INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED'09
24 - 27 AUGUST 2009, STANFORD UNIVERSITY, STANFORD, CA, USA

A DIALECTICAL APPROACH TO SOLVE INVENTIVE PROBLEMS USING GENETIC ALGORITHMS AND TRIZ: SEARCHING FOR A COMPUTER AIDED INNOVATION SHELL

Roberto Duran-Novoa, Noel Leon-Rovira, Eduardo Uresti
Instituto Tecnológico de Monterrey, México

ABSTRACT

Inventive problems need creativity to be solved, which is usually believed to be beyond comprehension and, thus, methodology. TRIZ and Genetic Algorithms (GAs) have shown that this is at least debatable. TRIZ (Russian acronym for Theory of Inventive Problem Solving) has proven its usefulness as a problem-solving method, as shown in the many practical applications and publications that have increased its presence in the product innovation process. With regard to GAs, these mimic the laws of natural selection to optimize technical systems. With the patentable results using GAs that some researchers have been able to generate, expectations about their possible contribution to enhance and accelerate the product innovation process are on the rise.

This article offers an analysis of the possibility of how both tools (TRIZ and GAs) may converge under the Laws of Dialectics, creating a new conceptual computational framework for aiding and enhancing problem solving and, thus, innovation. Relevant studies and potential opportunities are analyzed for the purpose of proposing a research approach towards a Computer-aided Innovation Shell.

Keywords: Innovation, optimization, dialectics, Triz, genetic algorithms.

1 INTRODUCTION

Problem solving is considered the most complex of all intellectual functions and is based on higher-order cognitive processes. In his book *How to Solve It* George Pólya provides general heuristics for solving problems of all kinds as a four-step approach [1]. Generations of scientists and engineers have relished Pólya's instructions on stripping away irrelevancies and going straight to the heart of the problem. Basically, there are two kinds of problems: those with known ways of solution, and those without them. Problems of the second type are called “inventive problems” (IP) [2]. Problem solving is inherent to product design, and inventive problem solving is required during product innovation processes.

An inventive problem may be defined as a human perception of a situation that has to be changed, but with at least one obstacle which impedes achievement of the desired goal. Such obstacles constitute a state of difficulty that needs to be resolved. In general, IPs are solved using heuristics, which in practice means trial and error and/or inspiration. Several methods have been developed to stimulate creativity for IP solving (Mind Mapping, Brain Storming, Morphological Matrix, C-Sketch, among others [3]), but none of them has tried to find a scientific method for creativity as arduously and at such depth as TRIZ.

TRIZ is a romanized acronym for the Russian “Теория решения изобретательских задач”, meaning “Theory of Inventive Problem Solving”. TRIZ has emerged as a problem-solving methodology based on the study of solution patterns of real-world problems. Research in this field began in 1946, with the hypothesis that creative innovations are based upon universal principles that can be identified, codified, and then be taught to make the process of creativity more predictable. Underlying this definition is the belief of TRIZ’s founder, G.S. Altshuller, that creativity could be turned into an “exact science”. This spirit can be found in the following words: *“Although people who have achieved a great deal in science and technology talked of the inscrutability of creativity, I was not convinced and disbelieved them immediately without argument.* ***Why should everything but creativity be open to scrutiny?*** *What kind of process can this be which unlike all others is not subject to control?”* [4].

Another way of looking for solutions for an IP is to think that the right answer is somewhere "out there", and that we are just ignorant about it. Computational optimization methods have become the most competent alternative to find and decide on the most convenient set of parameters that accomplish an objective. Among the most efficient computerized optimization methods nowadays are genetic algorithms (GAs) [5], a nature-inspired method that emulates natural genetic evolution. In section 3.1 a detailed explanation of how GAs work is given.

Why use GAs instead of other methods? There are numerous alternatives in many areas, but the versatility of evolutionary algorithms, in general, and GAs, in particular, makes them suitable for design-creativity tasks [6]. Also, it is reasonable to believe that the human evolution and the human creativity processes are closely related; however, the most important reason is the bulk of work that has been done in GAs regarding creativity. The words of Goldberg [7], one of the most cited GA authors, summarized this idea: "*In the latter category (configuration, staging, and planning), are activities that go by a variety of names—words such as innovation and creativity come to mind—but the important characteristic to keep in mind here is that these activities are often thought to be beyond the reach of computation*". Along this line of thought, some researchers have been able to replicate human inventions by computational means, and even to generate patentable results [8]. In situations like these, the borderline between optimization and creativity becomes blurred.

In brief, TRIZ and GAs are pointing in the same direction, but this is not enough to obtain a synergistic interaction, which is why dialectic laws are used as a framework to generate a coherent method integrating both tools as the basis for developing a computer-aided innovation shell.

