасно за кількома критеріями. Ці критерії можуть відображати оцінки різних якостей об'єкта (або процесу), з приводу яких приймається рішення, чи оцінки однієї і тієї ж його характеристики, але з різних точок зору [88].

Формально багатокритеріальна задача задається множиною X допустимих рішень і набором цільових функцій, f_1, \dots, f_n , що приймають дійсні значення.

Кожне можливе рішення характеризується деякою множиною значень цільових функцій оптимізації. На основі цього формується деяке розуміння переваг і недоліків знайдених рішень та можливість віддати перевагу одному рішенню над іншим. Оптимальним є рішення, яке є кращим за інші на основі значень цільових функцій [89, 90].

Цільові функції являють собою вирази, значення яких необхідно мінімізувати чи максимізувати. Іншими словами вони є також критеріями оптимальності, які дозволяють кількісно порівнювати декілька альтернативних рішень.

Одні алгоритми оптимізації пристосовані для пошуку максимуму, інші - для пошуку мінімуму. Проте незалежно від типу розв'язуваної задачі на екстремум можна користуватися одним і тим же алгоритмом, оскільки задачу мінімізації можна легко перетворити в задачу на пошук максимуму, помінявши знак цільової функції на зворотний.

Розмірність простору рішень може варіюватися в залежності від задачі та накладених обмежень. Слід зазначити, що дуже часто у зв'язку з обмеженнями оптимальне значення цільової функції досягається на жодному з кордонів області безлічі допустимих рішень задачі [91].

Локальними оптимумами називаються точки простору рішень, в яких цільові функції приймають найвищі значення в порівнянні з іншими точками в їх околі. Глобальний оптимум – є найбільш прийнятним рішенням для всієї множини рішень.

Він є кращим за всі інші рішення, що відповідають локальним і значенням цільових функцій.

Сутність багатокритеріальної задачі полягає в знаходженні її вирішення $x \in X$, яке в тому чи іншому сенсі максимізує чи мінімізує значення всіх функцій $f_i, i = 1, \dots, n$. Існування рішень, що максимізують буквально всі цільові функції є рідкісним винятком. Тому в теорії багатокритеріальних задач поняття оптимальності отримує різні і притому нетривіальні тлумачення .

1.5.2 Методи згортання векторного критерію

Один з ключових етапів розв'язання багатокритеріальної векторної задачі полягає в зведенні її до певної однокритеріальної задачі. Для цього будується скалярна цільова функція F , що враховує відносні важливості часткових критеріїв і є узагальненим представленням векторного критерія $Q(\bar{x})$ [92, 93].

Результуюча задача може бути описана наступним чином:

$$\min_{x \in D_x} F(w, Q(\bar{x})), \quad (1.5)$$

де $w = \{w_1, w_2, \dots, w_n\}$ – вагові коефіцієнти відносної важливості критеріїв.

Вагові коефіцієнти w_i визначають відносні важливості i – того критерію і визначають в кількісному вимірі його пріоритет відносно інших критеріїв.

Оскільки невдалий вибір цільової функції може привести до вибору не найкращі альтернативи, розроблено велику кількість підходів до їх побудови.

Як узагальнені критерії якості можуть бути використані функції F наступного вигляду [94, 95, 96]:

- адитивний критерій оптимальності

$$F(w, Q(\bar{x})) = \sum_{i=1}^S w_i Q_i(x), \quad (1.6)$$

- мультиплікативний критерій оптимальності

$$F(w, Q(\bar{x})) = \prod_{i=1}^S w_i Q_i(x), \quad (1.7)$$

- середньостепеневий критерій оптимальності

$$F(w, Q(\bar{x})) = \left[\frac{1}{S} \sum_{i=1}^S w_i Q_i(x) \right]^{1/p}, \quad (1.8)$$

де w_i - вагові коефіцієнти відносної важливості критеріїв.

$Q_i(x)$ – частковий критерій оптимальності

Найпростіший варіант часткової функції належності - значення відповідного i -го часткового показника варіанти вирішення m_r , нормовані на деякому інтервалі, зазвичай $[0..1]$.

У цьому випадку в якості бази для нормування часто обираються максимальні і мінімальні значення показників за вибіркою порівнюваних варіантів рішення. Математичне представлення нормованих показників m_i^{normed} виглядає наступним чином:

$$m_i^{normed} = \frac{m_i - m_i^{min}}{m_i^{max} - m_i^{min}}, \quad (1.9)$$

де m_i^{min} – мінімальне значення i -го часткового критерію;
 m_i^{max} – максимальне значення i -го часткового критерію.

Іншим варіантом нормування є приведення значення деякої характеристики до максимального значення цієї характеристики в просторі розглянутих варіантів рішень:

$$m_i^{normed} = \frac{m_i}{m_i^{max}} \quad (1.10)$$

Недоліки даного підходу обумовлені тим, що спроби знайти об'єктивну форму опису критеріїв будуються на суб'єктивному виборі діапазонів зміни даної величини, що не дозволяє отримувати стійкі результати при введенні відносних величин оцінки показників [97].

Слід зауважити, що незаперечна основа для визначення ваг існує тільки тоді, коли всі часткові критерії можуть бути однозначно відображені на деяку загальну шкалу, наприклад вартості або будь-якої фізичної величини. У протилежному випадку обґрунтований і зрозумілий вибір значень ваг досить скрутний.

Тому для визначення вагових коефіцієнтів розробники різних методик і моделей, як правило, використовують методи експертних оцінок.

Для того щоб якимось чином виявити порівняльні переваги експертів, щодо тих чи інших параметрів, розроблено ряд методів, результати яких за допомогою математичних маніпуляцій перетворюються в вагові коефіцієнти.

До цих методів зокрема відносяться: використання коефіцієнтів заміщення замість коефіцієнтів важливості [98], методи попарного порівняння показників за ступенем впливу на інтегральну перевагу [99].

Використання коефіцієнтів заміщення замість коефіцієнтів важливості. Дослідник з'ясовує у експерта, наскільки можна погіршити рішення за одним показником, щоб домогтися найкращого значення по іншому при незмінних значеннях інших показників. При цьому кожен перевагу досить задіяти тільки в одному зіставленні. Єдиний недолік – необхідність встановити точні (на суб'єктивному рівні) відношення між коефіцієнтами для часткових критеріїв.

Метод попарного порівняння. За грубою шкалою експерт попарно порівнює силу впливу показників на підсумкове рішення про вибір альтернативи. У найпростішому випадку експерту пропонується заповнити таблицю порівняння часткових критеріїв.

Якщо, наприклад, перший показник важливіше, ніж r -й, то в клітинку, розташовану на пересіченні першого рядка і r -го стовпця вноситься 2, в іншому випадку - 0. Якщо відповідь дати складно або експерт вважає показники рівноцінними, то в відповідному полі вноситься 1.

Порівняння пар показників здійснюється за дуже приблизною шкалою (всього три значення), але за рахунок того, що для з'ясування важливості кожного показника використовується кілька оцінок порівняльної важливості з іншими частковими показниками (різні вимірники приблизні, але їх декілька), точність відносних оцінок виявляється достатньо високою. Очевидно, що метод дає більш адекватні і стійкі результати при збільшенні кількості порівнюваних показників.

Попарне порівняння може здійснюватися і по шкалі, що має більше градацій, наприклад «значно важливіше», «істотно важливіше», «байдуже», «менш важливий», «істотно менш важливий».

У таблиці збираються відносні значення вагових коефіцієнтів часткових критеріїв. Далі вираховують абсолютні ваги показників, які використовуються для скаляризації векторних переваг (обчислення інтегральних функцій переваг). Для цього існує низка способів. Прийнятні результати, можна отримати шляхом визначення абсолютного ваги як частки отриманих балів в їх загальній сумі [100].

Пріоритет i -того показника в цьому випадку розраховується за формулою:

$$Z_i = \frac{\sum_{a=1, r \neq a}^R b_{ra}}{R \cdot (R-1)}, \quad (1.11)$$

де b_{ra} - значення в осередку, що стоїть на перетині k -го рядка і a -го стовпця.

Методу притаманні деякі недоліки, які не дозволяють розглядати його результати як абсолютно точне відображення структури переваг. По-перше,

відсутня об'єктивна основа для визначення діапазону вимірювання важливості. По-друге, результати обчислення абсолютних ваг сильно залежать від обраного чисельного еквіваленту найменшого значення шкали порівняння.

Наприклад, якщо обрана шкала $[0,1,2]$, то в деяких рядках матриці можуть виявитися лише нулі і відповідний показник може отримати нульову вагу. У цьому випадку він повинен бути виключений з аналізу, навіть якщо це недоцільно з практичної точки зору.

Якщо виключати показник немає можливості, то можна вдатися до двох способів корекції ваг, одержуваних безпосередньо за даними таблиці [101, 102].

Спосіб 1. Встановлюється мінімальна вага z_{min} , яку може мати показник, і відповідно до неї замість z_r використовується z_r^{\sim} , що обчислюється за такою формулою:

$$z_r^{\sim} = (z_r + z_{min})(1 + R \cdot z_{min}) \quad (1.12)$$

Спосіб 2. Встановлюється максимально допустиме відношення між максимальним і мінімальними вагами $h = z_{max} / z_{min}$. Ідея цього способу найбільш близька до коефіцієнтів заміщення.

1.5.3. Багатокритеріальність задачі формування план-графіку відновлювальної терапії

Завдання складання план-графіку відновлювальної терапії - це пошук найкращого значення для деякої множини характеристик вихідних даних. Вихідними даними для графіку є множина лікувальних планів для пацієнтів. Оптимізацію проводять по кільком частковим критеріям $Q_i(\bar{x}) (i = 1, 2, \dots, s)$, а отриману задачу називають багатокритеріальною або векторною оптимізацією. Тут ціллю є знаходження компромісу між частковими критеріями за якими потрібно оптимізувати графік.

Однією з причин, що призводять до багатокритеріальності, є множинність медичних, методичних і організаційних обмежень і вимог, які пред'являються до

графіку терапії. Багато з обмежень носять бажаний характер і їх можна звести до системи нерівностей [103, 104].

$$q_i(\bar{x}) \leq q_i^+, i = 1, 2, \dots, s, \quad (1.13)$$

Де q_i^+ - граничне значення i – того обмеження.

В цьому випадку часткові критерії оптимальності зазвичай в явному вигляді відсутні і їх доводиться вводити штучно за допомогою обмежень:

$$Q_i(\bar{x}) = \begin{cases} 0, \text{ якщо } q_i(\bar{x}) \leq q_i^+ \\ w_i (q_i(\bar{x}) - q_i^+), \text{ якщо } q_i(\bar{x}) > q_i^+ \end{cases} \quad (1.14)$$

Де w_i – ваговий коефіцієнт, що враховує важливість i -того обмеження ($\sum_{i=1}^s w_i = 1$)

Таким чином, рішення системи нерівностей (1.1) зводиться до вирішення завдання векторної оптимізації:

$$\min Q_1(\bar{x}), \min Q_2(\bar{x}), \dots, \min Q_s(\bar{x}) \quad (1.15)$$

Наступною причиною багатокритеріальності є необхідність отримання найбільш оптимального початкового графіку при різних можливих його варіантах, тобто забезпечення екстремуму цільової функції оптимізації при будь-яких значеннях критеріїв оптимальності. Цьому сприяє статичне або динамічне завдання вагових коефіцієнтів w_i - цільової функції оптимізації, які коригують процес оптимізації для різних початкових графіків.

При постановці завдання складання початкового графіку терапії одним з основних питань є вибір критерію оптимальності $Q_i(\bar{x})$. З одного боку, критерій повинен мати конкретний фізичний зміст, а з іншого – від нього вимагається, щоб він якомога повніше характеризував графік, який оцінюється. Однак вимоги

функціональної повноти важко задовольнити з допомогою лише одного скалярного показника, так як він зазвичай описує конкретну характеристику графіку. У зв'язку з цим доводиться розглядати сукупність показників $Q_1(\bar{x}), \dots, Q_s(\bar{x})$, кожен з яких має наочну фізичну інтерпретацію і дозволяє оцінити якість одержуваного рішення \bar{x} з різних точок зору.

Таким чином, необхідність забезпечення функціональної повноти показників, які конкретизують оптимальні властивості формованого графіку терапії, при одночасній їх фізичної наочності призводить до багатокритеріальності, яка впливає прямо з постановки задачі оптимального проектування.

Графік відновлювальної терапії складається з взаємозалежних блоків (процедур які складають плани відновлювальної терапії, множина яких в свою чергу складає графік роботи) оптимальність всього графіку визначається ефективністю і якістю його окремих частин, кожна з яких може бути охарактеризована, але принаймні, одним частковим критерієм оптимальності $Q_i(\bar{x})$ [105, 106, 107].

В цьому випадку отриманий графік можна вважати оптимальним, якщо забезпечуються екстремальні значення всіх критеріїв оптимальності як основних підцілей однієї спільної мети оптимізації.

ВИСНОВКИ ДО РОЗДІЛУ 1

1. Одним з найголовніших питань відновлювальної терапії та лікування в загальному є надання пацієнтам лікувальних закладів індивідуалізованих реабілітаційних послуг. Ефективне їх проведення неможливе без удосконалення та оптимізації планів проведення та графіків виконання процедур в лікувальних закладах.

2. Впродовж організації відновлювальної терапії до уваги слід брати множину обмежень які накладаються на процес - необхідність узгодження графіків використання медичного обладнання, наявність кваліфікованого медичного персоналу для проведення кожної конкретної процедури в певний момент часу, доступність відповідного устаткування тощо. Фактично для вирішення даної задачі необхідно побудувати повноцінний календарний графік функціонування медичного закладу, на який накладаються множини жорстких обмежень – умов що повинні обов'язково виконуватися для забезпечення коректності графіку, та м'яких – виконання яких є бажаним, власне індивідуалізації відносно інтересів пацієнтів.

3. Завдання складання план-графіку відновлювальної терапії - це пошук найкращого значення для деякої множини характеристик вихідних даних. Багатокритеріальність задачі побудови план-графіку відновлювальної терапії витікає з необхідності узгодження медичних, методичних і організаційних обмежень, що пред'являються до графіку, а також потреби отримання найбільш оптимального рішення при різноманітних варіаціях початкових даних – множини планів відновлювальної терапії.

4. Проблема побудови та багатокритеріальної оптимізації графіків відноситься до класу NP-повних. Для таких категорій задач характерною є відсутність детермінованих алгоритмів розв'язку за поліноміальний час в залежності від розмірності. Тому практичні потреби привели до появи наближених алгоритмів отримання субоптимальних рішень за прийнятний час.

5. Одним з найбільш перспективних, судячи по літературних даних, підходів для багатокритеріальної оптимізації графіків медичних закладів, що дає можливість уникнути локальних екстремумів та отримати множину глобальних субоптимальних рішень, є використання генетичних алгоритмів.

РОЗДІЛ 2. РОЗРОБКА МЕТОДІВ І АЛГОРИТМІВ ОЦІНКИ ТА ОПТИМІЗАЦІЇ ПЛАНІВ ВІДНОВЛЮВАЛЬНОЇ ТЕРАПІЇ НА ОСНОВІ МОДЕЛЕЙ НЕЧІТКОЇ ЛОГІКИ І НЕЙРОНЕЧІТКИХ ЗАСОБІВ.

2.1 Засади функціонування реабілітаційно-оздоровчих установ

Після закінчення реабілітації в лікувально-профілактичних закладах пацієнт скеровується в реабілітаційно-оздоровчі установи, що забезпечують санаторний етап реабілітації.

В ході санаторного лікування використовуються природні фізичні фактори. Це забезпечує високу ефективність терапії, реабілітації та профілактики, що в свою чергу дозволяє зменшити кількість медикаментозних засобів в лікуванні. В закладах санаторного лікування (санаторіях) використовуються лікувальні комплекси кліматотерапії, бальнеотерапії, грязелікування тощо.

Звично санаторії розташовуються в курортних місцевостях з присутністю природніх лікувальних факторів (мінеральні води, сприятливі кліматичні умови тощо), а також структури бальнеологічних, гідротехнічних та лікувально-профілактичних закладів [108].

Санаторії забезпечують комфортне розміщення пацієнтів для проведення відновлювальних заходів: лікувальні корпуси, пляжі, солярії, бювети мінеральних вод, інгаляторії, сауни, кабінети фізіотерапії, водо та болотолікування, спортивні споруди, кабінети лікувальної фізкультури, масажу, рефлексотерапії, психотерапії. Разом з спеціально обладнаними приміщеннями для проживання і відпочинку, спланованим режимом харчування, музичною, ароматичною, вокальною терапією, маршрутами для теренкуру, екскурсіями, туризмом, висококваліфікованим, обслуговуючим персоналом, санітарно-просвітньою роботою створюються необхідні умови для мінімального застосування медикаментозного лікування, а також позитивних результатів реабілітації [109].

Найбільш поширеними профілями санаторіїв для лікування пацієнтів є [110]:

- з хворобами серцево-судинної системи;

- з хворобами органів травлення;
- з хворобами органів дихання (не туберкульозного 8 походження);
- з хворобами жіночих статевих органів;
- з хворобами опорно-рухового апарату;
- з хворобами шкіри;
- з хворобами нирок і сечовивідних шляхів;
- з порушеннями обміну речовин.

Відповідно до існуючих положень скерування хворих на лікувальне санаторно-курортне лікування входить до обов'язків лікаря куратора та завідуючого відділенням, а там де такі вони відсутні - головного лікаря лікарні, поліклініки, диспансеру, медсанчастини. Лікарі-куратори відповідають за коректність добору хворих для санаторного та амбулаторного лікування на місцевих санаторіях та курортних місцевостях [111].

Основними закладами, що здійснюють реабілітацію, є установи наступних типів:

- реабілітаційно-лікувальні (лікувально-профілактичні),
- реабілітаційно-оздоровчі,
- реабілітаційно-протезні.

Тип та вид реабілітаційного закладу визначаються в ході його акредитації та подальшого ліцензування, що надає повноваження установам охорони здоров'я на виконання певних видів реабілітаційних послуг. Установа повинна володіти необхідним оснащенням відповідати організаційно-технологічним, медико-технологічним вимогам, санітарним нормам, мати позитивні показники кінцевих результатів, бути укомплектованою кваліфікованими кадрами.

В процесі відновлювальної терапії бере участь низка фахівців: лікарі терапевти, травматологи, невропатологи, нейрохірурги та інші спеціалісти (в залежності від характеру захворювань осіб, які підлягають реабілітації); фізіотерапевти, медичні сестри, спеціалісти з лікувальної фізкультури та спорту,

психотерапевти, експерти з праці, педагоги, психологи, спеціалісти з працетерапії, соціологи, представники органів соціального забезпечення тощо.

На базі лікарень питаннями відновлювального лікування займається окремий структурний підрозділ – відділення (кабінет) відновлювальної терапії.

Відділення забезпечує пацієнтів всім необхідним комплексом методів для ефективної реабілітації: фізіотрудопсихотерапією, лікувальною фізкультурою, лікувальними масажами, медикаментозними процедурами, що направлені на повне або часткове відновлення діяльності та функцій організму. [112].

Відділення або кабінет підтримує безпосередній зв'язок та тісні контакти з усіма лікарями та структурними підрозділами поліклініки. До відділення відновлювальної терапії поліклініки пацієнти надходять з відділів відновлювального лікування медичних установ для здійснення завершального етапу лікування або після закінчення гострого періоду захворювання для проведення комплексу відновлювальної терапії.

Основними задачами лікувального відділення є [113, 114, 115, 116]:

- використання комплексу всіх необхідних методів відновлювального лікування;
- неперервність, наступництво, послідовність та етапність;
- індивідуальний підхід до створення та здійснення програми лікування.

Вирішення цих задач визначає основні функції лікувального відділення:

1. складання індивідуального плану відновлювального лікування хворого та його реалізація з використанням сучасних методів та засобів;
2. засвоєння та практичне впровадження в діяльність відділення нових сучасних методів та засобів відновлювального лікування;
3. провадження консультативної та організаційно-методичної підтримки у питаннях відновлювального лікування поліклінічним закладам в галузях діяльності відділення;
4. двосторонній зв'язок з іншими підрозділами поліклініки, поліклініками та лікарнями, що відправляють хворих на відновлювальне лікування, а також закладами, що здійснюють спеціалізоване забезпечення;

5. проведення розглядів випадків необґрунтованого ведення хворих на різних стадіях лікування, неефективності проведених відновлювальних процедур та інш.;

6. здійснення комплексів заходів з підвищення кваліфікації лікарів та іншого медичного персоналу.

В залежності від потужностей лікувального закладу, а відповідно потужності відділення відновлювального лікування в його склад його можуть входити приміщення медичної фізкультури, процедурні кабінети, зали фізіотерапії, масажу, гідрокінезіотерапії, лікувальні басейни тощо.

Рентгенологічні обстеження, функціональна діагностика, лабораторні аналізи, консультації та огляди у окулістів, оториноларингологів, урологів та інших спеціалістів проводяться у відповідних підрозділах поліклініки. Для цілей лікування можуть бути використані операційні приміщення травматологічних лікарень [117].

Для кожного хворого, прийнятого у відділення відновлювального лікування, заповнюється окрема медична карта амбулаторного хворого, куди вносяться всі дані відносно лікувальних та оздоровчих процедур, що проводяться, план реабілітації, щоденник лікарських записів, дані функціональних обстежень та оглядів лікарів лікувальної фізкультури, фізіотерапії, трудотерапії тощо. Після закінчення лікування карта закривається виписним епікризом. В кожного хворого є процедурний лист, медичний персонал в хронологічному порядку відмічає всі заходи отриманого хворим лікування [118].

2.2 Індивідуальні програми реабілітації

Одним з ключових етапів відновлювальної терапії є складання індивідуальної програми відновлювального лікування хворого. В індивідуальну програму реабілітації входять наступні блоки [119]:

- анкетні дані;
- висновки медичної діагностики пацієнта;
- дані стосовно медичної реабілітації;

- дані стосовно соціально-побутової реабілітації;
- дані стосовно професійної реабілітації;
- етапний епікриз.

Під час огляду хворих лікарі визначають конкретні типи лікування (оперативне, консервативне), а також бажані та протипоказані умови праці, та конкретні трудові рекомендації .

Блок «анкетні дані» включає наступні дані про пацієнта: ім'я прізвище, по-батькові, вік, освіту, спеціальність та професію, місце роботи, адреси місць проживання та роботи. Ця інформація є необхідною для подальшої експертної діагностики, висновків, а також проведення контролю за ходом виконання індивідуальної реабілітації.

Блок «експертна діагностика і висновки» містить в собі інформацію про клініко-функціональний діагноз, прогноз захворювання, соціально-середовищну характеристику і професійний статус пацієнта. Клініко-функціональний діагноз включає в себе інформацію про форму основного та супутніх діагнозів, ускладнень основного та супутніх захворювань, а також ступінь функціональних порушень в організмі, викликаних цими хворобами [120].

В соціально-середовищній характеристиці відображається рівень матеріального забезпечення пацієнта, житлово-побутові умови, стан соціальних зв'язків пацієнта та соціально-правова поінформованість [121].

При описуванні професійного статусу пацієнта вносяться дані про професійний маршрут до моменту встановлення діагнозу та після (згідно з трудовою книжкою), рівень кваліфікації, умови і характер роботи, що виконувалася, режими праці та відпочинку.

В результаті експертної діагностики здійснюється визначення клінічного і трудового прогнозу захворювання (реабілітаційний потенціал – визначення динаміки захворювання, працездатності, соціально-середовищного та професійного статусу інваліда. Результати експертної діагностики є базисом для прийняття медико-експертного рішення про потребу в тих чи інших видах

реабілітації, пріоритетності її медичних, соціальних і професійних складових, а також встановлення етапності в проведенні реабілітаційних заходів [122].

У блоку «Медична реабілітація» описується перелік необхідних заходів та процедур, спрямованих на відновлення та компенсацію порушених або втрачених функцій організму, а також подальший розвиток здібностей пацієнта і поліпшення стану його життєдіяльності в цілому [123].

Приклад наповнення блоку медичної реабілітації наведено в табл 2.1.

Таблиця 2.1. Приклад заповнення блоку “Медична реабілітація” індивідуальної програми реабілітації пацієнта

Понеділок 2.05.2018		Відповідальний лікар	Кабінет
10:00	Тренування м'язів (мати з собою рушник)	Петренко С.А	12
11:00	Індивідуальна розмова з психологом	Іванов С.І	10
12:00	Візит до лікаря	Іванов С.І	14
13:00	Водна гімнастика	Іванов С.І	12
15:00	MS-кафе	Ткачук В.С.	10
Вівторок 13.05.2018			
09:30	MS-група: дихання і рух	Ткачук В.С.	12
11:00	Тренування м'язів (мати з собою рушник)	Романов В.І., Ткачук В.С.	14
13:00	Водна гімнастика	Ткачук В.С.	10
Середа 14.05.2018			
09:30	Водна гімнастика	Ткачук В.С.	11
10:00	Тренування м'язів (мати з собою рушник)	Романов В.І., Ткачук В.С.	22
11:30	Тренування (ходьба з палицями) - одягатися відповідно до погоди		

Продовження таблиці 2.1

13:30	Екскурсія	Ткачук В.С.	1 3
Четвер 15.05.2014			
09:30	MS-група: дихання і рух	Ткачук В.С.	1 4
11:30	Тренування (ходьба з палицями) - одягатися відповідно до погоди	Романов В.І., Ткачук В.С.	6
13:00	Індивідуальна розмова з психологом	Іванов С.І	1 2
14:00	Холестеринова група	Ткачук В.С.	2 5
15:15	Музична терапія у групах	Ткачук В.С.	1 2
П'ятниця 16.05.2018			
	Контрольне зважування щоп'ятниці з 7:00-8:00		
07:00	Водна гімнастика	Романов В.І., Ткачук В.С.	1 8
09:45	MS-група	Ткачук В.С.	1 9
14:00	Індивідуальна розмова з психологом	Коваль А.І.	1 0

Залежно від форми захворювань, важкості функціональних порушень, етапу перебігу захворювання та інших факторів призначаються ті чи інші методи лікування. Для вирішення питань надання пацієнту необхідних видів реабілітації, що не можуть бути здійснені за місцем проживання, хворий направляється в відповідні установи й організації розміщені в інших регіонах (центри реабілітації інвалідів, клініки, науково-дослідні установи тощо) [124].

Розробка індивідуальних програм реабілітації разом з їх впровадженням та наступний контролем і коректуванням вимагають значних зусиль, взаємного зв'язку виконавців, відповідальних за результати роботи на кожному етапі реабілітації. У зв'язку з цим необхідним є визначення виконавців конкретних заходів, терміни реалізації цих заходів.

Відповідно до «Положенням про індивідуальну програму реабілітації» координувати діяльність з розробки і реалізації індивідуальної програми реабілітації пацієнта повинні органи медико-соціальної експертизи.

Етапний епікриз (завершальний розділ програми реабілітації) містить поетапні висновки про хід виконання індивідуальної програми реабілітації, ефективність проведених лікувальних заходів та процедур, причини їх невиконання чи неповного виконання і визначає шляхи корекції з урахуванням змін психосоматичного, соціально-середовищного і професійного статусів хворого, а також соціально-економічних умов і можливостей реалізації індивідуальної програми реабілітації [125].

2.3. Формалізація процесу побудови план-графіку відновлювальної терапії

Беручи до уваги необхідність узгодження індивідуальних реабілітаційних планів між собою та в ресурсів клініки (побудови план-графіку), пропонується наступний підхід до вирішення задачі:

- *Крок 1:* Побудова план-графіку роботи клініки на основі індивідуальних планів реабілітації пацієнтів з урахуванням логіки та обмежень предметної області, а також специфіки роботи конкретної медичної установи (кількість доступного медичного обладнання для кожного конкретного типу процедур, максимальна кількість пацієнтів, працівників тощо)

- *Крок 2:* Оцінка якості план-графіку клініки за сукупністю категорій та їх параметрів (відгуки пацієнтів та медичного персоналу, використання виробничих можливостей клініки, розподілення навантаження робочої сили тощо)

- *Крок 3:* Оптимізація план-графіку клініки та планів відновлювальної терапії пацієнтів з метою покращення результатів оцінювання.

Узагальнену модель формування багатокритеріальної оцінки та оптимізації графіку клініки відображено на рис. 2.1.

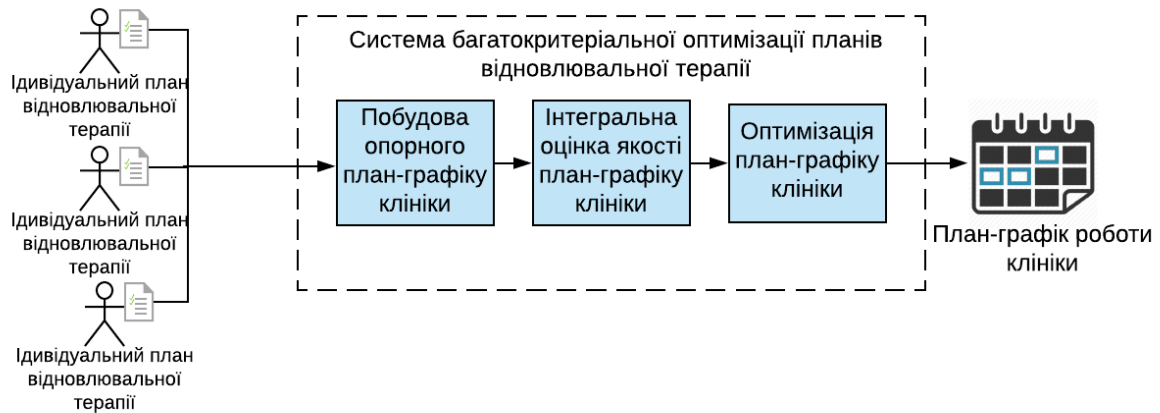


Рис. 2.1. Узагальнена модель формування, багатокритеріальної оцінки та оптимізації графіку клініки

Відповідно, задача побудови графіку відновлювальної терапії може бути подана у вигляді послідовності трьох процедур опрацювання даних: побудова опорного графіку; багатокритеріальна оцінка його якості; оптимізація графіку. Для виконання конкретної процедури доцільно виділити окрему підсистему (побудови графіку відділення, оцінювання якості графіків, оптимізації графіку). Кожна процедура повинна надавати універсальний інтерфейс роботи та інкапсулювати логіку своєї роботи [126]. Вхідні та вихідні дані для кожної з зазначених процедур наведено в табл. 2.2.

Таблиця 2. 2. Вхідні та вихідні дані процедур багатокритеріальної оцінки та оптимізації планів відновлювальної терапії.

Система	Вхідні дані	Вихідні дані
Побудова опорного план-графіку роботи клініки	Множина пацієнтів з розробленими планами відновлювальної терапії Математична модель функціонування клініки	Сформований опорний робочий план-графік клініки

Продовження таблиці 2.2

Оцінка якості роботи клініки	Сформований на першому етапі опорний план-графік роботи клініки Набір критеріїв оцінки якості план-графіку (використання виробничих можливостей клініки, виконання побажань пацієнтів, розподілення завантаженості лікарів тощо)	Оцінка план-графіку клініки за наведеною множиною критеріїв
Оптимізація графіку клініки	Результати оцінювання якості графіку на попередньому етапі Набір критеріїв для оптимізації графіку клініки	Оптимізований графік клініки (і як результат -оптимізовані плани відновлювальної терапії пацієнтів)

2.4 Метод побудови опорного план-графіку

2.4.1 Загальна схема методу

Для побудови опорного план-графіку розроблено універсальний (незалежний від виду процедур) метод призначення процедур. В його основі лежить принцип початкового розподілу найбільш критичних з них. Передбачається, що процедури в про процесі призначення беруться в порядку зниження критичності.

Під процедурою розуміється запит на проведення лікувального заходу протягом заданого часу за період планування (із зазначеним розподілом годин та дня тижня), для одного або декількох пацієнтів.

Узагальнена блок-схема процесу складання графіку клініки приведена на рис. 2.2. На першому етапі під час отримання початкової інформації проводиться розрахунок некритичності заявок. Для даного розрахунку визначається кількість допустимих часових проміжків та інших ресурсів, необхідних для проведення процедур.

