ORIGINAL ARTICLE

A new multi-gene genetic programming approach to nonlinear system modeling. Part I: materials and structural engineering problems

Amir Hossein Gandomi · Amir Hossein Alavi

Abstract This paper presents a new approach for behavioral modeling of structural engineering systems using a promising variant of genetic programming (GP), namely multi-gene genetic programming (MGGP). MGGP effectively combines the model structure selection ability of the standard GP with the parameter estimation power of classical regression to capture the nonlinear interactions. The capabilities of MGGP are illustrated by applying it to the formulation of various complex structural engineering problems. The problems analyzed herein include estimation of: (1) compressive strength of high-performance concrete (2) ultimate pure bending of steel circular tubes, (3) surface roughness in end-milling, and (4) failure modes of beams subjected to patch loads. The derived straight-forward equations are linear combinations of nonlinear transformations of the predictor variables. The validity of MGGP is confirmed by applying the derived models to the parts of the experimental results that are not included in the analyses. The MGGP-based equations can reliably be employed for pre-design purposes. The results of MSGP are found to be more accurate than those of solutions presented in the literature. MGGP does not require simplifying assumptions in developing the models.

Keywords Data mining · Structural engineering · Multi-gene genetic programming · Formulation

A. H. Gandomi (✉)
Department of Civil Engineering,
University of Akron, Akron, OH 44325-3905, USA
e-mail: a.h.gandomi@gmail.com

A. H. Alavi
School of Civil Engineering, Iran University
of Science and Technology, Tehran, Iran
e-mail: ah_alavi@hotmail.com

1 Introduction

Modeling of structural engineering nonlinear systems is a diverse research area where different kinds of methods can be utilized. Due to the large variety of this field, no method can impose itself as the best solution. Estimating both the structure and the parameters of the structural engineering problems makes their modeling process a difficult task. Different criteria for model classification can be characterized while dealing with a system modeling task [1]. A model can be classified as phenomenological or behavioral [2]. A phenomenological model is derived by considering the physical relationships governing the system. As a result, the structure of the model is selected according to the prior knowledge about the system. It is not always possible to design phenomenological models for many of the structural engineering systems because of their complexity. In order to overcome such a problem, the behavioral models are commonly employed. Such models approximate the relationships between the inputs and outputs based on a measured set of data without a need of prior knowledge about the mechanism that produced the experimental data. The behavioral models can provide very good results with a minimal effort [2]. Traditional statistical regression techniques are commonly used for the behavioral modeling purposes. The regression analysis can have large uncertainties. It has major drawbacks for idealization of complex processes, approximation, and averaging widely varying prototype conditions. The regression analysis tries to model the nature of the corresponding problem by a pre-defined linear or nonlinear equation. Another major constraint in application of the regression analysis is the assumption of normality of residuals.

In the case of the behavioral models, several alternative computer-aided pattern-recognition and data-classification

approaches have been developed. The idea is that a pattern-recognition system learns adaptively from experience and extracts various discriminators. Artificial neural networks (ANNs) are the most widely used pattern-recognition procedures. ANNs have been used for a wide range of materials and structural engineering problems [3–7]. Despite the acceptable performance of ANNs in most cases, they do not usually give a definite function to calculate the outcome using the input values. This approach is mostly appropriate to be used as a part of a computer program. However, more robust tools are still required to assess the behavior of many of the structural engineering problems.

Genetic programming (GP) [8] is a new approach that introduces completely new features and traits. GP is an extension of genetic algorithms (GAs). It is a supervised machine learning technique that searches a program space instead of a data space. The programs created by traditional GP are represented as tree structures and expressed using a functional programming language [8, 9]. GP overcomes the shortcomings of different computer-aided and statistical methods previously presented in the literature. The main advantage of the GP-based approaches over the regression and neural network techniques is their ability to generate prediction equations without assuming prior form of the existing relationships. Different strategies have been proposed to improve the classical GP [10]. GP and its variants have been successfully applied to various kinds of materials and structural engineering problems [9, 11–20]. Multi-gene genetic programming (MGGP) [21, 22] is a robust variant of GP. MGGP is designed to generate mathematical models of predictor response data that are "multi-gene" in nature, i.e., linear combinations of low-order nonlinear transformations of the input variables. The traditional GP representation is based on the evaluation of a single tree (model) expression. In multi-gene representation, a single GP individual (program) is constructed from a number of genes, each of which is a tree expression [22]. Despite remarkable prediction capabilities of the MGGP approach [22], applications of this method to civil engineering tasks are conspicuous by their near absence.

This study investigates the feasibility of using MGGP for simulating the complex behavior of the structural engineering systems. The formulation capabilities of the MSGP strategy are demonstrated by applying it to four practical examples of structural engineering. Further, a comparative study is conducted using the results obtained through MGGP and those of different models found in the literature. The MGGP approach overcomes the shortcomings of the traditional and ANN methods. The MGGP models were developed based on reliable experimental results collected from the literature.

2 Genetic programming

Genetic programming is a symbolic optimization technique that creates computer programs to solve a problem using the principle of Darwinian natural selection [9]. The breakthrough in GP then came in the late 1980s with the experiments of Koza [8] on symbolic regression. GP is an extension of genetic algorithms (GAs). This traditional GP technique is also referred to as tree-based GP [8]. The difference between GP and GA is related to the representation of the solution. A string of numbers is created by GA to represent the solution, while the GP solutions are computer programs commonly represented as tree structures. GP is relatively a new field of pattern-recognition methods in contrast to GA. A survey of the literature reveals the growing interest of the research community in GP.

In GP, a random population of individuals (computer programs) is created to achieve high diversity. A population member in GP is a hierarchically structured tree comprising functions and terminals. The functions and terminals are selected from sets of functions and terminals [9]. For instance, function set F can contain the basic arithmetic operations (+, −, ×, /, etc.), Boolean logic functions (AND, OR, NOT, etc.), or any other mathematical functions. The terminal set T contains the arguments for the functions and can consist of numerical constants, logical constants, variables, etc. [9]. The functions and terminals are chosen at random and constructed together to form a computer model in a tree-like structure with a root point with branches extending from each function and ending in a terminal. An example of a simple tree representation of a GP model is illustrated in Fig. 1 [9].

The creation of the initial population is a blind random search for solutions in the large space of possible solutions. Once a population of models is created at random, the GP algorithm evaluates individuals, selects individuals for reproduction, generates new individuals by mutation, crossover, and direct reproduction, and finally creates new generation in all iterations [8, 9]. During the crossover procedure, a point on a branch of each solution (program)
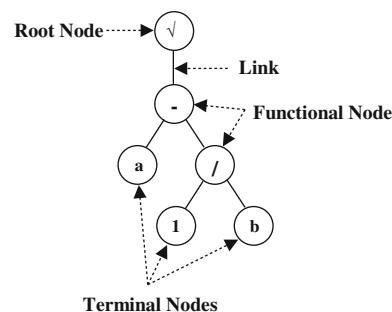
Fig. 1 The tree representation of a GP model ($\sqrt{(a - 1/b)}$)

is selected at random and the set of terminals and/or functions from each program are then swapped to create two new programs (see Fig. 2). The evolutionary process continues by evaluating the fitness of the new population and starting a new round of reproduction and crossover. During this process, the GP algorithm occasionally selects a function or terminal from a model at random and mutates it (see Fig. 3). The best program that appeared in any generation, the best-so-far solution, defines the output of the GP algorithm [8].

2.1 Multi-gene genetic programming

Symbolic regression is typically carried out through the traditional (standard) GP to evolve a population of trees. Each of the trees encodes a mathematical expression. The evolved expressions predict a $(N \times 1)$ output vectors (y) using corresponding $(N \times M)$ matrix of inputs (X) where N is the number of observations of the response variables and M is the number of input variables [22].

Contrary to the traditional GP, each symbolic model (and each member of the GP population) in MGGP is a weighted linear combination of the outputs from a number of GP trees. Each of these trees may be considered to be a "gene". A typical multi-gene model is shown in Fig. 4. This model predicts an output variable using three input variables (x_1, x_2, and x_3). Although this model structure contains nonlinear terms (e.g., tan), it is linear in the parameters with respect to the coefficients d_0, d_1, and d_2. In practice, the maximum allowable number of genes (G_{max}) for a model and the maximum tree depth (D_{max}) any gene may have can be specified by the user. Therefore, a remarkable control over the maximum complexity of the evolved models can be exerted. In particular, enforcing rigid tree depth restrictions (i.e., maximum depths of 4 or 5 nodes) usually results in the evolution of relatively compact models. The evolved models are linear combinations of low-order nonlinear transformations of the predictor variables [22].
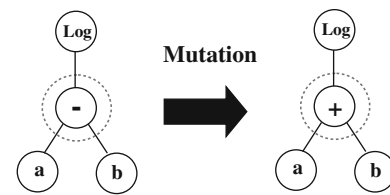

Fig. 3 Typical mutation operation in GP

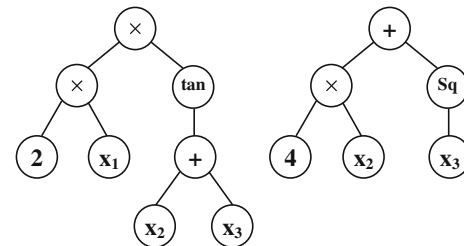$$y = d_0 + d_1(2x_1\tan(x_2+x_3)) + d_2(4x_2 + x_3^2)$$

Fig. 4 A typical multi-gene GP model

For each model, the linear coefficients are derived from the training data using ordinary least squares methods. It has been shown that multi-gene symbolic regression can be more accurate and computationally more efficient than the standard GP for symbolic regression [21, 22]. Also, Searson et al. [21] showed that the multi-gene approach can successfully be embedded within a nonlinear partial least squares algorithm. In MGGP, the initial population is constructed by creating individuals containing GP trees with different genes (between 1 and G_{max}) generated at random. During an MGGP run, genes are acquired and deleted using a tree crossover operator called two-point high-level crossover. This is used in addition to the traditional GP recombination operators and allows the exchange of genes between individuals. If the ith gene in an individual is labeled Gi, then a two-point high-level crossover is performed as shown by the following example. The first parent individual of this example contains three genes (G_1 G_2 G_3) and the second contains four genes (G_4 G_5 G_6 G_7).