2 DIALECTICS

The modern concept of Dialectics goes back to Hegel. His main contribution was the proposal of "dialectic evolution", which contains the idea of the inherent contradiction within every entity, the source of its self-movement and, thus, evolution. In essence, it can be said that Dialectics is the philosophy of how transformations operate [9].

2.1 The Three Laws of Dialectics

The order and name of dialectics law differ among sources, but without substantial changes in their meaning. They are as follows [10]:

a) Law of Opposites: Also known as "law of unity and struggle between opposites", it basically states that everything in nature is composed of a pair of opposites that together create an indissoluble unity. This kind of thought can be found in millenary literature (probably the oldest reference is the "Tao te King", attributed to Lao-Tse), and there are many understandable examples: electricity's positive and negative charge, an atom's protons and electrons, light and darkness, birth and death, and so on. This can be understood in another sense as the "identity": under certain conditions, the opposites turn into each other. As an example, a warm object gets cold and vice versa. Behind this unity and contradiction lies the reason that makes each entity dynamic, providing constant motivation for movement: the struggle between opposites.

b) Law of Negation: Everything in nature after coming into being grows, becomes mature, and then dies. The law of negation accounts for this tendency that leads to increasing the quantity of all things as a result. This is complementary to the law of opposites, which produces conflicts in each element and gives them motion, to later negate its own nature. This dynamic process of birth and destruction is what causes entities to advance, to evolve. This is also referred to as the cycle of thesis-antithesis-synthesis. In essence, everything in the material world is in a constant renewal process, where old entities die and new ones are born, in a spiral path from the inferior towards the superior.

c) Law of Transformation: Objects and phenomena are characterized by qualitative and quantitative aspects. Qualitative aspects refer to the distinctive qualities that define an entity conceptually, and quantitative aspects are related to magnitudes. The law of transformation states that continuous quantitative development results in qualitative changes sooner or later. The law of opposites makes also true the statement that quality and quantity are mutually interrelated. As an example, if a new kind of wheat were to be developed, there would first be a lot of experiments, every one slightly different from the others, and then the most productive one (quality) would be selected to increase the amount of wheat (quantity).The changes happen in "leaps" that interrupt the normal rhythm of development. These leaps are the decisive transformation between an old quality and a new one, but they happen because of the long process of accumulating quantitative changes.

3 GENETIC ALGORITHMS AND TRIZ

3.1 How Genetic Algorithms Work

GAs are part of evolutionary computing. They are basically blind trial and error procedures that assign values to function parameters, inspired in the evolution process observed in nature. In sexual reproduction, each parent's genes—combined with random genetic mutation—create organisms with new characteristics, and only the most fit organisms have higher possibilities of surviving and passing on their genes to succeeding generations. GAs follow that idea within a computational framework [11], acting on a population P(t) of candidate solutions for exploration in the search space by introducing variations into the population by means of idealized genetic recombination operators. The most important recombination operator is called *crossover* (Table 1). By means of the crossover operator, two structures in the new population exchange portions of their internal representation. Additionally, there is *mutation*, a secondary operator that increases the variability of the population by randomly changing each bit position of the structure in the new population with a probability equal to the mutation rate M. During each iteration step, called a *generation*, the structures in the current population are evaluated, and, on the basis of those evaluations, a new population of candidate solutions is formed by means of the recombination operators using the individuals of the former generation that showed the best performance. Then the edited "survivors" constitute the new generation to be processed. This cycle continues until a certain criterion is reached. Experimental studies indicate that GAs exhibit extremely high efficiency, consistently outperforming both gradient techniques and various forms of random search [12].

Table 1. An example of crossover

Parent Chromosomes (potential solutions)	Children, if cross point is the third position	Children, if cross point is the sixth position
1 1 1 1 1 0 0 0 0 0	*1 1 1* 0 0 1 1 1 1 1	*1 1 1 1 1 0* 1 1 1 1
0 0 0 0 0 1 1 1 1 1	0 0 0 *1 1 0 0 0 0 0*	0 0 0 0 0 1 *0 0 0 0*

The GA cycle can be summarized as follows [13]:

a) [**Start**] Generate random population of n chromosomes (suitable solutions for the problem)
b) [**Fitness**] Evaluate the fitness f(x) of each chromosome x in the population.
c) [**New population**] Create a new population by repeating following steps:
   i. [**Selection**] Select two parent chromosomes from a population according to their fitness (the better the fitness, the greater the chance to be selected).
   ii. [**Crossover**] With a crossover probability, crossover the parents to form new offspring (children). If no crossover was performed, offspring are an exact copy of the parents.
   iii. [**Mutation**] With a mutation probability, mutate new offspring at each position in the chromosome.
   iv. [**Accepting**] Place new offspring in a new population.
d) [**Replace**] Use the new generated population for another run of the algorithm.
e) [**Test**] If the end condition is satisfied, stop and return to the best solution so far.
f) [**Loop**] Return to b)

Over the generations it is expected that the results will converge towards an optimum (Figure 1). The size of the population and number of generations are critical to achieve this task, however, to increase them implies more computational time.