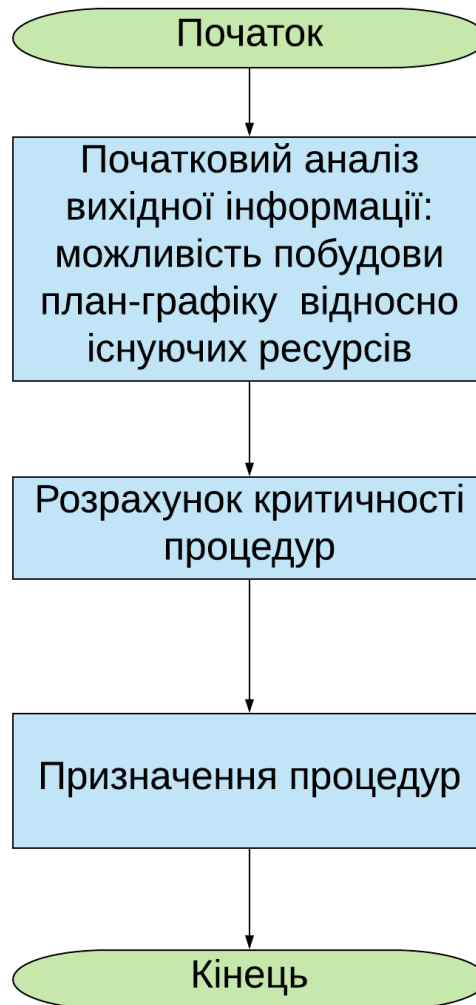


Рис. 2.2. Узагальнена блок-схема процесу складання графіку клініки.

Після формування вхідних даних та сортування заявок за некритичністю і узгодження вхідних даних за наявними ресурсами, проводиться розподіл процедур відповідно до їх некритичності і наявних ресурсів.

2.4.2 Пріоритезація процедур

В ході побудови план-графіку особлива увага відводиться визначенню критичності (пріоритетності) включення в нього процедур.

Критичність характеризується кількістю наявних можливих варіантів призначення процедур і пов'язана з частотою виникнення тупикових ситуацій. Очевидно, що найбільш критичними є заявки з жорстким призначенням.

На основі аналізу розробленої моделі виявлено, що на ступінь критичності процедур впливають такі чинники:

1. Вимога індивідуального плану реабілітації до розміщення процедури в деякий момент часу, чи вимога слідування послідовності процедур.
2. Кількість допустимих часових проміжків, протягом яких може бути проведено процедуру. Чим менше варіантів – тим критичніша процедура.
3. Кількість допустимих кабінетів, в яких може бути проведено процедуру.
4. Кількість допустимих часових проміжків для використання медичного обладнання.
5. Тривалість безперервних (спарених) процедур. Чим більший часовий проміжок, який потрібно виділити одноразово, тим менше ймовірність наявності необхідних ресурсів.
6. Розмір групи для групових процедур.
7. Загальна тривалість процедури, яка також впливає на кількість можливих варіантів призначення процедури.

Проведемо аналіз залежності ступеня критичності заявок від зазначених чинників. Критичність процедури залежить, перш за все, від того, скільки варіантів призначення процедури існує спочатку а також від того, наскільки буде знижено це значення в процесі складання план-графіку. Відношення кількості допустимих інтервалів часу проведення процедури до її тривалості показує число можливих варіантів по тимчасовому ресурсу. Кількість варіантів знижується в міру розподілу ресурсів. Швидкість зниження для різних процедур є різною.

Для вищої наочності в ході побудови план-графіку використано зворотну величину – некритичність процедури (J), яку було визначено за наступною формулою:

$$J = \frac{V * W}{D * q1 * R * q2 * F}, \quad (2.1)$$

де V – кількість допустимих часових ресурсів;

W – допустимі ресурси;

D – загальна тривалість процедури;

F – тривалість неперервного заняття;

$q1, q2$ – коефіцієнти корегування ваги факторів.

Чим більше значення, що отримується для заявки за цією формулою, тим меншою є її критичність. Як показали дослідження, величини розмірів груп і тривалості безперервних процедур можуть в середньому набувати різних значень. Вплив кожного з факторів залежить від конкретно сформованих умов. Через це можливі різні варіанти впливу на ступінь критичності процедури. Найбільш доцільним є введення коефіцієнтів ($q1, q2$), що сприяє збільшенню чи зменшенню ваги того чи іншого фактора.

2.4.3 Призначення процедури

За своїм принципом пошук необхідних ресурсів для призначення процедури можна розділити на два основних типи - пошук призначення для регулярних і нерегулярних процедур. Регулярними називаються процедури, що проводяться на певній регулярній основі. Нерегулярними, відповідно, ті, що не мають певного наперед визначеного фіксованого план-графіку.

Суть пошуку призначення для процедур з нерегулярним розподілом полягає в наступному. Спочатку проглядаються частково зайняті часові діапазони. Потім перебираються варіанти призначення процедури в порядку зменшення їх переваг.

З метою раціонального розподілу часового ресурсу в першу чергу проглядаються миготливі інтервали, вільні і допустимі відносно жорстких обмежень, найбільш бажані (в сенсі заданих вимог).

Під час пошуку призначення для процедур з регулярним розподілом проглядаються найбільш бажані і допустимі інтервали необхідної довжини. Після знаходження необхідного інтервалу в ньому проводиться пошук вільного устаткування, кабінетів та медичного персоналу. Потім проглядаються інші інтервали та необхідні ресурси, які є менш бажаними чи прийнятними.

2.4.4 Вирішення конфліктів

Конфліктна ситуація в план-графіку виникає тоді, коли неможливо здійснити призначення поточної процедури, іншими словами дана процедура конфліктує з уже призначеними.

Виходом із такої ситуації можуть стати:

1. Пошук переносів уже призначених процедур для запису на їх місце поточної процедури.
2. Послаблення або ігнорування вимог поточної процедури та подальшого пошуку можливостей її призначення.
3. Відкладання поточної процедури для її призначення після основного розподілу всіх заявок.

Якщо в результаті первинного пошуку і можливого деякого часткового послаблення обмежень не вдалося знайти можливості для призначення процедури (процедуру неможливо призначити без переносу уже призначених), виконується спроба переносу призначених процедур і запису поточної.

Суть переносу полягає в тому, що для деякої вже призначеної процедури відбувається пошук всіх допустимих часових проміжків та ресурсів, окрім

поточних. Для прискорення пошуку перепризначень використовуються найменш критичні призначені процедури.

Перенесення можна розділити на:

- перенесення процедури на інший час протягом дня чи часового проміжку;
- перенесення в інші кабінети;
- перенесення на інший час та інші кабінети.

Порядок пошуку перепризначень і ослаблення обмежень можна змінювати, а сам пошук також додатково налаштовувати, що в кінцевому підсумку збільшує час пошуку і, відповідно, якість план-графіку або навпаки. Пошук перепризначень можна налаштовувати від так званого "жадібного" пошуку, що шукає перші допустимі варіанти перепризначень і ігнорує вимоги, до перебирання великого числа варіантів відповідно до принципу "зменшення пріоритетів". Він полягає в послідовному переборі спочатку найкращих варіантів перепризначень, серед усіх допустимих для перенесення, потім менш прийнятних і т.д.

Якщо після перестановок вже призначених процедур не вдалося знайти призначення для поточної заявки, проводиться ослаблення вимог. Запропонований алгоритм дозволяє отримати варіант план-графіку навіть у випадку нестачі ресурсів або суперечливості заданих вимог, тому що в ньому передбачено послаблення вимог і запис заявок, призначення яких неможливо, в список відкладених заявок. Тобто існує можливість отримання так званих квазіоптимальних план-графіків [127], тобто план-графіків складених з певними відступами щодо за даної системи вимог. Послаблення вимог відбувається у відповідності з налаштуваннями, що можуть змінюватися.

Щодо практичного використання алгоритму, можна відзначити також те, що для прискорення генерації план-графіку на першому етапі може використовуватися налаштування процесу складання, аж до однопрохідного (без пошуку ланцюжків перепризначень), і без процедури послаблення вимог. Що дозволяє відразу виключити деякі помилки і завищені вимоги, тобто обійтися без

тривалого непродуктивного початкового пошуку, отримавши в результаті безліч непризначених процедур, а потім після узгодження всіх вимог та наявних ресурсів добудувати і оптимізувати план-графік.

2.4.5 Підбір часу для проведення процедури

Призначаючи чергову процедуру, необхідно вибирати час її проведення з проміжків, що залишилися допустимими після виконаних попередніх призначень. Проміжок часу є допустимим для проведення конкретної процедури у випадках, коли виконуються жорсткі обмеження: в даний час вільні всі відповідні пацієнти (пацієнт) і медичний персонал; є вільні кімнати, які підходять для проведення даної процедури. В рамках даної роботи як жорсткі також розглядаються: обмеження на час проведення процедур; наявність медичного персоналу; режим роботи кабінетів; вимоги до розподілення навантаження; вимоги слідування індивідуальному плану реабілітації пацієнта. Проміжки часу, що задовольняють повний набір жорстких обмежень, утворюють множину допустимого часу проведення даної процедури.

Метод рішення задачі полягає у виборі потрібного часу проведення процедур. Вибираючи його цілеспрямовано, можна добиватися різних результатів: компактного план-графіку для пацієнтів чи медичного персоналу, економного використання ресурсів тощо. Виходячи з пріоритетів осередків план-графіку для проведення процедур пацієнтам, можна визначити найбільш підходящий час проведення кожного конкретного виду для конкретного індивіда (індивідуалізація плану відповідно до побажань пацієнтів).

2.5 Скінченний автомат побудови план-графіку

Завдання складання початкового план-графіку можна розглядати як задачу розподілу наступних ресурсів – медичного персоналу, пацієнтів, лікувальних кабінетів та обладнання. Процес побудови опорного план-графіку доцільно

представити у вигляді скінченного автомата – абстракції, що використовується для характеристики шляхів зміни стану об'єкта, в залежності від досягнутого стану та зовнішньої вхідної інформації [128, 129].

2.5.1 Аналіз та вибір типу абстрактного автомата

Загальну схему автомата можна інтерпретувати як “чорну скриньку” [130], що здійснює перетворення вектору вхідних даних у вектор вихідних (рис 1.):

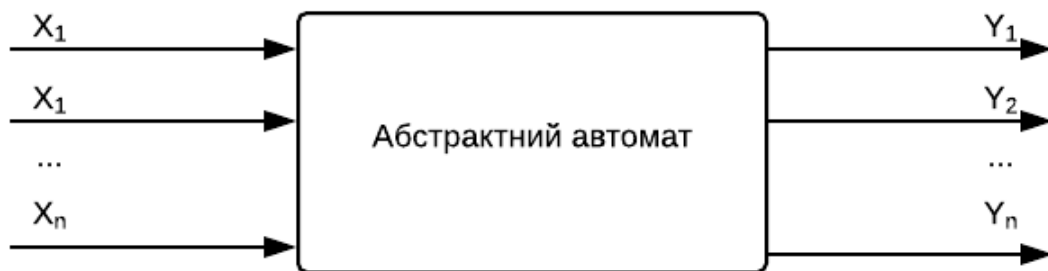


Рис. 2.3. Загальна схема абстрактного автомата

Математичну модель абстрактного автомата можна описати наступним чином [131]:

$$A = (X, Y, S, f_y, f_s, S_0), \quad (1)$$

де X – множина вхідних даних автомата;

Y – множина вихідних даних автомата;

S – множина допустимих станів автомата;

f_y – функція виходів автомата;

f_s – функція переходів з одного стану автомата в інший;

S_0 – початковий стан автомата.

З метою класифікації автоматів розглядають низку ознак, таких як визначеність функції переходів і функції виходів, однозначність заданих функцій, стійкість станів, скінченність множин вхідних, вихідних даних та станів [132]

За визначеністю характеристичних функцій автомат побудови план-графіку клініки є визначеним, оскільки всі $(s_i, x_k) \in X \times S$, де $s_i \in S, x_k \in X$, тобто характеристичні функції є визначеними для всіх пар вхідних даних та можливих станів.

За однозначністю функції переходів автомат є детермінованим, оскільки під дією деякого вхідного сигналу $x_k \in X$ автомат може перейти тільки в один стан $s_j \in S$. Механізм переходів є повністю визначеним, ймовірнісний перехід є неможливим.

За стійкістю станів автомат є стійким, якщо під дією деякого вхідного сигналу $x_k \in X$ було здійснено перехід у стан $s_j \in S$, то вихід з нього і перехід в інший стан можливий тільки при надходженні на вхід автомата іншого сигналу $x_z \in X, x_z \neq x_k$.

За скінченністю станів, вхідних та вихідних множин автомат є скінченним, оскільки множини X, Y, S є скінченними. $|X| \neq \infty, |Y| \neq \infty, |S| \neq \infty$.

Базовими для побудови скінченних, визначених, детермінованих, стійких автоматів є дві моделі - автомат Мілі та автомат Мура [133, 134].

Специфікуючи базову модель, автомат Мілі можна доповнити наступним відношеннями:

$$\begin{aligned} s(t+1) &= f_s(x(t), s(t)), \\ y(t) &= f_y(x(t), s(t)) \end{aligned} \quad (2.1)$$

де $s(t+1)$ – наступний стан, в який перейде автомат;

f_s – функція переходу в наступний стан автомата;

$x(t)$ – поточне значення вхідного сигналу;

$s(t)$ – поточний стан автомата;

$y(t)$ – значення виходу автомата;

f_y – функція виходу автомата.

З рівностей (2.1) видно, що у випадку автомата Мілі аргументами характеристичних функцій є поточне значення вхідного сигналу та поточний стан.

Автомат Мура можна означити наступним чином:

$$s(t + 1) = f_s(x(t + 1), s(t)), \quad (2.2)$$

$$y(t) = f_y(s(t)),$$

де $s(t + 1)$ – наступний стан в який перейде автомат;

f_s – функція переходу в наступний стан автомата;

$x(t)$ – поточне значення вхідного сигналу;

$s(t)$ – поточний стан автомата;

$y(t)$ – значення виходу автомата;

f_y – функція виходу автомата;

З відношень (2.2), випливає, що вихідний сигнал автомата однозначно визначається його поточним станом та не залежать від компонентів вектору вхідних сигналів .

З урахуванням специфіки предметної області, а саме того, що результат роботи автомата безпосередньо залежатиме від типу процедури, яку автомат отримає на вході для включення, в свою чергу включення процедури в план-графік відбуватиметься залежно від її характеристик та в залежності від поточного стану план-графіку, для побудови кінцевого автомата роботи клініки доцільним є використання саме моделі Мілі.

2.5.2 Побудова функціональної моделі скінченного автомата Мілі

Словесно процес роботи автомата побудови план-графіку клініки можна задати як послідовність наступних етапів [135]:

Етап 1. Отримання сукупності процедур для побудови план-графіку.

Етап 2. Початковий аналіз процедур на предмет побудови план-графіку за наявних обмежень та доступних ресурсів. У випадку неможливості побудови план-графіку за наявних процедур та обмежень автомат закінчує свою роботу. Для продовження роботи потрібно переглянути жорсткі обмеження, які накладаються на план-графік.

Етап 3. Розрахунок некритичності процедур відносно наявних ресурсів. Детально процес розрахунку некритичності наведено в розділі 2.4.1.

Етап 4: Сортування процедур за зростанням некритичності. З процедур формується пріорітезована черга в порядку зростання їх некритичності для побудови план-графіку.

Етап 5: Вибір найбільш критичної процедури з черги, для наступного включення в план-графік. Процес вибору процедури включає в себе послідовний аналіз на предмет задоволення множини критеріїв.

Етап 6. Пошук доступних часових проміжків для проведення процедури на основі результатів отриманих на *етапі 5*. У випадку відсутності вільних проміжків відбувається перехід на *етап 10* з наступним пошуком можливих перепризначень уже включених в план-графік процедур.

Етап 7. Пошук доступного матеріального забезпечення, а також вільного персоналу клініки для проведення процедури, вибраної на *етапі 5*, а також впродовж часових проміжків сформованих на *етапі 6*. У випадку відсутності вільних проміжків відбувається перехід на *етап 10* з наступним пошуком можливих перепризначень уже включених в план-графік процедур.

Етап 8. Аналіз можливих варіантів розміщення процедури в план-графіку на основі даних, отриманих в ході *етапів 6-7*, перевірка виконання обмежень предметної області.

Етап 9. Безпосередньо включення процедури в план-графік та перехід до *етапу 3*.

Етап 10. Пошук процедур, що можна перепризначити для включення поточної в план-графік. Після їх знаходження здійснюється вибір найменш критичної процедури, перепризначення їх на один з доступних проміжків та призначення поточної на її місце.

Етап 11. У випадку, коли не вдалося знайти процедуру для перепризначення, відбувається послаблення вимог до поточної та наступна спроба включення її до план-графіку.

Діаграма станів побудови опорного план-графіку клініки, що передбачає виконання запропонованих етапів 1..11 зображена на рис. 2.4.

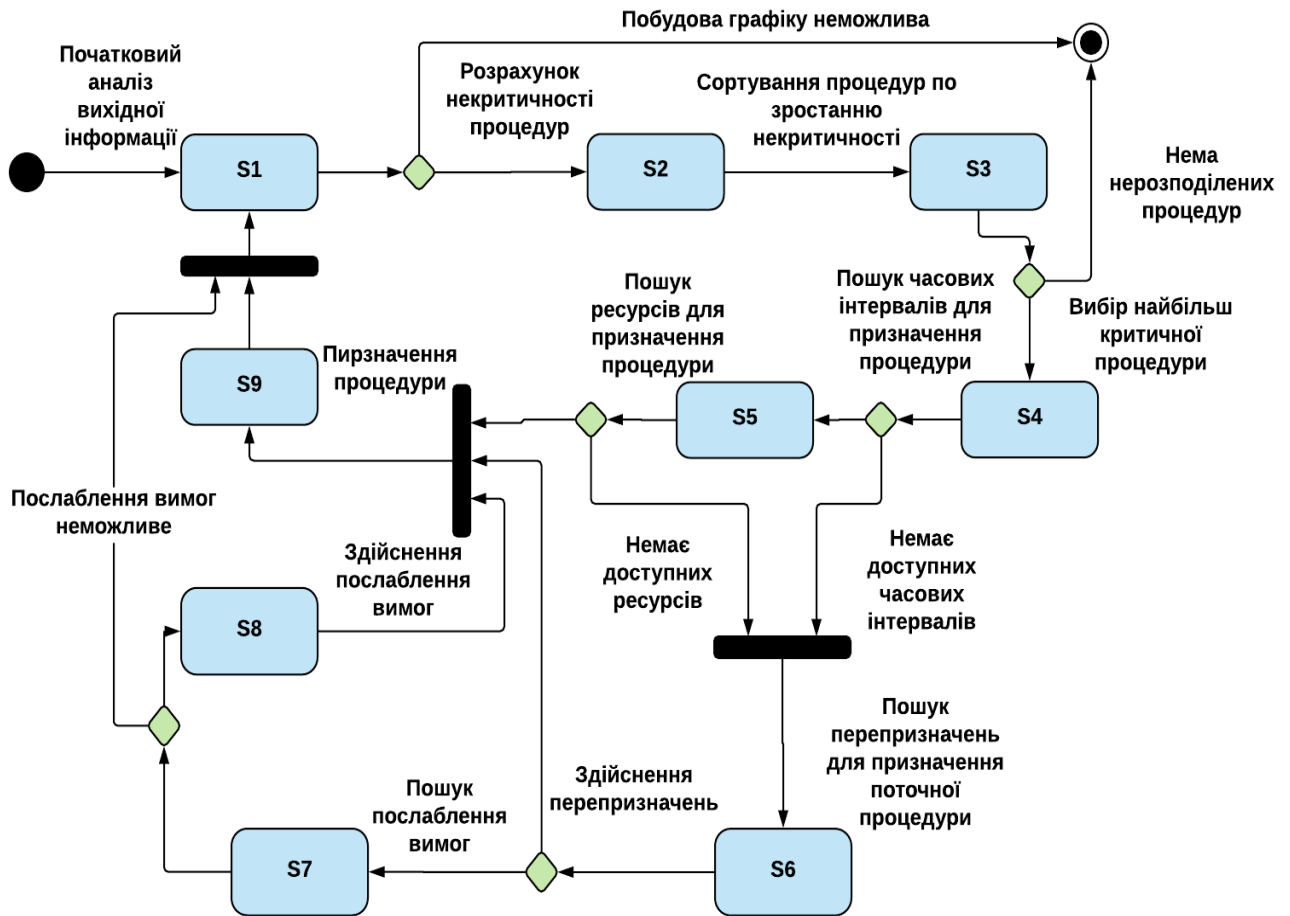


Рис. 2.4. Діаграма станів побудови опорного план-графіку клініки

Для автомата множина вхідних даних $X = \{x_0, x_1 \dots x_n\}$, – множина вхідних даних автомата; $Y = \{y_0, y_1 \dots y_n\}$ – множина вихідних даних автомата; $S = \{s_0, s_1 \dots s_n\}$ – розширена множина допустимих станів автомата; $f_y: S \rightarrow Y$ – функція виходів автомата; $f_s: X \times S \rightarrow Y$ – функція переходів з одного стану автомата в інший; $s_0 \in S$ – початковий стан автомата.

Стосовно досліджуваної предметної області множина вхідних даних X абстрактного автомата складається з набору лікувальних процедур, які формують план-графік. Множина вихідних даних Y є результуючим набором процедур які складають план-графік. Множина станів S визначається алгоритмом

функціонування автомата, містить одинадцять елементів та має вигляд $S = \{s_0, s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}\}$.

Кожна процедура y_n характеризується наступним кортежем:

$$y_n = (d, m, z, l, t_s, t_e), \quad (2.3)$$

де $d \in D$ – ідентифікатор пацієнта, для якого проводиться процедура;

$m \in M$ – ідентифікатор обладнання, необхідного для проведення процедури;

$z \in Z$ – ідентифікатор процедурного кабінету, де проводиться процедура;

$l \in L$ – ідентифікатор медичного персоналу, що проводить процедуру;

t_s – час початку процедури;

t_e – час закінчення процедури.

Функції переходів f_s між станами кінцевого автомата реалізуються у вигляді булевих функцій та визначаються окремо для кожного стану.

На початку своєї роботи автомат отримує вхідні дані та переходить в початковий стан (S_0).

Далі обчислюються загальний об'єм ресурсів, необхідних для проведення отриманого набору процедур та проводиться перевірка реалістичності побудови план-графіку. У випадку, коли наявні ресурси за наявних обмежень не є достатніми для побудови план-графіку, автомат виконує перехід в стан S_{10} та закінчує свою роботу. В протилежному випадку побудова план-графіку продовжується та здійснюється перехід в стан S_1 .

Після цього для кожної з процедур з набору виконується розрахунок некритичності та здійснюється перехід в стан S_2 . Детально процес розрахунку некритичності наведено в розділі 2.4.1. Далі проводиться пріоритезація процедур на основі сформованих оцінок S_3 .

На наступному кроці проводиться вибір процедури з найнижчим значенням некритичності для її включення до план-графіку та здійснюється перехід в стан S_4 .

Далі автомат проводить пошук часових проміжків, придатних для призначення процедури, а також необхідного медичного персоналу та обладнання для її проведення (S_5).

У випадку, якщо таких проміжків не знайдено, серед уже призначених процедур здійснюється пошук можливих перепризначень (S_6). Серед знайдених процедур переноситься найменш пріоритетна, а на її місце призначається поточна. Коли немає жодної процедури, що можна перепризначити, здійснюється перехід в стан S_7 та виконується спроба послаблення жорстких обмежень накладених на процедуру. В результаті автомат переходить в стан S_8 . Якщо в результаті послаблення обмежень вдається знайти доступні часові проміжки або в стан S_9 в протилежному випадку.

Маючи інформацію з попереднього кроку, проводиться аналіз часових проміжків, коли весь необхідний інвентар та медичний персонал доступні. Для кожного з знайдених часових фреймів проводиться аналіз задоволення обмежень предметної області. У випадку успіху автомат переходить в стан S_{10} .

Після знаходження відбувається вибір часового проміжку та устаткування на основі задоволення обмежень предметної області, запис процедури до плану графіку та включення її до вихідної множини даних (перехід в стан S_1).

Оскільки часткове задоволення жорстких обмежень (обмежень предметної області) є неможливим, умови переходів між станами кінцевого автомата реалізуються у вигляді булевих функцій, та визначаються окремо для кожного стану.

ВИСНОВКИ ДО РОЗДІЛУ 2

1. Беручи до уваги необхідність узгодження індивідуальних реабілітаційних планів та між собою та ресурсів клініки, запропоновано підхід до вирішення задачі багатокритеріальної оптимізації план-графіку її роботи, що складається з трьох основних етапів: побудова опорного план-графіку, що задовольняє жорстким обмеженням, оцінка якості план-графіку, послідовна оптимізація план-графіку з метою підвищення результатів оцінки.

2. Для побудови опорного план-графіку запропоновано універсальний метод, що забезпечує лінійну залежність часу вирішення відносно об'єму задачі. Метод ґрунтується на визначенні пріоритетності включення процедур до план-графіку та дозволяє врахувати вимоги індивідуального плану реабілітації, наявність необхідного кваліфікованого персоналу для проведення процедур, доступність інвентаря тощо. Розроблено механізми аналізу можливості побудови план-графіку за умови наявності певних множин планів реабілітації, вирішення конфліктних ситуацій при включенні процедур в план-графік, а також методи послаблення певних жорстких вимог при неможливості побудови план-графіку.

3. Для вирішення задачі побудови опорного план-графіку використано базову модель – скінченний автомат Мілі, що дозволяє найточніше врахувати специфіку досліджуваної предметної області. Розроблено скінченний автомат що реалізує послідовний метод побудови опорного план-графіку, а також реалізує механізми аналізу можливості побудови план-графіку за умов присутності певної множини обмежень, вирішення конфліктних ситуацій в план-графікі, та уможливорює послаблення деяких заданих жорстких обмежень при тупикових ситуаціях в процесі роботи.

РОЗДІЛ 3. РОЗРОБКА АРХІТЕКТУРИ СИСТЕМИ ОПТИМІЗАЦІЇ ПЛАНІВ ВІДНОВЛЮВАЛЬНОЇ ТЕРАПІЇ НА ОСНОВІ НЕЙРОНЕЧІТКИХ МОДЕЛЕЙ.

3.1. Методика оцінки та порівняння різних варіантів план-графіків з використанням нечіткої логіки

3.1.1. Принципи побудови систем нечіткої логіки

Для оцінки ефективності план-графіків можна застосувати кількісні методи, які враховуватимуть ступінь використання наявних в них ресурсів - час роботи чи простою медичного обладнання, завантаженість лікарів тощо. Проте ці класичні кількісні методи не враховують індивідуальних властивостей пацієнтів, що в результаті виражаються рівнем задоволення наданих йому лікарських послуг, та подаються в лінгвістичній формі.

Використання такого підходу дозволяє отримати максимально точну оцінку з точки зору пацієнта, але отримати результати важко компонувати і опрацювати (вибудувати в статистику). Окрім цього, медичні установи часто зацікавлені в агрегації результатів опитувань протягом певних проміжків часу. В результаті, виникає потреба перетворення лінгвістичних даних в кількісні показники [136].

В дослідженнях системи нечіткого логічного висновку часто називають контролерами нечіткої логіки [137]. Вперше вони були використані в процесах промисловості, і успішно застосовуються в даний час. [138].

Замість використання, класичних математичних моделей для опису системи, контролери нечіткої логіки послуговуються інтегрованими експертними знаннями. Знання в свою чергу представляються за допомогою близьких до розмовної мови структур, що описуються за допомогою лінгвістичних змінних та нечітких множин. Система нечіткої логіки здійснює трансформацію вхідних даних, слідуючи певному методу нечіткого логічного висновку [139].

Загальна структура контролера нечіткої логіки (рис. 3.1) включає в себе наступні елементи [140]:

- блок фазифікації;
- базу правил нечіткої логіки;
- блок прийняття рішень;
- блок дефазифікації.

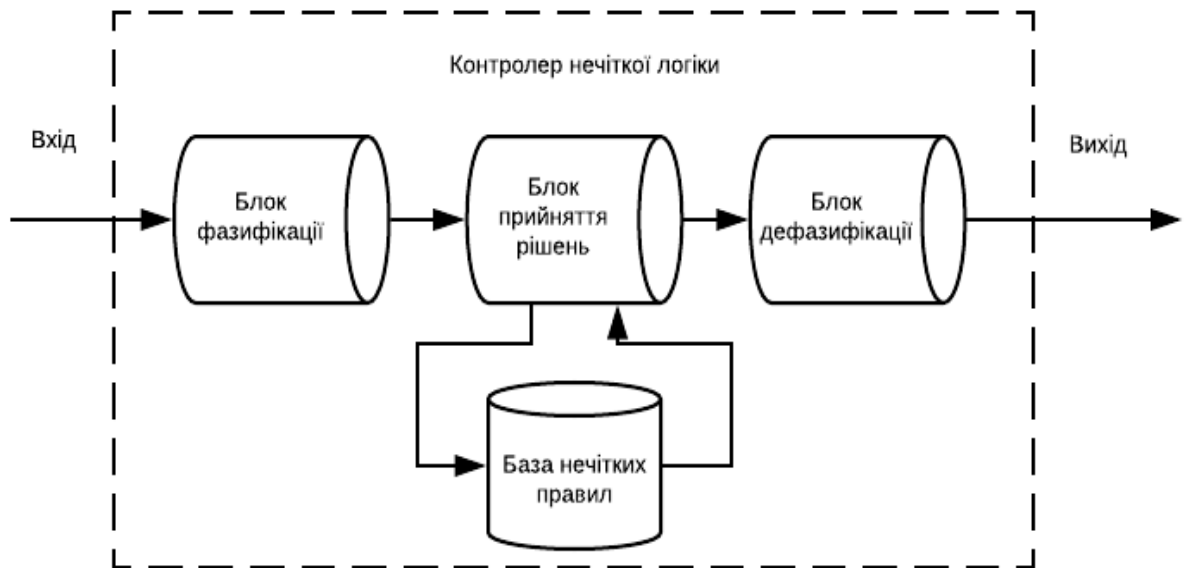


Рис. 3.1. Загальна структура контролера нечіткої логіки

Блок фазифікації є відповідальним за перетворення будь-якого чіткого вхідного значення в коректне значення в просторі нечіткої логіки. Нечіткі множини є розширенням класичних, де можливою є тільки повна належність елементів множині або її відсутність. Вони допускають також часткову належність елементів. Це забезпечується кількісним градуванням від помилкового до істинного (від 0 до 1) з використанням функцій належності [141]. Функція належності $\mu_X(x)$ пов'язує кожен елемент $x \in U$ з числом в інтервалі $[0 \dots 1]$, як показано в (3.1):

$$\mu_X: U \rightarrow [0..1] \quad (3.1)$$

Тобто, блок фазифікації виконує трансформацію даних поданих в чіткому вигляді $x \in U$ в нечітку множину $X \in U$ і $\mu X(x)$ задає ступінь належності в діапазоні $[0 \dots 1]$.

Можливими є два крайніх випадки. Якщо $\mu X(x) = 0$, то $x \notin X$ і в свою чергу $\mu X(x) = 1$, то $x \in X$ в класичному розумінні. Та в випадку $\mu X(x) = 0,4$ x належить X лише зі ступенем 0,4, або ж x не належить X зі ступенем 0,6.

Множину X разом з визначеною для неї функцією належності $\mu X(x)$ називають нечіткою та позначають $(X, \mu X)$. Процес трансформації певного чіткого значення (наприклад, $x = 0,6$) в нечітку множину (припустимо, $x = 0,6 \in X = [0, 1]$, $\mu X(x) = 0,6$) називають фазифікацією [142].

База правил нечіткої логіки. Більшість систем нечіткої логіки побудовані на основі знань. Тобто їх нечіткі моделі, описуються за допомогою продукційних правил формату «ЯКЩО... ТО...», які в свою чергу встановлюються та верифікуються на основі експертного досвіду людини про систему та реалізуються за допомогою виконання строгих логічних операцій.

Набір продукційних правил становить основу бази знань нечіткої логіки для досліджуваної задачі [8].

У загальному вигляді база правил R з кількістю складових k має вигляд:

$$R(k): \text{IF } x_1 \text{ is } X_{k1} \text{ AND } \dots \text{ AND } x_m \text{ is } X_{km} \text{ THEN } u \text{ is } U_k, \quad (3.2)$$

де $m \geq 1$ и $k > 1$.

У контексті предметної області медицини спрощена база правил R може мати наступний вигляд:

R (1): ЯКЩО поведінка лікарів є невивічливою, ТО рівень оцінки лікування низький;

R (2): ЯКЩО пояснення пацієнтам рішень медичного персоналу траплялося іноді, ТО рівень оцінки лікування середній;

R (3): ЯКЩО якість ліків відмінна, ТО рівень оцінки лікування високий.

Звісно, даний опис описує лише основну ідею, що не є повною і ефективною конструкцією для побудови інтегральної оцінки.

Блок прийняття рішень. Нечіткий логічний висновок здійснюється за допомогою продукційних правил, що складають базу знань. Нечітке введення перетворюється в управляючі дії, що також мають нечітку природу. Кількість продукційних правил залежить від кількості входів контролера та від функцій належності пов'язаних з кожним входом [143].

