

## РЕШЕНИЕ ЗАДАЧИ ПРОВЕРКИ ГРАФОВ НА ИЗОМОРФИЗМ С ИСПОЛЬЗОВАНИЕМ ГЕНЕТИЧЕСКОГО АЛГОРИТМА

Д.К.Саввинов, П.А.Хаустов  
Томский политехнический университет  
[dsk2@tpu.ru](mailto:dsk2@tpu.ru)

### Введение

В последнее время, благодаря уменьшающейся цене и увеличивающемуся качеству хранения информации, базы данных находят все большее применение в различных областях науки и техники. Однако вместе с этим увеличивается и сложность работы с такими базами. Одной из частых подзадач, возникающей в хемиоинформатике [1], автоматизации проектирования электронных схем [2], сравнении пар кинематических цепей [3], распознавании графических образов и др. является проверка графов на изоморфизм.

Основная идея в том, что хранящиеся в БД структуры представлены как некоторая разновидность графа. Соответственно, искомым в БД элемент также является графом, называемым эталонным. Отсюда очевидным образом возникает задача проверки двух графов на «одинаковость», или, иначе говоря, на изоморфизм.

### Понятие изоморфизма графов

Под изоморфизмом графов  $G(V_G, E_G)$  и  $H(V_H, E_H)$  здесь будет пониматься следующее. Пусть задана биекция  $f: V_G \rightarrow V_H$  между множествами вершин графов. Образы вершин  $u$  и  $v$  ( $u, v \in V_G$ ) обозначим как  $u'$  и  $v'$  ( $u', v' \in V_H$ ) соответственно. Тогда если для любой пары  $u$  и  $v$ , смежных в  $G$ , смежны их образы  $u'$  и  $v'$  (и наоборот), то граф  $G$  изоморфен графу  $H$  [4]. В случае если  $G$  и  $H$  являются помеченными графами, то накладывается дополнительное условие на совпадение меток ребер ( $u, v$ ) и ( $u', v'$ ).

Интуитивно, суть определения в том, что графы изоморфны, когда они совпадают с точности до переобозначения вершин.

На рис. 1 приведен пример двух изоморфных графов:

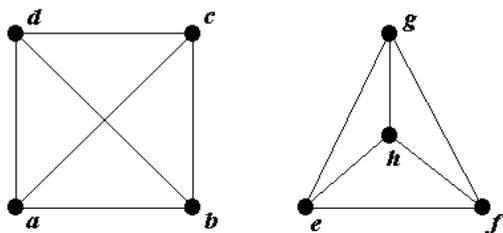


Рис. 1. Пример изоморфных графов.

Следует отметить, что задача проверки графов на изоморфизм является вычислительно трудной и принадлежит классу NP, однако ее NP-полнота остается недоказанной, равно как и не существует

на данный момент алгоритма, решающего эту задачу за полиномиальное время.

По этой причине до сих пор большой интерес представляют алгоритмы, позволяющие решать эту задачу относительно быстро. На сегодняшний день предложено множество различных эвристик, однако некоторые из них применимы только к графам специального вида [5]; другие же могут работать слишком долго или вовсе неправильно на некоторых графах [6].

### Понятие генетического алгоритма

В представленной работе изучается подход к решению задачи проверки графов на изоморфизм с точки зрения эволюционных вычислений.

В качестве основы был выбран генетический алгоритм. Он представляет собой алгоритм эвристического поиска, использующий в качестве методов подбора, комбинирования и отсеивания решений механизмы, сходные с естественным отбором в природе.

Введем несколько ключевых понятий, сформулированных в общих терминах (ниже будет дано разъяснение, какое значение они несут в конкретной решаемой задаче):

- Генотип – некоторое решение задачи, определенным образом закодированное.
- Ген – единичный элемент генотипа.
- Популяция (поколение) – совокупность генотипов на какой-то момент работы генетического алгоритма.
- Функция приспособленности – функция, оценивающая, насколько близко данное решение задачи (данный генотип) к идеальному.
- Кроссинговер (скрещивание) – получение нового генотипа при помощи комбинирования двух уже имеющихся.
- Мутация – получение нового генотипа при помощи случайного изменения уже имеющегося.
- Селекция (отбор) – процесс отсеивания генотипов с низкими значениями функции приспособленности.

В общем виде генетический алгоритм состоит из нескольких фаз [8]:

0. Задается начальная популяция (случайно или с помощью эвристик).
1. Размножение. Произвольно выбираемые генотипы скрещиваются с помощью оператора кроссовера, формируя новые особи.
2. Мутация. Несколько произвольных генотипов подвергаются мутации. Этот этап позволяет нахо-

дить принципиально новые решения, в том числе и те, которых вообще не было в начальной популяции.

