УДК 004.8

СОСТАВЛЕНИЕ РАСПИСАНИЯ УЧЕБНЫХ ЗАНЯТИЙ НА ОСНОВЕ ГЕНЕТИЧЕСКОГО АЛГОРИТМА

И. Ф. Астахова, А. М. Фирас

Воронежский государственный университет

Поступила в редакцию 14.06.2013 г.

Аннотация. В статье рассматривается модель построения оптимального расписания занятий, эволюционная модель, программный комплекс, реализующий генетический алгоритм, результаты вычислительного эксперимента.
Ключевые слова: оптимальное решение, генетический алгоритм, эволюционная модель, интерфейс пользователя, C#.
Annotation. In article the model of creation of optimum lesson schedule, evolutionary model, the program complex realizing genetic algorithm, results of computing experiment is considered.
Keywords: optimum decision, genetic algorithm, evolutionary model, interface of the user, C#.

ВВЕДЕНИЕ

Современные технические средства позволяют организовывать и планировать учебный процесс с использованием моделей, методов и алгоритмов искусственного интеллекта. В системе качества образования одним из основных критериев выступает оптимальность расписаний занятий и других составляющих учебного процесса. Для возможности применения моделей, методов и алгоритмов систем искусственного интеллекта необходимо формализовать учебный процесс. В настоящее время для решения задачи составления расписания применяется ещё один новый подход – нейронные сети (Пилиньский М. Рутковская Д.). Важнейшим недостатком применения этого подхода является сложность выбора начального состояния нейронной сети. В последние годы особое распространение получили исследования методов эволюционного поиска (Ерунов В.П., Морковин И.И. Каширина И.Л., Низамова Г.Ф.). Применение методов эволюционного поиска приводит к получению хороших результатов, однако имеет место высокая вычислительная трудоёмкость и относительная неэффективность на заключительных этапах эволюции. В работе Низамовой Г.Ф. используются методы системного анализа, что позволяет упростить решаемую задачу, но это приводит к жесткой привязке составленного расписания к преподавательскому составу.

ЭВОЛЮЦИОННАЯ МОДЕЛЬ СОСТАВЛЕНИЯ

расписания занятий учебного заведения

Рассматривается учебное заведение, в котором выделяются следующие группы объектов:

a) Множество обучающихся групп G.
b) Множество аудиторий A.
c) Множество дисциплин D.
d) Множество преподавателей P.
e) Множество учебных пар T (временных интервалов проведения занятий).

Если в указанной группе проводятся G занятия в аудитории A по дисциплине D, преподавателем P, во время учебной пары T , то функция принимает значение равное 1, в противном случае – 0.

Пусть необходимо определить

$$\alpha = (\alpha_1, \alpha_2, ..., \alpha_i, ..., \alpha_{N_{\text{блоков}}})$$
$$t = (t_1, t_2, ..., t_i, ..., t_{N_{\text{блоков}}}) \quad (1)$$
$$\rho = (\rho_1, \rho_2, ..., \rho_i, ..., \rho_{N_{\text{блоков}}}),$$

где $\alpha_i \in A$ – код аудитории, назначенный блоку занятий $z_i \in Z$; $t_i \in T$ – код учебной пары, назначенный первому занятию из блока занятий $z_i \in Z$. $\rho_i \in P$ – код преподавателя, назначенный блоку занятий $z_i \in Z$.

К расписанию предъявляется множество требований, и ограничений. Условно, весь ряд ограничений разбивается на обязательные ограничения и желательные. К обязательным относятся [3,4]:

– отсутствие накладок различного характера;

– отсутствие окон;

– соответствие типа проводимого занятия аудитории проведения;

– ограничение на объем ежедневно – проводимых занятий;

– обязательное проведение всех занятий, запланированных рабочим учебным планом.

Желательные (неосновные) требования представляются в следующем виде:

– обеспечение комфортности условий обучения, а именно минимизация переходов между аудиториями или корпусами, подбор аудитории, максимально приближенных к типу проводимых занятий и т.д.;

– пожелания преподавательского состава;

– равномерность нагрузки студентов в течение всего семестра, а также конкретного учебного дня.

