

Implicit Formae in Genetic Algorithms^{*}

Márk Jelasity¹ and József Dombi²

¹ Student of József Attila University, Szeged, Hungary
jelasity@inf.u-szeged.hu

² Department of Applied Informatics, József Attila University, Szeged, Hungary
dombi@inf.u-szeged.hu

Abstract. This paper discusses the new term *implicit forma*, which is useful for explaining the behaviour of genetic algorithms. Implicit formae are special predicates over the chromosome space that are not strongly connected to (though not independent of) the representation at hand. The new term is a generalization of the concept of formae such that every approach connected to formae (e.g. fitness distribution) is also relevant to implicit formae. After a short theoretical discussion, three examples are given for illustration, including the subset sum problem which is NP-complete.

1 Introduction

An understanding of how genetic algorithms (GAs) work is of major importance from the point of view of both theory and application. For a long time, the concept of schemata played the central role in GA theory [1]. However, it is now clear that this concept is itself not enough for a prediction of the behaviour of the GA [12]; at least some generalization of the concept is necessary for both binary and general representations.

For general representations, the concept of formae (e.g. [5]) has been introduced. This approach is especially useful in the design of genetic operators, but it should be mentioned that formae are very similar to schemata in the sense that they are strongly connected to the representation at hand (though the representation is normally designed using the previously chosen formae).

This paper shows that besides the carefully designed formae there are other factors capable of directing or even misleading the search. These factors seem to be treated as properties of formae in the literature, e.g. the variance of fitness [8, 10] or noise [11]. While understanding that these are useful tools for gaining information about the quality of a given representation of the problem at hand, we try to provide a deeper insight into the search process by introducing the term *implicit forma*. We believe that this approach will help in predicting and especially in explaining the behaviour of GAs in several problem classes, including real-world applications.

In section 2, we give a brief introduction to forma analysis, restricting ourselves only to the basic definitions, and implicit formae are then discussed. In section 3, the new term is illustrated through three case studies. One of the examples is the subset sum problem, an NP-complete combinatorial problem. The relation of the GA and the bit-hillclimber algorithm is also discussed on the basis of implicit formae.

^{*} The original publication is available at www.springerlink.com. In *The Proceedings of PPSN'96*, Springer, LNCS 1141, pp154–163, DOI:10.1007/3-540-61723-X_979

2 Formae and Implicit Formae

2.1 Formae

A discussion of formae is needed only to make it clear why the name *implicit forma* has been used to denote the properties discussed in the paper. Thus, a very basic knowledge suffices. A detailed description can be found in [5].

A representation maps the solution space S to a chromosome space C . Usually, every $x \in C$ can be regarded as an intersection of a set of predicates over C . If $C = \{0, 1\}^n$, then these predicates are the schemata of order 1. If C is the chromosome space of the permutation representation of the traveling salesman problem, then these predicates are the subsets of the set of all permutations with a fixed town at a given position.

Thus, a set of *alleles* (i.e. predicates that a chromosome may contain) can be assigned to every representation. A *forma* is simply the intersection of a subset of the alleles. The empty set is not a forma.

It is clear that if $C = \{0, 1\}^n$, then formae reduce to schemata, so a forma is a generalization of the concept of a schema.

From our point of view, the essence of the above definitions is that formae are predicates over the chromosome space C that are closely related to the representation at hand.

2.2 Implicit Formae

It has already been shown [3, 4] that every predicate over the space of all chromosomes C behaves according to the schema theorem for some appropriate genetic operators; in other words, its proportion is approximately determined by its observed fitness. The forma analysis is connected to this result, i.e. representation-independent operators are designed [10] to be "friendly" with the formae given by the representation.

However, it is possible that some other predicates over the chromosome space C are also treated as the formae, i.e. they obey the schema theorem and so are capable of directing the search. Examples of this phenomenon will be given in section 3. The existence of such predicates motivates our central definitions.

First, let us fix the main GA components; the chromosome space C , a set of genetic operators (selection, crossover, mutation) and their parameters designed not to be too disruptive (this assumption will be clarified later). We will refer to these components later as *fixed GA components*. The following definitions are independent of the objective function.