Після обробки вхідних даних за допомогою нечіткої логіки здійснюється їх зворотнє перетворення до чітких значень. Цей процес здійснюється **блоком дефазифікації**. Іншими словами блок дефазифікації математично об'єднує результати кожного правила в єдине чітке значення. Розроблено низку методів для виконання таких перетворень.

Методи максимуму. Найпоширеніший метод максимуму ґрунтується на виборі середнього значення серед множини елементів з максимальними ступенями істинності (метод MOM). Його перевага – простота та швидкість виконання. У випадку розгляду тільки елементів з найвищими ступенями приналежності, інформація, не пов'язана з правилами максимальної активації, ігнорується [144].

Методи, засновані на області. Одним з часто використовуваних методів заснованих на області, є вибір результату дефазифікації як центру ваги методу розподілу COG з ймовірностями виходу. Близьким аналогом є метод центру області (COA), що здійснює вибір чіткого значення як позиції, де область виведення може бути розділена навпіл. Якщо розподіл виходу є симетричним, обидва методи дають однакові результати [145].

Інші методи. На основі моделі COG розроблено низку алгоритмів навчання для усунення дефектів виведення. Загальною ідеєю цих методів полягає в виконанні деякого перетворення розподілу ваг виведення відповідно до генерованого набору параметрів. Загальну функцію перетворення нечіткого члена X можна подати наступним чином:

$$v_x(x) = \mu(x)T(x), \quad (3.3)$$

де $\mu(x)$ – нове значення функції належності елемента;

$x \in X, \mu_x(x)$ - оригінальне функції належності елемента;

$T(x)$ - довільна функція трансформації.

Ось деякі приклади таких функцій:

BADD метод [146]:

$$T(x) = \mu x(x)^{y-1}, \quad (3.4)$$

де y – автоматично визначений параметр.

SLIDE метод [146]:

$$T(x) = \begin{cases} 1 - \beta & (x \leq \alpha) \\ 1 & (x > \alpha) \end{cases} \quad (3.5)$$

де α і β – параметри, що необхідно визначити.

M -PTD метод [148]:

$$T(x) = \left[\sum_{j=0}^N \beta_j (\mu_x(x) - 0.5)^j \right]^2, \quad (3.6)$$

де β_j –регульовані параметри;

$\mu_x(x)$ и $T(x)$ – дискретні функції.

У всіх перерахованих вище методах вид перетворення, яке викликається $T(x)$, є спотворенням розподілу ймовірностей виходу, що збільшує чи зменшує ваги елементів з більш високими значеннями належності. Чим більше трансформація збільшує відмінності між низькими і високими значеннями належності, тим більше процес дефазифікації буде апроксимувати характеристику MOM [149].

3.1.2. Аналіз існуючих підходів до побудови систем нечіткого логічного висновку

Серед традиційних алгоритмів систем нечітких висновків найбільш відомими та популярними є алгоритми Mamdani і Takagi-Sugeno, які, зокрема, можуть бути використані для реалізації системи оцінки план-графіків [150].

Основними перевагами методу Мамдані є відсутність стандартів побудови правил та здійснення нечіткого логічного висновку, що ґрунтується на експериментах та експертному досвіді. Це дає можливість описання цього досвіду більш інтуїтивно та аналогічно людському підходу. Алгоритм Мамдані також має такі наступні загальні риси [150]:

- застосовується в випадку, коли бази числових даних є неповними та можуть бути доповнені експертними знаннями;
- в залежності від методу дефазифікації одні і ті ж правила дають різні результати (центру маси, мінімуму, перший максимуму тощо);
- припущення щодо вибору функції дефазування засноване лише на експериментах.

Система нечіткого логічного виведення Mamdani базується на наступній формі правил:

$$\mathbf{R: IF X IS A AND X IS B THEN Y IS C,} \quad (3.7)$$

де R – дане правило,

X – вхідні дані,

A, B, C – лінгвістичними змінні,

Системи нечіткого логічного висновку Takagi-Sugeno використовуються тоді, коли надається достатня кількість базових числових даних. Вони ефективні у розрахункових завданнях і добре поєднуються з оптимізацією та адаптивними

методами, що робить їх широко використовуваними у задачах управління та оцінки. Якщо параметри моделі варіюються в залежності від конфігурації, характеру чи розміру навантаження, то використання такої системи нечітких висновків є непрактичним, через потребу створення окремих моделей для кожної з ділянок розділу для формування відповідної керуючої дії, що може суттєво вплинути на продуктивність системи [152, 153].

Як результат, традиційні системи нечіткого логічного висновку мають низку негативних факторів, які впливають на можливості та доцільність їх застосування:

- необхідність компромісу між хорошою інтерпретативністю та хорошою точністю;
- мала кількість вхідних змінних;
- часозатратне налаштування;
- висока залежність від методу дефазифікації – ті ж правила дають різні результати в залежності від методу;
- необхідність наявності достатньої бази даних.

Всі ці фактори обмежують використання існуючих нечітких систем для задач керування та оцінки багатокomпонентних систем. В зв'язку з цим виникає необхідність використовувати іншу систему, яка компенсуватиме їх недостатки.

Однією з недавно представлених систем, які компенсують більшість недоліків традиційних нечітких систем висновків, є T-Controller [153].

Система T-Controller є оригінальним методом нечіткої логіки. Вона заснована на концепції моделювання геометричних даних.

T-Controller – нечіткий контролер – спосіб створення логічних рішень для нечіткої логічної системи. Він використовує принципово новий метод дефазифікації, що дозволяє отримувати нульову методичну помилку.

T-Controller має низку переваг перед традиційними системами нечіткого логічного висновку:

- логічний висновок і композиція об'єднуються в один конкретний крок;
- кількість правил обумовлена функціями лише вихідних змінних;

- швидка дефазифікація;
- висока точність Т-контролера - високошвидкісний геометричний метод дефазифікації з нульовою методичною помилкою (точніший «вхід» дає точний «вихід»);
- процедура налаштування відбувається швидше;
- простота виконання як в програмній, так і в апаратній версії.
- процедура проектування правил є інтуїтивно зрозумілою для експертів шляхом аналізу можливого ситуації для вихідної змінної.

Т-Controller використовує наступну форму правил:

$$R: \text{IF } X \text{ IS } A \text{ AND } X \text{ IS } B \text{ THEN } Y \text{ IS } C, \quad (3.8)$$

де R - це номер правила;

X - це номер введення;

A, B та C - лінгвістичні змінні, ключове слово I;

S позначає а пропозиція, ключове слово;

I позначає кон'юнкцію;

3.1.3. Розробка контролера нечіткої логіки оцінки оптимальності план-графіків лікувальних закладів

Для оцінки план-графіку лікувального закладу визначено п'ять критеріїв, що впливають на його якість з точки зору пацієнтів та медичного персоналу лікувальних установ:

- достатні для відновлення пацієнтів перерви між процедурами (не повинні бути коротшими 15 – 40 хвилин для кожного конкретного типу процедур);
- рівномірний розподіл навантаження на медичний персонал (відсутність вікон в план-графіку для медичного персоналу, триваліших за 45 хвилин);

- максимізація завантаження обладнання (менше 10 - 25% часу простою обладнання, призначеного для проведення кожного конкретного типу процедур);
- комфортний розмір груп для проведення процедур (не більше 95% максимально можливого розміру груп);
- комфортний режим проведення процедур для пацієнтів (процедури займають не більше 85% часу пацієнта протягом дня).

В якості результуючого узагальненого критерію виступає власне оцінка якості план-графіку функціонування медичного закладу.

На першому етапі проектування контролера нечіткої логіки на основі вхідних критеріїв формуються нечіткі лінгвістичні змінні. Для цього введено показники, засновані на штрафах, що встановлюються кожному критерію за незручні моменти в план-графіку. Значення порушень нормалізуються за наступною формулою [155]:

$$K_i^S = \frac{K_i^S - K_i^{min}}{K_i^{max} - K_i^{min}} \quad (3.8)$$

де K_i^S – поточне значення i -того критерія порушень;

K_i^{max} – максимальне можливе значення i -того критерія порушень;

K_i^{min} – мінімальне можливе значення i -того критерія порушень;

$K_i^{min} \leq K_i^S \leq K_i^{max}$;

$k_i^S \in [0,1]$.

В даному випадку використано використано наступні вхідні лінгвістичні змінні (Порушення–Режиму–Процедур, Порушення–Перерв–Між–Процедурами, Порушення–Розміру–Груп, Порушення–Розподіл–Навантаження–Персоналу, Порушення–Максимуму–Завантаження–Обладнання) і одна вихідна (Якість–Графіку) (рис. 3.2).

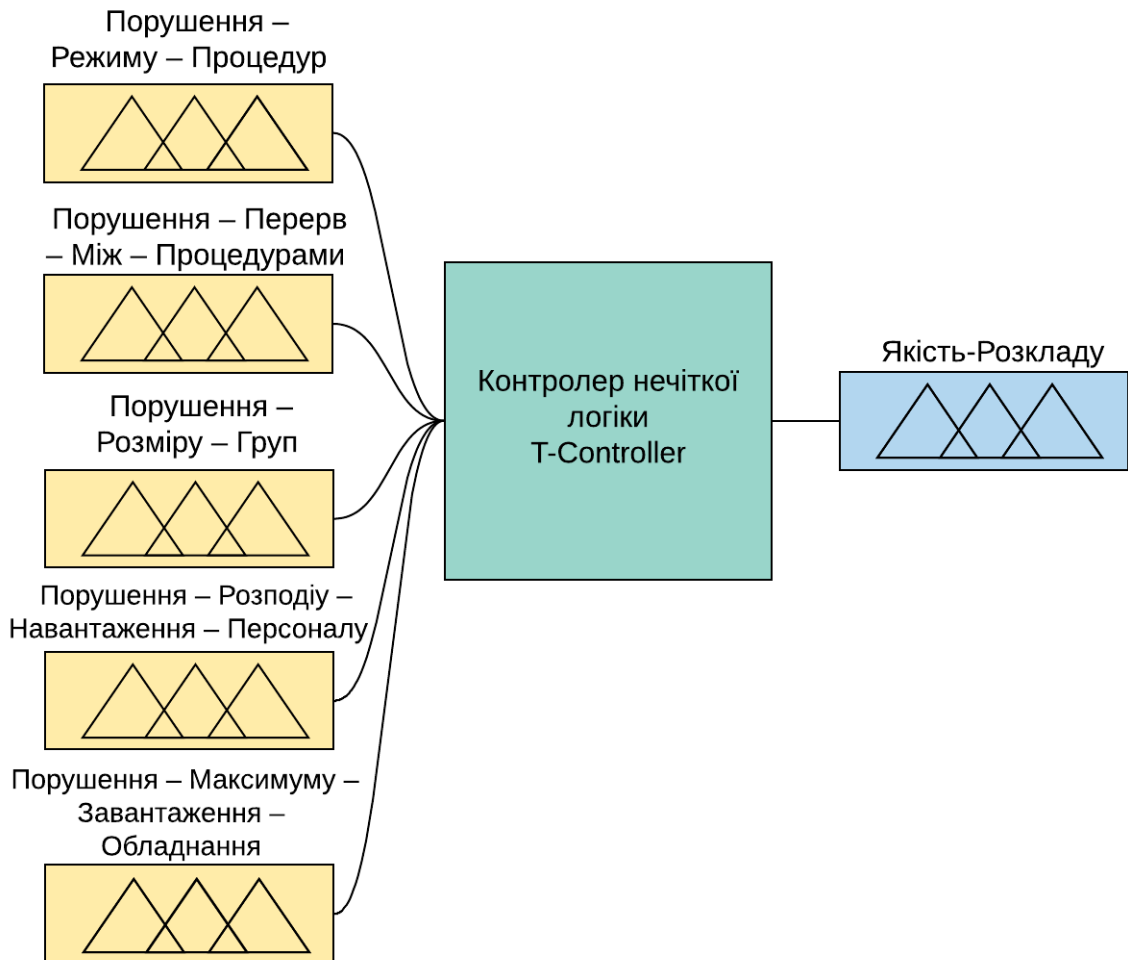
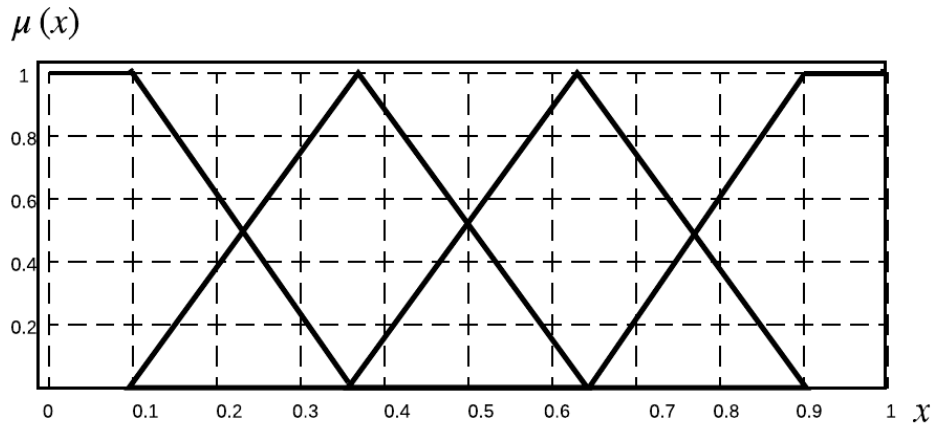


Рис. 3. 2. Структура системи нечіткої логіки для інтегральної оцінки якості план-графіку функціонування лікувального закладу

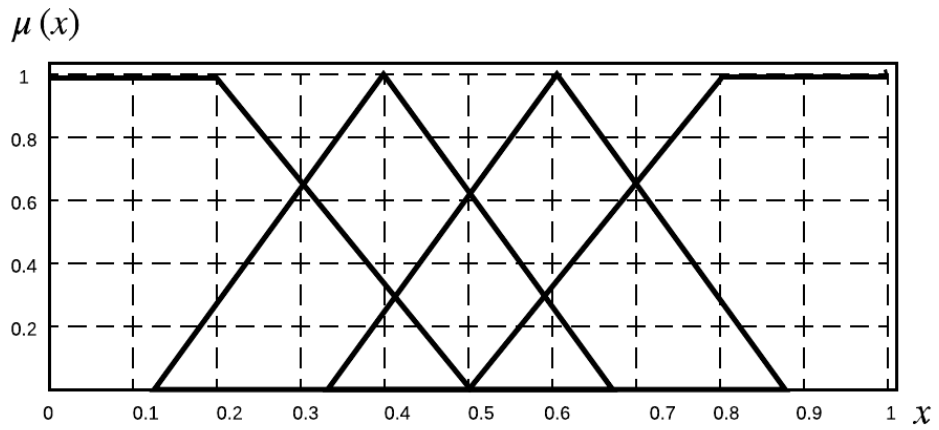
Функції належності вхідних змінних

Для підвищення ефективності роботи нечітких систем необхідно адекватно визначити функції належності для кожної лінгвістичної змінної. В даному контролері було означено чотири функції належності для кожної вхідної лінгвістичної змінної та п'ять – для вихідної.

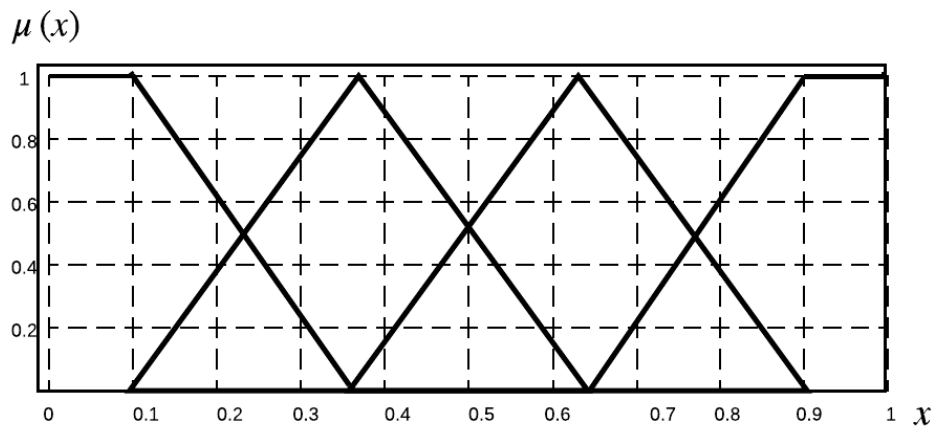
Наприклад, для означення змінної «Порушення–Режиму–Процедур» використано наступні терми подані в контексті кількості порушень: “Мало”, “Середнє”, “Багато” та “Дуже-Багато” (рис. 3а). Функції належності інших змінних відображені на рис.3б-3д.



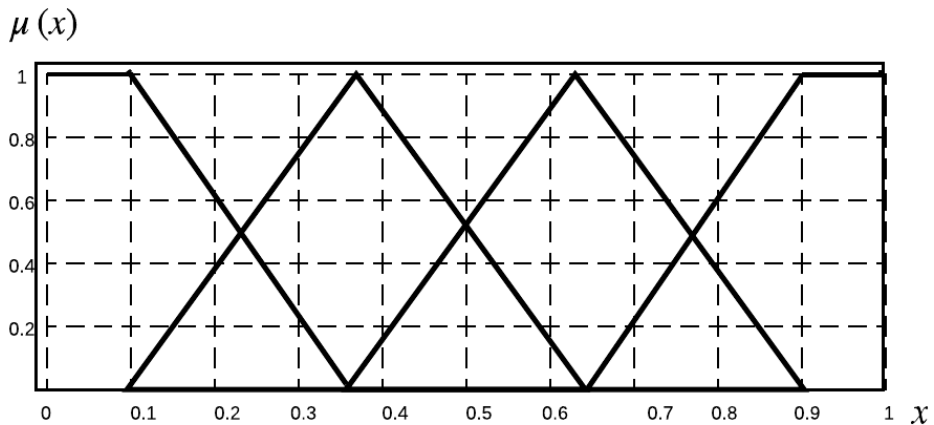
a.



б.

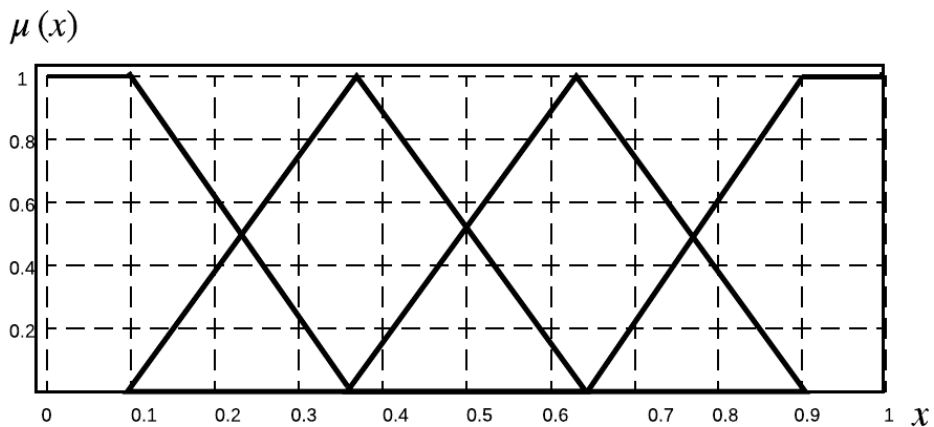


В.



Г.

Г.



Д.

Рис.3.3. Функції належності вхідних нечітких змінних *Порушення–Режим–Процедур* (а), *«Порушення–Перерв–Між–Процедурами»* (б), *«Порушення–Розміру–Груп»* (в), *Порушення–Розподілу–Навантаження–Персоналу»*(г), *«Порушення–Максимуму–Завантаження–Обладнання»* (д), контролера нечіткої логіки.

Для проведення інтегральної оцінки якості план-графіку одна вихідна змінна «Якість графіку», що використовується в контролері нечіткої логіки, має сім термів: *Незадовільний*, *Дуже-Поганий*, *Середньо*, *Вище-Норми*, *Задовільний*,

Якісний. Ці терми, які наведено на рис. 3.4, характеризують план-графік відповідно до результатів роботи системи нечіткого логічного висновку.

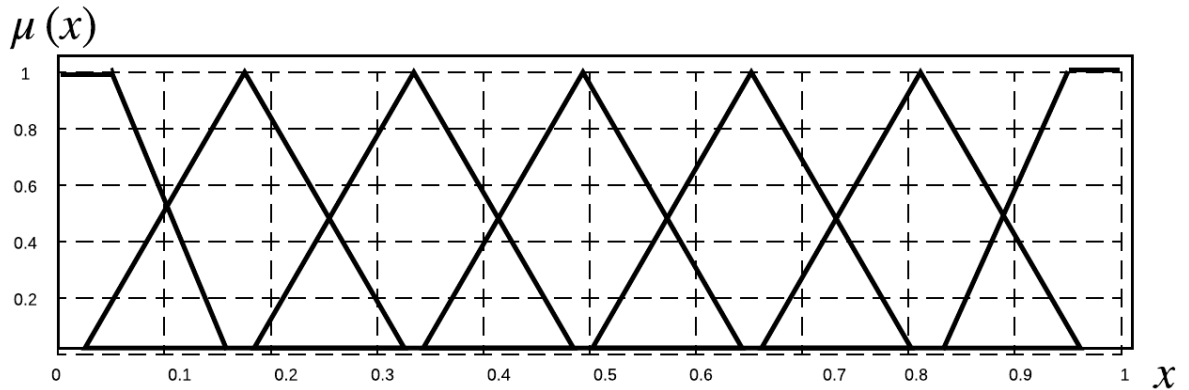


Рис 3.4. Функції належності вихідної нечітка змінної «Якість-Графіку»

Побудова продукційних правил нечіткого виведення

Для роботи системи логічного висновку сформовано набір продукційних правил, що формують базу знань нечіткого контролера. Для даної системи було розроблено базу, що містить 16 продукційних правил. Нижче наведено деякі з них:

1. **ЯКЩО** (Порушень-Режиму-Процедур Є Мало) **І** (Порушень-Перерв-Між-Процедурами Є Мало) **І** (Порушень-Розміру-Груп Є Мало) **І** (Порушень-Розподілу-Навантаження-Персоналу Є Мало) **І** (Порушень-Максимум-Завантаження-Обладнання Є Мало) **ТО** (Якість-Графіку Є Якісний)
2. **ЯКЩО** (Порушень-Режиму-Процедур Є Середнє) **І** (Порушень-Перерв-Між-Процедурами Є Середнє) **І** (Порушень-Розміру-Груп Є Середнє) **І** (Порушень-Розподілу-Навантаження-Персоналу Є Середнє) **І** (Порушень-Максимум-Завантаження-Обладнання Є Середнє) **ТО** (Якість-Графіку Є Середній)
3. **ЯКЩО** (Порушень-Режиму-Процедур Є Багато) **І** (Порушень-Перерв-Між-Процедурами Є Багато) **І** (Порушень-Розміру-Груп Є Багато) **І** (Порушень-Розподілу-Навантаження-Персоналу Є Багато) **І** (Порушень-Максимум-Завантаження-Обладнання Є Багато) **ТО** (Якість-Графіку Є Дуже-Поганий)

4. **ЯКЩО** (Порушень-Режиму-Процедур € Дуже-Багато) **І** (Порушень-Перерв-Між-Процедурами € Дуже-Багато) **І** (Порушень-Розміру-Груп € Дуже-Багато) **І** (Порушень-Розподілу-Навантаження-Персоналу € Середнє Дуже-Багато) **І** (Порушень-Максимум-Завантаження-Обладнання € Дуже-Багато) **ТО** (Якість-Графіку € Незадовільний)
5. **ЯКЩО** (Порушень-Режиму-Процедур € Мало) **АБО** (Порушень-Перерв-Між-Процедурами € Мало) **АБО** (Порушень-Розміру-Груп € Мало) **АБО** (Порушень-Розподілу-Навантаження-Персоналу € Мало) **АБО** (Порушень-Максимум-Завантаження-Обладнання € Мало) **ТО** (Якість-Графіку € Вище-Норми)

Зростання кількості порушень кожного критерія оптимальності графіку веде до погіршення результуючої його якості. Характер та інтенсивність впливу кожного критерію на інтегровану оцінку виражається за допомогою використання продукційних правил нечіткої логіки.

3.2 Метод Оптимізації План-графіків

3.2.1 Застосування генетичних алгоритмів для оптимізації план-графіків

В результаті проведеного аналізу було означено переваги генетичних алгоритмів в задачах оптимізації план-графіків, зокрема відновлювальної терапії [155]:

1. Генетичний алгоритм використовує кілька точок пошукового простору одночасно, а не здійснює переходи від точки до точки, як у традиційних методах. Це дозволяє подолати один з їхніх недоліків – небезпеку потрапляння в локальний екстремум цільової функції.

2. Генетичний алгоритм в процесі роботи не використовують ніякої додаткової інформації, що підвищує швидкість його роботи. Виключенням слугує множина допустимих значень параметрів і цільової функції в довільній точці.

3. Генетичний алгоритм використовує як ймовірнісні правила для побудови нових розв'язків, так і детерміновані правила для переходу від одних розв'язків до інших. Одночасне використання елементів випадковості та детермінованості дає значно більший ефект, ніж роздільне.

Зважаючи на вказані переваги для оптимізації план-графіку клініки розроблено метод на основі генетичних алгоритмів.

Принцип роботи генетичного алгоритму базується на моделюванні механізмів природної еволюції і популяційної генетики [157]. Графічно схему роботи алгоритму наведено на рис. 3. 5.

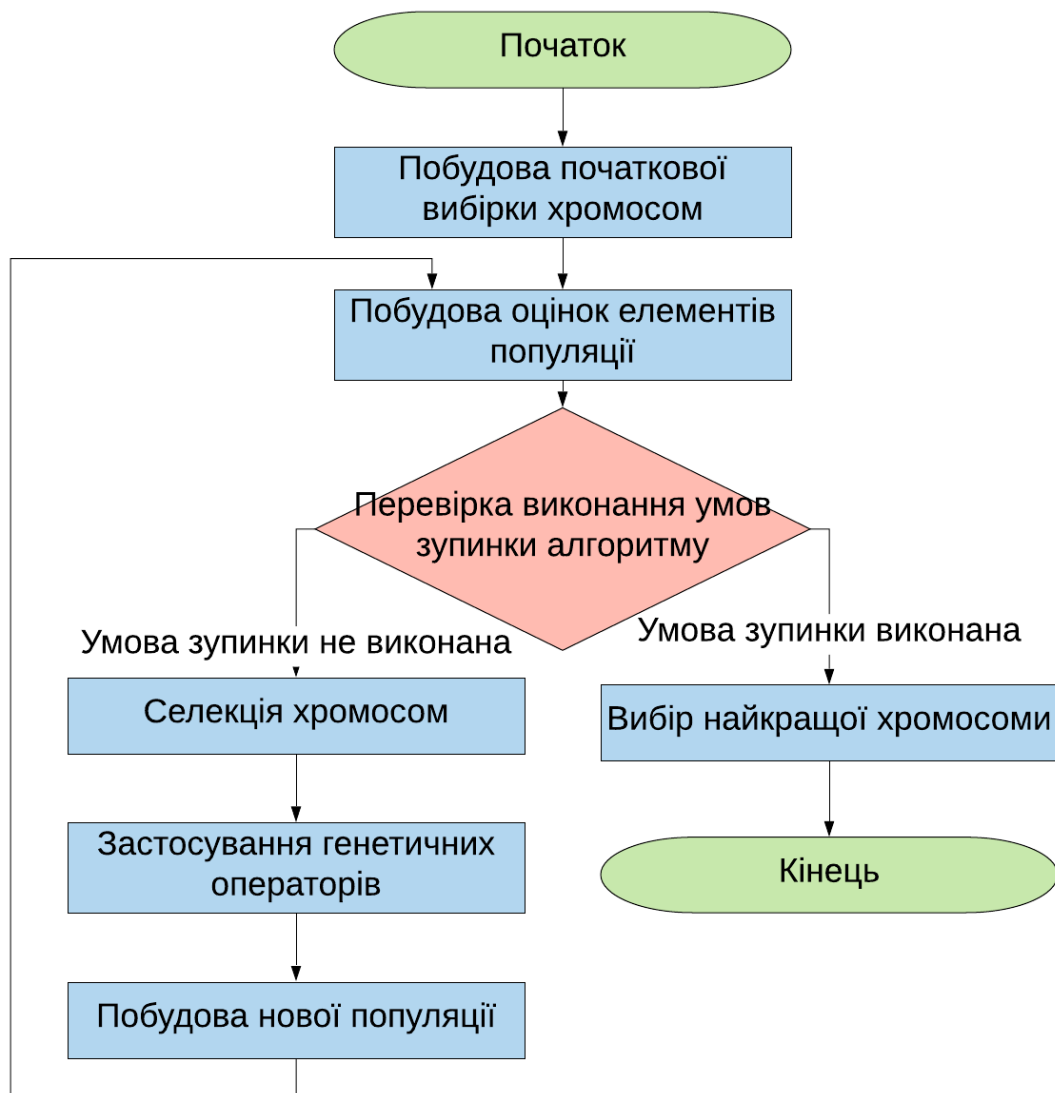


Рис 3.5. Загальна схема роботи генетичного алгоритму.

В контексті генетичних алгоритмів задача формалізується таким чином, щоб її рішення могло бути закодоване у вигляді хромосоми – набору аспектів план-графіку. Далі деяким способом створюється початкова множина хромосом – початкова популяція, з елементами якої за допомогою механізму оцінок асоціюються відповідні числові значення. З отриманої множини рішень обираються значеннями з найкращими результатами оцінки. До них застосовуються генетичні оператори – мутація та схрещування, що породжує нові рішення, які також піддаються оцінці та селекції. Процес триває ітеративно до моменту, поки не виконається деяка умова зупинки методу [158, 159, 160].

3.2.2 Розробка генетичного алгоритму оптимізації план-графіків

Представлення план-графіку функціонування клініки в вигляді хромосоми. До початку оптимізаційного процесу формується математичне представлення план-графіку функціонування клініки. Для цього було згруповано процедури за типами з урахуванням застосування необхідного обладнання, наявності медичного персоналу, вимог до періоду проведення процедур та можливості об'єднання пацієнтів у групи [161].

Для кожного типу згрупованих процедур період, охоплений план-графіком, було розбито на можливі часові інтервали їх проведення.

Математично множину часових проміжків проведення процедур представлено наступним чином:

$$S = \{s_i\},$$

$$\{s_i\} = \{s_i^p, s_i^d, s_i^g\}, \quad (3.9)$$

де s_i^p – дата проведення процедури;

s_i^d – час проведення процедури;

s_i^g – максимально допустима кількість паралельних процедур;

У результаті такого розбиття побудовано тривимірний масив, характеристики якого відповідають за розподіл процедур по днях місяця та

розподіл за часом впродовж дня; за урахуванням можливості проведення декількох або групових процедур в один і той самий часовий проміжок. Графічно даний масив зображено на рис.3.7.

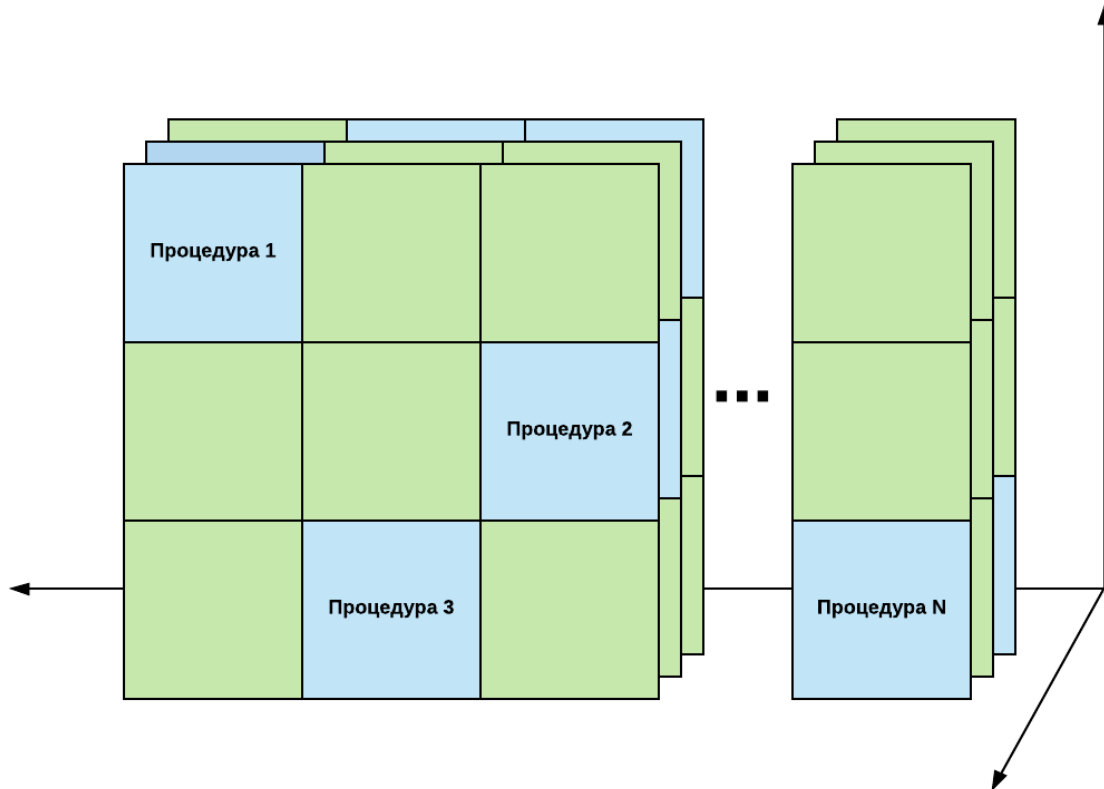


Рис 3.6. Представлення план-графіку функціонування клініки у вигляді тривимірного масиву

Побудова структури хромосоми. Після формування представлення план-графіку необхідно визначити структуру хромосоми – вектор значень, які оптимізуються [162]. У даному випадку кодування хромосоми у вигляді множини символічних чи бінарних даних є недоцільним через складність задачі та часові затрати на кодування та декодування проміжних результатів. Тому в даному випадку використано множину процедур, з яких складається план-графік.

