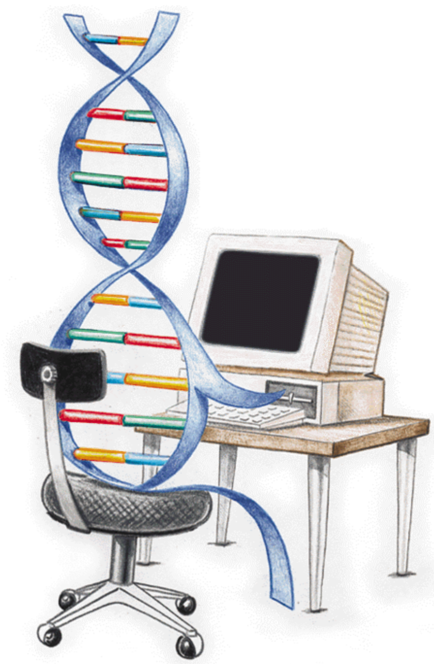
INTRODUCTION TO GENETIC ALGORITHMS



3 PRELIMINARY THREADS

- **Turing's 1948 thoughts on "combinations of genes"**
- **Searches in general**
- **Non-linearity (as illustrated by the giraffe example)**

TURING'S 3 APPROACHES TO MACHINE INTELLIGENCE

- **Turing made the connection between searches and the challenge of getting a computer to solve a problem without explicitly programming it in his 1948 essay "Intelligent Machines" (in *Mechanical Intelligence: Collected Works of A. M. Turing*, 1992).**

> **"Further research into intelligence of machinery will probably be very greatly concerned with 'searches' ... "**

TURING'S 3 APPROACHES TO MACHINE INTELLIGENCE — CONTINUED

1. LOGIC-BASED SEARCH

One approach that Turing identified is a search through the space of integers representing candidate computer programs.

2. CULTURAL SEARCH

Another approach is the "cultural search" which relies on knowledge and expertise acquired over a period of years from others (akin to present-day knowledge-based systems).

TURING'S 3 APPROACHES TO MACHINE INTELLIGENCE — CONTINUED

3. GENETICAL OR EVOLUTIONARY SEARCH

"There is the genetical or evolutionary search by which a combination of genes is looked for, the criterion being the survival value."

TURING'S "GENETICAL OR EVOLUTIONARY SEARCH"

- **from Turing's 1950 paper "Computing Machinery and Intelligence" …**

  "We cannot expect to find a good child-machine at the first attempt. One must experiment with teaching one such machine and see how well it learns. One can then try another and see if it is better or worse. There is an obvious connection between this process and evolution, by the identifications"

  "Structure of the child machine = Hereditary material"

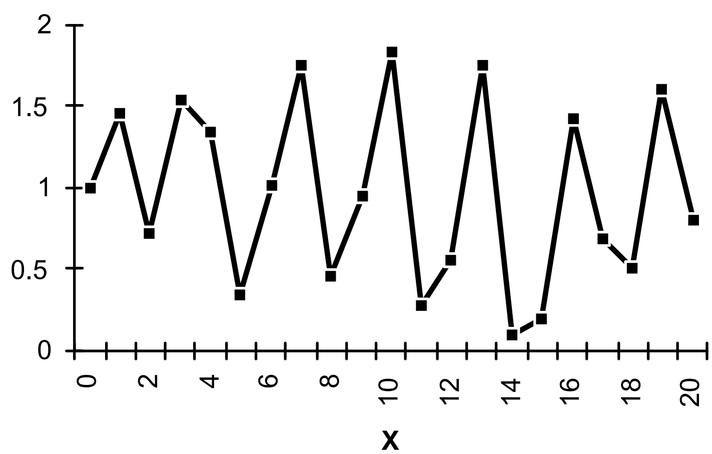
  "Changes of the child machine = Mutations"

  "Natural selection = Judgment of the experimenter"

SEARCH METHODS IN GENERAL

- **INITIAL STRUCTURE (E.G., A POINT OR POINTS IN THE SEACH SPACE OF THE PROBLEM)**
- **FITNESS MEASURE**
- **METHOD OF CREATING NEW STRUCTURE**
- **PARAMETERS**
- **TERMINATION CRITERION AND METHOD OF DESIGNATING THE RESULT**

SEARCHING A SPACE WITH ONE GLOBAL OPTIMUM POINT AND MANY LOCAL OPTIMA



SEARCH METHODS

- **NEITHER ENUMERATIVE RANDOM NOR BLIND RANDOM SEARCH USE ACQUIRED INFORMATION IN DECIDING THE FUTURE DIRECTION OF THE SEARCH**
- **HILL CLIMBING AND GRADIENT DESCENT (ASCENT) DO, IN FACT, USE ACQUIRED INFORMATION IN DIRECTING THE SEARCH**
- **HOWEVER, HILL CLIMBING AND GRADIENT DESCENT (ASCENT) ARE PRONE TO GETTING TRAPPED ON LOCAL OPTIMA AND THEREBY NEVER FINDING THE GLOBAL OPTIMUM. This is especially true for non-trivial search spaces**

EXAMPLE: OPTIMIZING THE DESIGN FOR A GIRAFFE

- **Long neck**
- **Long tongue**
- **Vegetable-digesting enzymes in stomach**
- **4 legs**
- **Long legs**
- **Brown coloration**

THE DESIGN OF A GOOD GIRAFE

Neck length	Tongue length	Carnivorous?	Number of legs	Leg length	Coloration
15.11 feet	14.2 inches	No	4	9.96 feet	Brown
Floating point	Floating point	Boolean	Integer	Floating point	Categorical

NON-LINEARITY

- **Taken one-by-one, most of the gene values (alleles) found in a giraffe contribute (alone) negatively to fitness**
- **For example, the long neck, such as the long neck,**
  - **requires considerable material and energy to construct**
  - **requires considerable energy to maintain**
  - **prone to injury (thereby hurting rate of survival and reproduction)**
- **Thus, maximizing single variables alone will not lead to the global optimum solution**

NON-LINEARITY — CONTINUED

- **There are 15 possible pairs of the 6 variables**
- **When the variables are taken in pairs, many combinations of pairs are <u>doubly detrimental</u>**
- **For example: Long neck and long tongue is <u>doubly detrimental</u>**

NON-LINEARITY — CONTINUED

- **But, certain combinations of multiple traits, when taken together, are "co-adapted sets of alleles" (Holland 1975) that yield a very fit animal for eating high acacia leaves in the jungle environment, having good camouflage, having high escape velocity when faced with predators, and exploiting a niche (i.e., less competition) with other animals feeding on low-hanging vegetation**

HAMBURGER RESTAURANT PROBLEM

THE PROBLEM: Assuming no knowledge of the hamburger business, search for the management strategy for operating restaurants that maximizes profits

HAMBURGER PRICE:

1 = $ 0.50 price
0 = $10.00 price

ACCOMPANYING DRINK:

1 = Coca Cola
0 = Wine

RESTAURANT AMBIANCE:

1 = Fast service
0 = Leisurely service with waiter in tuxedo

HAMBURGER RESTAURANT PROBLEM — CONTINUED

3-BIT CHROMOSOME (GENOME) FOR A FRENCH RESTAURANT WITH $10 PRICE, WINE, AND LEISURELY SERVICE

VARIABLE1	VARIABLE2	VARIABLE3
0	**0**	**0**

3-BIT CHROMOSOME (GENOME) FOR MC DONALD'S

VARIABLE1	VARIABLE2	VARIABLE3
1	**1**	**1**

THE SEARCH SPACE

1	000
2	001
3	010
4	011
5	100
6	101
7	110
8	111

- **Alphabet size K=2. Length L=3.**
- **Size of search space: $K^L=2^L=2^3=8$**

- **81-bit problems are very small for GA**
- **However, even if L is as small as 81, $2^{81} \sim 10^{27}$ = number of nanoseconds since the beginning of the universe 15 billion years ago**

- **Keep in mind a more practical example, say, L = 100 and M = 10,000**

FITNESS MEASURE WHEN THE McDONALD'S RESTAURANT (111) IS THE GLOBAL OPTIMUM

Strategy	Fitness
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

4 RANDOMLY CHOSEN POINTS FROM THE SEARCH SPACE OF THE PROBLEM

#	Price	Drink	Speed	Binary representation
1	High	Cola	Fast	011
2	High	Wine	Fast	001
3	Low	Cola	Leisurely	110
4	High	Cola	Leisurely	010

INFORMATION LEARNED IN WEEK 0

	Week 0	
1	011	3
2	001	1
3	110	6
4	010	2

HOW TO PROCEED TO WEEK *T*+1?

#	Price	Drink	Speed
1			
2			
3			
4			

WHAT IS LEARNED BY TESTING *M*=4 POINTS AT RANDOM DURING WEEK 0?

- **Best-of-generation individual strategy**
- **Worst-of-generation individual**

THE POPULATION OF SIZE *M*=4 GIVES AN ESTIMATE OF THE AVERAGE FITNESS OF THE SEARCH SPACE AS A WHOLE

	Week 0	
1	011	3
2	001	1
3	110	6
4	010	2
Total		12
Worst		1
Aver		3.0
Best		6

- **The average for these *M*=4 points is 3.0**
- **Note that the average fitness of the search space as a whole is actually 3.5. That is, this estimate is imperfect.**

IN ADDITION, THE POPULATION OF SIZE *M* = 4 GIVES AN ESTIMATE OF THE FITNESS OF VARIOUS SINGLE GENE VALUES

	Week 0	
1	011	3
2	001	1
3	110	6
4	010	2

- **The average fitness of the 3 individuals with a 1 in the 2nd position is 3.67 (i.e., above the average of 3.5 for this particular population)**
- **The average fitness of the 2 individuals with a 0 in the 1st position is 4.00 (i.e., above average)**

WHICH IS MORE RELIABLE?

