

8-Queen Problem

- * We have to place 8 Queen in a chessboard s.t 2 Queen can't check to each other.

Solⁿ: -

* 92 distinct
Solⁿ are
available.

64 C₈

Backtracking method.

→ 12 only symmetrical answer is available.

Best Solⁿ → No Violation of Cond's

Best Rep. → Permutation Based.

(1, 2, 3, ..., 8). - Decoding Mechanism.

↓
1st column
1 row

$$f(a) = -g(b)$$

1	2	3	4	5	6	7	8
2	1	3	4	6	7	8	5
3	4	1	2	5	7	8	6
4	3	5	1	2	6	8	7
5	6	7	8	1	2	3	4
6	7	8	5	2	3	4	1
7	8	5	6	3	4	1	2
8	5	6	7	4	1	2	3

quality .

$$g(b) = 7$$

$$f(x) = \frac{1}{g(b) + 1}$$

Fitness f_n associated with min^m violation.



* Mathematical foundation of Genetics Algorithm

* Binary Representation

set of Alphabet = {0, 1}.

Schema

Plural = Schemata.

↳ is the way to computing. Similarity b/w the string (Individual).

→ Template to represent all string in a particular way.

{0, 1, *, .. or #}

+ don't care symbol.

Suppose, chromosome/String (Individual) length is

01 → 2⁰P₁

→ 2^l → Search space

+ distinct string possible.

* Schemata available = 3^l.

0P₁ 0P₂
1 P₁ P₂

→ 3^l → Search Space

→ Complexity is going to increase when we are incorporating Schema concept in G.A.

$$3^l > 2^l$$



- * Generational or String pattern can be seen as Schema.

Suppose, length of Schema & String is 4

Schema \rightarrow *001
 \downarrow
 \hookrightarrow 0001 & 1001

No. of don't care

Q. Cardinality of alphabet set is m & length of string is l . How many strings available?

$$\text{Schema} = (m+1)^l \quad \begin{matrix} \overbrace{}^{\text{m+1}} & \overbrace{}^{\text{m+1}} \\ + & + \end{matrix} \quad l.$$

Phase-2 :-

Q. What is appropriate way for Incorporation?

One Way - Take one string & check individual string
Check family of Schemata. In which Schema
it will match & do the same process for
second string & so on.

Q. How many Schema matches with each Sch.

1 1 1 1 1

~~Ans~~

for each position we have $\{1, *\}$

$$\begin{array}{cccc} \cancel{1} & - & \cancel{1} & - \\ \hline & - & - & - \end{array} \rightarrow 2^5 = 32$$

Q. Pop^n size = n

How many Schemata?

$$n \times 2^l$$

String 11111

$\nwarrow 0 1 0 1 2$

Range of Schemata (n^l : ~~$n \times 2^l$~~)
for pop^n .

* Every Schema associated with Two properties:-

* Order

* Defining length.

* No. of fixed ~~Element~~ Element \rightarrow Order.

Ex:- ~~11***~~ \rightarrow Order = 2

~~1***~~ \rightarrow Order = 1

* Defining length \rightarrow will provide Span.

\nwarrow Difference b/w 1st & last fixed no.
position :-

$$11*** = \underline{\underline{-}}_1$$

$$1*** = 0$$

* Effect of Selection Operator :-

Individual or String x

$$\text{Probability of Selection} = \frac{f(x)}{\sum f}$$

Expected # of copy of an individuals

$$x = \frac{f(x)}{\bar{f}(t)}$$

$\bar{f}(t) \rightarrow \text{average}$

fitness
Value of

popl.
 $t=0$
initially

⇒ Schema H

Instance \rightarrow those string match by this Schema.

$n \rightarrow \text{popl size.}$

$l \rightarrow \text{length of string.}$

Schema H

⇒ There are 'm' instances associated with this Schema H within a popl A(t) at a given timestamp 't'.

$m(H, t)$



How many no. of chromosome in next generation matches with schema H.

$E[m(H, t+1)] \rightarrow$ Expected no. of members
of schema H in next generation.

$$E[m(H, t+1)] = \sum_{x \in H} \left(\frac{f(x)}{\bar{f}(t)} \right)$$

$$= \frac{1}{\bar{f}(t)} \sum_{x \in H} f(x). \quad (1)$$

Suppose, $\bar{f}(t, t)$ is average fitness of
schema H. \approx

$$\bar{f}(H, t) = \frac{\sum_{x \in H} f(x)}{m(H, t)} \quad (2)$$

* $E[m(H, t+1)] = \frac{m(H, t) \bar{f}(H, t)}{\bar{f}(t)}$

Conclusion Known as Grow or Decay Eqn.

