

DOI 10.66032/2221-2574-2026-1-3-72-76

УДК 004.89:519.71

МОДЕЛЬ АВТОМАТИЗИРОВАННОГО ФОРМИРОВАНИЯ РАСПИСАНИЯ НА ОСНОВЕ ГЕНЕТИЧЕСКОГО АЛГОРИТМА**Платонова Алла Сергеевна**

кандидат технических наук, доцент кафедры физики и прикладной математики, Муромский институт (филиал) ФГБОУ ВО «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых».

Репина Светлана Васильевна

студент, Муромский институт (филиал) ФГБОУ ВО «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых».

Адрес: 602264, Российская Федерация, Владимирская обл., г. Муром, ул. Орловская, д. 23.

E-mail для связи: allaplatoнова@inbox.ru

Аннотация: В данной статье рассматривается один из подходов к решению проблемы автоматизации составления расписания учебных занятий. Задача автоматизации расписания является многокритериальной задачей дискретной оптимизации. Предлагается математическая модель расписания, формализующая ключевые ограничения (учебная нагрузка, доступность преподавателей и аудиторий, санитарные нормы) и целевую функцию в виде штрафной функции. Для решения задачи разработан и реализован генетический алгоритм, который включает в себя процедуры инициализации популяции, турнирного отбора, двухточечного кроссовера и мутации, и учитывает специфику предметной области. В статье приводятся результаты вычислительных экспериментов на сгенерированных данных, демонстрирующие эффективность и сходимость алгоритма, а также позволяющие получить практически применимые расписания, удовлетворяющие заданным ограничениям.

Ключевые слова: расписание занятий, генетический алгоритм, дискретная оптимизация, штрафная функция, автоматизация управления образованием.

Введение

Известно, что составление расписания учебных занятий является одной из главных и наиболее трудоемких задач управления образовательным процессом. Задача планирования занятий требует учёта большого числа разнородных факторов: это и учебные планы, и нагрузка по классам, специализация и доступность преподавателей, и аудитории, закрепленные за предметами, и санитарно-эпидемиологические требования (СЭТ) к максимальной нагрузке, распределение сложных и лёгких предметов в течение дня и недели [1, 2]. Даже незначительное изменение в одном из параметров приводит к необходимости полного пересмотра всего расписания. Это, с одной стороны, создает дополнительную нагрузку на административный персонал, а, с другой стороны, не гарантирует нахождения оптимального решения. Применение современных подходов к автоматизации сложных многокритериальных задач, основанных на методах искусственного интеллекта и машинного обучения, демонстрирует высокую эффективность в самых различных предметных областях, включая оптимизацию процессов [3].

Актуальность работы определяется потребностью в инструментах, которые будут способны автоматизировать процесс составления расписания, минимизировать человеческий фактор и обеспечить формирование сбалансированного и корректного расписания [4]. Существующие программные продукты, как правило, либо генерируют допустимые, но не оптимальные варианты, либо не обладают достаточной гибкостью для учёта всех специфических требований учебного заведения [5].

Использование генетического алгоритма (ГА) — это решение очевидное в контексте поставленной задачи. Его применение позволяет не только вести поиск решения посредством имитации эволюционных механизмов, но и достигать качественных результатов за счёт приемлемого времени вычислений. Применение ГА для составления учебных расписаний активно исследуется в отечественной науке. Например, в работе [6] разработана программа на основе ГА для типичной средней школы, учитывающая требования СЭТ и пожелания учителей. В исследовании [7] выполнен обзор методов кодирования хромосомы для задачи составления расписания, где показано, что выбор представления решения существенно влияет на эффективность ГА. Также генетические алгоритмы применяются при составлении расписания в вузах [8]. Зарубежные исследования также подтверждают, что ГА доминируют в задачах подобного типа [9, 10].

Основная цель настоящей работы заключается в формализации процесса формирования расписаний, а именно построении математической модели и создании генетического алгоритма, предназначенных для генерации расписаний учебных занятий. При этом должны быть учтены все основные требования и ограничения, характерные для конкретного образовательного учреждения, а также обеспечено высокое качество итоговых решений.