- **The average fitness of the 3 individuals with a 1 in the 2nd position is 3.67 (i.e., above the average of the population)**
- **The average fitness of the 2 individuals with a 0 in the 1st position is 4.00 (i.e., above average)**

DECEPTION

- **The average fitness of the 2 individuals with a 0 in the 3rd position is 4.0 (i.e., above average)**
- **This information is “deceptive” and suggests the global optimum is somewhere other than where we know it actually is (namely, a 1 in the 3rd position)**

IN ADDITION, THE POPULATION OF SIZE *M* = 4 GIVES AN <u>ESTIMATE</u> OF THE FITNESS OF VARIOUS <u>PAIRS</u> OF GENE VALUES

	Week 0	
1	011	3
2	001	1
3	110	6
4	010	2

- **For example, the average fitness of the 2 individuals with a 10 in the 2nd and 3rd positions is 4.0 (i.e., above average)**
- **Again, this information about pairs is "deceptive" and suggests the global optimum is somewhere other than where we know it actually is (namely, a 11 in the 2nd and 3rd positions)**

SCHEMA

• A *schema* (plural: *schemata*) is a set of points from the search space with specified similarities.

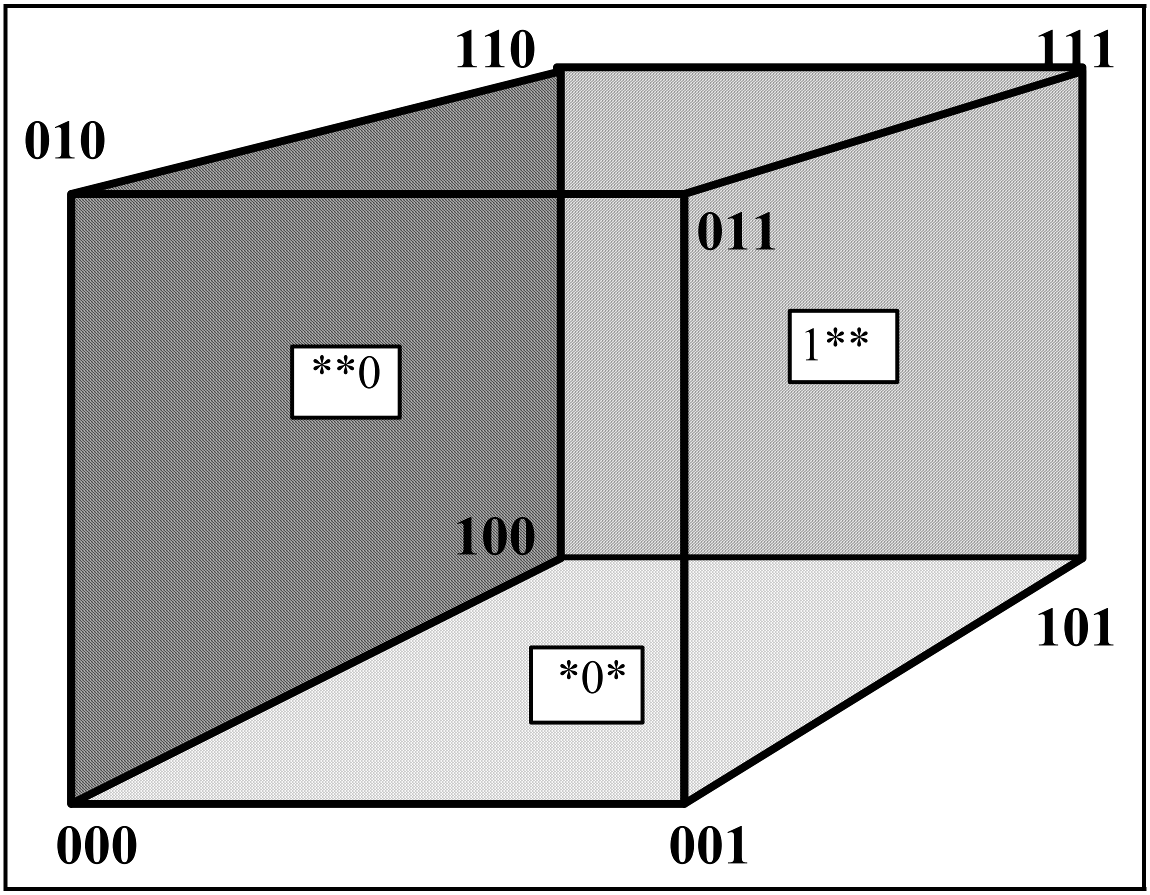
• A schema is described by a string of length *L* (i.e., same length as the strings in the search space) over an extended alphabet of size *K*+1 consisting of the alphabet of the representation scheme (e.g., 0 and 1 if *K*=2) and a *don't care symbol* (denoted by an *)

• The *order*, *O(H)* (or *specificity*) of a schema is the number of specified positions

• For example, the schema 1 is a schema of specificity 1 and consists of the set of points from the search space with a 1 in the 1st position (and we don't care what's in the remaining positions)**

$$H_1 = 1** = \{100, 101, 110, 111\}$$

SIX SCHEMATA OF SPECIFICITY 1



• H_1 = **1** = {100, 101, 110, 111} is the plane in the back**

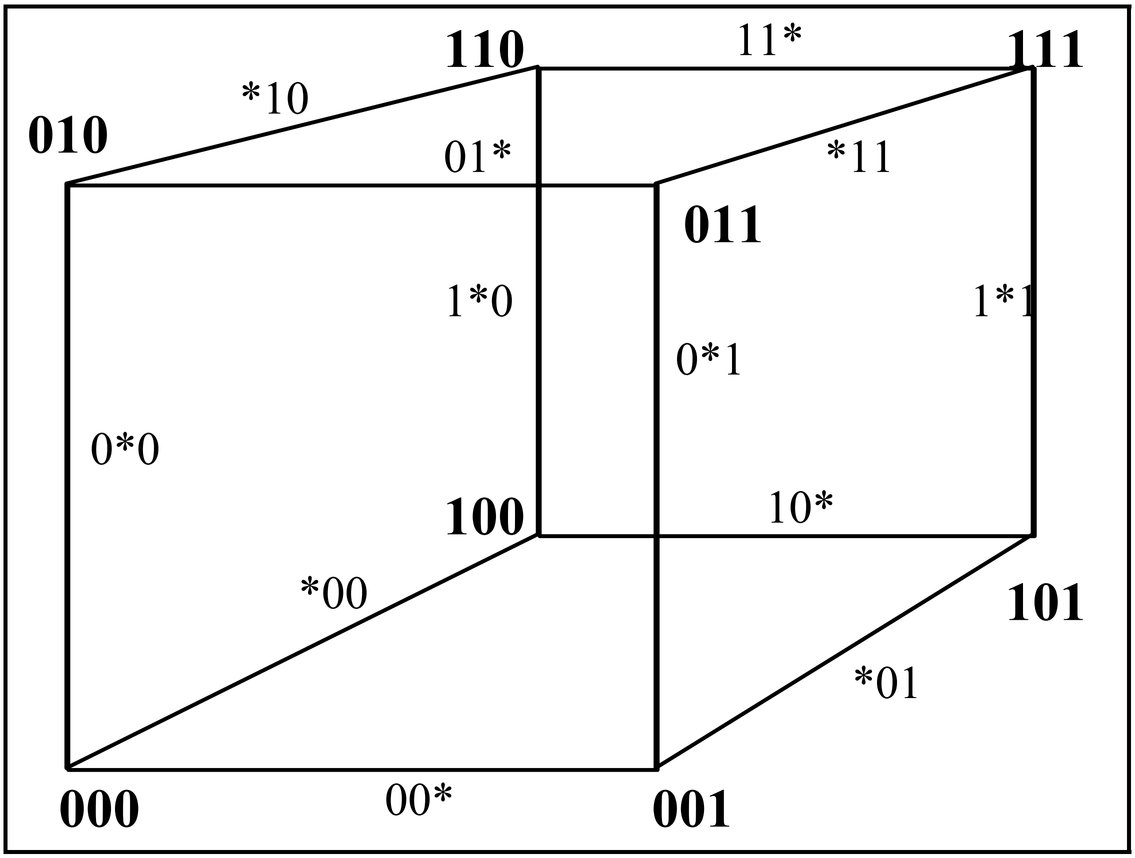
EXAMPLE OF SCHEMA OF SPECIFICITY 2

- **For example, the schema 11* is a schema of specificity 2 and consists of the set of points from the search space with a 1 in the 1st position and a 1 in the 2nd position (and we don't care what's in the remaining positions)**

$$H_2 = 11* = \{110, 111\}$$

- **This schema of specificity 2 is the line in the back connecting 110 and 111**

THE SCHEMATA



GEOMETRIC INTERPRETATION OF THE SCHEMATA

Schema specificity *O(H)*	**Hyperplane dimension**	**Geometric realization**	**Individuals in the schema**	**Number of such schemata**
3	0	Point	1	8
2	1	Line	2	12
1	2	Plane	4	6
0	3	Entire cube	8	1
			Total	27

THE 3^L=27 SCHEMATA

	Schema	Individual strings
1	000	000
2	001	001
3	00*	000, 001
4	010	010
5	011	011
6	01*	010, 011
7	0*0	000, 010
8	0*1	001, 011
9	0**	000, 001, 010, 011
10	100	100
11	101	101
12	10*	100, 101
13	110	110
14	111	111
15	11*	110, 111
16	1*0	100, 110
17	1*1	101, 111
18	1**	100, 101, 110, 111
19	*00	000, 100
20	*01	001, 101
21	*0*	000, 001, 100, 101

22	*10	010, 110
23	*11	011, 111
24	*1*	010, 011, 110, 111
25	**0	000, 010, 100, 110
26	**1	001, 011, 101, 111
27	***	000, 001, 010, 011, 100, 101, 110, 111

THE PROBLEM OF GOING FROM GENERATION 0 TO GENERATION 1 IS A PROBLEM OF ALLOCATING FUTURE TRIALS

GIVEN:

$d_{11}(0)$, $d_{12}(0)$, ..., $d_{43}(0)$

(12 binary inputs)

WANT TO SELECT:

$d_{11}(1)$, $d_{12}(1)$, ..., $d_{43}(1)$

(12 binary unknowns)

USING THE INFORMATION OBTAINED:

$f(H_k,0)$ or $m(H_k,0)$

(up to 27 statistics)

THERE IS A TRADEOFF BETWEEN EXPLOITATION (GREEDY HILL CLIMBING) AND ADVENTUROUS EXPLORATION

EXPLORATION VERSUS EXPLOITATION

- **Estimated cost of Exploration**
  $f_{current\text{-}best} - f_{current\text{-}average}$ = **\$6 – \$3 = \$3**

