Chapter 3

GAs: Why Do They Work?

Schema Theorem

- **SGA's features:**
  - binary encoding
  - fitness-proportional selection
  - one-point crossover
  - bit-flipping with fixed probability mutation

- Schema Theory assures that the search process carried out by a genetic algorithm develops in the direction of the proliferation of the most successful schemas/schemata, concomitantly with destroying unsuccessful schemas.

Definitions

- A **schema** represents all strings, which match it on all positions other than "*". The number of "*" in a schema determines the number of strings matched by the schema.

  - There are 3^m schemas for a *m*-length string.
  - A schema matches 2^r strings. (*r* : # of "*")
  - A string is matched by 2^m schemas. (*m*: the length of the string)

- $o(S)$, the order of a schema S, is the number of 0 and 1 positions.
- $\delta(S)$, the defining length of the schema S, is the distance between the first and the last fixed string positions.
- $\xi(S,t)$ is the number of strings in a population at the time *t*, matched by schema S.
- $eval(S,t)$ is the average fitness of all strings in the population at the time *t*, matched by the schema S.
- $\overline{F(t)}$, the average performance (or average fitness) of the entire population at the time *t*.

- $\xi(S,t)$ is the number of strings in a population at the time t, matched by schema S.

$$S = (****111**************************)$$

$$
\begin{aligned}
v_1 &= (100110100000001111111010011011111)\\
v_2 &= (111000100100110111001010100011010)\\
v_3 &= (000010000011001000001010111011101)\\
v_4 &= (100011000101101001111000001110010)\\
v_5 &= (000111011001010011010111111000101)\\
v_6 &= (000101000010010101001010111111011)\\
v_7 &= (001000100000110101111011011111011)\\
v_8 &= (100001100001110100010110101100111)\\
v_9 &= (010000000101100010110000001111100)\\
v_{10} &= (000001111000110000011010000111011)\\
v_{11} &= (011001111110110101100001101111000)\\
v_{12} &= (110100010111101101000101010000000)\\
v_{13} &= (1110\boxed{111}11010001000110000001000110)\\
v_{14} &= (010010011000001010100111100101001)\\
v_{15} &= (1110\boxed{111}01101110000100011111011110)\\
v_{16} &= (1100\boxed{111}10000011111100001101001011)\\
v_{17} &= (011010111111001111010001101111101)\\
v_{18} &= (011101000000001110100111110101101)\\
v_{19} &= (000101010011111111110000110001100)\\
v_{20} &= (101110010110011110011000101111110)
\end{aligned}
$$

- $eval(S,t)$ is the average fitness of all strings in the population at the time *t*, matched by the schema S.

$$eval(S,t) = \sum_{j=1}^{\rho} eval(v_{t_j}) / \rho$$

$$where \qquad \rho = \xi(S,t)$$

- $\overline{F(t)}$, the average performance (or average fitness) of the entire population at the time *t*.

$$\overline{F(t)} = \frac{F(t)}{populatio\_size}$$

$$where \qquad F(t) = \sum_{j=1}^{pop\_size} eval(v_{t_j})$$

Effects on Schema Dynamics

1. Effect of selection
2. Effect of crossover
   - Effect of selection & crossover
3. Effect of mutation
   - Effect of selection & crossover & mutation

S = (****111**************************)

t

$$
\begin{aligned}
v_1 &= (100110100000001111111010011011111)\\
v_2 &= (111000100100110111001010100011010)\\
v_3 &= (000010000011001000001010111011101)\\
v_4 &= (100011000101101001111000001110010)\\
v_5 &= (000111011001010011010111111000101)\\
v_6 &= (000101000010010101001010111111011)\\
v_7 &= (001000100000110101111011011111011)\\
v_8 &= (100001100001110100010110101100111)\\
v_9 &= (010000000101100010110000001111100)\\
v_{10} &= (000001111000110000011010000111011)\\
v_{11} &= (011001111110110101100001101111000)\\
v_{12} &= (110100010111101101000101010000000)\\
v_{13} &= (1110\boxed{111}11010001000110000001000110)\\
v_{14} &= (010010011000001010100111100101001)\\
v_{15} &= (1110\boxed{111}01101110000100011111011110)\\
v_{16} &= (1100\boxed{111}10000011111100001101001011)\\
v_{17} &= (011010111111001111010001101111101)\\
v_{18} &= (011101000000001110100111110101101)\\
v_{19} &= (000101010011111111110000110001100)\\
v_{20} &= (101110010110011110011000101111110)
\end{aligned}
$$

