

УДК 004.021:519.85:378.016:378.044

## ИСПОЛЬЗОВАНИЕ ГЕНЕТИЧЕСКОГО АЛГОРИТМА ДЛЯ РЕШЕНИЯ ЗАДАЧИ РАСПРЕДЕЛЕНИЯ УЧЕБНОЙ НАГРУЗКИ

Е.А. Корчевская\*, С.А. Ермоченко\*, Т.В. Никонова\*\*, Л.В. Маркова\*\*\*, Ю.А. Шпакова\*

\*Учреждение образования «Витебский государственный университет имени П.М. Машерова»

\*\*Учреждение образования «Витебский государственный технологический университет»

\*\*\*Институт тепло- и массообмена имени А.В. Лыкова НАН Беларуси

*В настоящей работе для составления учебной нагрузки преподавателей кафедры предлагается композиционный генетический алгоритм, который базируется на объектно-ориентированном представлении исходной информации, адаптированных к ней генетических операциях и структурированных объектах генетической оптимизации.*

*Цель работы – адаптация генетического алгоритма для решения задачи распределения учебной нагрузки на кафедре.*

**Материал и методы.** В качестве исходных использовались данные учебной нагрузки кафедры прикладного и системного программирования. Основным методом исследования является генетический алгоритм.

**Результаты и их обсуждение.** Исходной информацией при распределении нагрузки выступают множества дисциплин кафедры, преподавателей, различных видов занятий. Между данными множествами существуют зависимости, которые образованы благодаря организационному процессу работы кафедры. Разработанный генетический алгоритм состоит из нескольких этапов: формирования первого поколения случайным образом; селекции, позволяющей отобрать наиболее приспособленные особи; генерации нового поколения путем применения скрещивания более приспособленных; мутации.

**Заключение.** В результате работы предложена методика формирования нагрузки профессорско-преподавательского состава с помощью генетического алгоритма.

**Ключевые слова:** генетический алгоритм, распределение учебной нагрузки, оператор скрещивания, оператор мутации.

## USING A GENETIC ALGORITHM FOR SOLVING THE PROBLEM OF DISTRIBUTION OF ACADEMIC WORKLOAD

E.A. Korchevskaya\*, S.A. Yermochenko\*, T.V. Nikonova\*\*, L.V. Markava\*\*\*, Y.A. Shpakova\*

\*Education Establishment "Vitebsk State P.M. Masherov University"

\*\*Education Establishment "Vitebsk State Technological University"

\*\*\*A.V. Luikov Heat and Mass Transfer Institute of the National Academy of Sciences of Belarus

*In this paper, for compiling the academic workload of teachers of the department, a compositional genetic algorithm is proposed, based on the structuring of the initial information, genetic operations adapted to it, and an object-oriented representation of genetic optimization objects.*

*The purpose of the work is the adaptation of the genetic algorithm for solving the problem of distributing the teaching load at the department.*

**Material and methods.** The data of the academic workload of the Applied and System Programming Department were used as initial data. The main research method is a genetic algorithm.

**Findings and their discussion.** As initial information while distributing the workload is: a set of disciplines of the Department, the teachers of the Department, types of classes. Among these sets there are links arising from the organizational process of the Department work. The developed algorithm consists of several stages: "the scatter" – the stage at which the first generation is randomly compiled; the selection, which makes it possible to select the most appropriate individuals; the formation of a new generation that allows you to generate the following individuals by crossing the more fit; mutation.

**Conclusion.** As a result of the work, a method for forming the workload of the teaching staff using a genetic algorithm is proposed.

**Key words:** genetic algorithm, distribution of the academic load, crossover operator, mutation operator.

Основополагающей задачей при организации учебного процесса на кафедрах в учреждении высшего образования является задача автоматизированного формирования учебной нагрузки. Грамотное распределение учебных дисциплин обеспечивает равномерную загрузку профессорско-преподавательского состава, который отвечает за подготовку молодых специалистов [1].

