Effect of Size and Order of Variables in Rules for Multi-Objective Repair-Based Innovization Procedure

Abhinav Gaur and Kalyanmoy Deb
gaurabhi@msu.edu, kdeb@egr.msu.edu
COIN Report Number C2017001
www.coin-laboratory.com

***Abstract*—Innovization is a task of learning common principles that exist among some or all of the Pareto-optimal solutions of a multi-objective optimization problem. Except a few earlier studies, most innovization related studies were performed on the final non-dominated solutions found by an EMO algorithm. Recently, authors showed that these principles can be learned during an optimization run and simultaneously utilized in the same optimization run to repair variables to achieve a faster convergence to the Pareto-optimal set. Different principles learned during an optimization run can not only have different number of variables but, may also have variables that are common among a number of principles. Moreover, a preference order for repairing variables may play an important role for proper convergence. Thus, when multiple principles exist, it is important to use a strategy that is most beneficial for repairing evolving population of solutions. This paper makes a first attempt to assess and understand the effect of different strategies to make innovization-based repair of variables most useful. Based on results on test problems, the paper also makes useful suggestions, which require immediate further experimentation on more complex and real-world problems.**

I. INTRODUCTION

Multi-objective optimization (MOO) problems possess a unique property of having a set of 'equally good' or Pareto-optimal (PO) solutions, which cannot be differentiated without any further preference information. Thus, evolutionary algorithms (EAs) have shown to have an edge in solving MOO problem, simply because EAs are capable of finding and storing multiple Pareto-optimal solutions from generations to generations.

While multiple solutions allow decision-makers to choose a single preferred solution by making a comparative analysis of them, multiple Pareto-optimal (or high-performing) solutions may also provide users with valuable information about the problem being solved. The idea of learning from the PO solutions and deciphering design principles that are unique to the optimization problem was first presented in [1]. This concept is called *innovization* and it is pictorially illustrated in Figure 1.

Since an innovization task involves learning from a set of equivalent solutions, it is appealing to couple this idea to population-based optimization methods such as *evolutionary multi-objective optimization* or EMO algorithms. There are examples in the literature, where the innovization task has been

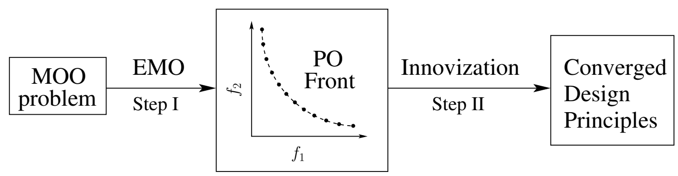

Fig. 1: The concept of innovization is illustrated.

conducted on the results of an EMO algorithm to decipher interesting knowledge about the problem. For example, [2] uses innovization to discover new design principles in three engineering case studies. Another example [3] proposes a novel recommendation tool for software refactoring based on innovization. [4] suggests a way to automate the finding of salient problem design principles for engineering problems. [5] uses the innovization task in a simulation based MOO problem and suggest different ways to couple the innovization with the EMO that can throw light on important principles of the problem.

However, some recent publications [6], [7], [8], [9] point to a growing trend of coupling the ideas of EMO with that of innovization in a way, that makes innovization an integral part of the optimization process itself as shown in Figure 2. Authors

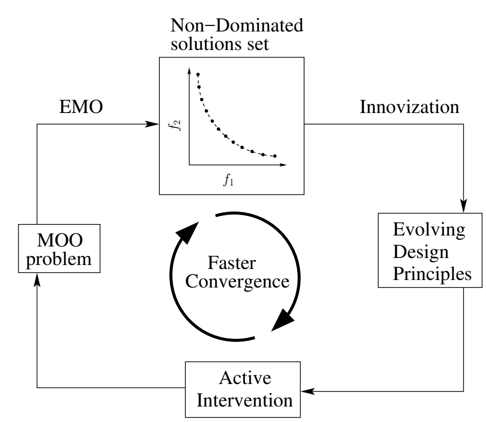

Fig. 2: The concept of EMO/I, a type of online-Knowledge Driven Optimization.

of [6] first discovered the salient principles by innovization task at the end of an MOO and then use those principles as heuristic for local search and obtain a faster convergence than the EMO applied alone. [7] suggests learning the innovized rules using decision trees and then adding the learned rules as if-then-else statement type of constraints during the EMO run. A recently suggested algorithm – evolutionary multi-objective optimization coupled with innovization (EMO/I) [8] – advocates learning principles among design variables having a power law structure and then using them to repair solutions during the optimization process. [9] uses an adaptive operator selection to track the efficacy of each evolutionary operator at consistently creating improving solutions and focus on using the most effective ones. In a recent survey publication [10], the authors have bracketed these methodologies under the umbrella of *Online Knowledge Driven Optimization*.