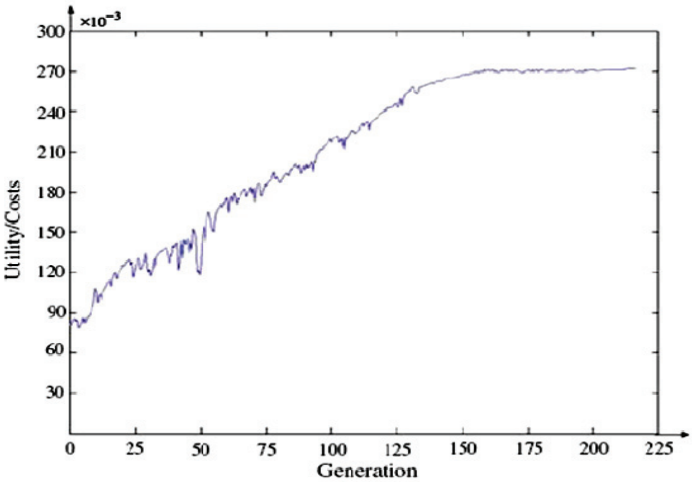

Figure 1.An example of evaluation improvement throughout generations **[**14]

3.2 TRIZ 40 Inventive Principles

TRIZ research has proceeded in several stages over the last sixty years. The three primary findings of the studies are the following [15]:

- Problems and solutions are repeated across industries and sciences. The classification of the contradictions in each problem predicts the creative solutions to that problem.
- Patterns of technical evolution are repeated across industries and sciences.
- Creative innovations use scientific effects outside the field where they were developed.

One of the most successful tools is the "40 Inventive Principles" (40-IP) [16]. These principles consist of a group of generic solutions that solved technical contradictions across many fields, which were deducted after analyzing millions of patents [15].The process of solving problems using the 40-IP consists basically of the following:

- Convert the specific problem into a general one.
- Look at the general solutions proposed.
- Translate one of them into a specific solution to your problem

A diagram of this process is shown in Figure 2.

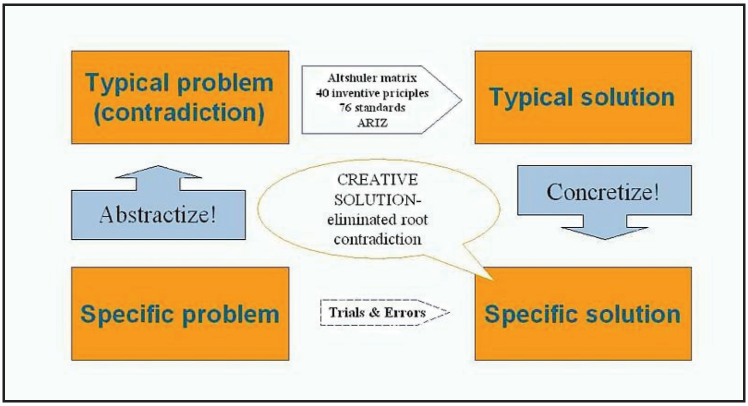

Figure 2: How TRIZ works [17]

The 40-IP (Table 2) are “40 basic methods of eliminating contradictions” [4]. They became popular mainly because of the “contradiction matrix” [18], on which the principles and the contradiction that they solve are placed in a format easy to handle, even for beginners.

Table 2. The "40 inventive principles", definitions and examples in [4][16][19]

1. Segmentation	11. Beforehand cushioning	21. Skipping	31. Porous materials
2. Taking out	12. Equipotentiality	22. "Blessing in disguise"	32. Color changes
3. Local quality	13. 'The other way round'	23. Feedback	33. Homogeneity
4. Asymmetry	14. Spheroidality - Curvature	24. 'Intermediary'	34. Discarding and recovering
5. Merging	15. Dynamics	25. Self-service	35. Parameter changes
6. Universality	16. Partial or excessive actions	26. Copying	36. Phase transitions
7. "Nested doll"	17. Another dimension	27. Cheap short-living objects	37. Thermal expansion
8. Anti-weight	18. Mechanical vibration	28. Mechanics substitution	38. Strong oxidants
9. Preliminary anti-action	19. Periodic action	29.Pneumatics and hydraulics	39. Inert atmosphere
10. Preliminary action	20. Continuity of useful action	30. Flexible shells and thin films	40. Composite materials

4 RELEVANT RESEARCH IN THE AREA

In his 1996 article, J. Gero [20] declares that design can be conceived as a “purposeful, constrained, decision-making, exploration and learning activity”. Creative design perturbs the schema to produce unexpected and incongruous results, which are still understandable either in a current or shifted context. He aims to develop a process-oriented view of computational design creativity, using the notions of unexpectedness and emergence**.** *Emergence* allows for the introduction of new behaviors and new functions and is the equivalent of a designer refocusing his/her attention and/or reinterpreting the results of his/her actions so far. New behaviors and new functions may emerge, which is the equivalent of changing the environment of the phenotype since the behaviors and functions represent the environment in such systems and, hence, the fitness for the environment.

