ESTIMATING STRENGTH OF RUBBERIZED CONCRETE USING EVOLUTIONARY MULTIVARIATE ADAPTIVE REGRESSION SPLINES

Min-Yuan CHENG, Minh-Tu CAO

Civil and Construction Engineering Department, National Taiwan University of Science and Technology, 43 Keelung Road, Section 4, Taipei, Taiwan

Abstract. This study proposes an artificial intelligence (AI) model to predict the compressive strength and splitting tensile strength of rubberized concrete. This Evolutionary Multivariate Adaptive Regression Splines (EMARS) model is a hybrid of the Multivariate Adaptive Regression Splines (MARS) and Artificial Bee Colony (ABC) within which MARS addresses learning and curve fitting and ABC implements optimization to determine the fittest parameter settings with minimal prediction error. *K*-fold cross validation was utilized to compare EMARS performance against four other benchmark data mining techniques including MARS, Back-propagation Neural Network (BPNN), Radial Basis Function Neural Network (RBFNN), and Genetic Programming (GP). Comparison results showed EMARS to be the best model for predicting rubberized concrete strength and study results demonstrated EMARS as a reliable tool for civil engineers in the concrete construction industry.

Keywords: multivariate adaptive regression splines, artificial intelligence, artificial bee colony, rubberized concrete, concrete strength, silica fume.

Introduction

There has been increased research interest in rubberized concrete in recent years. Rubberized concrete offers several benefits, including: 1) Environmental Friendliness. Traditional concrete aggregate is partially replaced by rubber from waste tires, a major source of environmental damage worldwide (Gesoğlu *et al.* 2010; Son *et al.* 2011); 2) Economy. Waste tires are relatively inexpensive and widely available. Waste tires have previously been considered as a partial replacement for coarse aggregate in mortar mixtures (Topçu, Sarıdemir 2008; Zheng *et al.* 2008) and Khaloo *et al.* (2008) stated that using rubber aggregate can provide a 25% volume benefit; and 3) Technological Advancement. The presence of rubber in concrete results in higher resilience, durability, and elasticity (Topçu, Avcular 1997). Rubberized concrete was found to provide good aesthetics, acceptable workability, and a smaller weight per volume than standard concrete.

In practice, rubberized concrete has been used in various applications. It is suitable for architectural applications that require light unit weight (e.g. nailing concrete, false façades, stone backing, interior construction), low-strength-concrete applications (e.g. sidewalks, driveways, and certain road construction applications), and/or high plastic-energy absorption (e.g. shock absorbers, crash barriers around bridges).

Using rubber in concrete reduces concrete strength (Khatib, Bayomy 1999). However, adding silica fume can diminish this reduction because silica fume increases the homogeneity and decreases the overall number of large pores in the cement paste (Güneyisi *et al.* 2004). Adding silica fume also results in denser interfaces between the cement paste and coarse aggregates (Bentur, Cohen 1987). In sum, while enhancing certain concrete properties, using rubber and silica fume in concrete significantly complicates the process of modeling concrete strength due to the potentially nonlinear mapping relationships between concrete ingredients and its strengths (Yeh 1998).

Clearly, a wide experience-based knowledge of the concrete composition versus strength relationship can help civil engineers understand the nature of concrete and how to optimize concrete mixtures (Yeh 1998). To minimize time and construction costs, accurate prediction of rubberized concrete strength is essential to pre-design and quality control success in real project applications. Also, determining the level of importance of each input variable is essential to adjusting the mix proportion in order to obtain the required strength and to providing civil engineers insight into the impact of input variables on the concrete strength output value.

To fulfill this challenging task, this study proposes a new novel artificial intelligence (AI) model that hybrid-

Corresponding author: Minh-Tu Cao
E-mail: *caominhtu2201@gmail.com*

Taylor & Francis
Taylor & Francis Group

izes Multivariate Adaptive Regression Splines (MARS) and Artificial Bee Colony (ABC). A number of studies have demonstrated that performances obtained by hybridized AI techniques are superior to those achieved by employing a single technique (Yang, Yau 2000). Cheng and Wu (2009) successfully built hybrid AI techniques and effectively applied them in the construction industry; Sánchez-Lasheras *et al.* (2012) integrated self-organizing map and MARS to accurately forecast corporate bankruptcy; and Lee and Chen (2005) used a combination of artificial neural networks and MARS to estimate credit scores.

