國立臺灣師範大學
National Taiwan Normal University

IEEE, CEC

Evolutionary Computing

An Illustrated Example

Prof. Chen-Chien Hsu
許陳鑑

Prof. Chen-Chien Hsu (許陳鑑)
Department of Electrical Engineering
National Taiwan Normal University
162 He-Ping East Road, Section 1, Taipei 10610, Taiwan.

Office: Room TA406, Tech. Building
Email: jhsu@ntnu.edu.tw
Tel: 7749-3551 (Ext. 3551)
http://isdlab.ie.ntnu.edu.tw/isdwebsite/index.html

An Illustrated Example

How do Simple Genetic Algorithms (SGA) work?

Components of a Genetic algorithm

1. A **genetic representation** for potential solutions to the problem
   基因表現　潛在の
2. A way to create an **initial population** of potential solutions
3. An **evaluation function** that plays the role of the environment, rating solutions in terms of "fitness"
4. Genetic **operators** that alter the composition of children
   改變　組成
5. Values for various **parameters** that the genetic algorithms uses (population size, probabilities of applying genetic operators)

Optimization problem:

– Objective function (cost function, fitness function)

– Without loss of generality, we assume maximization problems only.

– Maximization (default) vs minimization

$$\min f(x) = \max g(x) = \max\{-f(x)\}.$$

– Fitness (take positive values)

$$\max g(x) = \max\{g(x) + C\}.$$

↳ move all the value upper zero.

Formulation:

Now suppose we wish to maximize a function of k variables, $f(x_1, \ldots, x_k) : R^k \rightarrow R$. Suppose further that each variable x_i can take values from a domain $D_i = [a_i, b_i] \subseteq R$ and $f(x_1, \ldots, x_k) > 0$ for all $x_i \in D_i$. We wish to optimize the function f with some required precision: suppose six decimal places for the variables' values is desirable.

精確

小數點後6位

procedure evolution program 執行

begin

  $t \leftarrow 0$

  initialize $P(t)$

  evaluate $P(t)$ → 對 P(t) 評估

  while (**not** termination-condition) **do**

  begin

    $t \leftarrow t + 1$

    select $P(t)$ from $P(t-1)$

    Change alter $P(t)$

    evaluate $P(t)$

  end

end

Fig. 0.1. The structure of an evolution program

Genetic representation

- Coding and decoding
- Binary string of length m_i
    - Ex: 6 decimal places
    - Choose a smallest integer such that

upper bound

$$(b_i - a_i) \cdot 10^6 \leq 2^{m_i} - 1.$$

lower bound

$$x_i = a_i + decimal(1001...001_2) \cdot \frac{b_i - a_i}{2^{m_i} - 1}$$

- Chromosome (染色體) is represented by

  a binary string of length: $m = \sum_{i=1}^{k} m_i;$

m1	m2	m3

Initial population of potential solutions

- Randomly generated
- Knowledge of potential optima (problem specific knowledge) for incorporation into the initial population

合併

Selection process (roulette wheel)

Scaling mechanism:

- Calculate the fitness value $eval(\boldsymbol{v}_i)$ for each chromosome $\boldsymbol{v}_i$ ($i = 1, \ldots, pop\_size$).

- Find the total fitness of the population

$$F = \sum_{i=1}^{pop\_size} eval(\boldsymbol{v}_i).$$

- Calculate the probability of a selection p_i for each chromosome $\boldsymbol{v}_i$ ($i = 1, \ldots, pop\_size$):

$$p_i = eval(\boldsymbol{v}_i)/F.$$

- Calculate a cumulative probability q_i for each chromosome $\boldsymbol{v}_i$ ($i = 1, \ldots, pop\_size$): 累計の

$$q_i = \sum_{j=1}^{i} p_j.$$

Selection process:

- Generate a random (float) number r from the range $[0..1]$.
- If $r < q_1$ then select the first chromosome ($\boldsymbol{v}_1$); otherwise select the i-th chromosome $\boldsymbol{v}_i$ ($2 \le i \le pop\_size$) such that $q_{i-1} < r \le q_i$.

- Schema theorem:

Some chromosomes would be selected more than once: best chromosomes get more copies, the average stay even, the worst die off.

- What is wrong with that???

