Parametrization of interatomic potential functions using a genetic algorithm accelerated with a neural network

S. Bukkapatnam,[1] M. Malshe,[2] P. M. Agrawal,[2] L. M. Raff,[3] and R. Komanduri[2,*]

[1] *Industrial Engineering & Management, Oklahoma State University, Stillwater, Oklahoma 74078, USA*
[2] *Mechanical & Aerospace Engineering, Oklahoma State University, Stillwater, Oklahoma 74078, USA*
[3] *Chemistry Department, Oklahoma State University, Stillwater, Oklahoma 74078, USA*

(Received 13 July 2006; revised manuscript received 6 October 2006; published 8 December 2006)

A genetic algorithm (GA) can be used to fit highly nonlinear functional forms, such as empirical interatomic potentials from a large ensemble of data. The performance of a GA for fitting such functional forms is enhanced through an approach that is based on the use of a neural network (NN) to accelerate the computation of the fitness function for the GA. Application of the new approach for fitting an ensemble of potentials computed from *ab initio* calculations to a specified functional form (here, the Tersoff potential functional form is used as an example) has shown that the computational efficiency achieved through the use of a NN can reduce computational time by over two orders of magnitude. The potentials estimated from functions thus fitted were within 0.1% of the actual potential values. Specifically, the mean squared error (MSE) on molecular potentials was $<10^{-5}$ eV^2 for fitting Tersoff potentials, and <0.0025 eV^2 for fitting *ab initio* potential energies of isolated, 5-atom silicon clusters. Furthermore, since the potential was fitted to a physically meaningful Tersoff functional form, the resulting potential function appears to have the ability to extrapolate over a reasonable range of the parameter space, and may have a better accuracy in estimating the forces compared to that obtained from neural networks, which are often highly inaccurate when extrapolated. Hence, the method can be useful for rendering various molecular dynamics (MD) simulations more tractable. It is also apparent, based on the present investigation, that a Tersoff potential, albeit with different (GA parametrized) coefficients, is adequate for representing the *ab initio* potentials of 5-atom Si clusters.

DOI: 10.1103/PhysRevB.74.224102 PACS number(s): 02.70.Ns, 82.20.Kh, 82.20.Wt

I. INTRODUCTION

Many physical systems are inherently nonlinear. Effective modeling of nonlinear behaviors in these systems depends critically on fitting these highly nonlinear relationships using accurate, real data. Data-driven fitting of these functional relationships become extremely challenging, especially when the relationships are nonconvex and multimodal, and/or when the functional form has a combinatorial structure (Raff *et al.*[1]). The need for fitting such complex functional forms becomes critical in various engineering simulation applications.

Computer simulation techniques play an important role in investigations of various chemical, biological, and mechanical processes at the atomistic level. The theoretical principles underpinning the simulation models can be appropriately modified by benchmarking the results of the simulations with experimental observations. Thereafter, the simulations can be used to study the system behavior under different conditions for which experimental results are not readily available. The simulation results can thus guide future experimental plans. Therefore, computer simulations can complement both theoretical and experimental approaches. In conjunction with the developments in computer technology, the use of molecular dynamics (MD) and Monte Carlo (MC) simulations have greatly extended the range of problems that can be studied. Nowadays, simulation approaches are employed to study chemical reaction dynamics (Raff and Thompson,[2] Rahaman and Raff,[3,4] Kay and Raff,[5] Raff[6]). In engineering, such simulations are used to study the behavior of a material under varying conditions in various processes such as machining, tension, indentation, and melting (Komanduri *et al.*,[7–11] Agrawal *et al.*[12]). These simulations can provide useful insights into important mechanisms, such as phase transformation, dislocation dynamics which otherwise are not easily discernable by other techniques.

Central to MD/MC simulations is the development of potential-energy surfaces that describe the interactions among atoms within a system. The potential surfaces should lead to a realistic description of the interatomic interactions to match the experimental observations. Recently, we have conducted MD simulations of nanometric cutting of silicon with a diamond tool (Komanduri *et al.*[9]). We found the material ahead of the tool in the workpiece to undergo significant deformation. Under these conditions the atomic arrangements will be far from equilibrium and empirical potentials whose derivation is based on equilibrium conditions may not be adequate to describe experimental observations. We have also observed from these simulations that various clusters of Si—from Si_2 to Si_9 are formed, with the majority being Si_5 (Komanduri *et al.*[9] and Raff *et al.*[1]). Need, therefore, arises for the development of accurate potentials of these clusters in order to study their behavior further, which is the objective of this investigation.

The usual approach for developing a potential is to determine a functional form motivated by physical intuition, and adjust the parameters of the functional form to fit *ab initio* potential values obtained for a set of atomic configurations (Raff *et al.*,[1] Agrawal *et al.*[13]) and/or some values of the physical properties experimentally determined for the stated configurations (Agrawal *et al.*[12]). The *ab initio* potential values are derived by solving Schrödinger's equation (Pople *et al.*,[14] Raghavachari *et al.*,[15] Raghavachari and Trucks,[16] Moller and Plesset[17]). In order to be useful for the purpose of conducting MD studies, the fitted potential must not only provide an accurate fit to the database, it must also accurately interpolate between points in the database. In addition, it is often highly desirable that the surface permit reasonably accurate extrapolation to points that lie in regions of configuration space outside the range of the database.

Parametrization of a typical interatomic potential functional form (e.g., Tersoff[18–22]) is difficult because the functions tend to be nonlinear and combinatorial in both space and time. The analytic vinyl bromide potential surface developed by Rahaman and Raff,[3,4] which contained over 100 parameters, required some nine months of human effort to achieve an accurate fit to a database comprising both experimental and *ab initio* energies. The problem is exacerbated by the fact that every time a given functional form is fitted to a new system with a different chemical composition, the entire fitting procedure must be repeated. For example, a Tersoff potential fitted to the chemical and physical properties of bulk silicon will have very different values for the surface parameters than a Tersoff potential that is fitted to *ab initio* electronic structure energies for a set of small silicon clusters, $\mathrm{Si_n}$, where n is six or less. Similarly, a Brenner potential (Brenner[23]) with parameters fitted to the energies and properties of a carbon nanotube will not provide an adequate surface for bulk silicon or $\mathrm{Si_n}$ clusters. In each case, the entire fitting procedure must be repeated.

It may be possible to use methods such as genetic algorithms (GAs) and neural networks (NNs) to derive adequate empirical potentials for various applications (Raff *et al.*[1] and Hobday *et al.*[24]). If such methods can be made robust and sufficiently fast, much of the labor associated with obtaining analytic potential fits could be eliminated. Also, the range of systems that can be effectively addressed using MD and/or MC methods can be significantly extended.

Briefly, a genetic algorithm (GA) uses a stochastic global search method that mimics the process of natural biological evolution (Holland[25]). GAs operate on a population of potential solutions applying the principle of survival of the fittest to generate progressively better approximations to a solution. A new set of approximations is generated in each iteration (also known as generation) of a GA through the process of selecting individuals from the solution space according to their fitness levels, and breeding them together using operators borrowed from natural genetics. This process leads to the evolution of populations of individuals that have a higher probability of being "fitter," i.e., better approximations of the specified potential values, than the individuals they were created from, just as in natural adaptation.

The most time consuming part in one implemention of a GA is often the evaluation of the objective or fitness function. The objective function O[P] is expressed as sum squared error computed over a given large ensemble of data. Consequently, the time required for evaluating the objective function becomes an important factor. Since a GA is well suited for implementing on parallel computers, the time required for evaluating the objective function can be reduced significantly by parallel processing. A better approach would be to map out the objective function using several possible solutions concurrently or beforehand to improve computational efficiency of the GA prior to its execution, and using this information to implement the GA. This will obviate the need for cumbersome direct evaluation of the objective function.

Neural networks may be best suited to map the functional relationship between the objective function and the various parameters of the specific functional form (Hagan *et al.*[35]). This study presents an approach that combines the universal function approximation capability of multilayer neural networks to accelerate a GA for fitting atomic system potentials. The approach involves evaluating the objective function, which for the present application is the mean squared error (MSE) between the computed and model-estimated potential, and training a multilayer neural network with decision variables as input and the objective function as output. This trained neural network is then used as part of a GA for computing the objective function. This approach has been found to speed up a GA in general by eliminating redundant computation of the objective function and by precomputing offline relative to the GA. The results show that the set of parameters obtained using the approach described in this paper yield closer fits for Tersoff potential energies (MSE $<1.32\times10^{-6}\ \mathrm{eV}^2$) and *ab initio* potential energies (MSE $<0.0025\ \mathrm{eV}^2$) for isolated 5-atom silicon clusters. Thus, in cases where the objective function is expressed as MSE over a large number of data points, a NN can be a suitable means for faster estimation of the objective function to accelerate GA. It is also apparent, based on the present investigation, that a Tersoff potential, albeit with different (GA parametrized) coefficients, is adequate for representing the *ab initio* potentials of 5-atom Si clusters.

