Evolutionary Approaches for Real World Applications in 21st Century

Nikhil Padhye
Massachusetts Institute of Technology
Cambridge, MA 02139, USA
npdhye@mit.edu

ABSTRACT

Evolutionary computation is notably one of the fast growing fields of research and application and is becoming pervasive in several streams of science and engineering. This paper reviews its origin, investigates the reason for its growth and widespread applicability. The reasons for existence and advantages of other paradigms are also discussed. The goal of this paper is to underline root-causes necessary for successful deployment of evolutionary methods in diverse applications and challenges that need to be addressed in any such endeavour.

Categories and Subject Descriptors

I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods and Search

General Terms

Theory

Keywords

Evolutionary Computation, Genetic Algorithms, Real World Applications

1. INTRODUCTION

Modern practices in engineering and science are greatly benefiting from computational methods. Evolutionary computation, a family of computational methods which attempt to mimic the principles of nature, are finding major application in search, optimization and machine learning. More particularly, evolutionary methods are serving great usefulness in a variety of optimization tasks for e.g. nonlinear, non-smooth, constrained, multiple objectives, combinatorics, etc. Simplicity in implementation, availability of standard source codes and variety of options for modifications and adaptability in context to different scenarios can be attributed as major reasons for this success.

GECCO'12 Companion, July 7–11, 2012, Philadelphia, PA, USA.

Succinctly, the origins of evolutionary methods can be traced back to times when theory of adaptive systems was being developed. The original conception adopted a natural perspective from a biological example [6]. These initial proposals led to Genetic based machine learning (GBML) for e.g. [8, 9], and later the development of Genetic Algorithms [12]. The same era marked the development of Evolutionary Operations (Box, 1957), Simplex Method (Nelder and Mead, 1965), Evolutionary Strategies (Rechenberg 1964 and Schwefel 1965), Evolutionary Programming (Fogels, Owens and Walsh 1966). The consensus confluence of these developments led to the modern research community of Evolutionary Computation.

After the proposal of using Genetic Algorithms for function optimization, the task remained in hands of civil engineer David Goldberg who took the challenge and demonstrated the application of Genetic Algorithms for real world problems through his own dissertation [5], later followed by his seminal work in field of genetic algorithms [2] and an entire generation of diverse developments. Dejong's book [13] is another excellent modern resource for overview on rise of different evolutionary algorithms and adopting a unified outlook on these methods [13]. For a short overview on major developments over three decades in Genetic Algorithm the reader is referred to [15].

A complete recollection of salient contributors in evolutionary algorithms is neither possible nor the intended goal of this paper, rather we are tempted to understand its fast-paced growth and currently encountered challenges to seek potential research directions. Throughout this paper, while referring to GA the connotation to evolutionary algorithms is self-implied and vice-versa.

Although, current era is arguably favorable for growth and application of evolutionary methods, yet, like any other field of research evolutionary methods several face several challenges in advancements to next levels. Thus, the motivation of this paper is to discuss:

- The research opportunities for practitioners, both, in evolutionary and outside community.
- Challenges encountered in the application of evolutionary methods to real-world applications and related research questions to be answered.

The emergence of several evolutionary algorithms and departure from canonical evolutionary forms provokes a deeper and thorough understanding of this field. In a recent study [17], authors investigated how several forms of evolutionary algorithms can be studied in a unified frame-work and emphasized towards taking a holistic approach to several EC methods.

This paper draws its inspiration from similar study and centers its discussion around GAs, indisputably one of the prime evolutionary methods, but the ideas are equally extendable to other EC or related methods. This study emphasizes on answering "How can GAs be successfully applied in context to engineering or other real world problems?" Genetic Based Machine learning methods have also gained wide popularity but have not been treated as central interest of this study.

The rest of the paper is structured as follows: Section 2 presents a perspective on how computational tools are largely influencing science and engineering and discusses the rise of evolutionary computation in same light while revisiting the biological metaphor. Section 3 discusses the structure and strength of population based search like Genetic Algorithm. Section 4 discusses key issues and challenges associated in applying GA for any real world application problem. A comparison with other optimization techniques is also discussed. Section 5 discusses popular areas of GA application in context to engineering real world systems and also the potential areas of application. Finally, a summary of challenges and way forward is discussed in Section 6.

2. COMPUTATIONAL COMPETENCY AND SIMPLE GA UNDERSTANDING

The birth and advent of programmable machines i.e. computers has drastically influenced the practice of science and engineering. Put in simple terms the modern computational tools provide an altogether novel and efficient means for carrying out arithmetic operations which would not be possible other wise.

