Starkweather Paper

Wainwright

Crossover Operators for Permutation (Order Based) Chromosomes

Start at 2.2: Covers Order1, Order2, PMX, Cycle and Position Based Crossover Operators

Order1 Crossover: Figure 2

Parent1: a b c d e f g h i j
Parent2: c f a j h d i g b e

Randomly select two slice points in the chromosome as shown above.

Determine a selected parent as the basis for the crossover to produce one child, say Parent2 to produce Child2.

The offspring, called Child2, (since Parent2 is the basis), inherits the elements between the two crossover points inclusive from Parent2:

Hence Child2 so far looks like: _ _ a j h d i _ _ _

The remaining elements of Child2 are inherited from the alternate parent, Parent1, in the order in which they appear in that parent from the first position following the second crossover point. In this case start with h.

The elements are placed into Child2 beginning at the first position following the second crossover point skipping over all elements already present in the offspring, Child2.

Hence Child 2 becomes: f g a j h d i b c e

What happens when Parent1 is the basis producing Child1?

Child1 inherits from Parent1: _ _ c d e f g _ _ _

And then the rest of Child1 comes from Parent2 producing:

Child1: h i c d e f g b a j

Order2 Crossover: Figure 3

```
               *   *           *       *
Parent1:  a    b   c   d   e   f   g   h   i   j
Parent2:  c    f   a   j   h   d   i   g   b   e
```

Randomly select positions (genes), say 4, in the chromosome as shown above with *.

Determine a selected parent as the basis for the crossover to produce one child, say Parent1 to produce Child1.

The order in which the selected elements appear in the other parent (Parent2, a j i b) is imposed on the selected parent (Parent1) to produce Child1.

Hence find a j i b in Parent1 and rearrange them left to right in a j i b order to produce Child1.

Parent1: **a** **b** c d e f g h **i** **j**

Child1: **a** **j** c d e f g h **i** **b**

What happens when Paren2 is the basis producing Child2?

Find c d g i in Parent2 and rearrange them left to right in c d g i order to produce Child2.

Parent2: **c** f a j h **d** **i** **g** b e

Child2: **c** f a j h **d** **g** **i** b e

```
Parent 1:     a b c d e f g h i j
Parent 2:     c f a j h d i g b e
Cross Pts:      *         *
Offspring:    f g a j h d i b c e
```

Figure 2: Order Crossover #1

```
Parent 1:     a b c d e f g h i j
Parent 2:     c f a j h d i g b e
Cross Pts:        * *     *   *
Offspring:    a j c d e f g h i b
```

Figure 3: Order Crossover #2

PMX Crossover: Figure 4
No Mapping Conflicts

Parent1: a b c d e f g h i j
Parent2: d i j h a g c e b f

Randomly select two slice points in the chromosome as shown above.

The elements between the slice points determine a mapping of alleles:
c <-> j d <-> h e <-> a f <-> g

Perform the mapping on Parent1 to produce Child1 and perform the mapping on Parent2 to produce Child2 shown below:

Child1: e b j h a g f d i c
Child2: h i c d e f j a b g

```
Parent 1:    a b c d e f g h i j
Cross Pts:      *       *
Parent 2:    d i j h a g c e b f
Offspring:   h i c d e f j a b g
```

Figure 4: PMX Crossover Example 1

```
Parent 1:    a b c d e f g h i j
Cross Pts:      *       *
Parent 2:    c f a j h d i g b e
Offspring:   a j c d e f i g b h
```

Figure 5: PMX Crossover Example 2

```
Parent 1:    a b c d e f g h i j
Cross Pts:          *
Parent 2:    c f a j h d i g b e
Offspring:   c b a d e f g h i j
```

Figure 6: Cycle Crossover

```
Parent 1:    a b c d e f g h i j
Cross Pts:     * *     *     *
Parent 2:    c f a j h d i g b e
Offspring:   a b c j h f d g i e
```

Figure 7: Position Based Crossover

PMX Crossover: Figure 5
Mapping (One Conflict)

Parent1: a b c d e f g h i j
Parent2: c f a j h d i g b e

Randomly select two slice points in the chromosome as shown above.

The elements between the slice points determine a mapping of alleles:
c <-> a d <-> j e <-> h f <-> d Note d has a dual mapping.

The Children always receive the elements from the other parent between the slice points as follows: There is never a mapping problem between the slice points.

Child1: _ _ a j h d _ _ _ _
Child2: _ _ c d e f _ _ _ _

Now begin to fill in the other elements as best as possible based on the mapping, skipping elements involving in the mapping conflict: d with j and f.

This produces
Child1: c b a j h d g e i _
Child2: a _ c d e f i g b h

For every unassigned allele determine who is missing and assign them randomly or if order is important for the problem being solved, then perhaps assign the elements as they appear in the other parent. In this case in Child1 f is missing and in Child2 j is missing producing:

Child1: c b a j h d g e i **f**
Child2: a **j** c d e f i g b h

PMX Crossover: Figure 5
Mapping (Multiple Conflicts)

Parent1: a b c d e f g h i j
Parent2: c f e j h d i g b a

Randomly select two slice points in the chromosome as shown above.

