



ОТКРЫТАЯ НАУКА  
ИЗДАТЕЛЬСТВО

Международный журнал информационных технологий и  
энергоэффективности

Сайт журнала: <http://www.openaccessscience.ru/index.php/ijcse/>



УДК 004.42

## О РЕШЕНИИ ЗАДАЧ ДИДАКТОМЕТРИИ СРЕДСТВАМИ АБСТРАКТНОЙ ГРАФ-МАШИНЫ

Букачев Д.С.

*ФГБОУ ВО Смоленский государственный университет, Смоленск, Россия  
(21400, г. Смоленск, ул. Пржевальского, 4), e-mail: dsbuka@yandex.ru*

Статья посвящена вопросам разработки программного обеспечения в сфере математического моделирования психолого-педагогических процессов и явлений. В работе предложена концепция программного инструмента для решения задач дидактики на базе абстрактной граф-машины, реализующей универсальную графовую алгебраическую систему. Дидактическая граф-машина позволяет строить модели учебной темы, находить веса графа, его инварианты, оценивать сложность темы, определять количество ассоциативных связей между ее элементами, находить оптимальную траекторию обучения.

Ключевые слова: универсальная алгебраическая система, абстрактная граф-машина, дидактика, объектно-ориентированный подход.

## ABOUT THE SOLUTION OF DIDACTOMETRY PROBLEMS BY MEANS OF THE ABSTRACT GRAPH MACHINE

Bukachev D.S.

*Federal State Educational Institution of Higher Education Smolensk State University, Smolensk, Russia (21400, Smolensk, street Przewalski, 4), e-mail: dsbuka@yandex.ru*

Article is devoted to questions of software development in the sphere of mathematical simulation of psychology and pedagogical processes and the phenomena. In operation the concept of the software tool for the decision of tasks of a didactometry the basis of the abstract graph machine realizing the universal graph algebraic system is offered. Didactometrical graph machine allows to build models of an educational subject, to find weights of a graph, its invariants, to estimate complexity of a subject, to define the number of the associative communications between its elements, to find an optimum path of training.

Key words: the universal algebraic system, the abstract graph machine, didactometry, object-oriented approach.

Использование графовых моделей при решении современных прикладных задач весьма обусловлено. Благодаря своей наглядности и достаточно разработанной теории, часто бывает удобно описать структуру и свойства исследуемого объекта или процесса средствами дискретной математики. Такая формализация прикладной задачи дает возможность оперировать терминами теории графов и использовать в ходе исследования известные

алгоритмы. Методы математического исследования могут в определённой степени дополнить и развить традиционные методы психолого-педагогического анализа [2, 4, 7]. Это позволяет ставить вопрос о введении обоснованных количественных методов изучения педагогических явлений и объектов, установлении содержательных критериев для определения величин, характеризующих те или иные стороны этих явлений и процессов. При этом перспективной с точки зрения решения задач дидактики представляется формализация содержания предметной области. Использование графовых моделей в педагогике к настоящему моменту предлагалось многими авторами. Можно говорить о существовании определенных традиций графового моделирования в этой области [2, 5].

Модели, полученные при формализации разделов изучаемых дисциплин, часто характеризуются значительным объёмом, сложной структурой и динамикой. При решении конкретной задачи требуется выбрать оптимальный способ машинного представления графов: матрица смежности, матрица инцидентности, списковые структуры, отличающиеся друг от друга простотой доступа к данным, скоростью их обработки, “гибкостью” в использовании. Выбор структуры порождает необходимость модификации реализации операций над графами и базовых алгоритмов (волновой алгоритм, алгоритм Дейкстры, поиск циклических маршрутов, алгоритм Прима-Краскала). Несомненно, это вызывает ряд технических трудностей при программной реализации модели, поскольку разработчику зачастую приходится отвлекаться от рассматриваемой прикладной задачи и тратить время на реализацию давно известных алгоритмов.

Весьма актуальным является создание универсальных инструментов работы с графами, инвариантных относительно конкретных приложений и типов данных, сопоставляемых элементам графа. В работах [3, 6] предлагается оригинальный подход на основе алгебраизации теории графов и объектно-ориентированного подхода в программировании. Исследование многоосновных графовых алгебр, где множество-носитель – множество всех взвешенных графов, а роль операций выполняют известные алгоритмы на графах, позволяет говорить о возможности создания такой универсальной алгебраической системы, которая позволит работать с любыми графовыми моделями благодаря отделению операций структуры от операций над элементами типа.