t+1

$$
\begin{aligned}
v'_1 &= (011001111110110101100001101111000) \; (v_{11})\\
v'_2 &= (100011000101101001111000001110010) \; (v_4)\\
v'_3 &= (001000100000110101111011011111011) \; (v_7)\\
v'_4 &= (011001111110110101100001101111000) \; (v_{11})\\
v'_5 &= (000101010011111111110000110001100) \; (v_{19})\\
v'_6 &= (100011000101101001111000001110010) \; (v_4)\\
v'_7 &= (1110\boxed{111}01101110000100011111011110) \; (v_{15})\\
v'_8 &= (000111011001010011010111111000101) \; (v_5)\\
v'_9 &= (011001111110110101100001101111000) \; (v_{11})\\
v'_{10} &= (000010000011001000001010111011101) \; (v_3)\\
v'_{11} &= (1110\boxed{111}01101110000100011111011110) \; (v_{15})\\
v'_{12} &= (010000000101100010110000001111100) \; (v_9)\\
v'_{13} &= (000101000010010101001010111111011) \; (v_6)\\
v'_{14} &= (100001100001110100010110101100111) \; (v_8)\\
v'_{15} &= (101110010110011110011000101111110) \; (v_{20})\\
v'_{16} &= (111001100110000101000100010100001) \; (v_1)\\
v'_{17} &= (111001100110000100000101010111011) \; (v_{10})\\
v'_{18} &= (1110\boxed{111}11010001000110000001000110) \; (v_{13})\\
v'_{19} &= (1110\boxed{111}01101110000100011111011110) \; (v_{15})\\
v'_{20} &= (1100\boxed{111}10000011111100001101001011) \; (v_{16})
\end{aligned}
$$

Effects on Schema Dynamics

1. Effect of <u>selection</u>

$$\zeta(S,t+1) = \zeta(S,t) \cdot eval(s,t) / \overline{F(t)}$$
$$= \zeta(S,t) \cdot (1+\varepsilon)$$

Conclusion:

1. $\varepsilon > 0$ **above average** schema; exponentially increasing

   (An above average schema, S, receives an exponentially increasing number of strings in the next generations.)

2. $\varepsilon = 0$ **average schema**; staying the same level

   (An average schema, S, stays on the same level.)

3. $\varepsilon < 0$ **below average** schema; exponentially decreasing

   (A below average schema, S, receives an decreasing number of strings in the next generations.)

Effects on Schema Dynamics

2. Effect of selection & crossover

$$\zeta(S,t+1) \geq \left[\zeta(S,t)\cdot eval(S,t)/\overline{F(t)}\right]\left[1-p_c\frac{\delta(S)}{m-1}\right]$$
$$=\zeta(S,t)\cdot(1+\varepsilon)$$

Conclusion

The **short** , **above-average** schema will receive an exponentially increasing number of strings in the next generation.

Effects on Schema Dynamics

3. Effect of selection & crossover & mutation

$$\begin{aligned}\zeta(S,t+1) &\geq \left[\zeta(S,t)\cdot eval(S,t)/\overline{F(t)}\right]\left[1-p_c\frac{\delta(S)}{m-1}\right]\left[1-o(S)p_m\right]\\ &\cong \left[\zeta(S,t)\cdot eval(S,t)/\overline{F(t)}\right]\left[1-\rho_c\frac{\delta(S)}{m-1}-o(S)\rho_m\right]\\ &= \zeta(S,t)\cdot(1+\varepsilon)\end{aligned}$$

Conclusion :

The **above-average** schemata with **short** defining length and **low-order** will sampled at exponentially increased rate.

Conclusion

Theorem 1 (Schema Theorem):

Short, low-order, above-average schemata receive exponentially increasing trials in subsequent generation of a genetic algorithm.

Hypothesis 1 (Building Block Hypothesis):

A genetic algorithm seeks near-optimal performance through the juxtaposition of short, low-order, high-performance schemata, called the building blocks.