# Deception in Genetic Search

- Basic objective is to provide guidelines regarding what problem properties Genetic Search strategies, in particular Genetic Algorithms, will find difficult.
- Schema theory attempts to express GA search properties by showing that building blocks (short order and defining length schema) are used to efficiently sample the search space.
    - Low level building blocks identified and recombined to direct search towards above average regions of the search space;
- Problem is said to be deceptive if the building blocks identified actually lead the GA away from the global objective.

## Example

- Consider the following deceptive order – 3 function
    - For deception to take place order-1 and order-2 schema redirect the cases of higher fitness towards a low fitness individual, where schemas are measured genotypically.
    - Let the global optimum be 111; the global minimum be 000.
    - Lower order schema are now ordered to satisfy the following relationships to achieve deception,

| | |
|---|---|
| F(0**) > f(1**) | F(00*) > f(11*), f(01*), f(10*) |
| F(*0*) > f(*1*) | F(0*0) > f(1*1), f(0*1), f(1*0) |
| F(**0) > f(**1) | F(*00) > f(*11), f(*01), f(*10) |

- This might lead to the following specific fitness values for a deceptive function,

| | | |
|---|---|---|
| Deceptive Function 1 | f(000) = 28 | f(001) = 26 |
| | f(010) = 22 | f(100) = 14 |
| | f(110) = 0 | f(011) = 0 |
| | f(101) = 0 | f(111) = 30 |

- Our basic motivation will therefore be to,
    - Provide a framework for building problems with deceptive properties;
    - Assess GA performance on such problems.

## Definitions

- Hyperplanes – geometric interpretation of schema
    - For 3D space,
        - Points $\rightarrow$ schema of order 3 (or order $N$ in an $N$ dimensional space);
        - Lines $\rightarrow$ schema of order 2;
        - Planes $\rightarrow$ schema of order 1.
        - In N dimensional case, 'hyperplanes' of varying order result (points, lines and planes are all special cases of hyperplanes).
        - Schema and hyperplane can therefore be used interchangeably.
- Building Blocks
    - Special case of an above average fitness schema which have low order and defining length;
- Primary hyperplane (schema) competition (of order $n$)
    - The set of $2^n$ schema competitions of order $n$ involve the schema with $n$ bit values in the same location.
    - E.g. **0*0, **0*1, **1*0, **1*1 are the schema in a competition of order 2 primary hyperplanes.
    - Global winner of an order $n$ Primary hyperplane competition is that schema with the highest fitness among the $2^n$ schema.
        - No implication that such a winning schema leads to the globally optimal solution.
- Hyperplane containment
    - Schema $X$ ($s_x$) contains schema $Y$ ($s_y$) iff $o(s_y) > o(s_x)$
    - Concept leads to a hierarchy of primary hyperplane competitions where a lower order schema may contain a competition of a higher order schema.

- Deception
    - Lower order competitions provide hyperplane competition winners that do not correspond to the bit values of the global winner at order *n*.
- Deceptive Problem
    - There exist low order hyperplane competitions that *have the potential to lead* the genetic search process away from the hyperplane competition at order *n*.
- Fully Deceptive Problem (or subproblem) of order *n*
    - All lower order hyperplanes lead towards the same hyperplane of order *n*, which is not a global winner.
        - Such a hyperplane is a *deceptive attractor*.
        - Will see later that such an attractor corresponds to a schema with a bit pattern the complement of the actual global winner of a hyperplane competition at order *n*.
        - E.g. for an order-3 competition in which the global winner is **1*1**, the deceptive attractor is **0*0**.
- Consistently Deceptive Problem (or subproblem) of order *n*
    - Only the order 1 hyperplanes result in a deceptive attractor, where this may effect our ability to determine the global solution at order *n*.
    - Naturally, a fully deceptive problem is *always* Consistently Deceptive, but not necessarily the reverse.
- Deceptive Function
    - A consistently deceptive problem in which the number of bits encoding the solution space is also the order of the deception.
    - A deceptive function has the potential to be fully deceptive.
- Deceptive Building Block of order *n*
    - A schema, *H*, has a fitness higher than its' competitors, but all lower order schema at the same locations as *H* are misleading.
    - Results in the genetic search being lead away from the fit schema *H*.