MARS is a nonlinear and non-parametric regression methodology (Friedman 1991). One of MARS' greatest advantages is its ability to explore the complex nonlinear relationships between a response variable and predictor variables by fitting data into a series of predictor variable spline functions. MARS has been demonstrated particularly effective in handling prediction problems and been successfully employed in credit scoring (Lee *et al.* 2006), computer wholesaling (Lu *et al.* 2012), paper manufacturing (García Nieto *et al.* 2012), public water supply issues (Vidoli 2011), and engineering software (Zhou, Leung 2007). Many studies have further demonstrated the superiority of MARS over other data techniques (Leathwick *et al.* 2006; Samui 2012). However, to the best knowledge of the authors, MARS has not yet been applied to problems in the concrete industry.

In order to build a MARS model, users must select the following tuning parameters: maximum number of basis function M_{max}, penalty parameter (smooth parameter) d and maximum interaction between variables *mi*. It is worth noticing that these parameters control model complexity and generalization. Andalib and Atry (2009) stated that these parameters are important features to building a MARS model. Therefore, obtaining the optimal set of parameters for MARS is a crucial task; as proper parameter settings can help guarantee MARS prediction accuracy. Friedman's suggestions of parameter selection provides a large value ranges and is dependent on the data set at hand (Friedman 1991).

To overcome this drawback of MARS, our paper used ABC (Karaboga 2005) algorithm as a search engine to determine optimal MARS parameter values. ABC was introduced by Karaboga in 2005 and is a swarm intelligence-based optimization algorithm inspired by honeybee foraging behavior. The ability of ABC to find optimal solutions for large-scale problems was demonstrated in various works (Karaboga, Akay 2009; Li *et al.* 2010). Moreover, ABC is also a reliable tool when incorporated with other data mining techniques (Hong 2011). Therefore, ABC is a potentially suitable search engine for identifying suitable parameters of MARS, including M_{max}, d and *mi*.

Hence, the objective of this research was to develop and test the Evolutionary Multivariate Adaptive Regression Splines (EMARS). The authors created EMARS by fusing MARS and ABC in a manner that incorporated the strengths and avoided the weaknesses of each technique. This newly proposed model operates automatically without human intervention and accurately predicts rubberized concrete strength using different ingredient ratios. This study then compared the performance of EMARS against four other benchmark data mining techniques, including MARS, Back-propagation Neural Network (BPNN), Radial Basis Function Neural Network (RBFNN), and Genetic Programming (GP).

The remainder of this paper is organized as follows: the second section reviews related research works; the third introduces the EMARS model; the fourth describes data collection; the fifth validates the EMARS model compares simulations results; and the last presents conclusions.

1. Literature review

1.1. Previous works

There is increasing research interest in applying AI techniques to problems related to rubberized concrete. Numerous studies have proposed approaches for modeling rubberized concrete strength and mechanical properties. Eldin and Senouci (1994) first used Artificial Neural Network (ANN) to estimate the strength of rubberized concrete when rubber aggregate was used in place of coarse aggregate in the control mix and sand in the control mix.

Topçu and Saridemir (2008) recently applied ANN and fuzzy logic (FL) to predict the properties of rubberized concrete. They modeled rubberized concrete properties using input variables including cement, sand, water, fine crushed stone, coarse crushed stone, fine rubber, coarse rubber and output variables including the unit weight and flow table of fresh concrete (Topçu, Sarıdemir 2008). Gesoğlu *et al.* (2010) established a mathematical formula using ANN and Genetic Programming (GP) to model rubberized concrete properties and measure the influence of silica fume on rubberized concrete properties. More recently, Abdollahzadeh *et al.* (2011) used ANN to predict the compressive strength of rubberized concrete. However, their dataset of only 20 instances is likely inadequate to accurately assess model performance.