Математично множину процедур план-графіку представлено наступним чином:

$$\begin{aligned}
 T &= \{t_i^w, t_i^d, t_i^p, t_i^e\}, \\
 t_i^d &= \{t_1^d \dots t_n^d\}, n = \overline{1..N_d}, \\
 t_i^p &= \{t_1^p \dots t_k^p\}, k = \overline{1..N_p}, \\
 t_i^e &= \{t_1^e \dots t_l^e\}, l = \overline{1..N_e},
 \end{aligned}
 \tag{3.10}$$

де t_i^w – дата та час проведення процедури;

t_i^d – медичний персонал, що проводить процедуру;

t_i^p – множина пацієнтів, для яких процедура проводиться;

t_i^e – множина обладнання, необхідного для проведення процедури;

N_d – загальна кількість медичного персоналу клініки;

N_p – загальна кількість пацієнтів клініки;

N_e – загальна кількість медичного обладнання.

Графічно структуру хромосоми можна подати в наступному вигляді (рис. 3.7).

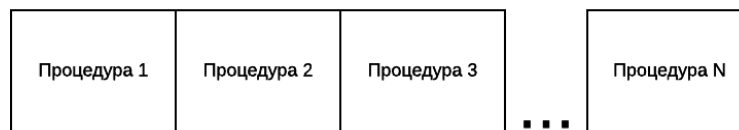


Рис. 3.7. Структура хромосоми задачі оптимізації план-графіку клініки

Побудова початкової вибірки план-графіків. Після формування структури хромосоми необхідно побудувати початкову популяцію план-графіків.

У даному випадку на етапі ініціалізації початкової популяції слід сформувати множину опорних план-графіків, що будуть відповідати жорстким обмеженням:

1) несуперечність план-графіку – медичний персонал та пацієнти не можуть знаходитися на двох процедурах одночасно;

2) послідовність процедур – їх необхідно проводити в порядку, визначеному індивідуальним планом лікування;

3) наявність обладнання та персоналу – не можна проводити процедури у випадку відсутності або часткового забезпечення устаткуванням чи лікарями.

Для побудови елементів популяції розроблено скінченний автомат. Його вхідним параметром є множина планів відновлювальної терапії з наборами процедур для включення в план-графік, а вихідним – сформований план-графік, що задовольняє зазначеним вище накладеним жорстким обмеженням (медичний персонал та пацієнти не можуть знаходитися на двох процедурах одночасно; процедури необхідно проводити в порядку, визначеному індивідуальним планом лікування; не можна проводити процедури у випадку відсутності або часткового забезпечення устаткуванням чи лікарями).

Докладний опис механізму роботи скінченного автомата, використаного в даній роботі наведено в розділі 2.

Побудова механізму селекції. Механізм селекції забезпечує зростання середнього значення функції придатності популяції за рахунок вибору особин з найвищими значеннями функції приналежності. Залежно від використовуваної схеми оптимізації вибираються різні механізми селекції. Найбільш широко застосовуються наступні схеми селекції [163]:

1. Прямо пропорційна селекція. Імовірність вибору особини t за використання даного методу прямо пропорційна значенню функції приналежності $f_i(t)$. Таким чином особинам з вищим значенням функції приналежності відповідає більш висока імовірність переходу до наступного покоління $p_i(t)$:

$$p_i(t) = f_i(t) / \sum_{i=1}^M f_i(t), \quad (3.11)$$

де $p_i(t)$ – імовірність переходу особини в наступне покоління;

$f_i(t)$ – функція приналежності особини;

M – розмір популяції.

2. Турнірна селекція. В основі селекції за допомогою турніру лежить наступна ідея: З популяції вибирається випадковим чином до k особин і найкраща з них копіюється в наступну популяцію. Процес повторюється N разів:

$$x_i(t + 1) = \{x_k(p_k), x_i(p_i) \mid \max(f_k(t); f_i(t))\}, \quad (3.12)$$

де $x_i(t + 1)$ – особина обрана для переходу в наступне покоління;
 $f_i(t), f_k(t)$ – функції приналежності особини що беруть участь в турнірній селекції;
 $x_k(p_k), x_i(p_i)$ – особини що беруть участь в турнірній селекції.

3. Елітарна селекція. З потомків отриманих з v в результаті застосування еволюційних операторів обираються особини з значенням функції приналежності вищим ніж хоча б в одній з батьківських особин:

$$x_i(t + 1) = \max(f_{pi}(t); \max(f_k(t); f_i(t))), \quad (3.13)$$

де $x_i(t + 1)$ – особина обрана для переходу в наступне покоління;
 $f_{pi}(t), f_i(t), f_k(t)$ – функції приналежності особини що беруть участь в елітарній селекції.

4. Правий погасаючий відбір. Проводиться сортування віх особин та обираються особини в яких значення функції приналежності вище середнього по популяції.

В даному випадку застосовується дворівнева селекція на основі правого погасаючого відбору та турнірної селекції. На першому кроці відкидаються особини з середніми значеннями функції приналежності. На другому – N разів повторюється процедура відбору і кращі особини потрапляють до нової популяції.

Побудова еволюційних операторів. Для організації оптимізаційного процесу необхідно створити механізми розвитку популяції. В більшості випадків використовують схрещування та мутацію.

Процедура схрещування в генетичному алгоритмі полягає в тому, що для двох обраних особин проводиться обмін частками генетичного коду між відповідними хромосомами. Для різних типів задач розроблено декілька варіантів операції кросоверу. Згідно з [163, 164] найбільша ефективність в задачах розподілу ресурсів та побудови план-графіків досягається за використання циклічного кросовера (Cycle Cross over), рівномірно-упорядкованого кросовера (Uniform Order-Based Crossover) або частково-узгодженого кросовера (Partially Matched Crossover).

Різні типи оператора кросовера володіють загальними позитивними властивостями: вони контролюють баланс між подальшим перевикористанням вже знайдених "хороших" областей простору пошуку та дослідженням нових. Це досягається за рахунок збереження основних блоків всередині батьківських хромосом і одночасного виділення нових областей в результаті обміну частинами хромосом.

Оператор мутації з деякою ймовірністю змінює значення декількох параметрів (генів) у хромосомі нової особини на інші, що входять у число припустимих значень даного гена. Мутації є випадковими, і не залежать від генетичного коду особини, ступеня її пристосованості, ні від того, чи приносять вони особині шкоду або користь.

Початкові та мутовані хромосоми можуть сильно відрізнятися одна від одної, що дозволяє алгоритму долати околиці локальних екстремумів.

Для даної задачі було розроблено низку операторів кросоверу та мутації. Перший оператор кросоверу реалізовано наступним чином: з популяції обирається дві особини, для яких визначаються точки здійснення схрещення. Далі випадковим чином визначаються часові інтервали, та формуються потомки. На останньому етапі роботи оператора проводиться перевірка сформованих особин на допустимість (виконання обмежень, накладених на план-графік) та можлива корекція.

Другий оператор кросовера полягає в здійсненні багаторазового схрещування часових інтервалів всередині самої хромосоми. Першим кроком здійснюється

пошук двох часових інтервалів, та їх обмін. Далі проводиться перевірка результуючої особини на відповідність жорстким обмеженням та її коригування.

Розроблені оператори мутації працюють наступним чином:

1) зміна часу проведення процедури, що полягає в перенесенні певної процедури з одного часового проміжку на довільний інший доступний часовий проміжок впродовж того самого дня;

2) зміна дати проведення процедури полягає в перенесенні певної процедури з одного часового проміжку на довільний інший доступний часовий проміжок іншого дня;

3) обмін пацієнтів процедур полягає у взаємозаміні пацієнтів двох однакових процедур;

4) обмін медичного персоналу полягає у частковій взаємозаміні кваліфікованого персоналу, що проводить дві процедури одного типу.

Кожен з вище наведених операторів з певною імовірністю застосовуються на кожній ітерації оптимізаційного процесу для особин отриманих в результаті роботи механізму селекції.

Нейро-нечітка багатокритеріальна оцінка прогресу оптимізації. Для оцінки прогресу оптимізації необхідно сформувавши механізм оцінки альтернатив, сформованих процесі роботи еволюційних операторів – функцію приналежності. Оцінка план-графіку (ступеня пристосованості особини) проводиться за вектором значень визначених критеріїв оптимальності. Для цього було введено механізм агрегування та нормалізації заснований на штрафах за порушення критеріїв. Далі здійснюється нечітке логічне виведення на основі отриманих результатів, фактично процес згортання векторного критерію. Використаний в даному випадку підхід докладно описаний в розділі 3.1.

Визначення критеріїв зупинки роботи методу оптимізації. За умови досягнення прийнятних результатів оптимізації, необхідно здійснити зупинку процесу оптимізації.

Генетичний алгоритм зупиняється, коли коефіцієнт втрати різноманіття у популяції перевищить встановлену межу. Це буде означати, що популяція

зійшлася до деякого значення. Можна також використовувати номер попереднього покоління або число попередніх поколінь, з якими буде відбуватися порівняння.

Найпоширенішими методами означення критеріїв зупинки є [165]:

1. Сходження значень функції приналежності

Функція приналежності порівнює середнє значення цільової функції хромосом популяції з найкращим значенням. Якщо середнє значення цільової функції відрізняється менш, ніж на задане число відсотків від значення цільової функції найкращої хромосоми, то генетичний алгоритм зупиняється. В основному це означає, що популяція вже зійшлася до якого-небудь значенням.

2. Задане цільове значення функції приналежності

ГА зупиняється, якщо поточне значення у кращого індивіда дорівнює або краще, ніж задане значення функції придатності. Якщо число задоволених вимог, для будь-якої особини, за кожним критерієм стає більшим від заданої для нього величини, то генетичний алгоритм зупиняється.

Для даної задачі, зупинка роботи методу відбувається, якщо впродовж визначеної кількості поколінь оптимізації не відбувається покращення хромосоми з найвищим рівнем оцінки. Це означає, що застосування вище наведених еволюційних операторів не спричинює позитивного впливу на якість результуючого план-графіку (чи взагалі спричиняє його погіршення). Іншими словами, зупинка алгоритму відбувається в момент досягнення екстремуму та в час збіжності значень функції приналежності.

Структура генетичного алгоритму оптимізації план-графіків відновлювальної терапії

В процесі роботи алгоритму виконуються наступні дії:

1. Проводиться ініціалізація початкової вибірки – побудова множини опорних план-графіків.

2. Кожна особина поточної популяції оцінюється за допомогою алгоритму нечіткої логіки на предмет ступеня задоволення поставлених обмежень.

3. Кращі рішення копіюються в нову популяцію на основі застосування операторів селекції.

4. На основі пропорційного відбору з поточної популяції вибираються хромосоми, які піддаються рекомбінації. До них застосовуються оператори кросоверу та мутації.

5. В випадку якщо нова популяція сформована, стара видаляється. Далі здійснюється перехід до п. 5, а у протилежному випадку – до п. 3.

6. Якщо не виконується жоден критерій зупинки оптимізаційного процесу, виконується перехід на п.1.

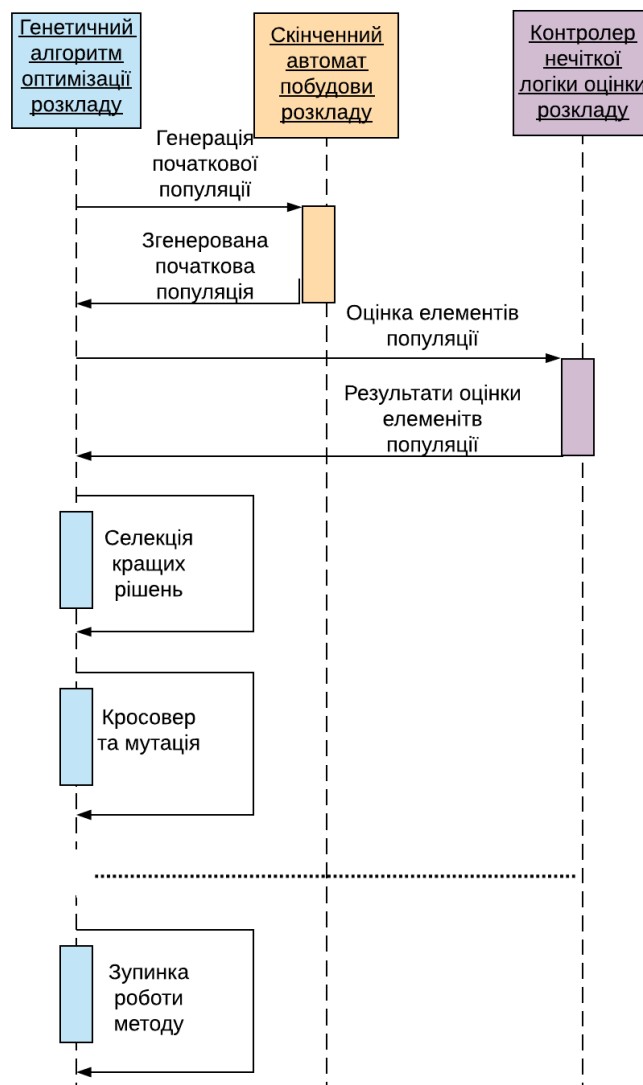


Рис 3.8. Структурна схема роботи генетичного алгоритму оптимізації план-графіків клініки

ВИСНОВКИ ДО РОЗДІЛУ 3

1. Використання систем нечіткого логічного висновку є ефективною альтернативою класичним методам згортання векторного критерію в задачах багатокритеріальної оцінки та оптимізації, зокрема в задачах побудови план-графіків. Класичні підходи до реалізації таких систем – Mamdani та Takagi-Sugeno мають низку негативних факторів, що обмежує можливості їх використання, тому доцільним є застосування іншої системи, що компенсує їх недоліки.
2. Однією з нещодавно представлених систем, що компенсують більшість недоліків традиційних систем нечіткого логічного висновку є T-Controller, що базується на концепції моделювання геометричних даних.
3. Зважаючи на переваги T-Controller перед класичними системами нечіткого висновку, розроблено контролер нечіткої логіки на основі системи T-Controller, що дозволяє провести кількісну оцінку оптимальності план-графіків відновлювальної терапії.
4. Для проведення багатокритеріальної оптимізації план-графіків відновлювальної терапії, найбільш перспективним є використання генетичних алгоритмів з огляду стійкості до потрапляння в локальні оптимуми, паралельне використання ними багатьох точок пошукового простору, комбінування імовірнісних та детермінованих правил для переходу від одних розв'язків до інших.
5. Запропоновано спосіб кодування план-графіку функціонування клініки у вигляді хромосоми, метод генерації початкової популяції план-графіків на основі дослідження проведеного в розділі 2, а також механізм селекції хромосом для їх подальшої оптимізації, що забезпечує підвищену швидкодію алгоритму за рахунок необхідності кодування та декодування кожної особини популяції.

6. З урахуванням специфіки задач синтезу план-графіків та розподілу ресурсів обґрунтовані та розроблені способи рекомбінації генів у хромосомі, що інтенсифікують процес пошуку оптимальної комбінації.
7. В якості механізму оцінки еволюційного процесу використано контролер нечіткої логіки, розроблений на основі системи T-Controller, що дозволяє провести
8. Використання контролера нечіткої логіки на основі T-Controller дало можливість провести кількісну оцінку та порівняння альтернатив побудованих в ході еволюційного процесу.

РОЗДІЛ 4. РОЗРОБКА І АПРОБАЦІЯ ПРОГРАМНОГО КОМПЛЕКСУ ОПТИМІЗАЦІЇ ПЛАНІВ ВІДНОВЛЮВАЛЬНОЇ ТЕРАПІЇ

4.1 Загальна архітектура та принципи функціонування програмного комплексу

Для вирішення задачі побудови та багатокритеріальної оптимізації план-графіків відновлювальної терапії розроблено спеціалізований програмний комплекс. Беручи до уваги необхідність використання системи великою кількістю користувачів застосування розподіленої архітектури програмного комплексу є очевидною потребою. Для реалізації таких систем існують два розповсюджені способи: використання “товстого” та “тонкого” клієнту [166, 167].

“Товстий” клієнт представляє собою архітектурне рішення, за використання якого логіка роботи застосунку переноситься на сторону клієнта (користувача), а сервер виступає лише як сховище даних. Такого роду клієнтом може виступати десктопний ужиток, веб-сайт або мобільний додаток.

Його використання має наступні переваги [169]:

- доступ до апаратних засобів клієнта (користувача), та, як наслідок – відсутність необхідності забезпечення великих власних серверних потужностей;
- можливість роботи без прямого або постійного доступу до мережі Інтернет (із обмеженою функціональністю);
- збільшення кількості клієнтів, які можуть обслуговуватись програмним комплексом паралельно;
- можливість використання технологій, що для операційних систем та платформ, що забезпечує – високу швидкодію.

Проте використання зазначеного підходу має і певні недоліки:

- пониження рівня безпеки та цілісності системи – частина даних зберігається на стороні клієнта на постійній основі;

- висока залежність від апаратної архітектури кожного клієнта – залежність від типу та версії операційної системи, встановлених сторонніх програмних продуктів тощо;

- необхідність прямого встановлення копії застосування на кожному клієнті;
- ускладнений процес переходу клієнтів на нові версії програмного продукту;

У випадку застосування архітектури “тонкого” клієнта вся інформація та логіка роботи ужитку зберігається на сервері. Це дає можливість усунути низку недоліків. Використання архітектури “тонкого” надає наступні переваги [170]:

- можливість легкої інтеграції зі сторонніми додатками;
- можливість масштабування за вимогою (при використанні хмарних ресурсів);

- кращий рівень безпеки та вища відмовостійкість;
- простіша підтримка клієнтів;
- незалежність від апаратної архітектури та операційної системи клієнтів;
- необхідність підтримки лише однієї операційної системи та набору сторонніх допоміжних ресурсів та додатків (що можуть не працювати при інших наборах параметрів);

- простота додавання нового функціоналу та розмежування доступів клієнтів до різних частин ужитку;

- вища швидкість розробки.

Як правило, реалізаціями “тонкого” клієнта є веб-орієнтовані ужитки які набрали популярності разом з поширенням мережі інтернет. Недоліками підходу “тонкого” клієнта є необхідність підтримки різних версій браузерів, розробка системи масштабування, необхідність побудови системи автоматизованого розгортання нових версій.

З огляду на наведений вище перелік переваг та особливості даного програмного продукту, було зроблено вибір в користь використання саме “тонкого” клієнта.

Програмний комплекс системи оптимізації планів відновлювальної терапії представляє собою трирівневу архітектуру, що складається з клієнтського середовища, сервера застосувань, а також сервера бази даних.

Використання такої структури надає низку переваг:

- ізоляцію сервера бази даних від клієнта шляхом розробки системи авторизації та автентифікації користувачів до сервера;

- немає необхідності у встановленні програмного забезпечення на робочих станціях користувачів;

- автоматичне централізоване оновлення програмного комплексу для всіх клієнтів за рахунок заміни модулів на сервері застосувань;

- централізований контроль за рахунок використання авторизації та автентифікації користувачів до серверу застосувань;

- можливість автоматичного горизонтального масштабування системи для збільшення продуктивності;

- незалежна конфігурація на кожному з рівнів структури;

До програмного комплексу системи оптимізації планів відновлювальної терапії входять наступні компоненти [171]:

- *Web-сервер*. В даній роботі використано хмарний сервіс OpenShift [172], що надається компанією RedHat. Він дозволяє швидко налаштовувати середовище для розгортання програмних продуктів шляхом вибору картриджів (віртуальних робочих станцій) для мови програмування та пов'язаних ужитків. Картриджі будуються на основі готових шаблонів, що забезпечує швидке розгортання програмних комплексів. В залежності від апаратних та програмних вимог картриджі можна конфігурувати, чи компоувати нові картриджі з вже існуючими.

- *Сервер бази даних*. Для роботи з даними обрано сервер MongoDB [173], що дозволяє швидко та зручно маніпулювати з даними великого об'єму. Перевага використання даної бази в тому, що вона є компактною і надає можливість дублювання записів, внаслідок чого зменшується ризик втрати даних, оскільки їх структура є розподіленою та зберігається в декількох місцях. Також MongoDB

забезпечує можливість зміни структури даних в процесі розгортання нових версій програмних комплексів за рахунок гнучкої схеми записів.

- *Сервер безперервної інтеграції.* Для безперервної інтеграції обрано сервер – Jenkins [174], що забезпечує легкість здійснення розгортання нових версій програмного комплексу. Перевагою використання саме цього продукту є підтримка його від спільноти, що забезпечує стабільне оновлення версій. Також Jenkins забезпечує гнучку конфігурацію та розширення власного функціоналу за рахунок використання плагінів.

- *Система контролю версій.* В даній роботі використано систему контролю версій GIT [175] та сервер контролю версій Bitbucket для зберігання вихідного програмного коду. GIT займає лідируючі позиції та зарекомендував себе в як надійну систему контролю версійності коду. Bitbucket є одним із найбільших репозиторіїв для спільної розробки програмного забезпечення. Він дозволяє переглядати збережений код, має функцію підсвічування синтаксису для різних мов програмування.

- *Сервер застосувань.* В його основу покладені методи, розроблені в розділах 2 та 3 даної роботи.

- *Сервер представлення клієнтської інформації.* В даній роботі використано сервер NodeJS [176], що дозволяє швидко побудову функціональних клієнтських представлень реального часу.

Принцип роботи архітектури програмного комплексу на рівні адміністратора представлено на рис 4.1

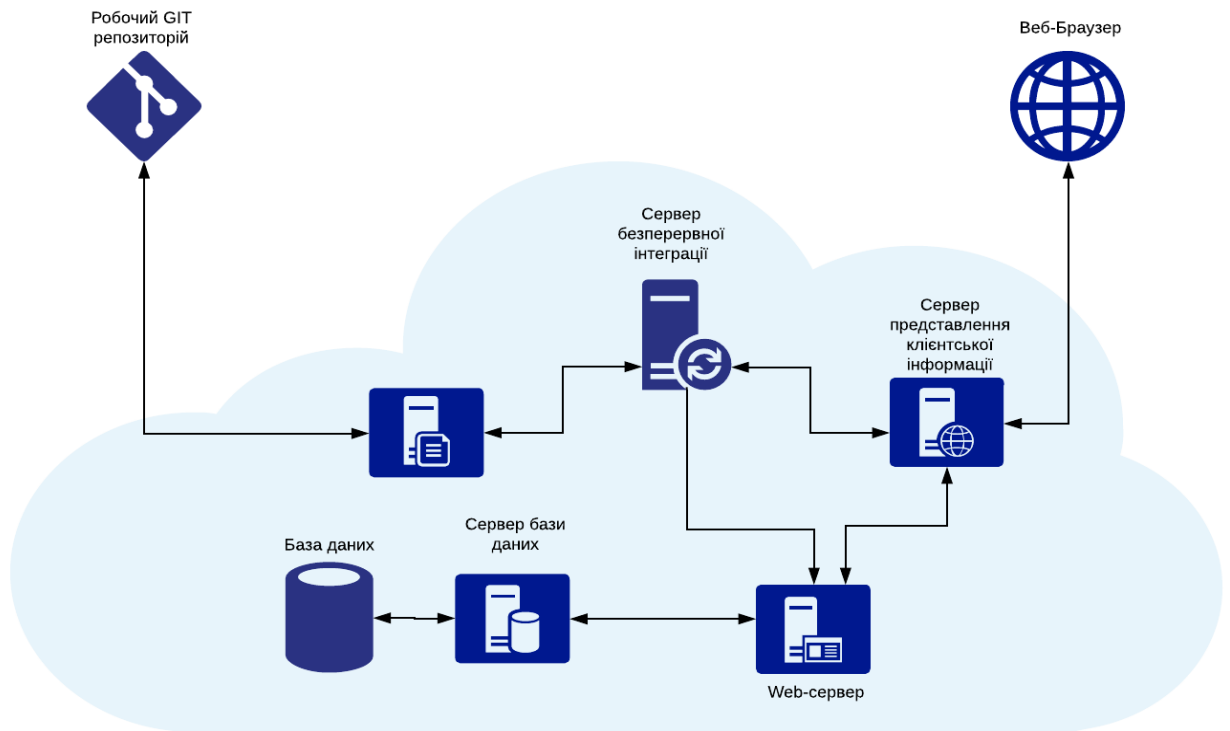


Рис. 4.1. Архітектура програмного комплексу оптимізації планів відновлювальної терапії

Доступ до вихідного коду сервера застосувань має адміністратор. У нього є можливість редагування поточного та внесення нового функціоналу. Для цього адміністратор робить зміни в локальному коді та зберігає їх локально на робочій старції. Після проведення циклів тестування адміністратор завантажує зміни на сервер контролю версій Bitbucket.

Оновлення програмного комплексу відбуваються за рахунок використання системи безперервної інтеграції за допомогою реалізованих підпрограм, а саме:

- завантаження останніх змін з репозиторія контролю версій;
- компіляція та перевірка вихідного коду на предмет помилок;
- побудова пакетів для розгортання на web-сервері;
- запуск автоматизованих тестів;
- зупинка сервера застосувань;
- заміна та розгортання пакетів на web-сервері;

- запуск web-сервера;

Адміністратор запускає набір задач для розгортання змін, що були внесені в сервер контролю версій. Після успішного виконання нова версія ужитку готова до роботи.

Робота клієнтів з програмним комплексом починається із запитів до сервера побудови представлення, що будує графічний інтерфейс для клієнтів. Запити, зроблені через графічний інтерфейс, надходять до сервера застосувань через web-сервер. Дані, які оброблюються сервером застосувань, передаються до сервера баз даних для зберігання.

За умови виникнення непередбачуваних ситуацій адміністратор має доступ до сервера баз даних для відновлення інформації або усунення помилок.

Після розгортання нової версії програмного комплексу на сервері застосувань він починає приймати запити, що надходять з графічного інтерфейсу. Отримані запити обробляються основним сервлетом, що делегує запити на подальшу обробку відповідним контролерам. Контролер відповідає за формування відповіді клієнту та використовує різноманітні сервісні компоненти, які в свою чергу звертаються до бази даних. Після успішного виконання контролер повертає результат своєї роботи назад до у відповідному форматі.

Сервер застосувань включає в себе наступні структурні рівні: представлення даних, бізнес логіки, доступу до даних.

Рівень представлення даних відповідає за конвертування кінцевої інформації для розміщення її на стороні клієнта. Цей рівень реалізований за допомогою REST-контролерів, що є частиною програмної екосистеми Spring Framework.

Рівень бізнес логіки реалізує основну логіку роботи програмного комплексу. Він реалізований за допомогою сервісів Spring Framework [177].

Рівень доступу до даних містить в собі реалізацію методів доступу до колекцій бази даних та класи безпосередньої роботи з ними.

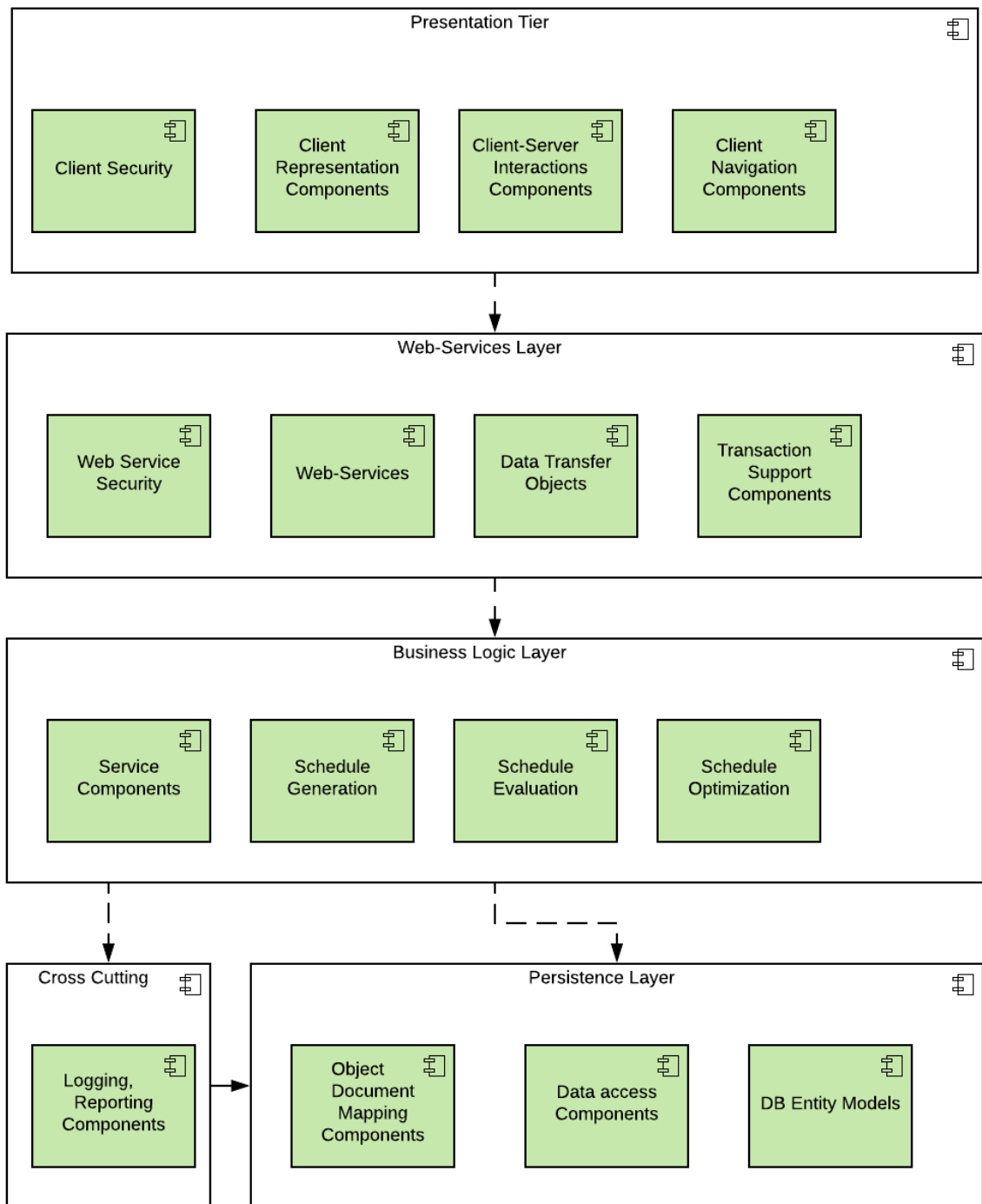


Рис. 4.2. Архітектура сервера застосувань програмного комплексу відновлювальної терапії

Рівень доступу до даних та бізнес рівень використовують наскрізну функціональність, яка містить в собі основні рішення, що впливають на роботу декількох рівнів (Cross-cutting).

4.2 Реалізація програмного комплексу

4.2.1 Опис основних модулів програмного комплексу

Класи програмного продукту було розділено на шість модулів (рис. 4.3) в залежності від функцій, які вони виконують.

