Khoi Duong

Prof. Yang

CS483

7/27/2022

QUIZ#2

1.

We have $f(x) = 100(x_1^2 - x_2)^2 + (1 - x_1)^2$

$x_1 \in [0, 2]$ and $x_2 \in [0, 2]$

Therefore, we have number of bits:

Num. of Bit = CEILING(LOG((2-0)*10^2,2),1) = 8

We have population size M = 1.5 * 8 (genes in chromosome) = 12 ≈ 16

Then randomly take 16 chromosomes from 0 to 2^8 - 1 as follow:

For x1:

ID	Random #	Conv to bin	decoded value
1	86	1010110	0.674509804
2	139	10001011	1.090196078
3	146	10010010	1.145098039
4	201	11001001	1.576470588
5	219	11011011	1.717647059
6	158	10011110	1.239215686
7	65	1000001	0.509803922
8	35	100011	0.274509804
9	151	10010111	1.184313725
10	169	10101001	1.325490196
11	75	1001011	0.588235294
12	79	1001111	0.619607843
13	56	111000	0.439215686
14	120	1111000	0.941176471
15	146	10010010	1.145098039
16	173	10101101	1.356862745

For x2:

ID	Random #	Conv to bin	decoded value
1	71	1000111	0.556862745
2	60	111100	0.470588235
3	183	10110111	1.435294118
4	72	1001000	0.564705882
5	224	11100000	1.756862745
6	170	10101010	1.333333333
7	55	110111	0.431372549
8	250	11111010	1.960784314
9	78	1001110	0.611764706
10	238	11101110	1.866666667
11	73	1001001	0.57254902
12	118	1110110	0.925490196
13	45	101101	0.352941176
14	85	1010101	0.666666667
15	76	1001100	0.596078431
16	128	10000000	1.003921569

To get min value, fitness function is as follows:

Fitness(x) = -f(x) = $-100(x_1^2 - x_2)^2 - (1 - x_1)^2$, and then

$$\min(\text{Fitness}(x)) = \max(-f(x))$$

Roulette Wheel

ID	Decimal # (x1)	Conv to bin	Decoded value	Decimal #(x2)	Conv to bin	decoded value
1	86	1010110	0.674509804	71	1000111	0.556862745
2	139	10001011	1.090196078	60	111100	0.470588235
3	146	10010010	1.145098039	183	10110111	1.435294118
4	201	11001001	1.576470588	72	1001000	0.564705882
5	219	11011011	1.717647059	224	11100000	1.756862745
6	158	10011110	1.239215686	170	10101010	1.333333333
7	65	1000001	0.509803922	55	110111	0.431372549
8	35	100011	0.274509804	250	11111010	1.960784314
9	151	10010111	1.184313725	78	1001110	0.611764706
10	169	10101001	1.325490196	238	11101110	1.866666667
11	75	1001011	0.588235294	73	1001001	0.57254902
12	79	1001111	0.619607843	118	1110110	0.925490196
13	56	111000	0.439215686	45	101101	0.352941176
14	120	1111000	0.941176471	85	1010101	0.666666667
15	146	10010010	1.145098039	76	1001100	0.596078431
16	173	10101101	1.356862745	128	10000000	1.003921569

decoded value	Fitness Value f(x)	F(x),Cmin*	Probability	Cum. Prob.	Prob. Slots	Rand # in [0,1]	Selected Chro (x1)	Selected Chro (x2)
0.556862745	-1.14428998	0	0	0	0-0	0.47939082	11011011	11100000
0.470588235	-51.5518126	50.40752	0.044250391	0.044250391	0-0.0443	0.94760647	10101101	10000000
1.435294118	-1.559759676	0.41547	0.000364721	0.044615112	0.0443-0.0446	0.48622496	11011011	11100000
0.564705882	-369.1849441	368.0407	0.32308556	0.367700672	0.0446-0.3677	0.80372297	100011	11111010
1.756862745	-142.9469909	141.8027	0.124481914	0.492182586	0.3677-0.4922	0.30054022	11001001	1001000
1.333333333	-4.150650749	3.006361	0.002639143	0.494821728	0.4922-0.4948	0.32324126	11001001	1001000
0.431372549	-3.180574384	2.036284	0.001787558	0.496609287	0.4948-0.4966	0.71551885	100011	11111010
1.960784314	-356.0104672	354.8662	0.311520307	0.808129594	0.4966-0.8081	0.07970072	11001001	1001000
0.611764706	-62.57585969	61.43157	0.053927882	0.862057476	0.8081-0.8621	0.91259134	10010010	1001100
1.866666667	-1.310283452	0.165993	0.000145718	0.862203194	0.8621-0.8622	0.68244113	100011	11111010
0.57254902	-5.301055357	4.156765	0.003649029	0.865852223	0.8622-0.8659	0.38329227	11011011	11100000
0.925490196	-29.47518889	28.3309	0.024870362	0.890722585	0.8659-0.8907	0.85308404	10010111	1001110
0.352941176	-2.875463378	1.731173	0.001519716	0.8922423	0.8907-0.8922	0.23951387	11001001	1001000
0.666666667	-4.80597827	3.661688	0.003214424	0.895456724	0.8922-0.8955	0.7476024	100011	11111010
0.596078431	-51.16802196	50.02373	0.043913479	0.939370203	0.8955-0.9393	0.47621209	11011011	11100000
1.003921569	-70.21019044	69.0659	0.060629702	0.999999905	0.9393 - 1	0.39225874	11011011	11100000
	Sum	1139.143	0.999999905					

2.

Roulette Wheel

F(x)	Round off	Probabilities	Cum. Prob.	Prob. Slot
1.890178	1.890178	0.096525619	0.096525619	0 - 0.0965
3.924753	3.924753	0.200425153	0.296950772	0.0965 - 0.2970
3.921282	3.921282	0.200247899	0.497198672	0.2970 - 0.4972
2.395398	2.395398	0.122325662	0.619524334	0.4972 - 0.6195
1.510119	1.510119	0.077117167	0.696641501	0.6195 - 0.6966
3.480308	3.480308	0.177728704	0.874370204	0.6966 - 0.8744
0.329135	0.329135	0.01680792	0.891178124	0.8744 - 0.8912
-0.49663	0	0	0.891178124	0.8912 - 0.8912
-0.3319	0	0	0.891178124	0.8912 - 0.8912
2.130965	2.130965	0.108821876	1	0.8912 - 1
Sum	19.582138	1		

3.

For crossover rate, first we need to setup a temporarily crossover rate. For example, assume that the crossover rate $P_c = 0.8$, the crossover rate will usually be in 0.5 to 0.95. Then select a random

number in [0,1], if it is less than P_c, then the crossover will be taken. And for parent selection, we have 2 methods:

- Method 1: 1-2, 3-4, … …, (n-1)-n Example: (1-2), (3-4), … …, (15-16)
- Method 2: 1-n/2, 2-(n/2+1), … … (n/2-1)-n Example: (1-9), (2-10), … … (8-16)

For mutation rate, first we need to setup a temporarily mutation rate. For example, assume that the mutation rate is $P_m = 0.025$, as the mutation rate will usually be in 0.1 to 0.001. Then we will calculate how many bits will be mutated in the present generation by using the formula:

choromosome size * population size * mutation rate

Then we will randomly select 4 bits in all chromosomes and switch values from 1 to 0, and from 0 to 1.

In a genetic algorithm, we need to select a crossover rate and a mutation rate as we want to mimic the evolution in our program. The crossover rate will create a child generation to be the next parent generation, while the mutation rate will react as a rate of change in the environment in order to create and choose the best individual in the generation.