1. Постановка задачи и математическая модель

Процесс составления расписания учебных занятий может быть представлен в формальных терминах как задача дискретной оптимизации. В качестве объекта исследования выступает общеобразовательная школа,

на примере которой рассматривается соответствующая постановка задачи. Предположим, что определены следующие множества объектов и параметров, описывающих структуру учебного процесса:

- $C = \{c_1, c_2, \dots, c_n\}$ — множество учебных классов;
- $T = \{t_1, t_2, \dots, t_k\}$ — множество учителей;
- $S = \{s_1, s_2, \dots, s_m\}$ — множество учебных предметов;
- $R = \{r_1, r_2, \dots, r_l\}$ — множество учебных кабинетов;
- $D = \{d_1, d_2, \dots, d_5\}$ — множество рабочих дней недели;
- $H = \{h_1, h_2, \dots, h_7\}$ — множество возможных временных слотов для занятий.

Для каждого класса c_i задан учебный план $P(c_i, s)$, в котором зафиксировано недельное количество академических часов, отводимых на изучение предмета s_i . В свою очередь, для каждого учителя t_i определено множество дисциплин, которые он уполномочен преподавать, а также установлена его временная доступность $A_t(d, h)$ в конкретный день и час. Помимо этого, учитель может исполнять функции классного руководителя, закреплённого за одним из классов. Данное обстоятельство приводит к возникновению дополнительного требования: первый урок в таком классе («Разговоры о важном») должен проводиться именно этим педагогом.

Расписание, обозначаемое как *Schedule*, представляет собой отображение, которое каждому классу c_i , дню недели d_i и временному слоту h_i ставит в соответствие либо кортеж, содержащий три элемента — предмет s_i , учителя t_i и кабинет r_i , — либо специальное значение «окно», означающее отсутствие занятия в данный момент времени.

Целевая функция в данной задаче ориентирована на минимизацию нарушений всех типов ограничений, налагаемых на расписание. С этой целью вводится штрафная функция, подлежащая минимизации и имеющая следующий вид:

$$P(\text{Schedule}) = w_1 \cdot P_{\text{count}} + w_2 \cdot P_{\text{hard}} + w_3 \cdot P_{\text{easy}} + w_4 \cdot P_{\text{first}} + w_5 \cdot P_{\text{overload}} + w_6 \cdot P_{\text{conflict}} + w_7 \cdot P_{\text{subgroups}}. \quad (1)$$

В состав штрафной функции (1) входят следующие компоненты:

- P_{count} , назначается за превышение предельно допустимого числа уроков в день для каждого класса — в соответствии с санитарно-эпидемиологическими требованиями;
- P_{hard} , фиксирует нарушения, связанные с постановкой так называемых «сложных» предметов (к ним относятся математика, физика и химия) на поздние временные слоты с 4-го по 7-й;
- P_{easy} , напротив, взимается за размещение «лёгких» дисциплин (музыка, изобразительное искусство, технология) в начале дня — на слотах 1–2;
- P_{first} , отражает отсутствие занятия «Разговоры о важном» первым уроком в понедельник у того учителя, который закреплён за данным классом в качестве классного руководителя;

- P_{overload} штрафует неравномерность распределения нагрузки на педагогов, ведущих занятия одновременно в разных параллелях;

- P_{conflict} , назначается за так называемые «накладки» — ситуации, когда один и тот же учитель или один кабинет оказываются задействованными в один и тот же временной интервал;

- $P_{\text{subgroups}}$, фиксирует ошибки, возникающие при разделении классов на подгруппы (например, для уроков иностранного языка).

Каждый из перечисленных компонентов представляет собой сумму нарушений соответствующего типа, взятых с весовым коэффициентом w_i . Все весовые коэффициенты w_i в данной работе приняты равными единице, что означает равнозначность всех типов нарушений. В зависимости от требований конкретного учебного заведения их значения могут быть скорректированы. В качестве примера рассмотрим штраф за превышение допустимой нагрузки для класса. Его вычисление производится по следующей формуле:

$$P_{\text{count}} = \sum_{c \in C} \sum_{d \in D} \max(0, L_{c,d} - M_{c,d}), \quad (2)$$

где $L_{c,d}$ — фактическое количество уроков у класса в день; $M_{c,d}$ — максимально допустимое количество уроков для этого класса и дня.

Таким образом, наша задача оптимизации сводится к поиску такого расписания *Schedule*, которое доставляет минимум функции (1). Полученное решение должно сводить к минимуму количество конфликтов и нарушений, стремясь при этом к нулевому значению штрафа.