This work is based on the EMO/I methodology suggested in [8]. The EMO/I used in this work learns rules with either a single variable, and/or power laws involving multiple variables of the population. It then makes variable repairs based on these rules to aid convergence of an EMO algorithm. Power laws are chosen as a target structure because they are widely present in physical systems [11], as well as a recent work [2] shows the presence of such rules in real world engineering case studies. The questions we seek answers of in this paper are as follows: What if there are multiple equally good rules present in the PO set which differ in number of variables involved and may also have variables common to many of these rules? In such a case, different strategies regarding which rules and variables to use first in the repairs can effect the sequence of variable repairs, and thus the overall convergence. We study this problem in this work with some test problems showing different scenarios and also make useful suggestions. We have used NSGA-II [12] algorithm as the EMO for this work.

In the remainder of the paper, we describe the proposed EMO/I method in Section II. Section III then details the innovization based repair strategies that are compared in this work. Thereafter, we present the different MOO problems that were used to test the EMO/I repair strategies along with the parameter values used in the experiments, in Section IV. The results are discussed for all the test problems in Section V. Finally, our conclude our main findings with some future studies inspired from the current work in Section VI.

II. Methodology

Figure 3 shows a flowchart of the EMO/I method proposed in [8]. Except for the three regular blocks, namely *the Rule Basis and Quality* block, *Learn* block and *Repair* block and two decision blocks, namely *L* and *R* blocks, the rest of the flowchart is that of a generic EMO algorithm. This work uses the popular NSGA-II algorithm as the EMO of choice. The following sections provide a brief description of this flowchart with a focus on the aforementioned blocks that are specific to EMO/I.

A. Rule Basis and Quality Block (RBQ)

This initial input to the EMO/I provides data on:

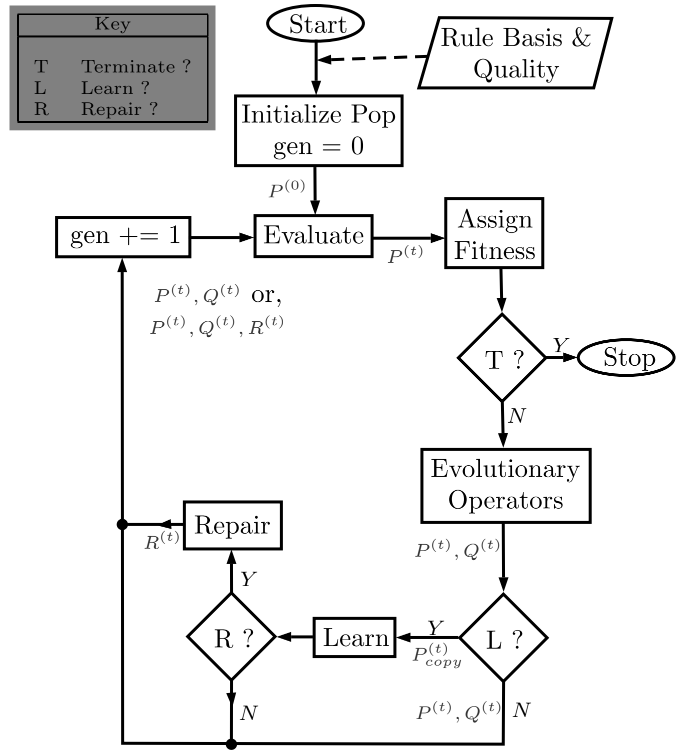

Fig. 3: The flowchart of EMO/I, which is a combination of a regular EMO algorithm with innovization. The algorithm blocks related to innovization are labeled with numbers in Grey circles.

1) the basis functions for the candidate rules and,
2) the threshold quality measure on some rule to consider it good enough for repairing population.

This section explains the former. Quality parameters of the rules will be explained in Section II-D. A *basis function* or *rule basis* in the context of innovization can be any scalar function of the design variables including variables, objectives, constraints and any function thereof. In this work, EMO/I searches for rules of the form given by Equation 1,

$$\Lambda_i \equiv \prod_j^n \lambda_{ij}(\mathbf{x})^{a_{ij}\cdot b_{ij}} = c_i \text{ where,}$$
$$a_{ij} \in \{0,1\}\ ,\ c_i, b_{ij} \in \mathbb{R},\ n = \text{\# of variables, where} \tag{1}$$

Λ_i represents i^{th} candidate rule, λ_{ij} represents the j^{th} basis function in i^{th} candidate rule, a_{ij} is a binary variable that decides if j^{th} basis function is present in i^{th} rule and, b_{ij}, c_i are rule parameters that are evaluated in the learning stage. The set of rules represented by Equation 1 is called *Candidate Rules*. This work considers candidate rules involving only design variables as basis functions. Furthermore, EMO/I in this work is set to look for rules involving a maximum of four variables, i.e. in a problem with n design variables, a maximum of $\sum_{k=1}^{4} {}^nC_k$ candidate rules are evaluated. The RBQ block sets the binary a_{ij} values for the candidate rules.

For example, for a two variable problem, there can be a total of three candidate rules with the rule basis given by Equation 2;

$$\{a_{11}=1,\ a_{12}=0\},\{a_{21}=0,\ a_{22}=1\},\{a_{31}=1,\ a_{32}=1\} \tag{2}$$

1) Assumption on Transition Points: This work defines a *transition point*. A couple of reasons for encountering transition points in PO fronts are;

a) some constraint becomes active (inactive) and forces (eases) the PO solutions to adhere to additional (fewer) rules,
b) the nature of one or more of the fitness functions changes significantly across some point.