Ограничения, налагаемые на расписание, описываются следующим образом [3-4]:

$$1.\quad \begin{gathered}\forall(a_i,t_j): a_i \in A, t_j \in \\ \in T(\exists!\, z_k : (a_i = a_k) \wedge (z_k \in Z^{t_j})) \vee \\ \vee(\neg\exists z_k : (a_i = a_k) \wedge (z_k \in Z^{t_j})),\end{gathered} \qquad (2)$$

где Z^{t_k} – множество блоков занятий, проводимых во время пары t_k.

Это ограничение гарантирует отсутствие накладок для аудиторий. Для каждой упорядоченной двойки элементов: аудитория и пара, аудитории существует либо единственный блок занятий из множества Z, что означает проведение занятия этого блока в этой аудитории в момент данной пары, либо отсутствие блока занятия, указывающее на то, что аудитория свободна.

$$2.\quad \begin{gathered}\forall(p_i,t_j): p_i \in P, t_j \in \\ \in T(\exists!\, z_k : (p_i = p_k) \wedge (z_k \in Z^{t_j})) \vee \\ \vee(\neg\exists z_k : (p_i = p_k) \wedge (z_k \in Z^{t_j})),\end{gathered} \qquad (3)$$

где Z^{t_j} – множество блоков занятий, проводимых во время пары t_k.

Это ограничение гарантирует отсутствие накладок для преподавателей: существует либо единственный блок занятий, которые ведет данный преподаватель во время заданной пары, либо этого блока не существует вообще.

$$3.\quad \begin{gathered}\forall(g_n,t_j): g_n \in G, t_k \in \\ \in T\sum z_i^e \le 1, i \in Z^{g_n} \bigcap Z^{t_j},\end{gathered} \qquad (4)$$

где Z^{g_n} – множество блоков занятий, в которых присутствует группа g_n, а Z^{t_j} – множество блоков занятий, проводящихся во время пары t_k.

Это ограничение обеспечивает отсутствие накладок для учебных групп, т.е. для каждой пары элементов: группа и пара, сумма компонент z_i^e вектора z_i блоков из множества $Z^{g_j} \bigcap Z^{t_k}$ не превышает единицы.

Во время конкретной пары группа находится на одном занятии, или проводится занятие только у одной из подгрупп, либо у обеих, либо занятий нет вообще.

$$4.\qquad \forall z_i \in Z\ a_i \in A^{z_i^a}, \qquad (5)$$

это есть соответствие типа аудитории проводимому занятию, т.е. для каждого блока занятия $z_i, z_i \in Z$ аудитория выбирается из допустимого подмножества аудиторий, код этого подмножества хранит компонента z_i^a.

$$5.\quad \begin{gathered}\forall(b_\tau,g_n): b_\tau \in B, g_n \in \\ \in G\sum z_i^e \le N_{\max}, i \in I_{g_n}^{b_\tau},\end{gathered} \qquad (6)$$

где $B = \{b_1, b_2, \ldots, b_N\}$ – множество учебных дней. Каждый элемент описанного множества, определяется следующим образом:

$$b_\tau = \{t_j \in T : t_j^d = b_\tau\}.$$

Таким образом выглядит ограничение, налагаемое на количество учебных пар, проводимых в течение одного учебного дня, означает, что для каждой пары элементов: группа и день, число проводимых пар не превышает максимально допустимого – $N_{\max}$.

$$6.\quad \begin{gathered}\forall(b_\tau,g_n): b_\tau \in B, g_n \in G \\ (\sum_{i \in I_{g_n}^{b_\tau}} z_i^e = t\_\max\_number_{g_n}^{b_\tau} - \\ -t\_\min\_number_{g_n}^{b_\tau} + 1) \\ \wedge(\forall t : t\_\min l_{g_n}^{b_\tau} \le t \le \\ \le t\_\max l_{g_n}^{b_\tau} \sum_{i: z_i \in Z^{g_n} \wedge t_j^p = t} z_i^e = 1)\end{gathered} \qquad (7)$$

$t\_\max\_number_{g_n}^{b_\tau} - t\_\min\_number_{g_n}^{b_\tau} + 1$, а также отсутствие окон у групп.

$t\_\max\_number_{g_n}^{b_\tau}$ – максимальный номер пары в течение дня b_τ для группы g_n,

$t\_\min\_number_{g_n}^{b_\tau}$ – минимальный номер пары в течение дня b_τ ,для группы g_n,

$I_{g_n}^{b_\tau} = \{i : (z_i \in Z^{g_n}) \wedge (t_j^d = b\tau)\}$ – множество номеров блоков занятий, проводимых для группы g_n во время дня b_τ.