Variation operators:

- **Recombination (Crossover)**
- **Mutation**

Recombination (Crossover) operator:

- Probability of crossover: p_c
- No of chromosomes which undergo crossover operation: $p_c \cdot pop\_size$

- Generate a random (float) number r from the range $[0..1]$;
- If $r < p_c$, select given chromosome for crossover.

$$\begin{matrix}(b_1 b_2 \ldots b_{pos} | b_{pos+1} \ldots b_m) \\ (c_1 c_2 \ldots c_{pos} | c_{pos+1} \ldots c_m)\end{matrix} \longrightarrow \begin{matrix}(b_1 b_2 \ldots b_{pos} | c_{pos+1} \ldots c_m) \\ (c_1 c_2 \ldots c_{pos} | b_{pos+1} \ldots b_m).\end{matrix}$$

Mutation operator

- Probability of mutation: p_m
- No of mutated bits: $p_m \cdot m \cdot pop\_size.$

- Generate a random (float) number r from the range $[0..1]$;
- If $r < p_m$, mutate the bit.

Selection ➔ crossover ➔ mutation

Illustrated example:

max $f(x_1, x_2) = 21.5 + x_1 \cdot \sin(4\pi x_1) + x_2 \cdot \sin(20\pi x_2)$

where $\underset{a_1}{-3.0} \le x_1 \le \underset{b_1}{12.1}$ and $\underset{a_2}{4.1} \le x_2 \le \underset{b_2}{5.8}$.

GA parameters:

pop_size=20 (樣本數)