This work assumes absence of transition points in the PO solutions. The test problems of Section IV are designed so that there are no transition points in the PO front of the problems. This way, the rules of the form given by Equation 1, if discovered, are applicable to all the PO solutions.

B. Decision Block-L

Once the EMO/I begins and a population of solution individuals is initialized, evaluated, assigned fitness and operated upon by genetic operators (crossover and mutation), the decision block 'L' decides if the algorithm shall begin learning the rules. This is necessary as in the initial phases of EMO algorithm, there may not be many solutions in the non-dominated set to allow any useful rule learning. To detect if an EMO has reached close to the PO front, any metric that can detect close proximity to the PO front can be used. This decision can be implemented many ways, some of which are:

- if some minimum number of function evaluations of EMO/I have passed or,
- if the number of solutions in the non-dominated set is above a threshold and has been stable for some generations etc.

EMO/I in this work uses a fixed value of 33% of maximum function evaluations in all test problems.

C. Learn Block

Once EMO/I meets the criterion set by the L-block, a copy of the parent population, $P^{(t)}_{copy}$, is sent to the 'Learn' block. This block learns the b_{ij} and c_i parameters shown in Equation 1 for each rule from the non-dominated solutions set of $P^{(t)}_{copy}$.

For rules having one variable, i.e. $\lambda_{ij}(\mathbf{x}) = x_j$ (say),

$$a_{ij}=1,\ b_{ij}=1,\ c_i=\mathcal{U}(\mu_i-\sigma_i,\mu_i+\sigma_i),\quad \text{where,} \tag{3}$$

μ_i and σ_i are the mean and standard-deviation respectively of j^{th} decision variable over the non-dominated solutions set, $\mathcal{U}(a,b)$ represents a uniform random distribution from a to b. In this work, the *coefficient of variation* is considered as a measure of the quality for such a rule. This quality parameter is later used the decision block R shown in Figure 3.

For rules having more than one variable, log-linear modeling followed by *multi-variable linear regression* [13] is applied to learn the b_{ij} and c_i parameters. For example, in a three variable problem, say EMO/I is learning the parameters of the following rule;

$$x_1^{b_1}x_3^{b_3}=c,\ \text{then} \tag{4}$$

taking log on both sides,

$$b_1\log x_1 + b_3\log x_3 = \log c,$$

which is a linear equation. If x_1 is chosen as the regressand then,

$$\begin{aligned}\log x_1 &= \frac{-b_3}{b_1}\log x_3 + \frac{\log c}{b_1},\ \text{or}\\ &= \hat{\beta}\log x_3 + \hat{\gamma}\ \text{where}\end{aligned} \tag{5}$$

$\hat{\beta}$ and $\hat{\gamma}$ estimates returned by a *Ordinary Least Square* linear regression (OLSR) method using the values of 1^{st} and 3^{rd} variables from non-dominated set. OLSR also returns the R^2_{adj} value which is later used to asses the quality of such a candidate rule in repair stage.

D. Decision Block-R

This block decides if some rule is good enough in quality to qualify for the the repair stage. As mentioned in Section II-C:

- the quality of one variable rules is measured using *coefficient of variation*, 'C', values and,
- the quality of rules with more than one variable is measured using R^2_{adj} value returned from OLSR.

EMO/I is provided with the maximum coefficient of variation C_{max} and minimum $R^2_{adj,min}$ values in the RBQ block at the start of EMO/I. A rule is said to qualify for repair in next stage as follows:

- for one variable rules, the coefficient of variation of the rule, say C, should be lower than C_{max} and,
- for rules with more than one variable, the R^2_{adj} value must be greater than $R^2_{adj,min}$ value.

If at least one such candidate rule exists, it passes the control to the *Repair* block shown in Figure 3, else it dumps the $P^{(t)}_{copy}$ population and passes the control back to regular EMO/I without making any repairs.

We call the set of rules for which the quality parameters surpass the threshold value of quality set in RBQ block as *Qualifying Rules*, and the union of set of variables constituting these qualifying rules as *Qualifying Variables*.

E. Repair Block

If the decision block-R finds at least one good quality rule, the $P^{(t)}_{copy}$ population is passed to the *Repair* block. Based on some qualifying rule, repair can be made to the entire population $P^{(t)}_{copy}$ as follows.

