Computational Materials Science 50 (2011) 2502–2516

Contents lists available at ScienceDirect

Computational Materials Science

journal homepage: www.elsevier.com/locate/commatsci

Phases in Zn-coated Fe analyzed through an evolutionary meta-model and multi-objective Genetic Algorithms

Pankaj Rajak [a], Ujjal Tewary [b], Sumitesh Das [b], Baidurya Bhattacharya [c], Nirupam Chakraborti [a,*]

[a] Department of Metallurgical & Materials Engineering, Indian Institute of Technology, Kharagpur 721 302, India
[b] R & D Division, TATA Steel, Jamshedpur 831 001, India
[c] Department of Civil Engineering, Indian Institute of Technology, Kharagpur 721 302, India

ARTICLE INFO

Article history:
Received 18 November 2010
Received in revised form 22 March 2011
Accepted 24 March 2011
Available online 19 April 2011

Keywords:
Fe–Zn system
Hot dip galvanization
Coating
Molecular Dynamics
Optimization
Neural network
Genetic Algorithms
Multi-objective optimization
Evolutionary algorithm
Pareto frontier

ABSTRACT

Various interfaces in Zn coated steel are examined for their stability on the basis of two conflicting requirements of minimum deformation at a maximum absorption of shear energy. The shearing process is simulated using a Molecular Dynamics technique and meta-models of both energy and strain are constructed using an Evolutionary Neural Network that itself evolved through a multi-objective Genetic Algorithm. Simultaneous optimization of deformation and energy absorption is conducted with a Predator–prey Genetic Algorithm and the resulting Pareto frontiers are analyzed and discussed. The findings show good correspondence with existing experimental observations.

1. Introduction

A multi-objective analysis of the Fe–Zn system was presented in a recent work [1] where the performance of an HCP zinc layer over BCC iron was appraised in terms of two conflicting requirements: shear failure at a maximum energy absorption, but with minimum deformation. In course of their routine applications, the coated region in the hot dip galvanized sheets are subjected to shear deformations and the studies of this nature can go a long way towards designing strong and stable coatings over a steel susceptible to severe environmental degradation.

Enough evidence [2,3] is available however to show that the coated layer in the hot dip galvanized iron is not a single phase region, rather it consists of several layers of various phases found in the Fe–Zn phase diagram [4], as shown schematically in Fig. 1. When subjected to shear force, this composite assembly of phases is expected to behave in a substantially different way than the pure Fe–Zn system considered earlier [1]. In fact, the shear resistances at the interfaces between any two pair of phases are expected to vary significantly as both their crystal structures and hardness values are known to be widely different [2,3]. Therefore, in this study, our earlier methodology [1] has been extended to all the interfaces present in a more realistic situation. The details are provided below.

2. The model

The characteristics of the various phases in the coated layer are summarized in Table 1, based upon information available in the literature [5–7]. The idea here is to separately subject the Fe–Γ, Γ–Γ_1, Γ_1–δ, δ–ζ, and ζ–η interfaces to shear forces till a predefined criterion for failure is satisfied. At that instant it is necessary to know both the amount of strain and the total amount of energy absorbed, so that both of them could be simultaneously optimized for each interface using the results of many such numerical experiments with systematic parameter variation. The basic computation scheme is summarized in Fig. 2. As evident from this figure, for each interface, the Molecular Dynamics (MD) simulations were performed for various combinations of temperature, shearing velocity and the upper layer thicknesses to obtain the strain energy ΔE, measured with respect to an equilibrated configuration, at a predefined failure point and the corresponding value of strain γ. The data set obtained from it is fed to an Evolutionary Neural Network (EvoNN) module to construct meta-models for both ΔE and γ. The details of EvoNN will be provided later. The optimization tasks for these two quantities are constructed using a Predator–prey

* Corresponding author.
E-mail address: nchakrab@metal.iitkgp.ernet.in (N. Chakraborti).

0927-0256/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.commatsci.2011.03.034

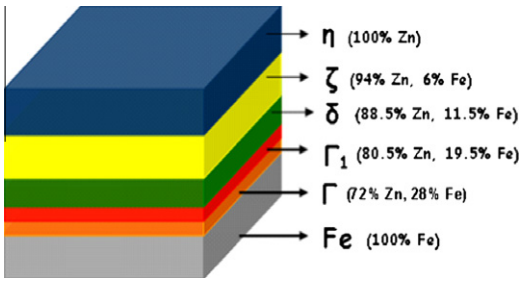

Fig. 1. Schematics of phases present in zinc coated iron [2,3].

Table 1
Characteristics of Fe–Zn intermetallic phases.

	η phase	ζ phase	δ phase	Γ_1 phase	Γ phase
Stoichiometry	Zn	$FeZn_{13}$	$FeZn_{10}$	Fe_5Zn_{21}	Fe_3Zn_{10}
Crystal structure	HCP	Monoclinic	Hexagonal	FCC	BCC
Symmetry group	Pc_3mmc	C2/m	$P6_3me$	F43m	I43m
Iron wt.%	0	5–6	7–11.5	17–19.5	23.5–28
Lattice parameters (Å)	2.612	a = 10.862 b = 7.608 c = 5.061 β = 100.53°	a = 12.83 c = 57.72	a = 17.98	a = 8.98
Atoms/unit cell	6	28	555	408	52

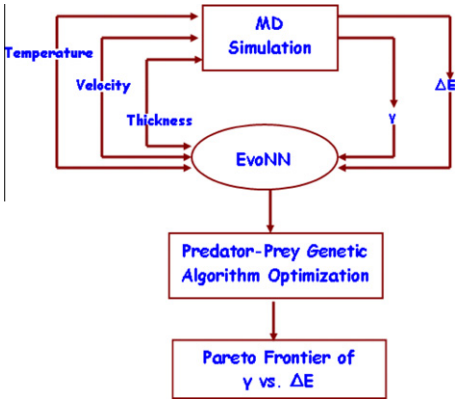

Fig. 2. Schematics of modeling and computation.

type Genetic Algorithms [8,9] described later. It should also be noted that the EvoNN module also used a Predator–prey type Genetic Algorithm for its own evolution from an initial set of network configurations, which again will be elaborated later. Before providing the details of the Predator–prey Genetic Algorithms, it is essential to elaborate the basic concepts of multi-objective optimization and Pareto optimality [10].

3. The multi-objective optimization task

Here for the every pair of contiguous phases we have attempted to simultaneously optimize both ΔE and γ at a predefined failure point. In order to design an interface that would fail with a minimum amount of deformation and that too after a maximum amount energy absorption starting from the equilibration point, ΔE was maximized and γ was minimized. This would be the ideal mechanical property combination required for designing a coated product. Since these two are conflicting requirements, the problem

Table 2
Morse potential parameters.

	D_0 (*eV*)	α ($Å^{-1}$)	r_0 (Å)
Fe–Fe	0.4216	1.3765	2.849
Zn–Zn	0.091552277	2.17861958	2.65552272
Fe–Zn	0.196464857	1.731724531	2.750560712

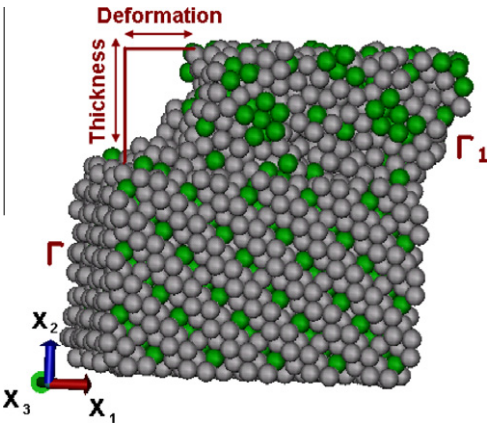

Fig. 3. Loading scheme of MD simulation block shown for Γ–Γ_1 system.

essentially works out to be a multi-objective problem, where none of the objectives would attain their individual best and a set of best possible tradeoffs between them, the so called Pareto frontier [10], contains the optimum. No feasible solution could dominate a member of the Pareto set and *weak dominance* [10] measure was invoked to implement that. Some pertinent mathematical details are briefly provided below considering a minimization type of problem, and the problem in hand could be easily transformed into one by simultaneously minimizing γ and the negative of ΔE.

In such a situation, for the task of simultaneous minimization of l objective functions of the type $f_i(\vec{x}),\ i \in I$, a solution vector $\tilde{x} \in X$, where X denotes the feasible search space, is considered to be Pareto optimal if for any other feasible solution $\vec{x} \in X$ either

$$\bigwedge_{i \in I}(f_i(\tilde{x}) = f_i(\vec{x})) \tag{1}$$

or there exists at least one $i \in I$ such that

$$f_i(\tilde{x}) < f_i(\vec{x}) \tag{2}$$

It is rather obvious that there is every likelihood of having more than one solution satisfying the Pareto optimality condition, and together they constitute the Pareto frontier.

In this study the bi-objective optimization of ΔE and γ was carried out using a Predator–prey Genetic Algorithm [8,9] used earlier in a number of related works [11–15].

4. Predator–prey Genetic Algorithms

This algorithm was used in several of our earlier studies [9,11,12,14,15]. It introduces two distinct entities, the prey and the predators, in a two dimensional computing lattice, emulating a forest. The *preys* denote a set of possible solutions and their initial members are randomly generated as an initial *population* of any Genetic Algorithms. The *predators* are externally induced artificial entity, and their task is to prune the prey population based upon a *function* Φ, related to the objective function values. For any prey i that is examined by a predator j for possible annihilation is given as a weighted sum of the two objectives $F_{1,i}$ and $F_{2,I}$, such that

$$\Phi_{ij} = \omega_j F_{1,i} + (1 - \omega_j) F_{2,i}; \quad 0 \leqslant \omega_j \leqslant 1 \tag{3}$$

where the weight value, ω_j, is uniquely generated for each predator through a uniform random number. Initially, the preys and the predators are randomly introduced in the computational lattice, and each node in the lattice, if not empty, would accommodate either a prey or a predator. The adjacent nodes in the lattice constitute the neighborhood for the predator or a prey, and standard configurations like Moore's or Von Neumann neighborhoods [16] are generally used. A predator is allowed to hunt only in its neighborhood, moves one step at a time, and for that matter, allowed only to kill the weakest prey determined on the basis of Eq. (3). Once it makes a kill, it takes the position of the annihilated prey and hunts again in its changed neighborhood. For each predator such hunts continue up to a prescribed maximum number of times that is dynamically adjusted at the end of each generation, on the basis of a target prey population. In case of a neighborhood devoid of preys, the predator is allowed to make one random move, which gets deducted from its permissible number of hunts. When all the predators exhaust their hunting quota, the members of the surviving prey population are allowed to move randomly one step at a time and the maximum number of such steps is predetermined. During such movements if it encounters an occupied position in the lattice, it loses one chance. Next the preys are allowed to perform *crossover* in their new neighborhood and *mutate*, as in any Genetic Algorithms. Each prey chooses a random partner in its own neighborhood for crossover. If no other prey resides in its neighborhood, the crossover process is aborted for the lone prey.