Notation 1. Let $\mathcal{P}(C)$ be the set of all predicates over C .

Definition 1 has a central role.

Definition 1. The degree of relevance of a given predicate $P \in \mathcal{P}(C)$ with respect to the fixed GA components is $r(= r(P))$ iff during the successive iterations of the GA (given by the fixed GA components and P as the objective function), starting from an infinite uniformly distributed random population, the proportion of P goes to r as the number of generations goes to infinity, where

$$P(x) = \begin{cases} 1 & \text{if } x \in P \\ 0 & \text{otherwise} \end{cases}$$

Definition 2. A given predicate $P \in \mathcal{P}(C)$ is relevant iff $r(P) > p_0$ and is neutral iff $r(P) = p_0$, where $p_0 = |P|/|C|$.

The rate of relevance is also important and would be worth discussing in more detail, but for our present purposes Definition 2 suffices.

Though this paper focuses on the experimental results, for illustration we give an analysis of predicate EVEN without the straightforward technical details.

Definition 3. $C = \{0, 1\}^n$, $EVEN \in \mathcal{P}(C)$; $EVEN(x)$ iff the number of 1s in x is even.

Through this simple example, we would like to emphasize an advantage of Definition 1: due to the very simple objective function, an *exact* dynamic analysis of the relevance level can be given even for realistic problems and predicates.

Theorem 1. Let the GA components be $C = \{0, 1\}^n$, 1-point crossover with a probability P_c , generational and proportional selection without the transformation of the objective function (i.e. the fitness function equals the objective function) and no mutation. Then, for a large enough n ,

$$r(EVEN) \approx (1 - P_c) + \frac{1}{2}P_c \quad (1)$$

Proof. If n is large enough, then for any $x \in C$ the probability that a randomly chosen half of x contains an even number of 1s is $1/2$.

Let p_t be the proportion of EVEN in the t th generation, and let $g(p_t)$ be the expected proportion in generation $t+1$ without the effect of the genetic operators. Here, $g(p_t) = 1/2$.

The disruption of the genetic operators under the above assumptions is

$$\text{dr}(g(p_t)) = g(p_t)(1 - P_c) + \frac{1}{2}P_c$$

It is trivial that $\text{dr} \circ g$ has a unique fixpoint x_0 in $(0, 1]$ which is given by the equation $\text{dr}(g(x_0)) = x_0$ and equals (1).

Theorem 2. Let the GA components be the same as in Theorem 1 except that the fitness function is the objective function incremented by 1. Then, for a large enough n ,

$$r(EVEN) \approx \frac{1}{2} \left(1 - \frac{3}{2}P_c + \sqrt{\left(\frac{3}{2}P_c\right)^2 - P_c + 1} \right) \quad (2)$$

Proof. The same as the proof of Theorem 1, except that $g(p_t) = 2p_t/(p_t + 1)$.

A trivial corollary immediately follows from Theorems 1 and 2.

Corollary 1. Under the assumptions of Theorem 1 or 2, if $P_c = 1$, then EVEN is neutral, and if $P_c \neq 1$, then EVEN is relevant with the relevance level given by (1) and (2), respectively.

Now we can clarify the assumption that the operators in the fixed GA components are designed not to be too disruptive. This simply means that we want the members of F to have a high degree of relevance (preferably around 1). In our terms, this is what applying representation-independent genetic operators provides [10].

The definition of the degree of relevance refers to infinite populations. In practice, the intuitive relevance also depends on the size of the predicate since very small predicates might not get a sample at all. Moreover, a relevant but *practically* irrelevant predicate (e.g. due to its small size) may also become *practically* relevant because of the effects of the changes in the sampling distribution (this can be thought of as a generalization of the building-block hypothesis [2]).

Finally, the implicit formae can be defined:

Notation 2. $F \in \mathcal{P}(C)$ is the set of all formae given by the fixed GA components. The formae will also be called *explicit formae* if emphasis of the difference from the implicit formae is necessary.