2. Генетический алгоритм для формирования расписания

Для решения сформулированной оптимизационной задачи был разработан генетический алгоритм. Опишем структуру алгоритма:

1. Кодирование решения и инициализация. Особь (хромосома) представляет собой закодированное расписание для всех классов. Структура данных — это многомерный массив, где для каждого класса, дня и слота зафиксирован идентификатор предмета, учителя и кабинета. Начальная популяция размером $N = 50 \dots 100$ создаётся с использованием жадных эвристик (то есть алгоритмов, которые на каждом шаге выбирают локально наилучшее решение), гарантирующих базовую допустимость решений. Назначение фиксированных занятий (например, «Разговоры о важном») и установка ограничений на максимальное количество уроков в день для каждого класса.

2. Функция приспособленности. Для оценки качества особей используется фитнес-функция $F(\text{Schedule}) = -P(\text{Schedule})$. Таким образом, чем меньше штраф, тем выше приспособленность и тем больше шансов у особи быть выбранной для скрещивания. Для расчета фитнес-функции используется формула (1).

3. Селекция. Для формирования родительского пула применяется турнирный подход с размером турнира, равным трём. Суть его сводится к следующему: из текущей популяции случайным образом извлекаются три особи, после чего среди них отбирается наилучшая — по максимальному значению фитнес-функции F . Именно эта особь и становится первым родителем. Затем вся процедура повторяется для выбора второго родителя. Подобный механизм, как показывает практика, обеспечивает необходимое селективное давление, но при этом не приводит к обеднению генофонда, поскольку сохраняется определённый уровень разнообразия внутри популяции.

4. Кроссовер. Генерация потомков осуществляется посредством двухточечного кроссовера, модифицированного с учётом специфики структуры расписания. Родительские хромосомы обмениваются фрагментами генетической информации — например, блоками, соответствующими расписанию для целой параллели классов или для отдельного дня недели. Вероятность совершения кроссовера задаётся параметром p_{cross} , значение которого находится в интервале $0,8...0,9$. Потомок, появившийся в результате скрещивания, наследует черты обоих родителей. При этом следует подчеркнуть: гены, закреплённые жесткими ограничениями (то есть занятия, привязанные к конкретному времени), защищены от модификации в ходе данной операции.

5. Мутация. Для поддержания генетической изменчивости и предотвращения преждевременной сходимости алгоритма к локальным оптимумам вводится оператор мутации. Он применяется с относительно невысокой вероятностью p_{mut} , лежащей в диапазоне $0,01...0,05$. Суть мутации состоит во внесении случайных изменений в отдельные, нефиксированные гены. Это может быть, к примеру, замена кабинета для некоторого урока или перестановка двух занятий местами в пределах одного дня. Все вносимые изменения подвергаются обязательной проверке на допустимость, чтобы не нарушить базовые ограничения задачи.

6. Элитизм. Для гарантии того, что лучшее найденное решение не будет потеряно, используется стратегия элитизма. Лучшие E особей текущего поколения (обычно 1–2) гарантированно переходят в новое поколение без изменений.

Процесс повторяется в течение фиксированного числа поколений G (например, до 1000) или до тех пор, пока не будет достигнуто расписание с нулевым значением штрафа ($P = 0$).

3. Результаты вычислительных экспериментов

Предложенный алгоритм составления расписания был реализован на языке программирования Python с использованием библиотек PyQt5 (для создания графического интерфейса) и openpyxl (для экспорта результатов в формат Excel).