Two *children* are produced after crossover and mutation are randomly placed in any unoccupied location in the lattice. To find an empty lattice position only a finite number of attempts are allowed, beyond which any unplaced child would be culled. This *migration* process leads to better mixing of the available gene pool – a prerequisite for maintaining excellent population diversity. The predators do not emulate any of these biological processes and thus their number remains fixed. The surviving prey population, after prescribed generations of predator activity, is ranked following the procedure of Fonsceca [17] and the best among them, the so called rank 1 members, approximate the Pareto frontier. In this procedure the rank of any individual, n, is given by

$$\Re_n = 1 + \Theta_n \tag{4}$$

where Θ_n denotes the number of individuals dominating n. The domination criteria are as discussed in the previous section.

We will now discuss the MD procedures for calculating energy and strain, followed by the strategy of constructing meta-models for them using the Evolutionary Neural Network.

Table 3
Ranges of input parameters for MD simulations.

Interface	Velocity, m s^{-1}	Temperature, K	Thickness (top layer) – x_2, Å	Total no. of atoms	Total no. of simulations
Fe–Γ	1–100	1–400	8.98–35.92	3873–5835	288
Γ–Γ_1	1–100	1–400	17.98–35.96	3430–4298	264
Γ_1–δ	1–100	1–400	22.141–33.211	18,314–19,882	264
δ–ζ	1–100	1–400	15.216–38.04	6080–8120	384
ζ–η	1–100	1–400	18.284–33.96	5226–6117	336

Table 4
Time steps for MD simulations.

MD simulation	
NVE (for equilibrating)	Time step: 0.5×10^{-15} s
Equilibration runs	30,000
NPT (for shear)	Time step: 1.0×10^{-15} s

5. Calculation of energy and strain

In this study the *Total Energy* (E_T) of the system of N atoms is computed from the onset of the shearing process at the top atomic layer of an equilibrated Fe–Zn phase assembly, and continued until its 'failure', defined conveniently as the relative interlayer displacement exceeding 2.9 Å, which is equivalent to the lattice spacing of Zn–Zn. The *Strain* (γ) measurement also pertains to that failure point. Both the parameters were computed in a Molecular Dynamics (MD) environment for which the Morse potentials were used for both Fe–Fe and Zn–Zn interactions. In its general form Morse Potential [18] is expressed as:

Table 5
Parameters used in training of the Evolutionary Neural Networks.

EvoNN parameters (training)	Value
No. of hidden nodes	5
Probability of node crossover	0.95
Probability of mutation	0.7
Prey population	300
Number prey preferred	290
Predator population size	20
Number of generations	200
Probability of prey movement	0.5
Lattice dimension (no_x * no_y)	40 * 40

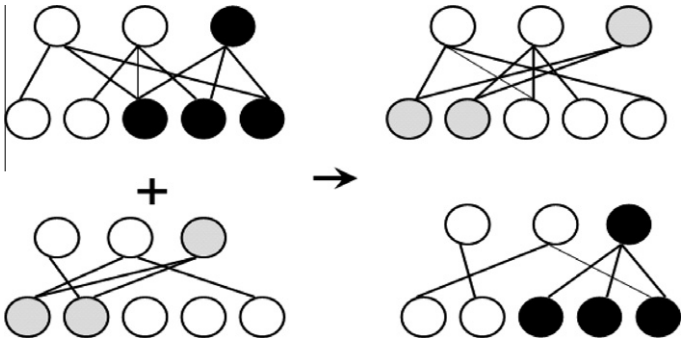

Fig. 4. Schematics of crossover process between the nodes.

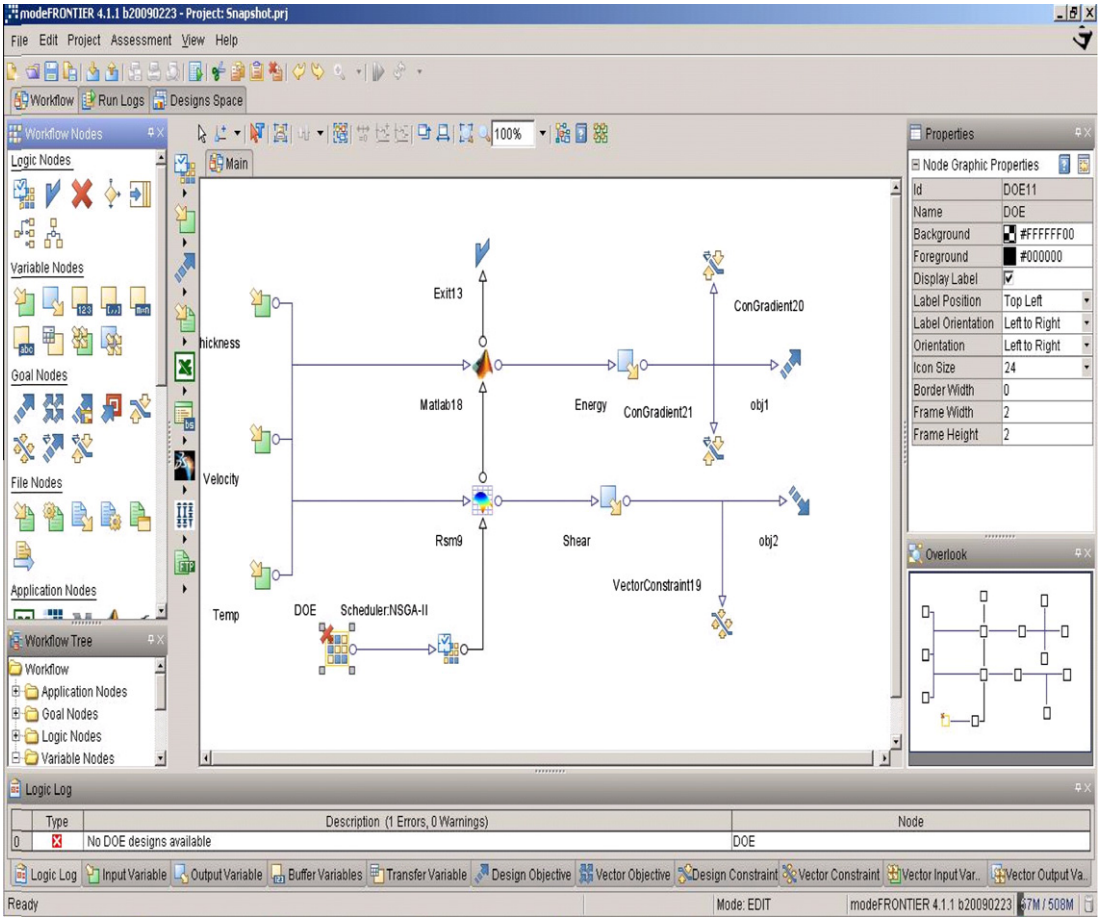

Fig. 5. A typical user interface of modeFRONTIER™.

$$E = D_0\left[e^{-2\alpha(r-r_0)} - 2e^{-\alpha(r-r_0)}\right]_{r<r_c} \tag{5}$$

where E denotes interaction energy between the two atoms (eV), r is the distance between the two atoms (Å), r_0 represents the equilibrium distance between the two atoms (Å), r_c is the cutoff distance (4.5 Å),

While α and D_0 are constants with respective dimensions of reciprocal distance ($Å^{-1}$) *and* energy (eV).

The Fe–Zn potential that was used earlier for the pure iron and zinc assembly studied earlier [1] could not be used satisfactorily for the complex Fe–Zn phases studied here. Morse potential was used once again for the Fe–Zn system, by adjusting its parameters through the combination rules [19] such that:

$$D_{0(\mathrm{Fe-Zn})} = \sqrt{D_{0(\mathrm{Fe})} * D_{0(\mathrm{Zn})}} \tag{6}$$

$$\alpha_{(\mathrm{Fe-Zn})} = \sqrt{\alpha_{(\mathrm{Fe})} * \alpha_{(\mathrm{Zn})}} \tag{7}$$

$$R_{0(\mathrm{Fe-Zn})} = \sqrt{R_{0(\mathrm{Fe})} * R_{0(\mathrm{Zn})}} \tag{8}$$

The numerical values of Morse potential parameters are presented in Table 2.

The ground state structures of the atoms adjacent to both sides of the interfaces were simulated using the CrystalMaker™ software [20]. Like our previous work [1] the shearing phenomena were simulated using the Molecular Dynamics code LAMMPS [21]. It was implemented by equilibrating the concerned interface and its adjacent regions first as an NVE ensemble and subsequently initiating the shearing process, treating the system as an NPT ensemble. The entire system, shown typically in Fig. 3 for Γ–Γ_1 system, consisted of two boxes – a lower box (Γ in this case) whose two lowest atomic layers were held fixed in all directions, and an upper box (Γ_1 in this case) whose two topmost layers were moved at constant velocity in x_1 direction to simulate shear loading. The system was assumed to be under plain strain condition in x_3 direction and hence periodic boundary condition was employed in the x_3 direction.

The size of the upper box was kept smaller in both x_1 and x_3 directions than the lower box. This was necessary because at the onset of shear failure, part or the entire upper box would slip relative to the lower box, and the hanging layer would behave unrealistically if the two box dimensions were exactly the same. Therefore, the other two boundaries (i.e., x_1 and x_2 planes) were kept under isolated boundary condition.