## Only Challenging Problems are Deceptive

- Theorem [1]:
    - "Given a fitness function for a problem representing some optimization task with a binary encoding of length *L*,

    **if**

    1) no deception occurs in any of the hyperplanes associated with that particular binary encoding and
    2) the winners of the *L* order-1 hyperplanes can be correctly determined,

    **then**

    the global optimum of the function is determined by the one string contained in the intersection of the *L* order-1 hyperplane competition winners."
- What does this mean?
- A deceptive problem will always produce at least two primary hyperplane competitions (as in two different *k*-arm bandit problems) whose solutions are different bit patterns.
    - The Schema Theorem predicts an exponential increase in the number of reproductive trials provided to a schema winner.
    - Such a winner, by definition cannot identify the solution to both hyperplane competitions.
- Implication
    - It is not possible, or desirable, to solve all hyperplane competitions correctly.
    - So long as the majority of the hyperplane competitions resolve in favor of the ideal objective, there is a good chance that the required solution will be found.

## Construction of fully deceptive functions

- Algorithm for constructing fully deceptive functions of order > 2 on binary gray coded representation;
    - Sort binary strings in terms of relative distance in a Hamming Space;
    - Number strings 1 to *N*;
    - String #1 is the global optimum;
    - String #*N* is the deceptive attractor;
    - Let string #2 take the fitness value '*B*';

- For $i$ = 3 to N
  - $F_d$(String(#$i$)) = $F_d$(String(#$i$ – 1)) + C;
- $F_d$(String(#$1$)) = $F_d$(String(#$N$)) + C;

- Notes
  - The sort operation of step 1 provides a Binomial distribution of 1's and 0's.
  - $F_d(\cdot)$ is the 'deceptive' function;
- Example Fully Deceptive Function
  - Order-4 fully deceptive function with $B$ = 0; $C$ = 2.

`Deceptive Function 2`

| | | | |
|---|---|---|---|
| `f(1111) = 30` | `f(0100) = 22` | `f(0110) = 14` | `f(1110) = 6` |
| `f(0000) = 28` | `f(1000) = 20` | `f(1001) = 12` | `f(1101) = 4` |
| `f(0001) = 26` | `f(0011) = 18` | `f(1010) = 10` | `f(1011) = 2` |
| `f(0010) = 24` | `f(0101) = 16` | `f(1100) = 8` | `f(0111) = 0` |

- Notes
  - The deceptive attractor, $s_d$ = 000, is a local optimum in the Hamming space;
  - Deceptive functions 1 and 2 are both fully deceptive;
    - The deceptive attractor has a basin of attraction, which spans the entire (Hamming) space, other than the point of the single isolated global optimum.

## Deceptive Attractor Theorem

- The following theorems will all be *static*,
  - They are based purely on observations regarding the relationship between binary strings and hyperplanes in an $N$-dimensional hypercube.
  - Modelling the GA as a dynamical system may not result in the same conclusions.
- Currently observed that,
  - IF
    - the problem is fully deceptive
  - THEN
    - The deceptive attractor is the complement of the required global optimum;

    - The order N information which would have lead to the global optimum now leads to the deceptive attractor.

- Theorem 2 [1]:
    - "In order for a function or building block of order *n* to be consistently deceptive in all relevant lower-order hyperplanes, the deceptive attractor must be the complement of the string which represents the global optimum in the deceptive function, or in the case of a deceptive building block, the deceptive attractor must be the complement of the schema representing the "global winner" of the relevant primary hyperplane competition at order *n* that is superior to all of its competitors."
- Note,
    - This theorem says nothing about the value of the deceptive attractor in a deceptive function.
    - Thus, does a deceptive attractor have to have a 'high' fitness value?
    - Or, does a deceptive attractor in a fully deceptive function have to represent a local optimum in Hamming space?
- Consider,

`Deceptive Function 3`

| | | | |
|---|---|---|---|
| `f(1111) = 30` | `f(0100) = 27` | `f(0110) = 5` | `f(1110) = 0` |
| `f(0000) = 10` | `f(1000) = 28` | `f(1001) = 5` | `f(1101) = 0` |
| `f(0001) = 25` | `f(0011) = 5` | `f(1010) = 5` | `f(1011) = 0` |
| `f(0010) = 26` | `f(0101) = 5` | `f(1100) = 5` | `f(0111) = 0` |

- Notes
    - The deceptive attractor, $s_d$ = 0000, defines a basin which is fully deceptive, but the fitness is only 1/3 of the global optimum.
    - The deceptive attractor is weaker than its neighbors and therefore cannot be a local optimum in Hamming space.
    - Such a deceptive attractor needs to be surrounded by strong neighbors however in order to hide the weak fitness of the attractor.