If the qualifying rule is a one variable rule such as,

$$\Lambda \equiv x_i,\ \text{then}$$

the i^{th} variable in some individual of $P^{(t)}_{copy}$ can be repaired by picking a random number from a *uniform distribution* between

$\mu_i-\sigma_i$ and $\mu_i+\sigma_i$, where μ_i and σ_i are the mean and standard-deviation respectively of i^{th} decision variable over the non-dominated solutions set and $\mathcal{U}(a,b)$ represents a uniform random distribution from a to b. This can be represented as;

$$\hat{x}_i = \mathcal{U}(\mu_i - \sigma_i, \mu_i + \sigma_i).$$

To explain the repair method involved in the second case, i.e. when the qualifying rule has more then one variable, we take the example shown in Equation 4. As shown in Equation 5, values of $\hat{\beta}$ and $\hat{\gamma}$ are obtained by OLSR. Thus, the regressand variable, x_1, can be repaired as

$$\hat{x}_1 = \exp\left(\hat{\beta}\log x_3 + \hat{\gamma}\right) \text{ and}$$

the regressor variable can be repaired as

$$\hat{x}_3 = \exp\left(\frac{\log x_1 - \hat{\gamma}}{\hat{\beta}}\right),$$

depending upon whether the chosen variable is a regressor or the regressand. For other cases as well, a similar logic follows. Such repairs are made to all solution individuals of the $P^{(t)}_{copy}$ population one by one to produce a repaired population which is named $R^{(t)}$ in Figure 3. Note the following important aspects of repairing an individual in $P^{(t)}_{copy}$:

1) At the end of every variable repair, if the repaired variable value lies outside its a priori defined bounds, then the repaired variable is set to the nearest bound value.
2) Recall the terms of 'qualifying rules' and 'qualifying variables' from Section II-D. In repairing an individual of $P^{(t)}_{copy}$, all its qualifying variables are repaired. But the sequence in which the variables are repaired and the qualifying rule used to repair a variable depends on different repair strategies that are explained in the following section.
3) During the repair of variables of an individual of $P^{(t)}_{copy}$, repaired variable values (of individual) in earlier repairs are used in the subsequent repairs.

III. Repair Strategies

This section discusses an important aspect of the Repair block which is the main investigative idea behind this work. The following two choices need to be made before any repair of the kinds illustrated in Section II-E is made to solution individuals of $P^{(t)}_{copy}$:

- choose one of the possibly many qualifying rules on which to base the repair and,
- choose one of the possibly many variables from the chosen qualifying rule.

For example, choosing a random rule from the qualifying rules pool and then choosing a random variable from the variables of the chosen qualifying rule can be one such strategy. This was the strategy which the authors employed in their earlier work [8]. Both these choices have an affect on:

1) the repair parameters used to making the repair and,
2) the sequence in which variables are repaired.

In this work, we tested a combination of three *rule-preference* strategies and three *variable-preference* strategies. The following sections discusses them one by one.

A. Rule Preference Strategies

This work investigated three kinds of rule-preference strategies. We call the number of variables in a rule to be the *length of the rule*. Recall the terms of 'qualifying rules' and 'qualifying variables' from Section II-D. In t^{th} generation of EMO/I, let there be 'm' qualifying rules, i.e. $\Lambda = \{\Lambda_1, \Lambda_2, \ldots, \Lambda_m\}$, with lengths $l = \{l_1, l_2, \ldots, l_m\}$. Then, the strategies are explained below:

I *No preference* : When repairing variables of an individual, no rule is given preference over others and one rule is chosen randomly from all qualifying rules.

II *Prefer long rules* : When repairing variables of an individual, this strategy prefers lengthier rules over shorter rules for getting selected to make a variable repair. The probability of selection of j^{th} rule from this pool is;

$$p_j = \frac{l_j}{\sum_{k=1}^{m} l_k}.$$

III *Prefer short rules* : When repairing variables of an individual, this strategy prefers shorter rules over longer rules for getting selected to make a variable repair. The probability of selection of j^{th} rule from this pool is;

$$p_j = \frac{1/l_j}{\sum_{k=1}^{m} 1/l_k}.$$

In all three strategies, once a variable repair is made based on a chosen qualifying rule to an individual, it is taken out of the pool of qualifying rules for subsequent repairs to be made to the said individual of $P^{(t)}_{copy}$. The sets Λ and l are updated accordingly and replenished again at the start of repair for next individual.

B. Variable Preference Strategies

We investigated three kinds of variable-preference strategies which are based on the frequency of a variable among the qualifying rules. In t^{th} generation of EMO/I, let there be 'm' qualifying variables, i.e. $V = \{v_1, v_2, \ldots, v_m\}$, with frequency in the qualifying rules $f_v = \{f_{v_1}, f_{v_2}, \ldots f_{v_m}\}$. The strategies are explained below:

I *No preference* : When repairing a variable of an individual based on a chosen qualifying rule, no variable is given a preference over others and one variable is chosen randomly from the variables of a chosen qualifying rule for repair.

II *Prefer common variables* : When repairing a variable of an individual based on a chosen qualifying rule, those variables of chosen qualifying rule that are more frequent among all qualifying variables, are given a higher preference for selection. Let variables of a chosen qualifying rule, say R, contain k variables from set V_q, i.e. $R \subseteq V_q$

with cardinality k. Then the probability of selecting one of those k variables, say v_j, of the rule is given by;

$$p_j = \frac{f_{v_j}}{\sum_{k=\{h|v_h \in R\}} f_{v_k}}.$$

III *Prefer uncommon variables* : When repairing a variable of an individual based on a chosen qualifying rule, those variables of chosen qualifying rule that are less frequent among all qualifying variables, are given a higher preference for selection. Let variables of a chosen qualifying rule, say R, contain k variables from set V_q, i.e. $R \subseteq V_q$ with cardinality k. Then the probability of selecting one of those k variables, say v_j, of the rule is given by;

$$p_j = \frac{1/f_{v_j}}{\sum_{k=\{t|v_t \in R\}} 1/f_{v_k}}.$$

In all three strategies, once a variable repair is made based on a chosen qualifying variable to an individual, it is taken out of the pool of qualifying variables for subsequent repairs to be made to the said individual of $P^{(t)}_{copy}$. The sets V_q and f_v are updated accordingly and replenished again at the start of repair for next individual.