pc=0.25

pm=0.01

4 decimal places (precision) → 取小數點後四位

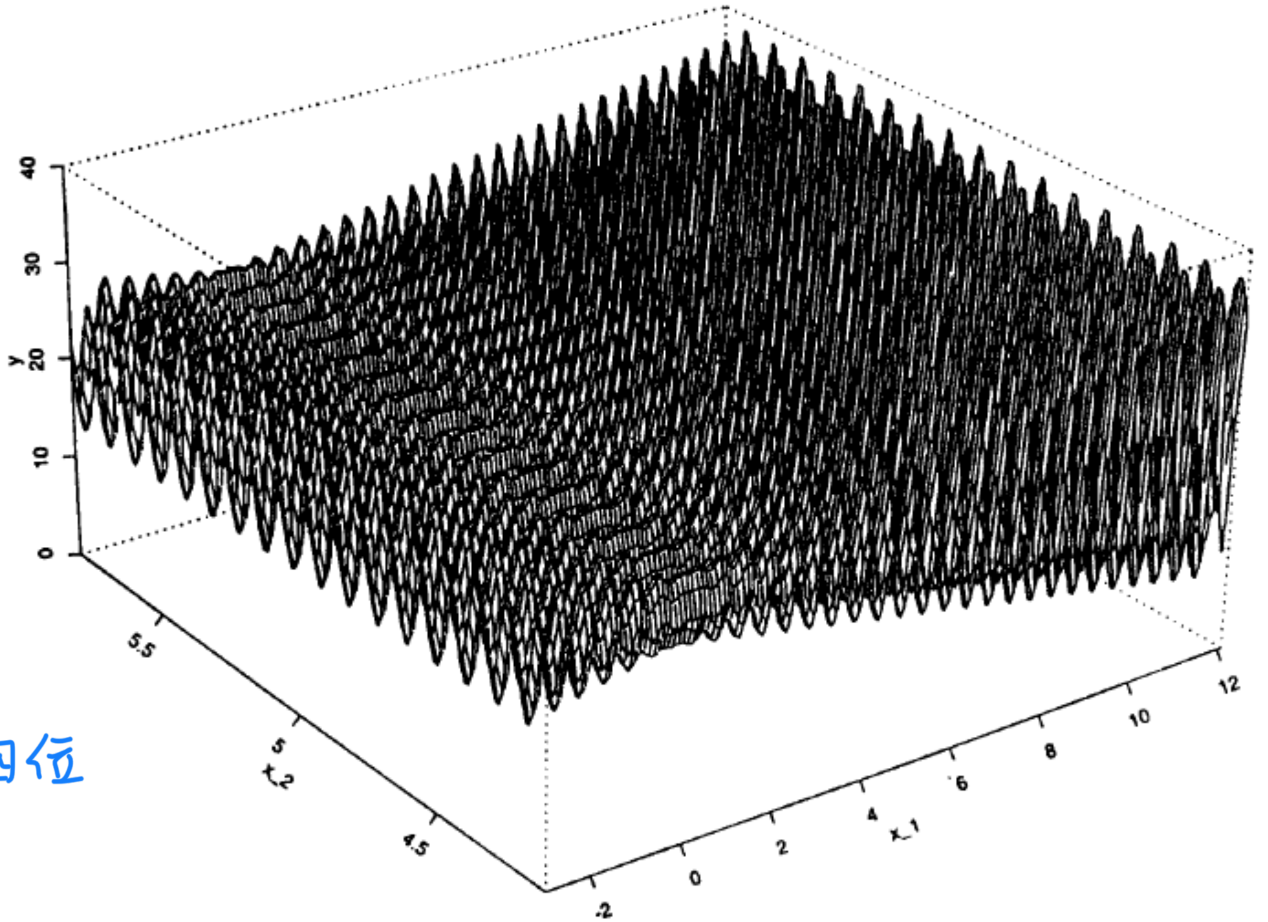

Fig. 2.1. Graph of the function $f(x_1, x_2) = 21.5 + x_1 \cdot \sin(4\pi x_1) + x_2 \cdot \sin(20\pi x_2)$

Representation: bit length

x_1: $2^{17} < 151000 \leq 2^{18}$. → 需要18个bits

x_2: $2^{14} < 17000 \leq 2^{15}$. → 需要15个bits

→ $m=18+15=33$

Example: $(\overset{x_1}{010001001011010000}\,|\,\overset{x_2}{111110010100010})$.

$$x_1 = \overset{a_1}{-3.0} + decimal(010001001011010000_2) \cdot \frac{\overset{b_1 - a_1}{12.1-(-3.0)}}{2^{18}-1} = 1.052426.$$

$$x_2 = \overset{a_2}{4.1} + decimal(111110010100010_2) \cdot \frac{\overset{b_2 - a_2}{5.8-4.1}}{2^{15}-1} = 5.755330.$$

$$\langle x_1, x_2 \rangle = \langle 1.052426, 5.755330 \rangle$$

→ $f(1.052426, 5.755330) = 20.252640.$

$f(x_1, x_2) = 21.5 + x_1 \cdot \sin(4\pi x_1) + x_2 \cdot \sin(20\pi x_2)$

Initialization:

$$
\begin{aligned}
\boldsymbol{v}_1 &= (100110100000001111111010011011111)\\
\boldsymbol{v}_2 &= (111000100100110111001010100011010)\\
\boldsymbol{v}_3 &= (000010000011001000001010111011101)\\
\boldsymbol{v}_4 &= (100011000101101001111000001110010)\\
\boldsymbol{v}_5 &= (000111011001010011010111111000101)\\
\boldsymbol{v}_6 &= (000101000010010101001010111111011)\\
\boldsymbol{v}_7 &= (001000100000110101111011011111011)\\
\boldsymbol{v}_8 &= (100001100001110100010110101100111)\\
\boldsymbol{v}_9 &= (010000000101100010110000001111100)\\
\boldsymbol{v}_{10} &= (000001111000110000011010000111011)\\
\boldsymbol{v}_{11} &= (011001111110110101100001101111000)\\
\boldsymbol{v}_{12} &= (110100010111101101000101010000000)\\
\boldsymbol{v}_{13} &= (111011111010001000110000001000110)\\
\boldsymbol{v}_{14} &= (010010011000001010100111100101001)\\
\boldsymbol{v}_{15} &= (111011101101110000100011111011110)\\
\boldsymbol{v}_{16} &= (110011110000011111100001101001011)\\
\boldsymbol{v}_{17} &= (011010111111001111010001101111101)\\
\boldsymbol{v}_{18} &= (011101000000001110100111110101101)\\
\boldsymbol{v}_{19} &= (000101010011111111110000110001100)\\
\boldsymbol{v}_{20} &= (101110010110011110011000101111110)
\end{aligned}
$$

Evaluation:

- Decode
- Calculate fitness values

$$\begin{aligned}
eval(\boldsymbol{v}_1) &= f(6.084492, 5.652242) = 26.019600 \\
eval(\boldsymbol{v}_2) &= f(10.348434, 4.380264) = \boxed{7.580015} \\
eval(\boldsymbol{v}_3) &= f(-2.516603, 4.390381) = 19.526329 \\
eval(\boldsymbol{v}_4) &= f(5.278638, 5.593460) = 17.406725 \\
eval(\boldsymbol{v}_5) &= f(-1.255173, 4.734458) = 25.341160 \\
eval(\boldsymbol{v}_6) &= f(-1.811725, 4.391937) = 18.100417 \\
eval(\boldsymbol{v}_7) &= f(-0.991471, 5.680258) = 16.020812 \\
eval(\boldsymbol{v}_8) &= f(4.910618, 4.703018) = 17.959701 \\
eval(\boldsymbol{v}_9) &= f(0.795406, 5.381472) = 16.127799 \\
eval(\boldsymbol{v}_{10}) &= f(-2.554851, 4.793707) = 21.278435 \\
eval(\boldsymbol{v}_{11}) &= f(3.130078, 4.996097) = 23.410669 \\
eval(\boldsymbol{v}_{12}) &= f(9.356179, 4.239457) = 15.011619 \\
eval(\boldsymbol{v}_{13}) &= f(11.134646, 5.378671) = 27.316702 \\
eval(\boldsymbol{v}_{14}) &= f(1.335944, 5.151378) = 19.876294 \\
eval(\boldsymbol{v}_{15}) &= f(11.089025, 5.054515) = \boxed{30.060205} \\
eval(\boldsymbol{v}_{16}) &= f(9.211598, 4.993762) = 23.867227 \\
eval(\boldsymbol{v}_{17}) &= f(3.367514, 4.571343) = 13.696165 \\
eval(\boldsymbol{v}_{18}) &= f(3.843020, 5.158226) = 15.414128 \\
eval(\boldsymbol{v}_{19}) &= f(-1.746635, 5.395584) = 20.095903 \\
eval(\boldsymbol{v}_{20}) &= f(7.935998, 4.757338) = 13.666916
\end{aligned}$$

minimized → $eval(\boldsymbol{v}_2)$

maximized → $eval(\boldsymbol{v}_{15})$

➔ $F = \sum_{i=1}^{20} eval(\boldsymbol{v}_i) = 387.776822.$

- The probability of a selection p_i for each chromosome v_i

$$
\begin{aligned}
p_1 &= eval(\boldsymbol{v}_1)/F = 0.067099 & p_2 &= eval(\boldsymbol{v}_2)/F = 0.019547 \\
p_3 &= eval(\boldsymbol{v}_3)/F = 0.050355 & p_4 &= eval(\boldsymbol{v}_4)/F = 0.044889 \\
p_5 &= eval(\boldsymbol{v}_5)/F = 0.065350 & p_6 &= eval(\boldsymbol{v}_6)/F = 0.046677 \\
p_7 &= eval(\boldsymbol{v}_7)/F = 0.041315 & p_8 &= eval(\boldsymbol{v}_8)/F = 0.046315 \\
p_9 &= eval(\boldsymbol{v}_9)/F = 0.041590 & p_{10} &= eval(\boldsymbol{v}_{10})/F = 0.054873 \\
p_{11} &= eval(\boldsymbol{v}_{11})/F = 0.060372 & p_{12} &= eval(\boldsymbol{v}_{12})/F = 0.038712 \\
p_{13} &= eval(\boldsymbol{v}_{13})/F = 0.070444 & p_{14} &= eval(\boldsymbol{v}_{14})/F = 0.051257 \\
p_{15} &= eval(\boldsymbol{v}_{15})/F = \boxed{0.077519} & p_{16} &= eval(\boldsymbol{v}_{16})/F = 0.061549 \\
p_{17} &= eval(\boldsymbol{v}_{17})/F = 0.035320 & p_{18} &= eval(\boldsymbol{v}_{18})/F = 0.039750 \\
p_{19} &= eval(\boldsymbol{v}_{19})/F = 0.051823 & p_{20} &= eval(\boldsymbol{v}_{20})/F = 0.035244
\end{aligned}
$$