The organization of the paper is as follows: a brief background on the interatomic potentials and a brief review of literature are presented in Sec. II; the effects of various parameters on the potential energy are presented in Sec. III; this is followed by Sec. IV on the results and discussion; and finally Sec. V presents the conclusions resulting from this study.

II. BACKGROUND AND LITERATURE REVIEW

A. Interatomic potential functions for silicon

The most important part of the MD/MC simulation is the development of the potential energy surfaces that can model the system potentials sufficiently close to reality. Interatomic potential functions are invariably used in MD simulations to capture the total potential energy of the system. In recent years, many empirical potentials for silicon, such as the Stillinger-Weber potential (Stillinger and Weber[26]), Tersoff potential (Tersoff[19–22]), Bolding-Andersen potential (Bolding and Andersen[27]), and Brenner potential (Brenner[23]) were developed and applied to a number of different systems. Each of the models for interatomic potentials differs in the degree of sophistication, functional form, useful range of values to suit a specific application. Not only are the surfaces for small clusters difficult to model (Balamane *et al.*,[28] Kaxiras[29]), even bulk materials, including crystalline and amorphous phases, solid defects and liquid phases have not been provided with a transferable, single potential function. The usual approach for developing empirical potentials is to arrive at a parametrized functional form motivated by physical intuition, and then to find a set of parameters by fitting to either *ab initio* data and/or experimental observations for various atomic structures. A covalent material presents an additional challenge because the potential function needs to capture complex quantum mechanical effects, such as chemical bond formation, hybridization, metallization, charge transfer, bond stretching, and bond bending. Despite a wide range of functional forms and fitting strategies, most of the current empirical potential function parametrizations have been successful in the regime for which they were parametrized, but have shown a lack of transferability. The Rahaman vinyl bromide potential (Rahaman and Raff[3,4]) is a typical example.

The Tersoff potential, which is one of the commonly used functional forms for modeling the interatomic interaction of group IV semiconductor materials, such as silicon is used in the present study. The functional form of Tersoff potential is given by (Tersoff[22])

$$V = \sum_i v_i = \frac{1}{2}\sum_{i \neq j} v_{ij},$$

$$v_{ij} = f_C(r_{ij})[f_R(r_{ij}) + b_{ij} f_A(r_{ij})]. \tag{1}$$

where V is the total potential energy of the system. f_R and f_A are repulsive and attractive pair potentials, respectively, and f_C is a cutoff function given by

$$f_R(r_{ij}) = A_{ij} e^{(-\lambda_{ij} r_{ij})},$$

$$f_A(r_{ij}) = -B_{ij} e^{(-\mu_{ij} r_{ij})}, \tag{2}$$

$$f_C(r_{ij}) = \begin{cases} 1, & r_{ij} < R \\ \dfrac{1}{2} + \dfrac{1}{2}\cos\left[\pi \dfrac{(r_{ij} - R)}{(S - R)}\right], & R < r_{ij} < S \\ 0, & r_{ij} > S, \end{cases}$$

where r_{ij} is the bond distance between atoms i and j, S is the cutoff radius, R is the inner cutoff radius, and $\lambda_{ij} = \lambda$, and $\mu_{ij} = \mu$ are potential parameters. The strength of each bond depends upon the local environment. It is lowered when the number of neighbors is relatively high. This dependence is expressed by the parameter b_{ij}

$$b_{ij} = \chi_{ij}(1 + \beta_i^{n_i} \zeta_{ij}^{n_i})^{-1/2n},$$

$$\zeta_{ij} = \sum_{k \neq i,j} f_C(r_{ik}) \omega_{ik} g(\theta_{ijk}), \tag{3}$$

$$g(\theta_{ijk}) = 1 + \frac{c_i^2}{d_i^2} - \frac{c_i^2}{d_i^2 + [h_i - \cos(\theta_{ijk})]^2},$$

which can diminish the attractive force relative to the repulsive force. Here

TABLE I. Parameters of Tersoff Potential for Silicon [Tersoff (Ref. 22)].

Tersoff Parameters	Values
A (eV)$\times 10^3$	1.8308
B (eV)$\times 10^2$	4.7118
λ (Å^{-1})	2.4799
μ (Å^{-1})	1.7322
$\beta \times 10^{-6}$	1.1
n	0.787
$c \times 10^5$	1.0039
d	16.217
h	−0.59825
R (Å)	2.7
S (Å)	3.0

$$\lambda_{ij} = \frac{\lambda_i + \lambda_j}{2}; \quad \mu_{ij} = \frac{\mu_i + \mu_j}{2}. \tag{4}$$

The term ζ_{ij} defines the effective coordination number of atoms taking into account the relative distance between two neighbors (r_{ij}, r_{ik}) and the bond angle θ_{ijk}. The function $g(\theta)$ has a minimum at $h = \cos(\theta)$. The parameter d determines sensitivity of the potential to the angle and c expresses the strength of the angular effects. The parameters R and S are not optimized but chosen so as to include first neighbors only for several selected, high-symmetry structures, such as graphite, diamond, as well as simple cubic and face centered cubic lattice structures.

The parameter x_{ij} strengthens or weakens heteropolar bonds in multicomponent systems. $x_{ii} = 1$, and $x_{ij} = x_{ji}$. Also, $\omega_{ii} = 1$. The potential was calibrated initially for silicon (Tersoff[19]) and subsequently for carbon (Tersoff[20,21]). The parameters chosen to fit theoretical and experimental data were for e.g., cohesive energy, lattice constants, and bulk modulus obtained from realistic and hypothetical configurations. Table I lists the parameters of Tersoff potential for silicon as specified in an earlier work (Tersoff[22]). These values are adequate for computing bulk parameters of silicon at room temperature. However, they do not agree with the potentials computed based on first-principle calculations at different temperatures and for isolated configurations. As a result, the application of the Tersoff potential with the parameters given in Table I may lead to erroneous MD computations.

Internal atomic coordinates are generally the most efficient way to specify the potential energy [see Eqs. (1)–(3)]. Internal coordinates are specified using bond distances, bond angles, and dihedral (torsional) angles. Figure 1 shows variables involved in internal coordinates. The dihedral angle is the angle between two planes passing through atoms *i*-*j*-*k* and *j*-*k*-*l*, respectively. It is measured as the angle between the vector normal to these planes. However, since the internal coordinates are highly coupled, it is convenient to solve the equations of motion during MD simulation in the Cartesian coordinate system, and convert Cartesian coordinates to

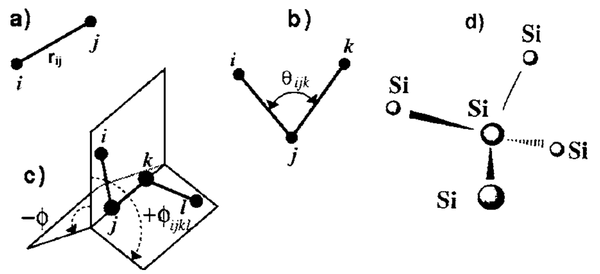

FIG. 1. Internal coordinates: (a) bond distance, (b) bond angle, (c) dihedral angle, and (d) configuration of 5 silicon atoms.

corresponding internal coordinates after imposing certain constraints [e.g., first neighbor represented as atom 2 in Gaussian 03 software (Frisch *et al.*[30])].

B. Applications of NNs for GA acceleration

Although GA uses physically meaningful functional forms and possess excellent extrapolation capabilities, fitness (or the objective) function evaluations are often a major bottleneck in using GAs and other evolutionary algorithms for optimization applications, including optimal parametrization of functional forms. Surrogate modeling approaches, including the use of NNs, have been attempted to minimize the computational efforts involved in fitness (objective) function evaluations (Buche *et al.*,[31] Yan and Minsker,[32] Kim and Khosla,[33] Hacioglu,[34] Hagan *et al.*[35]). These have been applied to address a variety of issues arising in aerospace applications from shape optimization to cost minimization. All these approaches use NNs and/or other modeling approaches as cheaper surrogates for costly simulations. It may be noted that the literature on GAs is quite extensive, covering a wide range of applications. In the following, due to constraints on space, only a few examples are given to provide an appreciation of the subject matter. Comprehensive reviews of applications GAs are available in the literature (see for e.g., Tang *et al.*,[36] Back *et al.*[37]).