Previous studies have generally applied similar ANN techniques with only minor modifications and some traditional regression techniques. This may represent a shortcoming of applying AI to the rubberized concrete field. A significant disadvantage of ANN is the need to select a large number of controlling parameters, e.g. number of hidden layers, number of neurons in hidden layers, learning rate, momentum to construct network (Samarasinghe 2006). This means that the ANN training process must be obtained via a gradient descent algorithm on the error space, which can be very complex and may contain many local solutions that may prevent an ANN model from converging on an optimal solution (Kiranyaz *et al.* 2009).

Additionally, ANN does not represent an explicit relationship between input and output parameters or even

verify the importance level of each input variable. For GP, this method does not perform a proper numerical estimation of model parameters (constants/coefficients). Essentially, it tends to yield functions that grow in length during evolutionary search (Davidson *et al.* 2003), potentially making the final model unnecessarily large and difficult to interpret.

1.2. Multivariate adaptive regression splines

MARS was first proposed by Friedman (1991) as a flexible procedure to organize relationships that are nearly adaptive or involve interactions with fewer variables. MARS makes no assumptions about the underlying functional relationship between dependent and independent variables in order to estimate the general functions of high-dimensional arguments given sparse data (Friedman 1991; Samui 2012). One further advantage of MARS is its ability to estimate the contributions of basis functions so that the additive and interactive effects of predictors are allowed to determine the response variable.

MARS is established by fitting a basis function (term) to distinct independent variable intervals. In general, splines (also called piecewise polynomials) have pieces that connect smoothly together. The interface points between pieces are called knots, denoted as t. MARS uses two-sided truncated power functions as spline basis functions, described in Eqns (1) and (2), Figure 1 ($q = 1$; $t = 0.5$) provides an illustration:

$$[-(x-t)]_+^q = \begin{cases} (t-x)^q & if\, x < t \\ 0 & otherwise \end{cases};\quad (1)$$

$$[+(x-t)]_+^q = \begin{cases} (t-x)^q & if\, x \geq t \\ 0 & otherwise \end{cases},\quad (2)$$

where: ($q \geq 0$) is the power to which splines are raised and determines the degree of smoothness of the resultant function estimate; $q = 1$ is the case used in this study; $[]_+$ ensures values are positive.

Interaction basis functions are produced by multiplying an existing term with a truncated linear function involving a new variable. In this case, both the existing term and newly produced interaction basis function are used to establish the MARS model. Setting maximum user order can restrict the search for new basis functions. Formulae for the interaction basis function and the general MARS function can be represented in Eqns (3) and (4), respectively:

$$B_m(x) = \prod_{j=1}^{K_m} [s_{K_j^m} \times (x_{K_j^m} - t_{K_j^m})]_+;\quad (3)$$

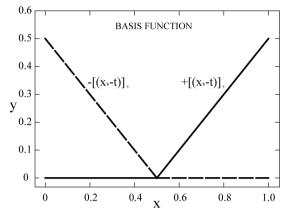$$\hat{y} = \hat{f}_M(x) = c_o + \sum_{m=1}^{M} c_m B_m(x),\quad (4)$$

Fig. 1. Basis function

where: Km is the number of truncated linear functions multiplied in the mth basis function. Km must be not larger than the maximum interaction among variables mi. Thus, the user must specify mi prior to establishing the MARS model. $x_{k_j^m}$ is the input variable corresponding to the jth truncated linear function in the mth term; $t_{K_j^m}$ is the knot value corresponding to variable $x_{K_j^m}$; $S_{K_j^m}$ is the selected sign +1 or −1; $\hat{y}$ is the dependent variable predicted by the MARS model; c_0 is a constant;

$B_m(x)$ is the mth basis function, which may be a single spline basis function; and c_m is the coefficient of the mth basis function.