These aforementioned strategies when combined together form a total of nine combinations. These are listed in the first nine rows of Table I. The tenth strategy is of a pure EMO algorithm without any innovization based repairs.

TABLE I: Different strategies for variable repair studied in this work.

ID	Rules Preference	Variable Preference	Abbrev.
1	None	None	NN
2	None	Common variables	NC
3	None	Uncommon variables	NU
4	Long rules	None	LN
5	Short rules	None	SN
6	Long rules	Common variables	LC
7	Long rules	Uncommon variables	LU
8	Short rules	Common variables	SC
9	Short rules	Uncommon variables	SU
10	——— Non-Innovized (No Repair) ———		NI

IV. Test Problems

All the test problems in this work have been derived from the ZDT1 [14] problem and have the following form.

$$\begin{aligned} \text{Minimize} \quad & f_1(\mathbf{x}) = x_1, \\ & f_2(\mathbf{x}) = g(\mathbf{x})\ h(f_1(\mathbf{x}), g(\mathbf{x})), \\ \text{Where} \quad & h(f_1, g) = 1 - \sqrt{f_1/g}. \end{aligned} \tag{6}$$

Every problem has a different g function, variable bounds and Pareto-optimal set and they are described in the following sections.

A. ZDT1-1 : No Linkage, Different Lengths of Rules

The ZDT1-1 problem given by Equations 6 and 7.

$$\begin{aligned} & g(\mathbf{x}) = 1 + |x_2 - 0.5| + |x_3 x_4 - 0.5| + |x_5 x_6 x_7 - 0.5| \\ & x_{1,2} \in [0, 1],\ x_{3,4,5,6} \in [0.5, 1],\ x_7 \in [0.5, 2]. \end{aligned} \tag{7}$$

The Pareto-optimal region for this problem is given by Equation 8.

$$\begin{aligned} & 0.0 \le x_1^* \le 1.0,\ x_2^* = 0.5,\ x_3^* \cdot x_4^* = 0.5,\ \text{and}, \\ & x_3^* \cdot x_4^* \cdot x_7^* = 0.5\ . \end{aligned} \tag{8}$$

We can see from Equation 8 that the PO solutions of ZDT1-1 have to adhere to certain rules in the variable space. None of these relations have any variable common among them, but they have different lengths. This problem is designed to test the repair strategies NI, NN, SN and LN. Refer to Table I for description of these strategies.

B. ZDT1-2 : Linked Principles, Same Length Rules

The ZDT1-2 problem given by Equations 6 and 9.

$$\begin{aligned} & g(\mathbf{x}) = 1 + |x_1 x_2 - 0.5| + |x_2^{0.5} x_3 - \sqrt{0.5}| + |x_2^2 x_4 - 0.25| \\ & + |x_2^{-0.5} x_5 - \sqrt{0.5}| \\ & x_{1,2} \in [0.5, 1],\ x_3 \in [\sqrt{0.5}, 1],\ x_4 \in [0.25, 1], x_5 \in [0.5, \sqrt{0.5}]. \end{aligned} \tag{9}$$

The Pareto-optimal region for this problem is given by Equation 10.

$$\begin{aligned} & x_1^* \cdot x_2^* = 0.5,\ (x_2^*)^{0.5} \cdot x_3^* = \sqrt{0.5},\ (x_2^*)^2 \cdot x_4^* = 0.25 \text{ and}, \\ & (x_2^*)^{-0.5} \cdot x_5^* = \sqrt{0.5}\ . \end{aligned} \tag{10}$$

We can see from Equation 10 that the PO solutions of ZDT1-2 have to adhere to certain rules in the variable space. All of these relations have x_2 variable in common, but all of these rules are of the same lengths. This problem is designed to test the repair strategies NI, NN, NC and NU. Refer to Table I for description of these strategies.

C. ZDT1-3 : Linked Principles and Different Length Rules

The ZDT1-3 problem given by Equations 6 and 11.

$$\begin{aligned} & g(\mathbf{x}) = 1 + |x_1 x_2 - 0.5| + |x_1^{0.5} x_2 x_3 - 0.5| + |x_5 - 0.5| \\ & + |x_1^{-0.5} x_2^2 x_3 x_4 - 0.25| \\ & x_{1,2} \in [0.5, 1],\ x_3 \in [\sqrt{0.5}, 1],\ x_4 \in [0.25, 1],\ x_5 \in [0, 1]. \end{aligned} \tag{11}$$

The Pareto-optimal region for this problem is given by Equation 12.

$$\begin{aligned} & x_1^* \cdot x_2^* = 0.5,\ (x_1^*)^{0.5} \cdot x_2^* \cdot x_3^* = 0.5,\ x_5^* = 0.5 \text{ and}, \\ & (x_1^*)^{-0.5} \cdot (x_2^*)^2 \cdot x_3^* \cdot x_4^* = 0.25\ . \end{aligned} \tag{12}$$

We can see from Equation 12 that the PO solutions of ZDT1-2 have to adhere to certain rules in the variable space which not only have variables common among them but also having varying lengths. This problem is designed to test and compare all the repair strategies given in Table I.