Fig. 2 Typical crossover operation in GP
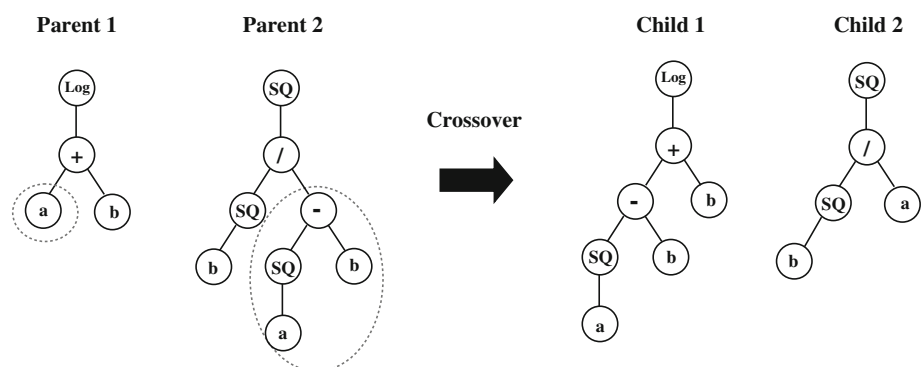
Two randomly selected crossover points are created for each individual. The genes enclosed by the crossover points are denoted by […].

$(G_1[\mathbf{G}_2]G_3)\,(G_4G_5[\mathbf{G}_6\mathbf{G}_7])$.

Thereafter, the genes enclosed by the crossover points are exchanged and the following two new individuals are resulted:

$(G_1G_6G_7G_3)\,(G_4G_5G_2)$.

The acquisition of new genes for both individuals is allowed by two-point high-level crossover. Also, genes are allowed to be removed. If an exchange of genes results in an individual containing more genes than G_{max}, genes are selected at random and deleted until the number of genes in the individual reaches G_{max}. In the MGGP algorithm, standard GP sub-tree crossover is referred to as low-level crossover. In this case, a gene is chosen at random from each parent individual. Then, the standard sub-tree crossover is applied and the created trees replace the parent trees in the otherwise unaltered individual in the next generation. The following are the six methods of mutating trees in MGGP: (1) sub-tree mutation, (2) mutation of constants using an additive Gaussian perturbation, (3) substitution of a randomly selected input node with another randomly selected input node, (4) set a randomly selected constant to zero, (5) substitute a randomly selected constant with another randomly generated constant, and (6) set a randomly selected constant to one [22].

The probabilities of each of the recombinative processes can be set by the user. The processes are grouped into categories called events. Thus, the probability of crossover events, direct reproduction events and mutation events can be specified. These must sum to one. The probabilities of event subtypes can also be specified by the user. For instance, it is possible to define the probability of a two-point high-level crossover taking place once a crossover event has been selected [22].

3 Application to structural engineering problems

This paper introduces the MGGP approach to obtain meaningful nonlinear relationships between various parameters of four practical structural engineering problems. The problems investigated are as follows:

I.   Prediction of compressive strength of high-performance concrete.
II. Evaluation of ultimate pure bending of steel circular tubes.
III. Prediction of surface roughness in end-milling.

IV. Classification of failure modes of beams subjected to patch loads.

Various parameters are involved in the MGGP predictive algorithm. The parameter selection affects the model generalization capability of MGGP. These parameters are selected based on some previously suggested values [22, 23] and after a trial-and-error approach. The parameter settings are shown in Table 1. In this study, basic arithmetic operators and mathematical functions are utilized to get the optimum MGGP models. The number of programs in the population is set by the population size. The number of generation sets the number of levels the algorithm uses before the run terminates. The proper number of population and generation often depends on the complexity of problems and on the number of possible solutions. A fairly large number of population and generations are tested to find models with minimum error. The programs are run until the runs automatically terminated. The maximum allowable number of genes in an individual and the maximum tree depth directly influence the size of the search space and the number of solutions explored within the search space. The success of the MGGP algorithm usually increases with increasing these parameters. In this case, the complexity of the evolved function increases and the speed of the algorithm decreases. The allowable number of genes and tree depth are, respectively, set to optimal values of 8 and 4 as trade-offs between the running time and the

Table 1 Parameter settings for the MGGP algorithm

Parameter	Settings
Function set	$+$, $-$, $\times$, $/$, $\sqrt{\ }$, exp, ln, sin, cos, tanh
Population size	200–500
Number of generations	100–400
Maximum number of genes allowed in an individual	8
Maximum tree depth	4
Tournament size	12
Elitism	0.01% of population
Crossover events	0.85
High-level crossover	0.2
Low-level crossover	0.8
Mutation events	0.1
Sub-tree mutation	0.9
Replacing input terminal with another random terminal	0.05
Gaussian perturbation of randomly selected constant	0.05
Direct reproduction	0.05
Ephemeral random constants	[-10 10]

complexity of the evolved solutions. GPTIPS toolbox [23], in conjunction with subroutines coded in MATLAB, is used to implement MGGP. Fitness function evaluates the evolved expressions to designate the best encoded expressions. The default GPTIPS multi-gene symbolic regression function is used to minimize the root-mean-squared error (RMSE) between the measured and predicted output.

For the analyses, the available data sets are randomly divided into training and validation subsets. The training data are used for learning (genetic evolution). The validation data are used to measure the performance of the programs evolved by MGGP on data that play no role in building the models. In order to obtain a consistent data division, several combinations of the training and testing sets are considered. The selection is such that the maximum, minimum, mean, and standard deviation of parameters were consistent in training and testing data sets. Out of the available data for each problem, approximately 80% of the data are taken for the training process and 20% are used for the validation of the MGGP models. The best models are chosen on the basis of a multi-objective strategy as below:

1. The simplicity of the model, although this is not a predominant factor.
2. Providing the best fitness value on the training set of data.

The first objective can be controlled by the user through the parameter settings (e.g., maximum tree depth or number of genes). Correlation coefficient (R), mean absolute error (MAE), and root-mean-squared error ($RMSE$) are used to evaluate the capabilities of the proposed correlations. R, MAE, and $RMSE$ are given in the form of formulas as follows:

$$R = \frac{\sum_{i=1}^{n} \left(h_i - \overline{h_i} \right) \left(t_i - \overline{t_i} \right)}{\sqrt{\sum_{i=1}^{n} \left(h_i - \overline{h_i} \right) \sum_{i=1}^{n} \left(t_i - \overline{t_i} \right)^2}} \tag{1}$$

$$MAE = \frac{\sum_{i=1}^{n} |h_i - t_i|}{n} \tag{2}$$

$$RMSE] \sqrt{\frac{\sum_{i=1}^{n} |h_i - t_i|^2}{n}} \times 100 \tag{3}$$

where h_i and t_i are respectively the measured and calculated output values for the ith output; $\overline{h_i}$ and $\overline{t_i}$ are respectively the average of the measured and calculated outputs, and n is the number of samples.

For the analysis of the classification problem, the output variable is decoded with a threshold value equal to 0.5. For a more detailed analysis of the classification accuracy of MGGP, its sensitivity, specificity, positive predictivity, and

Table 2 Confusion matrix

		Detected class	
		Class A	Class A
Actual class	Class A	TP	FN
	Class B	FP	TN

accuracy are obtained using (4–7). In general, the classification performance is presented by a confusion matrix as shown in Table 2. In this table, TP, TN, FP, and FN stand for true positive, true negative, false positive, and false negative, respectively. TP and TN are correct classifications, while FP and FN are incorrect classifications.

- True positive (TP): The model predicts that the class is "A" and the class of given instance is indeed "A".
- True negative (TN): The model predicts that the class is "B" and the class of given instance is indeed "B".
- False positive (FP): The model predicts that the class is "A" but the class of the given instance is "B".
- False negative (FN): The model predicts that the class is "B" but the class of the given instance is "A".

$$Sensitivity\,(\%) = \frac{TP}{TP + FN} \times 100 \tag{4}$$

$$Specificity\,(\%) = \frac{TN}{TN + FP} \times 100 \tag{5}$$

$$Positive\,predictivity\,(\%) = \frac{TP}{TP + FP} \times 100 \tag{6}$$

$$Accuracy\,(\%) = \frac{TP + TN}{TP + FP + FN + TN} \times 100 \tag{7}$$

3.1 Problem I: compressive strength of high-performance concrete

High-performance concrete (HPC) is a class of concretes that provides superior performance than the conventional types. HPC is defined by the American Concrete Institute (ACI) as a concrete that meets special combinations of performance and uniformity requirements. The performance characteristics of HPC are major concerns in construction of structural engineering applications. The enhanced performance characteristics of HPC are generally achieved by addition of various cementitious materials and chemical and mineral admixtures to the conventional concrete mix designs. A key property of an HPC mix is its compressive strength. The significance of the compressive strength in concrete technology is obvious. Developing accurate prediction models for the compression strength of HPC leads to saving costs and time and generating a successful concrete mixture [24].