Engineering simulation tools, such as Computational Fluid Dynamics, Numerical Heat Transfer, Finite Element Methods, Molecular Dynamics, etc., are based on underlying scientific laws and prove effective in engineering analysis. The physical laws of fluid-motion, heat-transfer, material deformation or atomistic interaction are programmed or abstractedly represented one way or the other into these packages and calculations for desired scenario are performed. However, the first step in development of any of these computational methods was identifying the underlying science and theory.

The overall exercise is to embed the observed physical, natural, engineering phenomenons into computational framework. If the physical/natural principles have been accurately modeled and represented; the programs perform as expected. Better the operation principle are represented the better is the performance. These software packages are based upon numerical calculation (which sometimes is a departure from closed form or analytical solutions) yet they are extremely accurate in terms of making prediction.

In the light of above discussions, we view evolutionary methods, either for optimization, search or machine learning tasks: as principles of nature encoded in form of computer programs employed for solving certain problems. Since there are no equations to describe the biological processes itself; the evolutionary methods are simply represented as heuristics in form of a computer program. And like any other computational tool, evolutionary algorithms (which are encodings of principles of nature) have gained large popularity and success in a variety of contexts.

Decades after initial conception and emergence of several evolutionary methods, the earliest notions of expressed surprise [7]:

> "Even founders of GA did not know why it works: Computer programs that "evolve" in ways that resemble natural selection can solve complex problems even their creators do not fully understand."

are now less starking, as now evolutionary algorithms are taken for granted as principles of nature borrowed for developing computational tools. The component of competition and survival of the fittest is a logical guiding principle for creation of newer and better solutions. This also explains why different evolutionary algorithms exist. Nevertheless, applying philosophical ideas from evolution to problem solving has been a major leap.

3. STRUCTURE AND STRENGTH OF A GA

Figure 1 shows the key steps involved in a simple Genetic Algorithm. The algorithm begins with *Initialization* of the population and then proceeds by repetition of *Selection*, *Crossover*, *Mutation* and *Replacement* steps until a termination criterion is met.

The *Selection* step represents the survival of the fittest in GA. The role of this operator is to preserve better individuals. *Crossover* and *Mutation* are responsible for creation of new individuals. The *Replacement* plan dictates which of the newly created members should be preserved in the subsequent population. As GA proceeds through several generations the population members are expected to continually improve.

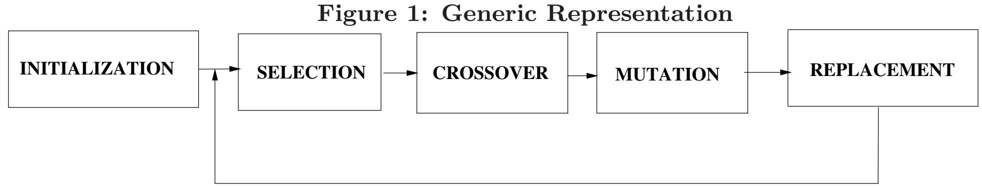
Nearly two decades ago when evolutionary algorithms were still in stage of research and development, a remarkable study [3] investigated and emphasized "Why Use GAs in applications?", and following reasons were cited:

- GAs solve hard problems quickly and reliably.
- GAs are easy to interface to existing simulations and models.
- GAs are extensible.
- GAs are easy to hybridize.

The anatomy of GA is unique and makes the search efficacious. The features of working with a population (i.e. carrying out a parallel search), choosing a suitable type of encoding scheme (real, binary, etc.), requirement of only function values (i.e. no need for calculating gradients), non-necessity of continuous search space, and probabilistic transition rules, make a GA algorithm attractive. These intrinsic characteristics empower the flexibility of modifying and devising customized GAs for a wide variety of solving problems. The initial success from solving simple optimisation problems to real-world applications has furthered the development of evolutionary methods.

The aspect of flexibility in terms of defining almost every component of a GA architecture (discussed more in the next

Figure 1: Generic Representation



section) and the ability to evolve a population based on desired rules has played a key role in tackling challenges like multi-modal, multi-objective, constrained, etc. optimisation problems.