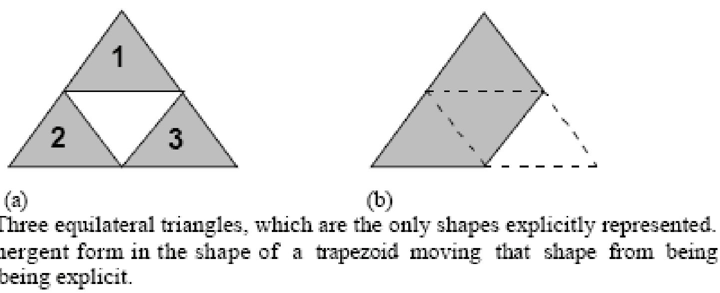

Figure 3. An example of emergence [20]

Using computational models, J. Gero proposes searching and exploring based on structures to be produced and evaluated. The most relevant operators are genetically inspired: combination, analogy, and mutation. To allow creative design, the length of the “individual” needs to be extended to include all aspects of design (function, behavior, knowledge, structure) and to increase the chances of emergence. The “law of transformation” is implicit in the above.

An example of this "transformation" spirit can be found in antenna design. Considered "a dark art", scientists automated trial and error using GAs. The result was a "corkscrew" antenna (Figure 4) that resembles nothing in previous designs, but achieves a significant improvement in performance [11]. This innovation was possible because the GA explored an area where no one had looked before, and through generations accumulated small improvements, leading finally to a completely new (and better) design.

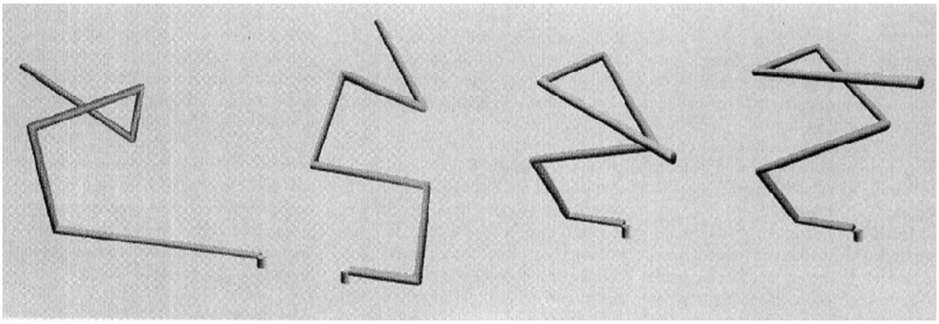

Figure 4. Antenna design using GAs [11]

In situations when quantitative changes do not deliver satisfactory results, it is suggested that new searches must be carried out through qualitative changes or paradigm shifts. A representation of the optimal solution search process is the "Pareto front" (Figure 5), which shows that there are constraints in the solution space that prevent the functions from reaching the "ideal performance". To displace this boundary, it is necessary to find new conceptual solutions that allow elimination of the constraints that hinder achieving the "ideal performance". Currently available CAD/CAE systems were originally conceived to facilitate only parametric variations on modeled parts. Adding to CAD/CAE systems the capacity to fulfill other types of variations autonomously, such as topological and shape redefinitions of the parts and assemblies, is one basis for the possibility of finding new concepts for searching through new dimensions for achieving "ideal results" (Figure 6) [21].

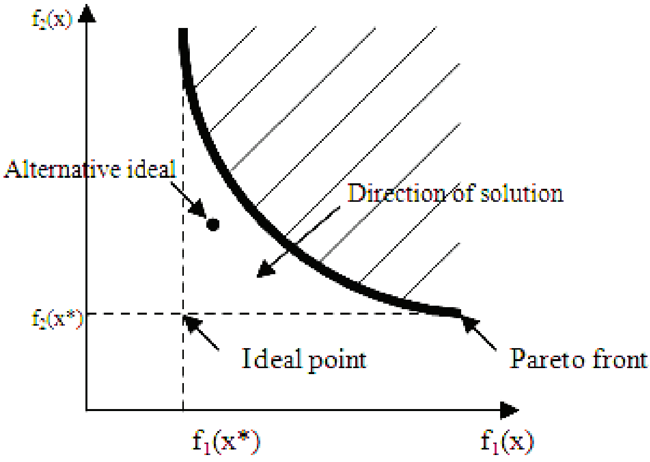

Figure 5. Pareto Diagram [22]