Yan and Minsker[32] used a dynamic meta modeling approach in which artificial neural networks (ANN) and support vector machines (SVM) were embedded into a GA optimization framework to replace time consuming flow and containment transport models. Data produced from early generations of the GA are sampled to train the ANN and SVM and the numerical models were periodically called to dynamically update the ANN and SVM, allowing the meta model to adapt to the area in which the GA was searching and providing more accuracy.

Farritor and Zhang[38] used a NN to evaluate the performance of modular robots designed using genetic algorithms. Candidate designs generated by human designers along with randomly generated designs were used to train a NN fitness function. Then, the GA evolves new designs that the human designer might not even conceive, let alone come up with an actual design.

Coit and Smith[39,40] presented an optimization approach using a GA to identify the preferred choice of components and the optimal levels of redundancy for a reliability design problem. For complex design problems, estimation of the reliability requires considerable effort. The approach was to use a NN to estimate system reliability as a function of the component reliabilities and the design configuration. In this way, multiple estimates of system reliability were available without solving a new probability model for each candidate solution.

Burton and Vladimirova[41] developed NN based adaptive resonance theory (ART) called ARTMAP to control the cluster creation process. The ART NN utilizes unsupervised learning and clustering algorithms to recognize patterns. The fitness of the individuals of the population is determined as a function of the degree of similarity between clustered patterns and the individual. Buche *et al.*[31] developed a Gaussian process model approach for fast surrogate evaluation of the objective function. Their approach performed better than other evolutionary optimization strategies for profile optimization of gas turbine and compressor blades of aircraft engines.

In all the aforementioned applications, the nonlinearity of the underlying real functional forms has necessitated acceleration using NNs and other surrogate models. In the present context, in addition to the complexity involved in function computation, the combinatorial nature of the objective function offers additional variables for NN acceleration. In fact, the application of GA was almost intractable for parametrization without acceleration as will become evident in Sec. III. Therefore, although GAs have been applied for the derivation of optimal atomic configurations (Smith *et al.*[42]), the likely computational overhead involved in repetitive determination of the objective function has prevented GAs from being applied for parametrizing interatomic potentials. The use of NN renders GAs attractive for such applications where the repetitive computation of the objective function is a major bottleneck.

III. PROBLEM DESCRIPTION AND APPROACH

Genetic algorithms (GAs) and other nonlinear functional approximation routines based on a stochastic gradient search require the specification and evaluation of an objective or fitness function to determine the quality of fit. Problems involving the repetitive evaluation of the objective function (such as, mean squared error (MSE) in fitting a complicated function) becomes computationally time consuming. This study presents an approach based on using the function approximation capability of NNs for computation of the objective function. The method is applied for parameter determination of interatomic potential for silicon using the Tersoff functional form (Tersoff[19–22]).

The objective function O[P] was defined as the mean squared error (MSE) between the potential energy calculated using a set of parameters for a Tersoff functional form and the actual energy values obtained from first-principle calculations for each of the N_c configurations (x_k), $k=1..N_c$ using one of the N_p set of parameters

$$\mathrm{O}[\mathrm{P}] = \frac{\sum_{k=1}^{N_c} (V_k - \hat{V}_k[\mathrm{P}])^2}{N_c} \tag{5}$$

where [P] is the set of parameter vector values for the chosen potential function, N_c is the number of configurations used,

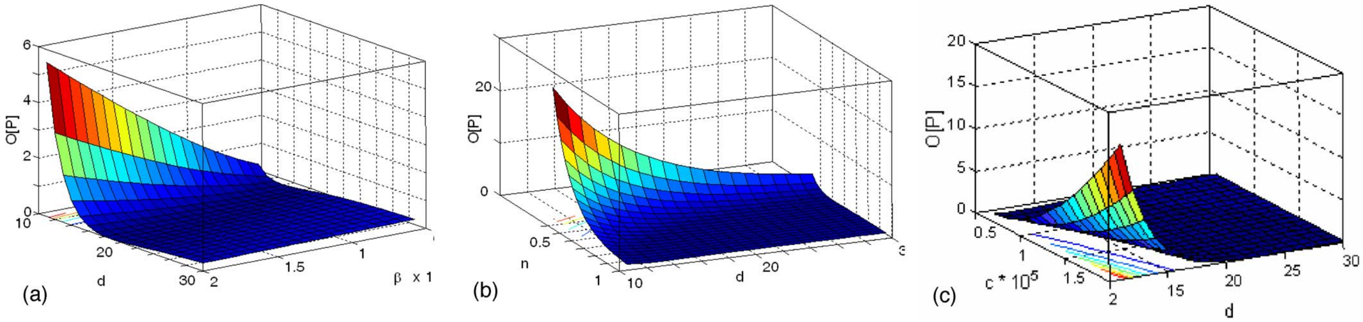

FIG. 2. (Color online) Variation of the objective function O[P] with (a) β and d (b) n and d and (c) c and d, respectively, showing large plateau regions.

$\hat{V}_k$ is the potential computed from the chosen function with parameter set at [P], Vk is the actual potential values to be fitted, obtained here from Tersoff or *ab initio* calculations.

As shown in Fig. 2, the gradients of O[P] are steep for small values of the parameters (β, d, and n). As the value of the parameters increase, the gradient of the objective function decreases until it reaches a large plateau region marked by near-zero gradients. Conventional gradient-based optimization methods tend to be inefficient for objective functions of such form because gradients tend to be computationally cumbersome to evaluate. Ill-conditioned numerical structures, slow convergence, and suboptimal solutions are known to manifest whenever gradient search algorithms are employed for multivariable combinatorial optimization with such objective functions.

GAs are more effective compared to traditional gradient-based methods. Our approach to NN-accelerated GA is summarized in Fig. 3. The approach involves off-line training of a NN to capture the relationship between the parameter set [P] and the objective function [O[P]], and using the NN estimates of [O[P]] as surrogates for GA optimization. The entire procedure of parameter optimization is carried out in multiple stages. All initial stages consist of two major components, namely, off-line NN training and GA parametrization, as shown in Fig. 3. The procedure begins with the generation of an initial set of parameter vector values [P], and computation of corresponding potential [V[P]] and hence, the objective function as described in Eq. (4). A multilayer NN is trained with potential energy parameters [P] as inputs and corresponding objective function O[P] as the output. Regularization and early stopping are used during training to avoid overfitting the training data. Next, the GA is executed based on NN estimates of O[P]. The GA consists of the following steps:

(a) *Initialization of population*: A set of parameter vector values [P] of the considered potential functional form (Tersoff[22]) are generated within a specified range. The dimensionality of the parameter vector P was 9 corresponding to the number of parameters used in the Tersoff potential. The set [P] contained 50 such elements; i.e., $N_p = 50$. Each parameter within a set was coded in a binary string of length 25.

(b) *Generating a new set of parameter values* [P] *using genetic operators*: The objective function set [O[P]] corresponding to a set of parameters generated [P], is evaluated using a NN, trained off line. Genetic operators, such as crossover and mutation (Holland[25]) are used to arrive at a new set of parameters for the Tersoff potential functional form. The new set of parameters for Tersoff potential is selected based on the fitness. In other words, the smaller the value of O[P], the higher the probability for a parameter vector value P to be selected. The GA terminates after the maximum number of iterations are reached, or if the GA solutions appear to have converged. If the fit at the end of the phase is below a threshold level, subsequent phases consisting of NN retraining and GA parametrization are undertaken. However, the range set for the parameter vector P is reduced and the length of binary string used to encode each of the parameters is increased, for example, from 25 to 50 bits to improve the resolution. Once the fit improves above a threshold level, it is assumed that the parameter values lie in the basin of the global minimum. A gradient based method is used to determine the best fit.

IV. IMPLEMENTATION DETAILS AND RESULTS

First, we evaluated the approach for its ability to approximate actual Tersoff potential parameters. This evaluation was necessary to benchmark the approximation capability of the GA-NN approach. Next, we applied this approach to approximate *ab initio* potential values with a Tersoff functional form.