This work uses the NSGA-II algorithm as EMO for solving the problems. The corresponding parameters used in solving the above problems are given in Table II.

TABLE II: EMO parameters used in the problems.

	ZDT1-1	ZDT1-2	ZDT1-3
Population Size	52	72	72
Max Func. Evals (FE)	10,400	21,600	21,600
Prob. of Crossover	0.9	0.9	0.9
Prob. of Mutation	1/7	1/5	1/5
Crossover Index	15	15	15
Mutation Index	20	20	20
Start of Learning	0.33 (Max FE)	0.33 (Max FE)	0.33 (Max FE)

V. Results on Test Problems

We now discuss the results for on the test problems one by one and make comparisons in terms of convergence rates for these strategies with each other as well as the no repair NI strategy.

A. *ZDT1-1 Results*

The ZDT1-1 problem is tested with four repair strategies namely, NI, NN, LN and SN where NI is NSGA-II without any repair strategy. The others are NSGA-II with some repair strategies as mentioned in Table I. Figure 4 and 5 show the median Generational Distance (GD) and Inverse Generational Distance (IGD) for 20 runs versus the function evaluations. It can be noted that:

1) All the innovized repair cases perform better than the NI (no repair) strategy in both GD as well as IGD. Hence the innovized repair strategies can be said to be have a faster convergence.
2) The three repair strategies, NN, LN and SN have a comparable performance in terms of median GD as well as IGD. Hence, all three perform equally well compared to each other.

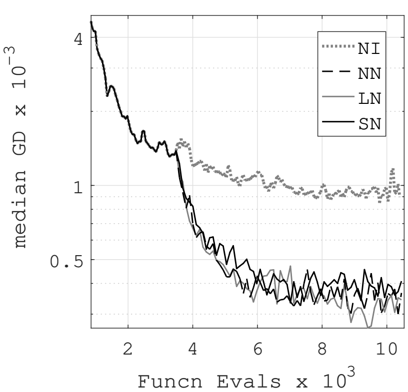

Fig. 4: GD versus Function Evaluations for ZDT1-1 problem. The lines represent a median of 20 runs.

B. *ZDT1-2*

The ZDT1-2 problem is tested with four repair strategies namely, NI, NN, NC and NU. Figure 6 shows the median GD

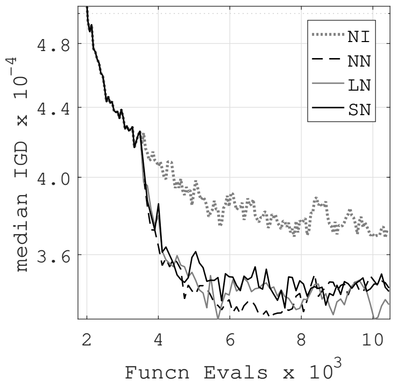

Fig. 5: IGD versus Function Evaluations for ZDT1-1 problem. The lines represent a median of 20 runs.

for 20 runs versus the function evaluations. It can be noted that:

1) All the innovized repair cases perform better than the NI or no repair strategy and hence can be said to be have a faster convergence.
2) The three repair strategies, NN, LN and SN have a comparable performance in terms of median GD.

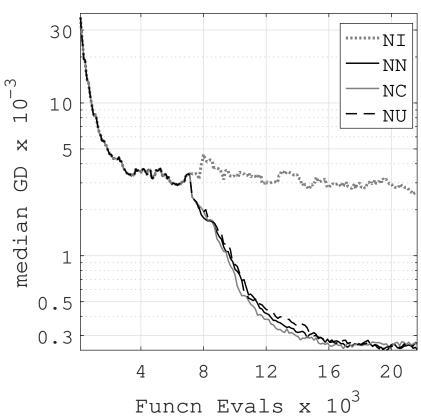

Fig. 6: GD versus Function Evaluations for ZDT1-2 problem. The lines represent a median of 20 runs.

C. *ZDT1-3*

The ZDT1-3 problem is tested with all the ten repair strategies listed in Table I. To avoid clutter, the comparison is shown in three parts.

1) Mixed strategies preferring short rule length: In this section, we present the results of repair strategies NI, SN, SC and SU. Figure 7 shows the corresponding results. As in other cases, NI approach has a slower convergence compared to the repair based strategies. The rest of the three strategies namely, SN, SC and SU all look close but SN looks marginally better than the other two strategies.

To investigate further, we conducted a *Wilcoxon Rank Sum Test* among the distribution of GDs (from the 20 runs) for the three cases at 20,000 function evaluations. The alternative hypothesis of the test, $\mathbf{H}_1$, states that the median GD at 20,00 Function Evaluations (FE) obtained by using SN repair strategy is less than the median GD at 20,000 FEs by using SC or SU strategy being tested at a 5% significance level. The alternative hypothesis is accepted in both cases and the corresponding p-value are shown in Table III.