Поскольку операции над графами (такие, как добавление/удаление элементов), а также реализация основных алгоритмов на графах фактически не зависят от характера данных, сопоставляемых элементам графа, и операций над этими данными, авторам удалось создать так называемую абстрактную граф-машину (АГМ) на базе списковых структур, работающую с абстрактными данными и позволяющую создавать весьма сложные и динамические графовые модели [3].

Для решения прикладной задачи с помощью АГМ разработчик должен создать реальную граф-машину, то есть построить класс-наследник, описать алгебры используемых данных, при необходимости задать порядок на множестве весов, после чего инициализировать виртуальную машину описанными им методами. Реальная граф-машина при таком построении будет оперировать уже понятиями той предметной области, в которой решается прикладная задача.

В настоящее время АГМ способна решать следующие задачи:

- создание динамических графовых моделей практически неограниченной сложности независимо от характера и алгебры данных, сопоставляемых вершинам графа (благодаря использованию списковых структур);

- нахождение минимального остова на неориентированных графах (алгоритм Прима-Краскала) и на графах смешанного типа (алгоритм комбинаторного характера);
- поиск минимального маршрута как по числу ребер (волновой алгоритм), так и исходя из весовых коэффициентов ребер (алгоритм Дейкстры);
- анализ модели и определение ее характеристик (нахождение инвариантов графа).

Последняя задача особенно актуальна при решении задач дидактики. В данной области при исследовании различных характеристик педагогического процесса, оценки трудности и сложности учебных задач наиболее естественным является использование именно графовых моделей в силу своей наглядности и простоты интерпретации. При построении модели конкретной учебной темы с вершинами графа ассоциируются элементы знаний по данной теме, а ребрам сопоставляется некоторая оценка сложности учебной деятельности, которую должны осуществить учащиеся при переходе от одного понятия к другому [1, 2]. Оценивая вес графа, определяя его инварианты, можно сделать вывод о сложности темы, количестве ассоциативных связей между ее элементами, построить оптимальную траекторию обучения [2].

Рассмотрим реализацию дидактической граф-машины (ДГМ) на базе АГМ.

#### *Листинг 1. Описание класса-наследника АГМ*

```
//описание типов данных, сопоставляемых элементам графа
TObjType=(Intermediate,Source,Final);
TData=record //тип данных вершины
    Obj:integer;
    ObjType:TObjType;
end;
TWeight=record //тип данных весового коэффициента
    Operation:integer;
    Difficulty:real;
end;
TMark=string; //тип данных метки
//типы указателей на данные вершины, весового коэффициента и метки
TDataPointer=^TData;
TWeightPointer=^TWeight;
TMarkPointer=^TMark;
//нейтралы по операции "+"
var
    DataNull:TData;
    WeightNull:TWeight;
    MarkNull:TMark;
type
    TGraph=class (TVGraph)
    private
        //конструкторы реальных данных
        procedure DataCreate;
        procedure WeightCreate;
        procedure MarkCreate;
        //деструкторы реальных данных
        procedure DataDestroy;
        procedure WeightDestroy;
        procedure MarkDestroy;
    public
        constructor Create;
```

```
//алгебры реальных данных
//создание нейтрального элемента
procedure RDataZero;
procedure RWeightZero;
procedure RMarkZero;
procedure RWeightInfinity; //создание «бесконечности»
procedure RWeightAdd; //операция «+»
function RWeightOrder:boolean; //функция порядка
end; {TGraph}
```

Для создания экземпляра реальной (дидактометрической) граф-машины требуется инициализация АГМ реальными конструкторами, деструкторами данных, сопоставляемых элементам графа, алгебраическими операциями над этими данными и отношением порядка на множестве весов.