For example, consider niche-formation methods in GAs for maintaining diversity. The analogy from nature is that within the environment there are different subspaces (niches) that can support different types of life (species, or organisms). Each niche survives on its own and co-exists with other niches. This simile has been successfully implemented to maintain diversity [11] in GA based searches. The take-home message is that nature's principle can indeed be employed problem solving. However, simply encoding such principles in an artificial environment does not guarantee success either, and a thorough investigation is often needed. Thus, freedom of introducing simple, elegant and/or intricate mechanisms to guide the search has, both, pros and cons. The key lies in developing an appropriate evolutionary logic to perform an effective search.

As a last word, the evolutionary methods are characterized as heuristics based on the philosophy of biological evolution (which does not have a mathematical representation). Thus, its unlikely to accurately predict their behavior (or in other words there may not be a generic proof-for-convergence).

4. ENGINEERING CHALLENGES AND EVOLUTIONARY OPTIMISATION

For features discussed in the previous section evolutionary algorithms promise to be a robust search procedure while scanning different regions of the search space in parallel. Their success has been well recorded on several test and real problems. However, for any new real world application (where optimum may not be known) the success of GA may be questionable.

The real world problems exhibit high degree of non-linearities in fitness evaluations, discontinuous search-spaces, non-linear constraints, other ill-conditionalities, etc. Often under these circumstances there is no or little a priori information about the problem at hand and following challenges are encountered:

- Genotype encoding i.e. representation of the search space in terms of GA variables (binary, real parameter, or both, etc.)
- Description of *Selection*, *Crossover*, *Mutation* operators depending upon the search space and problem features.
- Population sizing, selection pressure, crossover and mutation rates; to ensure a balance between exploration and exploitation.
- Constraint handling i.e. how are variable bounds, inequality and equality type constraints satisfied.
- Sensitivity analysis i.e. how the performance of GA depends upon parameter selection and regimes of robust operation.
- Estimation of fitness function either through prescribed objective functions, simulation outcome, etc.
- Real run-time for completion of search and/or optimisation.
- Robust performance i.e. effective and efficient search to avoid pre-mature convergence.
- Resource requirements for computation (for e.g. memory needs).

In order to arrive at an optimal solution an effective GA architecture is needed: Certain representations of the search spaces may turn to be more convenient and effective than the others. The iterative steps of *Selection*, *Crossover*, *Mutation* are extremely critical: What is a good selection scheme to prefer better individuals over the worse ones? What is an effective mechanism to create new solutions for a chosen problem representation via. crossover and mutation operations? What is a good replacement plan? In addition to the algorithmic description of these steps, an appropriate choice of GA parameters must be made for fast and accurate search.

The key question is: Whether *Selection*, *Crossover*, *Mutation* and *Replacement* are working in isolation? and if there is any synergistic effect while performing search?

During the development of canonical GAs (binary representation) special emphasis was placed on notion and importance of building blocks. The central being that good bits come together through genetic operations and population improves as the search advances. Even though modern evolutionary methods drastically differ from original proposals but the core working principle still stays the same.

It is worth stating that several choices for representing problem search space and selecting associated operations gives a high degree of flexibility to the user and in some cases allows to exploit the underlying problem structure, if known a priori. For e.g. see [17] for modification and development of an effective evolutionary algorithm to solve unimodal problems.

In particular, the real world problems pose several constraints and therefore finding an optimal yet feasible solution is a challenge. A reasonable strategy is to thoroughly investigate rich GA literature and seek hints for solving the problem at hand.

Another practical issue is the complexity in estimating the fitness value (particularly if it is an outcome of some time-consuming simulation). For e.g., in many engineering world problems the value of objective function is an outcome of simulations like CFD, FEA, MD, etc. where the computation time of one evaluation on a standard computing platform could range from a few minutes to several hours.

In a practical study performed on Optimization of Advanced Well Type and Performance; computational cost was cited as a major problem [1]. In such scenarios, the parallelizable structure of a GA can be exploited in and time consuming fitness evaluations can be done concurrently.

Another extremely important question is: "How much longer before the GA can be terminated?" Upfront there is no simple answer. There have been past attempts to model and devise theoretical models to predict the performance of canonical GAs [14]; however they are based on simplified assumptions and not accurate enough to make predictions for real situations. This necessarily is not a dead end.

Since, in engineering contexts the goal is often to find an approximate or close to optimal solution instead demanding an exact optima. Several population based metrics (diversity of the population, average fitness and standard deviation of the population, etc.) can be studied to understand the evolution of population. A thorough empirical study and experimentation can be performed to identify choice of GA parameters for ensuring robust and repeatably good performance. An empirical analysis shedding light on the working genetic operators, independently and synergistically, will be extremely useful in ensuring similar or bettering the performance of GA in the after-wards similar or related contexts.