- **Estimated cost of Not Exploring**
  $f_{global\text{-}best} - f_{current\text{-}best}$ = **\$100,000(???) - \$6**

SCHEMA FITNESS

- **The *schema fitness*, *f*(*H*), is the average of the fitness of all the points from the search space that are contained in the schema**

THE $2^L = 2^3 = 8$ SCHEMATA TO WHICH INDIVIDUAL 110 BELONGS

	Schema	Average fitness
1	**110**	**6**
2	**11***	**6**
3	**1*0**	**6**
4	**1****	**6**
5	*10	4
6	*1*	3.67
7	****0**	**4**
8	*******	**3**

COMPETING EXPLANATIONS WHY 110 PERFORMS AT 200%OF AVERAGE

Reason why 110 has fitness of 6 (200% of the average of the population)	
It's the low price	1**
It's the cola.	*1*
It's the leisurely service.	**0
It's the low price in combination with the cola.	11*
It's the low price in combination with the leisurely service.	1*0
It's the cola in combination with leisurely service.	*10
It's the precise combination of the low price, the cola, and the leisurely service.	110

COMPETING EXPLANATIONS WHY STRATEGY 010 PERFORMS AT 67%

Reason why 010 performs at only 67%	
It's the high price.	0**
It's the cola.	*1*
It's the leisurely service.	**0
It's the high price in combination with the cola.	01*
It's the high price in combination with leisurely service.	0*0
It's the cola in combination with leisurely service.	*10
It's the precise combination of the high price, the cola, and the leisurely service.	010

INCONSISTENT EVIDENCE FOR COMPETING EXPLANATIONS

- **0 ("It's the leisurely service") has 2 pieces of evidence for
  - 200% performance
  - 67% performance

- *1* ("It's the cola") has 3 pieces of evidence for
  - 200% performance
  - 67% performance
  - average performance

- *10 ("It's the cola in combination with leisurely service") has 2 pieces of evidence for
  - 200% performance
  - 67% performance

UNEQUAL AMOUNTS OF EVIDENCE

Reason why 110 performs at 200%		**# of points**
It's the low price	1**	1
It's the cola	*1*	3
It's the leisurely service	**0	2
It's the low price in combination with the cola	11*	1
It's the low price in combination with the leisurely service	1*0	1
It's the cola in combination with leisurely service	*10	2
It's the precise combination of the low price, the cola, and the leisurely service	110	1

UNEQUAL AMOUNTS OF EVIDENCE

Reason why 010 performs at only 67%		**# of points**
It's the high price.	0**	3
It's the cola.	*1*	3
It's the leisurely service.	**0	2
It's the high price in combination with the cola.	01*	2
It's the high price in combination with leisurely service.	0*0	1
It's the cola in combination with leisurely service.	*10	2
It's the precise combination of the high price, the cola, and the leisurely service.	010	1

THE NORMAL SITUATION

- **Competing explanations**
- **Inconsistent information**
- **Greater or lesser amounts of evidence (certainty)**

THE 27 SCHEMATA IN GENERATION 0

Schema		Gen 0	
#	H	m(H,0)	f(H,0)
1	000	0	0
2	001	1	1
3	00*	1	1
4	010	1	2
5	011	1	3
6	01*	2	2.5
7	0*0	1	2
8	0*1	2	2
9	0**	3	2
10	100	0	0
11	101	0	0
12	10*	0	0
13	110	1	6
14	111	0	0
15	11*	1	6
16	1*0	1	6
17	1*1	0	0
18	1**	1	6
19	*00	0	0
20	*01	1	1
21	*0*	1	1
22	*10	2	4
23	*11	1	3
24	*1*	3	3.67
25	**0	2	4
26	**1	2	2
27	***	4	3
Tot		32	96
Av			3.00
#		20	20

AFTER CREATION OF MATING POOL

Schema		Gen 0		Mating pool	
#	**H**	**m(H, 0)**	**f(H,0)**	**m(H, MP)**	**f(H,MP)**
1	**000**	**0**	**0**	**0**	**0**
2	**001**	**1**	**1**	**0-**	**0**
3	**00***	**1**	**1**	**0-**	**0**
4	**010**	**1**	**2**	**1**	**2**
5	**011**	**1**	**3**	**1**	**3**
6	**01***	**2**	**2.5**	**2**	**2.5**
7	**0*0**	**1**	**2**	**1**	**2**
8	**0*1**	**2**	**2**	**1-**	**3**
9	**0****	**3**	**2**	**2-**	**2.5**
10	**100**	**0**	**0**	**0**	**0**
11	**101**	**0**	**0**	**0**	**0**
12	**10***	**0**	**0**	**0**	**0**
13	110	1	6	2+	6
14	**111**	**0**	**0**	**0**	**0**
15	11*	1	6	2+	6
16	1*0	1	6	2+	6
17	**1*1**	**0**	**0**	**0**	**0**
18	1**	1	6	2+	6
19	***00**	**0**	**0**	**0**	**0**
20	***01**	**1**	**1**	**0-**	**0**
21	***0***	**1**	**1**	**0-**	**0**
22	*10	2	4	3+	4.67
23	***11**	**1**	**3**	**1**	**3**
24	*1*	3	3.67	4+	4.25
25	**0	2	4	3+	4.67
26	****1**	**2**	**2**	**1-**	**3**
27	*****	**4**	**3**	**4**	**4.25**
Tot		**32**	**96**	**32**	**136**
Av			**3.00**		**4.25**
#		**20**	**20**	**16**	**16**

AFTER CROSSOVER (GENERATION 1)

Schema		Gen 0		Mating pool		Generation 1	
#	**H**	**m(H, 0)**	**f(H,0)**	**m(H, MP)**	**f(H,MP)**	**m(H, 1)**	**f(H,1)**
1	**000**	**0**	**0**	**0**	**0**	**0**	**0**
2	**001**	**1**	**1**	**0**	**0**	**0**	**0**
3	**00***	**1**	**1**	**0**	**0**	**0**	**0**
4	010	1	2	1	2	2+	2
5	**011**	**1**	**3**	**1**	**3**	**0-**	**0**
6	**01***	**2**	**2.5**	**2**	**2.5**	**2**	**2**
7	**0*0**	**1**	**2**	**1**	**2**	**0-**	**0**
8	0*1	2	2	1	3	2+	2
9	**0****	**3**	**2**	**2**	**2.5**	**2**	**2**
10	**100**	**0**	**0**	**0**	**0**	**0**	**0**
11	**101**	**0**	**0**	**0**	**0**	**0**	**0**
12	**10***	**0**	**0**	**0**	**0**	**0**	**0**
13	**110**	**1**	**6**	**2**	**6**	**1-**	**6**
14	111	0	0	0	0	1+	7
15	**11***	**1**	**6**	**2**	**6**	**2**	**6.5**
16	**1*0**	**1**	**6**	**2**	**6**	**1-**	**6**
17	1*1	0	0	0	0	1+	7
18	**1****	**1**	**6**	**2**	**6**	**2**	**6.5**
19	***00**	**0**	**0**	**0**	**0**	**0**	**0**
20	***01**	**1**	**1**	**0**	**0**	**0**	**0**
21	***0***	**1**	**1**	**0**	**0**	**0**	**0**
22	***10**	**2**	**4**	**3**	**4.67**	**3**	**3.3**
23	***11**	**1**	**3**	**1**	**3**	**1**	**7**
24	***1***	**3**	**3.67**	**4**	**4.25**	**4**	**4.25**
25	****0**	**2**	**4**	**3**	**4.67**	**3**	**3.3**
26	****1**	**2**	**2**	**1**	**3**	**1**	**7**
27	*****	**4**	**3**	**4**	**4.25**	**4**	**4.25**
Tot		**32**	**96**	**32**	**136**	**32**	**136**
Av			**3.00**		**4.25**		**4.25**
#		**20**	**20**	**16**	**16**	**16**	**16**

TURING - 1948
" *COMBINATION OF GENES*"

- **1948 essay "Intelligent Machines" (in *Mechanical Intelligence: Collected Works of A. M. Turing*, 1992)**

> **"There is the genetical or evolutionary search by which a *combination of genes* is looked for, the criterion being the survival value."**

DEFINITION OF THE GENETIC ALGORITHM (GA)

The *genetic algorithm* is a probabalistic search algorithm that iteratively transforms a set (called a *population*) of mathematical objects (typically fixed-length binary character strings), each with an associated fitness value, into a new population of offspring objects using the Darwinian principle of natural selection and using operations that are patterned after naturally occurring genetic operations, such as crossover (sexual recombination) and mutation

PROBABILISTIC

- **The initial population is typically random**
- **Probabilistic selection based on fitness**
  - **Best is not always picked• Worst is not necessarily excluded**
  - **Better individuals are preferred**
  - **Nothing is guaranteed**
  - **Mixture of greedy exploitation and adventurous exploration**
  - **Similarities to simulated annealing (SA)**
- **Random picking of mutation and crossover points**
- **Often, there is probabilistic scenario as part of the fitness measure**
- **In many implementations, the choice of the genetic operations are also probabilistic**

THERE ARE MANY DIFFERENT POSSIBLE RUNS

RUN OF GA — WHEN THE McDONALD'S RESTAURANT (111) IS THE GLOBAL OPTIMUM (WITH C2 AND M3)

	Generation 0			**Mating pool**		**Generation 1**		
1	011	3	.25	011	3	C2	**111**	**7**
2	001	1	.08	110	6	C2	010	2
3	110	6	.50	110	6	R	110	6
4	010	2	.17	010	2	M3	011	3^+
Total		12			17			18
Worst		1			2			2
Aver		3.00			4.25			4.5
Best		6			6			7

2ND RUN OF THE GA — WITH McDONALD'S RESTAURANT (111) BEING THE GLOBAL OPTIMUM (WITH C1 AND M2)

	Generation 0			Mating pool		Generation 1		
1	011	3	.25	011	3	C1	010	2
2	001	1	.08	110	6	C1	**111**	**7**
3	110	6	.50	110	6	R	110	6
4	010	2	.17	010	2	M2	000	0^-
Total		12			17			15
Worst		1			2			0
Aver		3.00			4.25			3.75
Best		6			6			7