Таким образом, записывается ограничение отсутствия окон для учебных групп, т.е. для каждой пары элементов: день и группа, количество пар, проводимых в этой группе в текущий день должно равняться величине:

Требуется найти такой вариант выбора векторов α, t, ρ, удовлетворяющее ограничениям $(2)-(7)$, а также минимизирующее значение критерия потери качества K. Критерий качества основывается на желательных требованиях и имеет следующий вид;

$$K = \varphi(\alpha, t, \rho) = \sum_{i=1}^{N} c_i w_i(\alpha, t, \rho),$$

где w_i – значение штрафного коэффициента за невыполнение i-го требования, а w_i – оценка степени невыполнения i-го желательного требования.

К наиболее значимым желательным требованиям относятся:

1. пожелания преподавательского состава: Для формулировки данного требования рассматриваются две матрицы. Первая матрица называется матрицей запретов и выглядит следующим образом:

$$M_{\text{запретов}} = \begin{cases} \text{1, запрет проведения занятия} \\ \text{для } i \text{ - го преподавателя} \\ \text{во время } k \text{ - ой пары} \\ \text{0, отсутствие запрета} \end{cases}$$

Вторая матрица – матрица занятости формируется следующим образом:

$$M_{\text{занятости}} = \begin{cases} \text{1, проведение занятия} \\ \text{для } i \text{ - го преподавателя} \\ \text{во время } k \text{ - ой пары} \\ \text{0, отсутствие занятий} \end{cases}$$

$$\sum_{p_i \in P} \sum_{t_j \in T} \overline{(M_{\text{запретов}} \wedge M_{\text{занятости}})} \to \min$$

Есть требование учета пожеланий.

2. $\sum \sum (t\_\max_{p_i}^{b_\tau} - t\_\min_{p_i}^{b_\tau}) - N_{\text{блоков}} \to \min,$

где $t\_\max_{p_i}^{b_\tau}$ – максимальный номер пары в день b_τ у преподавателя p_j, а $t\_\min_{p_i}^{b_\tau}$ – минимальный номер пары в день b_τ у преподавателя p_j.

Это ограничение означает минимизацию количества окон у преподавателей.

3. $D_n^{\text{ср}} = \dfrac{1}{N_{\text{дней}}} \sum_{b_\tau} (M_n^{\text{ср}} - \left| I_{g_n}^{b_\tau} \right|)^2$ – желательное требование равномерности занятий.

Если среднее отклонение количества занятий для группы g_n

$$M_n^{\text{ср}} = \frac{1}{N_{\text{дней}}} \sum_{b_\tau \in B} \left| I_{g_n}^{b_\tau} \right|,$$

Отсюда, требование выглядит так:

$$\sum_{\underline{n=1, N_{\text{групп}}}} D_n^{\text{ср}} \to \min$$

На основании описанных требований строится целевая функция на основе минимизации штрафных показателей. Каждое нарушение ограничения или желательного требование увеличивает значение целевой функции в соответствии с коэффициентом значимости требования. В результате целевая функция в общем виде описывается формулой:

$$F_{\text{ц}} = \sum_{\underline{i=1..N_{\text{количество ограничений}}}} og_i \cdot k\_og_i) + K$$

На основании построенной математической модели (1–7) расписания производится реализация эволюционного поиска оптимального расписания с использованием генетического алгоритма.

ГЕНЕТИЧЕСКИЙ АЛГОРИТМ И ПРОГРАММНЫЙ КОМПЛЕКС РЕАЛИЗАЦИИ ПОСТРОЕННОЙ МОДЕЛИ

Разработанный алгоритм состоит из следующих шагов:

1. формирование начальной популяции;
2. селекция особей;
3. скрещивание особей случайными значениями функции пригодности;
4. операция мутации над потомством;
5. отбор особей в новую популяцию;
6. проверка критерия остановки алгоритма;
7. выбор наилучшей особи.

1. ФОРМИРОВАНИЕ НАЧАЛЬНОЙ ПОПУЛЯЦИИ

На первом этапе случайным образом формируется исходная популяция, состоящая из заданного числа N особей, где каждая особь популяции представляет собой отдельный вариант расписания (решение задачи).