Definition 4. The predicate $P \in \mathcal{P}(C)$ is an implicit forma iff it is relevant and is not an explicit forma (i.e. $P \notin F$).

It should be emphasized that a relevant predicate is not necessarily a useful predicate, in the sense that it is not necessarily capable of directing the search. Its usefulness depends on the particular objective function f , e.g. the variance of f in it and the properties of its "building blocks". Recall that the above definitions are independent of f .

3 Implicit Formae at Work

In this section, three case studies will be presented.

The first illustrates how (rather exotic) implicit formae can direct the search process. The second is the subset sum problem, where we analyze the GA from the basis of implicit formae. The third offers a possible way of creating problems in which the GA performs better than a simple hillclimber algorithm, again using implicit formae. Such problems have received much attention recently [7].

The following GA components are the same in all three examples: $C = \{0, 1\}^{100}$, 1-point crossover, mutation with $P_m = 0.003$ and a population size of 100. $P_c = 0.6$ in the last example, otherwise $P_c = 1$. The selection used was elitist and proportional. To perform the experiments, GENESIS was used modified so that it could trace our non-traditional implicit formae. The algorithms were run until 10^4 function evaluations in every experiment in each case. All functions were maximized.

3.1 An Example for Illustration: the Equal Blocks Problem

The objective function f of this example was designed especially to illustrate the idea of implicit formae. However, it should be noted that it may very well happen that real problems have features like this one. Its domain is C and for an $x \in C$ $f(x)$ is counted as follows:

Let us fix an ideal block size $b = 5$. Let us divide x into blocks that contain only 1s or 0s (e.g. 111|0|11|000). For every block containing b' elements let us subtract a penalty $|b - b'|$ from the objective function value and let us fix the optimum value at 0. It is clear that the optimal individual will contain 20 blocks with 5 elements in each. For illustration, we give the two optimal solutions of the 30-bit equal blocks problem:

000001111100000111110000011111, 111110000011111000001111100000

This task meets our needs because formae (i.e. schemata) have little meaning and high fitness variance. It may be thought that schemata like

*...*0111110*...*

have high fitness. However, their fitness variance is considerable because the function is extremely epistatic and is quite insensitive to shifting due to its inherent properties.

Twenty independent experiments were performed with the GA and also with the uniform random search. The averages of the solutions found were -23.3 and -212.2 , respectively.

An explanation of this result can be given on the basis of the existence of implicit formae. Let us define a predicate over C .

Definition 5. $[y, z]$ -blocknumber $\in \mathcal{P}(C)$, $x \in C$ $[y, z]$ -blocknumber(x) iff the number of blocks contained in x is in $[y, z]$.

Figures 1b and 1a support the following hypotheses:

- $[20, 30]$ -blocknumber is an *implicit forma*. As shown in Fig. 1a, $[20, 30]$ -blocknumber gained a proportion of almost 100% so it must be relevant and is clearly not an explicit forma. It is also interesting to note that (as shown in Fig. 1b) the expected and observed growth fits well. This also indicates that $[20, 30]$ -blocknumber is relevant.
- $[20, 30]$ -blocknumber has an *important role in the search process*. The typical S-curve in Fig. 1a is familiar from the analysis of the above average and low order schemata of low fitness variance.

To summarize the first example, it have been shown that an implicit forma the existence of which is not trivial from the representation played an important role in the search.

3.2 A Real Example: the Subset Sum Problem

We study the subset sum problem here. We are given a set $W = \{w_1, w_2, \dots, w_n\}$ of n integers and a large integer M . We would like to find an $S \subseteq W$ such that the sum of the elements in S is closest to, without exceeding, M . This problem is NP-complete.

We used the same coding and objective function as suggested in [6]: If $x \in C$ ($x = (x_1, x_2, \dots, x_{100})$), then let $P(x) = \sum_{i=1}^{100} x_i w_i$, and then

$$-f(x) = a(M - P(x)) + (1 - a)P(x)$$

where $a = 1$ when x is feasible (i.e. $M - P(x) \geq 0$) and $a = 0$ otherwise.