- **Note mutation did not improve things in this particular run (common)**
- **Note average fitness decreased (unusual)**

3RD RUN OF THE GA — WITH McDONALD'S RESTAURANT (111) BEING THE GLOBAL OPTIMUM (WITH C1 AND M1)

	Generation 0			Mating pool		Generation 1		
1	011	3	.25	011	3	R	011	3
2	001	1	.08	110	6	C1	110	6
3	110	6	.50	110	6	C1	110	6
4	010	2	.17	010	2	M1	110	6
Total		12			17			21
Worst		1			2			3
Aver		3.00			4.25			5.25
Best		6			6			6

- **Note incestuous crossover of 110 with itself**
- **Note that ¾ of the population is the same (i.e., a lack of diversity)**

THERE ARE MANY DIFFERENT POSSIBLE RUNS

- **Assuming no duplicates, (2^L choose M) = (8 choose 4) = 70 ways of choosing M = 4 individuals out of 8 to create initial random population**

  NOTE: There is no compelling reason to cull out duplicates at generation 0

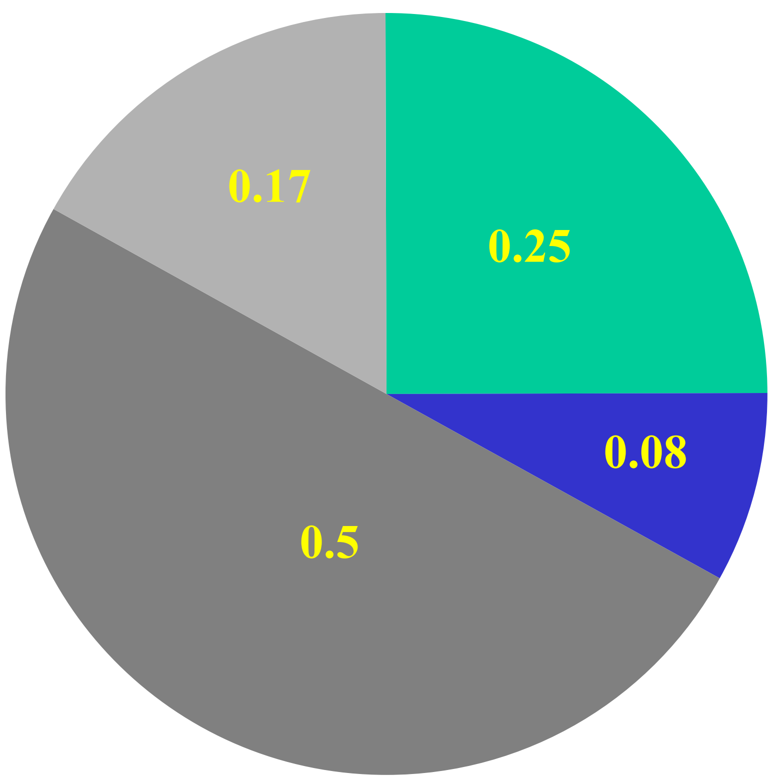
- **M^2 = 16 ways of choosing 2 parents for crossover (because reselection is allowed)**
- **$L - 1$ = 2 ways of choosing crossover point**
- **M = 4 ways of choosing 1 individual from mating pool for mutation**
- **L = 3 ways of choosing mutation point**
- **M = 4 ways of choosing 1 individual from mating pool for reproduction**
- **Total of 107,520 ways of doing just random creation, 1 crossover, 1 mutation, and 1 reproduction for 1 generation of this simple run**

THERE ARE MANY DIFFERENT POSSIBLE FITNESS LANDSCAPES

- **For example, consider 3-bit strings**
- **Consider just rankings of fitness values**
- **Assuming 9 different values of fitness, there are 8! = 362,880/9 = 40,320 different fitness landscapes**
- **Some are easier for the GA than others**

FITNESS MEASURE WHEN THE FRENCH RESTAURANT (000) IS THE GLOBAL OPTIMUM

Strategy	Fitness
000	7
001	6
010	5
011	4
100	3
101	2
110	1
111	0



RUN OF GA — WHEN THE FRENCH RESTAURANT (000) IS THE GLOBAL OPTIMUM (C2 AND M3)

	Generation 0			**Mating pool**		**Generation 1**		
1	011	4	0.25	011	4	C2	001	6
2	001	6	0.38	001	6	C2	011	4
3	110	1	0.07	001	6	R	001	6
4	010	5	0.31	010	5	M3	011	4
Total	16				21			20
Worst	1				4			4
Aver	3.75				5.25			5.00
Best	6				6			6

- **Note mutation did not improve things in this particular run (common)**
- **Note average fitness decreased (unusual)**

FRENCH RESTAURANT — CONTINUING TO GENERATION 2

	Generation 1			Mating pool		Generation 2		
1	001	6	0.30	001	6			
2	011	4	0.20	011	4			
3	001	6	0.30	001	6			
4	011	4	0.20	011	4			
Total		20			20			
Worst		4			4			
Aver		5.00			5.00			
Best		6			6			

- **Note fitness of mating pool didn't increase (unusual)**
- **Note 100% of population now has a 1 in 1st position**

GIRAFFE (*L*=7 CHROMOSOME OVER BINARY *L*=2 ALPHABET)

GOOD GIRAFE

Neck	Tongue	Carnivorous?	Number legs	Leg length	Coloration	Eye color
0 = short 1= long	0 = short 1= long	0 = no 1= yes	0 = 2 legs 1 = 4 legs	0 = short 1= long	0= luminescent 1- brownish	0 = brown 1= black
1	1	1	1	1	1	0 or 1

FOUR MAJOR PREPARATORY STEPS FOR GA

- **determining the representation scheme**
  - **structure**
  - **if the structure is fixed-length string, then alphabet size *K* and string length *L***
  - **mapping from a point in search space of the problem to a structure, and vice versa**
- **determining the fitness measure**
- **determining the control parameters**
  - **population size *M***
  - **number of generations *G***
  - **other control parameters**
- **determining the method for designating a result (e.g., best-so-far) and the criterion for terminating a run**

THE FOUR PREPARATORY STEPS FOR FOR THE HAMBURGER RESTAURANT PROBLEM

1. REPRESENTATION SCHEME

- **Fixed-length string, $K = 2$, $L = 3$**
- **Mapping:**

Left bit (Price):
1 = \$ 0.50 price
0 = \$10.00 price
Middle bit (Drink):
1 = Coca Cola
0 = Wine
Right bit (Ambiance):
1 = Fast service
0 = Leisurely service

THE FOUR GA PREPARATORY STEPS FOR THE HAMBURGER RESTAURANT PROBLEM — CONTINUED

2. FITNESS MEASURE

- **Profit in dollars**

THE FOUR GA PREPARATORY STEPS FOR THE HAMBURGER RESTAURANT PROBLEM — CONTINUED

3. CONTROL PARAMETERS FOR THE RUN

- **Major parameters**
  - **Population size, *M* = 4**
  - **Max number of generations, *G* = 6**
- **Secondary parameters**
  - **Probability of crossover, P_c = 90% is typical, but P_c = 50% for this small population (i.e., 2 = 50% of *M* = 4)**
  - **Probability of mutation, P_m = 1% is typical, but P_c = 25% for this small population**
  - **Probability of reproduction, P_r = 9% is typical, but P_r = 25% for this small population**

THE FOUR GA PREPARATORY STEPS FOR THE HAMBURGER RESTAURANT PROBLEM — CONTINUED

4. TERMINATION CRITERION AND RESULT DESIGNATION

- **Termination Criterion:** Global maximum (known to be 7) attained OR total of G = 6 generations have been run
- **Method of Results Designation:** Best-so-far (cached) individual from population