Of the three displacements u_1, u_2 and u_3 (along the respective axial directions), u_3 is identically zero due to the plane strain assumption. Since we are loading the block beyond shear failure, we allow for large strain magnitudes and use finite strain expressions, which in the 1–2 direction are:

$$E_{12} = \frac{1}{2}\left(\frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1}\right) + \frac{1}{2}\left(\frac{\partial u_1}{\partial x_1}\frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1}\frac{\partial u_2}{\partial x_2} + \frac{\partial u_3}{\partial x_1}\frac{\partial u_3}{\partial x_2}\right) \tag{9}$$

Because of the plane strain assumption it reduces to:

$$E_{12} = \frac{1}{2}\left(\frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1}\right) + \frac{1}{2}\left(\frac{\partial u_1}{\partial x_1}\frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1}\frac{\partial u_2}{\partial x_2}\right) \tag{10}$$

However, as will be shown subsequently, the displacement gradients $\partial u_2/\partial x_1$ and $\partial u_1/\partial x_1$ are found to be negligibly small in our simulations, and consequently, we use the following standard approximation for shear strain:

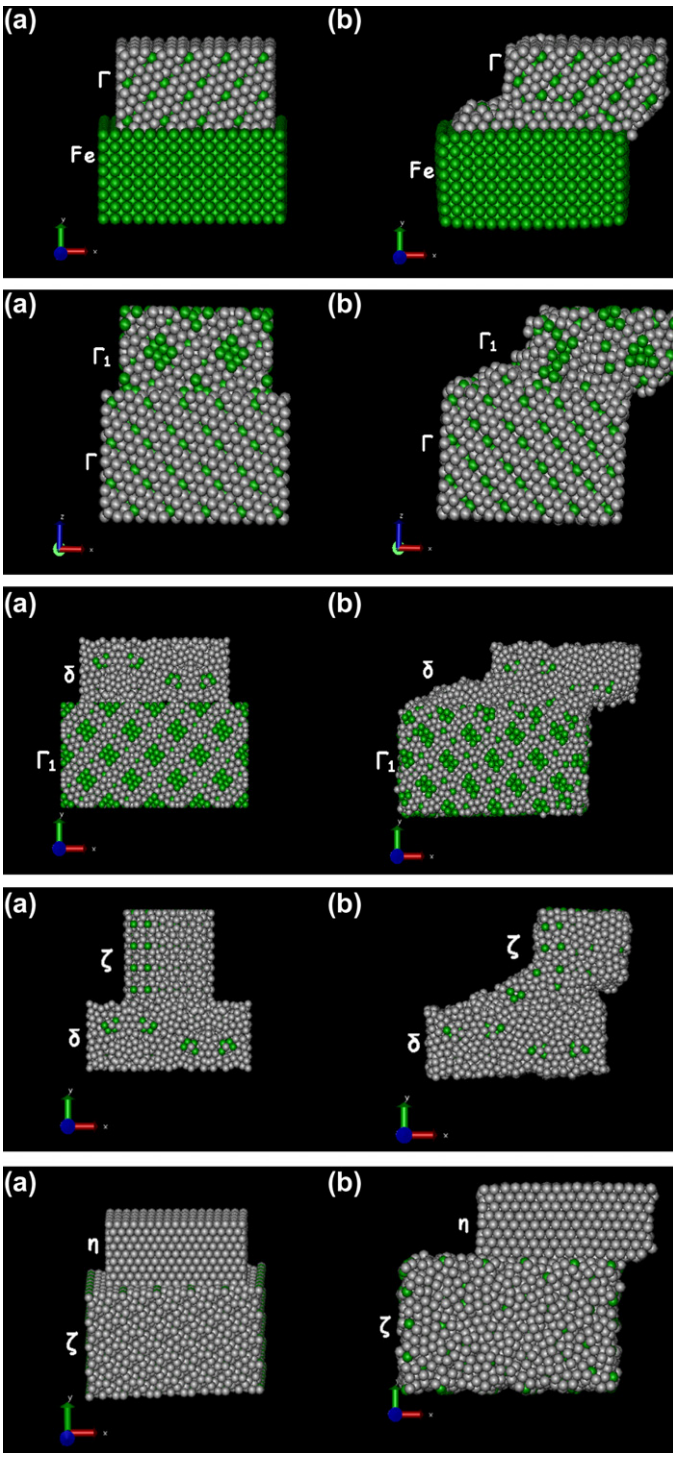

Fig. 6. The equilibrated (a) and failed (b) structures for various interfaces.

$$\gamma_{12} = 2E_{12} = \frac{\partial u_1}{\partial x_2} \tag{11}$$

which can be calculated simply as the ratio of the x_1-displacement and the phase thickness as elaborated in Fig. 3 for Γ–Γ_1 interface.

Shear failure is defined in our study as the slip of any layer at or above the interface in the simulation block by at least one lattice position. The equilibrium interatomic distance of Zn–Zn varied between 2.6 Å and 2.9 Å in the phases considered in this study. Thus, shear failure was identified as the relative displacement between any two adjacent layers above the interface exceeding 2.9 Å.

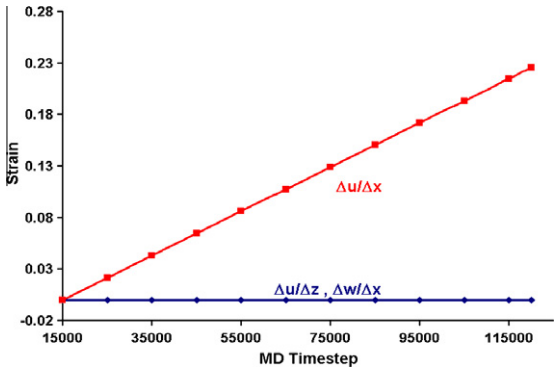

Fig. 7. Time history of displacement gradients for Γ–Γ_1 interface.

6. Meta-modeling with Evolutionary Neural Network

Although the values of ΔE and γ are both obtained using Molecular Dynamics, their recurrent calculation for the multi-objective optimization task would immensely increase the computing burden. To circumvent the problem, like in our previous work [1], a meta-modeling task was undertaken through an Evolutionary Neural Network [9,11] using the outputs of judiciously designed numerical experiments conducted through Molecular Dynamics. The input parameters that varied were: (i) the velocity at which the top layer is set to motion, (ii) thickness of the layers above interface, and (iii) temperature of simulation. The ranges of input parameter variation are provided in Table 3. The MD time steps are listed in Table 4. The Evolutionary Neural Networks (EvoNN) that were utilized to construct the meta-models using MD outputs, themselves evolved through a Predator–prey Genetic Algorithm. The idea, as explained in other places [9,11] is to subject a *population* of neural networks to a bi-objective optimizing process to work out a Pareto frontier between the complexity of network and its training error, both of which had been attempted to be minimized. In the population of the neural networks the topology and the magnitude of weights in the lower part of the network varied between the individuals and were subjected to crossover as shown in Fig. 4. The mutation scheme is elaborated in [9] and is not repeated here.

The upper portion of the network evolved through a linear least square procedure [22] and was not subjected to genetic evolution. For the linear problem that is solved in the upper part of the network, an evolutionary paradigm might not have any special advantage. However, genetically evolved near optimal inputs from the lower part of the network can immensely improve the performance of any gradient based solver employed at the upper part, and the Evolutionary Neural Net takes advantage of that. Since the linear least square procedure has a firm mathematical proof of convergence, the convergence of this hybrid procedure is also assured. The complexity of the network was measured through the total number of weights in the lower part of the network excluding the biases. Out of the networks presents in the Pareto frontier, a suitable one was selected through the corrected Akaike Information Criteria, like in a previous study [23], such that

$$AICc = AIC + \frac{2k(k+1)}{n-k-1} \tag{12}$$

where *AIC*, the Akaike Criterion is expressed as

$$AIC = 2k - n\ln(RSS/n) \tag{13}$$

Here the total number of connections in both upper and lower parts of the network including the biases determines k, n denotes the number of observations used and *RSS* is the residual sum of squares for the model.