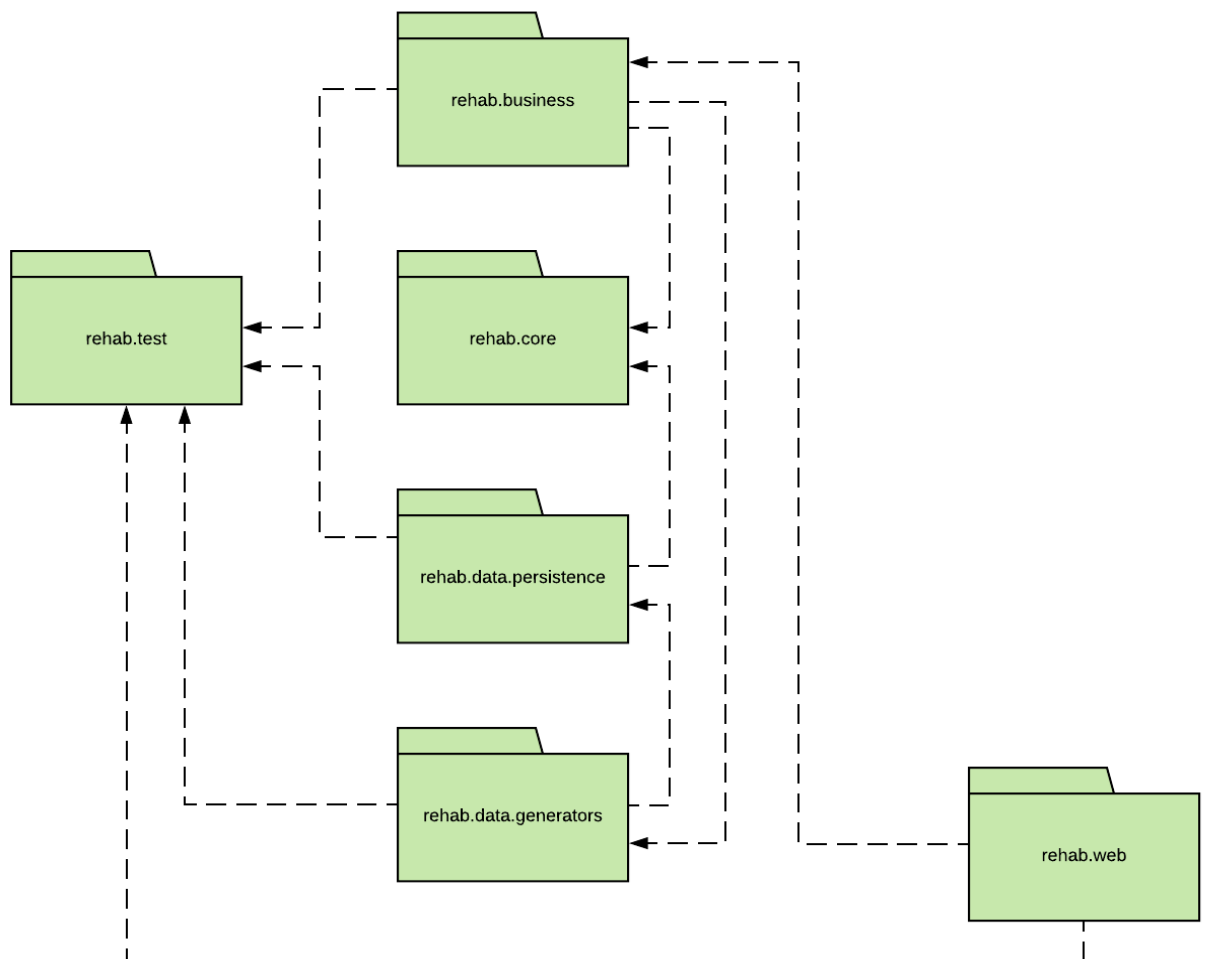


Рис. 4.3. Діаграма модулів програмного комплексу оптимізації планів відновлювальної терапії

Цими модулями є:

- rehab.core – містить в собі основні сутності предметної області: пацієнтів, плани реабілітації, медичне обладнання тощо;

- rehab.data.generators – містить в собі реалізацію методу побудови план-графіку та супутню логіку;

- `rehab.data.persistence` – містить описи класів для роботи з базою даних, а також сутності, що використовуються для збереження в базі;

- `rehab.business` – містить в собі реалізації методів оцінки та оптимізації план-графіку, а також використовує класи описані в модулі `rehab.core` та `rehab.data.generators`, `rehab.data.persistence` в процесі роботи;

- `rehab.web` – містить компоненти сервісів;

- `rehab.test` – містить юніт, а також інтеграційні тести всіх модулів, що входять до складу програмного комплексу.

Розглянемо детальніше кожен із модулів.

Діаграма класів модулю `rehab.core` наведена в додатку 2 на рис.1. Даний модуль містить наступні основні класи: `Clinic`, `RehabFacility`, `FacilitySchedule`, `DailyFacilitySchedule`, `Patient`, `Procedure`, `ProcedureType`.

Clinic представляє собою програмне представлення клініки. Включає в себе список пацієнтів, що проходять відновлювальну терапію, списки наявних лікарів, обладнання, кабінетів лікування та іншу необхідну інформацію.

RehabFacility це загальний клас, що представляє деякий ресурс, необхідний для проведення відновлювальної терапії – кабінет чи обладнання.

FacilitySchedule представляє план-графік використання / доступності певного ресурсу для лікування.

DailyFacilitySchedule являє собою план-графік використання / доступності певного ресурсу для лікування протягом дня.

Patient це програмне відображення пацієнта, яке включає в себе дані про пацієнта а також його план відновлювальної терапії.

Procedure це програмне відображення процедури, яке включає в себе необхідні ресурси, необхідний персонал, дату проведення, а також тип процедури.

ProcedureType відображає тип кожної конкретної процедури.

Діаграма класів модулю `rehab.data.generators` наведена в додатку 2 на рис. 2. Даний модуль містить наступні основні класи: *RehabPlanGenerator*, *RehabPlanBuilder*, *ProceduresBuilder*.

RehabPlanGenerator – програмна реалізація методу побудови планів відновлювальної терапії в контексті побудови план-графіків. В ході роботи даний клас використовує логіку класів *RehabPlanBuilder* та *ProceduresBuilder*.

RehabPlanBuilder – утилітарний клас для програмної ініціалізації планів реабілітації.

ProceduresBuilder – утилітарний клас для програмної ініціалізації процедур реабілітації.

Діаграма класів модулю `rehab.data.persistence` наведена в додатку 2 на рис. 3. Даний модуль містить наступні основні класи: *DurationToStringConverter*, *PeriodToStringConverter*, *StringToDurationConverter*, *StringToLocalDateConverter*, *StringToLocalTimeConverter*, *StringToPeriodConverter*, *ClinicRepository*, *FacilitiesRepository*, *FacilityScheduleRepository*, *PatientRepository*, *ProcedureTypeRepository*, *RehabPlanRepository*.

DurationToStringConverter – утилітарний клас для конвертації тривалостей в символічне представлення для наступного збереження в базі даних.

PeriodToStringConverter – утилітарний клас для конвертації тривалостей в символічне представлення для наступного збереження в базі даних.

StringToDurationConverter – утилітарний клас для конвертації символічного представлення тривалостей у вигляд об'єктів для подальшого документо-об'єктного відображення при роботі з базою даних.

StringToLocalDateConverter – утилітарний клас для конвертації символічного представлення тривалостей у вигляд об'єктів для подальшого документо-об'єктного відображення при роботі з базою даних.

StringToLocalTimeConverter – утилітарний клас для конвертації символічного представлення тривалостей у вигляд об'єктів для подальшого документо-об'єктного відображення при роботі з базою даних.

StringToPeriodConverter – утилітарний клас для конвертації символічного представлення тривалостей у вигляд об'єктів для подальшого документо-об'єктного відображення при роботі з базою даних.

ClinicRepository – клас для роботи з об'єктами клініки на рівні бази даних.

FacilitiesRepository – клас для роботи з об'єктами ресурсів клініки на рівні бази даних.

FacilityScheduleRepository – клас для роботи з об'єктами план-графіків використання ресурсів клініки на рівні бази даних.

PatientRepository – клас для роботи з об'єктами пацієнтів клініки на рівні бази даних.

ProcedureTypeRepository – клас для роботи з типами процедур на рівні бази даних.

RehabPlanRepository – клас для роботи з об'єктами планів відновлювальної терапії на рівні бази даних.

Діаграма класів модуля `rehab.data.business` наведена в додатку 2 на рис. 4. Даний модуль містить наступні основні класи: *ClinicScheduleGenerator*, *ScheduleEvaluator*, *ScheduleOptimizer*, *GenethicScheduleOptimizer*, *GeneticTerminationConditions*, *MutateProcedureOperators*, *OptimisationContext*, *ProcedureCandidateFactory*, *FuzzyFitnessEvaluator*, *ScheduleManager*.

ClinicScheduleGenerator – програмна реалізація методу побудови опорного план-графіку клініки.

ScheduleEvaluator – програмна реалізація контролера нечіткої логіки оцінки план-графіку клініки.

GenethicScheduleOptimizer – програмна реалізація методу оптимізації план-графіку клініки. Клас використовує функціональність *ClinicScheduleGenerator* та *ScheduleEvaluator*.

GeneticTerminationConditions – клас, що інкапсулює умови завершення роботи генетичного алгоритму.

MutateProcedureOperators – клас, що інкапсулює оператори схрещення та мутації, які використовуються в ході роботи генетичного алгоритму.

OptimisationContext – клас інкапсулює дані, що використовуються генетичним алгоритмом в ході роботи: хромосоми клінік, набори план-графіків пацієнтів тощо.

ProcedureCandidateFactory – клас, що використовується генетичним алгоритмом для виконання оператора селекції та побудови початкової популяції.

ScheduleManager – утилітарний клас, який дозволяє здійснювати різноманітні перетворення план-графіку.

Діаграма класів модулю *rehab.data.web* наведена в додатку 2 на рис. 5. Даний модуль містить наступні основні класи: *SpringBootApplication*, *BootMainContextConfig*, *MongoConfig*, *ClinicScheduleController*, *PatientsController*, *RehabFacilitiesController*, *MongoClinicScheduleService*, *MongoPatientService*, *MongoFacilityService*.

SpringBootApplication – основний клас програмного комплексу, що здійснює завантаження сервера ужитків.

BootMainContextConfig – конфігураційний клас для завантаження сервера ужитків.

MongoConfig – конфігураційний клас для роботи бази даних.

ClinicScheduleController – містить методи обробки запитів від клієнта на створення план-графіку та виведення його у відповідному форматі.

PatientsController – містить методи обробки запитів від клієнта на маніпуляції з пацієнтами клініки та виведення результатів у відповідному форматі.

RehabFacilitiesController – містить методи обробки запитів від клієнта на маніпуляції з ресурсами клініки та виведення результатів у відповідному форматі.

MongoClinicScheduleService – містить логіку предметної області для маніпуляції з план-графіком клініки виведення результатів у відповідному форматі.

MongoPatientService – містить логіку предметної області для маніпуляції з пацієнтами клініки виведення результатів у відповідному форматі.

MongoFacilityService – містить логіку предметної області для маніпуляції з ресурсами клініки виведення результатів у відповідному форматі.

4.2.2 Опис структури бази даних програмного комплексу

Як базу даних для програмного продукту використано нереляційну документо-орієнтовану базу даних MongoDB. В якості API для доступу до бази даних використано модуль Spring Framework - Data. Вибір нереляційної бази даних пов'язаний з тим, що вона дозволяє зручно і легко впорядковувати інформацію та зв'язки між пов'язаними даними. На рис. 4.4 представлена схема бази даних.

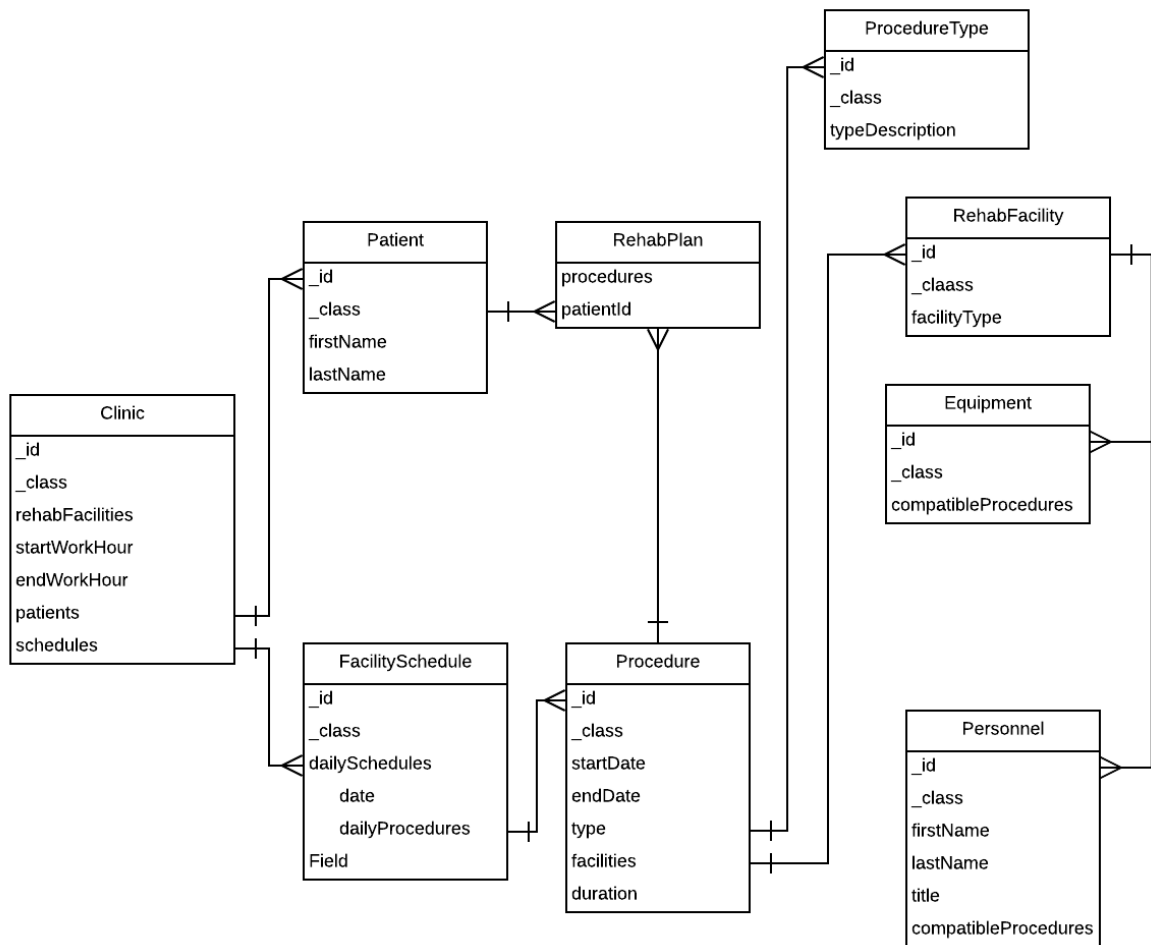


Рис. 4.4 Схема бази даних програмного комплексу оптимізації планів відновлювальної терапії

Основною таблицею системи є Clinic. Вона містить загальну інформацію про наявних пацієнтів, план-графіки, а також ресурси, наявні в клініки. За допомогою посилань таблиці Clinic до інших таблиць, дані зв'язуються з їх реальним представленням в таблицях Patient – що містить деталі про пацієнтів які зараз

перебувають на лікуванні, а також план-графіками використання обладнання та інших ресурсів в таблиці FacilitySchedule.

Таблиця 4.1. Опис таблиці FacilitySchedule

Ім'я поля	Тип поля	Призначення
_id	ObjectId	Унікальний ідентифікатор об'єктів в таблиці Clinic
_class	String	Назва класу для здійснення документо-об'єктного відображення
rehabFacilities	DBRef	Список лікувальних ресурсів, наявних в клініки
startWorkHour	Date	Початковий час призначення процедур впродовж дня
endWorkHour	Date	Кінцевий час призначення процедур впродовж дня
patients	DBRef	Список пацієнтів, що проходять відновлювальну терапію
schedules	DBRef	Список план-графіків використання обладнання клініки

В таблиці Patient зберігаються дані про пацієнтів та плани відновлювальної терапії, що вони проходять.

Таблиця 4.2. Опис таблиці Patient

Ім'я поля	Тип поля	Призначення
_id	ObjectId	Унікальний ідентифікатор об'єктів в таблиці Clinic
_class	String	Назва класу для здійснення документо-об'єктного відображення
firstName	String	Ім'я пацієнта
lastName	String	Прізвище пацієнта
rehabPlan	DBRef	План відновлювальної терапії пацієнта

Плани відновлювальної терапії пацієнтів зберігаються в таблиці RehabPlan.

Таблиця 4.3 Опис таблиці RehabPlan

Ім'я поля	Тип поля	Призначення
_id	ObjectId	Унікальний ідентифікатор об'єктів в таблиці RehabPlan
_class	String	Назва класу для здійснення документо-об'єктного відображення
procedures	Array	Список процедур, що входять до лікувального плану пацієнта
patientId	String	Ідентифікатор пацієнта, якому належить план лікування

Конкретні процедури, що входять до планів зберігаються в таблиці Procedure. За допомогою посилань вони також зв'язані з таблицею використання обладнання.

Таблиця 4.4 Опис таблиці Procedure

Ім'я поля	Тип поля	Призначення
_id	ObjectId	Унікальний ідентифікатор об'єктів в таблиці RehabPlan
_class	String	Назва класу для здійснення документо-об'єктного відображення
startDate	Date	Дата початку процедури у випадку, якщо вона уже включена в план-графік
endDate	Date	Дата закінчення процедури у випадку, якщо вона уже включена в план-графік
type	DBRef	Характеризує тип процедури
facilities	Array<DBRef>	Ресурси необхідні для проведення процедури
duration	Integer	Тривалість процедури

План-графіки використання медичного обладнання зберігаються в таблиці FacilitySchedule.

Таблиця 4.5 Опис таблиці FacilitySchedule

Ім'я поля	Тип поля	Призначення
_id	ObjectId	Унікальний ідентифікатор об'єктів в таблиці RehabPlan
_class	String	Назва класу для здійснення документо-об'єктного відображення
dailySchedules	Array<Object>	План-графік використання ресурсів впродовж дня
date	Date	Дата для якої складено план-графік
dailyProcedures	Array<Procedure>	Процедури що проводяться протягом дня

Загальний список ресурсів наявних в клініці агрегується в таблиці RehabFacility.

Таблиця 4.6 Опис таблиці RehabFacility

Ім'я поля	Тип поля	Призначення
_id	ObjectId	Унікальний ідентифікатор об'єктів в таблиці RehabPlan
_class	String	Назва класу для здійснення документо-об'єктного відображення
typeDescription	String	Опис типу ресурсу

Типи процедур відновлювальної терапії зберігаються в таблиці ProcedureType.

Таблиця 4.7 Опис таблиці ProcedureType

Ім'я поля	Тип поля	Призначення
_id	ObjectId	Унікальний ідентифікатор об'єктів в таблиці RehabPlan
_class	String	Назва класу для здійснення документо-об'єктного відображення
typeDescription	String	Опис типу процедури

Опис персоналу клініки зберігається в таблиці Personnel.

Таблиця 4.8 Опис таблиці Personnel

Ім'я поля	Тип поля	Призначення
_id	ObjectId	Унікальний ідентифікатор об'єктів в таблиці RehabPlan
_class	String	Назва класу для здійснення документо-об'єктного відображення
firstName	String	Ім'я медичного працівника
lastName	String	Прізвище медичного працівника
title	String	Назва посади медичного працівника
compatibleProcedures	Array<DBRef>	Процедури, що даний працівник може проводити

Опис обладнання наявного в клініки зберігається в таблиці Equipment.

Таблиця 4.9 Опис таблиці Equipment

Ім'я поля	Тип поля	Призначення
_id	ObjectId	Унікальний ідентифікатор об'єктів в таблиці RehabPlan
_class	String	Назва класу для здійснення документо-об'єктного відображення
compatibleProcedures	Array<DBRef>	Процедури, що вимагають наявності даного обладнання

4.3 Робота з вхідними даними програмного комплексу

Разом з прямим завантаженням вхідних даних через графічні форми, програмний комплекс може працювати з вхідними даними наступного вигляду:

- текстовими файлами в форматі JSON;

- текстовими файлами в форматі XML;
- скан-копіями та фотокопіями планів терапіїв форматах PNG та JPEG.

Вхідні дані завантажуються в систему користувачами з відповідними ролями через графічний користувацький інтерфейс. Головними завданнями інтеграції даних є формування повного і несуперечного набору на основі множини різноманітних вхідних даних, отриманих з різноманітних джерел.

Перетворення та інтеграція файлових даних виконується шляхом застосування вбудованих бібліотек мови Java. Для роботи з файлами JSON використано бібліотеку конверсії Jackson. У випадку зчитування вхідних даних в форматі XML застосовується частина Java SDK – JAXB. Обидві бібліотеки діють за схожим принципом і забезпечують автоматичну генерацію Java-об'єктів.

Під час внесення даних проводиться їх верифікація:

- відповідність нових даних уже збереженим в базі даних;
- відсутність колізій та суперечностей між вхідними даними;
- відсутність дублікатів вхідних даних;
- цілісність вхідних даних.

Через велику варіативність мобільних пристроїв, фотокамер, сканувального обладнання проблематичним є отримання зображень достатньої різкості для наступного, тому перед виділенням текстової інформації зображення нормалізуються, з використанням програмної реалізації нейромережевого методу підвищення роздільної здатності [178]

Для роботи з зображеннями використовується бібліотека з відкритим програмним кодом Tesseract, що дає можливість автоматизованого конвертування графічних даних до текстового вигляду.

Після підтвердження коректності даних здійснюється їх збереження до бази даних.

4.3.1 Зчитування та лінгвістичний аналіз вхідних даних

Після зчитування та нормалізації вхідних даних здійснюється їх лінгвістичний аналіз за допомогою спеціалізованого компоненту програмного комплексу. Основні функції компоненту лінгвістичного аналізу тексту полягають у вилученні змісту із заданої текстової інформації. Тобто лінгвістичний компонент здійснює аналіз введеної текстової інформації згідно з алгоритмом, описаним у методі автоматизованого формування логіко-лінгвістичних моделей тексту, виокремлюючи сутності, властивості та концептуальні відносини між ними, таким чином готуючи матеріал для виведення у вигляді логіко-лінгвістичної моделі. Лінгвістичний компонент являє собою багаторівневий перетворювач, що містить три рівні представлення тексту: морфологічний, синтаксичний та семантичний. Кожному рівню відповідає один з етапів виконання методу автоматизованого формування логіко-лінгвістичної моделі відповідно: ідентифікація вхідного тексту, концептуалізація та синтаксичний аналіз.

4.3.2 Усунення суперечностей вхідних даних

Опрацювання суперечностей та невизначеностей вхідних даних проводиться незалежно

Проблема опрацювання, та інтерпретації суперечностей виникає як на рівні інтеграції синтаксису та структури даних, так і на рівні інтеграції семантики. Проблема неповноти, неточності чи відсутності даних є особливо чутливою у процесах інтеграції, коли необхідно об'єднати та узгодити дані, які мають різну природу різні способи зображення та різну інтерпретацію. При цьому в інтегрованому наборі даних потрібно забезпечити не лише об'єднане подання даних, але й визначити порядок подальшого опрацювання різного роду невизначеностей.

Предметною областю порівняльного аналізу текстових документів сформованих після опрацювання графічних зображень є природна, флективна

мова. Тому ця система ґрунтується на стандартних шаблонах, в які включені основні слова, що стосуються певної предметної області.

Для порівняльного аналізу використовується база сформована на основі введених попередніх даних. Це дає змогу під час звернення до бази даних надавати конкретних значень елементам формальної системи і внаслідок цього формувати масиви характеристик для кожного конкретного слова, що подається на вхід інтелектуальної системи.

4.4. Обґрунтування вибору технологій розробки програмного комплексу

В ході реалізації програмного комплексу оптимізації планів відновлювальної терапії було використано наступні технології:

- мова програмування високого рівня Java версії 1.8;
- NoSQL база даних MongoDB;
- платформа Java Enterprise Edition;
- Spring Framework.

Як інтегроване середовище розробки було використано IntelliJ Idea версії 2016. Дане середовище надає можливість автоматизованог генерування діаграм класів та має широкий спектр різних засобів інтеграції з іншими засобами для розробки програмного забезпечення: системами контролю версій, web-серверами, засобами вимірювання покриття коду та ін.

Для побудови проекту та узгодження залежностей між зовнішніми бібліотеками використано систему автоматичної збірки проектів Gradle.

Платформа Java Enterprise Edition [179] надає API та засоби виконання для розробки та запуску корпоративного програмного забезпечення такого як мережеві сервіси, web-сервіси, системи розподіленого типу тощо.

Java – об'єктно-орієнтована мова програмування з C подібним синтаксисом. Основною перевагою її є незалежність від від типів операційних систем та їх архітектури, тобто, написаний на Java код можна запускати на більшості з наявних платформ без змін у початковому кодї та їх перекомпіляції [180].

Spring Framework – універсальний комплекс програмних бібліотек з відкритим вихідним кодом для Java-платформи. Spring MVC – модуль Spring Framework, що базується на сервлетах та дає можливість розробляти web-орієнтовані прикладні програми і REST-сервіси. Його архітектура поділяє систему на три частини: модель даних, вигляд даних та керування. Дана архітектура застосовується для відокремлення даних від інтерфейсу користувача, щоб зміни інтерфейсу користувача мінімально впливали на роботу з даними, а зміни в моделі даних могли здійснюватися без змін інтерфейсу користувача [181]. Spring Data – модуль Spring Framework для здійснення роботи з серверами баз даних, перевагою його використання є абстракція від кінцевого типу бази даних, що використовується програмним комплексом.

MongoDB – нереляційна (NoSQL) система управління базами даних. Вона була створена у відповідь на обмеження традиційної технології реляційних баз даних. У порівнянні з реляційними, база даних MongoDB дозволяє краще масштабування та забезпечує вищу продуктивність, а її модель даних усуває недоліки реляційної моделі та надає наступні переваги [182]:

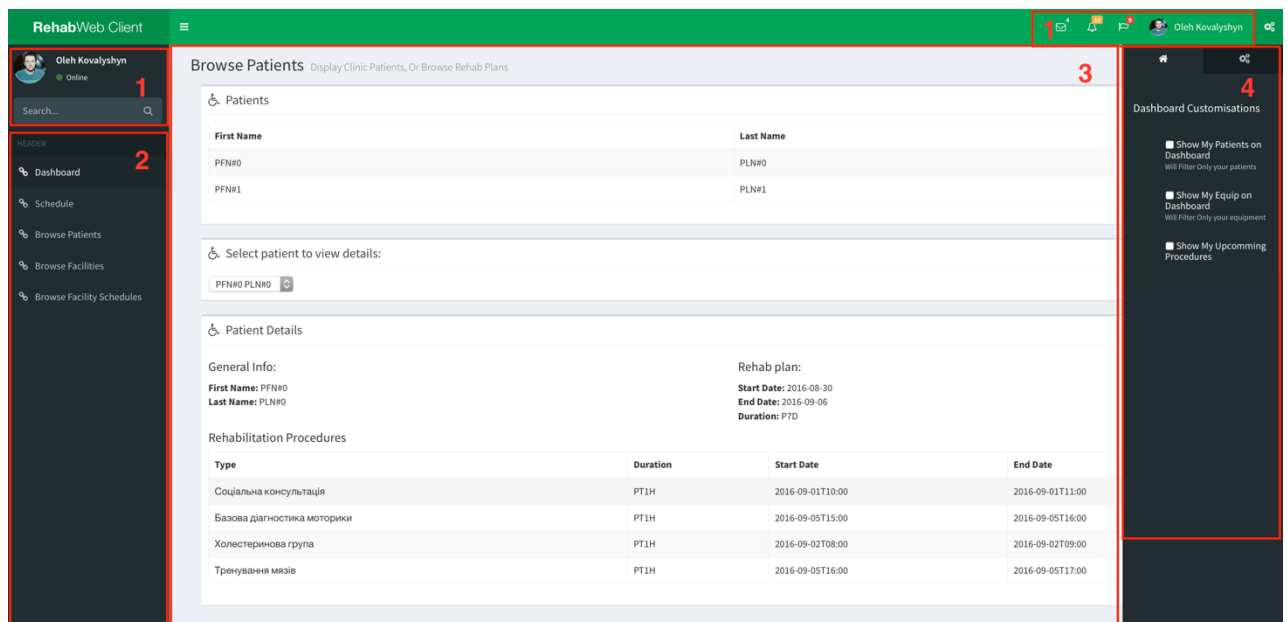
- документоорієнтованість – MongoDB не вимагає наявності чітко визначеної схеми даних. Кількість полів, вмісту та розмір одного документа може відрізнятися від одного документа до іншого;
- не вимагає розробки складних операторів об'єднання даних з різних таблиць;
- mongoDB підтримує динамічні запити до документів, використовуючи мову запитів на основі, майже таку ж потужну, як SQL;
- усуває необхідність використання систем об'єктно-реляційного відображення, тим самим підвищуючи швидкодію;
- можливість використання індексів побудованих на будь-яких полях документів, що пришвидшує пошук та запис даних.

MongoDB використовується для реалізації складних корпоративних систем з потребами високої швидкодії, наприклад: систем аналізу фінансових даних; моніторингу ринків інвестицій; міжнародних платіжних платформ; систем медичної інформації тощо.

4.5 Робота з графічним користувацьким інтерфейсом програмного комплексу

4.5.1 Загальні компоненти графічного користувацького інтерфейсу

Інтерфейс користувача (Рис. 4.5) програмного комплексу містить: меню користувача програми (1); головне меню програми (2); мультиекран (3) який в залежності від конфігурації (4) може складатись з списку усіх пацієнтів клініки або тільки пацієнтів поточного користувача обладнання закріпленого за поточним користувачем план-графіку функціонування клініки, або лише процедур поточного користувача.



4.5 Загальний вигляд інтерфейсу користувача.

Головне меню програми дозволяє навігацію від мультиекрану поточного користувача до конкретних функціональних екранів системи. Головне меню складається з таких підменю:

- “Перехід до мультиекрану користувача”, яке призначене для повернення до мультиекрану з функціональних екранів;
- “Пацієнти”, яке призначене для додання / видалення пацієнтів клініки та додання зміни реабілітаційних планів;
- “План-графік”, що призначене для побудови та оптимізації план-графіків клініки і планів відновлювальної терапії;

- “Обладнання”, яке призначене для додання / видалення пацієнтів клініки та додання зміни реабілітаційних планів;

Меню користувача програми містить наступні функції:

- “Профіль”, що призначене для перегляду профілю поточного користувача.
- “Допомога (F1)”, що викликає меню перегляду інструкцій користувача;
- “Вихід (Ctrl+E)”, що уможливорює вихід з програми без збереження налаштувань.

4.5.2. Управління пацієнтами та планами реабілітації

Маніпуляції з пацієнтами клініки та їх планами відновлювальної терапії проводяться з екрану «Пацієнти» (рис 4.6) який складається з: форми редагування процедури (1); форми редагування пацієнтів (2); форми перегляду поточних пацієнтів клініки (3); кнопок активізації режиму редагування пацієнта (4); панелі активізації форми детальної інформації про пацієнта (5); форми детальної інформації пацієнта (6); кнопок активізації режиму редагування процедур плану відновлювальної терапії (7).

The screenshot shows the 'Browse Patients' screen in the RehabWeb Client. The interface includes a sidebar with navigation options like Dashboard, Schedule, and Browse Patients. The main content area is titled 'Browse Patients' and contains several functional elements:

- 1:** 'Edit Procedure' button.
- 2:** 'Edit Patient' button.
- 3:** 'Patients' table listing patient information.
- 4:** 'Edit' buttons for each patient entry.
- 5:** 'Select patient to view details:' dropdown menu.
- 6:** 'Patient Details' section showing general info and rehab plan details.
- 7:** 'Edit' buttons for each row in the 'Rehabilitation Procedures' table.

Type	Duration	Start Date	End Date	Edit
Соціальна консультація	PT1H	2016-09-01T10:00	2016-09-01T11:00	<input type="checkbox"/>
Базова діагностика моторики	PT1H	2016-09-05T15:00	2016-09-05T16:00	<input type="checkbox"/>
Холестеринова група	PT1H	2016-09-02T08:00	2016-09-02T09:00	<input type="checkbox"/>
Тренування м'язів	PT1H	2016-09-05T16:00	2016-09-05T17:00	<input type="checkbox"/>

Рис 4.6 Екран “Управління пацієнтами та планами реабілітації”.

Додавання нового пацієнта здійснюється з екрану «Пацієнти» за допомогою форми «Додання пацієнта» (рис 4.7).