- Theorem 3 [1]:
    - "A deceptive attractor of order-*n* for a binary encoded problem cannot maintain full deception at order-*n* if it is weaker than any string or schema which differs from the deceptive attractor by exactly two bits."
- That is, the single deceptive attractor is replaced by more than one attractor, which will then break the control of the attractive basin and potentially reduce the effectiveness of the deception (with respect to the optimal hyperplane).

## Remapping Strategies

- Remapping is the process by which the representation scheme is completely (e.g. binary to gray coding) or partially changed.
- Only problem here is that *a priori* knowledge is necessary to identify whether the problem requires remapping.
- Knowledge of the exact location of any deceptive attractor is as difficult to ascertain as the global optima.

## Deception and Linkage

- Deception is much more difficult to detect when the degree of linkage between deceptive building blocks is weak.
    - Weak linkage → deceptive bits or building blocks are widely distributed across the length of the bit string.
    - Schema theorem already indicates that those schemas, which are concise and have above average fitness, will be reproduced with exponential rates of reproduction.
        - By distributing the deceptive building blocks, multiple instances of different deceptive schema will see reproduction.

## Deceptive test problem generator

- 30 bit function
    - Composed from 10 copies of a fully deceptive order 3 bit (sub)function.
    - Each subfunction is uniformally and maximally distributed across the length,
        - Say the subfunction has a bit at position $i$, $i + 10$ and $i + 20$.
- 40 bit function

    - Composed from 10 copies of a fully deceptive order 4 bit (sub)function.
    - Each subfunction is uniformally and maximally distributed across the length,
        - Say the subfunction has a bit at position *i*, *i* + 10, *i* + 20 and *i* + 30.
- Crossover,
    - 4 cases considered (tested individually)
        - 1-point crossover
        - Uniform crossover
        - Bit tagging
            - Separate tag bit used to denote order – crossover employs ONE parent to define the ordering.
            - 1-point crossover is then applied
            - provides for the basis for the evolution of bit order.
        - Distributed GA
            - Multi-population model,
                - Each population evolves independently;
                - Migration between populations permitted.
- Experimental Results

| Problem | Crossover | Pop. Size | Solved | Evaluations |
|---|---|---|---|---|
| Order-3 | 1-point | 200 | 27% | 10,000 |
| | Uniform | | 27% | |
| | Tagged | | 53% | |
| | 1-point | 2,000 | 38% | 50,000 |
| | Uniform | | 35% | |
| | Tagged | | 64% | |
| | Parallel | | 55% | |
| Order-4 | 1-point | 200 | 7% | 10,000 |
| | Uniform | | 3% | |
| | Tagged | | 16% | |

## Conclusions

- Tagged bits appear to provide a useful method for addressing the linkage problem
    - Let the GA evolve the relevant bit sequencing;
    - Other researchers have questioned its usefulness as the size of the search space is significantly increased.
- "Messy GAs" not evaluated, and might also provide a more robust scheme for dealing with deceptive problems;
- Multi-population results may also carry over to niche based methods (e.g. crowding) in which the population is able to follow multiple optima concurrently.

## Reference:


- Whitley L.D., "Fundamental Principles of Deception in Genetic Search," Foundations of Genetic Algorithms, Volume 1, Rowlins G.J.E. (ed). Morgan Kaufmann, ISBN 1-55860-170-8. 1991, pp 221-241.


## Additional Reading


- Grefenstette J.J., "Deception Considered Harmful," Foundations of Genetic Algorithms, Volume 2, Whitley L.D. (ed). Morgan Kaufmann, ISBN 1-55860-263-1. 1993, pp 75-91.
- Forest S., Mitchell M., "Relative Building-Block Fitness and the Building Block Hypothesis," Foundations of Genetic Algorithms, Volume 3, Whitley L.D. and Vose M.D. (ed). Morgan Kaufmann, ISBN 1-55860-356-5. 1995, pp 109-126.
- Horn J., Goldberg D.E., "Genetic Algorithm Difficulty and the Modality of Fitness Landscapes," Foundations of Genetic Algorithms, Volume 3, Whitley L.D. and Vose M.D. (ed). Morgan Kaufmann, ISBN 1-55860-356-5. 1995, pp 243-269.