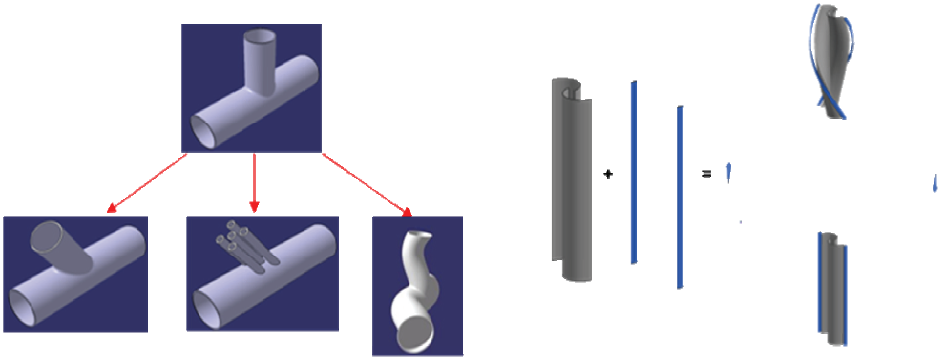

Figure 6. Examples of shape-topological variations and hybridizations [22]

Exploring the limits of the design space and automated creativity, researchers have been able to replicate patented solutions: one of the most important inventions of the 20th century in electrical engineering was the invention of negative feedback by AT&T's Harold S. Black (1927). This discovery "can be readily replicated by an automated design and invention technique patterned after the evolutionary process in nature, namely, genetic programming"[8]. Table 3 shows how the results improved with the availability of computer power.

Table 3. Progression in results by increasing the computational power [8]

System	Period	Speed-Up	Qualitative Nature of Results Produced by Genetic Programming
Serial Texas Instruments LISP	1987–1994	1 (base)	• Toy problems of the 1980s and early 1990s from the fields of artificial intelligence and machine learning
64-node Transtech transputer parallel	1994–1997	9	• Two human-competitive results involving 1-dimensional discrete data (not patent-related)
64-node Parsytec parallel	1995–2000	22	• One human-competitive result involving 2-dimensional discrete data • Numerous human-competitive results involving continuous signals analyzed in the frequency domain • Numerous human-competitive results involving 20th-century patented inventions
70-node Alpha parallel	1999–2001	7.3	• One human-competitive result involving continuous signals analyzed in the time domain • Circuit synthesis extended from topology and sizing to include routing and placement (layout)
1000-node Pentium II parallel	2000–2002	9.4	• Numerous human-competitive results involving continuous signals analyzed in the time domain • Numerous general solutions to problems in the form of parameterized topologies • Six human-competitive results duplicating the functionality of 21st-century patented inventions
Long (4-week) runs of 1000-node Pentium II parallel	2002	9.3	• Generation of two patentable new inventions

Since evolution has proven to be a good general strategy for problem solving, the possible interaction with GAs and design methodologies is beginning to be explored. With GAs and TRIZ as complements three Evolutionary Innovation Axioms were proposed [23]:

- The phenotypic representation (observable characteristic) of the model can be hierarchically ordered, with the geometric dimension as its minimal level. The next level is its genotypic representation (chromosome: 0's and 1's).
- A conflict can be expressed as the win-loss relationship between two or more phenotypic characteristics of a product, denominated "target functions" of the system, linked by a unique genotype.
- The genetic operators act at a genotypic level during the evolution of a product, while the innovation operators act at a phenotypic level, which implies a *cataclysmic* change in the product characteristics.

In this context the structure of inventive principles must be redefined, including analogies between GA operators and TRIZ.

Table 4. Some of the TRIZ inventive principles and their genetic interpretations [23]

TRIZ principle	Genetic interpretation
Segmentation	To divide the two individuals and combine the parts
Merging	To unify individuals
"Nested doll"	To insert part of an individual code into another individual
Homogeneity	To make an individual homogeneous.
Extraction	To eliminate a part of an individual
Copy	To replicate the fittest individuals

4.1 Opportunities and Challenges

The related research shows that GAs are a great tool for exploring the limits of design space, and with the notions of unexpectedness and emergence it is possible to solve inventive problems up to patent generation. The search must be carried on through qualitative changes or paradigm shifts that, by expanding the search space can move the Pareto front towards a new optimum.