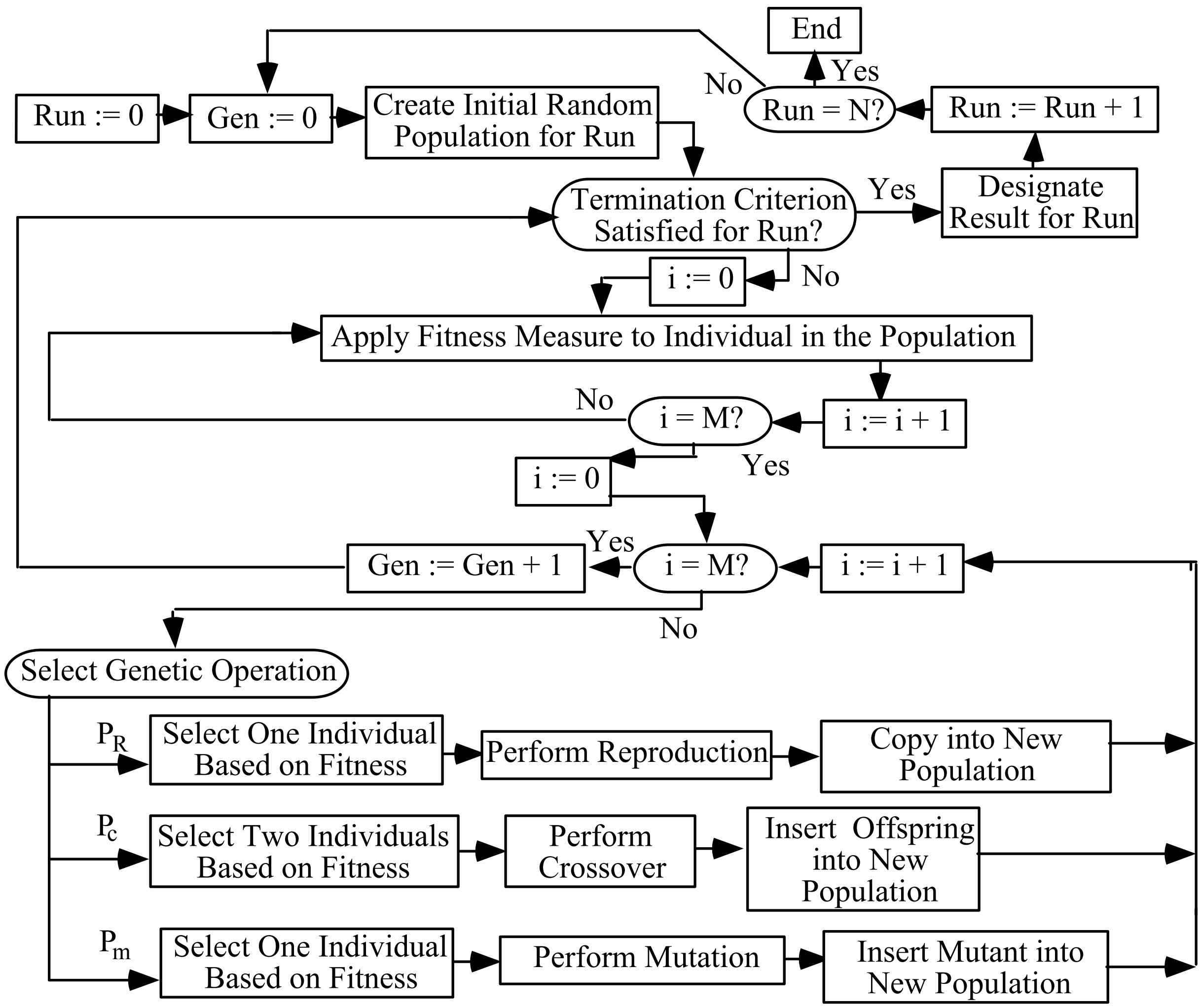
TABLEAU FOR HAMBURGER PROBELM

Objective:	**Find the globally optimum management strategy (consisting of 3 binary decisions) for running a hamburger restaurant.**
Representation scheme:	**• structure = fixed length string** **• alphabet size *K* = 2 (binary)** **• string length *L* = 3** **• mapping = each bit is binary decision**
Fitness cases:	**Only one.**
Raw fitness:	**The profit (0 – 7) derived from operating a hamburger restaurant with a particular strategy.** **Bigger is better. A fitness of 7 is best.**

Standardized fitness:	$f_{standardized} = f_{maximum} - f_{raw}$ $f_{standardized} = 7 - f_{raw}$ **Less is better. A fitness of 0 is best.**
Parameters:	**• Population size *M* = 4.** **• Maximum number of generations to be run *G* = 6.** **• Probability of crossover, P_c = 50%** **• Probability of mutation, P_m = 25%** **• Probability of reproduction, P_r = 25%**
Termination criteria:	**The GA has run for G_{max} generations OR the success predicate of the problem has been satisfied by the best-so-far individual in the population. The success predicate is that the known global optimum of 7 has been achieved.**

Result designation:	**The best-so-far (cached) individual in the population.**

FLOWCHART FOR GENETIC ALGORITHM



ELEMENTS OF GA FLOWCHART

- **Creation of the initial population (called *generation 0*) — usually random**
- **Evaluate fitness of each individual in the population for the current generation**
- **Select genetic operation (conventionally done probabilistically)**
  - **mutation (perhaps 1%)**
  - **crossover (perhaps 90%)**
  - **reproduction (perhaps 9%)**
- **Select one individual from the population probabilistically based on fitness (two if the operation is crossover)**
- **Perform the genetic operation**
- **Insert offspring into population**
- **Termination criterion**
- **Results designation**

- **NOTE: There are many (minor) variations on the GA**

GENERATION 0 = BLIND RANDOM SEARCH

	Gen 0	
1	011	3
2	001	1
3	110	6
4	010	2
Total		12
Worst		1
Aver		3.0
Best		6

- **Some individuals are better than others**
- **Produces estimate of average fitness of the population as a whole**

SOME INDIVIDUALS ARE (USUALLY) BETTER THAN OTHERS

"I think it would be a most extraordinary fact if no variation ever had occurred useful to each being's own welfare ... .

… But if variations useful to any organic being do occur, assuredly individuals thus characterised will have the best chance of being preserved in the struggle for life; …

… and from the strong principle of inheritance they will tend to produce offspring similarly characterised. …

…This principle of preservation, I have called, for the sake of brevity, Natural Selection."

--- Charles Darwin in *On the Origin of Species by Means of Natural Selection* (1859)

MATING POOL (DARWINIAN SELECTION) PRODUCED BY DARWINIAN FITNESS PROPORTIONATE SELECTION (FPR)

	Gen 0			Mating pool	
1	011	3	.25	011	3
2	001	1	.08	110	6
3	110	6	.50	110	6
4	010	2	.17	010	2
Total		12			17
Worst		1			2
Aver		3.0			4.25
Best		6			6

- **Improved average fitness of population**
- **Improved worst individual**
- **Less diversity**
- **Nothing new**

GA MUTATION OPERATION

One parental string chosen probabilistically based on fitness

Parent
010

Point 3 chosen at random as the mutation point

Parent
010

One offspring produced by the mutation operation

Offspring
011

- **VERY occasional**

GA CROSSOVER OPERATION

Two parental (fixed-length) strings chosen probabilistically based on fitness

Parent 1	Parent 2
011	110

Two crossover fragments (Interstitial point 2 chosen at random as the crossover point)

Crossover fragment 1	Crossover fragment 2
01–	11–

Two remainders

Remainder 1	Remainder 2
– – 1	– – 0

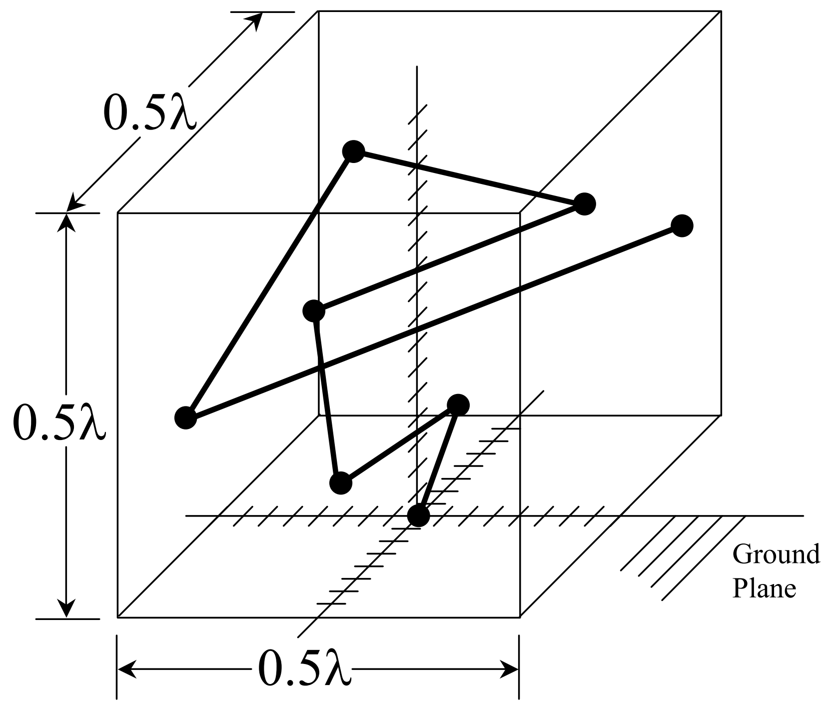
Two offspring produced by crossover operation

Offspring 1	Offspring 2
111	010

THE THREE GA OPERATIONS

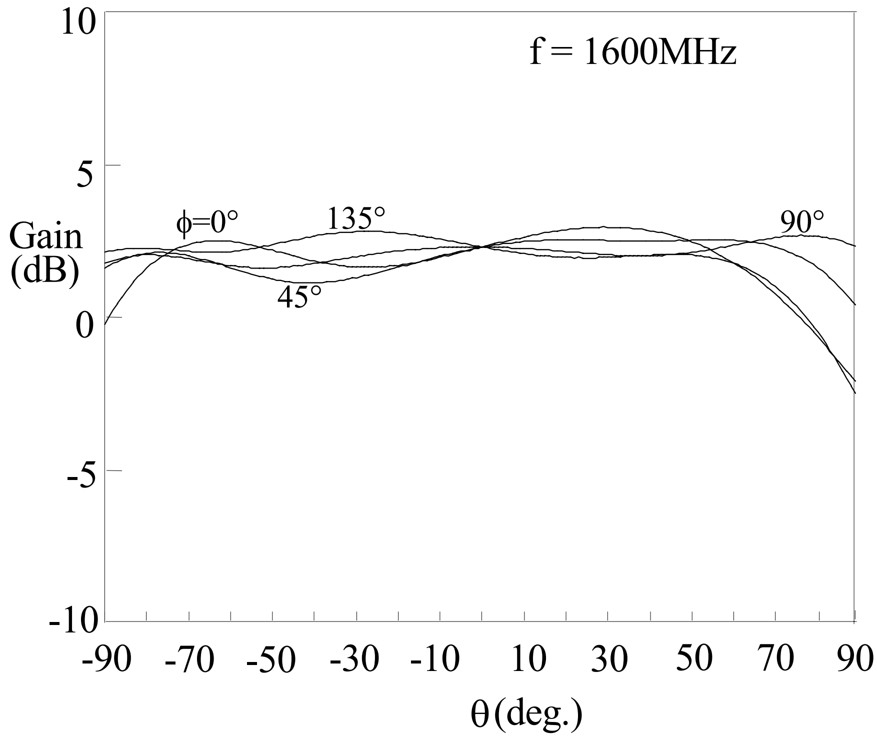
- **The individuals participating in all three genetic operations are chosen probabilistically based on fitness**
- **Reproduction reproduces traits (and combinations of traits) that are well adapted to the environment (the fitness measure)**
- **Mutation creates local variations. If the character string is binary, the changes are at Hamming distance 1 (or a low Hamming distance). Mutation preserves most combinations of traits that are known to be relatively good because it affects 1 (or only a couple) of positions in the chromosome.**
- **Crossover is disruptive (and creative).Crossover creates large variations, while preserving many <u>combinations of traits</u> that are known to be relatively good — particularly combinations that are nearby on the chromosome.**

WIRE ANTENNA DESIGN (ALTSHULER AND LINDEN 1998)



- **The problem is to determine the X-Y-Z coordinates of the three-dimensional position of the ends (X1, Y1, Z1, X2, Y2, Z2,… , X7, Y7, Z7) of seven straight wires so that the resulting seven-wire antenna satisfies certain performance requirements**
- **The first wire starts at feed point (0, 0, 0) in the middle of the ground plane**
- **The antenna must fit inside the 0.5λ cube**

WIRE ANTENNA DESIGN — CONTINUED



- **Antenna is for ground-to-satellite communications for cars and handsets**
- **We desire near-uniform gain pattern 10° above the horizon (for a right-hand circularly polarized signal)**
- **Fitness is measured based on the antenna's radiation pattern (over an infinite ground plane). The radiation pattern is computed using standard techniques (such as the NEC or National Electromagnetics Code)**

WIRE ANTENNA DESIGN — CONTINUED

• Since we want a near-uniform gain pattern, fitness is sum of the squares of the difference between the average gain and the antenna's gain.

