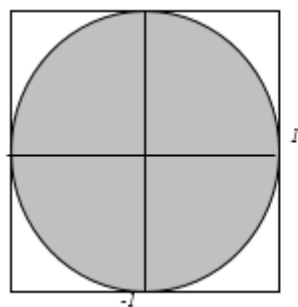


### ➤ Simulasi Monte Carlo: Menghitung nilai Pi

Metode monte carlo merupakan metode pencarian acak, tetapi dengan beberapa perbaikan yaitu tidak semua nilai pada solusi diacak ulang, tetapi dipilih satu nilai saja diantara barisan nilai solusi dan kemungkinan acak dari setiap kejadian solusi.

Misalkan: lingkaran berjari-jari 1, maka dapat dikatakan bahwa luas lingkaran tersebut adalah  $\pi$ . bila kita mengacak pasangan  $(x,y)$  dengan bilangan acak  $[0, 1]$  maka kita cukup memandang kuadran I pada bujur sangkar luar dari lingkaran berjari-jari 1. Maka kemungkinan titik  $(x, y)$  yang berada dalam lingkaran adalah :



$$P(x^2 + y^2 < 1) = \frac{A_{circle}}{A_{square}} = \frac{\pi}{4}$$

Dengan kata lain, bila dibangkitkan  $N$  pasang pasangan koordinat  $(x,y)$  menggunakan bilangan acak bernilai 0 hingga 1 maka ada  $m = \frac{Np}{4}$  yang berada dalam lingkaran pada Kuadran I. aturan ini dapat dikatakan bahwa: nilai  $\pi$  dapat dihitung dengan

$$\pi = 4 \frac{m}{n}$$

Dimana  $m$  adalah jumlah titik acak  $[0,1]$  yang masuk dalam lingkaran dan  $N$  adalah jumlah titik yang dibangkitkan.

### Listing Matlab:

```
% Memasukkan jumlah titik yang diacak
n=input('Jumlah titik acak = ');
% Mengacak n buah titik
x=rand(1,n);
y=rand(1,n);
% Menghitung jumlah titik
% yang masuk daerah lingkaran
p=(y<(1-x.^2).^0.5);
m=sum(p);
% Menghitung dan menampilkan nilai pi
mpi=4*m/n;
fprintf('pi = %1.5f\n',mpi);
```

Misalkan: input  $n=1000$

Maka  $\pi=3.19200$

### ➤ Metode Pencarian Acak: Random Walk

Metode paling sederhana dalam menyelesaikan suatu permasalahan tanpa harus melibatkan banyak operasional matematis dan struktur data yang rumit adalah metode pencarian secara acak. Dimana suatu penyelesaian diperoleh dengan mencoba-coba/*trial and error* dengan memanfaatkan bilangan acak. Misalkan untuk mencari nilai terbesar  $f(x, y) = x * \exp(-y)$  dapat dihitung dengan mengacak pasangan nilai  $(x, y)$  dan menghitung nilai  $f(x, y)$  kemudian mengacak lagi pasangan nilai  $(x, y)$  dan menghitung nilai  $f(x, y)$ . Bila  $f(x, y)$  yang baru lebih besar dicatat demikian seterusnya sehingga diperoleh  $f(x, y)$  paling besar.

#### Algoritma:

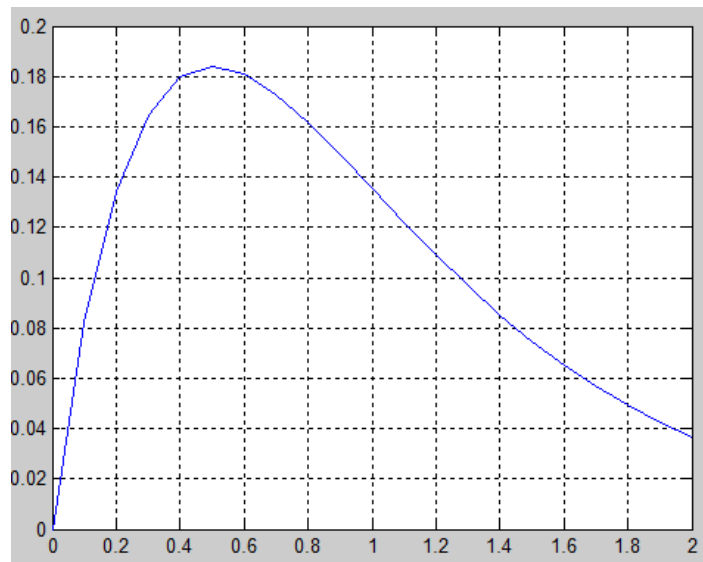
- Acak satu penyelesaian  $x$  yang mungkin.
- Hitung nilai fungsi dari penyelesaian tersebut  $f(x)$  anggap sebagai  $f\_maksimal$
- Acak lagi satu penyelesaian yang mungkin, dengan mengubah satu atau beberapa bagian kejadian pada ruang solusi.
- Hitung nilai fungsi dari penyelesaian tersebut  $f(x)$
- Bila  $f(x) > f\_maksimal$  maka  $f\_maksimal = f(x)$
- Ulangi langkah 2 sampai dinyatakan stop dengan kriteria stop yang ditentukan. Kriteria stop yang banyak dilakukan adalah jumlah iterasi atau nilai maksimal acuan.

Misalkan: mencari fungsi nilai maksimum dari fungsi  $f(x) = xe^{2x}$  dimana  $0 \leq x \leq$

2

Gambarkan fungsi:

```
x=0:0.1:2;
y=x.*exp(-2*x);
plot(x,y), grid
```



### Listing Matlab:

```
% Memasukkan jumlah iterasi
n=input('Jumlah iterasi = ');
% Pengacakan nilai awal
x=rand;
y=x*exp(-2*x);
% Random walk dengan n iterasi
for iterasi=1:n
    x1=rand;
    y1=x1*exp(-2*x1);
    if y1>y
        x=x1;
        y=y1;
    end
end
% Hasil
fprintf('Iterasi %d : ',n);
fprintf('Nilai maksimum %1.3f ',y);
fprintf('terjadi di x = %1.3f\n',x);
```

Misalkan  $N=100$ , maka:

Iterasi 100 : Nilai maksimum 0.184 terjadi di  $x = 0.506$

## **DAFTAR PUSTAKA**

Huda. 2004. Metode Monte Carlo dan Random Walks. <http://lecturer.eepis-its.edu/~huda/TK/kuliah4.ppt> (diakses 22/5/2015 Pukul 5:00 WIB).