The final MARS model was constructed using a two-stage process comprising a forward phase and backward phase. The forward phase started with the basis function $h_0(x) = 1$. Knots were then chosen automatically. Candidate knots were placed at random positions within the range of each predictor variable to define a pair of basis functions. At each step, the model adopted the knot and its corresponding pair of basis functions to give the maximum reduction in sum-of-squares residual error. This process of adding basis functions continued until the maximum number of basis functions M_{max} was reached (Friedman 1991). M_{max} is set by the user as referenced in Friedman (1991). The forward phase selection of the basis function leads to a very complex and overfitted model. Although this model has poor predictive abilities for new data (testing data), it fits the training data well.

To improve model predictive power, the authors employed a backward phase to delete the redundant basis functions that made the least contributions. Generalized cross-validation (GCV) was used as the deletion criterion (Sekulic, Kowalski 1992). GCV is the mean-squared residual error divided by a penalty that is dependent on model complexity and defined as follows:

$$LOF(\hat{f}_M) = GCV(M) = \frac{1}{n} \times \frac{\sum_{i=1}^{n} [y_i - \hat{f}_M(x_i)]^2}{(1 - C(M)/n)^2},\quad (5)$$

where n is the number of data cases. $C(M)$ is a complexity penalty that increases with number of basis functions in the model, defined as:

$$C(M) = (M+1) + d \times M , \qquad (6)$$

where M is the number of basis functions in Eqn (4) and parameter d is a penalty for each basis function included in the model. d may be also regarded as a smoothing parameter. A small d generates a large model with more basis functions, while a big d creates a small, smooth model with fewer basis functions (Sánchez-Lasheras *et al.* 2012). Reference (Friedman 1991) provides further suggestions on choosing the value of d and the influence of d on the final model.

1.3. Artificial bee colony algorithm

The Artificial Bee Colony (ABC) algorithm is a recently developed population-based optimization algorithm now widely used to solve multidimensional optimization problems. Karaboga first proposed the ABC concept in 2005 (Karaboga 2005). Its swarm intelligence-based optimization algorithm was inspired by honeybee foraging behaviour. ABC incorporates three kinds of honeybees, namely employed bees, onlooker bees, and scout bees. Figure 2 shows the ABC algorithm flowchart.
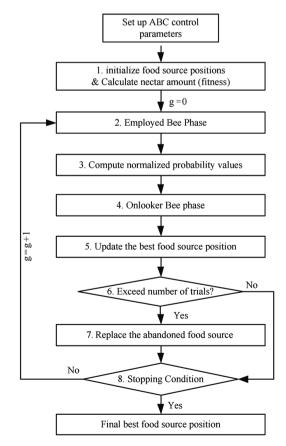

Fig. 2. ABC model

1.3.1. Initial population

ABC commences the search process by randomly generating an NF number of food source positions. NF does not change during the optimization process in the standard ABC algorithm. Eqn (7) was used as our initial population generator. Amount of nectar at each food source position was calculated after all food source positions were generated:

$$x_{i,0} = x_{i,\min} + rand[0,1] \times (x_{i,\max} - x_{i,\min}) , \qquad (7)$$

where: $x_{i,\min}$ and $x_{i,\max}$ indicates the minimum and maximum of the ith decision variable; $rand[0,1]$ denotes a uniformly distributed random number between 0 and 1; and $x_{i,0}$ is the ith decision variable (food source position) in the initial population.

1.3.2. Employed bee phase

Each employed bee chooses a new candidate food source position to update feasible solutions based on the neighbourhood of the previously selected food source. A candidate solution $u_{i,j}$ can thus be generated from the old solution $x_{i,j}$ as in Eqn (8). Employed bees are created from the strength of previously discovered sources.

$$u_{i,j} = x_{i,j} + \phi_{i,j}(x_{i,j} - x_{k,j}) , \qquad (8)$$

where: $\{k,i\} \in \{1,2,\ldots,NF\}$ and $j \in \{1,2,\ldots,D\}$ are randomly chosen indices, k must be different from i, and $\phi_{i,j}$ is a random number in the range $[-1,1]$.

