

ПРИМЕНЕНИЕ ГЕНЕТИЧЕСКИХ АЛГОРИТМОВ ДЛЯ РЕШЕНИЯ МАТЕМАТИЧЕСКИХ УРАВНЕНИЙ

Ученица 10 «А» класса
МОУ «Лицей № 5 имени Ю.А.Гагарина»
Черевичная Наталья Владимировна

Учитель математики
Должикова Наталья Юрьевна

Научный консультант
Зенович Андрей Васильевич
доцент Волгоградского государственного университета

Актуальность

— простота адаптации в прикладной среде программирования и информационного преобразования электронных данных,
— широта направлений текущего применения метода (Оптимизация функций, Оптимизация запросов в базах данных, Разнообразные задачи на графах, Настройка и обучение искусственной нейронной сети, Задачи компоновки, Игровые стратегии, Теория приближений, Искусственная жизнь, Биоинформатика).

Объект исследования: генетические алгоритмы.

Предмет исследования: применение генетических алгоритмов для нахождения решения математических уравнений

Цель исследования:
найти способ решения математических уравнений с помощью генетических алгоритмов.

Задачи:

1. Уточнить понятие генетического алгоритма.
2. Рассмотреть схему генетического алгоритма.
3. Проанализировать возможности генетических алгоритмов для решения различных задач оптимизации.
4. Создать программный продукт, реализующий разработанный алгоритм, подходящий для решения широкого класса уравнений в целых числах, проверить его на тестовом примере линейного уравнения.

Генетические

алгоритмы

Это поисковые алгоритмы, основанные на механизмах натуральной селекции и генетики. Они реализуют «выживание сильнейших» среди рассмотренных структур, формируя и изменяя поисковый алгоритм на основе моделирования эволюции поиска.

Генетические

алгоритмы

- адаптивные методы поиска, которые используются для решения задач функциональной оптимизации. Они основаны на механизмах и моделях эволюции, и генетических процессов биологических алгоритмов;
- это метод перебора решений для тех задач, в которых невозможно найти решение с помощью математических формул.

Основные отличия генетических алгоритмов от традиционных методов

- работают с кодами, в которых представлен набор параметров, напрямую зависящих от аргументов целевой функции;
- для поиска генетический алгоритм использует несколько точек поискового пространства одновременно;
- в процессе работы не используют никакой дополнительной информации, что повышает скорость работы;
- использует как вероятностные правила для порождения новых точек, так и детерминированные правила для перехода от одних точек к другим.

Преимущества генетических алгоритмов

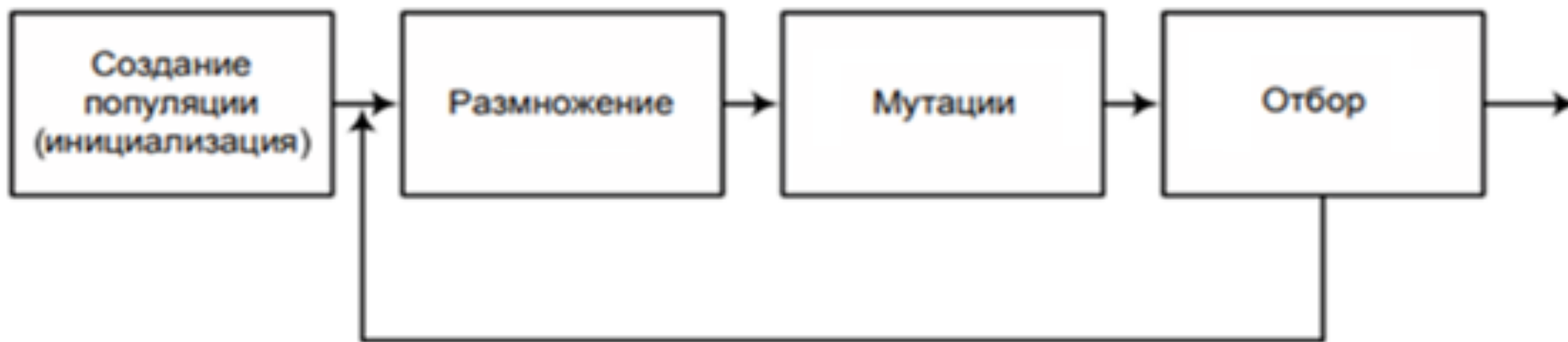
- они не требуют никакой дополнительной информации о поверхности целевой функции;
- разрывы, существующие на ней, незначительно влияют на эффективность оптимизации;
- генетические алгоритмы устойчивы к попаданию в локальные оптимумы;
- хорошо работают при решении задач многоцелевой оптимизации;
- могут быть использованы для широкого класса задач;
- просты и прозрачны в реализации; они могут быть использованы в задачах с изменяющейся средой.

