

# Raport tema0

Stratianu Bianca-Ionela

October 2020

## 1 Introducere

### 1.1 Introducere generala

Acest raport contine: Introducere, Metode, Experimente, Concluzii si Bibliografie si analizeaza diferenta performantelor intre metoda euristica si determina in cautarea minimului pentru o anumita functie.

### 1.2 Motivatie

Multe probleme din viata reala se regasesc ca fiind probleme de baza in domeniul informaticii. De exemplu cum ajungi dintr-un loc in altul in cel mai scurt timp, ajungandu-se astfel la problema optimizarii globale.

### 1.3 Descrierea problemei

Se da: o functie

$$f : A \rightarrow R, f : A \rightarrow R$$

pentru o multime A de numere reale

Se cere: un element

$$x_0 \in A, x_0 \in A$$

pentru care

$$f(x_0) \leq f(x), \forall x \in A, f(x_0) \leq f(x), \forall x \in A$$

## 2 Metode

Pentru a testa algoritmi de aflare a minimului global am folosit functia suma patratelor:

$$f(x) = \sum_{i=1}^n ix_i^2, x_i \in [-10, 10], i = 1, 2, \dots, n$$

Minime locale: nu avem, in afara de cel global.

Minimul global este

$$x = (0.0.0.....0)$$

Graficul functiei pentru n=2:

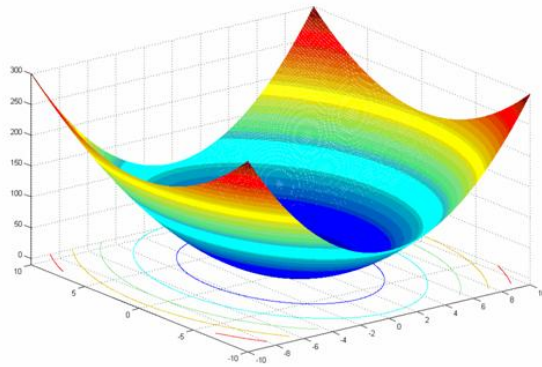


Figure 1: Suma patratelor

## 2.1 Algoritmul determinist

Algoritmul folosit este unul de cautare exhaustiva, in care cautam in puncte cat mai mici minimul.

```
determinist=function(n, nrparam) {  
  if(nrparam == 0){  
    if(f(n, x) < min){  
      min=f(n,x)  
    }  
  }  
  else{  
    i=-10;  
    while( i<= 10 ){  
      print(i)  
      x[n-nrparam+1]=i;  
      i=i+0.1  
      determinist(n, nrparam-1)  
    }  
  }  
}
```

## 2.2 Algoritmul euristic

Algoritmul euristic genereaza valori intre  $[-10, 10]$  aleatorii de 1000000 si alege valoarea minima gasita.

```
euristic=function(n){
    for(i in 1:n){
        x[i]=runif(1, -10, 10)
    }
    if( f( n , x ) < fm ){
        fm = f( n , x )
    }
    return(fm)
}
for(i in 1:1000000){
    c[i]=euristic(n)
}
min(c)
```

## 3 Experimente

Pentru efectuarea experimentului s-au utilizat:  $n=3$  (numarul de variabile), iar numarul de repetari fiind de 30 de ori.

### 3.1 Pentru algoritmul determinist

Timp	Minimul
1 min si 3 secunde	0

### 3.2 Pentru algoritmul euristic

Media solutiilor	Cea mai slaba solutie	Cea mai buna solutie	Timp
0.02747246	0.07404848	0.001135007	7 min si 10 secunde

## 4 Concluzii

Se poate observa ca timpul de executie al algoritmul eurisitic este mult mai mare decat al celui determinist, iar rezultatul este mai departe decat cel real. Pe de alta parte algoritmul determinist da solutia corecta si intr-un timp mult mai scurt. Totusi, in practica algoritmul euristic devine mai bun pentru date de intrare foarte mari.

## 5 Bibliografie

### References

- [1] <https://profs.info.uaic.ro/~eugennc/teaching/ga/>
- [2] [http://www-optima.amp.i.kyoto-u.ac.jp/member/student/hedar/Hedar\\_files/TestGO\\_files/Page674.htm](http://www-optima.amp.i.kyoto-u.ac.jp/member/student/hedar/Hedar_files/TestGO_files/Page674.htm)
- [3] <https://gitlab.com/eugennc/teaching/-/blob/master/GA/texample.pdf>
- [4] <https://gitlab.com/eugennc/teaching/-/blob/master/GA/texample.tex>
- [5] <https://www.youtube.com/watch?v=bSIAUkCD948>