Amount of nectar is used to compare the candidate solution (candidate food source position) with the old solution. The candidate solution will replace the old solution if its food source quality is equal to or better than the latter; otherwise, the old solution is retained. Food source information is shared with onlooker bees when employed bees return to their hive.

1.3.3. Probability calculation

The information sharing stage of the ABC algorithm generates collective intelligence. Probability value influences the behaviour of onlooker bees, which select food sources based on probability. Probability value is calculated as:

$$p_i = \frac{fit_i}{\sum\limits_{j=1}^{NF} fit_j} , \qquad (9)$$

where fit_i is the fitness value of the ith food source, which is proportional to the nectar amount of the food source in position i and given by Eqn (10):

$$fit = \begin{cases} 1/(1+f) & f \geq 0, \\ 1+abs(f) & f < 0, \end{cases} \qquad (10)$$

where: f_i is the objective function value of the ith food source; *abs* is absolute value.

1.3.4. Onlooker bee phase

In this stage, a random real number within the range [0,1] is generated for each food source. As in the employed bee phase, if the probability value p_i associated with that source is greater than this random value then the onlooker bee uses Eqn (8) to create a new candidate food source position. After evaluating the source, greedy selection is applied and the onlooker bee either updates the new position by removing or retaining the old solution.

1.3.5. Update the best food source position

The ABC algorithm updates the best food source position after termination of the onlooker bee phase. This process implements once for each cycle. A new best food source position will replace the old if the former provides an equal or better amount of nectar. Otherwise, the old food source position remains valid.

1.3.6. Scout bee phase

If solution x_i shows no improvement during the greedy selections of the employed bee phase and onlooker bee phase, its counter holding trials are incremented by 1; otherwise, the counter is reset to 0. This action updates the counters during the search process. Counters are then used to determine if a source is to be abandoned. If the value of the counter is greater than the number of user-specified trials, the source associated with this counter is assumed to be exhausted and is abandoned. The food source abandoned by its bee is replaced with a new food source discovered by the scout, using Eqn (7).

1.3.7. Stop condition

The optimization process terminates when the user-determined stop criterion is met. Maximum generation (G_{max}) was used as this study.
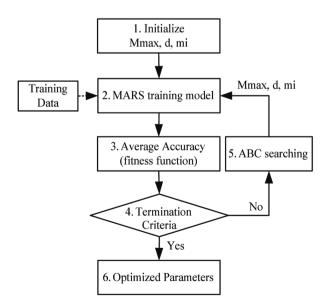

Fig. 3. EMARS model

2. Evolutionary multivariate adaptive regression splines

One may construct a MARS model with various choices of parameters including maximum basis function M_{max}, maximum interaction mi, and penalty parameter d. However, using MARS makes it difficult to set optimal parameters simultaneously. Appropriate parameter settings can significantly improve MARS prediction accuracy. Suggestions for selecting these parameter values are represented in Friedman (1991). However, it is difficult to select the best values because the range of choices is large and proper values may still lie outside suggested ranges. The authors thus proposed EMARS as a tool to help users address this challenging task. The EMARS model is illustrated in Figure 3.

2.1. Input data

Original data are normalized into a range of (0;1) to avoid attributes in greater numeric ranges dominating those in smaller numeric ranges as well as avoid numerical difficulties. The function used to normalize data is shown in Eqn (11). Orenstein *et al.* (1995) showed that ten folds are optimal (i.e. ten folds obtain the shortest acceptable bias and variance for validation testing time).

Data were first divided into 10 cross-fold training and testing sets. Training sets (nine of the ten) were then utilized to determine the best models. Finally, the models were tested via holdout data (the remaining set). Cross-fold training and testing were repeated ten times. Training data were then used to construct the EMARS model, which was divided into two sets, namely one set (80%) for training and the other for validation:

$$x_i^{nor} = \frac{x_i - x_{min}}{x_{max} - x_{min}} \, , \tag{11}$$

where: x_i is any data (input or output); x_{min} is the minimum value of the entire dataset; x_{max} is the maximum value of the entire dataset; and x_i is the normalized value of the data.