(1) If $f(H, t) \geq \bar{f}(t) \rightarrow$ no. of members/copy
will increase,

(2) If $f(H, t) < \bar{f}(t) \rightarrow$ copy will decrease.

* Average above schema will survive.

* Average below schema will die off.

$$m(H, t+1) = \frac{m(H, t)}{f(H, t)} f(H, t) \quad \text{--- (A)}$$

* Effect of Crossover on the Schema (H) :-

$p_c \rightarrow$ Crossover probability

$x_a \rightarrow$ random no. generated (0,1)

generated if $x_a < p_c \rightarrow$ apply crossover.

random position

$x \rightarrow 01110000 \rightarrow$ one string.

$H_1 \rightarrow * * * * * 0 *$

$H_2 \rightarrow * * * 1 0 * * *$

{ Schemata }

Q after applying Crossover what is chances of destroying Schema H_1, H_2 ?

* Chances of destroying H_1 is higher than H_2 .

Probability of destroying Schema H is given by $\frac{s(H)}{(l-1)}$

$l \rightarrow$ length of string

$s(H) \rightarrow$ defining length of Schema H .

$H_1 \rightarrow s(H) \Rightarrow 7-2 = \frac{5}{7}$ surviving = $\frac{2}{7}$

$H_2 = \frac{1}{7}$

surviving Probability

$H_2 = 1 - \frac{1}{7} = \frac{6}{7}$

Probability of destroying schema(H) $p_c \cdot \frac{s(H)}{(l-1)}$

Probability of surviving schema $H = 1 - p_c \cdot \frac{s(H)}{(l-1)}$

Conclusion:- Short defining length schema will survive
defining length \rightarrow Survive[↑].

* Combining Crossover & Selection Effect :-

$$m(H, t+1) \geq \frac{m(H, t) f(H, t)}{\pi(t)} \left[1 - p_c \cdot \frac{s(H)}{(l-1)} \right]$$

* Average above & short defining length schema will survive.

L-20

Effect of Mutation on Schema II :-

Take P_m in range of $\left[\frac{1}{\text{Pop. size}}, \frac{1}{\text{length of chromosome}} \right]$

Generate random no.

Schema :-

* 1 * 0 * 1 *

* Valid for fixed Element,

* → can take any value.
mutation doesn't destroy *

p_m → probability of destroying.

$(1-p_m)$ → probability of surviving.

for every position destroying probability p_m .

| * * * 0

$$\begin{aligned} \text{Total probability of surviving} &= (1-p_m) * (1-p_m) \\ &= (1-p_m)^2 \end{aligned}$$

Generally,

$$\text{Probability of surviving} = \cancel{(1-p_m)}^{\text{order}} (1-p_m)$$

order = no. of
fixed elements

* Combining crossover & Selection & Mutation Effect :-

$$m(H, t+1) \geq \frac{m(H, t) f(H, t)}{f(t)} \left[1 - \frac{s(H)}{l-1} \right] [1 - p_m]^{O(H)}$$

$$(1-x)^n = \sum_{k=0}^n \binom{n}{k} (-x)^k$$

Date _____
Page _____

$$m(h, t+1) \geq \frac{m(h, t)}{f(t)} f(h, t) \left[1 - b_c \frac{s(h)}{(l-1)} \right] \left[1 - o(h) \cdot p_m \right]$$

$$o(h) = 1$$

→ surviving probability is large.

Lower order schema has higher chance of surviving

$$m(h, t+1) \geq \frac{m(h, t)}{f(t)} f(h, t) \left[1 - b_c \frac{s(h)}{(l-1)} - o(h) \cdot p_m \right]$$

Final Conclusion :-

Average above short defining length & low order schemata has survive after selection, crossover & mutation operation
 * is called Schemata Theorem.

Also known as Fundamental Thm of Genetics Algorithm.

$$\underline{\text{Ex:-}} \quad f(x) = x^2$$

String No	x	Fitness Value	b ⁰	Expected count	factual count	8 th
1	01101	13	169	0.14	0.58	1
2	11000	24	576	0.49	1.97	2
3	01000	8	64	0.06	0.22	1
4	10011	19	361	0.31	1.23	1
	Sum		1170			
	Avg	$\frac{1170}{4} = 293$				Max-Set

Matting Pool

0110 19 corners. Random no = 4
11220

f11000 random no = 2

10031 $\mu m = 0.001$

<u>New child</u>	<u>x</u>	<u>f(x)</u>
01100	12	144
11001	25	625
11011	27	729
10000	16	256
		1754

$$\text{Avg} = 439$$

Max -729

This is String Processing.