A. Approximation of Tersoff potential

1. Neural network (NN) training for mean squared error (MSE)

A two-layered feedforward NN was trained for mean squared error (MSE) (i.e., between the potential calculated using Tersoff parameters and GA parameters) for $N_c = 1000$ configurations of 5-atom Si clusters. The NN consisted of 40 neurons in the hidden layer and one neuron in the output layer. The input exemplar patterns for training the NN consisted of 4000 arbitrarily chosen realizations of the nine-dimensional parameter vector [P] of Tersoff functional form and the corresponding [O[P]] obtained from Eq (5). Such a network is commonly denoted by the notation (9-40-1). The trained NN was tested using 500 element sets of [P] and [O[P]]. Figures 4(a) and 4(b) show a comparison of NN outputs for the testing sets during the first and the fourth stages of GA, respectively. The standard deviations of the actual

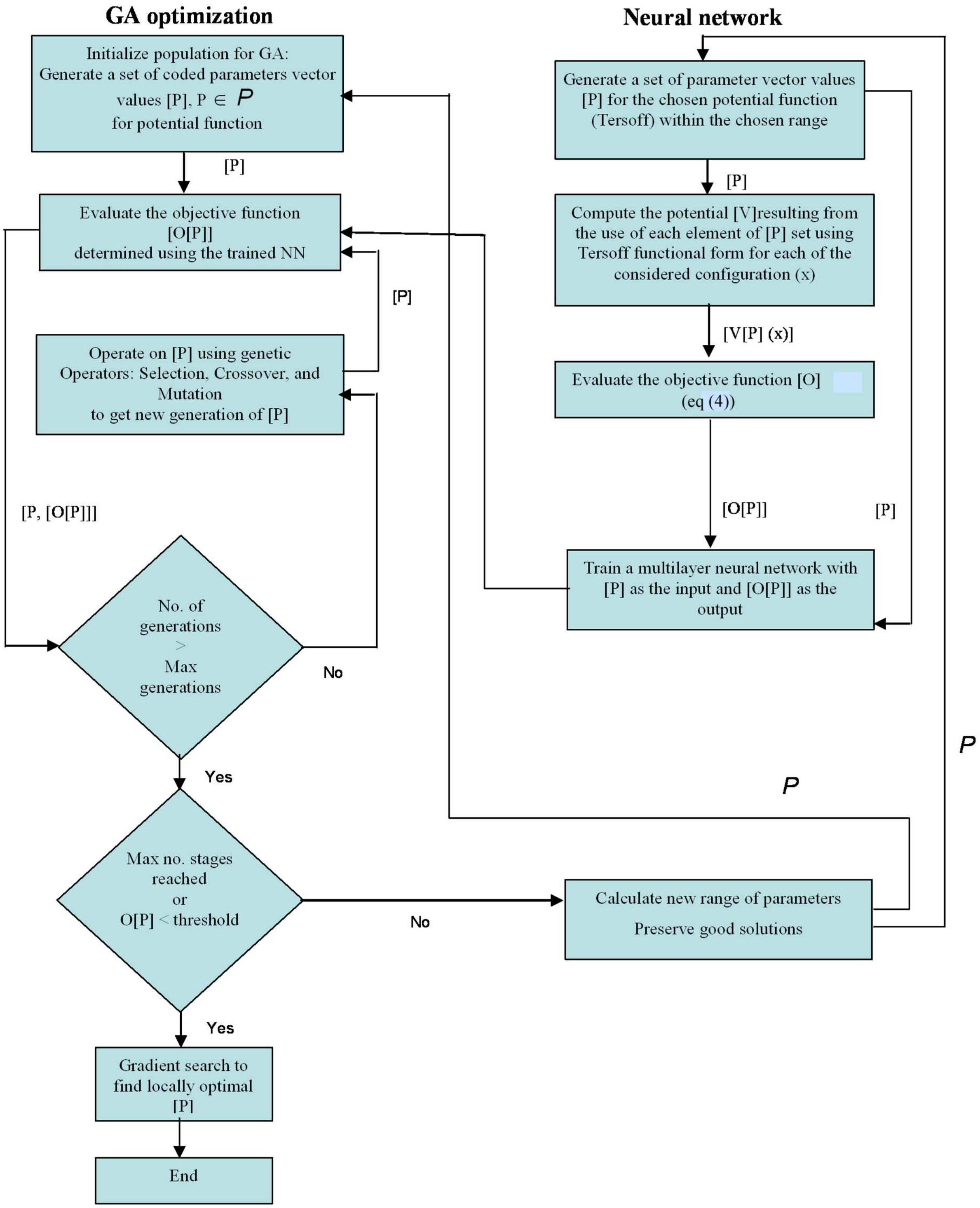
GA optimization
Initialize population for GA: Generate a set of coded parameters vector values [P], P ∈ P for potential function
[P]
Evaluate the objective function [O[P]] determined using the trained NN
[P]
Operate on [P] using genetic Operators: Selection, Crossover, and Mutation to get new generation of [P]
[P, [O[P]]]
No. of generations > Max generations
No
Yes
Max no. stages reached or O[P] < threshold
No
Yes
Gradient search to find locally optimal [P]
End
Neural network
Generate a set of parameter vector values [P] for the chosen potential function (Tersoff) within the chosen range
[P]
Compute the potential [V]resulting from the use of each element of [P] set using Tersoff functional form for each of the considered configuration (x)
[V[P] (x)]
Evaluate the objective function [O] (eq (4))
[O[P]]
[P]
Train a multilayer neural network with [P] as the input and [O[P]] as the output
P
P
Calculate new range of parameters
Preserve good solutions

FIG. 3. (Color online) Flowchart of the approach used for GA optimization.

outputs [O[P]] and those computed using a NN [O[P]] were 0.0636 eV^2 for Stage 1, and 0.0036 eV^2 for the fourth stage.

2. GA-search for optimal Tersoff parameters

The NN-estimated [O[P]] values were used to evaluate the fitness of various input sets [P] during GA. The nine parameters for Tersoff potential, namely, A, B, λ, μ, β, n, c, d, and h (see Table I) were first encoded in a binary string. During the initial stages 25 binary bits, and during the final stages 50 bits were used to represent each variable. Optimization of parameters was carried out in four stages. During each stage the range for the parameter set was narrowed to improve the accuracy in successive stages. The crossover and mutation rates were varied during each stage, as shown in Figs. 5(a) and 5(b), respectively. The mutation rates were

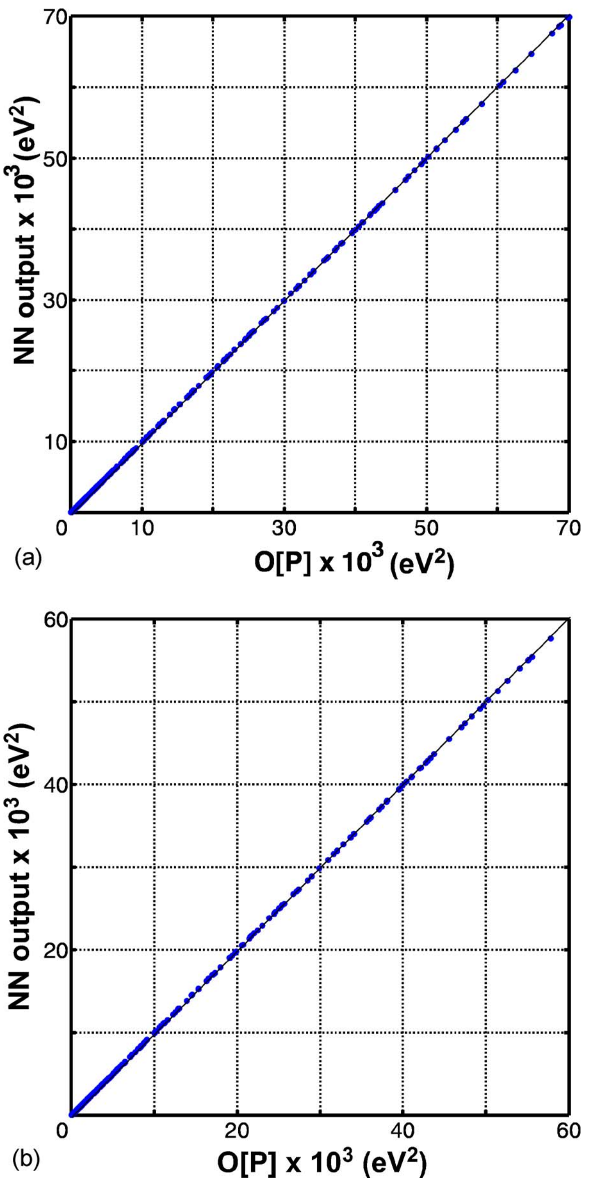

FIG. 4. (Color online) Comparison of the NN output with the objective function O[P] for the testing set: (a) first stage and (b) fourth stage, respectively.

gradually decreased during each stage from 0.92 to 0.12 using a functional form $[1-k_1(1+k_2e^{-n/N})^{-1}]$ where n is the current iteration and number N is the total number of specified iterations, and k_1 and k_2 are constants chosen for each stage to yield the mutation rate variation profiles shown in Fig. 5(a). The crossover rate was varied from 0.9 in Stage I to 0.62 in Stage IV as shown in Fig. 5(b). The mutation rate affects the convergence of the final solution. It was found that during the initial stages, a larger value for mutation rate facilitates searching through a large parameter space, while a smaller value for mutation rate towards the end facilitates convergence to the optimal solution.