2.2. MARS training model

In this stage, MARS is deployed to handle the underlying function. A new MARS model is built for each set of ABC-provided parameter values. ABC's greedy selector will compare model quality based on fitness function evaluation.

2.3. Fitness function

After the training process, MARS is applied to test for the validating set. To find the optimal set of tuning parameters, the following objective function is employed during the fitness function stage:

$$f = E_{training} + E_{validating} \, , \tag{12}$$

where: $E_{training}$ and $E_{validating}$ indicate training error and validating error, respectively. In Eqn (12) Root mean square error is used as the prediction error (Eqn (15)).

It is worth noticing that the fitness function in Eqn (12) indicates the trade-off between model generalization and model complexity. It is well known that good training data fitness often reflects an over-fitted model (Bishop 2006). Hence, the combination of training error and validating error can establish a model optimally balances minimum training error and model generalizability.

2.4. ABC searching

In this stage, ABC carries out the search for best parameter setting values, including: maximum basis function M_{max}, maximum interaction *mi*, and penalty parameter *d*. It regards these parameters as three independent variables. Objective function is prediction accuracy.

2.5. Stop condition

The optimization process is terminated once the stop criterion is satisfied. Prior to this point, the model will proceed to the next generation. Because ABC is used as the search engine, the stop criterion used is often either generation number or number of function evaluation *NFE*. The present study employed generation number.

2.6. Optimal prediction model

The optimal prediction model with fittest parameter settings is found when the stop criterion is fulfilled. This means that EMARS has completed the training process and is ready to predict new input patterns (testing data).

3. Data description

Data were extracted from previously published research studies (Gesoğlu *et al.* 2010; Güneyisi *et al.* 2004). Historical data used in this study comprised 70 distinct concrete mixtures that were cast and tested for compressive and tensile strengths. Natural sand (maximum particle size: 4 mm; fineness modulus: 1.39) and crushed limestone (maximum particle size: 20 mm; and fineness modulus: 5.6)

were used as fine and coarse aggregate, respectively. The experiment used two types of tire rubber, namely crumb rubber and tire chips, which were taken from used truck tires discarded after the second recapping. Crumb rubber is fine aggregate with a gradation close to that of the sand described above and determined based on ASTM C136 (2014) standard. Tire chips were produced using mechanical shredding and elongated between 10 and 40 mm. ASTM Type I Portland cement (28-day compressive strength of 50 MPa) and grade silica fume (91% SiO2 content) were used as cementitious materials.

The experiment used the two w/cm ratios of 0.60 and 0.40 with respective cement contents of 350 and 450 kg/m³. The six rubber content ratios ranged from 2.5% to 50% of total aggregate volume, including 2.5%, 5%, 10%, 15%, 25%, and 50%. Rubber content ratios were divided equally between crumb and chip, as previously mentioned. Silica fume was used as a partial replacement for cement to mitigate the loss in concrete strength. Amount of silica fume varied from 0% to 20% by cement weight. With the exception of aggregate constituents, all other mix design parameters, including cementitious material content, w/cm ratio (0.6 and 0.4), and aggregate volume, were held constant.

The experiment used 70 concrete batches. Each batch was mixed according to the ASTM C192 (2016) standard. Concrete mixtures had slump values of 180±20 and 140±20 mm for w/cm ratios of 0.60 and 0.40, respectively. These values were measured in accordance with ASTM C143 (2015). Three cube (150 mm) and 3 cylinder (150×300 mm) specimens were cast from each batch. Thus, a total of 210 cube and 210 cylinder specimens were tested for compressive strength and splitting tensile strength in accordance with ASTM C39 (2016) and ASTM C496 (2004), respectively. Test specimens were demoulded 24 hours after casting and then water cured for 7 days at an ambient temperature of 21±1 °C and 60±5% humidity. Additional details related to the 70 concrete mixtures and experimental process are presented in (Güneyisi *et al.* 2004).