Схемы Генетического Алгоритма

делится на четыре этапа:

- Создание популяции (инициализация);
- Размножение (скрещивание);
- Мутации;
- Отбор (селекция)

Общая схема генетического алгоритма



Генетические алгоритмы применяются для решения различных задач

- Оптимизация функций
- Оптимизация запросов в базах данных
- Разнообразные задачи на графах (задача коммивояжера, раскраска)
- Настройка и обучение искусственной нейронной сети
- Задачи компоновки
- Составление расписаний
- Игровые стратегии
- Теория приближений
- Искусственная жизнь
- Биоинформатика

**Написана программа на языке Паскаль,
реализующая алгоритм решения тестовой
математической задачи**

В качестве такой задачи было выбрано решение в натуральных числах уравнения $xa+yb+zc+rd=e$ с четырьмя неизвестными x, y, z, r .

Создание массива первого поколения

```
procedure InMas;  
var i, j: integer;  
begin  
    randomize;  
    nResh:=1;  
    for i:= 1 to n do  
        begin  
            for j:= 1 to 4 do  
                Mas [i, j]:= random (20)+1;  
            F[i]:= 1/ (abs (a*Mas [i, 1]+b*Mas [i, 2]+c*Mas [i, 3]+d*Mas [i, 4]-e)+1)  
        end  
    end;  
end;
```

Вычисление вероятностей скрещивания

```
procedure sum;
var i, j: integer; sum, sum1: real;
begin
  sum:=0;
  sum1:=0;
  WW[0]:=0;
  for i:= 1 to n do
    sum:=sum+F[i];
  for i:=1 to n do
    begin
      P[i]:=F[i]/sum;
      sum1:=sum1+P[i];
      WW[i]:=sum1;
    end;
end;
```

Создание будущего поколения, определение для будущих особей функции приспособленности

```

procedure budd ;
var i, j, k, rod1,rod2, gen: integer; u: real;
begin
  for i:=1 to m do
    Begin
      u:= random;
      k:=0;
      while(WN[k]<u) do k:=k+1;
      rod1:=k;
      u:= random;
      k:=0;
      while(WN[k]<u) do k:=k+1;
      rod2:=k;
      bud[6*i-5,1]:=Mas[rod1,1]; bud[6*i-5,2]:=Mas[rod2,2]; bud[6*i-5,3]:=Mas[rod2,3]; bud[6*i-5,4]:=Mas[rod2,4];
      bud[6*i-4,1]:=Mas[rod1,1]; bud[6*i-4,2]:=Mas[rod1,2]; bud[6*i-4,3]:=Mas[rod2,3]; bud[6*i-4,4]:=Mas[rod2,4];
      bud[6*i-3,1]:=Mas[rod1,1]; bud[6*i-3,2]:=Mas[rod1,2]; bud[6*i-3,3]:=Mas[rod1,3]; bud[6*i-3,4]:=Mas[rod2,4];
      bud[6*i-2,1]:=Mas[rod2,1]; bud[6*i-2,2]:=Mas[rod2,2]; bud[6*i-2,3]:=Mas[rod2,3]; bud[6*i-2,4]:=Mas[rod1,4];
      bud[6*i-1,1]:=Mas[rod2,1]; bud[6*i-1,2]:=Mas[rod2,2]; bud[6*i-1,3]:=Mas[rod1,3]; bud[6*i-1,4]:=Mas[rod1,4];
      bud[6*i,1]:=Mas[rod2,1]; bud[6*i,2]:=Mas[rod1,2]; bud[6*i,3]:=Mas[rod1,3]; bud[6*i,4]:=Mas[rod1,4];
    End;
  for i:=1 to 6*m do
    Begin
      u:=random;
      if u<ver then
        Begin
          gen:=1+random(4);
          bud[i,gen]:=1+random(20);
        End;
      W[i]:=1/(abs(a*bud[i,1]+b*bud[i,2]+c*bud[i,3]+d*bud[i,4]-e)+1);
      if W[i]>0.9999 then
        begin
          Writeln (W[i], ' ',bud[i, 1], ' ', bud[i, 2], ' ', bud[i, 3], ' ', bud[i,4]);
          for j:=1 to 4 do
            begin
              Resh[nResh,j]:= bud[i, j];
              bud[i, j]:=-1000;
            end;
          nResh:=nResh+1;
        End;
      end;
    end;
end;

```

Заключение

- Изучены генетические алгоритмы для разных задач оптимизации и реализован алгоритм, подходящий для решения широкого класса уравнений в целых числах.
- Алгоритм проверен на тестовом примере линейного уравнения.
- Проведен ряд вычислительных экспериментов с целью подбора оптимальных параметров алгоритма.
- В дальнейшем планируется реализация модификаций алгоритма для задачи коммивояжера, задачи поиска локальных минимумов функции, а также некоторых других задач.