• This sum is taken for angles Θ between -90° and +90° and all azimuth angles Φ from 0° to 180°

• The smaller the value of fitness, the better

105-BIT CHROMOSOME (GENOME)

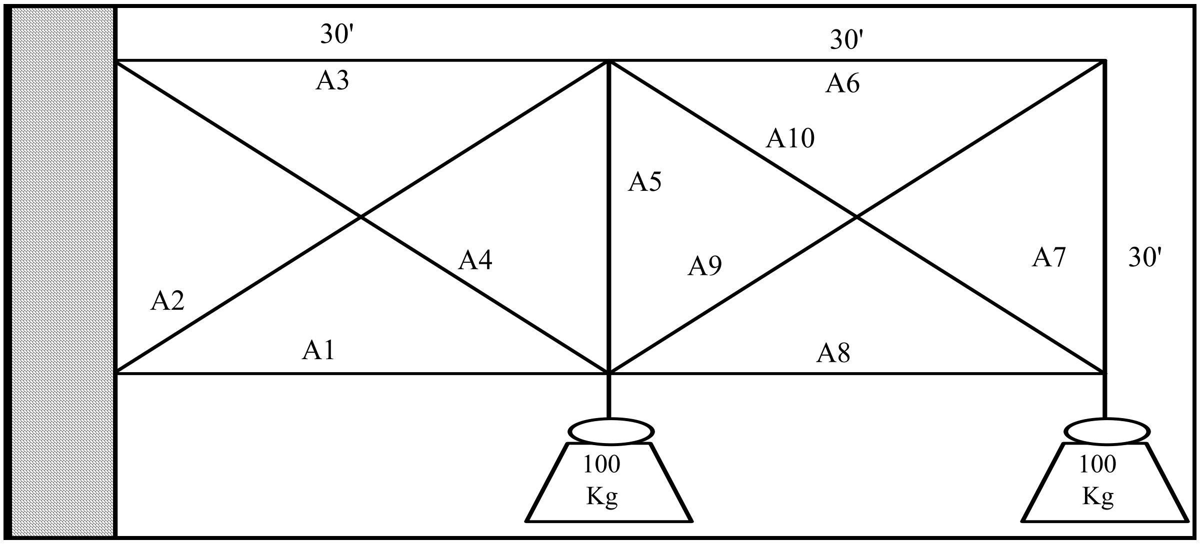
X1	Y1	Z1	X2	Y2	Z2	
+0010	-1110	+0001	+0011	-1011	+0011	etc.

• Each coordinate is represented as sign plus 4 bits

• Total chromosome is $3 \times 7 \times 5 = 105$ bits

• Population size is 500

10-MEMBER TRUSS PROBLEM (GOLDBERG AND SAMTANI 1986)



- **Truss has 10 members (six of length of 30 feet and four of length $30\sqrt{2} = 41$ feet)**
- **Prespecified topological arrangements of the 10 members, the load, and the wall**

Goldberg, David E. and Samtani, M.P. Engineering optimization via genetic algorithms. In *Proceedings of the Ninth Conference on Electronic Computation*. 1986. Pages 471-482.

10-MEMBER TRUSS PROBLEM — CONTINUED

- **The problem is to determine the cross-sectional areas (A1, … , A10) of each of the 10 members so as to minimize dollar cost of building a truss that supports the two loads**
- **The cost of material depends on volume of material used (i.e., the sum, over the 10 members, of the length of each member times its cross-sectional area)**
- **The truss must support the two masses**
- **There is a stress on each of the 10 members**
- **Thin members cost little, but would break**
- **Stress constraints must be satisfied**

FOUR MAJOR PREPARATORY STEPS 10-MEMBER TRUSS PROBLEM

1. REPRESENTATION SCHEME

40-BIT CHROMOSOME (GENOME)

A1	A2	A3	A4	A5	A6	A7	A8	A9	A10
0010	**1110**	**0001**	**0011**	**1011**	**0011**	**1111**	**0011**	**0011**	**1010**

- **Map the 4-bit string 0000 into 1-inch cross-sectional area**
- **Map 1111 into 6-inches**
- **Map other 14 4-bit strings proportionately**

- **Alternatively, map the 16 possible 4-bit strings into the 16 commercially available sizes of the material**

FOUR MAJOR PREPARATORY STEPS 10-MEMBER TRUSS – CONTINUED

2. FITNESS MEASURE

- **Two-part (multiobjective) fitness measure**
- **Decode the 40-bit chromosome into the 10 cross-sectional areas: A_1, A_2, ..., A_{10}.**
- **Compute the volume of each member of the truss as its cross-sectional area times its length (30 feet or $30\sqrt{2} = 42$ feet)**
- **Compute cost of each member**
- **Compute the sum, over the 10 members, the cost to get the total cost.**
- **Stresses are computed using standard mechanical engineering techniques.**
- **Penalize violations of stress constraints. For example, a stress that is 10% above the maximum set by the constraint for that member might be penalized (e.g., 110%).**
- **The smaller the total cost the better. The minimal cost is not known in advance (for purposes of the termination criterion).**

FOUR MAJOR PREPARATORY STEPS 10-MEMBER TRUSS – CONTINUED

3. MAJOR PARAMETERS

- **Population size, *M* = 200**
- **Maximum number of generations to be run, *G* = 50**

4. TERMINATION

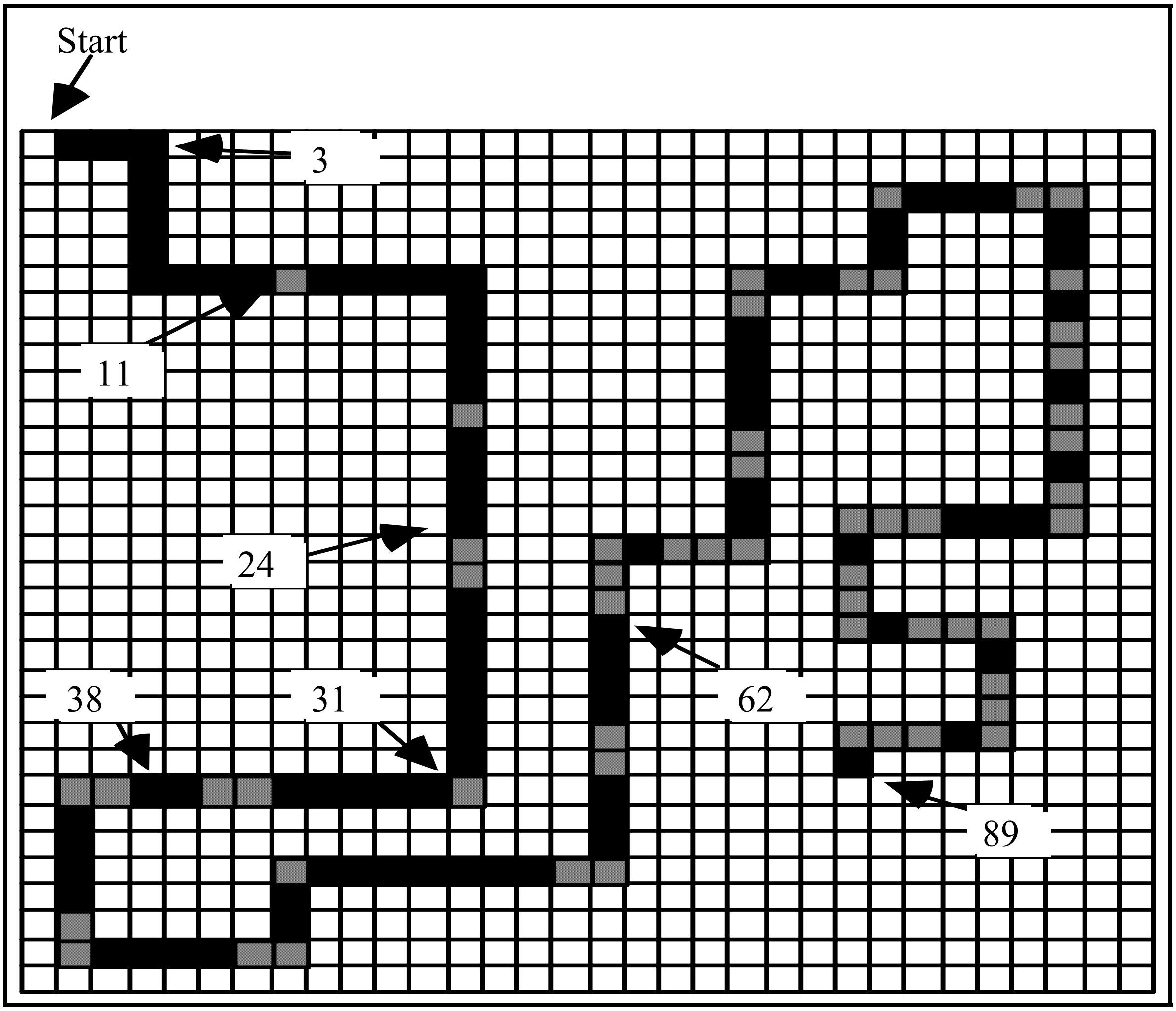
- **The minimal cost is not known in advance. The criterion for terminating a run (e.g., maximum number of generations to be run or plateau in fitness of best-of-generation individuals**
- **Method for designating a result is "best-so-far" individual**