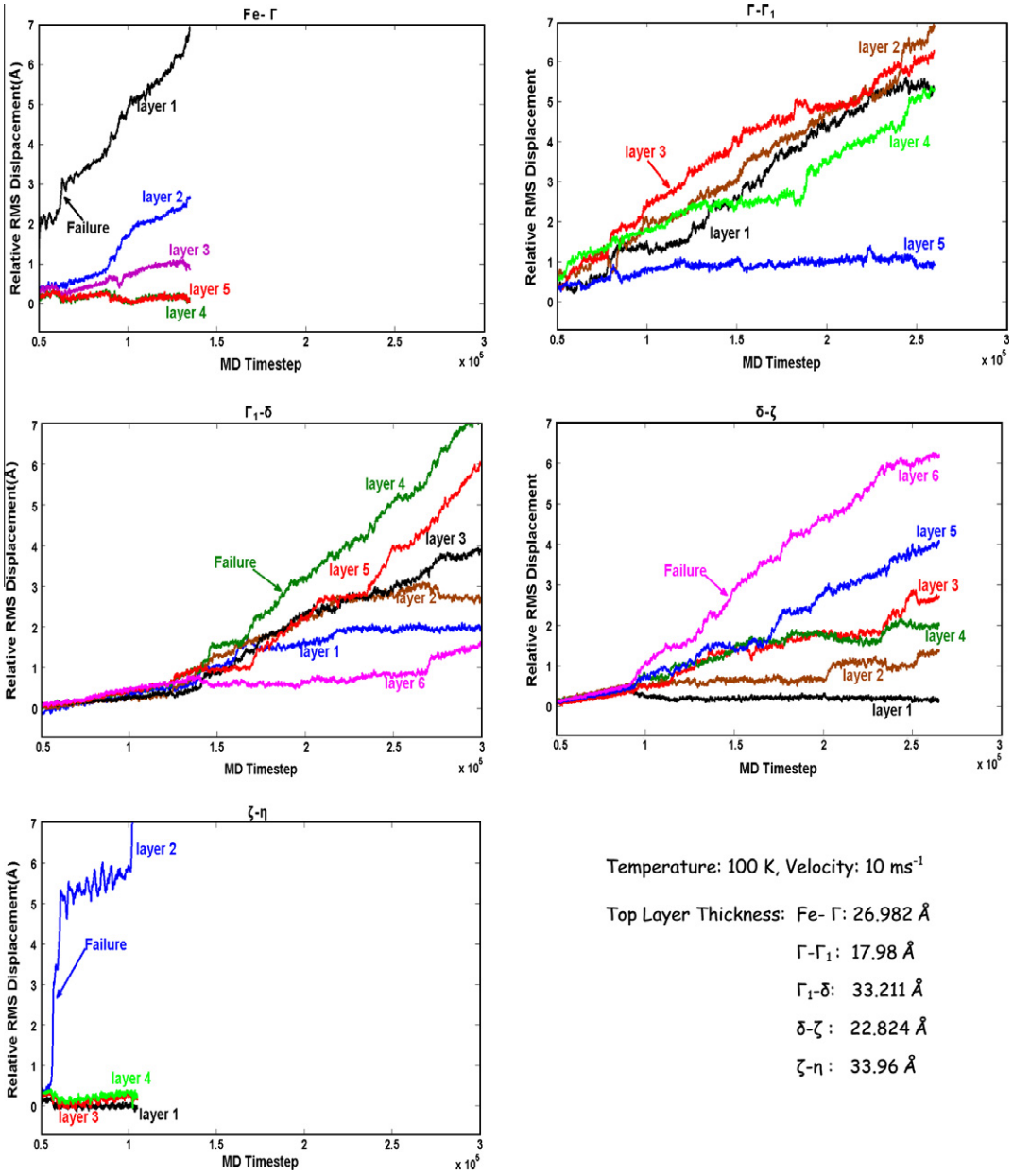

Fig. 8. Relative RMS displacement for various interfaces.

The GA parameters used in constructing the Evolutionary Neural Networks are provided in Table 5.

7. Modeling and optimization using modeFRONTIER™

During this study the evolutionary approach described above was pitted against the commercially available software modeFRONTIER™ [24]. The software comes with various modules that the user can easily integrate in a GUI interface as shown in Fig. 5. The data driven model could be constructed using the options of its built-in neural network or genetic programming [25] modules and for the optimization task the user gets to choose from a number of Genetic Algorithms, including MOGAII [26] and NSGAII [25].

8. Outcome of the MD simulations

The equilibrated and failed structures of the various interfaces are shown in Fig. 6. Typical time histories of the three displacement gradients are shown in Fig. 7. Clearly, the only significant distortion in the lattice is $\partial u_1/\partial x_2$ as discussed previously.

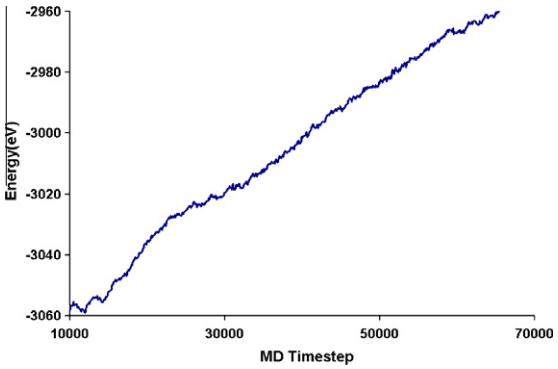

Fig. 9. Energy absorbed as a function of MD time step for Γ–Γe_1 interface.

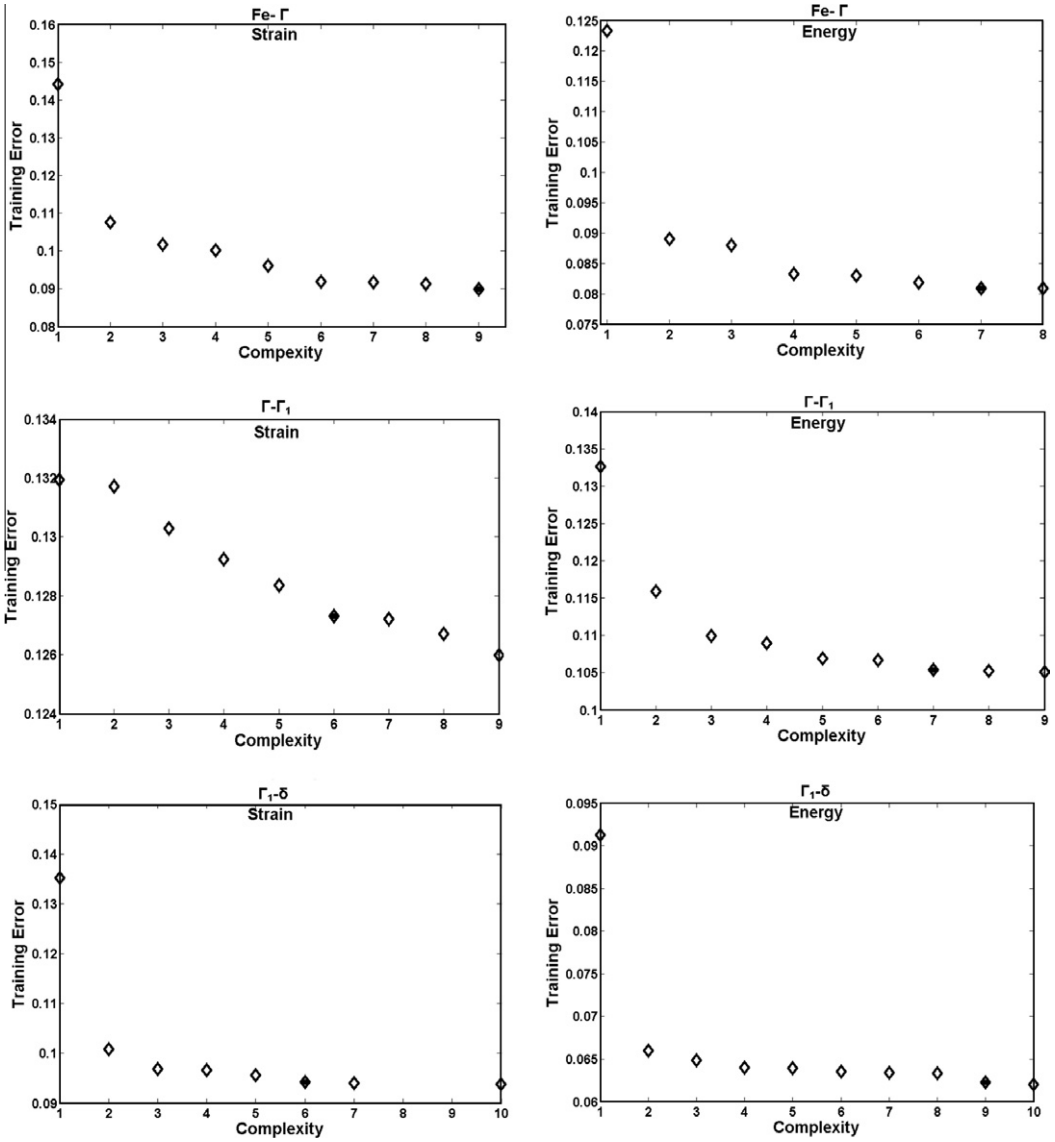

Fig. 10. Optimized networks for strain and energy at various interfaces. Each diamond indicates a separate neural network, and a darkened diamond indicates a network selected through corrected Akaike criteria.

Since LAMMPS provides RMS displacement (δ_{rms}) for any specified group of atoms, the point of shear failure may be identified from the δ_{rms} vs. time plot. It is not essential that the failure would occur exactly at the interface. To ascertain the region where the failure has occurred, we had divided the structure above the interface into different segments in vertical direction and called them layer 1, layer 2, layer 3 and so on. Most simulations ran with a total of 3400–20,000 atoms. After that we calculated the RMS displacement of near surface atoms in these layers and the layer that crossed the relative RMS displacement of 2.9 Å first, we assumed that the failure had occurred there. The computed RMS displacement plots are shown in Fig. 8, the assigned failure points are marked on each figure. Like our previous work [1] the noisy MD output was smoothed using a local linear least square regression routine (LOESS) available in the Curve Fitting Toolbox™ of MATLAB™.

The energy of the system at the failure point, measured with reference to the energy at the equilibration point, provided the shear energy absorbed up to the point of failure, and was taken as a value of the first objective function used in the subsequent task. A time history of the total energy of the system (E_T), as shown for the typical case of Γ–Γ_1 system in Fig. 9, is required to compute this objective function. The second objective, the shear strain at failure was calculated as the ratio of the displacement of the top two layers at failure to the coating thickness, as explained in Eq. (11).

9. Outcome of the objective function training

The Genetic Algorithms based Evolutionary Neural Network [9,11] could train both the objective functions efficiently. The Pareto frontiers between the error of training and the complexity of network are presented in Fig. 10 for all the relevant interfaces. Each diamond in these figures is a unique neural net, denoting one of the optimum tradeoffs between the error and complexity, and in each case a specific network, indicated as darkened diamond, was picked up using the corrected Akaike critera explained in [11]. The neural network module in modeFRONTIER™ was also utilized to come up with the same and their performances are compared for energy in Fig. 11 and in Fig. 12 for strain. It seems that modeFRONTIER™ in many cases has attempted to overfit the data a bit. Even a careful visual observation would confirm that, since the fitted curve tends to capture nearly every fluctuation in the ori-