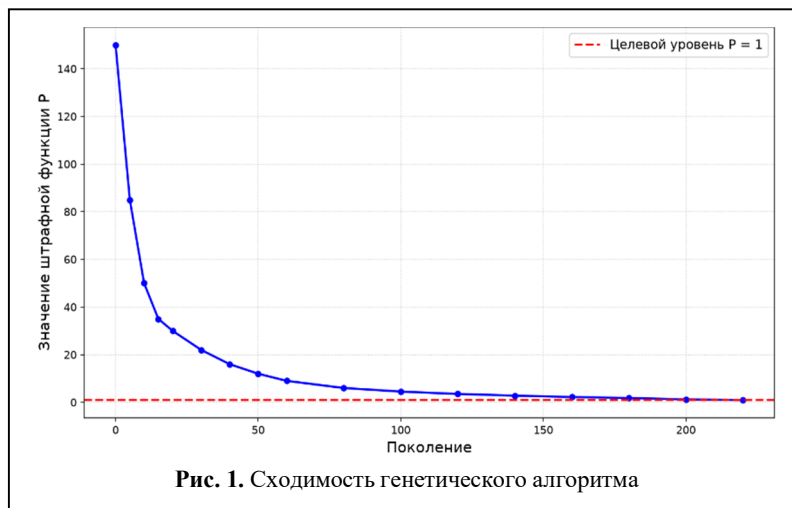


Рис. 1. Сходимость генетического алгоритма

Программная часть системы включает модули для ввода и редактирования исходных данных (классы, учителя, предметы, кабинеты, учебный план), позволяет настраивать параметры генетического алгоритма, визуализировать и экспортировать созданное расписание.

Для проверки эффективности работы алгоритма составления расписания проводились вычислительные эксперименты. Для этого были сгенерированы тестовые наборы данных, моделирующие структуру учебного процесса типовой средней общеобразовательной школы. Исходные параметры задавались в диапазонах, характерных для школ. Это, в итоге, позволило оценить работоспособность и сходимость алгоритма в условиях, приближенных к реальным.

Тестовый сценарий включал 11 классов (с 1-го по 11-й), 35 учителей с различной предметной специализацией, 18 учебных дисциплин с различным количеством часов в неделю согласно учебному плану, 20 учебных кабинетов с закреплённой специализацией. Базовые параметры генетического алгоритма были установлены следующим образом: размер популяции $N = 50$; количество поколений $G = 200$; вероятность кроссовера $p_{cross} = 0,85$; вероятность мутации $p_{mut} = 0,03$.

В ходе серии экспериментов были зафиксированы следующие ключевые показатели.

1. Сходимость алгоритма.

График на рис. 1 показывает динамику штрафа для лучшей особи. Видно, что вначале (поколения 1–20) штраф падает резко. Со значения 150 до значения 30. Это связано с устранением грубых конфликтов. Затем (поколения 21–180) падение становится плавным. И штраф достигает 2–3. В среднем, для нахождения $P \leq 1$ требуется 150–200 поколений. На ПК с Intel Core i5 это занимает 1–2 минуты.

2. Влияние параметров алгоритма на качество решения. Варьировались ключевые параметры генетического алгоритма с целью определения их оптимальных значений. В результате получены следующие закономерности. При $N < 30$ алгоритм сходится преждевременно. И финальный штраф > 50 . При $N > 100$ и

$G > 500$ качество улучшается незначительно, а время счёта растёт сильно. Оптимальным оказался диапазон $N = 50 \dots 70$ и $G = 200 \dots 300$. Для p_{mut} и p_{cross} оптимальны значения 0,02...0,05 и 0,8...0,9 соответственно.

3. Качество сгенерированных расписаний. Для всех классов тестового набора данных были сгенерированы расписания. В них не было обнаружено накладок по учителям и кабинетам, требования СЭТ по максимальной нагрузке для каждой возрастной группы были выполнены. Кроме того, получилось минимизировать неравномерность распределения учебной нагрузки по дням недели и соблюсти требования к постановке сложных и лёгких предметов.

Заключение

В результате проведённой работы созданы математическая модель и генетический алгоритм, которые позволяют автоматизировать процесс составления расписания учебных занятий. Новизна подхода заключается в комплексной штрафной функции, включающей санитарные нормы к условиям обучения, специфические требования школы (закрепление кабинетов, предметы в подгруппах, нагрузка на учителей) и адаптированные операторы генетического алгоритма. Это обеспечивает получение качественных решений задачи составления расписаний.

Эффективность предложенного алгоритма подтверждена вычислительными экспериментами. Показана его высокая сходимость и способность находить расписания с минимальным количеством нарушений. В качестве дальнейших направлений исследования планируется адаптация алгоритма для работы в режиме реального времени (перестройка расписания при заменах) и интеграция с системами электронного документооборота школы.