The convergence pattern of MSE between the GA parametrized Tersoff potential value and actual Tersoff potential value taken over all the configurations is summarized in Fig. 6 and Table II. It was observed that during the initial stages of GA, the use of multipoint crossover provides a faster convergence as opposed to single-point crossover (Bridges and Goldberg[43]). This could be because multipoint crossover tends to produce more variation among individuals during successive iterations, which helps in narrowing down the range of parameters during initial stages. On the other hand, single-point crossover along with high resolution (number of bits used in binary representation) helps to improve the accuracy.

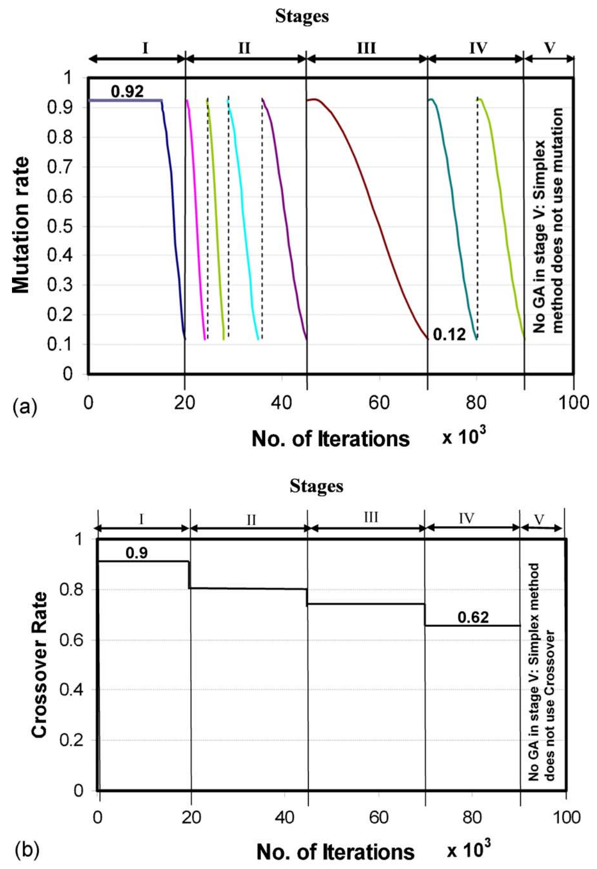

FIG. 5. (Color online) (a) Variation of the mutation rate during each stage with number of iterations. (b) Variation of the crossover rate during each stage with the number of iterations.

Figure 7 illustrates the closeness of potentials computed based on GA-optimized parameters $V[\mathrm{P}](x)$ relative to those

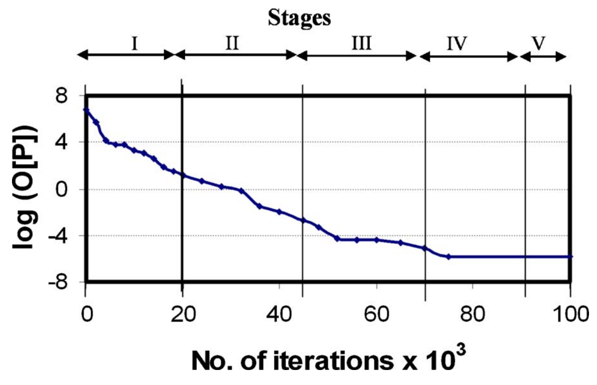

FIG. 6. (Color online). Variation of MSE with the number of iterations at various stages for parametrization of Tersoff potential.

TABLE II. Variation of MSE at different stages of GA fitting of Tersoff potential.

Stage no.	Stage I	Stage II				Stage III		Stage IV		Stage V
MSE (eV^2)	0.01	0.0019	4.72×10^{-4}	5.45×10^{-5}	4.32×10^{-5}	3.57×10^{-5}	2.53×10^{-5}	7.65×10^{-6}	1.43×10^{-6}	1.32×10^{-6}

obtained using original Tersoff parameters $V(x)$. It isto be noted that it took five stages to make the GA optimized Tersoff parameters values match the actual values of the Tersoff potential given in Table I (Tersoff[22]). At this point the MSE dropped below 10^{-5} eV^2. Among all Tersoff parameters, the potential appears to be most sensitive to variations in A and B. Consequently, the GA was able to determine the near-optimal settings of these values during the early stages itself. Significant training efforts spanning multiple stages of GA were needed to optimize the other, less sensitive parameters (e.g., λ and h). The advantage of using NN trained for MSE is that it obviates the need to calculate Tersoff potential energy for each set of parameters and for each configuration. The use of a NN has reduced the computational time from ~1 week without the NN to ~8 h with NN.

3. Approximation of ab initio energy using Tersoff functional form

Next, we investigated the fitting of *ab initio* potentials of 5-atom Si clusters using a Tersoff functional form. We used 10 000 configurations of Si_5 clusters from a database established from MD simulations of nanometric cutting of Si to compute the target objective function for training and testing. The configurations and their potentials were obtained after performing MD simulations of nanometric cutting of a silicon workpiece using a single point cutting tool (+15° rake angle and 10° clearance angle) at 1 Å cutting depth, 54.3 Å cutting width, and 491.2 $m\,s^{-1}$ cutting speed [see Fig. 8(a)].[1,9] A Tersoff potential with parameters corresponding to bulk Si (Table I) was employed in the initial step of the iterative MD simulation process. At every time step of the MD simulations, the configurations of silicon clusters that are present in front of the tool, in the chip, and within a few unit cell distances in the workpiece underneath the tool are stored. It was observed from an examination of a histogram of atomic configurations of Si in the primary and secondary deformation zones [see Fig. 8(b)] that out of about 39 000 configurations identified, some 57% of the Si atoms evolve into clusters of five atoms. The histogram also shows that Si_7 (20%) followed by Si_6 and Si_4 clusters (<10%) occur a significant number of times in the deformation zones during the machining of a Si workpiece. This observation has led us to first focus on deriving the potentials of five atom Si clusters as opposed to other topologies.

The database obtained from the cutting simulations was augmented with 10 000 additional Si_5 configurations near equilibrium. These configurations were obtained from the bulk material by placing thermal energy corresponding to a temperature of 300 K in the silicon workpiece and then following the vibrational motions of the lattice using MD with the Tersoff empirical potential for bulk silicon. Subsequent to

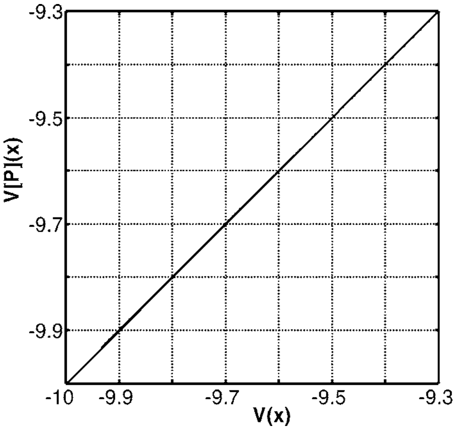

FIG. 7. Comparison of the potential computed using Tersoff $V(x)$ and GA optimized parameters $V[P](x)$. Energies are given in eV.

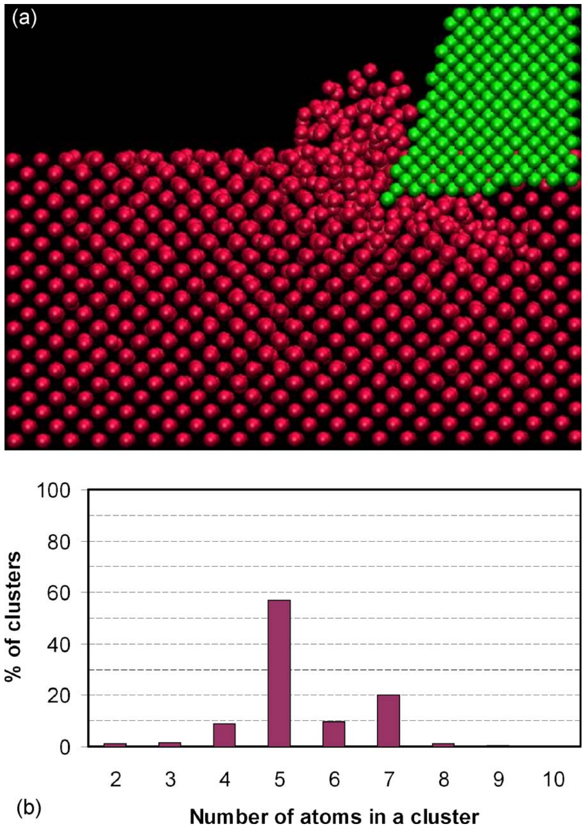

FIG. 8. (Color online) (a) Snapshot of MD simulation of nanometric cutting of silicon showing atomic configuration of Si workpiece and the diamond tool used. (b) Histogram showing the distribution of various Si clusters in the primary and secondary deformation zones ahead of the cutting tool in nanometric cutting as shown in (a).