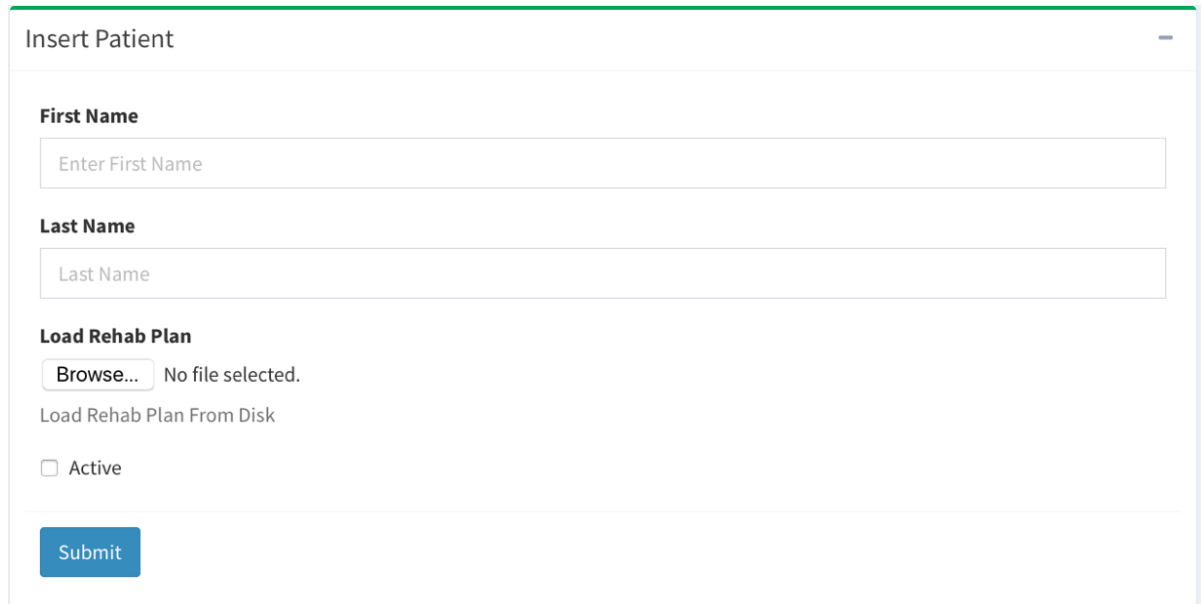


Рис 4.7. Форма додавання пацієнта.

Для додавання нового пацієнта до системи необхідно ввести його ім'я, прізвище а також завантажити план відновлювальної терапії з диску.

Для зміни контактних даних пацієнта, а також зміни плану відновлювальної терапії необхідно обрати пацієнта з списку існуючих пацієнтів та натиснути лівою кнопкою мишки на кнопку активізації режиму редагування пацієнта. В результаті активізується форма редагування деталей пацієнта (Рис. 4.8).

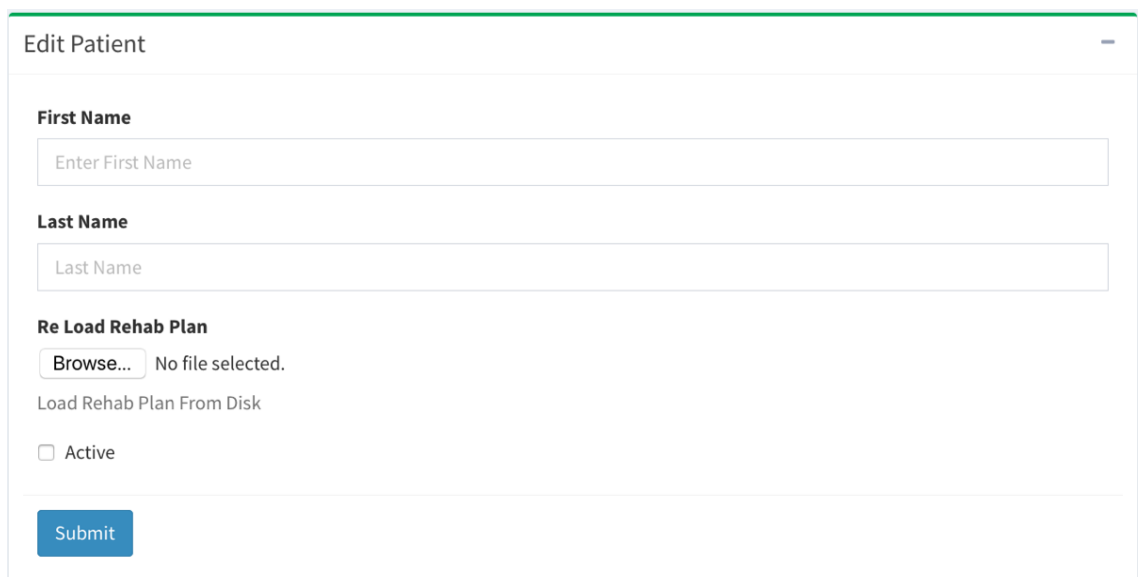


Рис 4.8. Форма редагування деталей пацієнта

У випадку необхідності зміни деталей процедур що входять до плану реабілітації, необхідно обрати пацієнта, для якого здійснюється заміна з випадаючого меню. На формі деталей плану реабілітації необхідно клікнути лівою кнопкою мишки на кнопку активізації режиму зміни деталей процедури. В результаті стане активною форма внесення змін до процедури (рис.4.8).

Рис 4.9. Форма внесення змін до процедури

4.5.3 Побудова та оптимізація план-графіку

Побудова та оптимізація план-графіку здійснюються на екрані «План-графік». (рис. 4.10), що складається з: форм побудови план-графіку для заданих відрізків часу (1); оптимізації план-графіку для певних часових відрізків (2); списку лікувальних приміщень(3); календаря для наочної репрезентації робочого план-графіку клініки. Перехід до екрану план-графіку здійснюється з головного меню програми. Побудова та оптимізація план-графіку здійснюється на основі даних попередньо введених на екрані «Пацієнти», за допомогою внутрішніх алгоритмів та внесених у систему обмежень.

The screenshot displays the 'RehabWeb Client' interface. On the left is a dark sidebar with navigation options: Dashboard, Schedule, Browse Patients, Browse Facilities, and Browse Facility Schedules. The main content area is titled 'Schedule generation page' and contains four numbered sections:

- 1. Schedule Options:** A form with 'Start Date Type' (dropdown), 'Start Date' (text input), 'Select new Procedure type' (dropdown), 'End Date' (text input), and a blue 'Build' button.
- 2. Optimisation Options:** A similar form with an 'Optimise' button.
- 3. Available rehabilitation facilities:** A grid of 15 checkboxes for different facility types, such as 'Кімната для: Візит до лікаря' and 'Кімната для: Соціальна консультація'.
- 4. Calendar:** A monthly calendar for June 2017, showing dates from 28 to 10.

4.10. Екран побудови та оптимізації план-графіків відновлювальної терапії

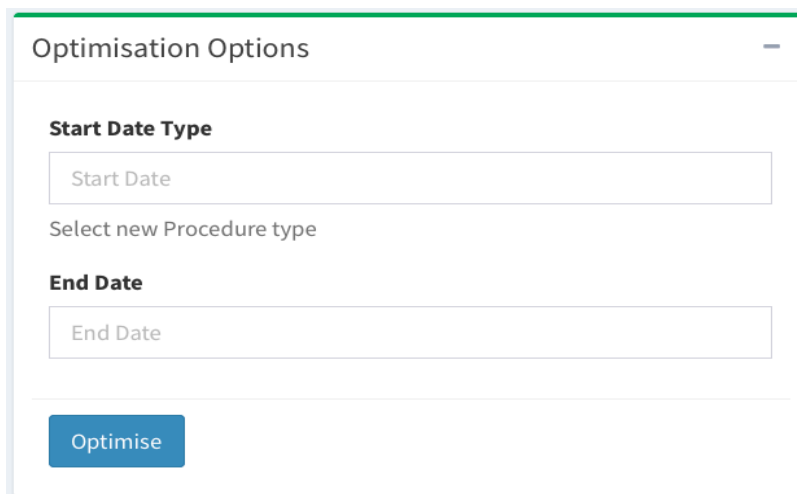
Для початку побудови план-графіку необхідно ввести дати початку та закінчення часового відрізка побудови план-графіку до форми його побудови (рис 4.11), та натиснути лівою клавішею миші на кнопку «Збудувати». Після закінчення роботи алгоритму результати будуть відображені на вбудованому календарі системи.

This is a close-up view of the 'Schedule Options' form. It contains the following elements:

- Start Date Type:** A dropdown menu.
- Start Date:** A text input field.
- Select new Procedure type:** A dropdown menu.
- End Date:** A text input field.
- Build:** A blue button at the bottom.

4.11. Форма задання параметрів побудови план-графіку

Для оптимізації план-графіку (рис 4.12) необхідно ввести дати початку та закінчення часового відрізка оптимізації план-графіку, і натиснути лівою клавішею миші на кнопку «Оптимізувати». Після закінчення роботи алгоритму оптимізації результуючі зміни в план-графікі відобразяться на календарі.



The image shows a software dialog box titled "Optimisation Options". It has a white background and a thin border. At the top left is the title "Optimisation Options" and a minus sign icon. Below the title, there is a section labeled "Start Date Type" with a text input field containing "Start Date". Underneath this is the text "Select new Procedure type". Below that is a section labeled "End Date" with another text input field containing "End Date". At the bottom of the dialog is a blue button with the text "Optimise" in white.

4.12. Форма задання параметрів оптимізації план-графіку

Отримані в процесі побудови та оптимізації план-графіки автоматично зберігаються в базі даних, та стають доступними всім користувачам системи з відповідними правами доступу.

4.6. Практична апробація програмного комплексу та порівняння з існуючими системами відновлювальної терапії .

Роботу розробленого програмного комплексу апробовано на тестових вибірках, побудованих на основі реальних планів відновлювальної терапії (табл. 4.10). Для оцінки ефективності роботи методу з точки зору пацієнтів та медичного персоналу були проведені опитування за десятибальною шкалою щодо якості розроблених план-графіків до і після їх багатокритеріальної оптимізації. Сумарно було опрацьовано відповіді п'ятдесяти семи респондентів. Агреговані значення результатів опитування наведено в таблиці 4.11.

Таблиця 4.10

Результати оптимізації розкладів відновлювальної терапії

Вибірки	Початкові та кінцеві результати оцінки	Результат нечіткої оцінки	Нормалізовані значення порушень критеріїв				
			критерій №1	критерій №2	критерій №3	критерій №4	критерій №5
№1	початкове	0.4049	0.3717	0.1441	0.3614	0.4049	0.1429
	кінцеве	0.3366	0.3037	0.1135	0.3304	0.3366	0.1419
	різниця	↑ 0.0683	↑ 0.0681	↑ 0.0306	↑ 0.031	↑ 0.0683	↑ 0.001
№2	початкове	0.4324	0.4324	0.4143	0.3385	0.3667	0.4102
	кінцеве	0.3461	0.2703	0.2857	0.3363	0.2667	0.3813
	різниця	↑ 0.0863	↑ 0.1621	↑ 0.1286	↑ 0.0022	↑ 0.1	↑ 0.0289
№3	початкове	0.4324	0.4324	0.4143	0.3385	0.3667	0.3751
	кінцеве	0.3385	0.3378	0.3	0.3315	0.2667	0.2981
	різниця	↑ 0.0939	↑ 0.0946	↑ 0.1143	↑ 0.007	↑ 0.1	↑ 0.077

Таблиця 4.11

Результати опитування пацієнтами та медичним персоналом якості розроблених план-графіків до і після їх багатокритеріальної оптимізації

Вибірки	Результати оцінки вибірок пацієнтами			Результати оцінки вибірок медичними працівниками		
	початкова оцінка	кінцева оцінка	різниця	початкова оцінка	кінцева оцінка	різниця
№1	6.7	8.4	↑ 24.6%	5.3	8.1	↑ 42.3%
№2	7.1	9.2	↑ 29.0%	6.4	7.9	↑ 20.9%
№3	6.5	8.8	↑ 34.5%	7.0	9.1	↑ 32.0%

Наведені в табл.4.10, 4.12 результати досліджень свідчать, що розроблений метод забезпечує оптимізацію план-графіків відновлювальної терапії без відриву від середовища їх проведення – медичного закладу, з врахуванням вимог процесу лікування як жорстких обмежень, а також побажань пацієнтів та медичного персоналу – м'яких обмежень. Використання генетичного алгоритму дає можливість досягнути розв'язку, який відповідає множині накладених критеріїв.

Для розробленого програмного комплексу було здійснено порівняння відносно найбільш розповсюджених на ринку систем, що використовуються в ході організації процесу реабілітації. До них відносять такі: Универсальная Система Учета "програма реабилитации", ECLIPSE Practice Management Software, Chartlinks, OptimisPT, Ptlive, TheraPlatform, My Rehab Pro, PracticePerfect.

Розроблений програмний комплекс дає можливість одночасної взаємодії великої кількості лікарів, персоналу та пацієнтів, тому використання архітектури тонкого клієнта дозволяє налагодити, спростити та пришвидшити взаємодію між користувачами. Порівняльну характеристику існуючих та розробленого програмного комплексу наведено у таблиці 4.10 (« - » - елемент відсутній, « -/+ » - елемент частково присутній, « + » - елемент присутній).

Таблиця 4.12

Порівняльний аналіз систем відновлювальної терапії

Критерій	Универсальная Система Учета	ECLIPSE Practice	Chartlinks	OptimisPT	Ptlive	TheraPlatform	My Rehab Pro	PracticePerfect	Розроблений
Підтримка операційних систем									
Windows	+	+	+	+	+	+	+	+	+
Linux	-	-	-	-	+	+	+	+	+
MacOS	-	-	-	-	+	+	+	+	+
Підтримка мобільних платформ									
iOS	-	-	-	-	-	+/-	+	+	+
Android	-	-	-	-	-	+	+	+	+
Наявність рівнів доступу користувачів	+	+	+	+	-	+/-	-	-	+

Продовження таблиці 4.12

Наявність бази даних	+	+	+	+	-	+	-	-	+
Автоматичне внесення вхідних лінгвістичних даних	-	-	-	-	-	-	-	-	+
Автоматична побудова план-графіків	-	-	-	-	-	-	-	-	+
Автоматизована оптимізація план-графіків	-	-	-	-	-	-	-	-	+
Підтримка планів відновлювальної терапії	-	-	-	-	+/-	+/-	+/-	+/-	+
Облік пацієнтів та медичного обладнання	+	+	+	+	+		+	-	+
Опис якісних ознак якості план-графіків	-	-	-	-	-	-	-	-	+

В результаті порівняльного аналізу систем можна зробити висновок, що розроблений програмний комплекс задовольняє усі вимоги до ПЗ та може успішно використовуватись у сучасних медичних закладах. До ключових переваг можна віднести наявність засобів автоматичної генерації та багатокритеріальної оптимізації план-графіків відновлювальної терапії, модуль обліку пацієнтів та наявність адаптивних графічних інтерфейсів для різних типів користувачів.

ВИСНОВКИ ДО РОЗДІЛУ 4

1. В розділі розглянуто реалізацію програмного комплексу, призначеного для побудови оцінки та оптимізації план-графіків відновлювальної терапії, в якому реалізовано методи і алгоритми розглянуті в попередніх розділах. Використання архітектури “тонкого клієнта” забезпечує високий рівень відмовостійкості, незалежність від апаратної архітектури клієнтів та їх операційних систем та простоту додавання нового функціоналу.

2. Застосування трирівневої архітектури програмного комплексу дає можливість автоматичного централізованого оновлення програмного комплексу, незалежного конфігурування на кожному з рівнів архітектури, централізованого контролю за рахунок використання механізмів авторизації та автентифікації.

3. Розроблений програмний комплекс надає можливість розгортання на більшості сучасних платформ та операційних систем за рахунок використання мови програмування високого рівня – Java.

4. Використання сучасних технологій розробки: платформи Java Enterprise Edition, Spring Framework, спрощує розуміння програмного комплексу для сторонніх розробників в процесі його подальшого розвитку

5. Застосування нереляційної бази MongoDB забезпечує високу продуктивність роботи програмного комплексу, можливість масштабування, а також високу цілісність даних, за рахунок їх дуплікації.

6. Розроблений графічний користувацький інтерфейс дає можливість зручної роботи на більшості сучасних пристроїв, за рахунок автоматичної перебудови вікон під розширення екрану клієнта.

ВИСНОВКИ

У дисертаційній роботі розв'язано актуальну наукову задачу розроблення ефективних методів автоматизованої побудови план-графіків медичних закладів та їх багатокритеріальної оптимізації завдяки використанню елементів нечіткої логіки та генетичних алгоритмів.

1. Проведеним аналізом існуючих методів складання план-графіків встановлено, що їх використання на практиці є недостатньо ефективними, оскільки вони не здатні забезпечити врахування багатоваріантності та провести багатокритеріальну оцінку. У зв'язку з цим існує необхідність розроблення нових ефективних методів автоматизованого складання план-графіків медичних установ та їх оптимізації, які б врахували ступінь використання наявних в них ресурсів та оцінювали якість медичних послуг з точки зору пацієнта, використовуючи підхід перетворення лінгвістичних термінів в кількісні змінні за допомогою нечіткого логічного контролера.

2. Дослідженнями процесу розроблення план-графіку для проведення відновлювальної терапії встановлено, що він повинен враховувати множини обмежень, як жорстких – умов, що повинні обов'язково виконуватися для забезпечення коректності план-графіку, так і м'яких – виконання яких є бажаним. До першої з них відносять узгодження план-графіків використання медичного обладнання, наявність кваліфікованого медичного персоналу, визначений час для проведення кожної конкретної процедури, доступність до відповідного устаткування тощо, а до другої – побажання пацієнтів та врахування їх інтересів. На підставі цього встановлено, що побудова план-графіку медичного закладу є багатокритеріальною задачею, вирішення якої потребує узгодження пред'явлених до нього медичних, методичних і організаційних обмежень та отримання найбільш оптимального рішення при різноманітних варіаціях початкових даних – множини планів відновлювальної терапії.

3. Для вирішення багатокритеріальної задачі розроблення та оптимізації план-графіку запропоновано підхід, що складається з трьох основних етапів:

побудови опорного план-графіку, що задовольняє жорстким обмеженням; оцінки його якості; послідовної оптимізації план-графіку.

4. Для побудови опорного план-графіку запропоновано універсальний метод, який ґрунтується на використанні базової моделі – скінченного автомату Мілі, що дозволяє найточніше врахувати специфіку досліджуваної предметної області. Розроблений скінченний автомат забезпечує процедуру послідовної побудови опорного план-графіку за умов присутності певної множини обмежень, вирішення в ньому конфліктних ситуацій та уможлиблює послаблення деяких заданих жорстких обмежень при тупикових ситуаціях в процесі роботи.

5. Для трансформування нечітких даних, отриманих в результаті збору відгуків пацієнтів, у кількісні змінні було розроблено контролер нечіткої логіки на основі системи T-Controller. Завдяки цьому стало можливим провести оцінку відгуків пацієнтів про якість роботи клініки та встановити рівень їх вдоволеності від отриманих лікувальних послуг.

6. Реалізувавши на практиці контролер нечіткої логіки та агрегувавши експертні знання, було здійснено багатокритеріальну оцінку якості план-графіку медичних установ як зі сторони пацієнтів, так і самих клінік. Побудована в результаті цього інтегральна оцінка виступає базисом для подальшої оптимізації цих план-графіків, дозволяє впливати на них як зі сторони ефективного використання ресурсів, так і зі сторони вдоволеності пацієнтів від отриманих послуг, що сприяє суттєвому підвищенню ефективності діяльності закладів медичного спрямування.

7. Проведено багатокритеріальну оптимізацію план-графіків відновлювальної терапії завдяки використанню методів генетичних алгоритмів. В них було проведено кодування план-графіку функціонування клініки у вигляді хромосоми, здійснено генерацію початкової популяції план-графіків, обґрунтовано та розроблено механізм селекції хромосом та способи рекомбінації у них генів, що в сукупності дозволили інтенсифікувати процес пошуку оптимальної комбінації план-графіку лікувального закладу.

8. Для побудови, оцінки та оптимізації план-графіків відновлювальної терапії було розроблено програмний комплекс, в якому використано архітектуру “тонкого клієнта”. Завдяки цьому забезпечується незалежність від апаратної архітектури клієнтів та їх операційних систем, простота додавання нового функціоналу, висока безпека та високий рівень стійкості проти відмов.

9. Запропонований програмний комплекс є основою інформаційної технології автоматизованої побудови та оптимізації план-графіків відновлювальної терапії. Використання під час його розробки сучасних технологій Java Enterprise Edition, Spring Framework, нереляційної бази MongoDB забезпечує високу продуктивність роботи програмного комплексу, можливість масштабування та високу цілісність даних.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] «The rehabilitation of disabled people in Ukraine: Zakon Ukrainy vid 06.10.2005 No 2961,» 10 08 2018. [Онлайновий]. Available: <http://zakon2.rada.gov.ua/laws/show/2961-15>. [Дата звернення: 10 08 2018].
- [2] В. В. Пожидаєв, Фізіотерапія. Організація роботи фізіотерапевтичних кабінетів і відділень в центрах медичної реабілітації та санаторно-курортних закладах, Купріянова О.О., 2006.
- [3] «The implementation of decisions of enlarged meeting of the Health Ministry of Ukraine, November 2, 2012: nakaz MOZ Ukrainy vid 07.12.2012 r. No 999,» 11 10 2018. [Онлайновий]. Available: <http://mozdocs.kiev.ua/view.php?id=14162>. [Дата звернення: 11 10 2018].
- [4] «Про затвердження Положення про індивідуальну програму реабілітації інваліда,» 04 01 2017. [Онлайновий]. Available: <http://zakon.rada.gov.ua/laws/show/757-2007-%D0%BF>. [Дата звернення: 10 11 2018].
- [5] I. Sarancha, «The role of social rehabilitation centers in the socialization persons with injuries of musculoskeletal system,» *Visn. Kamianets-Podil. nats. un-tuim. I. Ohienka. Korekts. pedahohika i psykhohohiia*, т. 3, pp. 119-117, 2011.
- [6] Н. Tymoshko, The organization of activity of public and specialized for Social Services, Nizhyn: Vyd. PP Lysenko, 2011.
- [7] I. O. Medicine, Crossing the Quality Chasm: A New Health System for the 21st Century. Washington, DC: The National Academies Press; 2001. 2., Washington, DC: The National Academies Press, 2001.
- [8] R. Gafarov E, A. A. Lazarev, «Proof of NP-hardness particular case of the problem of minimizing the total delay for one device.,» *Izvestiya AN: Theory and management systems*, т. 3, pp. 120-128, 2006.
- [9] H. Lee, I. Vlaev, D. King, E. Mayer, A. Darzi , P. Dolan, «Subjective well-being and the measurement of quality in healthcare. . 2013; 3.09.027, 99:27-34. doi: 10.1016/j.socscimed.201,» *Soc Sci Med*, т. 99, pp. 27-34, 2013.
- [10] W. E. Smith, «Various optimizers for single stage production,» *Naval Research Logistics Quarterly*, т. 3, p. 59–66, 1956.
- [11] В. Танаєв, В. Шкурба, Введение в теорию расписаний, Наука, 1975.
- [12] H. Gantt, «ASME Transactions,» *AMS*, т. 24, p. 1322–1336, 1903.
- [13] R. Bellman, «Mathematical aspects of scheduling theory,» *Journal of the Society of Industrial and Applied Mathematics*, т. 4, p. 168–205, 1956.

- [14] M. Johnson S, «Optimal two- and three-stage production schedules with setup times included,» *Naval Research Logistics Quarterly*, т. 1, p. 61–68, 1954.
- [15] J. R. Jackson, Scheduling a production line to minimize maximum tardiness, т. 43, Los Angeles: Los Angeles University of California Management Sciences Office of Technical Services, 1955.
- [16] W. Conway R, L. Maxwell W , W. Miller L, Theory of Scheduling, MA: Addison-Wesley Reading, 1967.
- [17] Ю. А. Зак, Прикладные задачи теории расписаний и маршрутизации перевозок, Editorial URSS, 2018.
- [18] В. Сергиенко И, Задачи дискретной оптимизации. Проблемы, методы решения, исследования, Наукова думка, 2003.
- [19] О. Ковалишин, «Аналіз методів оптимізації розкладів в контексті відновлювальної терапії,» *Науковий вісник НЛТУ України: зб. наук. праць*, т. 28, № 8, pp. 136-140, 2018.
- [20] E. G. Coffman, Computer and job-shop scheduling theory, New Yourk: Wiley, 1984.
- [21] P. Baptiste, C. Le Pape , W. Nuijten, «Constraint-based scheduling: applying constraint programming to scheduling problems, Academic Publishers,» Springer US, 2001.
- [22] J. Thompson , K. Dowsland, «Variants of simulated annealing for the examination timetabling problem. Annals of Operational Research, 63, 195.,» *Annals of Operations Research* , т. 63, pp. 105-128, 1996.
- [23] А. А. Лазарев, «Теория расписаний. Оценки абсолютной погрешности и схема приближенного решения задач теории расписаний,» МФТИ, 2008.
- [24] Э. Г. Коффман, Теория расписаний и вычислительные машины, Москва: Наука, 1984.
- [25] J. Blazewicz, K. Ecker, E. Pesch, G. Schmidt , J. Weglarz, Handbook on Scheduling: From Theory to Applications, Springer, 2007.
- [26] Б. Мильнер, Теория организации. 6-е изд. (перераб. и доп.), Москва: ИНФРА, 2008.
- [27] И. Л. Акулич, «Математическое программирование в примерах и задачах,» Лань, СПб, 2011.
- [28] А. Кофман, Введение в прикладную комбинаторику, Москва: Наука, 1975.
- [29] Э. Рейнгольд, Ю. Нивергельт , А. Део, «Комбинаторные алгоритмы. (Теория и практика),» Мир, Санкт-Петербург, 1980.

- [30] R. Gomory, An all integer programming algorithm, Research Report. RC-189, 1960.
- [31] И. В. Романовский, «Алгоритмы решения экстремальных задач,» Наука, Москва, 1977.
- [32] О. А. Павлов , Л. О. Павлова, «ПДС-алгоритми для важкорозв'язуємих комбінаторних задач. Теорія і методологія розробки.,» Поличка «Карпатського краю», Ужгород, 1998.
- [33] W.-J. Hoesve, M. Hunting , C. Kuip, «The AIMMS Interface to Constraint Programming,» AIMMS, 2012.
- [34] R. Barták, «Constraint-Based Scheduling: An Introduction for Newcomers,» *IFAC Proceedings Volumes*, т. 36, № 3, pp. 75-80, 2001.
- [35] Y. Bukchin, «Constraint programming for solving various assembly line balancing problems,» *Omega*, т. 78, pp. 57-68, 2018.
- [36] K. Apt, Principles of Constraint Programming, New York: Cambridge University Press, 2003.
- [37] R. Graham, E. Lawler , J. Lenstra, «Optimization and approximation in deterministic sequencing and scheduling: A survey.,» *Annals of Discrete Mathematics*, т. 5, pp. 287-326, 1979.
- [38] P. Brucker , S. Knust, «Resource-constrained project scheduling and timetabling,» *Lecture Notes in Computer Science*, т. 2079, pp. 277-293, 2001.
- [39] G. Zhu, J. Bard , G. Yu, «A branch-and-cut procedure for the multimode resource-constrained project-scheduling problem,» *INFORMS Journal on Computing*, т. 18, № 3, pp. 377-390, 2006.
- [40] K. Neumann, C. Schwindt , N. Trautmann, «Scheduling of continuous and discontinuous material flows with intermediate storage restrictions. European Journal of Operational Research,» т. 165, № 2, pp. 495-509, 2005.
- [41] T. Kis, «A branch-and-cut algorithm for scheduling of projects with variable-intensity activities,» *Mathematical Programming*, т. 103, № 3, pp. 515-539, 2005.
- [42] J. Jozefowska, M. Mika, R. Rozycki, G. Waligora , W. J. , «Solving the discrete-continuous project scheduling problem via its discretization,» *Mathematical Methods of Operations Research*, т. 52, № 3, pp. 489-499, 2005.
- [43] M. Vanhoucke, «Setup times and fast tracking in resource-constrained project scheduling,» *Computers and Industrial Engineering*, т. 54, № 4, pp. 1062-1070, 2008.

- [44] В. С. Михалевич, Методы последовательной оптимизации в дискретных сетевых задачах оптимального распределения ресурсов, Москва: Наука, 1983.
- [45] N. Beldiceanu , E. Contejean, «Introducing global constraints in CHIP,» *Journal of Mathematical and Computer Modeling*, т. 20, pp. 97-123, 1994.
- [46] F. Rossi, P. Van Beek , T. Walsh, Handbook of Constraint Programming, Elsevier Science, 2006.
- [47] S. Bistarelli, U. Montanari , F. Rossi, «Semiring-based constraint solving and optimization,» *Journal of the ACM*, т. 44, pp. 201-236, 1997.
- [48] S. Brailsford, C. Potts , B. Smith, «Constraint satisfaction problems: Algorithms and applications,» *European Journal of Operational Research*, т. 119, pp. 557-581, 1999.
- [49] S. Arnborg, «E□cient algorithms for combinatorial problems on graphs with bounded decomposability □ a survey // BIT.. V. 25. P. 2-23.,» *BIT - Ellis Horwood series in artificial intelligence*, т. 25, № 1, pp. 2-23, 1985.
- [50] N. Barnier , P. Brisset, «Graph coloring for air tra□c □ow management,» *Annals of Operations Research*, т. 130, pp. 163-178, 2004.
- [51] N. Beldiceanu, Global constraints as graph properties on structured network of elementary constraints of the same type, Uppsala: Technical Report T2000/01, SICS , 2000.
- [52] N. Biggs, E. Lloyd , R. Wilson, Graph Theory 1736-1936, Oxford: Oxford University Press, 1976.
- [53] C. Bliet, B. Neveu , G. Trombettoni, «Using graph decomposition techniques for solving continuous CSPs,» в *Principles and Practice of Constraint Programming — CP98.*, Berlin, 1998.
- [54] A. Blum , M. Furst, «Fast planning through planning graph analysis,» *Artificial Intelligence*, т. 90, № 1, pp. 281-300, 1997.
- [55] A. Bachelu, Multi-criteria comparison between algorithmic, constraint logic and speci□c constraint programming on a real scheduling problem, Proc. PACT97, 1997.
- [56] M. Baumgart, Priority-driven constraints used for scheduling at universities, Proc. PACT97, 1997.
- [57] J. Gregory, Constrained Optimization In The Calculus Of Variations and Optimal Control Theory, CRC Press, 2018.
- [58] S. Martello , P. Toth.-Chichester, «Knapsack problems: algorithms and computer implementations,» John Wiley & Sons, Ltd, 1990.

- [59] H. Schwefel, Numerical optimization of computer models, Schwefel.-Chichester: Wiley, 1981..
- [60] S. Goodman , S. Hedetniemi, «Introduction to the Design and Analysis of Algorithms,» McGraw-Hill, 1977.
- [61] P. Brucker , S. Knust, Complex scheduling, Berlin: Springer, 2006.
- [62] J. R. Koza, Genetic Programming IV: Routine Human-Competitive Machine Intelligence, Berlin: Springer, 2005.
- [63] S. Kirkpatrick, C. Gelatt , M. Vecchi, «Optimization by simulated annealing,» *Science*, т. 220, pp. 671-680, 1983.
- [64] В. Костенко , А. В. Калашников, «Исследование различных модификаций алгоритмов имитации отжига для решения задачи построения многопроцессорных расписаний,» в *Тр. седьмой междунар. научн. конфер. "Дискретные модели в теории управляющих систем"*, Москва, 2006.
- [65] B. Kuipers, Qualitative Reasoning : Modeling and Simulation with Incomplete Knowledge, MIT Press, 1994.
- [66] P. Prosser, «reactive scheduling agent,» в *Proceedings of the 11th International Joint Conference on Artificial Intelligence IJCAI-89*, Detroit, 1989.
- [67] В. Васильев, В. Коноваленко , Ю. Горелов, Имитационное управление неопределенными объектами, Наукова думка, 1989.
- [68] R. Rodosek , M. Wallace, «A generic model and hybrid algorithm for hoist scheduling problems,» в *Proc. 4th Int. Conf. on Principles and Practice of Constraint Programming (CP98)*, 1998.
- [69] Л. А. Овчаров , С. Вентцель Е., Теория вероятностей и ее инженерные приложения: учебное пособие для вузов, Высшая школа, 2007.
- [70] А. Н. Скурихин, «Генетические алгоритмы,» *Новости искусственного интеллекта.*, т. 4, pp. 6-48, 1995.
- [71] T. Blickle , L. Thiele, «Comparison of Selection Schemes used in Genetic Algorithm,» ТИК-Report, Blickle, L. Thiele, 1992.
- [72] R. Bruns, «Direct Chromosome Representation and Advanced Genetic Operators for Production Scheduling,» в *Proc. of 5th Int. Conf. on GA*, San Francisco, 1993.
- [73] T. N. Bui , B. R. Moon, «A new approach on the traveling salesman problem by genetic algorithms,» *Evolutionary Computation*, т. 3, pp. 7-12, 1994.
- [74] P. Cohoon J, N. Martin W , S. Richards D, A multi-population genetic algorithm for solving the k-partition problem on hyper-cube. -, San Diego: ICGA, 1991..
- [75] E. A. Eiben, «Introduction to Evolutionary Computing,» Springer-Verlag, Berlin, 2003.