~~Computed by average value of fitness of its members.~~

~~#~~ Now, Schema processing.

<u>Max</u>	String No	matched	Average fitness Value	fitness of Schema
H ₁ 1 * * * *	2, 4	576 + 361	2	= 468.5
H ₂ 1 0 * *	2, 3	576 + 64	2	= 320
H ₃ 1 * * * 0	2	576		= 576

Actual Selection probability of Scheme 31.

$$m(t+1) = \frac{m(t)}{\bar{f}(t)} f(t)$$

$$m(n_1, t+1) = \frac{2 \times 469}{293} = 3.20 \stackrel{\text{actual}}{\Rightarrow} 3.$$

$$m(H_2, t+1) = 2 \times \frac{320}{293} = 2.18$$

$$m(H_3, t+1) = 2 \times \frac{576}{293} = 1.97$$

String # matches schema with

2, 3, 4

2, 3

2, 3

~~(1/2) generation~~
matching pool

After ~~crossover~~ Crossover :-

($- \beta_c S(H)$)

$\ell - 1$

	$S(H)$	$m(H, t+1)$
H_1	0	$3.20 \times (1-0) = 3.20$
H_2	1	$2.18 \times (1 - \frac{1}{4}) = 2.18 \times 0.75 = 1.64$
H_3	0	$1.97 \times (1 - \frac{4}{4}) = 0$

After mutation :-

$(1 - \beta_m)^{\ell+1}$

$\beta_m = 0.001$

	$m(H, t+1)$	$(1 - \beta_m)^{\ell+1}$
H_1	1	$3.20 \times (1)^2 = 3.20$
H_2	2	$2.18 \times 0.75 \times (1 - 0.001)^2 = 1.64$
H_3	2	$1.97 \times 0 \times 1 = 0$

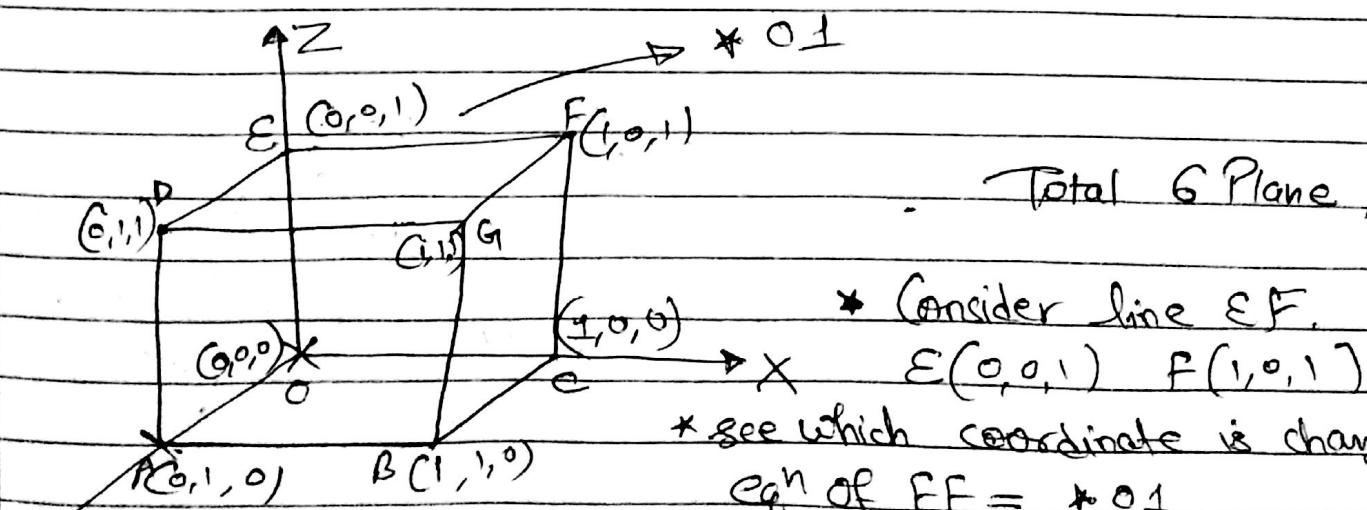
After all operations String # matches Schema with 3rd generation

H_1	2, 2, 0	2, 3, 3	2, 3, 4
H_2	2		2, 3
H_3	0		4

All Schemas which satisfy the Schema Th^m are called building blocks.