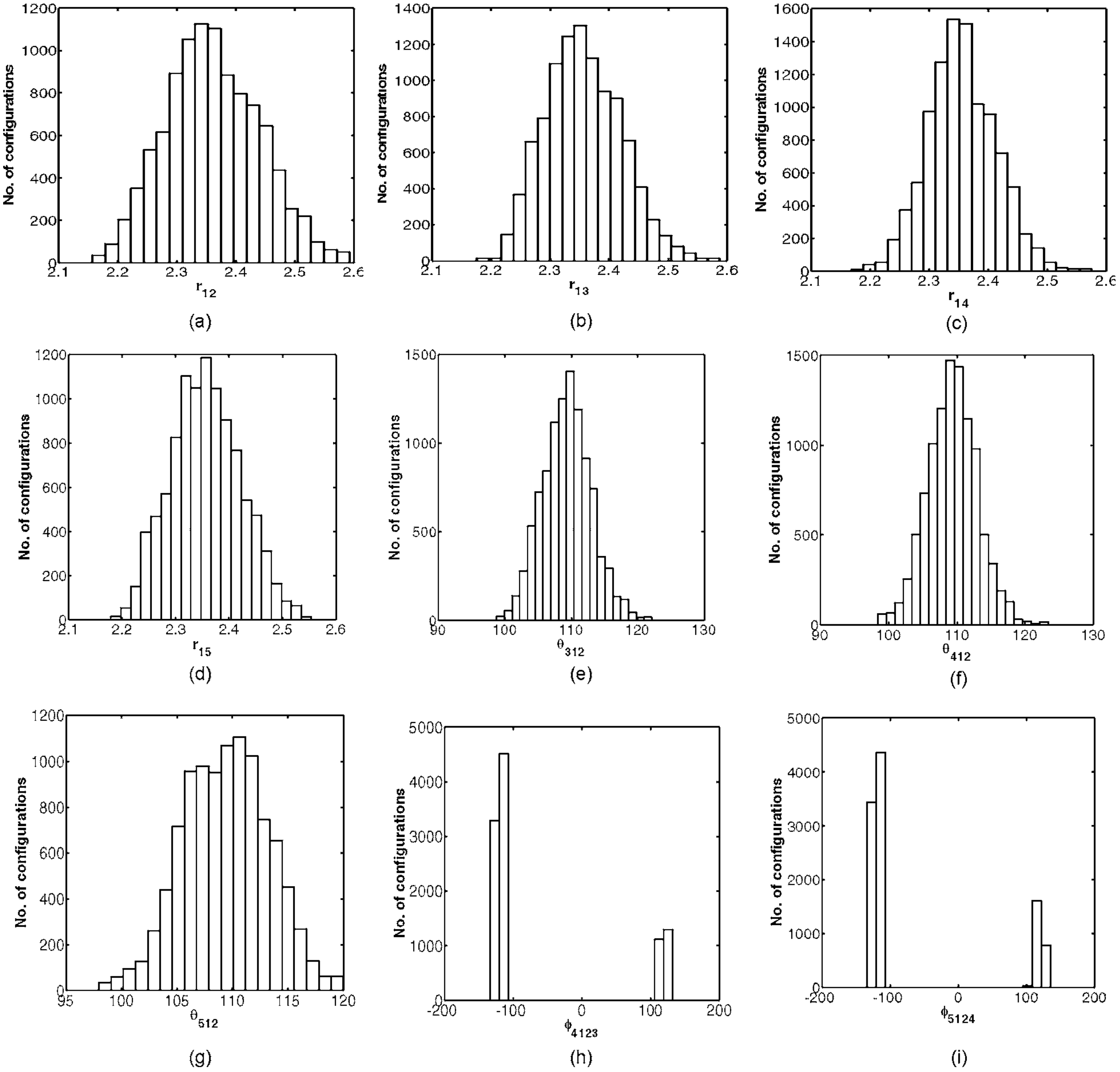

FIG. 9. (a)–(i) Histograms of the number of configurations as a function of internal coordinates, (a)–(d) bond distance (r), (e)–(g) bond angles (θ), and (h)–(i) dihedral angle (ϕ).

the MD calculations, the energies and the force fields for each of the 28 000 stored configurations are computed using the density functional theory (DFT) with a 6-31G** basis set and the B3LYP procedure for incorporating correlation energy. Next, the 10 000 configurations needed for GA training and testing were randomly selected from among the available configurations in the database.

The objective function was computed for every one of the randomly generated 9000 parameter vector realizations, and trained a two-layer NN to approximate the objective function [O[P]] as described in Sec. IV A. Figures 9(a)–9(i) show histogram plots of the number of configurations as a function of nine internal coordinates used to calculate the *ab initio* energies using Gaussian 03 software (Frisch *et al.*[30]). The standard deviation of the errors between the objective functions was 0.00847 eV^2 during Stage 1 NN approximation and 0.00358 eV^2 in the final stage (Stage IV) of NN approximation. Figures 10(a) and 10(b) show how the NN estimate of O[P] compares with actual values for different settings of [P]. The low values of the standard deviation, evident from these two figures, indicate the closeness of the NN estimate of O[P] to the actual O[P], whose computation is extremely cumbersome. Next, the GA was trained for 5 stages, and the NN was retrained at the end of Stages I through III. In the final step, a Nelder-Mead simplex algorithm (Nelder and Mead[44]) was used to arrive at the optimal set of parameters. The specific GA parameters are used as follows:

(i) Stage I: 100 individuals, 15 bits to represent each variable, 4 point crossover.

(2) Stage II: 50 individuals, 20 bits to represent each variable, 2 point crossover

(3) Stage III: 50 individual, 50 bits to represent each variable, 2 point crossover.

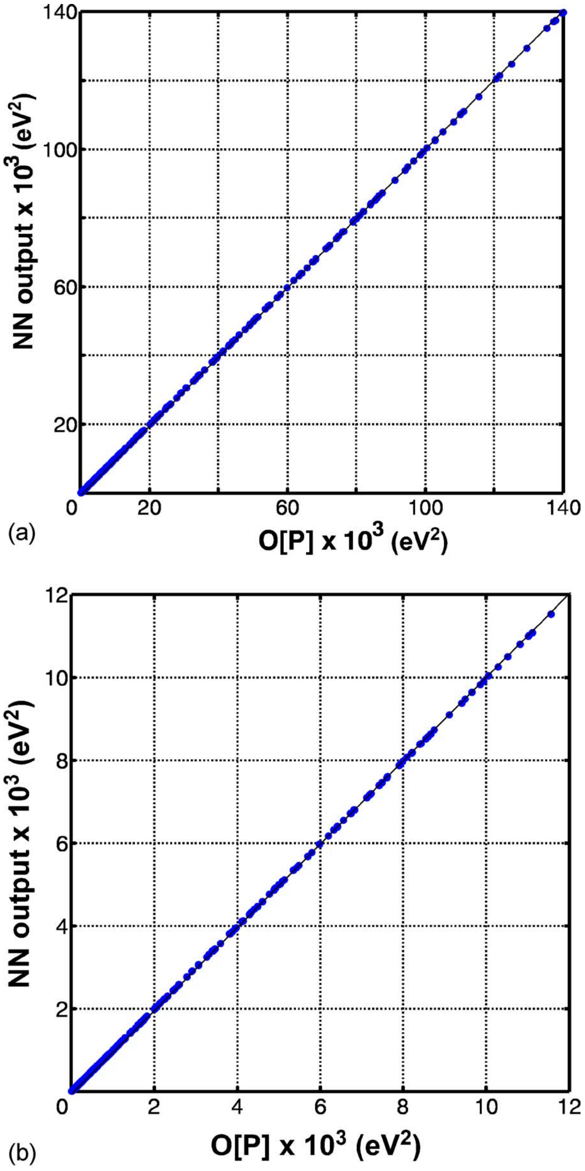

FIG. 10. (Color online). Comparison of the NN output with the objective function O[P] for the testing set at (a) first and (b) fourth stages, respectively.