In its current state, behavior modeling of the compressive strength of HPC containing these additives is inherently more difficult than for the concrete without them. In order to provide accurate assessment of the performance characteristics of the HPC mix, the effects of these parameters should also be incorporated into the model development. Therefore, the MGGP approach is utilized to formulate the compressive strength (σ) and the influencing variables as follows:

$$\sigma = f(K, FA, A) \tag{8}$$

where K—ratio of water and superplasticizer summation to binder $((W + S)/B)$, B—binder content $(C + BF + F)$, W (Kg/m^3)—water content, C (Kg/m^3)—cement content, BF (Kg/m^3)—blast furnace slag content, F (Kg/m^3)—fly ash content, S (Kg/m^3)—superplasticizer content, FA (Kg/m^3)—fine aggregate content, A (day)—age of specimens.

The above variables are chosen as the input variables on the basis of an extensive trial study and literature review [24–27]. The significant influence of the employed variables in determining σ is well understood. A reliable database consisting of tests on mixtures with a wide range of aggregate gradation and properties is obtained from the literature [25, 26] to develop the generalized models. The database contains 1,133 test results. The descriptive statistics of the variables used in the model development are given in Table 3. For the analysis, 907 sets (80%) are taken for the training process and the rest of the values (20%) are used for the testing of the generalization capability of the model.

Chen [27] used GP to build an empirical model for σ upon a database of 600 records of HPC. Chen and Wang [28] proposed an incorporating improved grammatical evolution (GE) into the genetic algorithm (GA), called GEGA, to estimate the compressive strength of HPC. Mousavi et al. [24] developed prediction models for the σ of HPC using a hybrid method combining genetic programming with orthogonal least squares (GP/OLS) algorithms. For more verification, the MGGP formulation results are compared with the results of these recent studies.

3.1.1 MGGP-based formulation for compressive strength of HPC mixes

The MGGP-based formulation of the compressive strength (σ) is as given below:

$$\sigma(MPa) = -50.54 + 14.01 \cos(\cos(K)) \ln(FA \times A) \\ - 7.072 \ln(2.171489K) \ln(FA \times A) \tag{9}$$

Comparisons of the MGGP-predicted versus experimental compressive strength of HPC are shown in Fig. 5. As it is seen, the MGGP model is able to predict the target values

with high degree of accuracy. Figure 6 shows the variation of the best (log values) and mean fitness with the number of generations. It can be observed from this figure that the fitness value decreases with increasing the number of generations. The best fitness was found at the 145th generation (fitness = 7.3603). The statistical significance of each of the three genes of the derived model is visualized in Fig. 7. According to Fig. 7a, the weight (coefficients) of the bias term is higher than the other genes. Figure 7b depicts the degree of significance of each gene evaluated using p-values. As it is seen, the contribution of the genes to explain variations in σ is very high, as their relevant p-values are very low and are approximately equal to 0. The statistical significance of the first gene (Gene 1) is lower than the bias term and the second gene. Also, Fig. 8 presents the population of the evolved models in terms of their complexity (number of nodes) as well as their fitness. The generated models that perform relatively well and are much less complex than the best model in the population can be identified in this figure. The best model in the population is highlighted with a red circle. Each green circle represents a model that is not strongly dominated by other models in the population in terms of fitness and model complexity.

Besides, statistical performance of the MGGP, GP [27], GEGA [28], and GP/OLS [24] models, on the entire database, is summarized in Table 4. It can be observed from this table that the MGGP model has produced better results than the GP, GEGA, and GP/OLS-based models.

3.2 Problem II: ultimate pure bending of steel circular tubes

Circular hollow steel tubes have widely been used in large-scale engineering applications due to their good energy absorption characteristics under pure bending. Deformations of the circular tubes under bending show considerable changes to their cross-section profile along the tube length [29, 30]. High nonlinearity in this phenomenon complicates the behavior analysis of the steel circular tubes under pure bending. Precise evaluation of the behavior of the circular tubes under bending using the conventional analytical solutions is not an easy task. Also, the obtained solutions are often difficult to be used in routine design practice.

In this stage, the feasibility of using MGGP for developing accurate prediction models for the ultimate pure bending of the steel circular tubes is investigated. Four predictor variables included in the analysis are the tube thickness (t), tube diameter (d), yield strength of steel (f_y), and modulus of elasticity of steel (E). Therefore, the MGGP formulation of the ultimate pure bending (M_u) is considered to be as follows:

$$M_u = f(t, d, f_y, E) \tag{10}$$

**Table 3** Descriptive statistics of the variables used in the model development (Problem I)

Parameter	W (%)	C (%)	B (%)	F (%)	S (%)	CA (%)	FA (%)	A (day)	σ (MPa)
Mean	0.0785	0.1178	0.0319	0.0270	0.0027	0.4126	0.3295	44.06	35.84
Standard error	0.0003	0.0013	0.0011	0.0009	0.0001	0.0010	0.0010	1.80	0.48
Median	0.0789	0.1148	0.0107	0.0000	0.0028	0.4181	0.3300	28.00	34.67
Mode	0.1023	0.1493	0.0000	0.0000	0.0000	0.4181	0.2665	28.00	33.40
Standard deviation	0.0111	0.0427	0.0361	0.0309	0.0024	0.0325	0.0330	60.44	16.10
Sample variance	0.0001	0.0018	0.0013	0.0010	0.0000	0.0011	0.0011	3,653.15	259.23
Kurtosis	0.0746	−0.5786	−0.5435	−0.8526	1.0700	−0.4130	0.0455	13.81	−0.16
Skewness	0.2561	0.4731	0.7555	0.6297	0.7403	−0.3986	−0.1940	3.47	0.42
Range	0.0608	0.1806	0.1503	0.1127	0.0131	0.1625	0.1662	364.00	80.27
Minimum	0.0514	0.0448	0.0000	0.0000	0.0000	0.3173	0.2480	1.00	2.33
Maximum	0.1122	0.2254	0.1503	0.1127	0.0131	0.4798	0.4141	365.00	82.60

Fig. 5 Predicted versus experimental compressive strength values using the MGGP model: **a** training data, **b** validation data
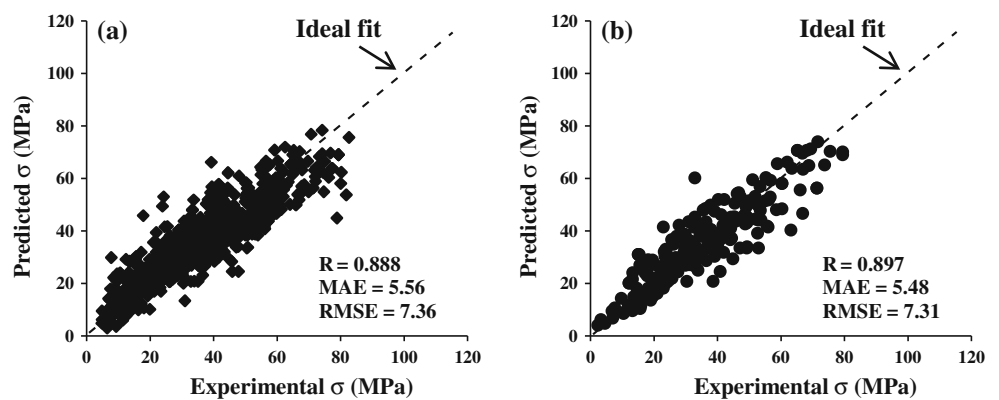