- [76] L. J. Fogel , A. J. Owens, «Artificial Intelligence through Simulated Evolution,» Walsh.-USA, 1966.
- [77] D. Goldberg, Genetic Algorithms In Search, Optimization, and Machine Learning, USA: Addison-Wesley Publishing Company, 1989.
- [78] L. Davis, «Handbook of Genetic Algorithms,» USA, New York, 1992.
- [79] A. H. Land , A. G. Doig, «An automatic method of solving discrete programming problems,» *Econometrica*, т. 28, № 3, p. 497–520, 1960.
- [80] Е. В. Панкратьев , А. М. Чеповский, «Алгоритмы и методы решения задач составления расписаний и других экстремальных задач на графах больших размерностей,» *Фундаментальная и прикладная математика*, т. 9, pp. 235-251, 2003.
- [81] L. Bianco, M. Caramia , P. Dell’olmo, «Solving a preemptive project scheduling problem with coloring techniques,» Springer, Boston, MA, 1999.
- [82] A. Khachaturyan, S. Semenovskaya , B. Vainshtein, «Statistical-Thermodynamic Approach to Determination of Structure Amplitude Phases,» *Sov.Phys. Crystallography*, т. 24, № 5, p. 519–524, 1979.
- [83] V. Černý, «Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm,» *Journal of Optimization Theory and Applications*, т. 45, p. 41–51, 1996.
- [84] J. Holland, «Adaptation in Natural and Artificial Systems,» Bradford Book, 1992.
- [85] О. С. Ковалишин, «Аналіз методів оптимізації розкладів в контексті відновлювальної терапії,» *Науковий вісник НЛТУ України: зб. наук. праць* , т. 28, № 8, pp. 136-140, 2018. .
- [86] R. Nakano, «Conventional Genetic Algorithm for Job Shop Problems,» в *Proc. of 4th Int. Conf. on GA, Morgan Kaufmann Publ.*, San Francisco,, 1991.
- [87] S. Kobayashi, L. Ono , M. Yamamura, «An Efficient Genetic Algorithm for Job Shop Scheduling Problem,» в *Proc. of 6th Int. Conf. on GA, Morgan Kaufmann Publ.*, San Francisco, 1995.
- [88] Z. Michalewicz, «Genetic Algorithms + Data Structures = Evolution Programs,» Springer-Verlag Berlin, Heidelberg, 1996.
- [89] D. Wolpert , W. Macready, «No free lunch theorems for optimization.,» *IEEE Transactions on Evolutionary Computation*, т. 1, № 1, pp. 67-82, 1997.
- [90] В. ПОДИНОВСКИЙ , В. Д. НОГИН, «Парето-оптимальные решения многокритериальных задач,» Гаука, Москва, 1982.
- [91] S. Zavriev, «On the Global Optimization Properties of Finite-difference Local Descent Algorithms,» *J. of Global Optimization.*, т. 3, pp. 63-78, 1993.

- [92] И. Г. Черноруцкий, «Методы оптимизации и принятия решений: Учебное пособие,» Издательство «Лань», Санкт-Петербург, 2001.
- [93] В. Михайлович , А. Кукса, «Методы последовательной оптимизации в дискретных сетевых задачах оптимального распределения ресурсов,» Наука, Москва, 1983.
- [94] И. Ю. Мирецкий, «Субоптимальные решения задачи об М станках,» *Изв. РАН. ТуСУ.*, т. 1, pp. 45- 78, 2004.
- [95] В. Branke, K. Deb, K. Miettinen , R. Slowinski, «Multiobjective Optimization: Interactive and Evolutionary Approaches (Lecture Notes in Computer Science),» Springer, 2008.
- [96] A. Osyzka, «An approach to multicriterion optimization problems for engineering design,» *Computer Methods in Applied Mechanics and Engineering*, т. 15, № 3, pp. 309-333, 1978.
- [97] N. R. Conn, M. Gould N , L. Toint Ph, Trust-Region Methods, MPS/SIAM Series on Optimization, SIAM and MPS, 2000w.
- [98] Р. Х. Кини Р. Л., Принятие решений при многих критериях: предпочтения и замещения, Москва: Радио и связь, 19981.
- [99] Т. Саати, «Принятие решений. Метод анализа иерархий,» Радио и связь, 1993.
- [100] J. Jahn, Vector Optimization: Theory, Applications and Extensions, Berlin: Springer-Verlag, 2004.
- [101] Ю. К. Машунин, «Методы и модели векторной оптимизации,» Наука, Москва, 1986.
- [102] В. В. Токарев, «Методы оптимальных решений. Многокритериальность. Динамика. Неопределенность,» ФИЗМАТЛИТ, Москва, 2011.
- [103] Т. Н. Кукушкина, Ю. М. Докиш , Н. А. Честякова, Руководство по реабилитации больного, частично утративших трудоспособность, Москва: Медицина, 1989.
- [104] В. Н. Сокрут, В. Н. Казаков , В. Я. Уманский, «Медицинская реабилитация в Украине : современные положения и перспективы развития,» *Новости медицины и фармации в мире.*, т. 10, № 138, pp. 19-20, 2003.
- [105] В. Л. Мисула І.Р., Медична , соціальна реабілітація : Навчальний посібник, Тернопіль: ТДМУ, 2-5.
- [106] К. И. Журавлева, «Организация и эффективность реабилитации в медицинских учреждениях.,» Медицина, 1980.

- [107] О. В. Сергієні, «Організація , управління системою медичної реабілітації інвалідів : Методичні рекомендації,» КУІМБР, Дніпропетровськ, 2001.
- [108] С. П. Миронов, «Состояние ортопедо-травматологической службы в Российской Федерации и перспективы внедрения инновационных технологий в травматологии и ортопедии,» *Вестник травматологии и ортопедии им. Н. Н. Приорова.*, т. 4, pp. 10-18, 2010.
- [109] С. П. Миронов, А. Еськин Н. , Т. М. Андреева, «Состояние специализированной амбулаторной травматолого-ортопедической помощи пострадавшим от травм и заболеваний с патологией костно-мышечной системы / С. П. Миронов,» *Вестник травматологии и ортопедии им. Н. Н. Приорова*, т. 1, pp. 3-8, 2010.
- [110] А. В. Іпатов, О. М. Мороз, В. А. Голик, Р. Я. Перепелична , І. Я. Ханюкова, Основні показники інвалідності та діяльності медико-соціальних експертних комісій України за 2014 рік, Дніпро: Роял- Принт, 2015.
- [111] А. В. Іпатов, О. М. Мороз , Н. О. Гондуленко, сновні показники інвалідності та діяльності медико-соціальних експертних комісій України за 2016 рік: аналітико-інформаційний довідник, Дніпро: Пороги, 2017.
- [112] І. Башкін, Е. Макарова, А. Різік , М. Кавакзе, «оль і місце фізичної реабілітації у загальній системі охорони здоров'я населення,» *Теорія і методика фізичного виховання і спорту*, т. 3, pp. 25-29, 2006.
- [113] А. М. Гринштат, «Національна система фізичної реабілітації і оздоровлення (програма),» *Трибуна*, Томи 1 з 23-4, pp. 43-45, 1998.
- [114] О. Д. Дубогай , М. В. Євтушок, Довідник основних понять та термінів з оздоровчої фізичної культури, реабілітації та здоров'язберігаючих технологій, НПУ ім. М.П. Драгоманова, 2007.
- [115] И. В. Дубровский, Реабилитация в спорте, Физкультура и спорт, 1991.
- [116] О. А. Ежова, «Актуальные вопросы физической реабилитации,» *Фізичне виховання, спорт і культура здоров'я у сучасному суспільстві*, т. 3, pp. 155-158, 1999.
- [117] В. М. Мухін, Фізична реабілітація : підручник для студ. вищ. навч. закладів фізичного виховання і спорту, Олімпійська література, 2005.
- [118] Т. В. Демиденко , Ф. В. Голтдбат, Руководство для среднего медицинского персонала неврологических реабилитационных отделений, Медицина, 1977.
- [119] В. М. Козлюк, А. Д. Яремчук , И. К. Следзевская, Организация восстановительного лечения, Здоровье, 1977.

- [120] Т. Н. Кукушкина, М. Докиш Ю., А. Чистякова Н., Руководство по реабилитации больных, частично утративших трудоспособность, Медицина, 1989.
- [121] В. Н. Сокрут, В. Н. Казаков, В. Я. Уманский, «Медицинская реабилитация в Украине : современные положения и перспективы развития,» *Новости медицины и фармации в мире*, т. 10, № 138, 2003.
- [122] І. Р. Мисула, Л. О. Вакуленко, Медична, соціальна реабілітація : Навчальний посібник, Тернопіль: ТДМУ, 2005.
- [123] Ю. В. Вороненко, В. Ф. Москаленко, Соціальна медицина та організація охорони здоров'я, Тернопіль: Укрмедкнига, 2000.
- [124] В. П. Мурза, Психолого-фізична реабілітація., Київ: Олан, 2005.
- [125] К. О. Сорока, Основи теорії систем і системного аналізу, Харків: ХНАМГ, 2004.
- [126] D. Kempe, L. Schulman, O. Tamuz, «Quasi-regular sequences and optimal schedules for security games,» *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1625-1644, 2018.
- [127] J. Anderson, Automata theory with modern applications, Cambridge: Cambridge University Press, 2006.
- [128] «Finite State Machines,» 12 06 2010. [Онлайновий]. Available: <https://brilliant.org/wiki/finite-state-machines/>. [Дата звернення: 09 11 2018].
- [129] J. Hopcroft, R. Motvani, G. Ullman, Introduction to Automata Theory, Languages and Computing, Cambridge: Cambridge Press, 2000.
- [130] T. Koshy, Discrete Mathematics with Applications, Academic Press, 2004.
- [131] R. M. Keller, Classifiers, Acceptors, Transducers, and Sequencers, Harvey Mudd College, 2001.
- [132] J. Hopcroft, An $n \cdot \log n$ algorithm for minimizing states in a finite automaton, Standford: Standford University, 1971.
- [133] A. Aho, R. Sethi, J. Ullman, Compilers: Principles, Techniques, and Tools, Addison-Wesley, 1986.
- [134] R. O. Tkachenko, O. S. Kovalyshyn, «A method of making up a clinic schedule with use of finite state automaton,» *Econtechmod: An international quarterly journal on economics in technology, new technologies and modeling processes*, т. 5, № 3, pp. 131-134, 2016.
- [135] C. Schneider, Wagemann C., «Standards of Good Practice in Qualitative Comparative Analysis (QCA) and Fuzzy-Sets,» *Comparative Sociology*, т. 9, № 3, pp. 387-418, 2010.

- [136] K. Tanaka, *An Introduction to Fuzzy Logic for Practical Applications*, New York: Springer-Verlag , 1996.
- [137] J. Mendel, H. Hagsras, W. Tan, H. Ying, *Introduction To Type-2 Fuzzy Logic Control: Theory and Applications*, Wiley-IEEE Press, 2014.
- [138] O. S. Kovalyshyn, Y. I. Gabriel, «Development of a management systems model of automatic control with using fuzzy logic,» *Econtechmod: An international quarterly journal on economics in technology, new technologies and modeling processes*, т. 3, № 4, pp. 87-90, 2014.
- [139] M. Jamshidi, N. Vadiiee, T. J. Ross, *Fuzzy Logic and Control: Software and Hardware Applications*, FSS, 1994.
- [140] G. Chen, T. T. Pham, *Introduction to Fuzzy Sets, Fuzzy Logic, and Fuzzy Control Systems*, CRC Press , 2000.
- [141] L.-X. Wang, *A course in fuzzy systems and control*, Hong Kong: Hong Kong Univ. of Science and Technology, 1997.
- [142] J. Yen, Langari R., *Fuzzy Logic: Intelligence, Control, and Information*, Pearson Education, 1999.
- [143] W. Pedrycz, F. Gomide, *An Introduction to Fuzzy Sets: Analysis and Design*, MIT Press, 1998.
- [144] W. Pedrycz, *Fuzzy Modelling: Paradigms and Practice*, Springer Science & Business Media, 1996.
- [145] A. Di Nola, S. Sessa, W. S. E. Pedrycz, *Fuzzy Relation Equations and Their Applications to Knowledge Engineering*, Springer Science & Business Media, 2013.
- [146] S. Miyamoto, *Fuzzy Sets in Information Retrieval and Cluster Analysis*, Springer Science & Business Media, 2012.
- [147] T. Jaques, S.-Y. Huang, *Stochastic Versus Fuzzy Approaches to Multiobjective Mathematical Programming under Uncertainty*, Springer Science & Business Media, 2012.
- [148] R. Lowen, M. R. Roubens, *Fuzzy Logic: State of the Art*, Springer Science & Business Media,, 2012.
- [149] H. J. Lilly, *Fuzzy Control and Identification*, John Wiley & Sons, 2010.
- [150] I. Iancu, A. Mamdani, *Type Fuzzy Logic Controller. Fuzzy Logic - Controls, Concepts, Theories and Applications,*, InTech, 2012.
- [151] A. Benzaouia, A. El Hajjaji, *Advanced Takagi–Sugeno Fuzzy Systems*, Springer International Publishing, 2014.

- [152] Р. О. Ткаченко, О. С. Ковалишин, «Багатокритеріальна оцінка оптимальності розкладів медичних закладів з використанням нечіткої логіки.» *Енергетика та автоматика: журнал. НУБіП.*, т. 1, pp. 79-98, 2018.
- [153] О. Tkatchenko, R. Tkachenko, Y. Hirniak, O. Ivakhiv, P. Mushenyk, «Rule-based Fuzzy System of Improved Accuracy.» в *Proceeding of the 56-th International Colloquium: Innovation in Mechanical Engineering – Shaping the Future*, 2011.
- [154] R. O. Tkachenko, O. S. Kovalyshyn, « A method of assessing of clinic patients feedback with fuzzy logic.» . *Вісник Львівського державного університету безпеки життєдіяльності: зб. наук. праць.*, т. 15, pp. 82-88., 2017.
- [155] R. O. Tkachenko, O. S. Kovalyshyn, «Multicriteria optimization of medical institutions schedules on the basis of neuro fuzzy models and evolutionary algorithms.» *Econtechmod: An international quarterly journal on economics in technology, new technologies*, т. 15, № 3, pp. 53-60, 2017.
- [156] R. J. Bauer, *Genetic Algorithms and Investment Strategies*, Willey, 1994.
- [157] L. Chambers, *Practical Handbook of Genetic Algorithms: Complex Coding Systems, Volume III*, CRC Press , 1998.
- [158] P. J. Fleming, A. M. Zalzala, *Genetic Algorithms in Engineering Systems*, Institution of Electrical Engineers Stevenage, 1997.
- [159] D. Fogel, *Evolutionary Computation: The Fossil Record*, Wiley-IEEE Press, 1998.
- [160] О. С. Ковалишин, «Нейронечіткий генетичний алгоритм оптимізації планів відновлювальної терапії.» *Вісник Національного університету «Львівська Політехніка»*. зб. наук. праць. Серія: Інформаційні системи та мережі, т. 901, pp. 1-10, 2018.
- [161] D. Kalyanmoy, *Multi-Objective Optimization Using Evolutionary Algorithms*, John Wiley & Sons, 2001.
- [162] M. Vose, *The Simple Genetic Algorithm: Foundations and Theory*, MIT Press Cambridge, 1998.
- [163] D. Cole, *An Introduction to Genetic Algorithms for Scientists and Engineers*, World Scientific Publishing Co. Pte. , 1999.
- [164] K.-F. Man, K. S. Tang, S. Kwong, *Genetic Algorithms Concepts and Designs*, Springer International, 1999.
- [165] T. D. Gwiazda, Reference, *Genetic Algorithms*, TOMASHGWIAZDA E-BOOKS, 2006.
- [166] С. Назаров, *Архитектура и проектирование программных систем*, Научная Мысль, 2016.

- [167] L. Bass, P. Clements, *Software Architecture in Practice (3rd Edition) (SEI Series in Software Engineering)*, Addison-Wesley, 2012.
- [168] N. M. Richard N. Taylor, *Software Architecture: Foundations, Theory, and Practice*, John Wiley and Sons, 2009.
- [169] W. Tang, J.-h. Lee, B. Song, «Multi-Platform Mobile Thin Client Architecture in Cloud Environment,» *Procedia Environmental Sciences* , т. 11, pp. 499-504, 2011.
- [170] О. С. Ковалишин, « Інформаційна технологія оптимізації розкладів відновлювальної терапії,» *Науковий вісник Луцького НТУ. зб. наук. праць. Серія «Комп'ютерно-інтегровані технології: освіта, наука, виробництво»* , т. 33, pp. 85-89, 2018.
- [171] G. Shipley, D. Graham, *OpenShift for Developers. A Guide for Impatient Beginners*, O'Reilly, 2016.
- [172] K. Chodorow, *MongoDB: The Definitive Guide: Powerful and Scalable Data Storage*, O'Raily.
- [173] J. Smart, *Jenkins: The Definitive Guide: Continuous Integration for the Masses*, O'Raily, 2011.
- [174] S. Chacon, *Pro Git (Second Edition)*, Apress, 2018.
- [175] D. Heronn, *Node.js Web Development: Create real-time server-side applications with this practical, step-by-step guide*, 3rd Edition, Packt Publishing, 2016.
- [176] I. Cosmina, R. Harrop, C. Schaefer, C. Ho, *Pro Spring 5 An In-Depth Guide to the Spring Framework and Its Tools*, Apress Publishing, 2017.
- [177] Р. О. Ткаченко, І. В. Ізонін, Н. В. Веретеннікова, К. О. С. О.Л. Березко, «Спосіб зміни роздільної здатності набору сценарних зображень». Україна Патент 8, 25 Квітень 2018.
- [178] A. Gupta, *Java EE 7 Essentials: Enterprise Developer Handbook*, O'Riley, 2013.
- [179] R. Urma, M. Fusco, A. Mycroft, *Java 8 in Action*, Manning Publishing, 2014.
- [180] J. Sharma, *Getting started with Spring Framework: covers Spring 5*, O'Riley, 2017.
- [181] K. Chodorow, *Scaling MongoDB: Sharding, Cluster Setup, and Administration*, O'Riley, 2011.
- [182] «Про затвердження Типового положення про реабілітаційну установу змішаного типу для інвалідів і дітей-інвалідів: наказ Міністерства праці та соціальної політики України від 26.05.2009 № 196,» 26 05 2009. [Онлайновий]. Available: <http://zakon2.rada.gov.ua/laws/show/z0761-09>. [Дата звернення: 01 11 2018].

ДОДАТКИ

ДОДАТОК 1. Список публікацій здобувача за темою дисертації та відомості про апробацію результатів дисертації

Наукові праці, в яких опубліковані основні наукові результати дисертації:

1. Tkachenko R.O., Kovalyshyn O.S. Multicriteria optimization of medical institutions schedules on the basis of neuro fuzzy models and evolutionary algorithms. *Econtechmod: An international quarterly journal on economics in technology, new technologies and modeling processes*. 2017. Vol.6. No.3. P. 53-60. (***Index Copernicus***).
2. Tkachenko R.O., Kovalyshyn O.S. A method of assessing of clinic patients feedback with fuzzy logic. *Вісник Львівського державного університету безпеки життєдіяльності: зб. наук. праць*. 2017. № 15. С. 82-88. (***Ulrich's Periodicals Directory***)
3. Tkachenko R.O., Kovalyshyn O.S. A method of making up a clinic schedule with use of finite state automaton. *Econtechmod: An international quarterly journal on economics in technology, new technologies and modeling processes*. 2016. Vol.5. No.3. P. 131-134. (***Index Copernicus***)
4. Kovalyshyn O.S., Gabriel Yu. I. Development of a management systems model of automatic control with using fuzzy logic. *Econtechmod: An international quarterly journal on economics in technology, new technologies and modeling processes*. 2014. Vol. 3.No.4. P. 87 -90. (***Index Copernicus***)
5. Ткаченко Р.О., Ковалишин О.С. Багатокритеріальна оцінка оптимальності розкладів медичних закладів з використанням нечіткої логіки. *Енергетика та автоматика: журнал. НУБіП*. 2018. №1. С. 79-98. (***Ulrich's Periodicals Directory, Index Copernicus, Agris, Base***)

6. Ковалишин О.С. Аналіз методів оптимізації розкладів в контексті відновлювальної терапії . *Науковий вісник НЛТУ України: зб. наук. праць*. 2018. Том 28. №8. С. 136-140. (*Index Copernicus*)

7. Ковалишин О.С. Нейронечіткий генетичний алгоритм оптимізації планів відновлювальної терапії. *Вісник Національного університету «Львівська Політехніка»*. зб. наук. праць. Серія: Інформаційні системи та мережі. № 901. 2018. С.1-10.

8. Ковалишин О.С. Інформаційна технологія оптимізації розкладів відновлювальної терапії. *Науковий вісник Луцького НТУ*. зб. наук. праць. Серія «Комп'ютерно-інтегровані технології: освіта, наука, виробництво». 2018. №33. С.85-89. (*Index Copernicus, Universal Impact Factor, Open Academic Journals Index*).

Наукові праці, які засвідчують апробацію матеріалів дисертації:

9. Ковалишин О.С. Вирішення задачі багатокритеріальної оптимізації розкладів за допомогою систем нечіткої логіки. Інформаційне суспільство: технологічні, економічні та технічні аспекти становлення: тези доповідей міжнародної наукової інтернет-конференції, м. Тернопіль, 17 травня, 2017 р. Випуск 19. Частина 1. Тернопіль, 2017. С. 40–42.

10. Ковалишин О.С. Оптимізація розкладів лікувальних закладів методом генетичного алгоритму. Інтелектуальні системи прийняття рішень і проблеми обчислювального інтелекту: матеріали міжнародної наукової конференції, 22-26 травня 2017 р. ПП Вишемирський В. С., 2017. С.274-275.

11. Ковалишин О.С. Кінцевий автомат побулови розкладу медичних закладів. Інтегровані інтелектуальні робототехнічні комплекси (ПРТК-2016): матеріали дев'ятої міжнародної науково-практичної конференції, м. Київ, 17-18 травня 2016 р. Київ: НАУ, 2016. С. 22 – 24.

12. Ковалишин О.С. Задача багатокритеріальної оцінки та оптимізації планів відновлювальної терапії пацієнтів медичних закладів. Информационные

системы и технологии: материалы 4-ой международной научно-технической конференции, г. Харьков, 21-27 сентября 2015 г. Харьков, НТМТ, 2015. С. 70-72.

13. Ковалишин О.С. Прикладні аспекти використання систем нечіткого логічного висновку в задачах багатокритеріальної оптимізації. Інформаційна безпека та комп'ютерні технології: тези доповідей III Міжнародної науково-практичної конференції, м. Кропивницький, 19-20 квітня 2018 року. Кропивницький: ЦНТУ, 2018. С. 300-302.

Наукові праці, які додатково відображають наукові результати дисертації:

14. Р.О.Ткаченко, І.В. Ізонін, Н.В. Веретеннікова, О.Л. Березко, О.С. Спосіб зміни роздільної здатності набору сценарних зображень. пат. 124680 Україна: МПК G 06 K 9/36. №u201706713; заявл. 29.06.2017; опубл. 25.04.2018, Бюл. № 8. 6 с.

15. Ковалишин О.С., Ткаченко Р.О., Ізонін І.В. Комп'ютерна програма "Програмний комплекс побудови та оптимізації розкладів відновлювальної терапії": свід. про реєстр. автор. права на твір № 75244 від 06.12.2017. Зареєстр. в Міністерстві економічного розвитку і торгівлі України, заявка 10.10.2017, № 76059.

ДОДАТОК 2. Діаграми класів розробленого програмного комплексу

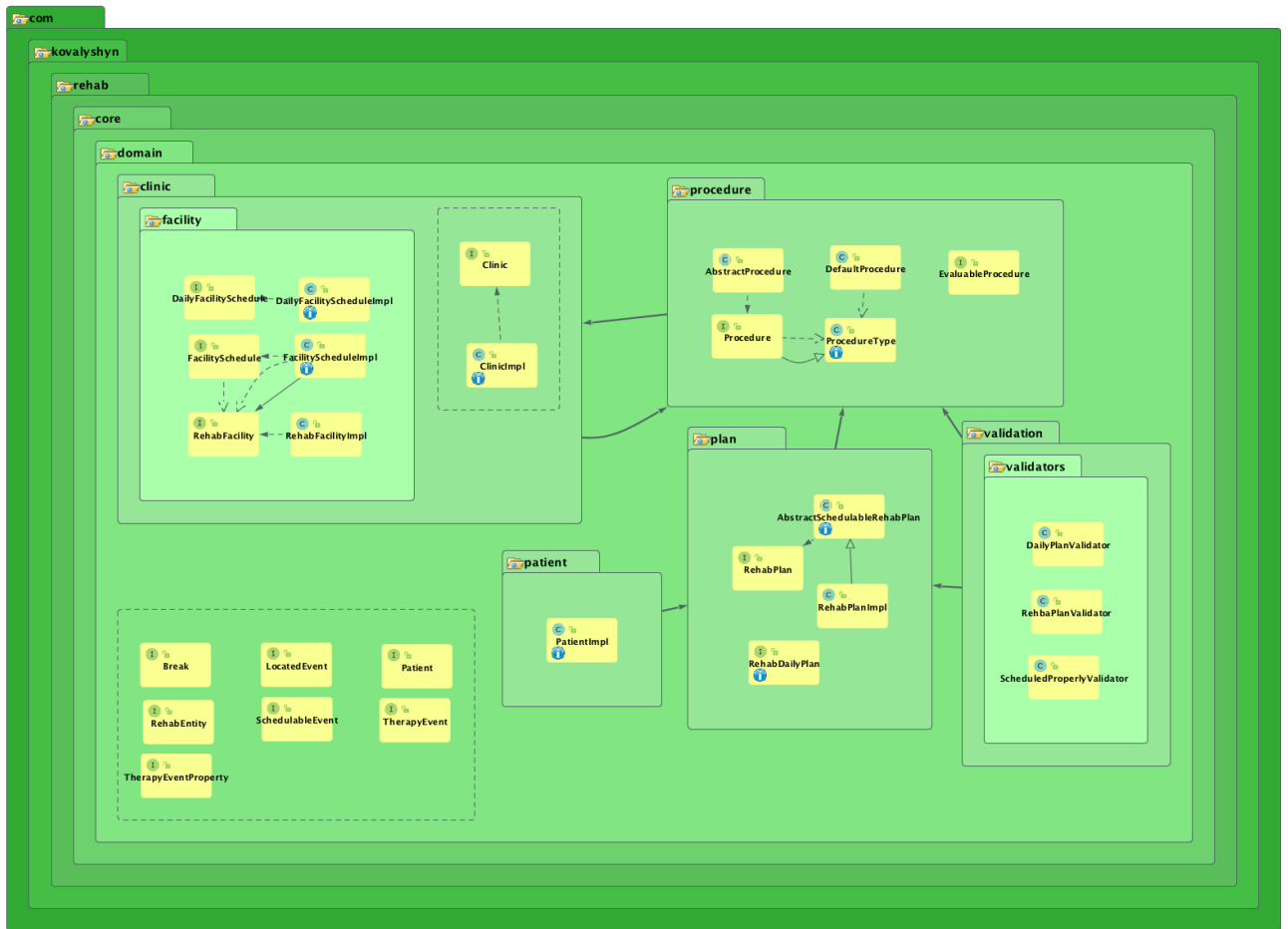


Рис 1. Діаграма класів модулю `rehab.core` програмного комплексу оптимізації планів відновлювальної терапії

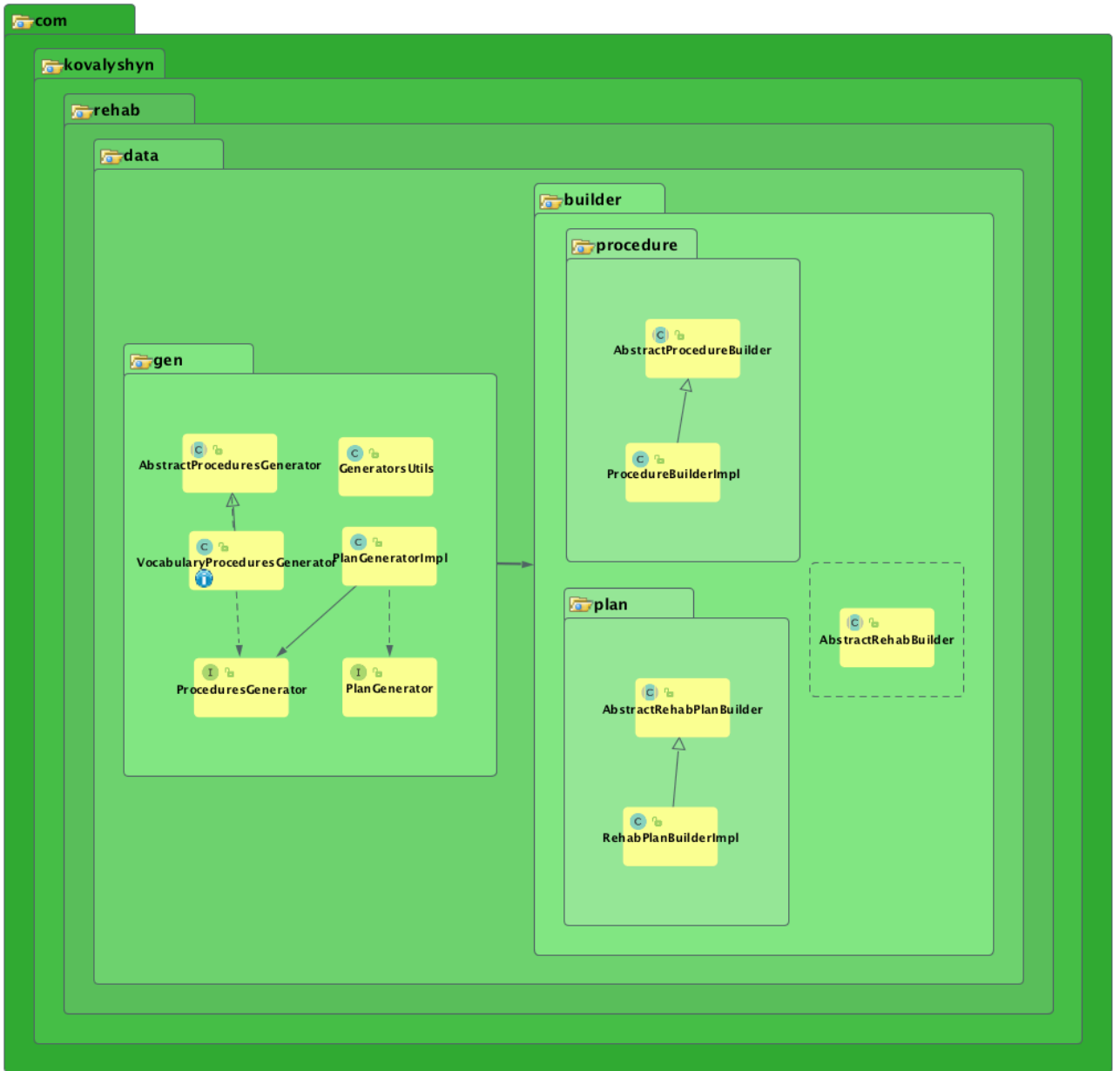


Рис 2. Діаграма класів модулю rehab.data.generators програмного комплексу оптимізації планів відновлювальної терапії