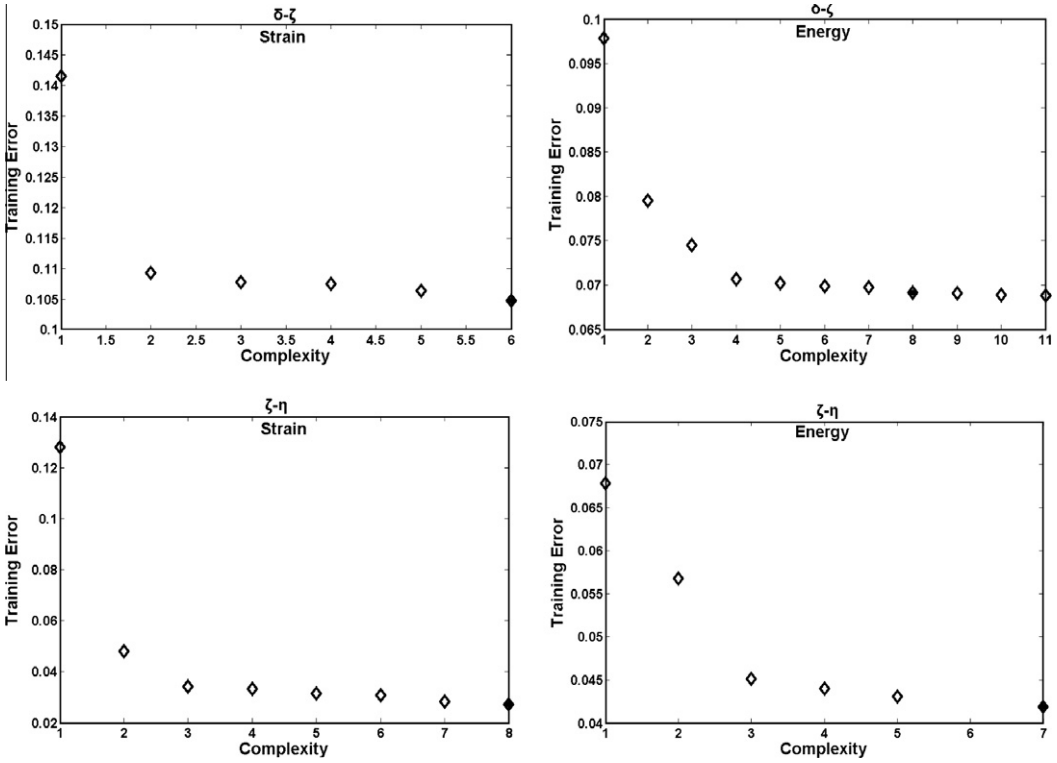

Fig. 10 (*continued*)

ginal data set. This however has a colossal effect on the optimization process, which we will discuss shortly.

10. The computed Pareto frontiers

The objective functions calculated by the evolutionary networks were optimized using the same Predator–prey Genetic Algorithm [8,9] that was used to generate them. The GA parameters used for this are listed in Table 6. For the modeFRONTIER™ objectives, constructed using its neural network module, optimization was carried out using the well known strategies of NSGAII [25] and MOGAII [26], provided also as modules in the software. The computed Pareto frontiers are presented in Fig. 13. The evolutionary neural net could compute the Pareto frontiers in all the cases. The modeFRONTIER™ on the other hand, could accurately compute the frontier accurately only in case of ζ–η interface. For the rest, it had just generated the solutions at the prescribed limiting values. Considering the proven track record of the two optimization strategies used, it seems that the problem lies with the over fitting at the neural net module of modeFRONTIER™ and similar problems were encountered in some of our earlier studies as well [14,15,23]. The best possible combinations of ΔE and γ at failure, expectedly would be found along the Pareto contours shown in Fig. 13. Since there are marked differences between the Pareto contours for different phase pairs, some galvanized steel may very well preferentially fail along a particular phase boundary depending upon the optimum combination of the absorbed energy and the corresponding strain. These frontiers however represent the functional space between ΔE and γ and each point on the frontier corresponds to a unique set of the decision variables: (i) the temperature of simulation, (ii) thickness of the interfacial layer, and (iii) the velocity at which

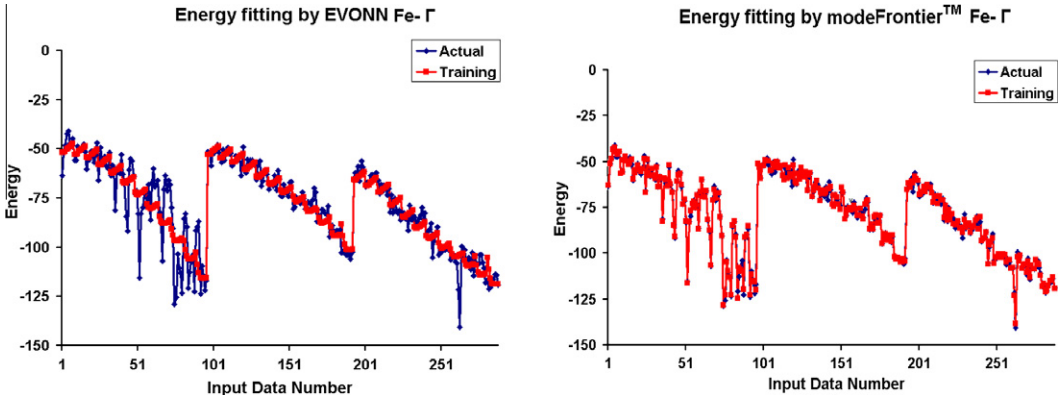

Fig. 11. Comparison of network training by Evolutionary Neural Net (EVONN) and modeFRONTIER™ for energy. Numbers along the abscissa are the identifiers for the input data number and each data point is generated by a distinct MD simulation with its unique set of parameters.

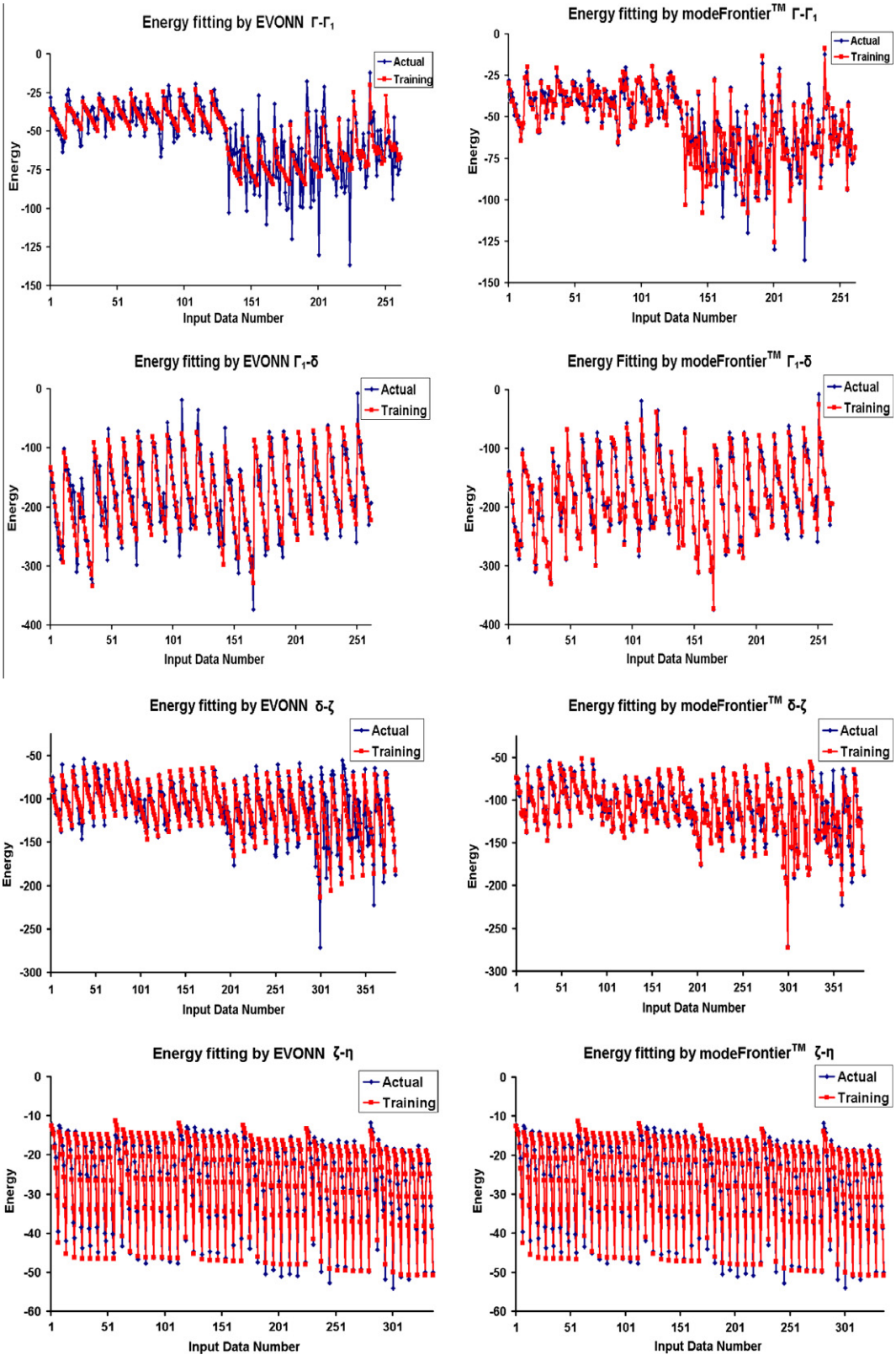

Fig. 11 (*continued*)

the shear has been initiated. Different combinations of these decision variables would lead to Pareto optimality as shown in Table 7 for the regions of Pareto frontiers marked as a, b and c in Fig. 13. The specific role played by each of them to influence the objective functions needs to be analyzed further, for which some further analyses have been performed, as described below.