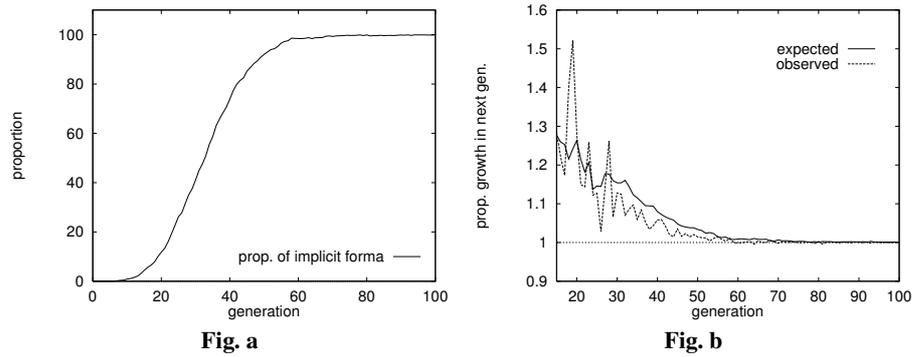


Fig. 1. (a) the proportion growth and (b) the expected and observed growth of the implicit forma [20, 30]-blocknumber. Average of 20 independent runs.

When creating a problem instance, elements of W were drawn randomly from the interval $[0, 10^4]$ instead of $[0, 10^3]$ (as was done in [6]) to obtain larger variance and thus a harder problem. The sum of all of the elements in W was 455784 and the sum to be created was 10^5 . (It should be noted that optimal solutions do exist for the examined problem instance.)

After studying several experiments with the GA, a hypothesis seemed reasonable. The GA tends to sample individuals in which the number of 1s is close to $100 \cdot 10^5 / 455784 \approx 22$. That means that the numbers in W are treated as probability variables for which the expected value of the sum of any subset with 22 elements is 10^5 .

In other words, it is assumed by the hypothesis that the GA "figures out" how the problem instance was generated.

After forming the above hypothesis, four algorithms were run 50 times independently:

GA As described earlier.

HYPO A direct implementation of the hypothesis. Every bit is set to 1 with a probability of $22/100 = 0.22$ independently.

RAND Uniform random search.

HILLCLIMB Starting from a random solution, a randomly chosen bit is inverted and the new solution replaces the old one if it is not worse. This process is iterated.

The averages of the solutions were -4.36 , -4.65 , -27177 and -302.4 , respectively. GA found 12, while HYPO found 6 optimal solutions during the 50 runs. The results clearly reveal that, the hypothesis is reasonable. However, the average number of bits in the 50 solutions of the GA is 28.9, which is slightly more than predicted. Figure 2 sheds some light on this issue. The higher peaks tend to belong to relatively small values from W , while the lower proportions indicate a relatively large value. This is because individuals containing large values tend to die off at the very beginning of the search.

It is now time to explain exactly what the hypothesis means. Clearly, it says nothing about any particular element or subset of W . The only important feature is the number of 1s in an individual, according to the hypothesis.

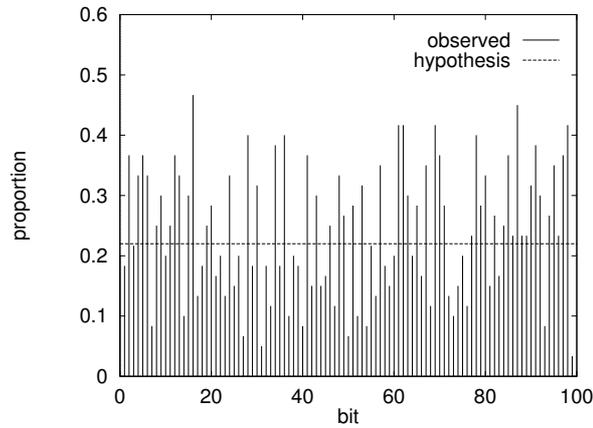


Fig. 2. Proportion of value 1 for a given bit over the solutions of the 50 independent runs of the GA. The proportion indicated by the hypothesis is also shown.