2. СЕЛЕКЦИЯ ОСОБЕЙ

На этапе происходит отбор (селекция) наиболее приспособленных особей (вариантов расписания), имеющих более предпочтительные значения функции пригодности по сравнению с остальными особями.

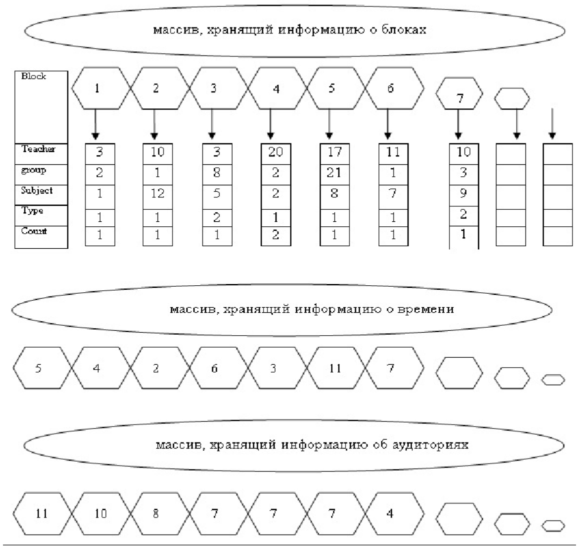


Рис. 1. Схема особи

3. КРОССИНГОВЕР

Третьим этапом генетического алгоритма является скрещивание. На этом этапе основную функцию выполняет оператор кроссинговера. Это языковая конструкция, позволяющая на основе скрещивания хромосом родителей создавать хромосомы потомков.

4. ОПЕРАЦИЯ МУТАЦИИ

К некоторым из особей получившимся после скрещивания применяется оператор мутации. Мутация играет достаточно важную роль в работе генетического алгоритма: вносит дополнительное разнообразие в текущую популяцию и тем самым расширяет пространство поиска оптимального решения.

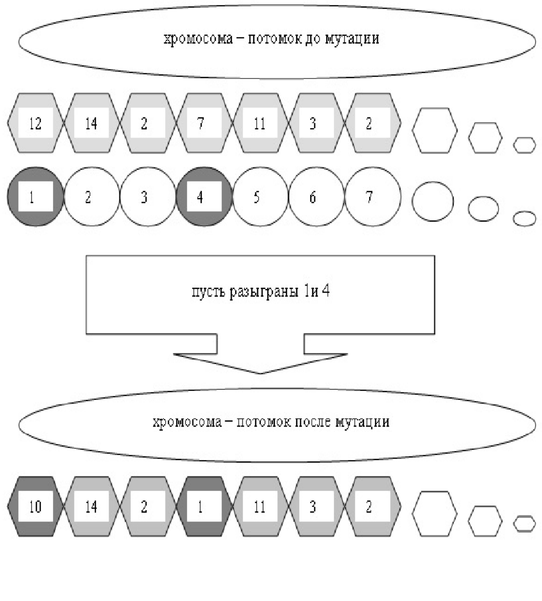


Рис. 2. Двухточечный оператор мутации

5. ОПЕРАТОР ОТБОРА

Оператор отбора выполняет функции фильтрующего инструмента, который выделяет в составе популяции особи, имеющие низкое значение функции пригодности. Выделенные найденные слабые особи удаляются из популяции до тех пор, пока численность не становится исходной.

