CROSSOVER OPERATORS

1

(1) Single point crossover → X site (cross site)

Parent 1 (P_1) → 110|11100

Parent 2 (P_2) → 101|01011

Child 1 → C_1 → 11001011

Child 2 → C_2 → 10111100

(2) Two point Cross Over: X_1 X_2

P_1 → 11|01|1100

P_2 → 10|10|1011

C_1 → 11101100

C_2 → 10011011

(3) Three Point crossover (N=3) X_1 X_2 X_3

P_1 → 10|1101|11|010

P_2 → 11|0011|01|101

C_1 → 10001111101

C_2 → 11110101010

(4) Uniform crossover (involves a Mask chromosome)

P_1 → 10101101

P_2 → 11011001

M → 10110110

C_1 → 11101101

C_2 → 10011001

child 1's allele value is P_1's if corresponding Mask " is 1 and P_2's if " " is 0. } CHILD 1

CHILD2 → { for child2: assign P_1's value if Mask bit = 0 and P_2's value if Mask bit = 1

(5) Three parent crossover

P_1 → 11010110

P_2 → 10101101

P_3 → 11101010

child → 11101110

* child = P_1 if P_1 and P_2 are equal @ corresponding bit position / allele value
* otherwise assign P_3's allele value to child (offspring)

⑥ Precedence Preservative crossover – Generally applied with Permutation Encoding / representation.

Parent 1 → ~~A B C D E F~~ (A B C D E F)

Parent 2 → ~~A C B E F D~~ (A C B E F D)

select Pid → 1 2 1 1 2 2

Offspring (child) → $\boxed{\text{A C B D E F}}$

⑦ Partially Matched crossover

(P1) A = 9 8 4 | 5 6 7 | 1 3 2 10 Permutation Coding.

(P2) B = 8 7 1 | 2 3 10 | 9 5 4 6

C1 = 9 8 4 2 3 10 1 6 5 7

C2 = 8 10 1 5 6 7 9 2 4 3

⑧ Ordered crossover:-

P_1 = 9 8 4 | 5 6 7 | 1 3 2 10

P_2 = 8 7 1 | 2 3 10 | 9 5 4 6

P_2 = 8 H 1 | 2 3 10 | 9 H 4 H (H → Hole as mapping happens)

Holes to be slided post 2nd x site

P_2 → 2 3 10 | H H H | 9 4 8 1

4 values from mate = 5 6 7

∴ child2 = $\boxed{2\ 3\ 10\ 5\ 6\ 7\ 9\ 4\ 8\ 1}$

IIIly P_1 → 9 8 4 | 5 6 7 | 1 H H H

→ 5 6 7 | H H H | 1 9 8 4

∴ C_1 → 5 6 7 | 2 3 10 | 1 9 8 4 //

~~Ordered~~ / ~~Crossover~~ :- Cyclic Crossover

$P_1 =$ 9 8 2 1 7 4 5 10 6 3

$P_2 =$ 1 2 3 4 5 6 7 8 9 10

$C_1 = \boxed{9 \quad \underline{2} \quad \underline{3} \quad 1 \quad \underline{5} \quad 4 \quad \underline{7} \quad \underline{8} \quad 6 \quad \underline{10}}$ ⊗

~~$C_2 =$ 1 4 6~~

1 2 3 4 5 6 7 8 9 10

9 8 2 1 7 4 5 10 6 3

$C_2 = \boxed{1 \quad 8 \quad 2 \quad 4 \quad 7 \quad 6 \quad 5 \quad 10 \quad 9 \quad 3}$ ⊗

(Permutation Encoding)

Advanced operators.

Dominance, Diploidy, Abeyance

↳ diploidy – pair of chromosomes

dominance – genotype to phenotype mapping.

↳ Previous representations – haploid (or) single stranded chromosome

Diploid – Genotype carries 'one (or) more pairs' of

↳ chromosomes (Homologous chromosomes), each containing information for the same function

↳ Issue of Redundancy / Duplications. Pairs of genes that decode to same fn. (values)?

↳ consider following diploid structure :-

{ AbCDe
{ aBCde

A, a } → different characteristics

↳ Could be "Eye color" – A → Blue; a → Brown, etc.

But both are not possible @ any time.

↳ Conflict resolved by operator – Dominance.

↳ An allele is dominant if expressed in phenotype.

assuming 'Caps' are all dominant.

dominant allele ← (A)bCDe ⟶ ABCDe

recessive ← (a)BCde

Dominance / Recessive Rules

{ Aa → A } — heterozygous
{ AA → A } — homo "
{ aa → a } → " "

∴ Dominance is a genotype → phenotype map. fn.

Diploidy → Way for Remembering alleles / Combinations '''
Previously
↓
Dominance → Protect / Shield Such alleles from harmful for. Selection.

↳ Diploidy allows multiple solns (to same problem) to be carried along, however only one soln is expressed.

Hollstein's study :- Diploidy + Evolving Dominance

Each binary gene → described by two genes (values) ⟶ Modifier and Functional Gene.

Functional Gene ⟶ 0/1 values ⟶ decodes to a normal parameter.

Modifier Gene ⟶ M/m "

Dominance ⟶ of 0 if atleast one 'M' allele is present.

	OM	Om	1M	1m
OM	0	0	0	0
Om	0	0	0	1
1M	0	0	1	1
1m	0	1	1	1

Two locus Dominance Map

↳ Two locus scheme evolved to a simpler 1 locus
- Third allele @ each locus

↳ allele values from {0,1,2} set.

2 → dominant 1
1 → Recessive 1

Rules: 2 and 1 ⟶ 1

$\boxed{\begin{array}{l} 2 \text{ dom } 0 \\ 0 \text{ dom } 1 \end{array}}$

	0	1	2
0	0	0	1 (2)
1	0	1	1 (1/2)
2	1 (2)	1 (1/2)	1 (2)

Selection Schemes

1

↳ objective of evolving selection methods that reduce stochastic errors of Roulette wheel selection.

↳ • few in literature
(a) Deterministic Sampling
(b) Remainder Stochastic Sampling w/o replacement "b"
(c) Stochastic Sampling w/o "c"
(d) b with replacement
(e) c with replacement
(f) Stochastic tournament (Ranking scheme)

e <=> Roulette wheel method
c <=> Expected value model (R3 plan)

(a) → prob. of selection = $f_i / \Sigma f$.
Ex. No. of Individuals for a string; e_i

$$\boxed{e_i = p_{select_i} \cdot n}$$

↳ Each string is allocated copies = integer part of e_i
↳ Population sorted on "fractional part"
↳ Remainders of strings to make population — top of the sorted list.

* b and (d) start similarly to Deterministic Sampling
↳ (Integer part) No. of Copies

In (d) — fractional parts of e_i — determine weights for roulette wheel to fill up the rest of the population.

In (b) - fractional parts - probabilities

Thus if e_i = 1.5
1 sure copy and another with prob 0.5.

(6) Stochastic Tournament - Tournament selection
↳ one of two competing (drawn @ random) is declared winner and inserted in new population
↳ Successive pairs of individuals are drawn using r/w selection.

↳ Gen. observation of good perfo. by. (b) in most applications

Critical Events of Interest – not only start soln of Previous but also those that yield a improved objective fn values (local TS optimum)

Meta heuristic → Method for solving class of computational problems by combining user given black box procedures — in the hope of finding an (heuristics) efficient / robust procedure.