To express this in our terminology, there are implicit formae, based on the number of 1s in a chromosome, that play a mayor role in the optimization process. This motivates the following definition.

Definition 6. $[y, z]$ -1s $\in \mathcal{P}(C)$, $x \in C$. $[y, z]$ -1s(x) iff the number of 1s in x is in $[y, z]$

$[24, 34]$ -1s was traced by GENESIS and the statistics are shown in Figs 3b and 3a.

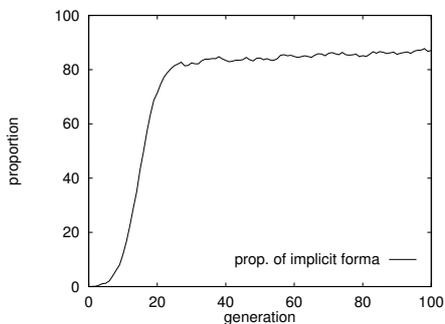


Fig. a

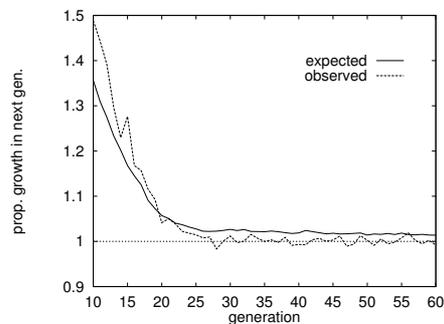


Fig. b

Fig. 3. (a) the proportion growth and (b) the expected and observed growth of the implicit forma $[24, 34]$ -1s. Average of 20 independent runs.

The graphs are very similar to those of the previous example, the equal blocks problem, so the conclusions are also very similar; in the case of the subset sum problem (with

the GA components and the problem instance generation method used here), implicit formae play an important role.

3.3 When Will a GA Outperform Hillclimbing?

The title of this section is borrowed from [7]. Here, using implicit formae, we will try to point out some basic differences between the GA and hillclimbing through a simple example. We believe that this approach can be generalized, however. Moreover, using the definitions given in [10], more general representations could also be considered.

It is well known that functions that are easy for the GA (e.g. royal road functions) are easy (if not easier, see [7, 9]) for the bit-hillclimber, i.e. the algorithm HILLCLIMB in section 3.2. This is because HILLCLIMB can easily combine *explicit* formae (here schemata) in the case of such problems.

But what about *implicit* formae? As we have seen, the GA can handle several implicit formae besides the explicit ones, and these implicit formae are not necessarily "relevant" for HILLCLIMB. The example of this section illustrates this effect. Let us consider the function

$$f(x) = \begin{cases} \|x\| & \text{if } \|x\| \text{ is even} \\ -\|x\| & \text{otherwise} \end{cases}$$

where $\|x\|$ is the number of 1s in x .

This function is extremely hard for HILLCLIMB since every x with even $\|x\|$ is a local optimum from which HILLCLIMB cannot get out. On the other hand (as shown in section 2.2), EVEN is an implicit forma if $P_c < 1$ and $P_m = 0$. On the basis of this observation, P_c was set to 0.6.

20 independent experiments were performed with RAND, HILLCLIMB and the GA. The average best results were 67.7, 49.2 and 83.3, respectively. Observe that HILLCLIMB is considerably worse than RAND. Figure 4a indicates that EVEN is relevant with a relevance level of approximately 0.66.