Furthermore, requirements of many engineering applications are often met if an efficient, feasible and practically acceptable solution is found; rather than locating an exact optima. GA based procedures can be extremely useful in finding an improved solution which can be physically tested and shown to work.

For example, the study done in [16] evolved X-band antenna design and flight prototype. The antenna was evolved to meet a challenging set of mission requirements, most notably the combination of wide beamwidth for a circularly-polarized wave and wide bandwidth. The highest performance antenna found using a genetic algorithm was actually fabricated and tested. Thereby, if an evolutionary approach can help find an engineering solution that meets the specifications and functional requirements, GAs can be declared successful.

For GA methods to gain wider applicability and acceptability the final search based solutions should be fabricated, tested, implemented, etc. in form of physical hardware of some sub-system. In other words, the success will hugely rely upon embedding and representing physical functional requirements into an artificial system and finding solutions to it. Moreover, once a certain degree of success is demonstrated for a real-world application, it is likely that GAs performance can be improved through further trials and modifications on same or similar applications. The advantage of applying GAs on real-world systems rises from the fact that often a good engineering-solutions can be validated through physical arguments and performance tests.

Such research studies should also enable traditional optimisation communities who express inhibitions in adopting evolutionary methods due to their lack of convergence-proof and reliable performance.

The earlier arguments favored GAs over traditional optimisation techniques (based on no gradient requirements, etc. Section 3) However, limitations faced by classical optimisation methods (requirements of continuous search spaces, existence of gradients, multi-modality, etc.) have recently been addressed by deterministic Global Optimisation methods. These include: Dynamic Programming, Branch and Bound method, Inner approximation, Outer approximation, Cutting method (normal boundary intersection), Response surface methods, etc. Some of these methods work with linearized approximation of objective function/constraints and can guarantee ϵ-convergence. This is also a fast growing area of research in terms of theoretical developments and practical applications.

Conventional GAs success relies on the growth of building blocks that are essential for creation of fitter individuals as the search proceeds. Thus, for GA hard problems which have needle in a hay-stack type description, deterministic or exhaustive optimisation methods might be the only option.

An exhaustive comparison of several optimisation techniques, heuristics vs. deterministic, is beyond the scope current discussion. However, for modern optimisation applications an ambidextrous approach or even a hybrid one seems more viable. The strength of evolutionary approach comes from their flexible architecture though non-deterministic performance. Custom designed hybrid optimisation algorithms can exploit both these properties.

5. POTENTIAL AREAS OF EVOLUTIONARY SEARCH AND OPTIMISATION

The goal of this section to review the areas and optimisation problems being successfully addressed by evolutionary methods. Although, reader is guarded that this is not an exhaustive or complete representation, rather just an attempt to highlight the penetration of evolutionary methods in various application domains. In light of the developments occurring in other computational techniques, we also discuss new interdisciplinary ventures where GAs can be fruitfully applied in future.

In recent years, the role of GAs has advanced from simple optimisation and search (Type A) to discovery of new knowledge and principles (Type B). Type A research is direct and straight-forward application of GA methods. The success of this approach relies upon two aspects: (i) the performance of GA itself on the search or optimisation problem and, (ii) the accuracy of problem representation in GA architecture for e.g. analytically or experimentally derived models or simulation results employed for numerical estimates of fitness. In order to succeed and obtain a practically viable solution both (i) and (ii) requirements should be met. To ensure a good GA performance the discussion from Section 4 becomes relevant. However, the (ii) requirement is more domain-specific to the problem.

Type B research is a more recent development in evolutionary methods. In this type of work, evolutionary search has been employed to identify analytical laws that underlie physical phenomena in nature [18]. Here, Symbolic regression (an evolutionary based method) was employed for searching the space of mathematical expressions while minimizing various error metrics. Applying evolutionary based searches for developing phenomenological models in physical sciences is a very promising future direction of research.

The success of both Type A and B types of work is an outcome of underlying effectiveness of evolutionary search.