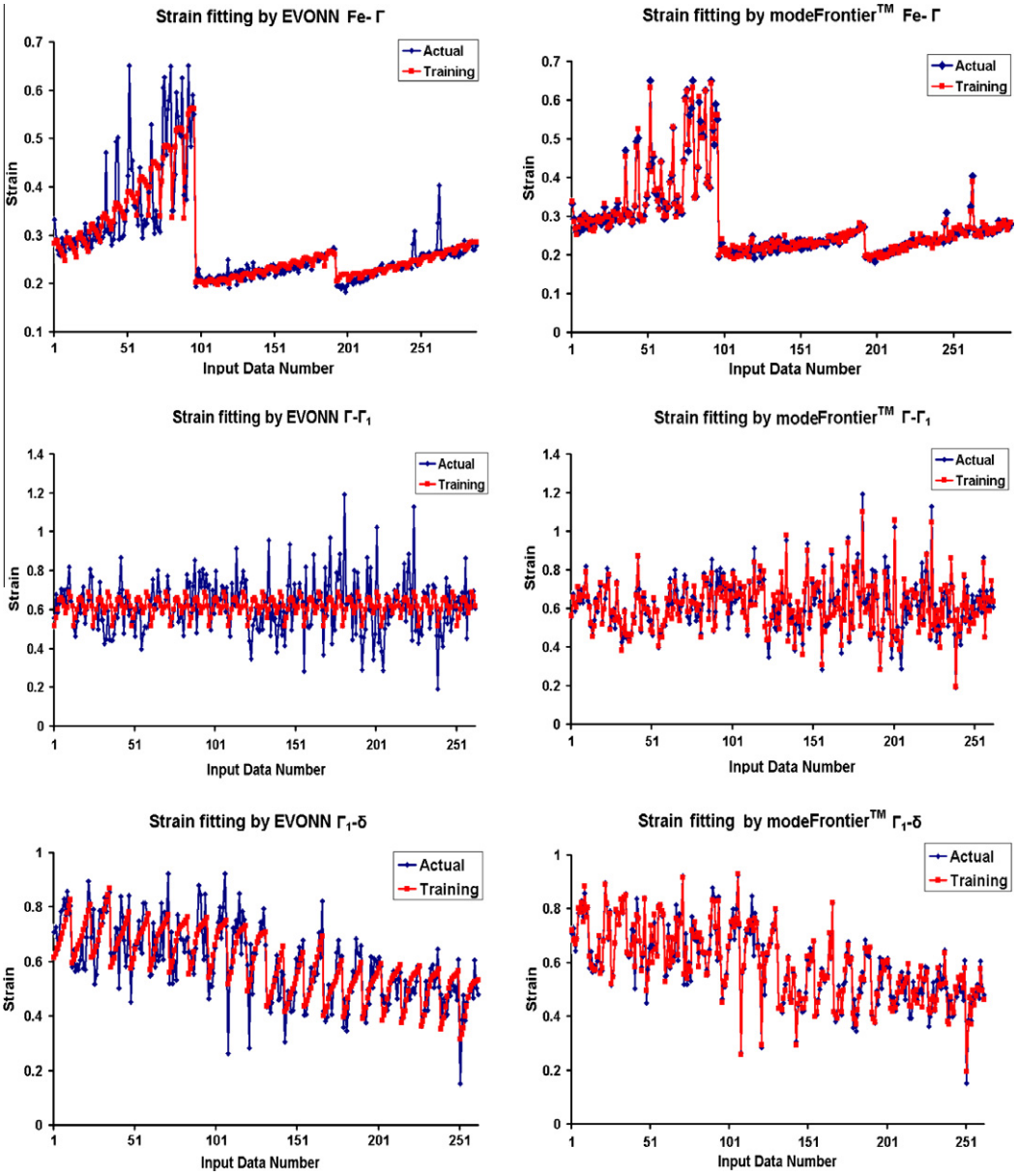

Fig. 12. Comparison of network training by Evolutionary Neural Net (EVONN) and modeFRONTIER™ for strain. Numbers along the abscissa are the identifiers for the input data number and each data point is generated by a distinct MD simulation with its unique set of parameters.

11. The influence of individual variables

In a complex problem like the one in hand the role played by the individual variables is seldom easy to ascertain. The variables often tend to influence each other and isolating their individual impact on the objective function usually remains a difficult task. The method that we have adopted in this study to accomplish this has evolved through a series of our previous work [14,15,27] and the basic strategy could be summarized as follows:

- In any model (e.g. the model of either ΔE or γ at failure in this case) every variable input is held at the base level, except for one.
- The pertinent variable is arbitrarily perturbed both below and above the base level following some definite patterns like sudden changes, gradual changes, holding above or below the base level etc. and the model output is recorded.
- If the trend of the model predictions follows exactly the nature of the perturbation given to the input variable, their interdependence is considered to be *direct*. Alternately, if an increase in the input variable causes a decrease in the output space and vice versa, their mutual dependence is considered to be *inverse*. In some cases the responses could also be *mixed* and the analysis might point to *no dependence* as well.

Following this strategy the variable responses for both the objectives were analyzed for all the phases. Some typical results presented in Fig. 14 for the Fe–Γ interface. The input data numbers along the abscissa refer to the synthetic data points used to generate the desired input profiles (i.e. the thickness profile shown in the panels at the top, and the velocity and temperature profiles in the middle and the bottom panels respectively). Similar analyses were carried out for all the remaining interfaces and the major findings are summarized in Table 8.

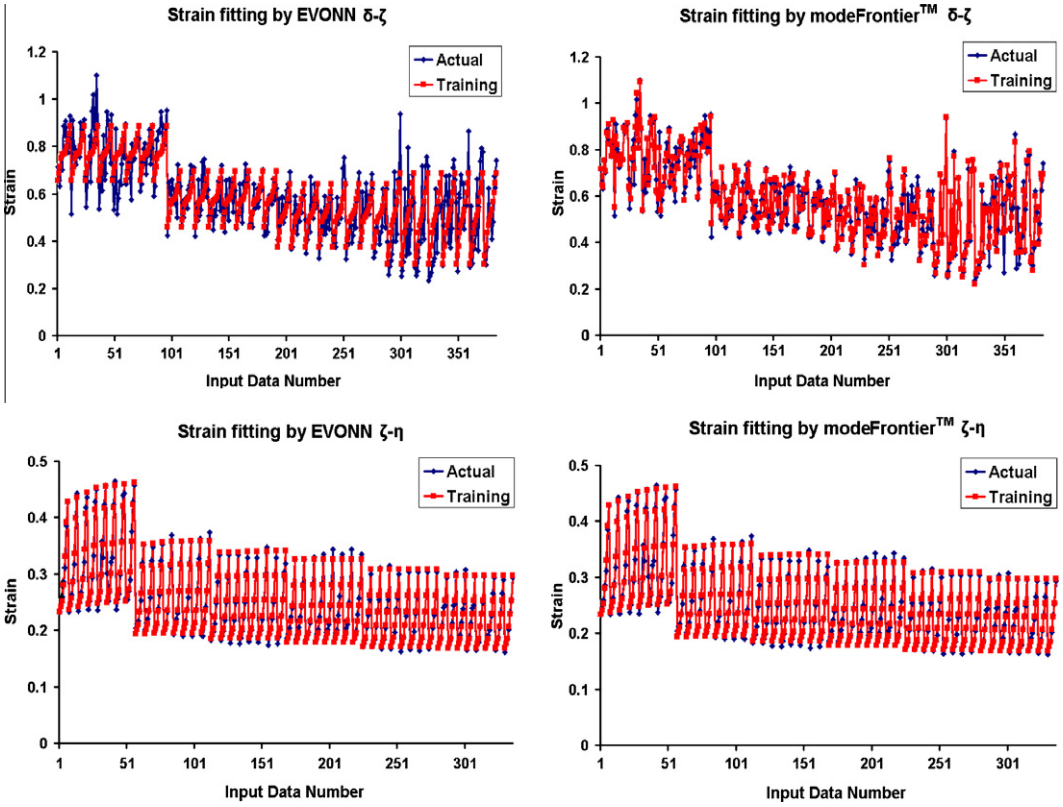

Fig. 12 (*continued*)

Table 6
Parameters used in optimization of energy and strain.

Optimization	
Prey population	180
Number prey preferred	300
Predator population size	20
Number of generations	200
Probability of prey movement	0.5
Probability of mutation	0.16
Lattice dimension (no_x * no_y)	100 * 100

A careful observation of Table 8 would reveal some interesting trends. Expectedly, being the primary source of energy input, velocity is directly correlated with the energy absorbed in all cases. The shear strain also followed the same trend, except for the Γ–Γ_1 case which has shown bit of a mixed response, some of which may not actually be significant. With an increase in temperature the atomic mobility increases and shear failure occurs at a lower energy input. This trend is corroborated in all cases except for ζ–η, a fairly complex assembly where even after the prescribed amount RMS displacement of 2.9 Å, the extent of bond breaking seems to be rather small, as shown in Fig. 6. Temperature however has affected the strain responses in a mixed way in some cases, and often the model could not pick up any dependence. It is likely that in these complex phase assemblies certain segments deformed differently than the others and it would be a formidable task to distinguish that from some of the noise already present in the Molecular Dynamics data. With increasing thickness the expected direct proportionality to energy absorbed is observed in three cases out of total five. The remaining two showed bit of a mixed response, a factor contributed by the unit cell height and the regions exposed to shear at different thickness values. Three of the configurations Γ_1–δ, δ–ζ and ζ–η have shown the expected inverse trend of strain with the thickness variation. The mixed trend and no response shown by the remaining two again point towards the complexities of these phase assemblies and their representative unit cells.

12. Identifying the strong and weak interfaces

To meaningfully augment our understanding of this system, it is imperative that we should try to identify the interface that offers the best tradeoff between the two conflicting objectives: maximum energy absorption and minimum deformation at the onset of failure. It is equally important to identify the one that performs worst in terms of those criteria. To achieve this, the Pareto solutions shown in Fig. 13 for all the phases are mixed together and ranked once again amongst themselves using Eq. (4). The idea behind this second round of ranking is quite simple. The Pareto solutions found separately for various interfaces are now pitted against each other for a global tradeoff between ΔE and γ at the failure point. The interfaces that are now able to provide a significant number of rank 1 solutions are expected to be more stable under the shear force compared to the others that they dominate. Alternately, if some interface is unable to produce any rank one solution in this situation is expected to be weaker compared to any interface that does. The results of this analysis are summarized for ranks 1–6 in Fig. 15, constructing a series of frequency plots, which provide some interesting revelations through these global rankings. It appears that all the Pareto solutions belonging to the ζ–η interface belong to the global rank 1 shown in Fig. 15; while none of the Fe–Γ Pareto solutions appear there, not even up to rank 6. The best tradeoff therefore is offered by the ζ–η interface, while Fe–Γ performs worst. Additionally, a close examination of Fig. 15