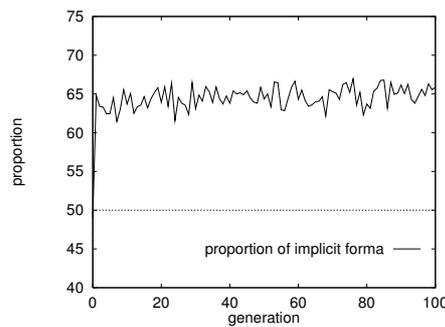


Fig. a

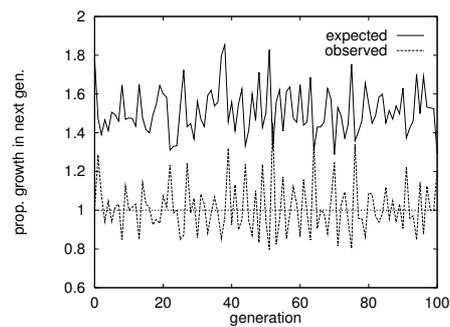


Fig. b

Fig. 4. (a) the proportion growth and (b) the expected and observed growth of the implicit forma EVEN. Average of 20 independent runs and a typical single run, respectively.

As shown in Fig. 4b, in spite of the constantly strong pressure, EVEN cannot go further than 66% after a quick increase at the very beginning of the search. However, the relevance level of 0.66 is enough to outperform both RAND and HILLCLIMB.

Let us make a final remark. It may be thought that EVEN is a very artificial property which will not be encountered in the case of a real problem. However, for instance, it is a well-known fact in chemistry that atoms that have an *even* number of nucleons in their nuclei are always more stable than those with an odd number of nucleons.

4 Happy or Sad Conclusions?

In this paper, we have examined the implicit formae, the invisible forces that can direct the genetic search as strongly and definitely as explicit formae. The only problem is that in the case of a particular problem we know only the explicit formae and this can make analysis of the behaviour of the GA quite difficult.

One solution could be to determine (enumerate) all of the implicit formae for a given task and examine all of them with the tools of the GA analysis. This may be a lot of work since it is not trivial at all what the implicit formae of a given representation are, even if it is simple. In spite of this, for commonly used domains it may worth doing this analysis. However, for real applications, the representation (and thus the chromosome space C) and the operators tend to be different, difficult and problem-specific so the situation is not too hopeful.

On the other hand, this property of GAs of handling previously unknown implicit formae is very useful because it makes possible for the GA to be independent from the representation in the sense, that there is a probability of performing an effective search when only few information is available when designing the representation and so the schemata (or formae) doesn't seem to be too informative.

References

1. J.H. Holland (1975) *Adaptation in Natural and Artificial Systems*. University of Michigan Press (Ann Arbor).
2. D. E. Goldberg (1989), *Genetic algorithms in search, optimization and machine learning*, Addison-Wesley, ISBN 0-201-15767-5.
3. N.J. Radcliff (1991) Equivalence Class Analysis of Genetic Algorithms. *Complex Systems*, 5(2):183-205
4. M.D. Vose (1991) Generalizing the Notion of Schemata in Genetic Algorithms. *Artificial Intelligence*
5. N.J. Radcliff (1992) Non-linear Genetic Representations. In R. Männer and B. Manderick editors, *Parallel Problem Solving from Nature 2*. pp259-268, Elsevier Science Publishers/North Holland (Amsterdam)
6. S. Khuri, T. Bäck, J. Heitkötter (1993) An Evolutionary Approach to Combinatorial Optimization Problems, in *The Proceedings of CSC'94*.
7. M. Mitchell, J.H. Holland (1993) When will a Genetic Algorithm Outperform Hillclimbing? (SFI working paper)
8. N.J. Radcliff, F.A.W. George (1993) A Study in Set Recombination. In *The Proceedings of ICGA'93*.

9. A. Juels, M. Wattenberg (1994) Stochastic Hillclimbing as a Baseline Method for Evaluating Genetic Algorithms. Technical Report, UC Berkeley
10. N.J. Radcliffe, P.D. Surry (1994) Fitness Variance of Formae and Performance Prediction. In L.D. Whitley and M.D. Vose editors, *Foundations of Genetic Algorithms III*, Morgan Kaufmann (San Mateo, CA) pp51-72
11. H. Kargupta (1995) Signal-to-noise, Crostalk and Long Range Difficulty in Genetic Algorithms. In *The Proceedings of ICGA'95*.
12. M.S White, S.J. Flockton (1995) Modeling the Behaviour of the Genetic Algorithm. In *The proceedings of GALEZIA'95*