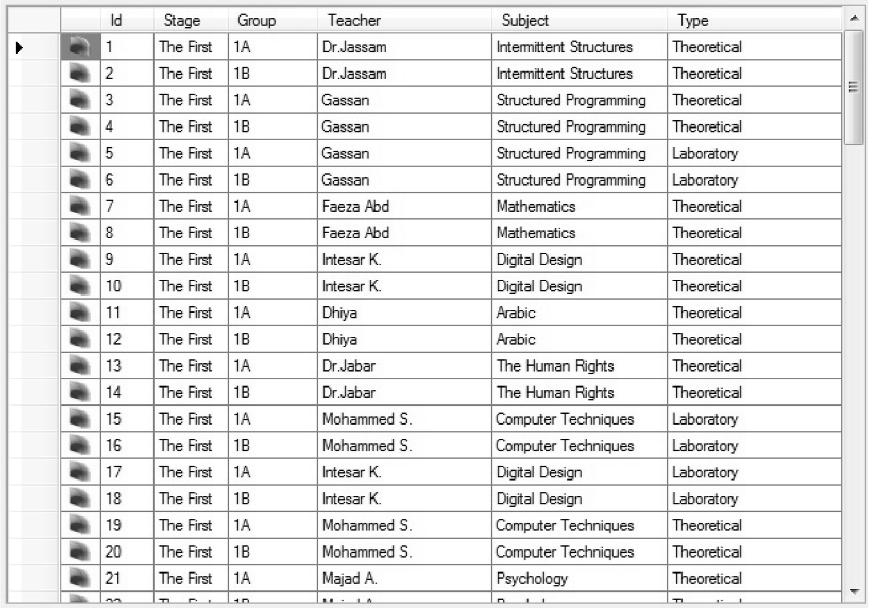
6. ПРОВЕРКА УСЛОВИЯ ПРЕКРАЩЕНИЯ РАБОТЫ АЛГОРИТМА

Новое поколение, сформированное в результате применения операторов отбора, скрещивания, мутации, называемое популяцией потомков, заменяет родительскую популяцию, после чего выполняется проверка условия прекращения работы алгоритма. Она основана на использовании приращения функции пригодности, т.е. если в течение нескольких поколений особей приращение значения функции пригодности наиболее "лучшего" индивидуума оказывается незначительным, то работу алгоритма завершается. При выполнении заданного в алгоритме условия останова осуществляется переход к следующему этапу, в противном случае происходит переход к этапу селекции и процесс поиска оптимального решения продолжается.

7. ВЫБОР «ЛУЧШЕГО» РЕШЕНИЯ

На этом этапе среди полученных особей выбирается наилучшая особь, которая и будет являться решением задачи. Под лучшей особью понимается та, у которой значение функции пригодности является минимальным.

Описанный алгоритм был реализован с использованием языка C #.[1–2] Для отладки данного алгоритма использовались экспериментальные данные по факультету компьютерной науки вуза Ирака. В испытаниях принимало участие порядка ста предметов. Между описанными дисциплинами были определены связи, которые характеризуют очередность следования предметов друг за другом. Информация о дисциплинах и связях между ними хранится в базе данных, работа с которой реализуется через интерфейс. На рис. 3 представлена форма обработки дисциплин, соответствующих специальности «компьютерной науки», где каждая строка является блоком.



Id	Stage	Group	Teacher	Subject	Type
1	The First	1A	Dr.Jassam	Intermittent Structures	Theoretical
2	The First	1B	Dr.Jassam	Intermittent Structures	Theoretical
3	The First	1A	Gassan	Structured Programming	Theoretical
4	The First	1B	Gassan	Structured Programming	Theoretical
5	The First	1A	Gassan	Structured Programming	Laboratory
6	The First	1B	Gassan	Structured Programming	Laboratory
7	The First	1A	Faeza Abd	Mathematics	Theoretical
8	The First	1B	Faeza Abd	Mathematics	Theoretical
9	The First	1A	Intesar K.	Digital Design	Theoretical
10	The First	1B	Intesar K.	Digital Design	Theoretical
11	The First	1A	Dhiya	Arabic	Theoretical
12	The First	1B	Dhiya	Arabic	Theoretical
13	The First	1A	Dr.Jabar	The Human Rights	Theoretical
14	The First	1B	Dr.Jabar	The Human Rights	Theoretical
15	The First	1A	Mohammed S.	Computer Techniques	Laboratory
16	The First	1B	Mohammed S.	Computer Techniques	Laboratory
17	The First	1A	Intesar K.	Digital Design	Laboratory
18	The First	1B	Intesar K.	Digital Design	Laboratory
19	The First	1A	Mohammed S.	Computer Techniques	Theoretical
20	The First	1B	Mohammed S.	Computer Techniques	Theoretical
21	The First	1A	Majad A.	Psychology	Theoretical

Рис. 3. Список дисциплин по специальности «компьютерная наука»

Рисунок показывает работу генетического алгоритма, которая состоит из следующих шагов:

1. необходимо выбрать нужные условия из пяти возможных;

2. установить размер популяции (на рис. Это значение равно 1000);

3. определить количество повторов алгоритма (на рис. Данное значение равно 120, но получен результат раньше);

4. определить размер мутации (на рис. Это значение равно 0.01).