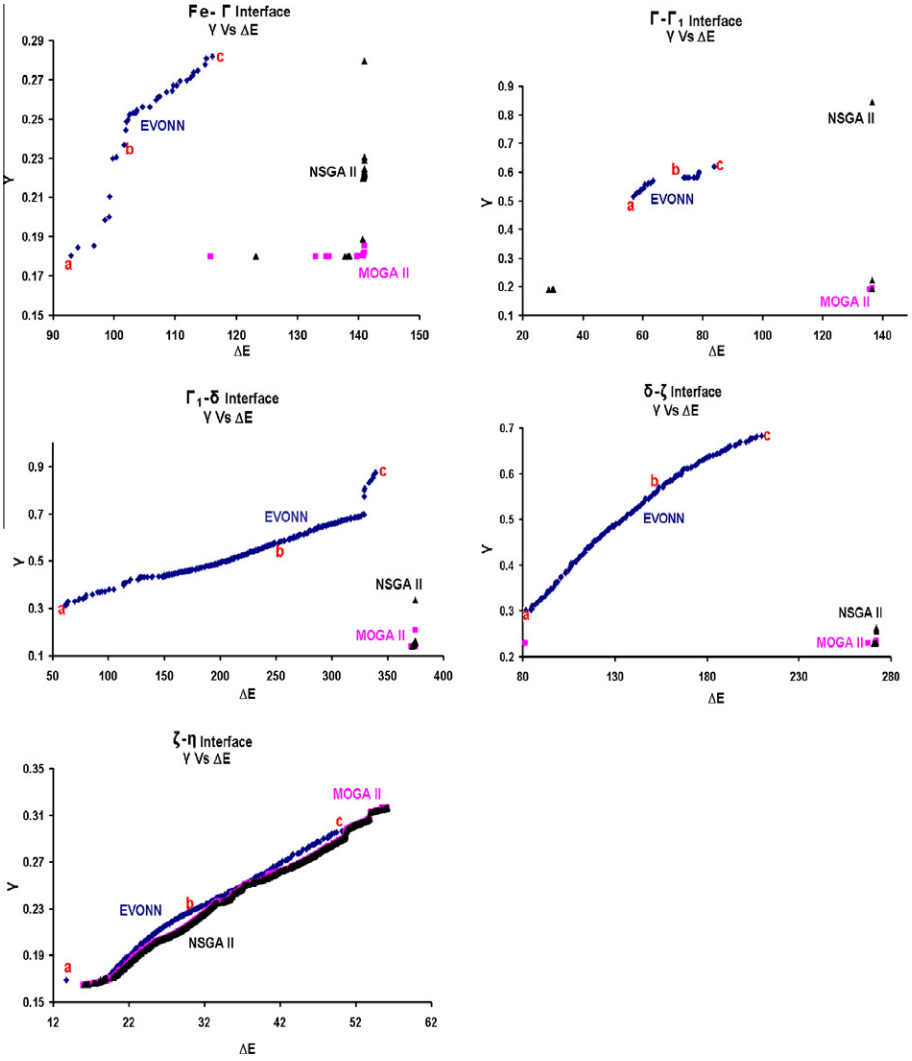

Fig. 13. Pareto-frontier between energy and strain at the onset of failure for various interfaces.

reveals that the δ–ζ interface also performs reasonably well, since it is present in the global rank 1 and is adequately represented in rank 2 and beyond. Now referring back to Fig. 1, we realize the significance of this. It is the η phase that faces the environment and its excellent stability with its adjacent ζ phase, in terms of the two objectives considered here, provides the Zn coated steel its basic resistance to environmental degradation, for which it is conventionally used. The reasonably good stability of the next phase pair δ–ζ further reinforces it. Since the phases are of different stiffness, any high shearing effects encountered in the top layers need not be experienced at the same level by the phases down below. Equally interesting is the fact that Fe–Γ, the weakest link, is deep inside the

Table 7
Decision variables in the regions a, b and c of the Pareto frontiers shown in Fig. 13.

	Thickness, Å	Velocity, m s^{-1}	Temperature, K	ΔE (eV)	Strain
Fe–Γ (a)	22.18051	99.92152	19	92.9775	0.1804
Fe–Γ (b)	19.5043	99.38773	33	101.6274	0.2366
Fe–Γ (c)	35.56571	96.51162	117	116.0908	0.2819
Γ–Γ_1 (a)	35.58559	63.24879	122	73.61414	0.57939
Γ–Γ_1 (b)	33.12147	1.001511	32	56.9149	0.513146
Γ–Γ_1 (c)	35.00034	99.72182	74	83.80537	0.617963
Γ_1–δ (a)	32.9853	1.0015	232	61.9089	0.3133
Γ_1–δ (b)	33.1293	88.8723	148	252.0466	0.578
Γ_1–δ (c)	22.14107	99.52462	63	338.5061	0.8719
δ–ζ (a)	38.0372	1.001511	38	81.5624	0.3013
δ–ζ (b)	37.30854	71.3685	2	153.768	0.5696
δ–ζ (c)	37.83411	99.2559	2	209.7349	0.6832
ζ–η (a)	27.99763	1.001511	1	13.71509	0.168764
ζ–η (b)	27.98705	49.47878	92	30.45331	0.228685
ζ–η (c)	27.93955	99.06698	132	50.17142	0.296272

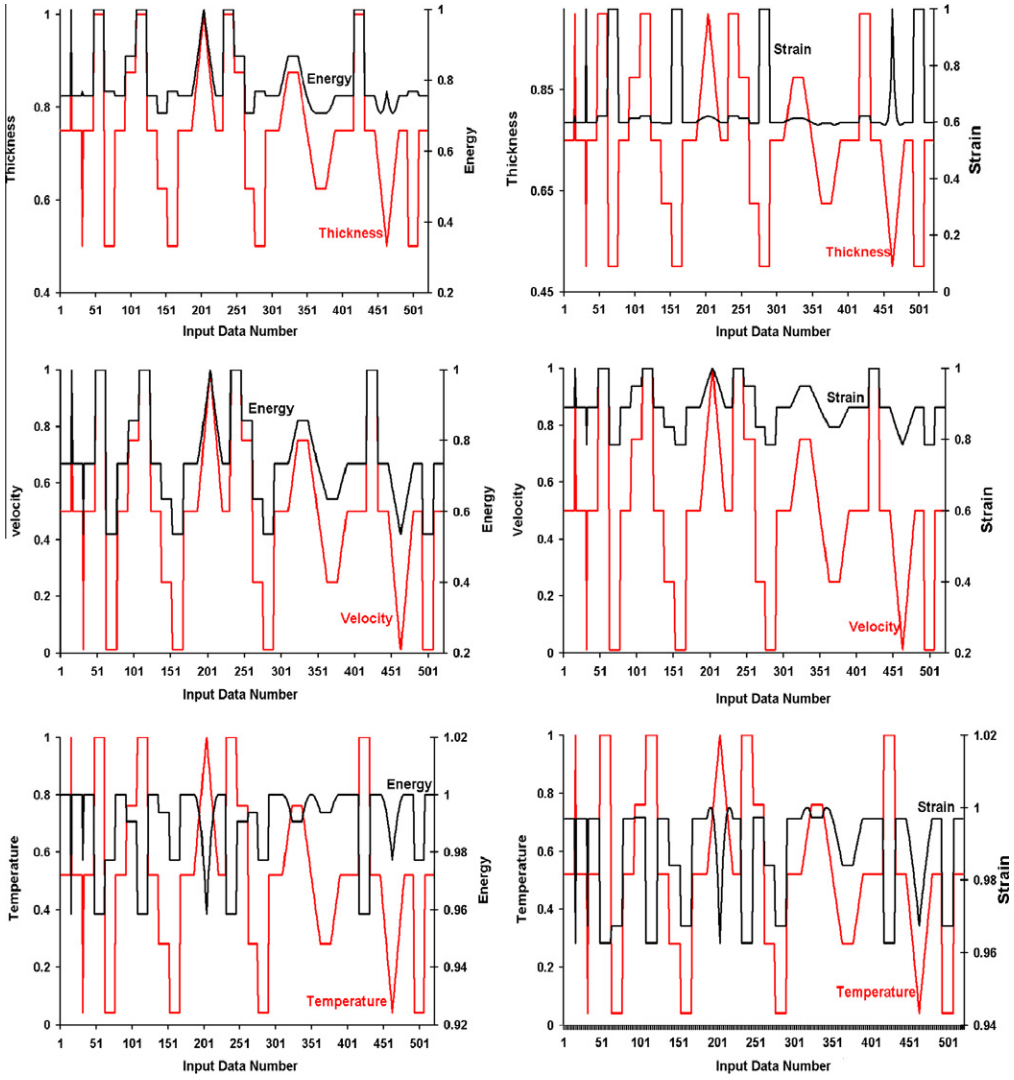

Fig. 14. Responses of arbitrary individual variable perturbations on E_Γ (left) and γ (right) for the Fe–Γ interface. The perturbed variables are indicated on the figures. Numbers along the abscissa are the identifiers for the input data number. Details of input data generation are provided in the text and major findings are summarized in Table 8.

coating and therefore, in most cases, is reasonably protected from any direct exposure to the shearing effects. However, after pronged exposure to the environment, the upper layers might ultimately yield to corrosion related processes, exposing the Γ phase to large shearing strains, which, in turn, would lead to an easy failure at the weak Fe–Γ interface.

At this stage it is highly relevant to point out that there is ample experimental evidence [2] that the in Zn-coated Fe failure occurs

Table 8
Summary of Input variable responses to energy absorbed and shear strain.