### *Листинг 2. Конструктор ДГМ*

```
constructor TGraph.Create;
begin
  inherited Create;
  //замена абстрактных конструкторов и деструкторов реальными
  InitConstructors(DataCreate,WeightCreate,MarkCreate);
  InitDestructors(DataDestroy,WeightDestroy,MarkDestroy);
  //инициализация АГМ алгебрами элементов
  InitVDataAlgebr(RDataZero,nil);
  InitVWeightAlgebr(RWeightZero,RWeightInfinity,RWeightAdd,RWeightOrder);
  InitVMarkAlgebr(RMarkZero,nil);
end; {Create}
```

После создания экземпляра класса-наследника АГМ реальная граф-машина во всех унаследованных алгоритмах будет оперировать реальными данными и операциями над ними конкретной предметной области. В качестве примера рассмотрим описание реальной алгебры весовых коэффициентов, использующейся при инициализации дидактометрической граф-машины.

### *Листинг 3. Реализация алгебры весовых коэффициентов ДГМ*

```
procedure TGraph.WeightCreate; //конструктор
begin
  New(TWeightPointer(Regs[RegResult]));
end; {WeightCreate}

procedure TGraph.WeightDestroy; //деструктор
begin
  Dispose(TWeightPointer(Regs[RegResult]));
end; {WeightDestroy}

procedure TGraph.RWeightZero; //нейтрал по операции «+»
begin
  TWeightPointer(Regs[RegResult])^:=WeightNull;
end; {RWeightZero}

procedure TGraph.RWeightInfinity; //бесконечность
begin
  TWeightPointer(Regs[RegResult])^.Difficulty:=Infinity;
```

```
end; {RWeightInfinity}

procedure TGraph.RWeightAdd;          //операция «+»
begin
  if (TWeightPointer(Regs[1]).Difficulty/2+
      TWeightPointer(Regs[2]).Difficulty/2)>(Infinity/2) then
    TWeightPointer(Regs[RegResult]).Difficulty:=Infinity
  else
    TWeightPointer(Regs[RegResult]).Difficulty:=
      TWeightPointer(Regs[1]).Difficulty+TWeightPointer(Regs[2]).Difficulty;
end; {RWeightAdd}

function TGraph.RWeightOrder: boolean; //отношение порядка на множестве весов
begin
  if TWeightPointer(Regs[1]).Difficulty<
     TWeightPointer(Regs[2]).Difficulty then
    Result:=true
  else
    Result:=false;
end; {RWeightOrder}
```

Достаточно простая схема инициализации абстрактной машины реальными данными и операциями над ними даёт возможность использования АГМ в качестве универсального инструмента работы с моделями из различных предметных областей. Решение задач дидактики является одним из актуальных приложений абстрактной граф-машины.

### Список литературы

1. Ахо А., Хопкрофт Дж., Ульман Дж. Структуры данных и алгоритмы. – М.: Издательский дом «Вильямс», 2000. – 384 с.
2. Букачев Д.С., Мунерман В.И. О принципах реализации виртуальных алгебраических машин. //Системы компьютерной математики и их приложения: Материалы международной конференции. – Смоленск, СмолГУ, 2006. – с. 55-56.
3. Бурбаки Н. Теория множеств. – М.: Мир, 1968.
4. Левин Н.А., Мунерман В. И. Алгебраический подход к оптимизации обработки информации. – Системы и средства информатики. Спецвыпуск. Математические модели и методы информатики, стохастические технологии и системы. Москва: ИПИ РАН 2005. – с. 279-294.

### References

1. Aho A., Hopcroft Dzh., Ulman Dzh. Struktury dannyh i algoritmy. – M.: Izdatel'skij dom «Vil'jams», 2000. – 384 s.
  2. Bukachev D.S., Munerman V.I. O principah realizacii virtual'nyh algebraicheskikh mashin. //Sistemy komp'yuternoj matematiki i ih prilozhenija: Materialy mezhdunarodnoj konferencii. – Smolensk, SmolGU, 2006. – s. 55-56.
  3. Burbaki N. Teorija mnozhestv. – M.: Mir, 1968.
  4. Levin N.A., Munerman V. I. Algebraicheskij podhod k optimizacii obrabotki informacii. – Sistemy i sredstva informatiki. Specvypusk. Matematicheskie modeli i metody informatiki, stohasticheskie tehnologii i sistemy. Moskva: IPI RAN 2005. – s. 279-294.
-