- Cumulative probabilities q_i for each chromosome v_i

$$
\begin{aligned}
q_1 &\overset{=P_1}{=} 0.067099 & q_2 &\overset{=q_1+P_2}{=} 0.086647 & q_3 &\overset{=q_2+P_3}{=} 0.137001 & q_4 &\overset{=q_3+P_4}{=} 0.181890 \\
q_5 &= 0.247240 & q_6 &= 0.293917 & q_7 &= 0.335232 & q_8 &= 0.381546 \\
q_9 &= 0.423137 & q_{10} &= 0.478009 & q_{11} &= 0.538381 & q_{12} &= 0.577093 \\
q_{13} &= 0.647537 & q_{14} &= 0.698794 & q_{15} &= 0.776314 & q_{16} &= 0.837863 \\
q_{17} &= 0.873182 & q_{18} &= 0.912932 & q_{19} &= 0.964756 & q_{20} &= 1.000000
\end{aligned}
$$

- Spin the roulette wheel 20 times

$\boxed{0.513870}$	0.175741	0.308652	0.534534	0.947628
0.171736	0.702231	0.226431	0.494773	0.424720
0.703899	0.389647	0.277226	0.368071	0.983437
0.005398	0.765682	0.646473	0.767139	0.780237

$q_{10} < 0.513870 < q_{11}$

➔ v_{11}

⇒ Selection Process

New population:

$$v'_1 = (011001111110110101100001101111000)\ (v_{11})$$
$$v'_2 = (100011000101101001111000001110010)\ (v_4)$$
$$v'_3 = (001000100000110101111011011111011)\ (v_7)$$
$$v'_4 = (011001111110110101100001101111000)\ (v_{11})$$
$$v'_5 = (000101010011111111110000110001100)\ (v_{19})$$
$$v'_6 = (100011000101101001111000001110010)\ (v_4)$$
$$v'_7 = (111011101101110000100011111011110)\ (v_{15})$$ *
$$v'_8 = (000111011001010011010111111000101)\ (v_5)$$
$$v'_9 = (011001111110110101100001101111000)\ (v_{11})$$
$$v'_{10} = (000010000011001000001010111011101)\ (v_3)$$
$$v'_{11} = (111011101101110000100011111011110)\ (v_{15})$$ *
$$v'_{12} = (010000000101100010110000001111100)\ (v_9)$$
$$v'_{13} = (000101000010010101001010111111011)\ (v_6)$$
$$v'_{14} = (100001100001110100010110101100111)\ (v_8)$$
$$v'_{15} = (101110010110011110011000101111110)\ (v_{20})$$
$$v'_{16} = (100110100000000111111010011011111)\ (v_1)$$
$$v'_{17} = (000001111000110000011010000111011)\ (v_{10})$$
$$v'_{18} = (111011111010001000110000001000110)\ (v_{13})$$
$$v'_{19} = (111011101101110000100011111011110)\ (v_{15})$$ *
$$v'_{20} = (110011110000011111100001101001011)\ (v_{16})$$

V_{15} breeds!

V_2 dies out!

Crossover (pc=0.25)

0.822951	$\boxed{0.151932}$	0.625477	0.314685	0.346901
0.917204	0.519760	0.401154	0.606758	0.785402
$\boxed{0.031523}$	0.869921	$\boxed{0.166525}$	0.674520	0.758400
0.581893	0.389248	$\boxed{0.200232}$	0.355635	0.826927

$$\begin{aligned} \boldsymbol{v}_2' &= (100011000|1011010011110000011100 10) \\ \boldsymbol{v}_{11}' &= (111011101|10111000010001111011110) \end{aligned}$$

crossover

pos=9

$$\begin{aligned} \boldsymbol{v}_2'' &= (100011000|10111000010001111011110) \\ \boldsymbol{v}_{11}'' &= (111011101|10110100111100000111001 0). \end{aligned}$$

$$\begin{aligned} \boldsymbol{v}_{13}' &= (00010100001001010100|1010111111011) \\ \boldsymbol{v}_{18}' &= (11101111101000100011|0000001000110) \end{aligned}$$

pos=20

$$\begin{aligned} \boldsymbol{v}_{13}'' &= (00010100001001010100|0000001000110) \\ \boldsymbol{v}_{18}'' &= (11101111101000100011|1010111111011). \end{aligned}$$