**Fig. 6** Variation of the best and mean fitness with the number of generations
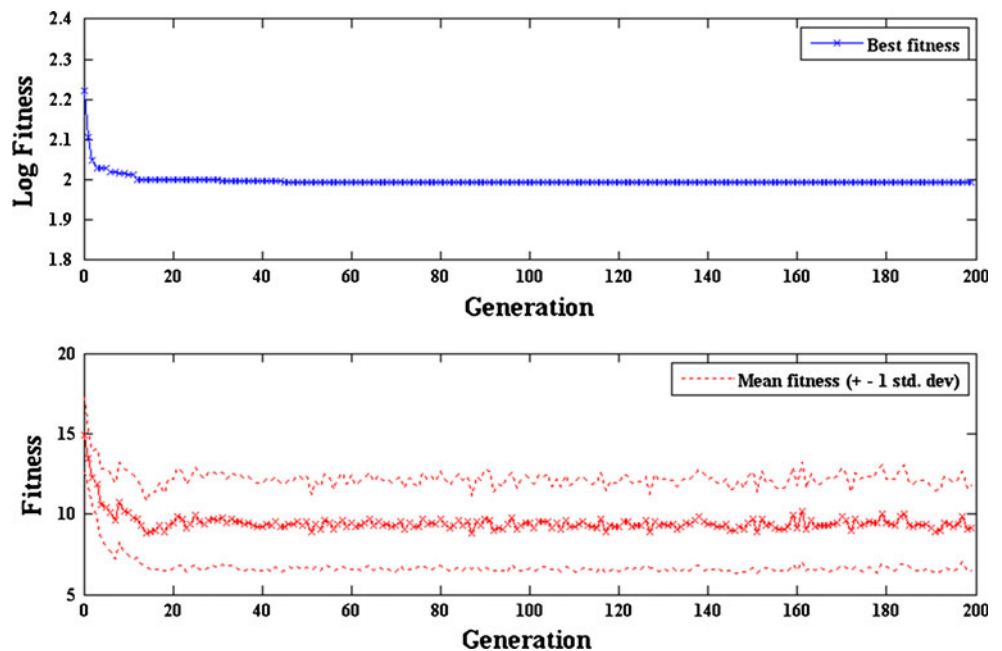
The significant influence of the employed variables in determining M_u is well understood [30–32]. The database used to calibrate and validate the MGGP model is obtained from the literature [30] and comprises a series of 55 pure bending tests conducted on cold-formed tubes. The descriptive statistics of the variables used in the model
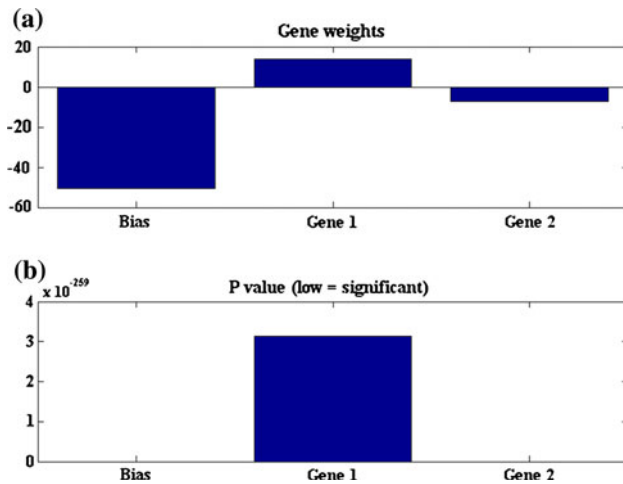
Fig. 7 Statistical properties of the evolved MGGP model (on training data)
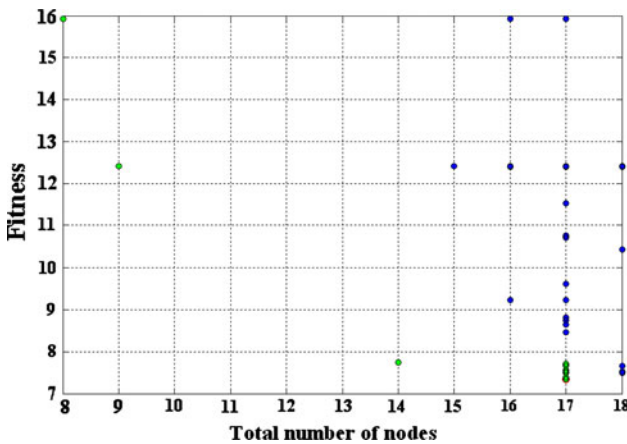
Fig. 8 Population of the evolved models in terms of their complexity and fitness

Table 4 Overall performance of different models for the compressive strength prediction

Model	Performance		
	R	MAE	RMSE
Traditional GP [21]	0.872	6.53	8.44
GEGA [22]	0.878	6.62	8.94
GP/OLS [17]	0.882	5.76	7.59
MGGP	0.890	5.54	7.35

development are given in Table 5. For the analysis, 44 values of the data are taken for the training process and the rest of the values are used for the testing of the generalization capability of the model. Recently, Shahin and Elchalakani [30] developed an ANN prediction model for the M_u of cold-formed tubes. The results obtained by the ANN model were included in the comparative study.

3.2.1 MGGP-based formulation for ultimate pure bending of steel circular tubes

The MGGP-based formulation of the ultimate pure bending (M_u), in terms of t (mm), d (mm), f_y (MPa), and E (MPa), is as given below:

$$
\begin{aligned}
M_u(MPa) = {} & 0.813 \times 10^{-11} + 0.1669 \times 10^{-11} t d^6 \\
& + 0.1431 \times 10^{-6} t^2 d^2 f_y \\
& + 0.3905 \times 10^{-5} d^2 f_y - 0.3905 \times 10^{-5} E \\
& - 0.3403 \times 10^{-7} d^3 f_y
\end{aligned}
\tag{11}
$$

Comparisons of the MGGP-predicted versus experimental ultimate pure bending are shown in Fig. 9. It can be seen that the MGGP model gives precise estimates of the target values. Figure 10 shows the variation of the best (log values) and mean fitness with the number of generations. As can be seen in this figure, the fitness value decreases with increasing the number of generations. The best fitness was found at the 78th generation (fitness = 0.3507). The statistical significance of the genes of the derived model is visualized in Fig. 11. As shown in Fig. 11a, the weight of the bias term is higher than the other genes. Figure 11b depicts the degree of significance of each gene using p-values. With the exception of the bias term (p-values = 0.61), the contribution of the other genes to explain variations in M_u is very high, as their relevant p-values are very low. The population of the evolved models in terms of their complexity and fitness is presented in Fig. 12. The performance statistics of the MGGP and ANN [30] models, on the entire database, are summarized in Table 6. It can be observed from this table that the MGGP model outperforms the ANN-based solution.

3.3 Problem III: surface roughness in end-milling

Milling is an important machining process. Roughness of the milled surface has a considerable influence on the functional properties of the product. Fatigue strength and corrosion resistance are remarkably improved by a high-quality milled surface [33, 34]. The high quality of the surface after milling results in avoiding further machining of the surface. The above facts suggest the necessity of developing precise models for determining the surface roughness. The parameters influencing the surface roughness are mainly divided into controlled and non-controlled cutting parameters. Some of the important controlled parameters are spindle speed, feed rate, and depth of cut. There are also many non-controlled parameters such as the vibrations, tool wear, machine motion errors, and material non-homogeneity of the tool and workpiece. The non-controlled parameters are often difficult to be obtained and whose interactions cannot exactly be determined [35].

Table 5 Descriptive statistics of the variables used in the model development (Problem II)

Parameter	t (mm)	d (mm)	f_y (MPa)	E (MPa)	M_u (kNm)
Mean	2.43	81.37	408.87	200,741.82	5.83
Standard error	0.08	2.70	4.01	1,418.14	0.34
Median	2.52	89.10	408.00	200,000	5.40
Mode	2.52	89.30	365.00	211,000	4.30
Standard deviation	0.60	19.99	29.77	10,517.21	2.49
Sample variance	0.36	399.51	886.08	110,611,737.37	6.19
Kurtosis	0.42	−0.26	−0.18	−0.84	−0.74
Skewness	−0.90	−0.61	0.47	−0.18	0.15
Range	2.45	76.80	108.00	36,000	9.80
Minimum	0.90	33.60	365.00	182,000	0.80
Maximum	3.35	110.40	473.00	218,000	10.60

Survey of the previous surface roughness research reveals that particular efforts devoted to the determination of the most precise model for surface roughness prediction were based on the multiple regression analysis [36].

The current work introduces the MGGP technique to predict the surface roughness in end-milling. Several predictor variables are included in the modeling process. The MGGP-based formulation of the surface roughness, R_a (μm), is considered to be as follows:

$$R_a = f(S, F, D, V) \tag{12}$$

where S (min^{-1})—spindle speed, F (mm/min)—feed rate, D (mm)—depth of cut, and V (μV)—vibrations.

The database used to developing the MGGP model is obtained from the literature [35]. It comprises a total of 156 measurements of the surface roughness. The descriptive statistics of the variables used in the model development are given in Table 7. For the MGGP analysis, 120 sets are taken for the training process and the remaining 36 data sets are used for the testing of the model.

Progress has recently been made in the ability to predict the surface roughness. In this context, Brezocnik et al. [35] obtained an empirical prediction model for the surface roughness utilizing the traditional GP. The predictions made by the derived MGGP model are compared with the traditional GP model. The GP model is developed upon the same database employed herein. For the comparison purposes, the traditional GP-based formulation is given below:

$$
\begin{aligned}
R_a(\mu m) = {} & 2.68327 - \frac{(S - 7.13018F)F}{SV} - 2\frac{D}{S} \\
& + \frac{7.13018F(7.13018 - V + 2S - F - 2D)(7.13018F + D)}{S^2(-S^2 + SF + D)} \\
& - S + \frac{V(V + 14.2604F)}{F(-2S + 8.13018F + 3D)} \\
& + \frac{\left(29.5207F + S(-S^2 + 7.13018F + SF)/(7.13018\,F2 + SD)\right)}{D(7.13018F + 3D)}.
\end{aligned} \tag{13}
$$

3.3.1 MGGP-Based Formulation for Surface Roughness in End-Milling

The prediction equation for the surface roughness (R_a), for the best results by the MGGP algorithm, is as given below:

$$
\begin{aligned}
R_a(\mu m) = {} & 0.02374 + 0.01535F - 0.161 \\
& \times 10^{-5}FV + 0.00001797V \\
& - 0.00003594S - 0.322 \times 10^{-5}SF \\
& + 0.003074D \\
& - 0.4612 \times 10^{-5}(-0.04 + F + D)^2 \tag{14}
\end{aligned}
$$