После заполнения всех данных алгоритм начинает работать. По истечении нескольких минут появляется результат составления расписания факультета. Для получения результата, представленного на рисунке, потребовалось 5 минут [1 - 2].

Скорость сходимости данного алгоритма регулируется несколькими важными характеристиками. Одна из самых важных – численность популяции. На протяжении работы алгоритма («жизни популяции») ее численность не меняется. Это связано с операциями селекции,

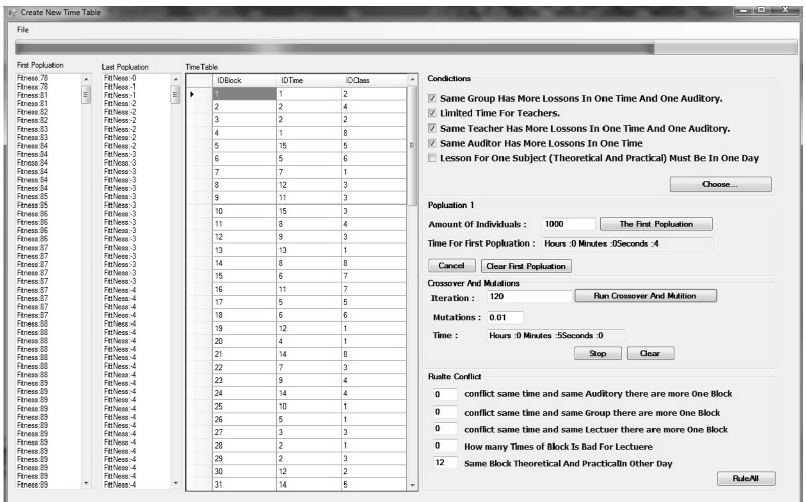


Рис. 4. Интерфейс работы алгоритма

кроссинговера, мутации и естественного отбора. Это можно проследить на рис. 5, с учетом того, что численность популяции должна оставаться равной ста.

На рис. 6 представлен результат работы алгоритма – расписание. Оно состоит из даты, времени, групп, преподавателей, предметов, аудиторий и типа занятий. Это расписание со-

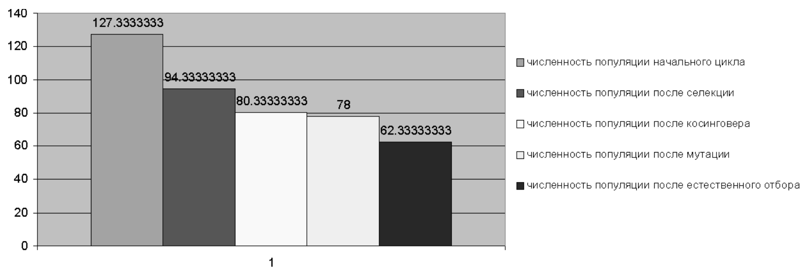
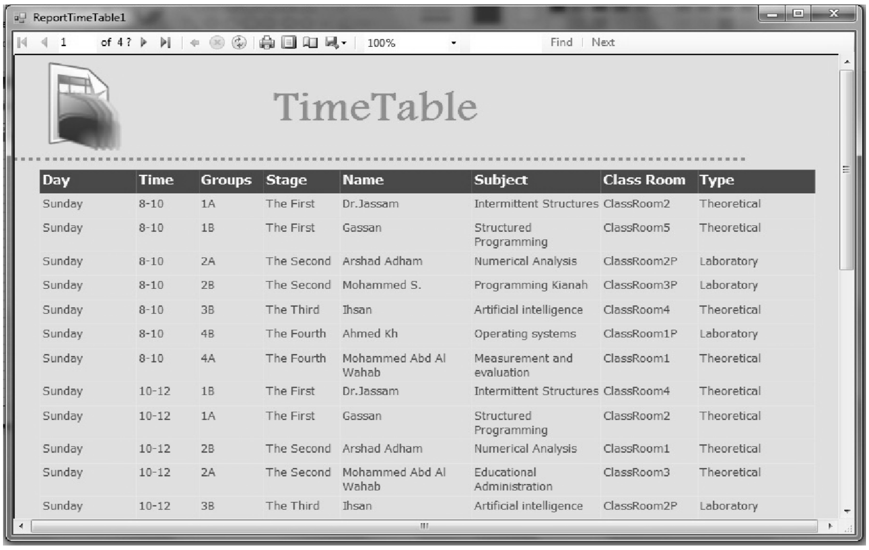