Многообразие форм, направлений, различных категорий и запросов к ним приводит к большому числу версий учебной нагрузки, из которых необходимо выбрать оптимальную. Данный процесс является многокритериальной, трудоемкой задачей, с учетом множества дисциплин, преподаваемых на кафедре, а также динамичного и многочисленного состава кафедры. Сложности формирования нагрузки связаны с тем, что задача составления оптимальной схемы распределения обусловлена следующими особенностями: трудоемкостью построения математической модели благодаря большому набору ограничений; многообразием направлений преподаваемых дисциплин; существованием всевозможных критериев оптимальности, которые трудоемко проверить вручную.

Основными связанными структурными объектами, фигурирующими в учебном процессе кафедры, являются дисциплины, преподаватели, должности и виды нагрузки [2]. Благодаря специфике организации учебного процесса на кафедре каждый преподаватель занимает определенную должность, имеет свое направление научного исследования, собственную специализацию, а также проводит конкретные занятия.

Цель работы – адаптация генетического алгоритма для решения задачи распределения учебной нагрузки на кафедре.

**Материал и методы.** В качестве исходных использовались данные учебной нагрузки кафедры прикладного и системного программирования. Основным методом исследования является генетический алгоритм. В настоящей работе для составления учебной нагрузки преподавателей кафедры предлагается композиционный генетический алгоритм, в основе которого лежит объектно-ориентированное представление исходных структурных объектов – участников учебного процесса кафедры.

Генетический алгоритм – это эволюционный алгоритм эвристического вида, состоящий из последовательности действий, которые путем случайного подбора оптимальной нагрузки позволяют найти решение или комбинацию решений к поставленной задаче. Данный алгоритм состоит из нескольких этапов: формирования первого поколения случайным образом; селекции, позволяющей отобрать наиболее приспособленные особи; генерации нового поколения путем применения скрещивания более приспособленных; мутации [3]. Все этапы и генетические операции адаптированы и представлены в виде объектов.

Суть алгоритма заключается в том, что через конечное количество циклических повторений будет найдено оптимальное решение задачи, которое удовлетворяет некоторым критериям. Для каждой задачи данные критерии являются уникальными. Оценка полученных решений происходит на этапе отбора (селекции). Выделяются наиболее важные критерии, по которым будет происходить анализ полученных решений. Далее создается следующее поколение, которое будет модифицировано с помощью тех же генетических операций до того момента, пока мы не придем к оптимальному результату с точки зрения установленных ограничений.

Генетический алгоритм является переменным и допускает различные модификации. Поэтому мы не можем в общем виде точно и однозначно указать этапы. Часто происходит модифицирование определенных шагов и включение операции мутации. Мутация – это выбор с некоторой вероятностью особей и внесение определенных для данной задачи изменений. Такая операция используется для трансформации последующих результатов. Скрещивание позволяет усовершенствовать с точки зрения поставленных критериев результат, изменить привычный ход итераций и за более короткое время найти оптимальное решение [4].

**Результаты и их обсуждение.** Исходной информацией при распределении нагрузки выступают:

а) множество  $D = \{d_1, d_2, \dots, d_{Nd}\}$  дисциплин кафедры;

б) множество  $G = \{g_1, g_2, \dots, g_{Ng}\}$  преподавателей кафедры;

в) множество  $P = \{p_1, p_2, \dots, p_{Np}\}$  видов занятий,

где  $Nd$  – число дисциплин кафедры,  $Ng$  – количество преподавателей кафедры,  $Np$  – количество видов занятий.

Между данными множествами существуют зависимости, которые сформированы в соответствии с организационным процессом работы кафедры. Поскольку исходные данные являются объемными и разнородными, то целесообразно для формирования объектов использовать приемы агрегирования и декомпозиции, что позволит исходную информацию представить в виде совокупности объектов и применить объектно-ориентированную парадигму программирования.

На основе имеющейся информации о нагрузке кафедры, об учебных планах различных специальностей можно сказать, что объекты D, P и G влияют на то, какие именно занятия и по каким дисциплинам должен провести определенный преподаватель.