The main opportunities and challenges are as follows:

- The versatility and multi-objective capabilities of GAs fit the requirements of inventive problems.
- TRIZ principles can be used in optimization as analogous operators, without the necessity of modifying the fundamental GA structure.
- Dialectics brings an appropriate theoretical foundation to both TRIZ and GAs.
- TRIZ experts claim that innovations lie outside "an industry's range of accepted ideas and principles" [24]. GAs have done this in an automatic manner.
- It has been said that "several thousand good engineering heads cannot compete with the billions of years evolution has had to experiment, select and refine" [25]. Using GAs, they can partially solve this drawback in technical evolution.
- GAs, like any other blind optimization algorithm system, are considered "better" if their results are more replicable. In an innovative context, this could be a drawback.
- Mutations are minimal in evolutionary systems. In innovation, radical changes are encouraged, which could affect performance.
- In optimization, mixing problems is not common practice, but in innovative problems it is easy to find products that include the features of others.
- To effectively apply a GA, a "fitness function" is necessary. But, the problem is how to evaluate "inventiveness"?
- In human heads, the search space is not limited (except perhaps by culture), so the search space is unlimited. The integration of bigger search spaces could create computational problems.

5 INTEGRATION OF GAs AND TRIZ: PROPOSED APPROACH

There are almost no published studies connecting TRIZ and GAs (Science Direct and ProQuest searches deliver no relevant results) probably because the subject has been treated only superficially, despite the fact that there is an evident and complementary desire to optimize, predict and innovate. In other words, both methods look to increase the performance of the innovation process in a complex environment.

One of the most difficult problems in multi-objective optimization is to determine how to measure the quality of a solution; the most common technique is "visual inspection" [26]. Setting arbitrary goals, performing multiple runs, and applying statistical procedures constitute proposals, but arbitrary goals are not easy to define either.

5.1 TRIZ Hierarchical Structure

It can be said that TRIZ has an inductive origin (patent analysis), and that is probably one of the main reasons why the 40-IP are the subject of critics, due to their non-logical sequencing, overlap, abstraction level, and the difficulty in remembering all of them [27]. With Dialectics as the origin, the inventive principles can be reorganized and extended into a more deductive-axiomatic structure, which must be done to make them compatible with the way GAs (and evolutionary algorithms in general) work. The principal assumptions or axioms to do this were as follows:

- There are inventive principles repeated and yet to be discovered.
- Every principle has an opposite.(Opposites)
- Every principle has levels, where inferior ones are more specific and usually the result of the mix of higher level principles. (Negation)
- Innovations are probably related to superior levels and their interactions, and common solutions can be found in inferior ones. Technological advances or discoveries related to superior levels will influence inferior ones, and vice versa (Transformation).

Considering the previous, the researchers arranged the 40-IP using an affinity diagram [3], which resulted in the following 7 Dialectical Principles (DiP):

1. **Separation** (Unification)
2. **Minimize** Action (Maximize)
3. **Balance** (Unbalance)
4. **Pre-action** (Post-action)
5. **Change** (Uniform)
6. **Displacement** (Alignment)
7. **Closed cycle** (Open cycle)

Together with the basic principles, there are two complementary operators:

- Physical/Temporal
- Integrated/Disintegrated

An easy way to understand this approach is to "convert" a classical 40-IP to this new classification format:

- Principle 2 "Taking out": Could be reconverted into Separation-Physical-Disintegrated. This will classify the principle as a third level one (1 principle + 2 operators).
- Principle 5 "Merging": Could be reconverted into Unification-Integrated. This will classify the principle as a second level principle (1 principle + 1 operator) that can be specified into a third level principle by declaring if the Unification is Physical or Temporal.

A similar process can be carried out with the rest of the 40-IPs, with the exception of "22 Blessing in disguise" and "25 Self-service", since they are declarations of the ideal final result [27] and, consequently, they should be considered in every situation.

5.2 Dialectical Principles in an Evolutionary Environment

If Dialectical Principles are to be used as part of a GA, they need an interpretation feasible for handling by the algorithm. The codification in this situation was thought to be implemented in binary codes because of wider edit options and higher emergence probabilities. The interpretations are the following:

1. **Separation** (Unification): A randomly selected part of the chromosome is separated and replaced with its analogous part the other individual. This is not restricted to a "father-mother" relationship, and therefore a new individual can be composed from several parents.
2. **Minimize Action** (Maximize): It affects the extent of the dialectic operator application.
3. **Balance** (Unbalance): A part or the whole chromosome is made symmetrical.
4. **Pre-action** (post-action): The dialectical operator can be applied before (or after) the Separation.
5. **Change** (uniform): A bit (or a group of bits) of the chromosome is randomly changed.
6. **Displacement** (Alignment): A regular or uniform part of a chromosome is replaced by randomly generated bits.
7. **Closed cycle** (Open cycle): A part of the chromosome is isolated from the general cycle to avoid modifications.