Рис. 5. Динамика изменения генов



Day	Time	Groups	Stage	Name	Subject	Class Room	Type
Sunday	8-10	1A	The First	Dr.Jassam	Intermittent Structures	ClassRoom2	Theoretical
Sunday	8-10	1B	The First	Gassan	Structured Programming	ClassRoom5	Theoretical
Sunday	8-10	2A	The Second	Arshad Adham	Numerical Analysis	ClassRoom2P	Laboratory
Sunday	8-10	2B	The Second	Mohammed S.	Programming Kianah	ClassRoom3P	Laboratory
Sunday	8-10	3B	The Third	Ihsan	Artificial intelligence	ClassRoom4	Theoretical
Sunday	8-10	4B	The Fourth	Ahmed Kh	Operating systems	ClassRoom1P	Laboratory
Sunday	8-10	4A	The Fourth	Mohammed Abd Al Wahab	Measurement and evaluation	ClassRoom1	Theoretical
Sunday	10-12	1B	The First	Dr.Jassam	Intermittent Structures	ClassRoom4	Theoretical
Sunday	10-12	1A	The First	Gassan	Structured Programming	ClassRoom2	Theoretical
Sunday	10-12	2B	The Second	Arshad Adham	Numerical Analysis	ClassRoom1	Theoretical
Sunday	10-12	2A	The Second	Mohammed Abd Al Wahab	Educational Administration	ClassRoom3	Theoretical
Sunday	10-12	3B	The Third	Ihsan	Artificial intelligence	ClassRoom2P	Laboratory

Рис. 6. Результаты работы алгоритма

ставлено для одного факультета на один семестр, где каждая учебная неделя состояла из пяти дней по 3 пары в день. Для его составления были задействованы 8 групп, 18 преподавателей, 53 предмета, 7 аудиторий. На основе предложенной математической модели предложена эволюционная модель для составления расписания вуза, разработан программный комплекс на языке C# и проведен вычислительный эксперимент в конкретной ситуации.

ЗАКЛЮЧЕНИЕ

В работе на основе сформулированных требований к виду и качеству учебного расписания построена эволюционная модель, генетический алгоритм ее реализации. Результаты экспериментальных расчетов показали, что разработанный комплекс может быть успешно реализован для составления расписаний учебных занятий.

СПИСОК ЛИТЕРАТУРЫ

1. Microsoft Corporation. Разработка Windows-приложений на Visual Basic .NET и Visual C# .NET. Учебный курс MCAD/MCSD. / – Пер. с англ. – М.: Русская редакция, 2003 – 512 с.

2. *Робинсон С., Корнес О.* и др. C# для профессионалов, в 2-х томах / – Пер. с англ. – М.: Лори. , 2005. – С. 1002.

3. *Коробкин А.А.* Использование агрегативного генетического алгоритма для составления расписа-

ния / Коробкин А.А. //Вестник Воронежского государственного технического университета, том 5, № 11,2009. –С. 184–186.

4. *Астахова И.Ф.* Разработка информационной системы построения расписания / И.Ф. Астахова, Т.В. Курченкова // Математика. Образование. Экология. Гендерные проблемы. : Материалы межд. конф. – Т. 2.– М.: Прогресс-Традиция, 2001. – С. 287–290.

Астахова И. Ф. – доктор технических наук, профессор кафедры математического обеспечения ЭВМ факультета прикладной математики, информатики и механики Воронежского государственного университета, профессор. Телефон: (473) 2-208-638-314. E-mail: astachova@list.ru.

Фирас А. М. – аспирант кафедры прикладного и математического анализа факультета прикладной математики, информатики и механики Воронежского государственного университета. Тел.: (473) 2-208-638-314. E-mail: astachova@list.ru

Astachova I. F. – Doctor of Technical Science, Professor of the chair mathematical software, Voronezh State University. Phone: (473) 2-208-638-314. E-mail: astachova@list.ru

Firas A. M. – Postgraduate student of the chair applied and mathematical analyses, Voronezh State University. Phone: (473) 2-208-638-314. E-mail: astachova@list.ru