Comparisons of the MGGP-predicted versus measured surface roughness are shown in Fig. 13. It can be seen from Fig. 13 that prediction accuracy of the MGGP model is very good for both of the training and validation data sets. Figure 14 shows the variation of the best and mean fitness
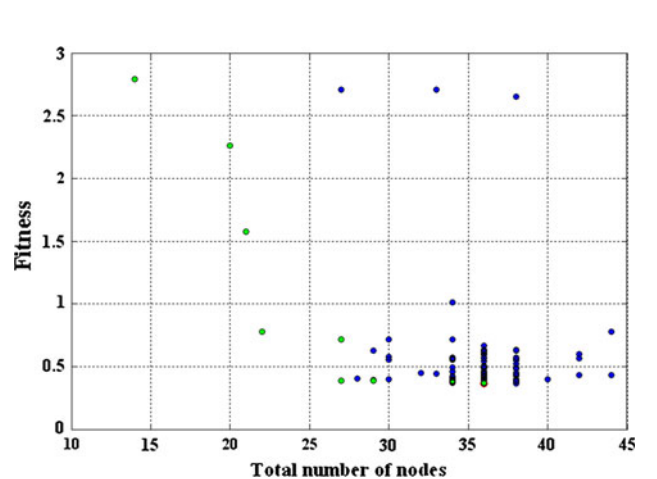
Fig. 9 Predicted versus experimental ultimate pure bending values using the MGGP model: **a** training data, **b** validation data
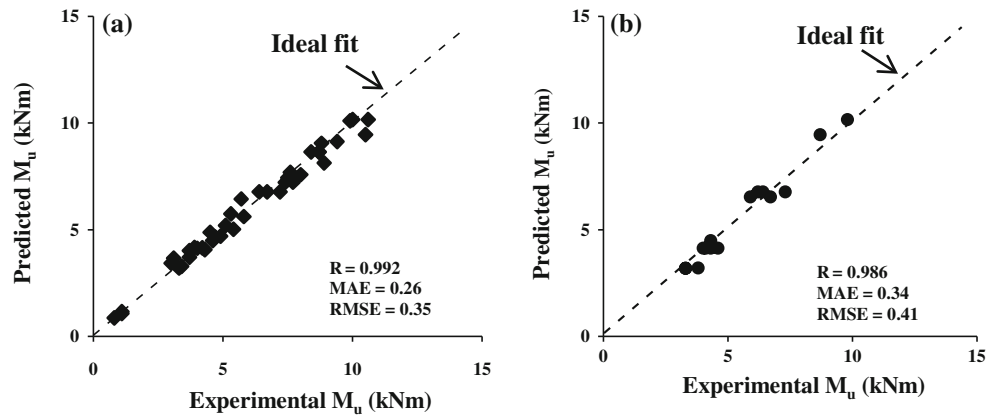
Fig. 10 Variation of the best and mean fitness with the number of generations
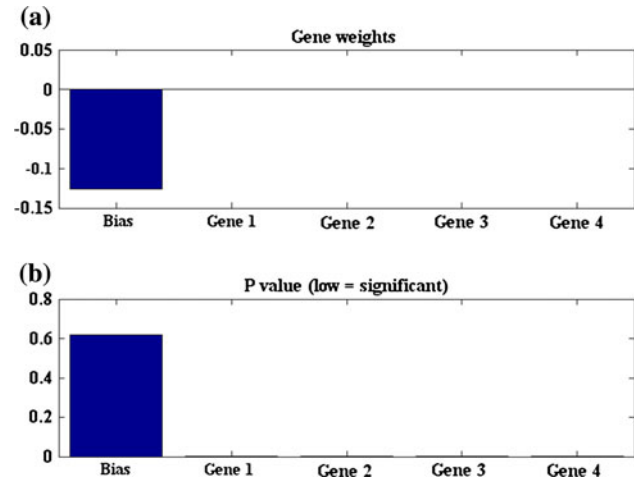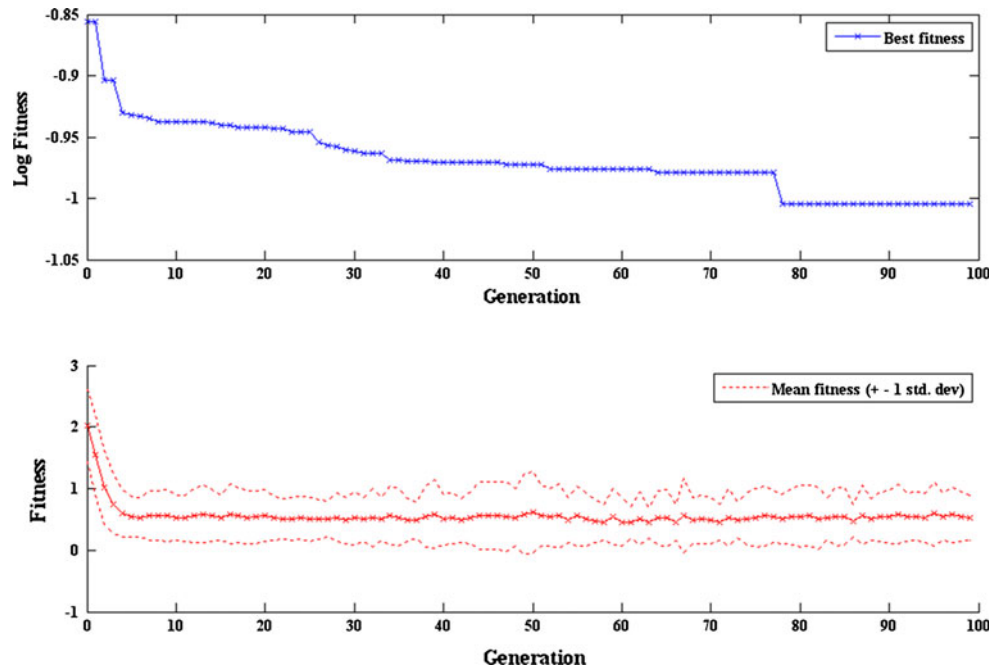
Fig. 11 Statistical properties of the evolved MGGP model (on training data)

Fig. 12 Population of the evolved models in terms of their complexity and fitness

Table 6 Overall performance of different models for the ultimate pure bending prediction

Model	Performance		
	R	MAE	RMSE
ANN [24]	0.988	0.30	0.37
MGGP	0.996	0.29	0.37

values with the number of generations. As illustrated in this figure, the fitness value decreases with increasing the number of generations. The best fitness was found at the 189th generation (fitness = 0.24878). The statistical significance of the evolved genes is visualized in Fig. 15. As shown in Fig. 15a, the weights of the bias term and Gene 2 are higher than the other genes. Figure 15b depicts the degree of significance of the evolved genes. With the exception of the bias term (p-values = 0.33), the other genes have significant contributions to the evaluation of R_a. The population of the evolved models in terms of their complexity and fitness is presented in Fig. 16.

The performance statistics of the MGGP and traditional GP [35] models, on the entire database, are summarized in

Table 8. It can be observed from this table that the MGGP model notably outperforms the GP model. In addition, the MGGP model is significantly short and straightforward in comparison with the more complicated equation provided by GP.

3.4 Problem IV: steel beams patch load behavior

Patch load actions on steel girders are frequently encountered in structural problems such as crane girders and secondary beams reactions. Transversal web stiffeners can be utilized at the location of the concentrated load points to provide the necessary resistance. In the cases where the exact position of the concentrated load is not previously known, the ultimate resistance of unstiffened webs should be evaluated. This is related to the structural behavior of the webs. Developing precise models for the analysis of the structural behaviors associated with the beam collapse is not an easy task due to the participation of several influencing parameters [37]. The failure process of a beam subjected to a patch load can be described in three major phases. The first phase lasts until some yielding is diagnosed in the surface of the web. The second
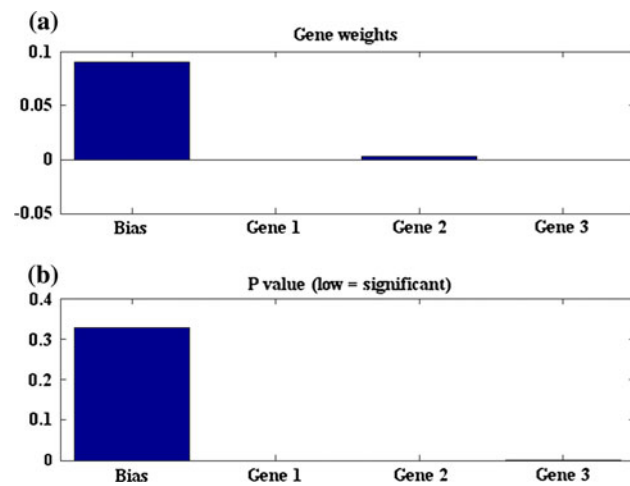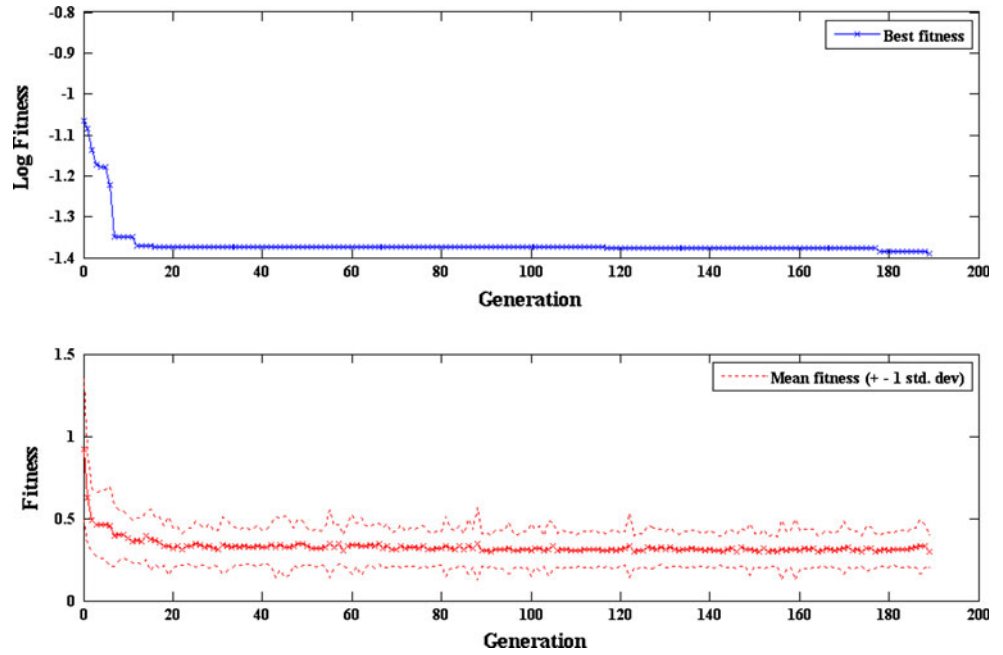
Table 7 Descriptive statistics of the variables used in the model development (Problem III)