Литература

1. Ceschia S., Di Gaspero L., Schaerf A. Educational Timetabling: Problems, Benchmarks, and State-of-the-Art Results. *European Journal of Operational Research*. 2023. Vol. 308, No. 1. Pp. 1–18. DOI: 10.1016/j.ejor.2022.07.011
2. Tan J.S., Goh S.L., Kendall G., Sabar N.R. A survey of the state-of-the-art of optimisation methodologies in

school timetabling problems. *Expert Systems with Applications*. 2021. Vol. 165. Art. no. 113943. DOI: 10.1016/j.eswa.2020.113943

3. Платонова А.С., Рыжкова М.Н. Применение машинного обучения для сегментации пользователей в маркетинговых исследованиях. *Радиотехнические и телекоммуникационные системы*. 2025. № 3. С. 47–53. DOI: 10.24412/2221-2574-2025-3-47-53

4. Низамова Г.Ф. Математическое и программное обеспечение составления расписания учебных занятий на основе агрегативных генетических алгоритмов: дис. ... канд. техн. наук: 05.13.11. Уфа, 2006. 132 с.

5. Яндыбаева Н.В. Генетический алгоритм в задаче оптимизации учебного расписания вуза. *Современные наукоемкие технологии*. 2009. № 11. С. 97–98.

6. Лисицин Л.А., Лисицин А.Л., Лисицин А.Л. Когнитивный подход к разработке расписаний занятий на основе модификации генетического алгоритма. *Известия Юго-Западного государственного университета*. 2024. Т. 28, № 1. С. 148–162. DOI: 10.21869/2223-1560-2024-28-1-148-162

7. Смирнов А.С. Сравнительный анализ методов представления хромосомы в генетических алгоритмах для задачи составления учебного расписания. *Вестник молодежной науки*. 2026. № 5 (52). С. 1. DOI: 10.46845/2541-8254-2025-5(52)-20-20

8. Жабко Г.П. Автоматизация составления расписания в вузе с использованием SMT-решателей и генетического алгоритма. Системный анализ в проектировании и управлении: сборник научных трудов XXVIII Международной научно-практической конференции, 27–29 июня 2024 года: [в 2 частях]. Ч. 1. С. 478–487. DOI: 10.18720/SPBPU/2/id24-486

9. Budiarto A.S., Satvika Iswari N.M., Dharma E.M. Application of Genetic Algorithm for School Timetable Scheduling. 2024 IEEE International Conference on Intelligent Technology and Its Applications (ICITIA). Denpasar, Bali, Indonesia, 2024. Pp. 1–6. DOI: 10.1109/ICITIEE63403.2024.10920446

10. Criollo J., Riofrio D., Grijalva F., Vega-Sánchez J., Flores-Moyano R., Baldeon-Calisto M. School Timetable Optimization Model Using Genetic Algorithms Considering Hard and Soft Constraints. *ETCM 2025 – 9th Ecuador Technical Chapters Meeting*. IEEE, 2025. DOI: 10.1109/ETCM67548.2025.11304269

Поступила 12 апреля 2026 г.

English

MODEL OF AUTOMATED SCHEDULE GENERATION BASED ON A GENETIC ALGORITHM

Alla Sergeevna Platonova — PhD. in Engineering, Associate Professor, Murom Institute (branch) of the "Vladimir State University named after Alexander and Nikolay Stoletovs".

Svetlana Vasilyevna Repina — Student, Murom Institute (branch) of the "Vladimir State University named after Alexander and Nikolay Stoletovs".

Address: 602264, Russian Federation, Vladimir region, Murom, Orlovskaya str., 23.

E-mail: allaplatonova@inbox.ru

Abstract: The article considers the problem of automating class scheduling in a general education school. This problem belongs to the class of multi-criteria discrete optimization problems and is characterized by a large number of constraints: sanitary and epidemiological norms (SanPiN), availability of teachers and classrooms, curricula, distribution of complex and easy subjects during the day and week, as well as specific school requirements, such as conducting "Conversations about Important Things" as the first lesson with the class teacher and dividing classes into subgroups for