GA TABLEAU FOR 10-MEMBER TRUSS

Objective:	**Find the globally optimum combination of cross-sectional areas for the 10 members of the truss.**
Representation scheme:	**• structure = fixed length string** **• alphabet size *K* = 2 (binary)** **• string length *L* = 40** **• mapping = each 4-bit group of the 40-bit string corresponds to the cross-sectional area (with granularity of 16) of one of the 10 members of the truss.**
Fitness cases:	**Only one.**
Raw fitness:	**Raw fitness = cost (weight) of 10 members with penalty for constraint violations (uses packaged evaluation program). Less is better.**

Standardized fitness:	**Same as raw fitness.**
Parameters:	**• Population size *M* = 200.** **• Maximum number of generations to be run *G* = 50.**
Termination criteria:	**The GA has run for *G* generations.**
Result designation:	**The best-so-far individual in the population.**

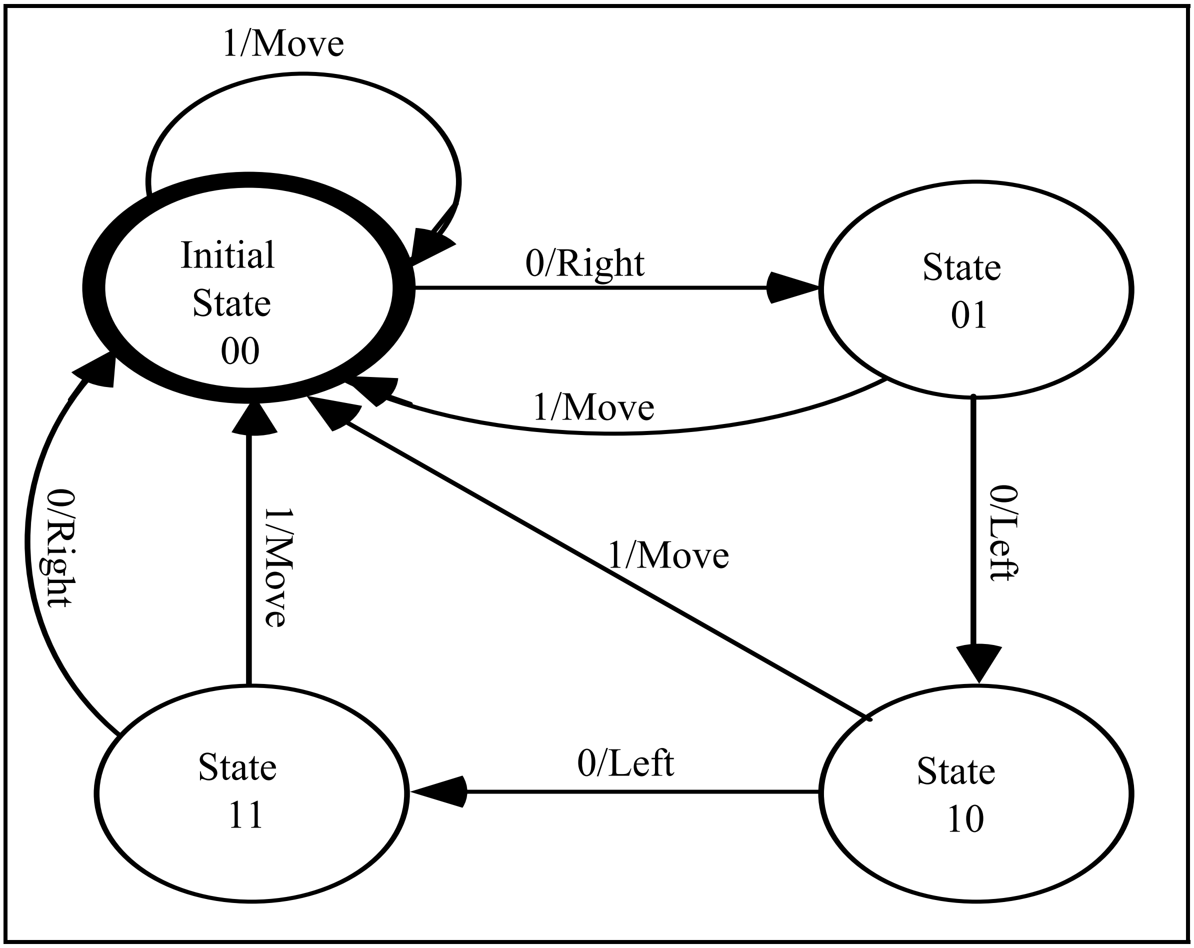
ARTIFICIAL ANT – SANTA FE TRAIL



ARTIFICIAL ANT PROBLEM

- **The state of the ant is its position and facing direction**
- **At time step 0, ant starts in upper left corner facing east**
- **The any can perform 4 operations**
  - Ant can move forward
  - Ant can also turn right or turn left at each time step (thereby changing its facing direction)
  - Ant can perform a no-op (neither moving nor turning)
- **Ant can also sense what is immediately ahead of it in its facing direction and take conditional action based on whether or not food is present**
- **Grid is toroidal**
- **The goal is to eat all 89 pieces of food on the trail**
- **The trail has single, double, and triples gaps, including gaps at corners**

STATE TRANSITION DIAGRAM FOR 4-STATE FINITE AUTOMATON



STATE TRANSITION TABLE FOR 4-STATE FINITE AUTOMATON

	Current state	Input	New state	Operation
1	00	0	01	10=Right
2	00	1	00	11=Move
3	01	0	10	01=Left
4	01	1	00	11=Move
5	10	0	11	01=Left
6	10	1	00	11=Move
7	11	0	00	10=Right
8	11	1	00	11=Move

34-BIT CHROMOSOME (GENOME)

00	**0110**	**0011**	**1001**	**0011**	**1101**	**0011**	**0010**	**0011**

ARTIFICIAL ANT PROBLEM

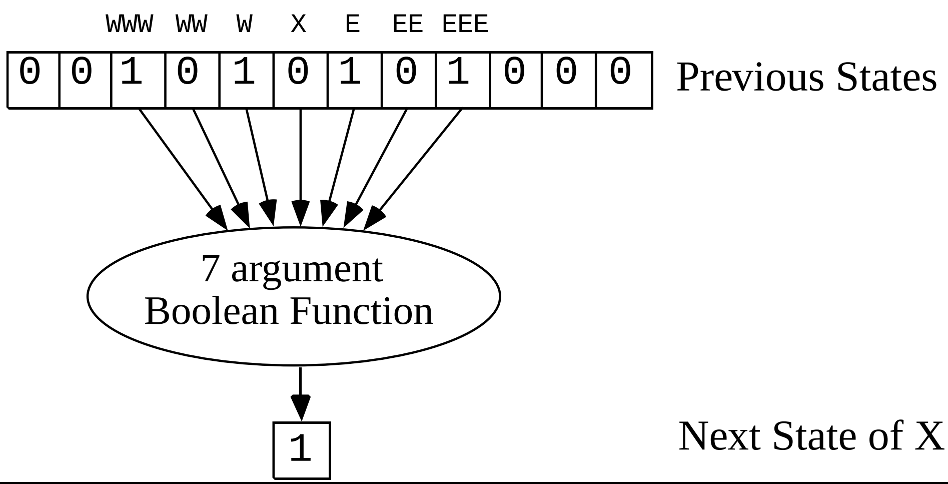
- **A four-state finite automaton is not sufficient to solve this problem (Jefferson, Collins, et al. 1991). It does into infinite loop when there is a single gap in food trail.**
- **This is a reminder that the choice of the representation may help or hinder (and may even preclude) finding a solution to the problem**
- **A 32-state automaton (453-bit genome) is more than sufficient. (453 = 64 substrings of length 7 plus 5 additional bits representing the initial state, out of 32 possible states)**
- **Similarly, the choice of the granularity of the coding of floating-point numbers may also affect the ability of the GA to find a solution (e.g., 4 bits for the 10-member truss, 5 bits for each coordinate of the endpoint of the antenna wires)**

GA TABLEAU FOR THE ARTIFICIAL ANT PROBLEM

Objective:	**Find a control strategy for the artificial ant**
Representation scheme:	• **structure = fixed length string** • **alphabet size *K* = 2 (binary)** • **string length *L* = 34 (for 4-state automaton) and *L* = 453 (for 32-state automaton)** • **mapping (for the 4-state automaton): The first 2 bits are the initial state and each 4-bit group is a line of state transition table of the automaton.**
Fitness cases:	**Only one.**
Raw fitness:	**Raw fitness = amount of food eaten (comes from running the simulation).**

Parameters:	**• Population size *M* = 65,536.** **• Maximum number of generations to be run *G* = 200.**
Termination criteria:	**The GA has run for *G* generations OR the best-so-far individual scores 89.**
Result designation:	**The best-so-far individual in the population.**

CELLULAR AUTOMATA RULE DISCOVERY USING GA



#	WWW	WW	W	X	E	EE	EEE	**Rule**
0	0	0	0	0	0	0	1	$\mathbf{a}_0$
1	0	0	0	0	0	1	0	$\mathbf{a}_1$
2	0	0	0	0	1	0	0	$\mathbf{a}_2$
3	0	0	0	0	1	1	0	$\mathbf{a}_3$
4	0	0	1	1	0	0	0	$\mathbf{a}_4$
…	…	…	…	…	…	…	…	**…**
127	1	1	1	1	1	1	01	$\mathbf{a}_{127}$

128-BIT CHROMOSOME (GENOME)

A0	A1	A2	…	A127
a_0	a_1	a_2	…	a_{127}

PATCH ANTENNA DESIGN

GENOME OF N^2 BITS

1	**1**	**0**	**1**	**1**	**1**	**1**
1	**1**	**1**	**1**	**0**	**1**	**1**
1	**0**	**1**	**1**	**1**	**0**	**0**
1	**1**	**1**	**1**	**1**	**1**	**1**
1	**1**	**0**	**1**	**1**	**1**	**1**
1	**1**	**1**	**1**	**0**	**1**	**1**
1	**1**	**0**	**1**	**1**	**1**	**0**

Johnson, J. Michael and Rahmat-Samii, Yahya. 1999. Genetic algorithms and method of moments (GA/MOM) for the design of integrated antennas. *IEEE Transactions on Antennas and Propagation*. 47(10) 1606 - 1614. October 1999.