Interface	Input variable	Response to energy absorbed	Response to shear strain
Fe–Γ	Thickness	Mixed	Mixed
Fe–Γ	Velocity	Direct	Direct
Fe–Γ	Temperature	Inverse	Mixed
Γ–Γ_1	Thickness	Direct	No response
Γ–Γ_1	Velocity	Direct	Mixed
Γ–Γ_1	Temperature	Inverse	No response
Γ_1–δ	Thickness	Mixed	Inverse
Γ_1–δ	Velocity	Direct	Direct
Γ_1–δ	Temperature	Inverse	Mixed
δ–ζ	Thickness	Direct	Inverse
δ–ζ	Velocity	Direct	Direct
δ–ζ	Temperature	Inverse	No response
ζ–η	Thickness	Direct	Inverse
ζ–η	Velocity	Direct	Direct
ζ–η	Temperature	Direct	Direct

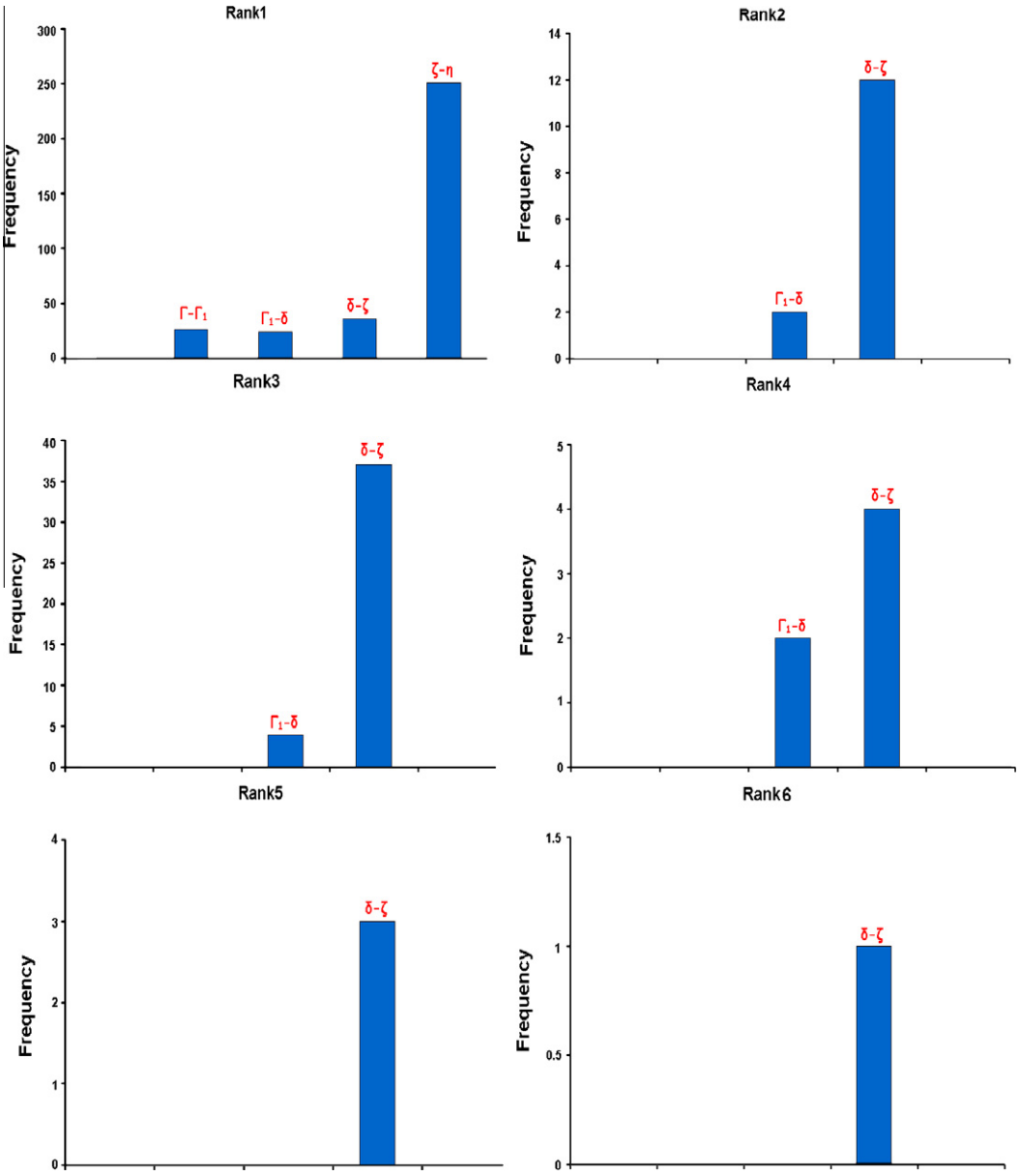

Fig. 15. Frequency plots for the global ranks of Pareto solutions presented in Fig. 13.

along the Γ phase layer/substrate interface and the high ductility of ζ phase, and hence its high energy absorption, is known to protect the coating from failure. The computation work presented here fully supports this. Although the computations were performed here with empirical potentials and with rather limited number of atoms, their success in predicting the correct physical trend could be very well attributed to the evolutionary meta-modeling approach adopted here and the subsequent multi-objective optimization. Relevance of Genetic Algorithms in materials design problem is thus, once again, demonstrated.

13. Concluding remarks

Although there is sufficient literature [28–39] now that deals with Genetic Algorithms applications for atomic assemblies, the attempts to apply the multi-objective Genetic Algorithms for such problems is still somewhat limited [40]. This study shows some obvious advantages of a multi-objective approach. Molecular Dynamics, although known for its rigorousness, could be computationally prohibitive for scaling up the computations for more realistic physical system. In such a scenario, the meta-modeling approach using the Evolutionary Neural Network presented in this study can make a significant difference by combining the ease of calculation with reliability of results and in principle, can augment the earlier strategies based upon the response surface method and the likes [41]. The present approach is currently being applied to determine the influence of dislocations on the system studied, which we expect to report in a subsequent publication.

Acknowledgments

Financial support from TATA Steel is thankfully acknowledged. We would like to thank Arunima Singh and G.R. Dinesh Kumar for doing some preliminary calculations. A number of constructive comments from the anonymous reviewers are also highly appreciated. Those led to some definite improvements in the paper.

References

[1] B. Bhattacharya, G.R.D. Kumar, A. Agarwal, Ş. Erkoç, A. Singh, N. Chakraborti, Comput. Mater. Sci. 46 (2009) 821–827.
[2] A.R. Marder, Prog. Mater. Sci. 45 (2000) 191–271.
[3] www.metalplate.com/galvanizing/TIS4.HTM.
[4] H. Okamoto, J. Phase Equilibr. Diff. 28 (2007) 317–318.
[5] C.H.E. Belin, R.C.H. Belin, J. Solid State Chem. 151 (2000) 85–95.
[6] W.H. Zhu, H.M. Jin, P. Wu, H.L. Liu, Phys. Rev. B 70 (2004) 165419.
[7] A.S. Koster, J.C. Schoone, Acta Crystallogr. B 37 (1981) 1905–1907.
[8] X. Li, Lect. Notes Comput. Sci. 2632 (2003) 207.
[9] F. Pettersson, N. Chakraborti, H. Saxén, Appl. Soft Comput. 7 (2007) 387–397.
[10] C.A.C. Coello, D.A. Van Veldhuizen, G.B. Lamont, Evolutionary Algorithms for Solving Multi-objective Problems, Kluwer Academic Publishers, New York, 2002.
[11] F. Pettersson, A. Biswas, P.K. Sen, H. Saxén, N. Chakraborti, Mater. Manuf. Process. 24 (2009) 320–330.
[12] A. Agarwal, F. Pettersson, A. Singh, C.S. Kong, H. Saxen, K. Rajan, S. Iwata, N. Chakraborti, Mater. Manuf. Process. 24 (2009) 274–281.
[13] F. Pettersson, H. Saxén, K. Deb, Mater. Manuf. Process. 24 (2009) 343–349.
[14] D. Govindan, S. Chakraborty, N. Chakraborti, Steel Res. Int. 81 (2010) 197–203.
[15] A. Biswas, O. Maitre, D.N. Mondal, S.K. Das, P.K. Sen, Collet, N. Chakraborti, Mater. Manuf. Process., in press., doi:10.1080/10426914.2010.544809.
[16] R. Dewri, N. Chakraborti, Model. Simul. Mater. Sci. 13 (2005) 173–183.
[17] C.M. Fonseca, Multiobjective genetic algorithms with applications to control engineering problems. Ph.D. Thesis, Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield, UK, 1995.
[18] Ş Erkoç, Annu. Rev. Comput. Phys. IX (2001) 1–103.
[19] D. Huo, Y. Liang, K. Cheng, J. Mech. Eng. Sci. 221 (2007) 259–266.
[20] www.crystalmaker.com.
[21] http://lammps.sandia.gov.
[22] F. Pettersson, N. Chakraborti, S.B. Singh, Steel Res. Int. 78 (2007) 890–898.
[23] A. Agarwal, U. Tewary, F. Pettersson, S. Das, H. Saxén, Ironmak. Steelmak. 37 (2010) 353–359.
[24] S. Poles, M. Vassileva, D. Sasaki, Lect. Notes Comput. Sci. 5252 (2008) 329–348.
[25] K. Deb, Multi-Objective Optimization by Evolutionary Algorithms, John Wiley & Sons, Chichester, 2001.
[26] S. Poles, P. Geremia, F. Campos, S. Weston, M. Islam, Lect. Notes Comput. Sci. 4403 (2007) 633–644.
[27] M. Helle, F. Pettersson, N. Chakraborti, H. Saxén, Steel Res. Int. 77 (2006) 75–81.
[28] N. Chakraborti, Int. Mater. Rev. 49 (2004) 246–260.
[29] B. Hartke, Struct. Bond. 110 (2004) 33–53.
[30] L.D. Lloyd, R.L. Johnston, S. Salhi, J. Comput. Chem. 26 (2005) 1069–1078.
[31] C.V. Ciobanu, C.Z. Wang, K.M. Ho, Mater. Manuf. Process. 24 (2009) 109–118.
[32] N. Dugan, S. Erkoc, Mater. Manuf. Process. 24 (2009) 250–254.
[33] N. Chakraborti, R. Kumar, J. Phase Equilibr. 24 (2003) 132–139.
[34] S.M. Woodley, Mater. Manuf. Process. 24 (2009) 255–264.
[35] T.E.B. Davies, D.P. Mehta, J.L. Rodriguez-Lopez, Gilmer, H. G, C.V. Ciobanu, Mater. Manuf. Process. 24 (2009) 265–273.
[36] N. Chakraborti, K. Misra, P. Bhatt, N. Barman, R. Prasad, J. Phase Equilibr. 22 (2001) 525–530.
[37] L.D. Lloyd, R.L. Johnston, S. Salhi, Development of a genetic algorithm for optimization of nanoalloys, in: GECCO 2004, PT 2, Proceedings, Book Series: Lect. Notes Comput. Sci., vol. 3103, 2004, pp. 1316–1317.
[38] C.E. Mohn, S. Stølen, W. Kob, Mater. Manuf. Process. (2011) 26, doi:10.1080/10426914.2011.55202.
[39] M.M. Ali, Ind. Manage. Optim. 6 (2010) 29–46.
[40] C.A.C. Coello, R.L. Becerra, Mater. Manuf. Process. 24 (2009) 119–129.
[41] I.N. Egorov-Yegorov, G.S. Dulikravich, Mater. Manuf. Process. 20 (2005) 569–590.