Классификацию всех преподавателей кафедры можно представить в следующем виде:

1. Преподаватель-стажер (Множество  $G_1$ ).
2. Преподаватель (Множество  $G_2$ ).
3. Старший преподаватель (Множество  $G_3$ ).
4. Доцент (Множество  $G_4$ ).
5. Профессор (Множество  $G_5$ ).

Очевидно, что  $G = G_1 \cup G_2 \cup G_3 \cup G_4 \cup G_5$ .

За каждой из вышеперечисленных должностей закреплено количество часов, а именно обязательный минимум и максимум – это определенный интервал, в который должна вложиться вся нагрузка преподавателя. Также у каждого преподавателя есть специализация, которая обеспечивает закрепление за определенным набором дисциплин.

Для формального описания  $j$ -го преподавателя  $k$ -й квалификации, работающего на  $d$ -й должности, введем кортеж:

$$G = \{g_j\}, g_j = \{g_d^j, g_k^j\}.$$

Входными данными для этой задачи являются фамилия преподавателя, должность, часовой диапазон нагрузки, соответствующий должностям, уровень и возможность преподавателя в проведении дисциплины, название дисциплины и ее вид, общая нагрузка для каждой дисциплины.

Алгоритм реализован в среде разработки Visual Studio 2019, на языке программирования C++. Создан один базовый класс, который называется `object`. От него будут наследоваться два класса, включающие в себя информацию о преподавателях и дисциплинах. Основная функция класса `object` – сохранять в себе общее количество часов. Так, в дисциплинах оно изначально известно, а для преподавателя оно вычисляется. Также создано два метода, с помощью которых возможен доступ к данным. Они позволяют при создании объекта класса вносить изменения или просто использовать существующие данные. Все классы в программе созданы с учетом принципа инкапсуляции. Существуют закрытые поля с данными, доступными только в зоне видимости класса, и публичные, которые содержат конструкторы и различные методы.

От класса `object` наследуется два класса `teacher` и `discipline`. Они хранят в себе всю информацию, которая была дана изначально, а также накапливают полученную.

В работе исследуется возможность применения генетического алгоритма для распределения нагрузки. Задача нахождения оптимального распределения нагрузки при помощи генетического алгоритма заключается в разработке критериев, по которым мы будем оценивать наши возможные решения.

В классе `teacher` осуществляется публичное наследование – публичные и защищенные данные наследуются без изменения уровня доступа к ним. Реализован динамический массив, заполняющийся строковыми значениями, которые означают, какие виды занятий может проводить преподаватель. Каждый объект класса содержит в себе ассоциативный контейнер, который хранит в себе пары, состоящие из ключа и значения. Ключ – это категория, в которой может вести дисциплину преподаватель, а значение – вес, насколько преподаватель профессионален в данном направлении.

Также в классе создана определенная система для хранения результатов после первого и последующих составлений поколений генетического алгоритма. Она реализована через список – последовательный контейнер такой, что любой элемент знает только о предыдущем и о следующем элементах. Данные, которые хранит список, – структуры, состоящие из трех полей: вид занятия (лекция, практика, лабораторная работа), время и указатель на дисциплину, часы которой мы присваиваем данному преподавателю.

Далее создан класс *discipline*, содержащий информацию о предметах. Он содержит четыре приватных поля: название дисциплины, категория и два списка. Первый список создан для упорядоченной записи часов по разным видам занятий, второй список – для записи значений после составления популяции.

Для организации этапа отбора требуется сформулировать условия, предъявляемые к особям популяции. Все требования могут быть обязательными и желательными. К первой группе относятся требования, невыполнение которых делает процесс распределения учебной нагрузки кафедры невозможным (в нашем случае необходимо уложиться во временные интервалы). Эти требования будем рассматривать в качестве обязательных ограничений оптимизационной задачи распределения нагрузки. Вторая группа включает в себя требования, выполнение которых является желательным. К числу данных требований относятся: соблюдение равномерности распределения нагрузки в течение двух семестров, соблюдение соответствия между характером занятий (у преподавателя должны быть и лекционная нагрузка и лабораторные занятия); требования, связанные с обеспечением комфорта преподавателей и студентов.