L-21 Schema is hyperplane :-

* plane is 3D having linear eqn in their variable? and for higher dimension it is hyperplane.



* End of line :-

$$FG = 1 * 1$$

$$OC \Rightarrow *00$$

Plane :- DEFG

$$Eq^n \rightarrow **1$$

ADED $\rightarrow 0**$

Any Schema can represent in hyperplane.
in higher dimension.

Mid Sem :-

Q. $f(x) = x^2 + y^2$ where $1 \leq x \leq 15$
 $y \geq 3$ $x+y=7$
 $y = 7-x$

$$f(x) = x^2 + (7-x)^2$$

$$f(x) = 2x^2 - 14x + 49$$

$$7 \geq y \geq 3$$

$$x \in [1, 4]$$

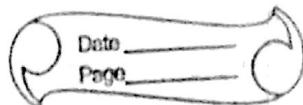
$x \rightarrow$ Integer based $\rightarrow \frac{1}{2} \int_{3}^{4x}$

$\frac{9}{2}$
 $\frac{8}{3}$
 $\frac{7}{4}$

} crossover \rightarrow single pt.

not apply as
first & last position
is same.

Chapter - 4



Genetic Programming :-

- * Is used for list of Progr. Initially

Major challenge of AI

→ Artificial Intelligence

Automatic Programming.

KOZA :-

$$t^2 = \alpha \frac{r^3}{\text{distance from Sun}}$$

Classification model

if ($n=2$) AND ($\text{Salary} \geq 8000$)

Then good else bad

Not applied
mutation operator
in this initial
book but
steps are
similar.

If formula then good else bad.

Orbit Time Problem :- To find. relationship b/w
time period & average distance from sun.

Steps :-

Step 1 - Every Candidate Solⁿ can be represented by
PARSE Tree.

Write list of Program :-

(defn orbital-period ()

; Moons ;

;) P → command

Setf used to assign

1.52 to A

Value to Variable

(setf A 1.52)

List Programming :-

Date _____
Page _____

(defn orbital-period ()

; Mars ;

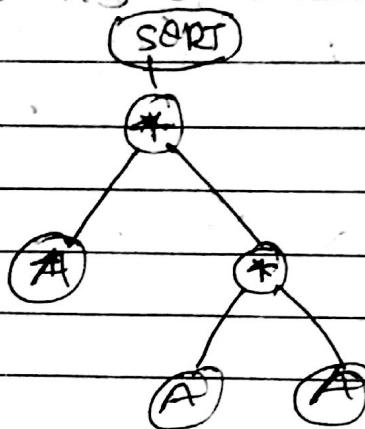
(setf A 1.52)
(-sqrt (* A (* AA)))

$$T^2 = A^3$$

$$T = \sqrt{A^3}$$

feature :-

- * Operator have the priority. to this operators \rightarrow fn
operands are known as Terminal.



Step 2 - Each Candidate soln can be expressed as a Parse Tree.

And this consists of functions & terminals
+ non leaf nodes + leaf nodes.