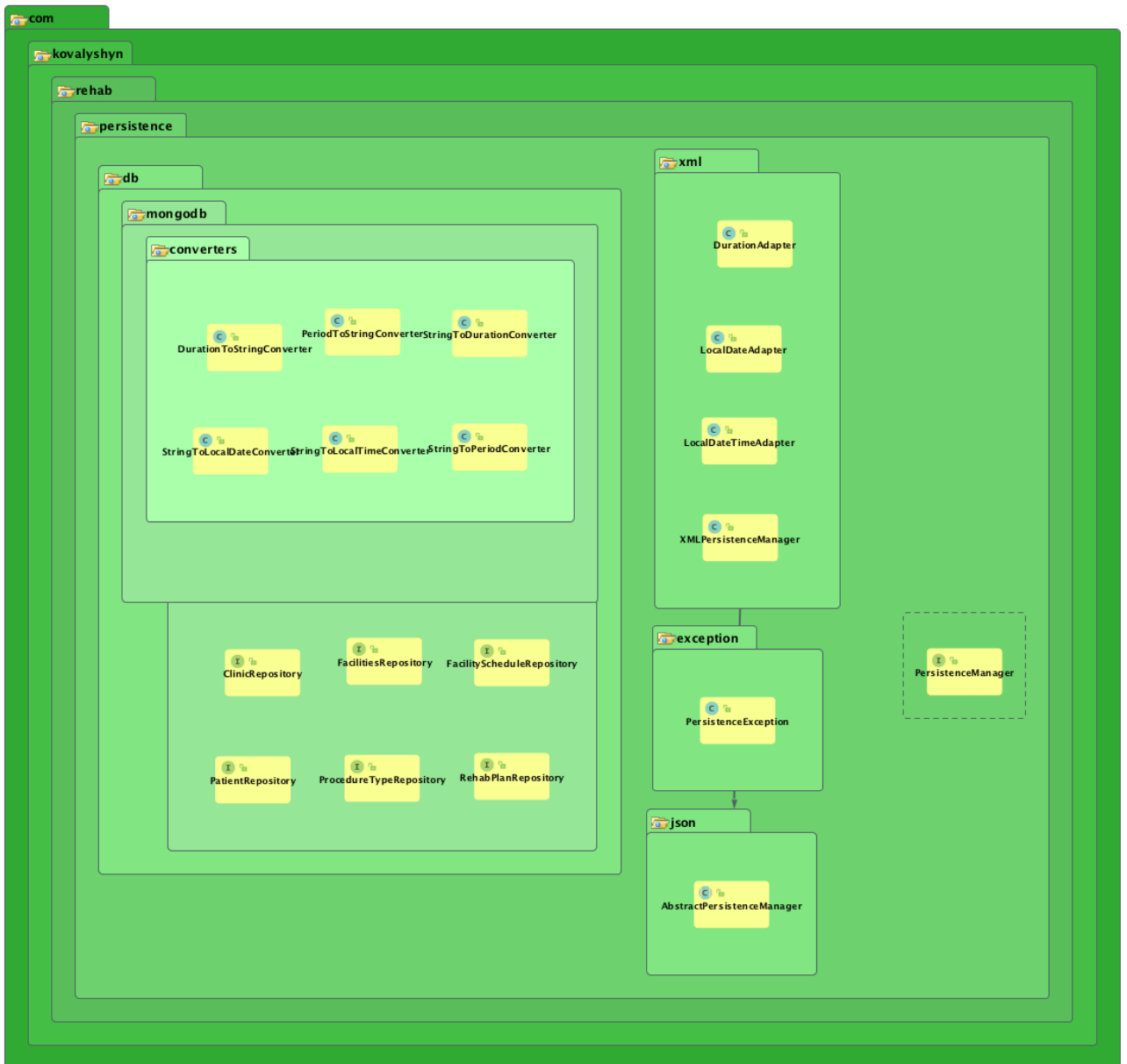


Рис 3. Діаграма класів модулю rehab.data.persistence програмного комплексу оптимізації планів відновлювальної терапії

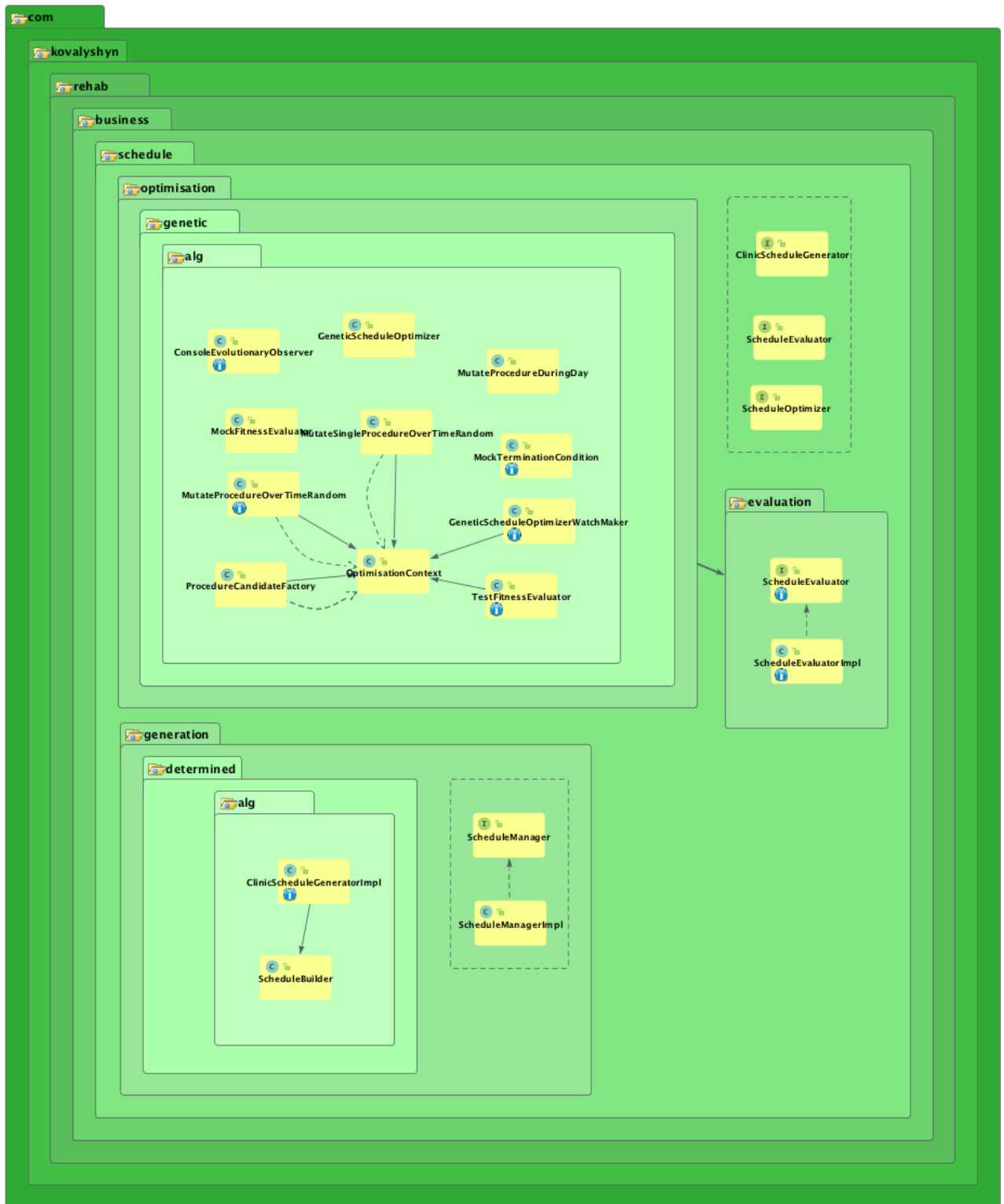


Рис 4. Діаграма класів модулю `rehab.business` програмного комплексу оптимізації планів відновлювальної терапії

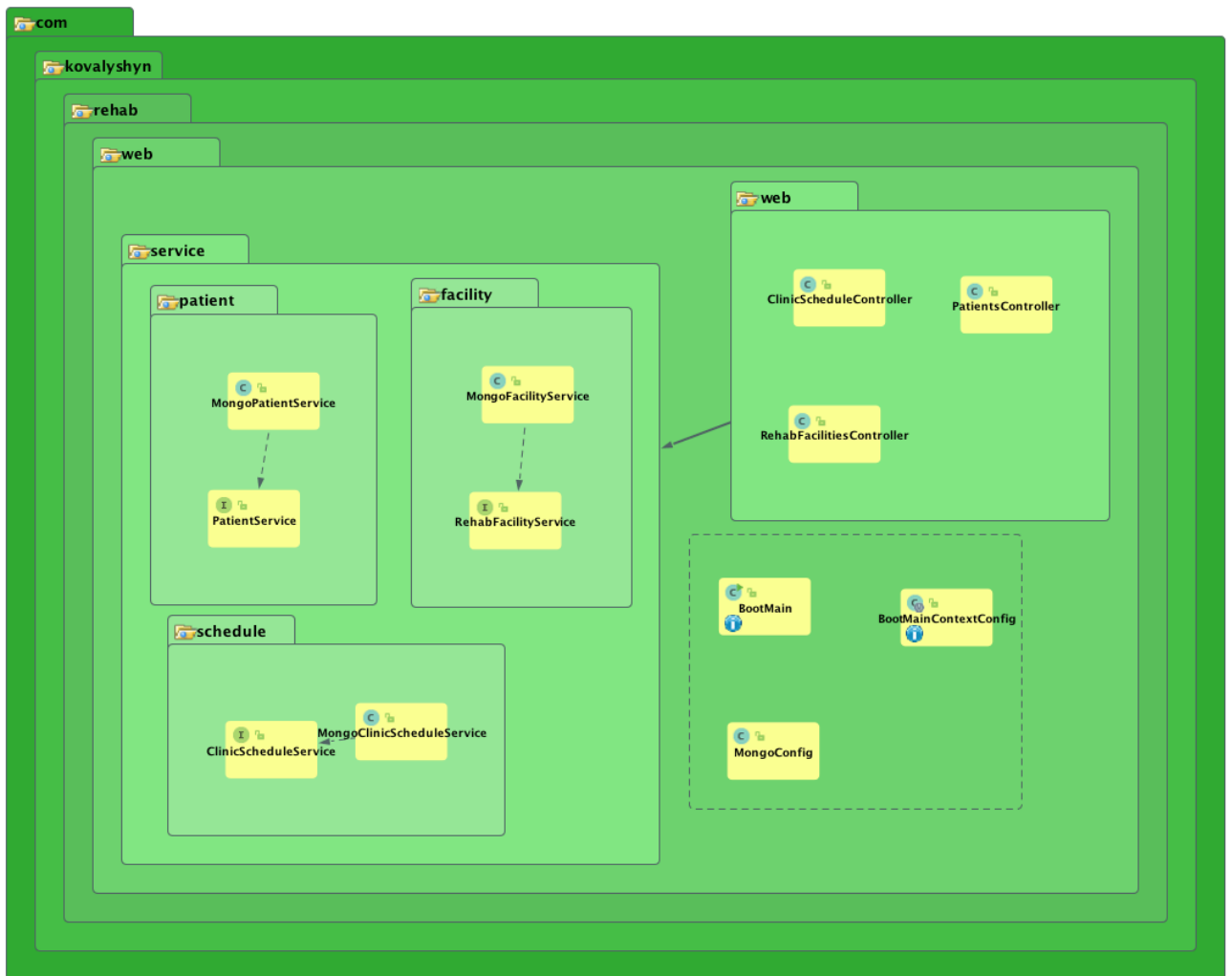


Рис 5. Діаграма класів модулю rehab.web програмного комплексу оптимізації планів відновлювальної терапії

ДОДАТОК 3. Результати апробації розробленого методу оптимізації планування клініки

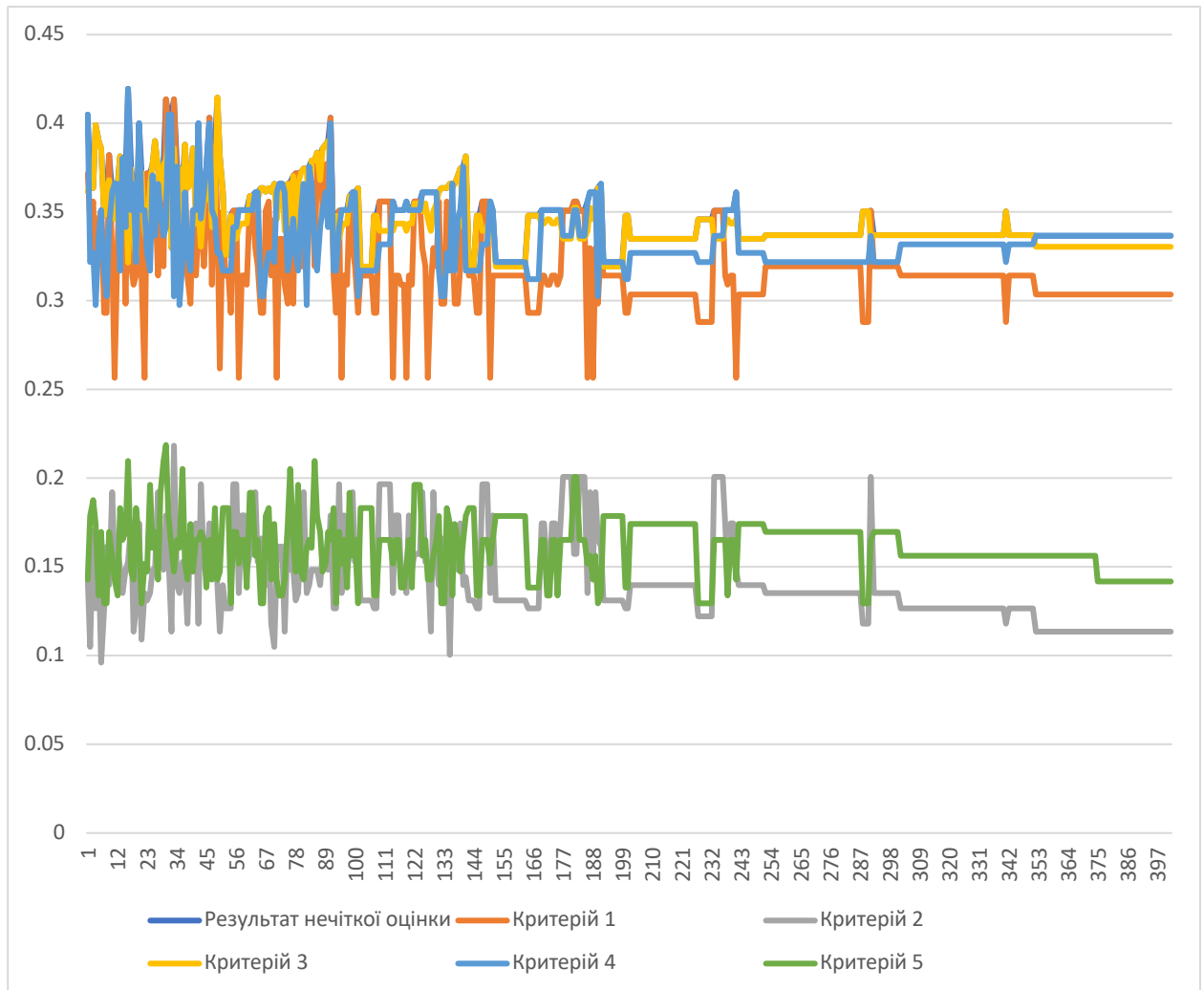


Рис 1. Послідовність функціонування методу оптимізації розкладів відновлювальної терапії (тестова вибірка №1)

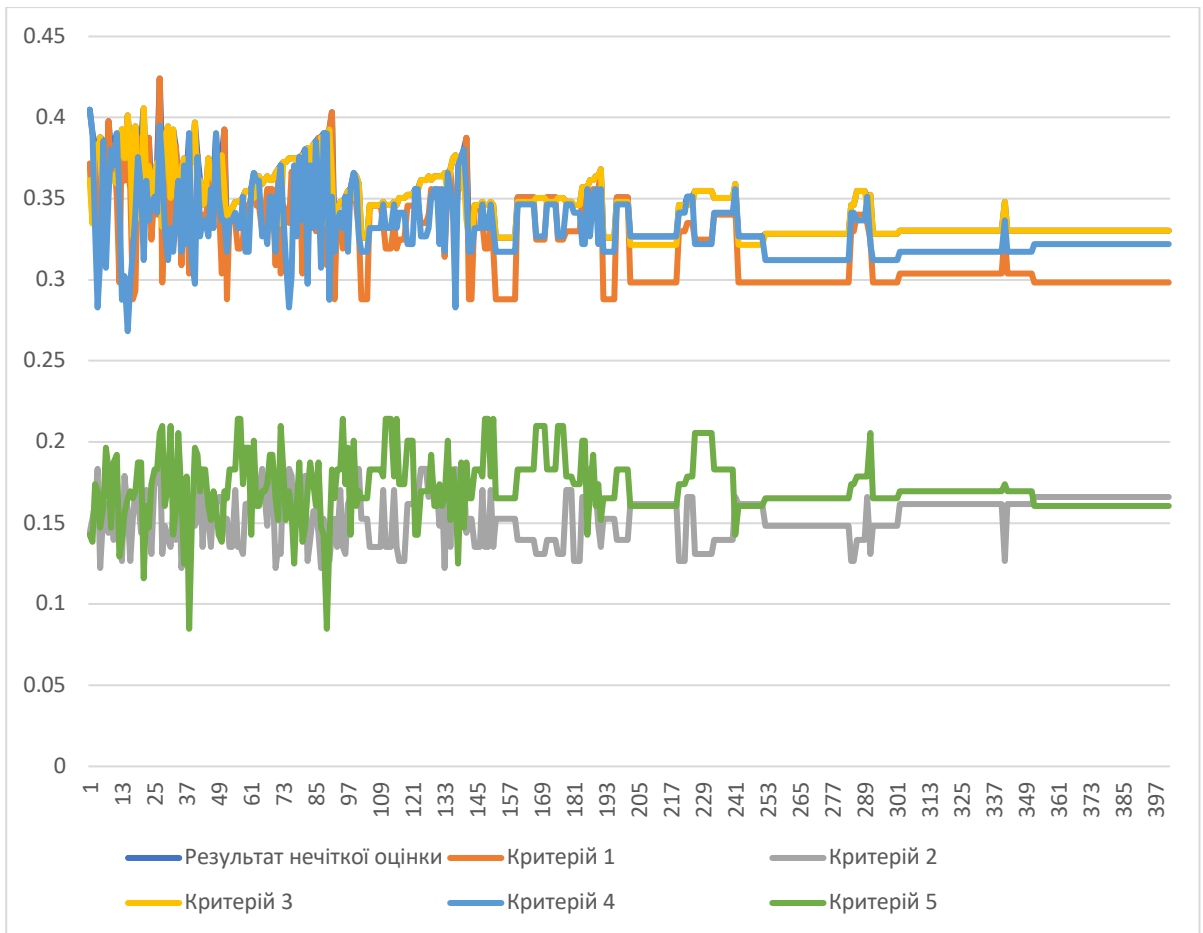


Рис 2. Послідовність функціонування методу оптимізації розкладів відновлювальної терапії (тестова вибірка №2)

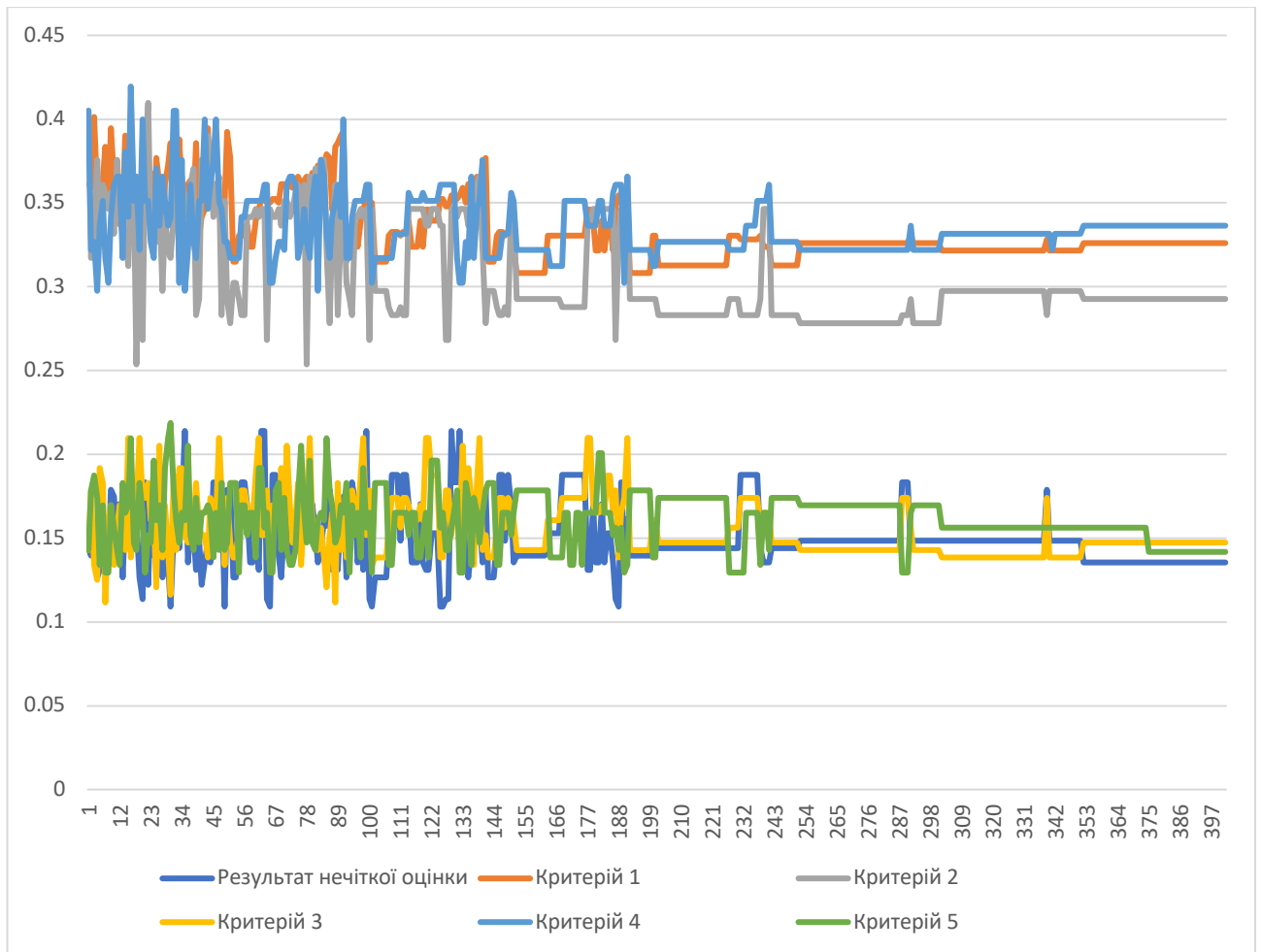


Рис 3. Послідовність функціонування методу оптимізації розкладів відновлювальної терапії (тестова вибірка №3)

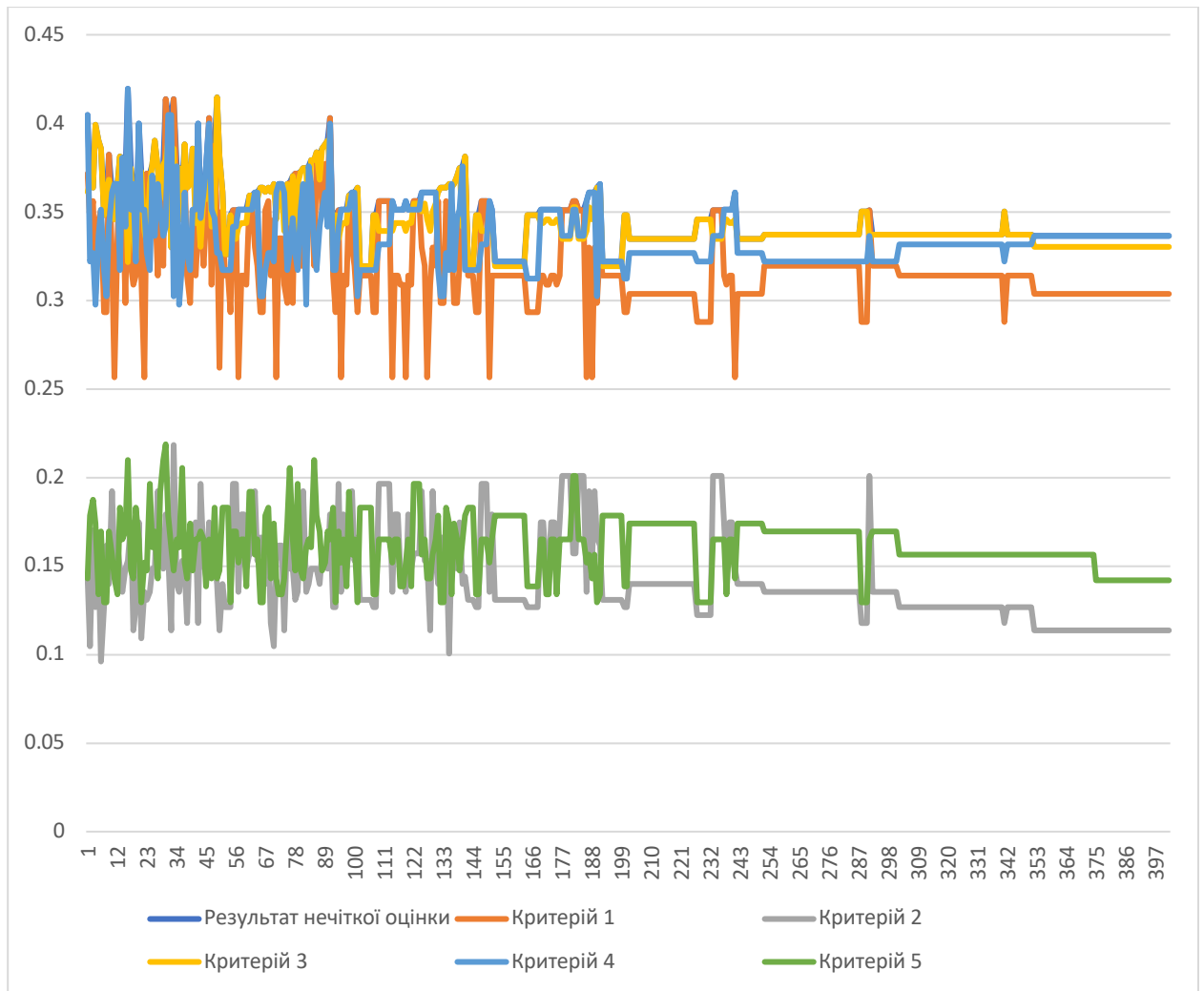


Рис 4. Послідовність функціонування методу оптимізації розкладів відновлювальної терапії (тестова вибірка №3)

ДОДАТОК 4. Акти впровадження результатів дисертаційних досліджень

Дочірнє підприємство "Санаторно-курортний комплекс "Моршинкурорт" ЗАТ "Укрпрофоздоровниця Відділення Лікувально-діагностичний центр"		
Вхід. №	427	
" 19 "	11	20 18 р

“ЗАТВЕРДЖУЮ»
Головний лікар відділення
«Лікувально-діагностичний центр»
ССК «Моршинкурорт»
Половинко Н.Ф.
2018 року

АКТ

про впровадження результатів дисертаційної роботи
на здобуття наукового ступеня кандидата технічних наук
Ковалишина Олега Степановича

Комісія у складі:

голови: головного лікаря відділення «Лікувально-діагностичний центр» ССК «Моршинкурорт» Половинко Н.Ф., членів комісії – лікарів відділення Лагоцького С.В. та Падук А.А. склали цей акт про те, що в результаті спільних наукових досліджень з розробки систем оптимізації планів відновлювальної терапії було запропоновано програмний комплекс (розробник Ковалишин О.С), який дозволяє в автоматизованому режимі складати розклади роботи медичних закладів. Він представляє собою тривірневу архітектуру, що складається з клієнтського середовища, сервера застосувань, а також сервера бази даних.

Комісія підтверджує працездатність розробленого програмного комплексу. Його використання дозволило розробити автоматизований розклад роботи відділення «Лікувально-діагностичний центр» ССК «Моршинкурорт». Цим розкладом враховано цілісну інформацію про пацієнтів, наявний штат медичних працівників, лікувальні приміщення і завантаженість розміщеного в них медичного обладнання, часові відрізки проведення процедур тощо.

Комісія звертає увагу на те, що удосконалена за допомогою запропонованого програмного комплексу процедура складання автоматизованого розкладу сприяє підвищенню ефективності лікувально-профілактичних і оздоровчо-реабілітаційних заходів медичних закладів відновлювальної терапії.

Акт складений для представлення до спеціалізованої вченої ради із захисту дисертацій і не є підставою для фінансових розрахунків.

Члени комісії:

Слагоденко Лагоцький С.В.

Падук Падук А.А.



“ЗАТВЕРДЖУЮ»
 Головний лікар Санаторій
 «Київ» плюс
 Курорт Моршин
 Суслик Є.В.
 11 листопада 2018 року



АКТ

про впровадження результатів дисертаційної роботи
 на здобуття наукового ступеня кандидата технічних наук
 Ковалишина Олега Степановича

Комісія у складі:



голови: головного лікаря ТОВ санаторій «Київ» плюс Суслика Є.В., членів комісії – лікаря відділення Левицької Т.М. та головної-медичної сестри Прядко Д.С. склали цей акт про те, що в результаті спільних наукових досліджень з розробки систем оптимізації планів відновлювальної терапії було запропоновано програмний комплекс (розробник Ковалишин О.С), який дозволяє в автоматизованому режимі складати розклади роботи медичних закладів. Він представляє собою трирівневу архітектуру, що складається з із клієнтського середовища, сервера застосувань, а також сервера бази даних.

Комісія підтверджує працездатність розробленого програмного комплексу. Його використання дозволило розробити автоматизований розклад роботи в санаторію «Київ» плюс. Цим розкладом враховано цілісну інформацію про пацієнтів, наявний штат медичних працівників, лікувальні приміщення і завантаженість розміщеного в них медичного обладнання, часові відрізки проведення процедур тощо.

Комісія звертає увагу на те, що удосконалена за допомогою запропонованого програмного комплексу процедура складання автоматизованого розкладу сприяє підвищенню ефективності лікувально-профілактичних і оздоровчо-реабілітаційних заходів медичних закладів відновлювальної терапії.

Акт складений для представлення до спеціалізованої вченої ради із захисту дисертацій і не є підставою для фінансових розрахунків.

Члени комісії:

 Левицька Т.М.
 Прядко Д.С.



«ЗАТВЕРДЖУЮ»
Головний лікар

Вендзилович Ю.М.
2018 року

АКТ

про впровадження результатів дисертаційної роботи
на здобуття наукового ступеня кандидата технічних наук
Ковалишина Олега Степановича


Комісія у складі: голови: головного лікаря Вендзиловича Ю.М., членів комісії – лікарів Сулиги І.Б. та Сандурської М.В. склала цей акт про те, що результати дисертаційного дослідження Ковалишина О.С. «Продукційна система нечіткої логіки для оптимізації планів відновлюваної терапії», а саме розроблений ним програмний комплекс для автоматизованого складання та оптимізації розкладу у медичному закладі були впроваджені у Львівському обласному державному клінічному лікувально-діагностичному ендокринологічному центрі.

Комісія підтверджує, що даний програмний комплекс є працездатним. Використовуючи його, було удосконалено розклад роботи нашого закладу. Завдяки цьому стало можливим врахувати низку обмежень та факторів: склад медичних працівників; відповідність медичних послуг послідовності плану терапії пацієнтів; наявність відповідного медичного обладнання на момент проведення лікування; забезпеченість спеціалізованими приміщеннями для проведення процедур.

Комісія звертає увагу на те, що оптимізований і запроваджений розклад Львівського обласного державного клінічного лікувально-діагностичного ендокринологічного центру сприяє отриманню максимального ефекту від раціонального використання наявних в ньому ресурсів, збільшенню кількісних та якісних показників лікування, підвищенню рівню вдовolenості працівників та пацієнтів.

Акт складений для пред'явлення до спеціалізованої вченої ради із захисту дисертацій і не є підставою для фінансових розрахунків.

Члени комісії:

 Сулига І.Б.

 Сандурська М.В.



УКРАЇНА



СВІДОЦТВО

про реєстрацію авторського права на твір

№ 75244

Комп'ютерна програма "Програмний комплекс побудови та оптимізації розкладів відновлювальної терапії"

(вид, назва твору)

Автор(и) Ковалишин Олег Степанович, Ткаченко Роман Олексійович, Ізонін Іван Вікторович

(повне ім'я, псевдонім (за наявності))

Дата реєстрації

06.12.2017



Державний секретар Міністерства економічного розвитку і торгівлі України **О. Ю. Перевезенцев**

Коваленко Д. С.

УКРАЇНА



ПАТЕНТ

НА КОРИСНУ МОДЕЛЬ

№ 124680

**СПОСІБ ЗМІНИ РОЗДІЛЬНОЇ ЗДАТНОСТІ НАБОРУ
СЦЕНАРНИХ ЗОБРАЖЕНЬ**

Видано відповідно до Закону України "Про охорону прав на винаходи і корисні моделі".

Зареєстровано в Державному реєстрі патентів України на корисні моделі **25.04.2018.**

Заступник міністра економічного розвитку і торгівлі України

М.І. Тітарчук