В генетическом алгоритме решения задачи распределения нагрузки каждая особь является одним из возможных решений задачи, т.е. вариантом распределения нагрузки. В итоге особями в генетическом алгоритме будут являться объекты двух созданных классов *discipline* и *teacher*. Каждая особь состоит из двух хромосом (дисциплина, преподаватель). И после первого и последующих созданий популяций особи будут сохранять в себе ссылки: особь, имеющая данные о преподавателе, будет содержать ссылку на дисциплину и наоборот. Это означает, что первая и вторая хромосомы связаны связью «однозначного соответствия».

Поиск оптимального распределения нагрузки с использованием генетических алгоритмов будет проходить в несколько этапов. Формирование начального поколения производится с помощью произвольного «разброса» часов по преподавателям при помощи функции, которая находит случайное значение в указанном диапазоне. Начальное значение диапазона всегда равно 1, а число часов, выделенное на определенный вид занятий, и явится конечным значением диапазона. Когда случайное число будет выявлено и записано в список и преподавателя и дисциплины, функция вычитает полученное значение из количества часов, отведенных на данный вид занятий. Когда все часы одной дисциплины будут рассмотрены и разбросаны по преподавателям, можно будет переходить к следующей особи и так до тех пор, пока не закончится массив дисциплин. Массив преподавателей заполняется последовательно, с поочередным прохождением по индексам.

На этапе отбора следует самые приспособленные особи перевести в следующую популяцию и сформировать новые особи. Для выбора оптимальных вариантов нагрузки необходимо реализовать функции пригодности для обязательных и желательных требований. Те особи, которые удовлетворяют требованиям, переходят в следующее поколение, а некоторое количество особей в новой популяции заполняется особями, полученными в результате кроссинговера и мутации особей.

Для создания разного рода критериев создан абстрактный класс, который содержит только один чисто виртуальный метод. В дальнейшем во всех наследуемых классах будет происходить его переопределение. Благодаря этому методом можно гибко пользоваться в коде, а именно создавать объекты классов наследуемых от абстрактного и всегда вызывать функцию, имеющую одно и то же имя, но выполняющую разный алгоритм.

В основе одной функции пригодности лежит оценка того, есть ли возможность у преподавателя вести дисциплину определенного вида. Так как в начальные данные, помимо фамилии и должности, входит еще и информация, насколько преподаватель специализируется в каком-либо направлении. Примерами направлений кафедры системного и прикладного программирования являются специалисты в разработке на C++, Java, языке Assembler, HTML, Python.

Для установления условий, при которых особь не способна перейти в следующее поколение, создан абстрактный класс с единственным чисто виртуальным методом, реализующим этап селекции. В производных классах этот метод переопределяется и производит оценку качества полученного решения. Первое условие проверяет, входит ли полученное число часов в промежуток минимума и максимума преподавателя, второе же учитывает возможность конкретного сотрудника преподавать определенные дисциплины.

Скрещивание (кроссинговер) особей происходит по следующей схеме: случайным образом из числа наиболее приспособленных выбираются две особи. Далее для каждой пары отобранных особей случайным образом производится обмен участками генетического кода между соответствующими хромосомами родительских особей. Данный процесс не позволяет избежать лишних проверок на корректность особей, полученных в результате процедуры скрещивания, поскольку у каждого преподавателя существует набор дисциплин, которые он не может преподавать.

К некоторым особям применяется оператор мутации. В предлагаемом алгоритме этот оператор изменяет значение нескольких генов особи на другие допустимые для данного гена значения, например, заменяет дисциплину для некоторого преподавателя на любую из допустимого.