The application of this "Dialectical Evolutionary Algorithm" cycle should be as follows:

a) Analyze the kind of contradiction existing in the inventive problem.
b) Determine the DiP to be used.
c) [**Start**] Generate random population of n chromosomes (suitable solutions for the problem).
d) [**Fitness**] Evaluate the fitness f(x) of each chromosome x in the population.
e) [**New population**] Create a new population by repeating the following steps until the new population is complete.

i. [**Selection**] Select the chromosomes from a population according to their fitness (the more fitness, the greater the chance to be selected).
ii. [**Determine**] the application extent of the DiP.
iii. [**Apply**] the adequate DiP (sequence or parallel)
iv. [**Accepting**] Place new offspring in a new population.
v. [**Replace**] Use the new generated population for another run of the algorithm.

f) [**Test**] If the end condition is satisfied, stop, and return the best solution in current population.
g) [**Loop**] Go to step c.

6 DISCUSSION

The main hypothesis is that GAs and TRIZ can be integrated using Dialectics, and the initial thesis-antithesis is the challenge. The synthesis must not be found through negotiation, as is commonly done in multi-objective optimization approaches, but through paradigm shift as occurs in invention.

The basic structure of TRIZ 40-IP has to be modified and the search algorithm should not be limited to traditional genetic operators. This integration will lead to a new approach that is able to solve inventive problems in a more efficient way

Since innovating requires seeing what others do not, the inclusion of new "search dimensions" will lead from optimization to innovation. From this viewpoint, the difference between traditional computerized optimization techniques and computer -aided innovation (CAI) is being established in this research by considering the capability of expanding the search space for enhanced performance of the technological system, when the search process of the traditional optimization approach does not deliver the required performance enhancement. A computer-aided innovation shell (CAI-shell) is hereafter defined as a new system which, on the one hand, aids in identifying the most likely constraints of the optimization process that have to be eliminated or overcome as they hinder achieving the ideal performance; and, on the other hand , is able to be integrated into CAD/CAE systems that can expand autonomously the original search space into new areas and look for "inventive solutions" likely to deliver the requested performance.

Thus, a CAI-shell is similar to two main ways human inventors look for new inventive solutions:

1. New conceptual approaches based on establishing new shapes, topologies or physical principles that add new capabilities to the existing systems.
2. Hybridizing the original system with others that add value to the new emerging one.

However, adding a new search dimension is the main challenge of this approach because theoretically there are infinite options but that would increase exponentially the computational time. A possible analogy for this situation is the "infinite monkey theorem" [28]: It states that a monkey hitting keys at random on a typewriter keyboard for an infinite amount of time will almost surely type a given text, such as the complete works of Shakespeare. An interpretation of this is that with enough time and/or power, finding any possible solution is likely, but since infinite search time and power are impossible, a rational approach discriminating new search space dimensions is required to narrow the search to available time and power. In this situation, given their experimental foundations, TRIZ operators can point towards new feasible search space dimensions. However, it is still necessary to solve the problem of how to measure the quality of a solution. Since visual inspections are the most commonly used technique, setting arbitrary goals, performing multiple runs and applying statistical procedures have been proposed, but "arbitrary goals are not easy to define either" [26].

A possible approach to attack both problems, i.e., the reduction of the search space and the measurement of the quality of a solution, is to do it indirectly. Instead of reproducing the strong individuals (since we do not know what "strong" means), it is better to "kill the weak". For example, if the delivered solution is part of the conventional body of knowledge that does not solve the inventive problem, then the evaluation should be low, and the individuals with unknown performance should be reproduced.

This **"negative"** approach could also be used to reduce the search space. If the 40-IP shows that some principles are not useful to solve a conflict in the inventive problem definition, then the associated search space can be discarded. If there are **7 DiP**, to eliminate one means a 14% reduction.

If no direction of solution is known, through preliminary evaluations the most promising DiP for each problem group can be analyzed based on performance, with almost no human intervention (and thus prejudgment and mental inertia, among others). However, if indeed we know how to solve the problem (the problem turns into a task), the human intervention could be made from the moment of selection of the operators, which would save considerable amounts of processing time. In other words, the problem characteristics/level will determine the adequate computational "force" to be applied.

It is clear that the relation between TRIZ and GAs, although there are some related studies that can provide guiding criteria, is just beginning. To effectively integrate them, it is necessary first to analyze and develop the links between problem-solving theories and evolutionary computation in general, the greatest challenge being the definition of the "objective function" to evaluate the performance. The **"negative"** approach seems to be more plausible.

The presented work is part of ongoing research aimed at applying these ideas to solve practical and theoretical problems.

The authors acknowledge the support received from Tecnológico de Monterrey through grant number CAT101 to carry out the research reported in this paper.

REFERENCES

[1] G. Polya, *How to Solve It: A New Aspect of Mathematical Method*, Princeton University Press, 1988.

[2] "*What is an Inventive Problem?*. http://www.ideationtriz.com/source/12_Inventive_Problem.htm

[3] K.N. Otto and K.L. Wood, *Product Design: Techniques in Reverse Engineering and New Product Development*, Prentice Hall, 2001.

[4] G. Altshuller, *Creativity As an Exact Science*, CRC, 1984.

[5] D.E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Professional, 1989.