Parameter	S (min − 1)	F (mm/min)	D (mm)	V (μV)	R_a (μm)
Mean	1,163.46	380.02	71.64	1,353.36	2.64
Standard error	22.92	13.44	3.31	28.26	0.07
Median	1,250.00	381.00	76.20	1,352.17	2.59
Mode	1,500.00	609.60	25.40	–	3.05
Standard deviation	286.32	167.84	41.36	353.03	0.87
Sample variance	81,978.91	28,169.64	1,710.61	124,628.07	0.76
Kurtosis	−1.40	−1.39	−1.47	−0.59	−0.20
Skewness	−0.18	0.06	0.17	0.21	0.34
Range	750.00	457.20	101.60	1,604.28	3.81
Minimum	750.00	152.40	25.40	638.19	0.94
Maximum	1,500.00	609.60	127.00	2,242.47	4.75

Fig. 13 Predicted versus measured surface roughness values using the MGGP model: **a** training data, **b** validation data

(a) R = 0.963　MAE = 0.19　RMSE = 0.25

(b) R = 0.896　MAE = 0.27　RMSE = 0.33
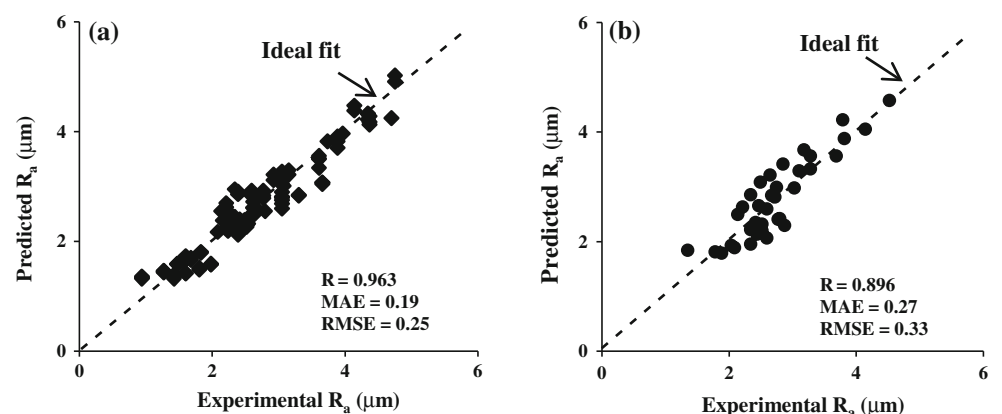
Fig. 14 Variation of the best and mean fitness with the number of generations

Fig. 15 Statistical properties of the evolved MGGP model (on training data)

Fig. 16 Population of the evolved models in terms of their complexity and fitness

step extends from the yielding stage until some small folds can be detected in the web and the last phase goes up to the failure point. Accordingly, the collapse can be classified in three different cases [38, 39]: (1) web yielding, (2) web buckling, and (3) web crippling. Figure 17 shows a schematic representation of the possible failure modes.

In this stage, the robustness of MGGP is demonstrated by applying it to the classification of the failure modes of beams subjected to patch loads. The resistance of a steel beam subjected to a concentrated load depends on different geometrical and physical parameters. The MGGP-based
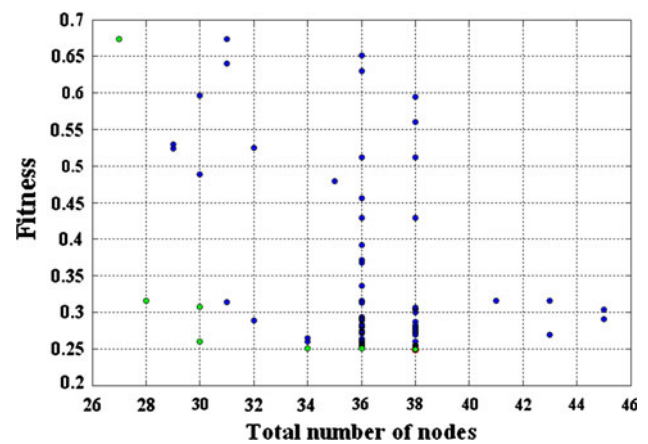
Table 8 Overall performance of different models for the surface roughness prediction

Model	Performance		
	R	MAE	RMSE
Traditional GP [29]	0.937	0.49	0.58
MGGP	0.951	0.21	0.27

formulation of the beam structural collapse form (SC) is considered to be as follows:

$$SC = f\left(b, h, t_w, t_f, b_f, w, F_w, F_f, \beta\right). \tag{15}$$

**Fig. 17** Schematic failure modes for beams subjected to patch loading [33]
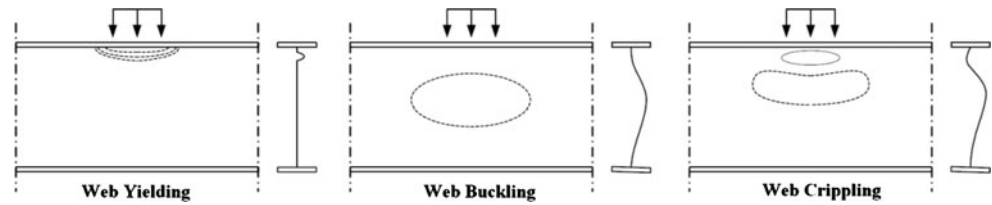
The following values were assigned to the failure modes according to the physical phenomenon associated with each path load case:

1. Web buckling collapse
2. Web crippling collapse
3. Web yielding collapse

The threshold value of the output is equal to 0.5. The following are the geometrical and material properties included in the analysis:

b (mm)—web panel length, h (mm)—web height, t_w (mm)—web thickness, t_f (mm)—flange thickness, b_f (mm)—flange width, w (mm)—applied load length, F_w (MPa)—web yield stress, F_f (MPa)—flange yield stress,
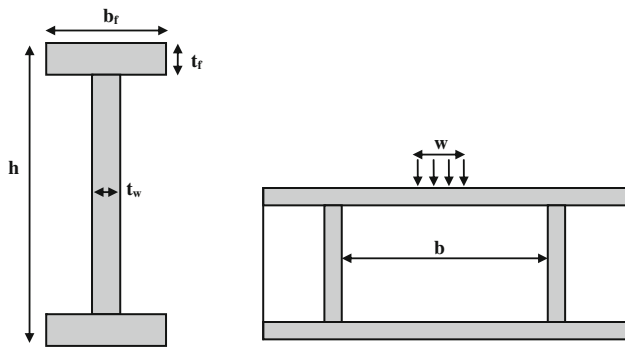
Fig. 18 Geometrical parameters of the steel beams involved in the model development

and β—normalized slenderness parameter $\left(\frac{(w+9t_f)\sqrt{F_w/E}}{t_w}\right)$; E (MPa) is the Young's modulus.

Geometrical parameters of the steel beams involved in the model development are illustrated in Fig. 18. A comprehensive database presented by Fonseca et al. [37] is employed for the model development. The database consists of 251 experimental results for the ultimate load and the collapse-associated physical phenomenon. The descriptive statistics of the variables used in the model development are given in Table 9. Of the available data, 201 sets are taken for the training process and 50 data sets are used for the testing of the model.

3.4.1 MGGP-based formulation for beam structural collapse mode

The MGGP-based empirical relationship to classify the failure modes (SC) of beams subjected to patch loads is as given below:

$$
\begin{aligned}
SC = {}& 1.931 - 0.00002596\,(c - 25.95)\,\big(c + 2F_f + F_w\big) \\
& - 0.8205\,\tanh\big(\tanh(t_f - 6.526769)\big) \\
& + 1.119\,\tanh(t_w - 4.93353) \\
& + 0.01255\,\tanh(t_w - 2.07549)\,(t_w + 2\,c + \beta). \quad (16)
\end{aligned}
$$

When $0.5 \le SC < 1.5$, the case is marked as "buckling collapse mode"; for $1.5 \le SC < 2.5$, the case is classified

Table 9 Descriptive statistics of the variables used in the model development (Problem IV)

Parameter	b (mm)	h (mm)	t_w (mm)	b_f (mm)	t_f (mm)	w (mm)	F_w (MPa)	F_f (MPa)	β	SC
Mean	1,374.23	524.65	5.36	167.58	11.89	67.44	285.35	291.44	1.64	2
Standard error	87.37	17.91	0.21	5.44	0.36	3.90	3.49	2.96	0.08	–
Median	1,000.00	500.00	4.30	150.00	11.20	50.00	280.50	280.00	1.21	2
Mode	600.00	500.00	2.00	100.00	12.00	5.00	250.00	250.00	1.59	3
Standard deviation	1,386.89	284.24	3.35	86.38	5.66	61.87	55.42	46.92	1.21	–
Sample variance	1,923,450.07	80,791.64	11.23	7,461.45	32.08	3,827.73	3,071.23	2,201.78	1.45	–
Kurtosis	17.21	4.82	1.63	3.48	2.45	1.64	−0.99	−0.62	2.38	–
Skewness	3.68	1.74	1.14	1.55	1.26	1.30	0.29	0.63	1.58	–
Range	9,500.00	1,662.00	19.01	456.00	29.25	300.00	235.00	192.00	5.70	2
Minimum	300.00	138.00	0.99	45.00	3.05	0.00	178.00	221.00	0.32	1
Maximum	9,800.00	1,800.00	20.00	501.00	32.30	300.00	413.00	413.00	6.01	3

as "crippling collapse mode", and for $2.5 \leq SC < 3$, the failure mode is marked as "yielding collapse mode". Comparisons of the MGGP-predicted and detected failure modes are shown in Table 10. The results indicate that the developed model is efficiently able to classify the web buckling, crippling, and yielding modes. The performance of the MGGP model on the training data is better than that on the validation data sets. Also, the accuracy of the model for the classification of the web yielding collapse mode is higher than that for the other modes. The overall accuracy of MGGP for correctly classifying the buckling, crippling, and yielding collapse modes is equal to 96.02%.