**Заключение.** В результате генетического алгоритма с применением операторов отбора, скрещивания и мутации формируется новое поколение. Часть родительской популяции, удовлетворяющей условию отбора, также переходит в новую популяцию. Данный процесс повторяется до тех пор, пока не будет выполняться условие останова. Предлагается в качестве такого условия использовать следующее: некоторый процент особей имеет одинаковые значения функции приспособленности. В результате работы наиболее приспособленные особи будут переходить в новое поколение и накапливаться. В какой-то момент их количество и качество достигнет необходимого уровня.

В результате работы предложена методика формирования учебной нагрузки с помощью генетического алгоритма. Данная методика реализована с помощью объектно-ориентированной парадигмы программирования на языке программирования C++ с учетом основных принципов объектно-ориентированного программирования и позволяет на основании исходных данных оптимальным образом распределить нагрузку кафедры. В работе предложены различные варианты генетических операторов селекции, мутации, скрещивания, формирования начальной популяции и выполнена оценка эффективности.

#### ЛИТЕРАТУРА

1. Адаменко, Н.Д. Некоторые аспекты учебного процесса на факультете математики и информационных технологий / Н.Д. Адаменко, Л.В. Маркова, Е.А. Корчевская // Наука – образованию, производству, экономике: материалы 73-й Регион. науч.-практ. конф. преподавателей, научных сотрудников и аспирантов, Витебск, 11 марта 2021 г.: в 2 т. / Витеб. гос. ун-т; Е.Я. Аршанский [и др.]. – Витебск, 2021. – Т. 1. – С. 528–529.
2. Маркова, Л.В. Особенности учебного процесса в режиме офлайн / Л.В. Маркова, Н.Д. Адаменко, С.А. Ермоchenko, Е.А. Корчевская // Вестник Віцебскага дзяржаўнага ўніверсітэта. – 2022. – № 3(116). – С. 83–88.
3. Маслов, М.Г. Разработка моделей и алгоритмов составления расписаний в системах административно-организационного управления: дис. ... канд. техн. наук: 05.13.18 / М.Г. Маслов; Моск. гос. ун-т прикладной технологии. – М., 2004. – 217 л.
4. Яндыбаева, Н.В. Генетический алгоритм в задаче оптимизации учебного расписания вуза / Н.В. Яндыбаева // Современные наукоемкие технологии. – 2009. – № 11. – С. 97–98.

#### REFERENCES

1. Adamenko N.D., Markova L.V., Korchevskaya E.A. Nauka – obrazovaniyu, proizvodstvu, ekonomike: materialy 73 Region. nauch.-prakt. konf. преподаvatelei, nauchnykh sotrudnikov i aspirantov, Vitebsk, 11 marta 2021 g.: v 2 t. / Viteb. gos. un-t [Science – for Education, Industry, Economy: Proceedings of the 73rd Regional Scientific and Practical Conference of Teachers, Researchers and Postgraduate Students, Vitebsk, March 11, 2021: in 2 Volumes], Vitebsk, 2021, 1, pp. 528–529.
2. Markova L.V., Adamenko N.D., Yermochenko S.A., Korchevskaya E.A. Vesnik Vitsebskaga dziazhaunaga universiteta [Bulletin of Vitebsk State University], 2022, 3(116), pp. 83–88.
3. Maslov M.G. Razrabotka modelei i algoritmov sostavleniya raspisaniy v sistemakh administrativno-organizatsionnogo upravleniya: dis. ... kand. tekhn. nauk [Development of Models and Algorithms of Academic Schedule Elaboration in Systems of Administrative and Organizational Management: PhD (Technology Science) Dissertation], Mosk. Gos. un-t prikladnoi tekhnologii, M., 2004, 217 p.
4. Yandybayeva N.V. Sovremenniyе naukoemykiye tekhnologii [Contemporary Science Loaded Technologies], 2009, 11, pp. 97–98.

Поступила в редакцию 06.06.2023

Адрес для корреспонденции: e-mail: Korchevskaya.Elena@gmail.com – Корчевская Е.А.