The elements between the slice points determine a mapping of alleles:
c <-> e d <-> j e <-> h f <-> d Note d,e have a dual mapping.

The Children always receive the elements from the other parent between the slice points as follows: There is never a mapping problem between the slice points.

Child1: _ _ e j h d _ _ _ _
Child2: _ _ c d e f _ _ _ _

Now begin to fill in the other elements as best as possible based on the mapping, skipping elements involving in the mapping conflict: (d with j and f) and (e with c and h)

This produces
Child1: a b e j h d g _ i _
Child2: _ _ c d e f i g b a

For every unassigned allele determine who is missing and assign them randomly or if order is important for the problem being solved, then perhaps assign the elements as they appear in the other parent. In this case in Child1 c,f are missing and in Child2 h,j are missing producing:

Child1: a b e j h d g **c** i **f** *Assigned c,f in the order appearing in P2
Child2: **h j** c d e f i g b a * Assigned h,j in the order appearing in P1

*For simplicity, I assigned the missing alleles in the order they appeared in the other parent. The assignment could be random.

Cycle Crossover: Figure 6

				*						
Parent1:	a	b	c	d	e	f	g	h	i	j
Parent2:	c	f	a	j	h	d	i	g	b	e

Randomly select a start position for the cycle to begin denoted by *.

Determine a selected parent as the basis for the crossover to produce one child, say Parent1 to produce Child1.

The element at the cycle starting point of the selected parent, Parent1, is inherited by Child1. The element which is in the same position in the other parent cannot then be placed in this position so its position is found in the selected parent, Parent1, and is inherited from that position by the child. This continues until the cycle is completed by encountering the initial item, d, in the other parent. Any elements which are not yet selected in the child are inherited from the other parent. Note cycle crossover always preserves the position of elements from one parent or the other without any disruption.

				*						
Cycle Order		3		1	7	2	5	6	4	8
Parent1:	a	b	c	d	e	f	g	h	i	j
Parent2:	c	f	a	j	h	d	i	g	b	e
Child1:	c	b	a	d	e	f	g	h	i	j

				*						
Cycle Order		7		1	3	8	5	4	6	2
Parent2:	c	f	a	j	h	d	i	g	b	e
Parent1:	a	b	c	d	e	f	g	h	i	j
Child2:	a	f	c	j	h	d	i	g	b	e

Position Based Crossover: Figure 7

This operator is intended to preserve position information during the crossover operation.
It is very similar to Order1.

		*	*			*			*	
Parent1:	a	b	c	d	e	f	g	h	i	j
Parent2:	c	f	a	j	h	d	i	g	b	e

Randomly select several positions (genes), say 4, in the chromosome as shown above with *.

Determine a selected parent as the basis for the crossover to produce one child, say Parent1 to produce Child1.

The elements in the selected positions are inherited in the child as follows:

Child1: _ b c _ _ f _ _ i _

The remaining elements are inherited from the other parent in the order in which they appear (from the beginning of the chromosome) skipping over all elements which already appear in the child. The remaining elements are: a, d, e, g, h, j which appear in the order of a, j, h, d, g, e in Parent2.

Child1: a b c j h f d g i e

Similarly, using the same selected positions and Parent2

Child2 is: c <u>f</u> <u>a</u> e g <u>d</u> h i <u>b</u> j

Reminder from "Example Crossover and Mutation Techniques" Handout (160)
Repeated here

Uniform Crossover

Parent 1 0 1 1 0 0 1 1 0 1 0 1 0

Parent 2 1 1 0 0 1 0 0 0 1 1 0 1

U 1 0 1 1 1 0 1 0 1 1 1 0 The Uniform bit string is randomly generated

Rules for generating Child1: if Ui = 0, select the ith bit from Parent1
if Ui = 1, select the ith bit from Parent2

Hence

Child1 is 1 1 0 0 1 1 0 0 1 1 0 0

Rules for generating Child2: if Ui = 0, select the ith bit from Parent2
if Ui = 1, select the ith bit from Parent1

Hence

Child2 is 0 1 1 0 0 0 1 0 1 0 1 1

Note Uniform crossover can be used on most any chromosome representation where you want to select the allele from one parent or the other. It does not have to be bits.

Reminder from “GA Notes-Some well known problems” Handout (170)
Repeated here

The “Cut-and-crossfill” crossover for Permutations.
Two Parents produce two Children

1. Select randomly a slice position in the chromosomes. Choices are 1..N-1.

2. Slice both parents into two parts (left and right) at this same position.

3. Copy the left part of Parent 1 directly to Child 1 and the left part of Parent 2 directly to Child 2

4. Scan Parent 2 left to right and fill in the right part of Child 1 with values from Parent 2, skipping alleles it already contains.

5. Repeat for Parent 1 and Child 2.

Example

Parent 1: 3 6 1 | 8 5 4 2 7
Parent 2: 2 8 1 | 7 6 4 3 5

Child 1: 3 6 1 | 2 8 7 4 5
Child 2: 2 8 1 | 3 6 5 4 7