TABLE III: Table showing the p-value of the *Wilcoxon rank sum test* conducted between the SN versus SC and SU strategies for ZDT1-3 problem at 20,000 function evaluations.

Strategy Comparison	SN vs SC	SN vs SU
p-value	0.0026	0.0152

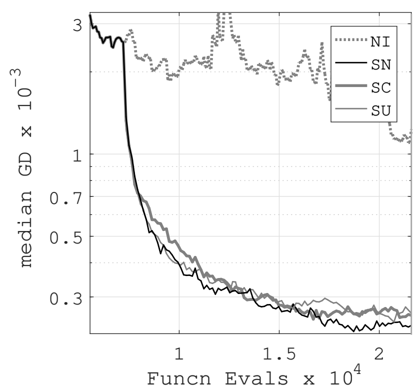

Fig. 7: GD versus Function Evaluations for ZDT1-3 problem. The lines represent a median of 20 runs.

2) Mixed strategies preferring long rule length: Next we compare the long rule preferring strategies, i.e. LN, LC, LU along with NI. Since SN is found to be best performing strategy among the short-rule-preferring strategies, that is also plotted for comparison purposes. Figure 8 shows the plots of median GDs for LN, LC, LU, NI and SN strategies. As can be seen, NI is outperformed by all others. Also, SN seems to be performing much better than the LN, LC and LU strategies.

To ascertain if SN strategy is actually better than LN, LC and LU, we conduct a Wilcoxon rank sum test similar to the one conducted in Section V-C1. As before, the alternative hypothesis of the test, $\mathbf{H}_1$, states that the median GD at 20,000 FEs obtained by using SN repair strategy is less than the median GD at 20,000 FEs by using LN, LC or LU strategy being tested at a 5% significance level. The alternative hypothesis is accepted in all three cases and the corresponding p-values are shown in Table IV.

TABLE IV: Table showing the p-value of the *Wilcoxon rank sum test* conducted between the SN versus LN, LC and LU strategies for ZDT1-3 problem at 20,000 function evaluations.

Strategy Comparison	SN vs LN	SN vs LC	SN vs LU
p-value	6.6×10^{-9}	3.2×10^{-8}	8.7×10^{-11}

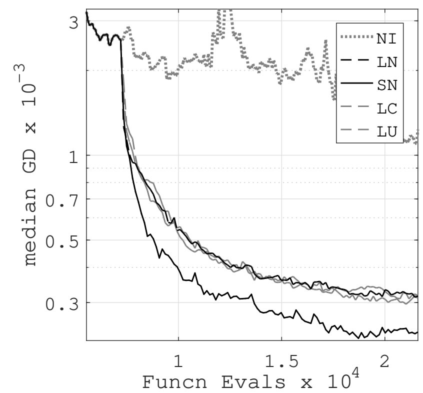

Fig. 8: GD versus Function Evaluations for ZDT1-3 problem. The lines represent a median of 20 runs.

3) Mixed strategies with no preference on rule length: Next we compare strategies which put no emphasis on rule lengths but mixed strategies in terms of emphasis on frequency of qualifying variables. Also, as the SN strategy has been found till now to be better than SC, SU, LC, LU, LN in Sections V-C1 and V-C2, thus strategy SN is also put in the mix for comparison. As can be seen in Figure 9, the SN strategy again seems to be performing better in median GD than the NN, NC and NU repair strategies. The Wilcoxon rank sum test results in Table V show that the SN strategy is better than the other three repair strategies.

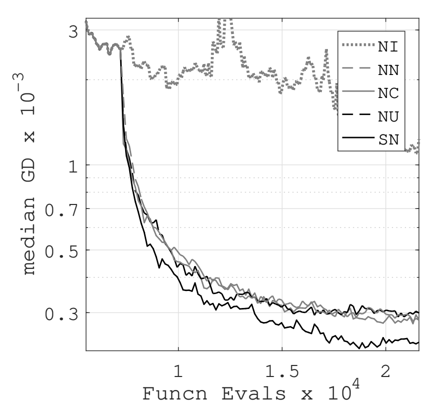

Fig. 9: GD versus Function Evaluations for ZDT1-3 problem. The lines represent a median of 20 runs.

TABLE V: Table showing the p-value of the *Wilcoxon rank sum test* conducted between the SN versus NN, NC and NU strategies for ZDT1-3 problem at 20,000 function evaluations.

Strategy Comparison	SN vs NN	SN vs NC	SN vs NU
p-value	1.3×10^{-7}	0.0013	4.6×10^{-7}

In the next section, we discuss conclude with the main findings of our experimental results. We also discuss some important future directions for this work.