(4) Stage IV: 25 individual, 75 bits to represent each variable, single point crossover.

(5) Stage V: Nelder-Mead simplex method (does not use genetic operators)

The variation of the best fitness function (O[P]) values at the end of each stage is summarized in Fig. 11. Table III summarizes the parameter values for Tersoff functional form that seem to adequately fit the *ab initio* potentials of 5 atom Si clusters. The MSE of the GA fit was reduced from 0.74 eV^2 to about 0.28 eV^2 during four stages of GA execution (see Fig. 11 and Table III). The final stage of Nelder-Mead simplex reduced the errors by two orders of magnitude to 0.0026 eV^2. In contrast, the MSE resulting from the use of a Nelder-Mead simplex method, instead of a GA, for fitting has yielded a MSE of 0.28 eV^2 and those from the application of a simplex method with Stage I GA results as starting points has yielded an average MSE of about 0.12 eV^2. These results suggest that it would be necessary to have the GA converge to the best possible solution (leading to the identification of the correct valley containing the global minima) prior to the application of a Nelder-Mead simplex (Nelder and Mead[44]). We next tested the GA-parametrized Tersoff functional form to fit the potentials of 28,200 configurations (which included those not used in the NN for fitting the GA). The results plotted in Fig. 12 show some dispersion around the 45° line, corresponding to a standard deviation of 0.0507 eV. Most of the points (≈90%) lie along the 45° line (although in the figure we see mainly those points which are outside the line) indicating that the GA-parametrized model estimates the *ab inito* potentials accurately for a vast majority of the configurations.

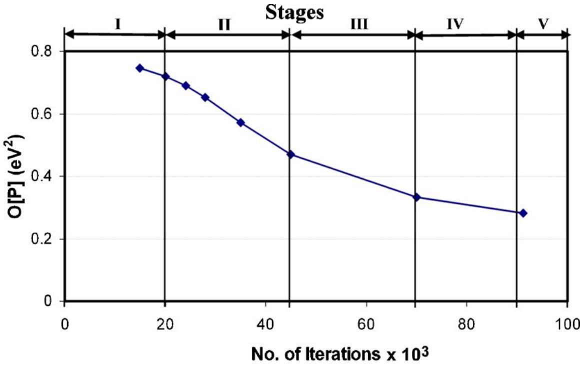

FIG. 11. (Color online) Variation of objective function O[P] (here, MSE) at various stages of GA fitting of *ab initio* potentials to Tersoff functional form.

We next compared the extrapolation capability of the GA-estimated potential function. Figure 13 shows that both NN and GA optimized Tersoff potential fit *ab initio* energy very well within a range of data that was used for the fitting. NN fit is accurate for bond distances up to 3 Å, whereas GA-Tersoff is accurate up to 2.7 Å. This can be attributed to the fact that for GA-Tersoff, one starts with an assumed functional form based on some physical intuition involved in the bonding environment and then adjust parameters to fit *ab initio* energy using the functional form. As a result, the accuracy of the fit critically depends on the underlying functional form used, whereas in the case of NN fitting, no functional form is assumed and the accuracy of the fit depends on the number of neurons used. It should be noted that NN fit is good only over a range of data that was used for fitting, i.e., a NN fit is good only for interpolation and not for extrapolation. An additional advantage of a GA-Tersoff method is that it can be used for extrapolation. As seen in Fig. 13, for bond distances <2.1 Å, both GA-Tersoff and NN fit start deviating from *ab initio* energy. While the energy predicted by the NN fit remains almost constant, the energy predicted by GA-Tersoff follows the trend of *ab initio* energy although the magnitudes deviate somewhat from the *ab initio* energy. For molecular dynamics/Monte Carlo simulations involving nonequilibrium configurations, it is important that the fit be reasonably accurate even for extrapolation. The loss of accuracy in GA-Tersoff extrapolations beyond 2.7 Å may be attributed to the fact that the Tersoff functional form in Eq. (1)

TABLE III. Variation of mean square error (MSE) and Tersoff potential function parameter estimates at different stages of GA fitting of *ab initio* energies.

MSE	Stage I	Stage II				Stage III	Stage IV		Stage V
(eV^2)	0.74552	0.71826	0.68913	0.6532	0.57253	0.4693	0.3331	0.28162	0.002567
$A \times 10^3$	1.8345	5.2328	9.7774	2.3203	2.3042	4.8184	5.2688	4.6293	3.5831
$B \times 10^2$	4.3224	5.8889	1.2736	5.8423	1.3211	4.8441	4.2669	3.8348	4.1021
λ	12.1868	17.5864	8.2376	10.3244	19.2320	3.2461	2.3431	2.2423	2.9402
μ	2.7139	5.5381	8.38749	12.2139	2.3149	2.3429	1.2320	2.0831	1.5392
β	0.5875	0.2436	0.1335	0.2201	0.0087	0.0018	0.0003	0.0006	9.38E−05
η	0.18947	0.01201	0.31941	0.2349	0.7923	0.5121	0.6903	0.77428	0.66311
$c \times 10^3$	3.2817	3.1344	2.7214	3.2899	1.1947	1.0265	1.03791	1.0335	1.0686
d	11.9426	31.6264	66.3297	54.2320	19.2322	14.3446	20.2361	22.1385	18.5568
h	0.1294	0.2957	0.21987	0.2342	−0.02923	−0.7038	−0.6613	−0.8343	−0.8205

uses $R=2.7$ Å as the inner cutoff on the bond distance, and it may be possible that the functional form used to attenuate the interaction effects beyond $r>R$ may not adequately capture the actual atomic interactions in 5 atom Si clusters.

As stated earlier, the present study focuses on five-atom clusters because the majority of Si clusters ahead of the tool in a machining simulation experiment consisted of five atoms. Our future work will investigate the parametrization of potential functions for other clusters so that one can have accurate potential surfaces established for various MD simulations of nanometric cutting, chemical mechanical polishing of silicon, etc.

Although the Tersoff functional form with the parameters given in Table III provides an excellent fit to the Si_5 *ab initio* energies obtained from density functional theory (DFT), we would not expect this potential to accurately represent the energies for smaller or larger silicon clusters. The nature of the bonding changes significantly when a silicon atom is removed from an Si_5 cluster to form Si_4. Therefore, we would expect a correspondingly significant change in the potential parameters required to describe smaller clusters. The bonding changes are probably less significant as the size of the cluster is increased. Consequently, we would expect the parameters for the Si_5 system to more accurately describe Si_6 or Si_7 than Si_4.

Also, it may be possible to develop a robust potential for Si_n $(n>3)$ clusters by combining the present GA-NN method with *ab initio* DFT energies to obtain the Tersoff parameters for a series of clusters. Subsequently, these data could be used to train a neural network to permit interpolation of the parameter values for any given cluster size. As n increases, we would expect the interpolated values to approach those for bulk silicon. We are currently investigating this possibility.

V. CONCLUSIONS

A genetic algorithm (GA) approach is presented to fit atomic potentials to a physically meaningful form. A GA-based parametrization process requires a repetitive computation of the objective function, which in this case is expressed as the mean squared error (MSE) taken over all configurations of interest (here, about 28 000 configurations of 5-atom Si clusters were used), between the potentials determined from the *ab initio* calculations and those estimated using the GA-selected parameter vector values. A NN is used to learn the relationship connecting the parameter vector and the ob-

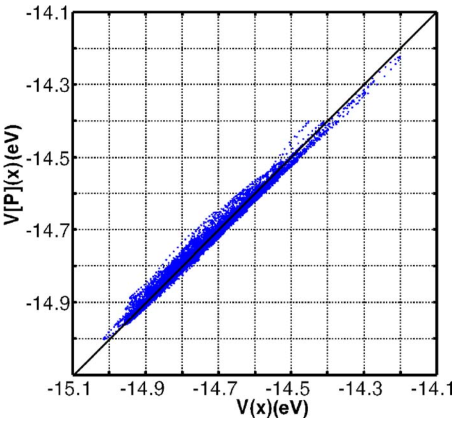

FIG. 12. (Color online) Comparison of the *ab initio* potential computed using Gaussian 03 $V(x)$ and GA-optimized parameters for Tersoff potential $V[P](x)$.