- * Correctness of that function is given by the no. of branches of that node

~~Correctness~~
no. of branches = no. of argument for that fn

Q. Function nodes & Terminal nodes ?

Ans

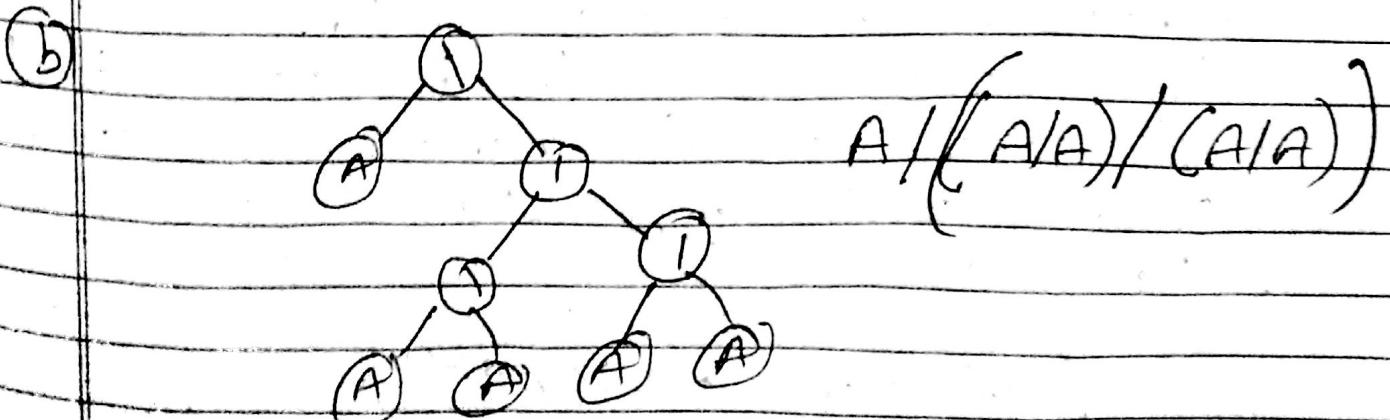
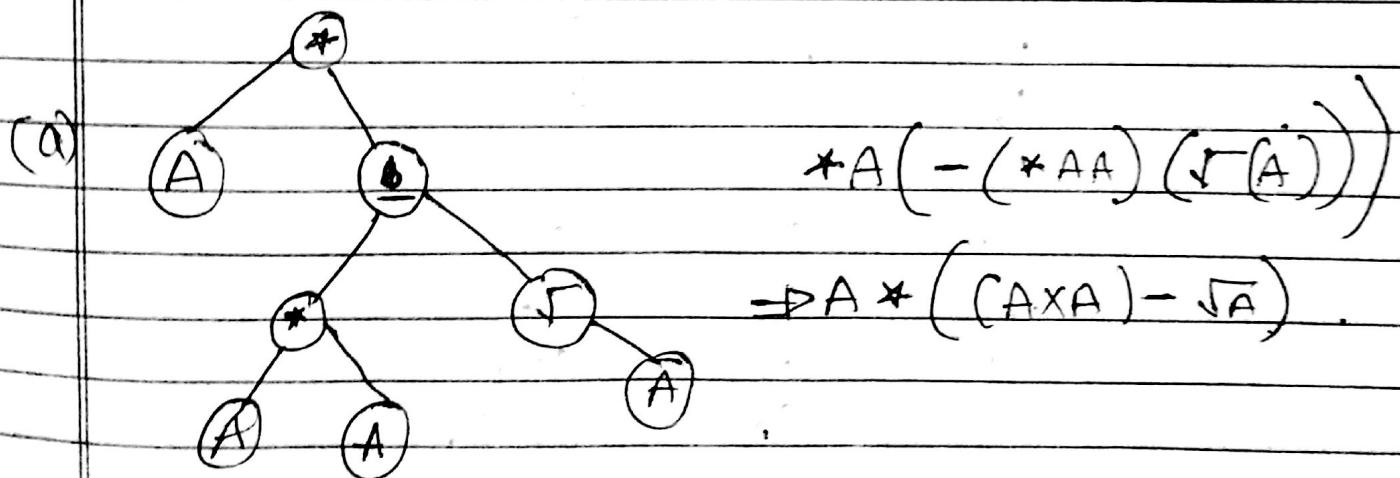
10. choose the appropriate functions & Terminal set.

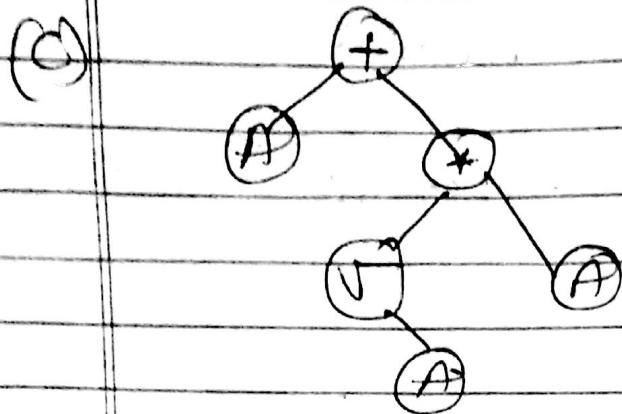
Terminal set = {A}

function set = { +, -, /, *, √ }

1(i) Initial Population generation randomly.
size of popl = 3.

1(ii) Generate 3 parse tree with help of Terminal & function sets.





$$A + (\sqrt{A} * A)$$

Step 2. Compute the fitness of each chromosome / Expression / Solution / Individual / Program / Parse Tree in the population by running it on a set of fitness cases (Historical Data). & consider fitness fn.

Planet	A	T
V	0.72	0.61
E	1	1
M	1.02	1.04
J	5.20	11.9
S	9.053	29.4
U	19.1	83.4

* fitness of Program is a function of the fitness Cases on which it perform correctly or some fitness fn might given partial credit for a program for getting close to correct output.

Partial credit \rightarrow 20% (approx)

$A = 0.72 \quad T = 0.41 \quad \text{& } 0.81$ if we get T in the range the range \rightarrow we count 1.