Current population:

$$
\begin{aligned}
v'_1 &= (011001111110110101100001101111000)\\
&\boxed{v''_2 = (100011000101110000100011111011110)}\\
v'_3 &= (001000100000110101111011011111011)\\
v'_4 &= (011001111110110101100001101111000)\\
v'_5 &= (000101010011111111110000110001100)\\
v'_6 &= (100011000101101001111000001110010)\\
v'_7 &= (111011101101110000100011111011110)\\
v'_8 &= (000111011001010011010111111000101)\\
v'_9 &= (011001111110110101100001101111000)\\
v'_{10} &= (000010000011001000001010111011101)\\
&\boxed{v''_{11} = (111011101101101001111000001110010)}\\
v'_{12} &= (010000000101100010110000001111100)\\
&\boxed{v''_{13} = (000101000010010101000000001000110)}\\
v'_{14} &= (100001100001110100010110101100111)\\
v'_{15} &= (101110010110011110011000101111110)\\
v'_{16} &= (100110100000001111111010011011111)\\
v'_{17} &= (000001111000110000011010000111011)\\
&\boxed{v''_{18} = (111011111010001000111010111111011)}\\
v'_{19} &= (111011101101110000100011111011110)\\
v'_{20} &= (110011110000011111100001101001011)
\end{aligned}
$$

Mutation (pm=0.01)

Bit position	Random number
112	0.000213
349	0.009945
418	0.008809
429	0.005425
602	0.002836

Bit position	Chromosome number	Bit number within chromosome
112	4	13
349	11	19
418	13	22
429	13	33
602	19	8

Current population:

$$
\begin{aligned}
v_1 &= (011001111110110101100001101111000)\\
v_2 &= (100011000101110000100011111011110)\\
v_3 &= (001000100000110101111011011111011)\\
v_4 &= (011001111110\boxed{0}10101100001101111000)\\
v_5 &= (000101010011111111110000110001100)\\
v_6 &= (100011000101101001111000001110010)\\
v_7 &= (111011101101110000100011111011110)\\
v_8 &= (000111011001010011010111111000101)\\
v_9 &= (011001111110110101100001101111000)\\
v_{10} &= (000010000011001000001010111011101)\\
v_{11} &= (111011101101101001\boxed{0}11000001110010)\\
v_{12} &= (010000000101100010110000001111100)\\
v_{13} &= (000101000010010101000\boxed{1}0000100011\boxed{1})\\
v_{14} &= (100001100001110100010110101100111)\\
v_{15} &= (101110010110011110011000101111110)\\
v_{16} &= (100110100000001111111010011011111)\\
v_{17} &= (000001111000110000011010000111011)\\
v_{18} &= (111011111010001000111010111111011)\\
v_{19} &= (11101110\boxed{0}1011100001000111111011110)\\
v_{20} &= (110011110000011111100001101001011)
\end{aligned}
$$

mutation $\Rightarrow$ $0 \to 1$, $1 \to 0$

Evaluation:

$$
\begin{aligned}
eval(\boldsymbol{v}_1) &= f(3.130078, 4.996097) = 23.410669\\
eval(\boldsymbol{v}_2) &= f(5.279042, 5.054515) = 18.201083\\
eval(\boldsymbol{v}_3) &= f(-0.991471, 5.680258) = 16.020812\\
eval(\boldsymbol{v}_4) &= f(3.128235, 4.996097) = 23.412613\\
eval(\boldsymbol{v}_5) &= f(-1.746635, 5.395584) = 20.095903\\
eval(\boldsymbol{v}_6) &= f(5.278638, 5.593460) = 17.406725\\
eval(\boldsymbol{v}_7) &= f(11.089025, 5.054515) = 30.060205\\
eval(\boldsymbol{v}_8) &= f(-1.255173, 4.734458) = 25.341160\\
eval(\boldsymbol{v}_9) &= f(3.130078, 4.996097) = 23.410669\\
eval(\boldsymbol{v}_{10}) &= f(-2.516603, 4.390381) = 19.526329\\
\Aboxed{eval(\boldsymbol{v}_{11}) &= f(11.088621, 4.743434) = 33.351874}\\
eval(\boldsymbol{v}_{12}) &= f(0.795406, 5.381472) = 16.127799\\
eval(\boldsymbol{v}_{13}) &= f(-1.811725, 4.209937) = 22.692462\\
eval(\boldsymbol{v}_{14}) &= f(4.910618, 4.703018) = 17.959701\\
eval(\boldsymbol{v}_{15}) &= f(7.935998, 4.757338) = 13.666916\\
eval(\boldsymbol{v}_{16}) &= f(6.084492, 5.652242) = 26.019600\\
eval(\boldsymbol{v}_{17}) &= f(-2.554851, 4.793707) = 21.278435\\
eval(\boldsymbol{v}_{18}) &= f(11.134646, 5.666976) = 27.591064\\
eval(\boldsymbol{v}_{19}) &= f(11.059532, 5.054515) = 27.608441\\
eval(\boldsymbol{v}_{20}) &= f(9.211598, 4.993762) = 23.867227
\end{aligned}
$$

If there is a super gene !?

ex: $eval(V_{15}) = f(\quad) = 10000$

⇒ all the result will be V_{15}

V_{else} will be dies out.

➔ Total fitness=447.049688 (vs 387.77)

Evolution of 1000 generations

$v_1 = (111011110110011011100101010111011)$
$v_2 = (111001100110000100010101010111000)$
$v_3 = (111011110111011011100101010111011)$
$v_4 = (111001100010000110000101010111001)$
$v_5 = (111011110111011011100101010111011)$
$v_6 = (111001100110000100000100010100001)$
$v_7 = (110101100010010010001100010110000)$
$v_8 = (111101100010001010001101010010001)$
$v_9 = (111001100010010010001100010110001)$
$v_{10} = (111011110111011011100101010111011)$
$v_{11} = (110101100000010010001100010110000)$
$v_{12} = (110101100010010010001100010110001)$
$v_{13} = (111011110111011011100101010111011)$
$v_{14} = (111001100110000100000101010111011)$
$v_{15} = (111001101010111001010100110110001)$
$v_{16} = (111001100110000101000100010100001)$
$v_{17} = (111001100110000100000101010111011)$
$v_{18} = (111001100110000100000101010111001)$
$v_{19} = (111101100010001010001110000010001)$
$v_{20} = (111001100110000100000101010111001)$

near optima

$eval(v_1) = f(11.120940, 5.092514) = 30.298543$
$eval(v_2) = f(10.588756, 4.667358) = 26.869724$
$eval(v_3) = f(11.124627, 5.092514) = 30.316575$
$eval(v_4) = f(10.574125, 4.242410) = 31.933120$
$eval(v_5) = f(11.124627, 5.092514) = 30.316575$
$eval(v_6) = f(10.588756, 4.214603) = 34.356125$
$eval(v_7) = f(9.631066, 4.427881) = 35.458636$
$eval(v_8) = f(11.518106, 4.452835) = 23.309078$
$eval(v_9) = f(10.574816, 4.427933) = 34.393820$
$eval(v_{10}) = f(11.124627, 5.092514) = 30.316575$
$\boxed{eval(v_{11}) = f(9.623693, 4.427881) = 35.477938}$
$eval(v_{12}) = f(9.631066, 4.427933) = 35.456066$
$eval(v_{13}) = f(11.124627, 5.092514) = 30.316575$
$eval(v_{14}) = f(10.588756, 4.242514) = 32.932098$
$eval(v_{15}) = f(10.606555, 4.653714) = 30.746768$
$eval(v_{16}) = f(10.588814, 4.214603) = 34.359545$
$eval(v_{17}) = f(10.588756, 4.242514) = 32.932098$
$eval(v_{18}) = f(10.588756, 4.242410) = 32.956664$
$eval(v_{19}) = f(11.518106, 4.472757) = 19.669670$
$eval(v_{20}) = f(10.588756, 4.242410) = 32.956664$

Observations:

- Best fitness of chromosomes ever derived:

  38.827553, at generation 396.

- Stochastic errors of sampling
  隨机の

➔Elitism is preferred, i.e., store the best ever individual 傑出の