3. Селекция. Из популяции выбирается определенное количество генотипов, с наилучшими значениями функции приспособленности. Этот этап вкупе с этапом №1 позволяет постепенно улучшать решения, находящиеся в популяции, отыскивая таким способом экстремум функции приспособленности.

4. Формирование нового поколения. На этом этапе формируется новое поколение. Этот этап стоит выделить отдельно, так как в некоторых случаях для улучшения сходимости и способности находить глобальные (а не локальные) экстремумы, новая популяция должна набираться разнообразней, нежели просто несколько «лучших» в смысле функции приспособленности генотипов из предыдущей.

На рисунке 2 схематически описана работа генетического алгоритма:



Рис. 2. Схема работы генетического алгоритма  
**Алгоритм решения задачи**

Теперь можно пояснить, какой смысл несут эти термины применительно к задаче проверки графов на изоморфизм.

- **Генотип.** Пусть вершины и в эталонном графе  $G$ , и в проверяемом графе  $H$  нумеруются 1 до  $n$ , где  $n$  – число вершин (очевидно, что оно должно быть одинаковым для обоих графов). Тогда проверка графа на изоморфизм сводится к поиску такой перестановки (перенумерации) вершин графа  $H$ , что он полностью совпадает с графом  $G$  – а, значит, совпадают их матрицы смежности (или любые другие структуры, описывающие ребра в графе).

Отсюда логично будет определить генотип, как некоторую перестановку чисел от 1 до  $n$ .

- **Ген.** Из вышесказанного прямо следует, что ген – это значение  $i$ -ого элемента конкретной перестановки.

- **Функция приспособленности** – в качестве функции приспособленности использовался подсчет количества ребер, не противоречащих устанавливаемой биекции, а также некоторые другие эвристики.

- **Кроссинговер.** Следует заметить, что применение классических операторов кроссинговера в данном случае не применимо, так как дочерний

генотип может оказаться некорректным (например, у разных генов будет одинаковое значение). По этой причине были использованы специфические разновидности оператора кроссинговера: кроссинговер порядка, циклический кроссинговер, частично отображающий кроссинговер [7].

- **Мутация.** Также как и в случае с кроссинговером, классические операторы мутации могут приводить к появлению некорректных генотипов. Для решения этой проблемы в данной работе использовались модифицированные операторы мутации. Под точечной мутацией понимается перестановка двух соседних генов генотипа.

При инверсии порядок следования генов на некотором интервале генотипа изменяется на обратный.

В качестве сальтации использовалась перестановка к пар генов (в каждой паре гены менялись местами).

### Заключение

Таким образом, был показан эволюционный подход к решению задачи о проверке графов на изоморфизм. Следует отдельно подчеркнуть, что генетические методы не могут гарантировать поиска решения за полиномиальное время. Тем не менее, они способны найти весьма «хорошее» в некотором смысле решение, причем сделать это значительно быстрее альтернативных эвристических алгоритмов.

### Литература

1. Трофимов М. И., Смоленский Е. А. Применение индексов электроотрицательности органических молекул в задачах химической информатики // Известия Академии наук. Серия химическая. – 2005. – С. 2166-2176.
2. Курейчик В. М., Глушань В. М., Щербаков Л. И. Комбинаторные аппаратные модели и алгоритмы в САПР. – М.: Радио и связь, 1990 – 216 с.
3. Rao A.C., Varada Raju D. Application of the Hamming number technique to detect isomorphism among kinematic chains and inversions // Mech. Mach. Theory. – 1991. – №26(1). – P.55–75.
4. Харари Ф. Теория графов. – М.: Мир, 1973.
5. Spielman D.A. Faster isomorphism testing of strongly regular graphs: Technical report. – University of California Berkeley, Computer Science Division, Berkeley, California, 1996.
6. Abdulrahim M., Misra M. A Graph Isomorphism Algorithm for Object // Recognition, Pattern Analysis and Applic. – 1998. – V. 1. – №3. – P. 189-201
7. Батищев Д. И., Неймарк Е.А., Старостин Н.В. Применение генетических алгоритмов к решению задач дискретной оптимизации. Нижний Новгород, 2007, 85 с.
8. Гладков Л. А., Курейчик В. В., Курейчик В. М. Генетические алгоритмы / Под ред. В. М. Курейчика. – 2-е изд., исправл. и доп. – М.: ФИЗМАТЛИТ, 2010. – 368 с.