It is expected that development of more robust GA (or like) methods and insights on performance in a wide-range complex scenarios will further its application. Some areas of application where GAs have found utility are listed (in random order) as follows:

- Image processing
- Tomography
- Computational Fluid Dynamics
- Reservoir Optimisation in oil-fields
- Weather prediction
- Evolutionary Robotics
- Energy Systems
- Online Control
- Structural Optimisation
- Medical Imaging
- Space applications
- Molecular Dynamics
- Financial Markets
- Engineering Design
- Rapid Prototyping
- Manufacturing Sciences
- Drug Design and Pharmaceuticals
- Protein Folding
- Scheduling
- Vehicle Routing
- Micro Electro-Mechanical Systems

The above list is of-course not exhaustive but only hints on the variety and diversity of the applications in which GAs are finding utility nowadays. Several of these areas are posed with single objective, multi-objective, multi-modal, dynamic, combinatorial, etc. optimisation tasks. It is again worth stating that success of GA based search or optimisation (Type A) for any of the above or other domains relies both on criterion (i) and (ii).

It is expected that in coming times GAs will find increasing application in optimisation of large scale engineering systems which are highly multi-disciplinary. The unique GA trait allows flexible encoding the problem, freedom to define customised genetic operations, and ability to handle a variety of optimisation tasks makes them attractive for future Optimisation and Design of Advanced Mechanical Systems, Advanced Manufacturing processes, etc.

Real world challenges also serve as an opportunity for proposing and devising new evolutionary methods. For e.g., Computational Evolutionary Embryogeny [19] was recently proposed and effectively applied to evolve the configurations of 3-D structures. The central idea behind this approach was to represent structures in form of cells which grew according to certain rules. This approach arguably exhibited advantages over conventional evolutionary approaches for structural problems.

Thus, it is likely that a particular evolutionary architecture or representation may be naturally more suitable for a certain application and whenever an opportunity presents researchers should expend efforts on identifying such evolutionary settings. However, the challenges listed in Section 4 still need to be addressed.

The above mentioned structural problem falls in Type A category and it is worth emphasizing that success of above approach in finding acceptable 3-D structures comes from accurate estimation of structural properties of a finite element mesh. The modern simulation tools are fairly reliable and that is a foreseeable drive for coupling and applying evolutionary methods for search and optimisation. For e.g. similar approaches could be applied for design of novel composites (exhibiting uniquely desired properties which rely upon arrangement of constitutive materials and their arrangements within the volume). With advances in multiphysics (approaches considering multiple physical models, for e.g., combining chemical kinetics and fluid mechanics, FEA with Molecular Dynamics, etc.) and simulation tools, GAs can be applied for designing metamaterials (artificial materials engineered to have properties that may not be found in nature), functionally gradient materials (materials exhibiting directional properties), nano-textured surfaces, etc. The prediction of such material properties based on analytical models is limited or almost impossible, thus using simulation packages in conjunction with GA based search in order to meet specified requirements is a promising direction for future. GAs could not only be useful in the design of such materials; but can optimize the overall manufacturing and fabrication steps.

Another promising area for evolutionary methods yet to be explored is in medical engineering and sciences. The earliest attempts made by Holland were directed towards understanding *adaptation*. As stated in [10] the three main components needed to investigate *adaptation* are:

1. The environment, E, of the system undergoing adaptation.
2. The adaptive plan, τ, whereby the system's structure is modified to effect improvements.
3. A measure, μ, of performance, i.e., the fitness of the structures for the environment

Above is simplest and most general form to represent adaption. With vast modern computing capabilities, artificial settings can be created to study the growth and multiplication of viruses. The obvious challenge lies in specifying E, τ and μ. One plausible route is to utilize extensive experimental data and observations available from existing studies; the artificial settings can be trained or itself evolved until a benchmark performance (similar to experimental observations) is obtained on the spread and growth of organisms. If successful, this approach would lead to identification of rules that govern evolution of several deadly viruses. Artificially Life (AI-Life) methods have been employed to simulate population behavior; however the adaptive systems approach seems more attractive, as their genesis is rooted in the metaphor of biological evolution whereas AI-life methods are modeled on cognition.

The scope of applications discussed in this section has completely excluded Genetic Based Machine Learning methods, but these methods by no means should be disregarded for their aptness. Broadly speaking, GBML have promising potential in nueroscience applications for "cognition and learning" and feature selection for classification purposes.

6. MOVING FORWARD

With computational methods dominating all domains of science and engineering, the evolutionary methods will continue to find an increasing application. The implicit parallelism of GAs arising from their population based search approach enables them to concurrently explore different parts of the search spaces. The flexibility of a GA structure allows us to define customised selection, crossover and mutation operations, and also incorporate other features to improve the GA search. Nature's abundance can be sought for both inspiration and new mechanisms in order to solve different classes of problems.