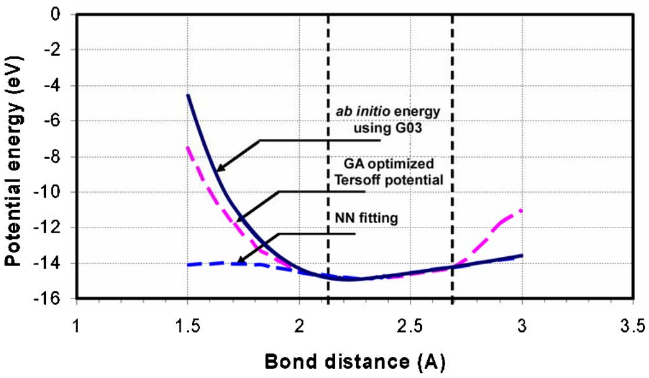

FIG. 13. (Color online) Variation of NN- and GA-Tersoff-estimates of *ab initio* potential with bond distances compared against the variation of actual *ab initio* potential.

jective function. The NN thus trained provides surrogate estimates of the objective function, thereby obviating the need to compute the objective function for every parameter value selected during the GA fitting process. In fact, the use of a NN is the key to reducing the overall computational time for parametrization from a few weeks to a few hours. The GA has been found to quickly establish the correct "range" of the parameter values to optimize a nonlinear and perhaps combinatorial objective function. Since the objective function has significant slope changes and long plateaus, a simplex (Nelder-Mead) search is used during the final stage of the parametrization process to ensure faster convergence to the optimal [P] values. It has been shown that the approach can correctly identify the parameters used to fit Tersoff potential to within ±0.0023 eV (corresponding to a ~95% confidence limit) accuracy. The results also show that that it is possible to fit *ab initio* potentials of 5-atom Si clusters to a Tersoff functional form within ±0.0507 eV accuracy. More importantly, the GA-parametrized fit to Tersoff functional form has excellent extrapolation capabilities compared to the use of NNs.

ACKNOWLEDGMENTS

This project is funded by grants from the National Science Foundation (DMI-0200327, DMI-0428356, and DMI-045766). We thank G. Hazelrigg, J. Cao, and M.L. Realff of the Division of Design, Manufacturing, and Industrial Innovation, B. M. Kramer, Engineering Centers Division, and J. Larsen Basse, Tribology and Surface Engineering program for their interest and support of this work. This project was also sponsored by a DEPSCoR grant on the Multiscale Modeling and Simulation of Material Processing (F49620-03-1-0281). The authors thank Brett Conner, Craig S. Hartley, and Jaimie Tiley, Program Managers of the Metallic Materials Program at AFOSR for their interest in and support of this work. One of the authors (R.K.) also thanks, A. H. Nelson, Jr. Endowed Chair in Engineering for additional support.

*Corresponding author. Email address: ranga@ceat.okstate.edu

[1] L. M. Raff, M. Malshe, M. Hagan, D. I. Doughan, M. G. Rockley, and R. Komanduri, J. Chem. Phys. **122**, 084104 (2005).

[2] L. M. Raff and D. L. Thompson, in *Theory of Chemical Reaction Dynamics Vol. III*, edited by M. Baer (CRC Press, Boca Raton, FL, 1985), p. 1.

[3] A. Rahaman and L. M. Raff, J. Phys. Chem. A **105**, 2147 (2001).

[4] A. Rahaman and L. M. Raff, J. Phys. Chem. A **105**, 2156 (2001).

[5] R. D. Kay and L. M. Raff, J. Phys. Chem. A **101**, 1007 (1997).

[6] L. M. Raff, *Principles of Physical Chemistry* (Prentice Hall, New York, 2001).

[7] R. Komanduri, N. Chandrasekaran, and L. M. Raff, Wear **242**, 60 (2000).

[8] R. Komanduri, N. Chandrasekaran, P. R. Mukund, and L. M. Raff, Wear **240**, 113 (2000).

[9] R. Komanduri, N. Chandrasekaran, and L. M. Raff, Philos. Mag. B **81**, 1989 (2001).

[10] R. Komanduri, L. M. Raff, and A. Chandrasekaran, Philos. Mag. Lett. **82**, 247 (2002).

[11] R. Komanduri, R. Narulkar, and L. M. Raff, Philos. Mag. **84**, 1155 (2004).

[12] P. M. Agrawal, L. M. Raff, and R. Komanduri, Phys. Rev. B **72**, 125206 (2005).

[13] P. M. Agrawal, A. N. A. Samadh, L. M. Raff, M. Hagan, S. T. Bukkapatnam, and R. Komanduri, J. Chem. Phys. **123**, 224711, (2005).

[14] J. A. Pople, J. S. Binkley, and R. Seeger, Int. J. Quantum Chem., Symp. **10**, 1 (1976).

[15] K. Raghavachari, J. A. Pople, E. S. Replogle, and M. Head-Gordon, J. Phys. Chem. **94**, 5579 (1990).

[16] K. Raghavachari and G. W. Trucks, J. Chem. Phys. **91**, 1062 (1989).

[17] C. Moller and M. S. Plesset, Phys. Rev. **46**, 618 (1934).

[18] J. Tersoff, Phys. Rev. Lett. **56**, 632 (1986).

[19] J. Tersoff, Phys. Rev. B **37**, 6991 (1988).

[20] J. Tersoff, Phys. Rev. Lett. **61**, 2879 (1988).

[21] J. Tersoff, Phys. Rev. B **38**, 9902 (1988).

[22] J. Tersoff, Phys. Rev. B **39**, 5566 (1989).

[23] D. W. Brenner, Phys. Rev. B **42**, 9458 (1990).

[24] S. Hobday, R. Smith, and J. BelBruno, Modell. Simul. Mater. Sci. Eng. **7**, 397 (1999).

[25] J. H. Holland, *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence* (MIT, Cambridge, MA, 1975).

[26] F. H. Stillinger and T. A. Weber, Phys. Rev. B **31**, 5262 (1985).

[27] B. C. Bolding and H. C. Andersen, Phys. Rev. B **41**, 10568 (1990).

[28] H. Balamane, T. Halicioglu, and W. A. Tiller, Phys. Rev. B **46**, 2250 (1992).

[29] E. Kaxiras, Comput. Mater. Sci. **6**, 158 (1996).

[30] M. J. Frisch, *et al.*, GAUSSIAN 03 Software (Gaussian, Inc., Pittsburgh, PA, 2003).

[31] D. Buche, N. N. Schraudolph, and P. Koumoutsakos, IEEE Trans. Syst., Man, Cybern., Part B: Cybern. **35**, 183 (2004).

[32] S. Yan and B. Minsker, A Dynamic Meta-Model Approach to Genetic Algorithm Solution of a Risk-Based Groundwater Remediation Design Model, American Society of Civil Engineers (ASCE) Environmental & Water Resources Institute (EWRI) (Philadelphia, PA, 2003) (http://ces.uiuc.edu/emsa/conference/syan-2003—01.pdf).

[33] J.-O. Kim and P. K. Khosla, in *Proceedings of the 1992 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Raleigh, NC, 1992 (Robotics Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, 1992), p. 279.

[34] A. Hacioglu, Aircraft Eng. and Aerospace Technology. Int. Journal 77(5), 369 (2005).

[35] M. T. Hagan, H. B. Demuth, and M. Beale, *Neural Network Design* (PWS, Boston, MA, 1996).

[36] K. S. Tang, K. F. Man, S. Kwong, and Q. He, IEEE Signal Process. Mag. **13**(6), 22 (1996).

[37] T. Back, U. Hammel, and H. P. Schwefel, IEEE Trans. Evol. Comput. **1**(1), 3 (1997).

[38] S. Farritor and J. Zhang, "Using a neural network to determine fitness in genetic design," *ASME Design Engineering Technical Conferences* (2001).

[39] D. Coit and A. Smith, "Using a neural network as a function evaluator during GA search for reliability optimization," *Artificial Neural Networks in Engineering (ANNIE)* (1995).

[40] D. Coit and A. Smith, Comput. Oper. Res. **23**, 515 (1996).

[41] A. R. Burton and T. Vladimirova, "Utilization of an adaptive resonance theory neural network as a genetic algorithm fitness evaluator," *Proceedings of IEEE International Symposium on Information Theory* (1997).

[42] R. Smith, S. Hobday, and J. J. BelBruno, Nucl. Instrum. Methods Phys. Res. B **153**, 247 (1999).

[43] C. L. Bridges and D. E. Goldberg, in *Proc. of the Foundations of Genetic Algorithms (FOGA)*, 301 (1991).

[44] J. A. Nelder and R. Mead, Comput. J. **7**, 308 (1965).