VI. Conclusions

It has been amply demonstrated in the EMO literature that Pareto-optimal solutions possess certain unique properties which can remain as useful solution principles for problem solving tasks. The method of finding multiple Pareto-optimal solutions and a procedure of deciphering them to reveal solution principles were together termed as a task of 'innovization'. In most studies of automated innovization, rules involving independent variable sets and rules having more or less similar number of variables were considered. In this paper, we have initiated a study in which rules with a differing number of variables and common variables have been considered. In such cases, there exists certain obvious and important questions:

1) Should a rule with fewer variables be given preference for repairing population members in an EMO?
2) Should a variable that appears less commonly to multiple rules be repaired with those that appear more commonly?
3) Should any particular order of preference of variables for repair be followed for an efficient use of rules compared to a a random order?

Although these questions may have to be considered together to have the right answer, in this paper, we have considered them one at a time on a few test problems created to isolate each case separately. Based on our limited simulation results presented in this paper, we make the following observations:

1) When qualifying rules have no common variables, then all the repair strategies based on length of rules have similar performance.
2) When qualifying rules have common variables but have the same rule lengths, even then all the repair strategies have similar performance.
3) When qualifying rules have varying lengths as well as common variables among them, then the innovized repair strategy SN, i.e. preferring to repair variables of present in smaller rules over variables present in lengthier rules, works best.

These observations are interesting and are supported by statistically significant results. It will now be interesting to use these observations as principles for a repair-based innovization procedure and solve more complex and real-world problems.

Some aspects of the current EMO/I procedure that we will expand upon in future work are:

a) Extending the EMO/I procedure to handle problems that may have transition points in the PO front. One of the ways to achieve this is to cluster the points in the objective or constraint space to find out different partitions of the PO front over which different variable rules may be applicable.
b) Filtering out of redundant rules that come about from combination of smaller length qualifying rules. Such rules slow down the convergence of EMO/I.
c) Using a non-parametric method to detect if it is safe for EMO/I to learn rules from the solutions. Currently, EMO/I starts learning after 33% of maximum function evaluations. Any metric that can detect close proximity to the PO front can be used for this purpose. For example, in going from generation t to $t+1$, if majority of individuals of population $P^{(t+1)}$ come from $P^{(t)}$ and not from the offspring population $Q^{(t)}$ in NSGA-II, this can be a sign of EMO slowing down and coming close to PO front.

References

[1] K. Deb and A. Srinivasan, "Innovization: Innovating design principles through optimization," in *Proceedings of the 8th annual conference on Genetic and evolutionary computation*. ACM, 2006, pp. 1629–1636.

[2] K. Deb, S. Bandaru, D. Greiner, A. Gaspar-Cunha, and C. C. Tutum, "An integrated approach to automated innovization for discovering useful design principles: Case studies from engineering," *Applied Soft Computing*, vol. 15, pp. 42–56, 2014.

[3] M. W. Mkaouer, M. Kessentini, S. Bechikh, K. Deb, and M. Ó Cinnéide, "Recommendation system for software refactoring using innovization and interactive dynamic optimization," in *Proceedings of the 29th ACM/IEEE international conference on Automated software engineering*. ACM, 2014, pp. 331–336.

[4] S. Bandaru and K. Deb, "Automated discovery of vital knowledge from pareto-optimal solutions: First results from engineering design," in *Evolutionary Computation (CEC), 2010 IEEE Congress on*. IEEE, 2010, pp. 1–8.

[5] A. Ng, K. Deb, and C. Dudas, "Simulation-based innovization for production systems improvement: An industrial case study," in *Proceedings of The International 3rd Swedish Production Symposium, SPS'09, Goteborg, Sweden, 2-3 December 2009*. The Swedish Production Academy, 2009, pp. 278–286.

[6] K. Deb and R. Datta, "Hybrid evolutionary multi-objective optimization and analysis of machining operations," *Engineering Optimization*, vol. 44, no. 6, pp. 685–706, 2012.

[7] A. H. Ng, C. Dudas, H. Boström, and K. Deb, "Interleaving innovization with evolutionary multi-objective optimization in production system simulation for faster convergence," in *Learning and Intelligent Optimization*. Springer, 2013, pp. 1–18.

[8] A. Gaur and K. Deb, "Adaptive use of innovization principles for a faster convergence of evolutionary multi-objective optimization algorithms," in *Proceedings of the 2016 on Genetic and Evolutionary Computation Conference Companion*. ACM, 2016, pp. 75–76.

[9] N. Hitomi, H. Bang, and D. Selva, "Extracting and applying knowledge with adaptive knowledge-driven optimization to architect an earth observing satellite system," in *AIAA Information Systems-AIAA Infotech@ Aerospace*, 2017, p. 0794.

[10] S. Bandaru, A. H. Ng, and K. Deb, "Data mining methods for knowledge discovery in multi-objective optimization: Part a-survey," *Expert Systems with Applications*, vol. 70, pp. 139–159, 2017.

[11] M. E. Newman, "Power laws, pareto distributions and zipf's law," *Contemporary physics*, vol. 46, no. 5, pp. 323–351, 2005.

[12] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *Evolutionary Computation, IEEE Transactions on*, vol. 6, no. 2, pp. 182–197, 2002.

[13] A. C. Rencher, *Methods of multivariate analysis*. John Wiley & Sons, 2003, vol. 492.

[14] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: Empirical results," *Evolutionary computation*, vol. 8, no. 2, pp. 173–195, 2000.