Figure 19 shows the variation of the best and mean fitness values with the number of generations. As illus-

trated in this figure, the fitness value decreases with increasing the number of generations. The best fitness was found at the 148th generation (fitness = 0.25397). The statistical significance of each of the evolved genes is shown in Fig. 20. As shown in Fig. 20a, the weights of the bias term and Genes 2 and 3 are higher than the other genes. Figure 20b demonstrates the degree of significance of each gene. As it is seen, the p-values for all the genes are approximately equal to 0. Thus, the contribution of the genes in determining SC is very high. The statistical significance of Gene 1 (p-values $= 0.88 \times 10^{-9}$) is slightly lower than the other genes. The population of the evolved models in terms of their complexity and fitness is presented in Fig. 21.

Table 10 The classification accuracies achieved by the MGGP model for the failure behavior evaluation

Samples	Training data			Validation data		
	Class 1 (Buckling)	Class 2 (Crippling)	Class 3 (Yielding)	Class 1 (Buckling)	Class 2 (Crippling)	Class 3 (Yielding)
TP	70	32	91	27	2	16
TN	123	191	102	18	43	29
FP	0	8	0	1	4	0
FN	6	0	2	3	1	1
Sensitivity (%)	92.11	100.00	97.85	90.00	66.67	94.12
Specificity (%)	100.00	95.98	100.00	94.74	91.49	100.00
Positive predictivity (%)	100.00	80.00	98.08	96.43	33.33	96.67
Accuracy (%)	96.98	96.54	98.97	91.84	90.00	97.83

Fig. 19 Variation of the best and mean fitness with the number of generations
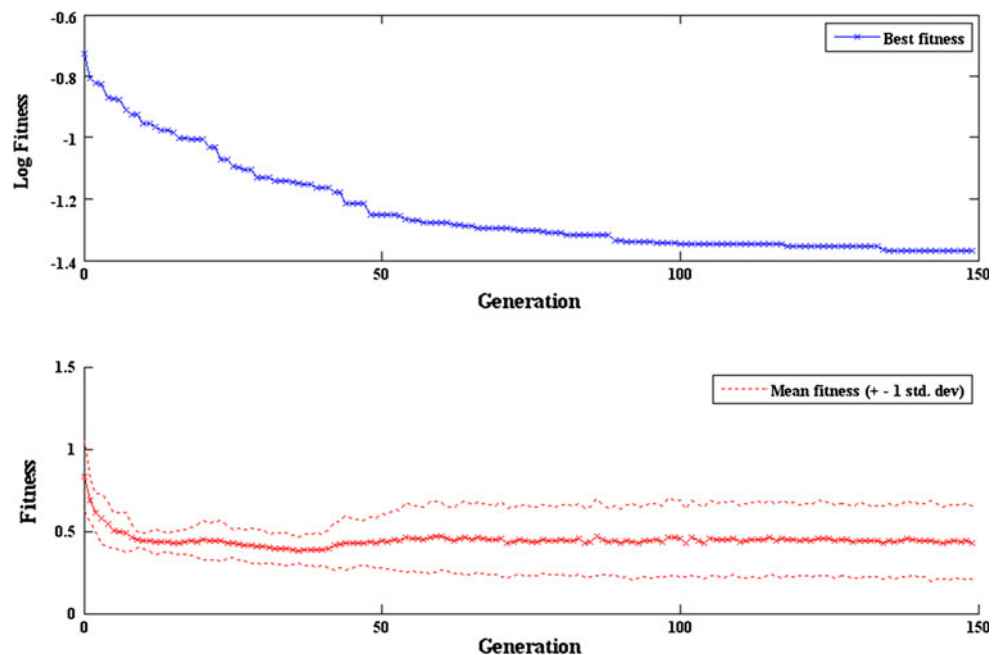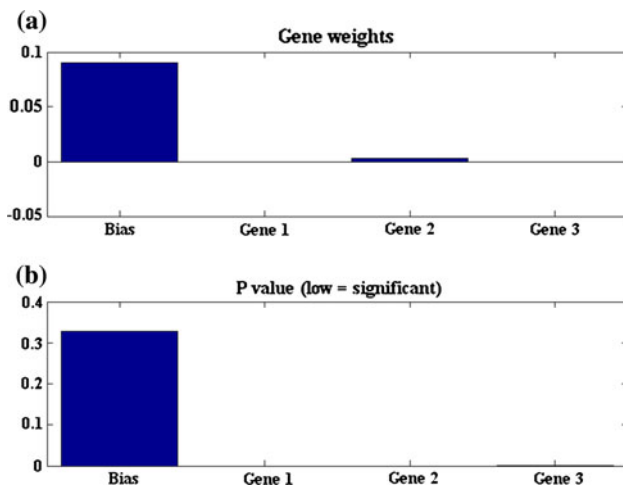
Fig. 20 Statistical properties of the evolved MGGP model (on training data)

4 Performance analysis and model validity

Different prediction equations were obtained for the assessment of four complex structural engineering systems using the MGGP approach. Based on a logical hypothesis [40], if a prediction model gives $R > 0.8$, and the error values (RMSE or MAE) are at the minimum, there is a strong correlation between the predicted and measured values. The model can therefore be judged as very good. It can be observed from Figs. 5, 9 and 13 that the MGGP models (Problems I, II, III) with high R and low RMSE or MAE values are able to predict the target values to an acceptable degree of accuracy. Meanwhile, the MAE and RMSE values are not only low but also as similar as possible for the training and validation sets, which suggests that the proposed models have a very good predictive ability (low values) and generalization performance (similar values) [41]. The results, shown in Table 10 and Fig. 19, for the classification-type problem (Problem IV) indicate that the accuracy of the derived model is very high. As shown in Tables 4, 6, and 8, the MGGP models outperform the ANN, traditional GP, GEGA, and GP/OLS models previously developed for the behavior modeling of the investigated systems.

According to Frank and Todeschini [42], the minimum ratio of the number of objects over the number of selected variables for model acceptability is 3, but often a safer value of 5 is more reasonable. In the present study, the relevant ratios for the problems are much higher. The minimum of this ratio belongs to the ultimate pure bending problem and is equal to $55/4 = 13.75$. Furthermore, new criteria recommended by Golbraikh and Tropsha [43] are checked for external validation of the MGGP prediction models (Problems I, II, III) on the validation data sets. It is suggested that at least one slope of regression lines (k or k') through the origin should be close to 1. Models are considered valid, if they satisfy the required above conditions. The validation criteria and the relevant results obtained by the MGGP models are presented in Table 11. The model validity results ensure that the models derived by means of MGGP are strongly valid, have the prediction power, and are not chance correlations.

The conventional statistical techniques often assume linear, or in some cases nonlinear, relationships between the output and the predictor variables, which is not always true. In most cases, the best models developed using the commonly used statistical approaches are obtained after controlling just some equations established in advance. Thus, they cannot efficiently consider the interactions between the dependent and independent variables.

There are some important differences between MGGP and ANNs. ANNs suffer from some shortcomings including lack of transparency and knowledge extraction. The knowledge extracted by ANNs is stored in a set of weights that cannot be properly interpreted. ANNs require the structure of the network (e.g., transfer functions, number of hidden layers, and neurons) to be identified a priori. Due to the large complexity of the network structure, ANNs do not give a transparent function relating the inputs to the corresponding outputs. The main advantage of MGGP over ANNs is that GP generates a transparent and structured representation of the investigated system [44]. MGGP has a powerful ability to model the mechanical behavior without any prior assumptions. The best solutions (equations) evolved by this technique are determined after controlling numerous preliminary models, even millions of linear and nonlinear models. Furthermore, the models derived using

	Item	Formula	Condition	Problem I compressive strength	Problem II pure bending	Problem III surface roughness
Table 11 Statistical parameters of the MGGP models for the external validation	1	Eq. (1)	$R > 0.8$	0.897	0.986	0.896
	2	$k = \frac{\sum_{i=1}^{n}(h_i \times t_i)}{h_i^2}$	$0.85 < K < 1.15$	1.00	0.98	0.97
h_i—actual output value for the ith output; t_i—predicted output value for the ith output; n—number of sample	3	$k' = \frac{\sum_{i=1}^{n}(h_i \times t_i)}{t_i^2}$	$0.85 < K' < 1.15$	0.97	1.02	1.02
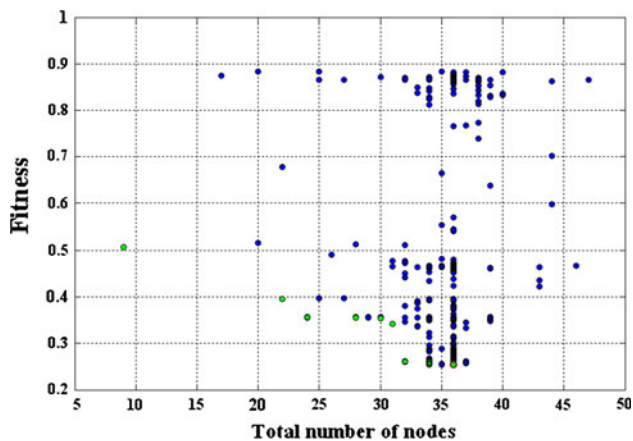
Fig. 21 Population of the evolved models in terms of their complexity and fitness

MGGP are simple and straightforward and can easily be manipulated in practical circumstances. As more data become available, the MGGP models can be improved to make more accurate predictions for a wider range.