a foreign language and computer science. The relevance of the work is due to the fact that manual planning in modern schools is extremely labor-intensive and does not guarantee optimality, while existing software products either generate feasible but not the best options or are insufficiently flexible to take into account all the requirements of a particular educational institution. A genetic algorithm simulating natural selection is proposed as a solution method. The choice is due to its effectiveness for problems with a large search space and many constraints. A mathematical model has been developed that formalizes all types of constraints as a penalty function. For each type of violation, a separate penalty component is introduced: exceeding the permissible number of lessons, incorrect placement of complex and easy subjects, absence of "Conversations about Important Things" as the first lesson, uneven teacher workload, time overlaps, and errors in dividing into subgroups. All weight coefficients are taken equal to one, but the model allows them to be adjusted to the requirements of a particular school. The genetic algorithm includes population initialization based on greedy heuristics, tournament selection, two-point crossover with protection of fixed genes, and mutation of non-fixed elements. An elitism strategy is used to preserve the best solutions. The algorithm is implemented in Python with a PyQt5 graphical interface and an Excel export module. Computational experiments were conducted on test data simulating a typical school with 11 classes, 35 teachers, 18 subjects, and 20 classrooms. The experiments showed that with optimal parameters (population size 50–70, number of generations 200–300, crossover probability 0.85, mutation probability 0.03), the algorithm converges to a solution with minimal penalty in 1–2 minutes on a standard PC. For all classes, schedules were obtained without overlaps in teachers and classrooms, with compliance with sanitary norms and an even distribution of workload across the days of the week. The novelty of the work lies in the comprehensive penalty function that combines sanitary norms with the specific requirements of the school, and in the adapted operators of the genetic algorithm. The practical significance consists in the possibility of using the developed system to automate the work of school administration, reduce time costs for planning, and improve the quality of the educational process. **Keywords:** isochrony, distributed parallel computing, network structure information system.

References

1. *Ceschia S., Di Gaspero L., Schaerf A.* Educational Timetabling: Problems, Benchmarks, and State-of-the-Art Results. *European Journal of Operational Research*. 2023. Vol. 308, No. 1. Pp. 1–18. DOI: 10.1016/j.ejor.2022.07.011
2. *Tan J.S., Goh S.L., Kendall G., Sabar N.R.* A survey of the state-of-the-art of optimization methodologies in school timetabling problems. *Expert Systems with Applications*. 2021. Vol. 165. Art. no. 113943. DOI: 10.1016/j.eswa.2020.113943
3. *Platonova A.S., Ryzhkova M.N.* Machine Learning Application for User Segmentation in Marketing Research. *Radio Engineering and Telecommunication Systems*. 2025. No. 3. Pp. 47–53. DOI: 10.24412/2221-2574-2025-3-47-53
4. *Nizamova G.F.* Mathematical and Software Support for Creating a Class Schedule Based on Aggregate Genetic Algorithms: Cand. Sci. (Eng.) Dissertation: 05.13.11. Ufa, 2006. 132 p.
5. *Yandybaeva N.V.* Genetic Algorithm in the Problem of Optimizing the University Curriculum. *Modern Science-Intensive Technologies*. 2009. No. 11. Pp. 97–98.
6. *Lisitsyn L.A., Lisitsyn A.L., Lisitsyn A.L.* A Cognitive Approach to Developing Class Schedules Based on a Modified Genetic Algorithm. *Bulletin of the South-West State University*. 2024. Vol. 28, No. 1. Pp. 148–162. DOI: 10.21869/2223-1560-2024-28-1-148-162
7. *Smirnov A.S.* Comparative Analysis of Chromosome Representation Methods in Genetic Algorithms for the Problem of Creating a Class Schedule. *Bulletin of Youth Science*. 2026. No. 5 (52). Pp. 1. DOI: 10.46845/2541-8254-2025-5(52)-20-20
8. *Zhabko G.P.* Automation of School Timetable Generation in a University Using SMT Solvers and a Genetic Algorithm. *Systems Analysis in Design and Management: Proceedings of the XXVIII International Scientific and Practical Conference, June 27–29, 2024: [in 2 parts]. Part 1*. Pp. 478–487. DOI: 10.18720/SPBPU/2/id24-486
9. *Budiarto A.S., Satvika Iswari N.M., Dharma E.M.* Application of Genetic Algorithm for School Timetable Scheduling. 2024 IEEE International Conference on Intelligent Technology and Its Applications (ICITIA). Denpasar, Bali, Indonesia, 2024. Pp. 1–6. DOI: 10.1109/ICITIE63403.2024.10920446
10. *Criollo J., Riofrío D., Grijalva F., Vega-Sánchez J., Flores-Moyano R., Baldeon-Calisto M.* School Timetable Optimization Model Using Genetic Algorithms Considering Hard and Soft Constraints. *ETCM 2025 – 9th Ecuador Technical Chapters Meeting*. IEEE, 2025. DOI:10.1109/ETCM67548.2025.11304269