[6] P.J. Bentley, ed., *Evolutionary Design by Computers, Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, 1999.*

[7] D.E. Goldberg, *"The race, the hurdle, and the sweet spot: Lessons from genetic algorithms for the automation of design innovation and creativity," Evolutionary Design by Computers, 1999, pp. 105–118.*

[8] J. Koza, M. AKeane, M. JStreeter, T. PAdams, and L. WJones, *"Invention and creativity in automated design by means of genetic programming," Artificial Intelligence for Engineering Design, Analysis and Manufacturing:AI EDAM, vol. 18, Aug. 2004, p. 245.*

[9] J. Lewis, F. Engels, and K. Marx, *Introducción al Estudio de la Filosofía y del Materialismo Dialéctico (Versión M.Ortuzar), Ed. Frente Cultural México. D. F., 1932.*

[10] O. Yajot, *Que es el Materialismo Dialéctico, Moscú: Editorial Progreso., 1973.*

[11] S. Williams, "*Unnatural Selection*," *Technology Review*, vol. 108, Feb. 2005, p. 54.

[12] V. Prushinskiy, G. Zainiev, and V. Gerasimov, *Hybridization: The New Warfare in the Battle for the Market*, Instantpublisher.Com, 2005.

[13] *"Genetic Algorithm Description - Introduction to Genetic Algorithms"* http://www.obitko.com/tutorials/genetic-algorithms/ga-basic-description.php.

[14] J. Jiao, Y. Zhang, and Y. Wang, *"A generic genetic algorithm for product family design," Journal of Intelligent Manufacturing, vol. 18, Apr. 2007, p. 233.*

[15] E. Domb, K. Barry, and M. Slocum, "*TRIZ - What Is TRIZ?," The Triz Journal*, 2007.

[16] E. Domb, *"40 Inventive Principles With Examples," The Triz Journal*.

[17] *"TRIZ - Wikipedia, the free encyclopedia,"* http://en.wikipedia.org/wiki/TRIZ.

[18] G. Altshuller and L. Shulyak, *"40 Principles: TRIZ Keys to Technical Innovation," Translated by Lev Shulyak, Technical Innovation Center, Worcester, MA, 1998*.

[19] *"Contradiction Matrix and the 40 Principles for Innovative Problem Solving"* The TRIZ journal

[20] J.S. Gero, *"Creativity, emergence and evolution in design," Knowledge-Based Systems, vol. 9, 1996, pp. 435-448.*

[21] N. Leon-Rovira, J. Gutierrez, O. Martinez, and C. Castillo, "Optimization vs. innovation in a CAE environment towards a 'computer aided inventing'environment'," *Proceeding of IFIP 18th World Computer Congress, Topical Sessions, Building the Information Society*, 2004, pp. 487–495.

[22] N. Leon-Rovira, "The future of Computer Aided Innovation," *Computer-Aided Innovation (CAI): Ifip 20th World Computer Congress, Proceedings of the Second Topical Session on Computer-aided Innovation, Wg 5.4/Tc 5 Computer-aided Innovation, Sept*, Springer-Verlag New York Inc, 2008, p. 3.

[23] N. Leon-Rovira and H. Aguayo, *"Algoritmos Genéticos Y Triz: Posibilidades De Integración Y Aplicación En La Sistematización De Los Principios De Inventiva ," Guadalajara, Jalisco, México.: 2008.*
[24] J. Terninko, E. Domb, and J. Miller, *"The Seventy-six Standard Solutions, with Examples ," The Triz Journal, Jun. 2006.*
[25] P. Ball, *"Life's lessons in design," Nature, vol. 409, Jan. 2001, p. 413*.
[26] C.A. Coello, *"An updated survey of GA-based multiobjective optimization techniques," ACM Computing Surveys (CSUR), vol. 32, 2000, pp. 109-143.*
[27] D. Mann, *"Evolving the Inventive Principles," The Triz Journal*, Aug. 2002.
[28] *"Infinite monkey theorem - Wikipedia, the free encyclopedia,"* http://en.wikipedia.org/wiki/Infinite_monkey_theorem.

Contact: Roberto Duran-Novoa
Instituto Tecnológico de Monterrey
Department of Mechatronic Engineering
Av. Eugenio Garza Sada2501, Colonia Tecnológico
CP:64849, Monterrey, N.L.
México
Tel: Int+81 83284000 ext 5346
E-mail: durannovoa@uwalumni.com

Roberto is a doctoral candidate at the Tecnológico de Monterrey, México. He received his BS in M.E. from the Universidad Técnica Federico Santa Maria, Chile, and his MS degree in I.E. from the University of Wisconsin-Madison, USA. He has worked as a planning engineer and part-time professor, and his research interests center on the evolution (development and optimization) of products and processes, based on computer-assisted methods and client-oriented design methodologies.