The existing mathematical analyses and deductions on GAs performance are based upon several simplifications and assumptions, and therefore in real contexts the performances show deviations from predictions. Thus, tools and practical strategies to study and understand the performance of GAs will enable their deployment with greater degree of reliance. Even approximate or weak closed form expressions on convergence will be useful. The coming times will continue to find newer and increasing applications for evolutionary methods as the underlying mechanics is better understood.

Stand-alone discovery of new evolutionary-like methods will continue, however the motivation for their existence and implications should be made clear in modern array of different evolutionary methods. Emergence of new evolutionary schemes targeted towards specific applications are likely to ensue a quantum jump.

A classic article [4], discussed the aspect of borrowing metaphors from biology and brain and developing machine learning techniques, concluded by saying:

"As the journey continues, we are confident that an approach based on the abstraction of natural example, combined with careful theoretical and computational investigation, will continue to chart useful territory on the landscape of machine learning."

Past decades have indeed witnessed this development and in addition have marked equal if not more applications of GA for search and optimisation.

The discussion presented in this article is aimed at motivating the practioneers, both, in GA and outside community to apply evolutionary methods in advancing frontiers of engineering and discovering new scientific principles. The key question to ask in any such attempt would be "What is the best way to design GA?".

7. REFERENCES

[1] I. Aitokhuehi, L.J. Durlofsky, V. Artus, B. Yeten, and K. Aziz. Optimization of advanced well type and performance. In *Proceedings of 9th European Conference on the Mathematics of Oil Recovery*.

[2] D. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, New York, 1989.

[3] David E. Goldberg. Genetic and evolutionary algorithms come of age. *Communications of the ACM*, 37(3):113–119.

[4] David E. Goldberg and John H. Holland. Genetic algorithms and machine learning. *Machine Learning*, 3:95–99, 1988.

[5] D. E. Golderg. Computer-aided gas pipeline operation using genetic algorithms and rule learning (doctoral dissertation, university of michigan). Dissertation Abstracts International, 44(10), 3174B. (University Microfilms No. 8402282).

[6] J.H. Holland. Outline for a logical theory of adaptive systems. *Journal of the Association for Computation Machinery*, 3:297–314, 1962.

[7] John H. Holland. Genetic algorithms. `http://www2.econ.iastate.edu/tesfatsi/holland.gaintro.htm`.

[8] John H. Holland. Hierarchical descriptions of universal spaces and adaptive systems. (Technical Report ORA Projects 01252 and 08226). Ann Arbor: University of Michigan, Department of Computer and Communication Sciences.

[9] John H. Holland. Hierarchical descriptions of universal spaces and adaptive systems. In A.W. Burks (Ed.) Essays on cellular automata (pp. 320-353). Urbana: University of Illinois Press.

[10] John H. Holland. *Adaption in Natural And Artificial Systems*. MIT Press, 1975.

[11] Jeffrey Horn, David E. Goldberg, and Kalyanmoy Deb. Implicit system: Niching in a learning classifier nature's way. *Evolutionary Computation*, 2(1):37–66, 1994.

[12] K. A. De Jong. An analysis of the behavior of a class of genetic adaptive systems (doctoral dissertation, university of michigan).

[13] Kenneth A. De Jong. *Evolutionary Computation: A Unified Approach*. MIT Press, 2006.

[14] Kenneth A.De Jong. Genetic algorithms are not function optimizers. *Foundations of Genetic Algorithms*, 2:5–17, 1993.

[15] Kenneth De Jong. Genetic algorithms: A 30 year perspective.

[16] J.D. Lohn, D.S. Linden, G.S. Hornby, and W.F. Kraus. Evolutionary design of an x-band antenna for nasa's space technology 5 mission. In *Antennas and Propagation Society International Symposium, 2004. IEEE*, volume 3, pages 2313 – 2316 Vol.3, june 2004.

[17] Nikhil Padhye, Piyush Bhardawaj, and Kalyanmoy Deb. Improving differential evolution through a unified approach. *Journal of Global Optimization*, pages 1–29, 2012.

[18] M. Schmidt and H. Lipson. Distilling free-form natural laws from experimental data. *Science*, 324(5923):81–85.

[19] Or Yogev, Andrew A. Shapiro, and Erik K. Antonsson. Computational evolutionary embryogeny. *IEEE Transactions on Evolutionary Computation*, 14(2), 2010.