ITERATED PRISONER'S DILEMMA GAME

		Player 2	
		Cooperate	**Defect**
Player 1	**Cooperate**	(3,3)	(0,5)
	Defect	(5,0)	(1,1)

ITERATED PRISONER'S DILEMMA GAME — CONTINUED

- **71-bit genome**
- **64 bits specify player 1's next action, given that player 1 took 3 previous actions and player 2 took 3 previous actions ($8 \times 8 = 64$)**
- **On move 3 only, 4 bits specify player 1's next action, given that player 1 took 2 previous actions and player 2 took 2 previous actions ($2 \times 2 = 4$)**
- **On move 2 only, 2 bits specify player 1's next action, given that player 1 took 1 previous action and player 2 took 1 previous action**
- **On move 1 only, 1 bit specifies player 1's action**

ARTIFICIAL ANT WITH NEURAL NET

- **Jefferson, Collins, et al. (1991) also successfully searched for and discovered a multilayer neural net enabling the artificial ant to traverse a food trail.**
- **Neural networks consist of processing elements that are connected with various weighted signal lines.**

ARTIFICIAL ANT WITH NEURAL NET — CONTINUED

- **Jefferson, Collins, et al. started by deciding that the neural net would have**
  - **two linear threshold processing elements in the input layer (representing the two possible sensory inputs of the ant),**
  - **four linear threshold processing elements in the output layer (for the four possible operations of the ant), and**
  - **five linear threshold processing elements in the hidden layer (AN IMPORTANT CHOICE)**

ARTIFICIAL ANT WITH NEURAL NET — CONTINUED

- **Jefferson, Collins, et al. also decided that the network would be fully connected between consecutive layers in the forward direction, and they decided that the output of each processing element of the hidden layer would feed back into all processing elements of that layer.**

- **Consequently, the five processing elements in the hidden layer and the four processing elements in the output layer each had seven inputs (the outputs from both processing elements of the input layer and the outputs from all five processing elements of the hidden layer).**

ARTIFICIAL ANT WITH NEURAL NET — CONTINUED

- **Once the arrangement of linear processing elements and their connections is established, a neural net is defined by the values of various floating-point numbers representing the weights on various signal lines connecting the various linear processing elements, the thresholds of the linear processing elements, and the initial activation levels of the linear processing elements.**
- **The representation scheme for a neural network can therefore be a binary string of 520 bits that encodes this set of floating-point numbers. As such, it is similar to the "vanilla" representation scheme used by Goldberg and Samtani for the ten-member truss.**
- **Using a population size of 65,536, Jefferson, Collins, et al. were successful in finding a neural network to solve the artificial ant problem.**

WAYS TO MEASURE FITNESS

- **Simulators (e.g., SPICE, NEC, FEA)**
- **Evaluate fitness by operating a tethered robot in actual environment for a certain period of time (e.g., 30 seconds)**
- **Evaluate fitness of electrical filter circuit by connecting computer to a bread-board with the circuit and collecting outputs for particular inputs**
- **Evaluate fitness by using Field Programmable Gate Arrays (FPGA) or Field Programmable Transistor Array (FPTA) to process thousands (or millions) of fitness cases (combinations of inputs) at very high speed**
- **Laboratory experiments lasting a day or longer for each generation of a run**

THE "ART" OF GA'S

- **Finding a chromosomal representation of your problem**

- **Finding a chromosomal representation with the right genetic linkages (i.e., the crossover operator is congenial to the problem).**

- **Finding a good fitness measure**

CHARACTERISTICS OF GA

- **Works with a population (Alternative disjunctive solutions)**
- **Stochastic**
- **Operates from the fitness measure only**
- **Weak assumptions about fitness measure**
- **Operates on a coding**
- **Interim performance is important**
- **No central memory or bookkeeping**
- **Audit trails**
- **Robust (when environment changes)**
- **Incremental learning possible**
- **Modeled after a known working system**
- **Highly parallelizable, Asynchronous, Fault-tolerant, Decentralized local control**
- **Implicit Parallelism**
- **Optimal allocation of trials**

OTHER FORMS AND VARIATIONS OF GENETIC ALGORITHM

- **Variable-length strings**
- **Real-valued genes**
- **Data structures as genomes**
- **Problem-specific crossover operations (particularly for permutation problems or other problems of constrained optimization)**
- **Messy genetic algorithms**
- **Classifier systems**

VARIABLE-LENGTH STRING GENETIC ALGORITHM

Two parental (fixed-length) strings chosen probabilistically based on fitness

Parent 1	Parent 2
000	11111

Randomly and independently choose crossover point for each parent (between 1 and `LENGTH-1` of that parent) yielding two crossover fragments

Crossover fragment 1	Crossover fragment 2
0 - -	1111-

Two remainders

Remainder 1	Remainder 2
- 00	- - - -1

Two offspring produced by crossover operation

Offspring 1	Offspring 2
01	111100

VARIABLE-LENGTH STRING GENETIC ALGORITHM

- **The population contains individuals of different length beginning at generation 0**
- **Typically the parents participating in crossover are of different length (3 and 5 in example)**
- **Typically offspring resulting from crossover differ in length (2 and 6) from each other and from their parents**
- **Even if one individual mates with itself, the offspring are usually of different length (because the crossover points are chosen independently).**

AIRPLANE WING DESIGN USING VARIABLE-LENGTH STRING GENETIC ALGORITHM (U. S. DESIGN PATENT 0363696)

- **Designed new shape for airplane wing**

Gage, Peter J., Kroo, I.M., and Solieski, I. P. 1994. A variable complexity genetic algorithm for topological design. *Proceedings of Symposium on Multidisciplinary Analysis and Optimization*. Washington, DC: American Institute of Aeronautics and Astronautics.

Kroo, Ilan M., McMasters, John H., and Pavek, Richard J. 1995. *Large Airplane with Nonplanar Win*g. U. S. Design Patent number USD0363696. Applied for on June 23, 1993. Issued October 31, 1995.

PARAMETERS FOR CONTROLLING A RUN OF THE GENETIC ALGORITHM

TWO MAJOR NUMERICAL PARAMETERS

Population size *M*

Maximum number *G* of generations to be run

MINOR NUMERICAL PARAMETERS

Probability p_c of crossover = 90%.

Probability p_r of reproduction = 9%.

Probability p_m of mutation = 1%.

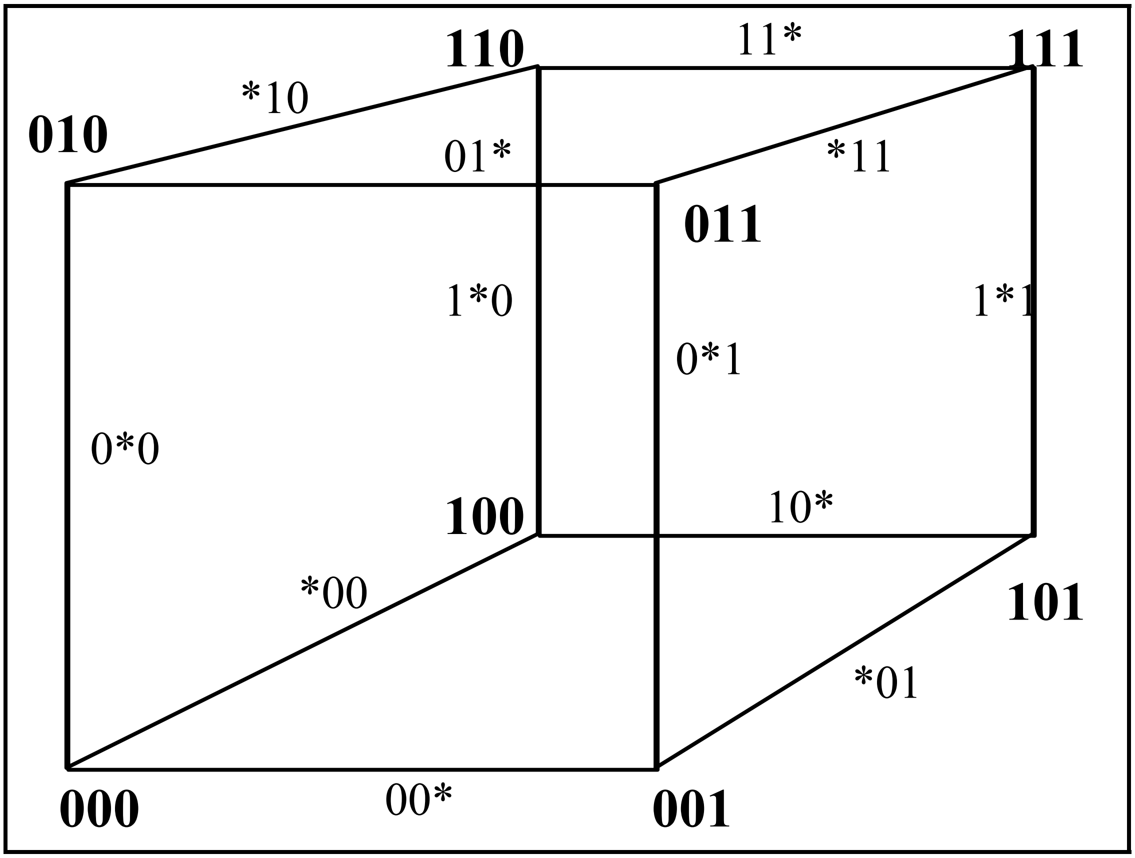
SIX QUALITATIVE VARIABLES

- **Generative method for initial random population: random bit generation.**
- **Basic selection method: fitness proportionate (*cf*. tournament selection)**
- **Spousal selection method: fitness proportionate.**
- **Elitist strategy: Not used.**

COMMON MISTAKES IS APPLYING GENETIC ALGORITHMS

- **Population is MUCH TOO small**
- **Mutation rate is TOO HIGH**
- **Excessive GREED is introduced**
- **Improper initialization**
- **Fitness is not adequately gradated**
- **Hand-crafted crossover operators cause mutation to be introduced for virtually every crossover**
- **Hand-crafted crossover operators are not in sync with the problem**
- **Misapplying rules of thumb**

SEARCH SPACE FOR THE HAMBURGER RESTAURANT PROBLEM



CROSSOVER OPERATION

The offspring 111 of parents 110 and 011 lie at intersection of schema 11* and **1