However, one of the goals of introducing expert systems into the design processes is better handling of the information in pre-design phase. In the initial steps of design, information about the features and properties of targeted output or process is often imprecise and incomplete [44, 45]. Nevertheless, it is idealistic to have some initial estimates of the outcome before performing any extensive laboratory or field work. The MGGP approach introduced in this research is based on the data alone to determine the structure and parameters of the models. Thus, the models developed using MGGP are considered to be mostly valid for use in preliminary design stages and should cautiously be used for final decision-making. For more reliability, the results of the MGGP-based analyses are suggested to be compared with those obtained using deterministic methods.

5 Conclusions

In this paper, a promising extension of the classical GP, namely MGGP, is employed for the analysis of the structural engineering systems. MGGP integrates the capabilities of the GP and linear regression methods to formulate the nonlinear behavior without any need to predefine the structure of the nonlinear model. The capabilities of the MGGP methodology are illustrated by application to four practical problems. The structural problems considered are the assessment of compressive strength of HPC mixes, ultimate pure bending of circular tubes, surface roughness, and structural collapse modes of beams subjected to patch loads. Reliable databases from the previously published experimental results are used to develop the models.

Several predictive equations were obtained by means of MGGP. Despite high nonlinearity in the behavior of the investigated systems, the proposed MGGP models give reasonable estimations of the target values for both the prediction and the classification-type problems. The validity of the models is tested for a part of test results beyond the training data domain. Furthermore, the MGGP prediction models efficiently satisfy the conditions of different criteria considered for their external validation. The proposed MGGP models efficiently incorporate the effects of several parameters representing the nonlinear behavior of the structural systems. A major advantage of MGGP lies in its powerful ability to generate relatively compact models. The evolved models provide valuable analysis tools for practical pre-planning and pre-design purposes.

References

1. Walter E, Pronzato L (1997) Identification of parametric models from experimental data. Springer, London
2. Metenidis MF, Witczak M, Korbicz J (2004) A novel genetic programming approach to nonlinear system modelling: application to the DAMADICS benchmark problem. Eng Appl Art Int 17:363–370
3. Guzelbey IH, Cevik A, Gogus MT (2006) Prediction of rotation capacity of wide flange beams using neural networks. J Constr Steel Res 62(10):950–961
4. Guzelbey IH, Cevik A, Erklig A (2006) Prediction of web crippling strength of cold-formed steel sheeting using neural networks. J Constr Steel Res 62(10):962–973
5. Gandomi AH, Alavi AH (2011) Applications of computational intelligence in behavior simulation of concrete materials. In: Yang XS, Koziel S (eds) Chapter 9 in Computational optimization and applications in engineering and industry. Springer, Berlin, pp 225–249
6. Guven A (2011) A multi-output descriptive neural network for estimation of scour geometry downstream from hydraulic structures. Adv Eng Softw 42:85–93
7. Pala M (2006) A new formulation for distortional buckling stress in cold-formed steel members. J Constr Steel Res 62:716–772
8. Koza JR (1992) Genetic programming, on the programming of computers by means of natural selection. MIT Press, Cambridge
9. Gandomi AH, Alavi AH, Arjmandi P, Aghaeifar A, Seyednour R (2010) Genetic programming and orthogonal least squares: a hybrid approach to modeling of compressive strength of CFRP-confined concrete cylinders. J Mech Mater Struct 5(5):735–753
10. Gandomi AH, Alavi AH (2011) Multi-stage genetic programming: a new strategy to nonlinear system modeling. Inf Sci. doi:10.1016/j.ins.2011.07.026
11. Gandomi AH, Alavi AH, Yun GJ (2011) Nonlinear modeling of shear strength of SFRC beams using linear genetic programming. Struct Eng Mech 38(1):1–25
12. Cevik A, Göğüs MT, Güzelbey IH, Filiz H (2010) Soft computing based formulation for strength enhancement of CFRP confined concrete cylinders. Adv Eng Softw 41:527–536
13. Gandomi AH, Alavi AH, Yun GJ (2011) Formulation of uplift capacity of suction caissons using multi expression programming. KSCE J Civil Eng 15(2):363–373

14. Gandomi AH, Alavi AH, Mirzahosseini MR, Moqaddas NF (2011) Nonlinear genetic-based models for prediction of flow number of asphalt mixtures. J Mater Civil Eng ASCE 23(3):248–263

15. Gandomi AH, Tabatabaei SM, Moradian MH, Radfar A, Alavi AH (2011) A new prediction model for the load capacity of castellated steel beams. J Construct Steel Res 67(7):1096–1105

16. Cevik A (2007) A new formulation for longitudinally stiffened webs subjected to patch loading. J Construct Steel Res 63:1328–1340

17. Cevik A, Sonebi M (2008) Modelling the performance of self-compacting SIFCON of cement slurries using genetic programming technique. Comput Concr 5:475–490

18. Cevik A, Guzelbey IH (2007) A soft computing based approach for the prediction of ultimate strength of metal plates in compression. Eng Struct 29:383–394

19. Cevik A (2007) A new formulation for web crippling strength of cold-formed steel sheeting using genetic programming. J Construct Steel Res 63:867–883

20. Cevik A (2007) Genetic programming based formulation of rotation capacity of wide flange beams. J Construct Steel Res 63:884–893

21. Searson DP, Willis MJ, Montague GA (2007) Co-evolution of nonlinear PLS model components. J Chemometr 2:592–603

22. Searson DP, Leahy DE, Willis MJ (2010) GPTIPS: an open source genetic programming toolbox for multigene symbolic regression. Proceedings of international multi conference on engineering computer science, Hong Kong

23. Searson DP (2009) GPTIPS: genetic programming and symbolic regression for MATLAB

24. Mousavi M, Gandomi AH, Alavi AH, Vesali M (2010) Modeling of compressive strength of HPC mixes using a combined algorithm of genetic programming and orthogonal least squares. Struct Eng Mech 36(2):225–241

25. Yeh IC (2006) Analysis of strength of concrete using design of experiments and neural networks. J Mater Civil Eng 18(4):597–604

26. Yeh IC, Lien L (2009) Knowledge discovery of concrete material using genetic operation trees. Expert Syst Appl 36:5807–5812

27. Chen L (2003) A study of applying macro evolutionary genetic programming to concrete strength estimation. J Comput Civ Eng 17:290–294

28. Chen L, Wang T (2010) Modeling strength of high-performance concrete using an improved grammatical evolution combined with macro genetic algorithm. J Comput Civil Eng 24:281–288

29. Calladine CR (1988) Theory of shell structures. Cambridge University Press, Cambridge

30. Shahin M, Elchalakani M (2008) Neural networks for modelling ultimate pure bending of steel circular tubes. J Constr Steel Res 64:624–633

31. Sherman DR (1986) Inelastic flexural buckling of cylinders, "steel structures". Proceedings of steel structural conference on: record advances in application design, Budva, pp 339–357

32. Sherman DR (1976) Tests of circular tubes in bending. J Struct Eng 102(ST11):2181–2195

33. Lou SM (1997) Development of four in-process surface recognition systems to predict surface roughness in end milling. Ph D thesis, Iowa State University, Iowa

34. Lou SM, Chen CJ, Li MC (1998) Surface roughness prediction technique for CNC end-milling. J Ind Technol 15(1):1–6

35. Brezocnik M, Kovacic M, Ficko M (2004) Prediction of surface roughness with genetic programming. J Mater Proc Tech 157–158:28–36

36. Huang L, Chen JC (2001) A multiple regression model to predict inprocess surface roughness in turning operation via accelerometer. J Ind Technol 17(2):1–8

37. Fonseca ET, Vellasco MMBR, Vellasco PCGS, De Andrade SAL, Pacheco MAC (2008) A neuro-fuzzy evaluation of steel beams patch load behavior. Adv Eng Softw 39:558–572

38. Bergfelt A (1979) Patch loading on slender web. Influence of horizontal and vertical web stiffeners on the load carrying capacity, S79:1. Chalmers University of Technology Publication, Goteborg, pp 1–143

39. Gozzi J (2007) Patch loading resistance of plated girders—ultimate and serviceability limit state. Doctoral Thesis, Lulea University of Technology

40. Smith GN (1986) Probability and statistics in civil engineering. Collins, London

41. Pan Y, Jiang J, Wang R, Cao H, Cui Y (2009) A novel QSPR model for prediction of lower flammability limits of organic compounds based on support vector machine. J Haz Mater 68:962–969

42. Frank IE, Todeschini R (2002) The data analysis handbook. Elsevier, Amsterdam, The Netherland, 1994

43. Golbraikh A, Tropsha A (2001) Beware of q2. J Mol Graph Model 20:269–276

44. Alavi AH, Ameri M, Gandomi AH, Mirzahosseini MR (2011) Formulation of flow number of asphalt mixes using a hybrid computational method. Constr Build Mater 25:1338–1355

45. Kraslawski A, Pedrycz W, Nyström L (1999) Fuzzy neural network as instance generator for case-based reasoning system: an example of selection of heat exchange equipment in mixing. Neural Comput Appl 8(2):106–113