Table 1 shows input and output factors and their corresponding statistical descriptions. Concrete mixtures include cement (C), silica fume (SF), water (W),

Table 1. Input and output parameters

Description	Notation	Parameters	Min.	Max.	Avg.	Std.
Cement (kg/m³)	C	X_1	280.00	450.00	360.00	53.65
Silica fume (kg/m³)	SF	X_2	0.00	90.00	40.00	29.15
Water (l/m³)	W	X_3	180.00	210.00	195.00	15.11
Super-plasticizer (kg/m³)	SP	X_4	5.25	13.50	9.38	4.15
Coarse aggregate (kg/m³)	CA	X_5	522.60	1076.40	898.67	173.13
Fine aggregate (kg/m³)	FA	X_6	338.40	697.00	581.91	112.11
Crumb rubber (kg/m³)	CR	X_7	0.00	110.40	33.45	35.48
Tire rubber (kg/m³)	TC	X_8	0.00	203.70	61.71	65.46
Compressive strength (MPa)	CS	Y_1	7.10	85.77	44.13	22.26
Tensile strength (MPa)	TS	Y_2	0.70	4.70	3.05	1.10

super-plasticizer (SP), coarse aggregate (CA), fine aggregate (FA), crumb rubber (CR), and tire chips (TC). Composition plays a role as a prediction model input variable when compressive strength (CS) and splitting tensile strength (TS) are outputs.

4. Simulation results and comparisons

4.1. Evaluation criteria

The present study used R^2, *MAPE* and *RMSE* to compare the performance of EMARS against other models. R^2 expresses degree of similarity between predicted and actual values, with R^2 values close to 1 indicating predicted and actual values to be very similar. Low *RMSE*, *MAPE* values indicate high confidence in model-predicted values.

The R-square coefficient R^2 measures how well the considered independent variables account for the measured dependent variable. Higher values correlate with greater model predictive capability. The mathematical formula for computing R^2 is:

$$R^2 = 1 - \frac{SSE}{SST} = 1 - \frac{\sum_{i=1}^{n}(Y_{i,actual} - Y_{i,predicted})^2}{\sum_{i=1}^{n}(Y_{i,actual} - \overline{Y}actual)^2}, \quad (13)$$

where: *SST* – total of sum square; *SSE* – sum of square due to error; $Y_{i,predicated}$ – predicted value; $Y_{i,actual}$ – actual values; Y_{actual} – average of actual values; and *n* – sample size.

The mean absolute percentage error (*MAPE*) is a statistical measure of predictive accuracy. It usually indicates accuracy as a percentage. *MAPE* is useful for evaluating the performance of predictive models because of its relative values. *MAPE* effectively reflects relative differences between models because it is unaffected by the size or unit of actual and predicted values. *MAPE* is stated using the following equation:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{\left|Y_{i,predicted} - Y_{i,actual}\right|}{Y_{i,actual}} \times 100. \quad (14)$$

Root mean squared error (*RMSE*) computes the square error of the prediction compared to actual values and computes the square root of the summation value. The *RMSE* is thus the average distance of a data point from the fitted line measured along a vertical line. This tool is efficient at assessing undesirably large differences. *RMSE* is stated using the following equation:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}[Y_{i,predicted} - Y_{i,actual}]^2}. \quad (15)$$

4.2. EMARS experimental results

The authors used k-fold cross-validation method to assess model performance. This study used a stratified 10-fold cross-validation approach. Random selection divided the 70 patterns of data into 10 distinct folds. Each fold was employed in turn as testing data, with remaining folds employed as training data, ensuring that all data set instances were applied in both training and testing phases. Parameters of ABC search were set as follows: maximum iteration number = 200; population size = $6 \times D$ where *D* is number of tuning parameters; the range of M_{max} is [5,40]; *d* belongs to range [0,6]; and *mi* has a range of [1,8]. The proposed model was run in Matlab environment.

Table 2 shows testing and training results for the EMARS model. As can be seen, the proposed model provided a very good value of R^2 for both testing and training processes ($R^2_{compressive}$ = 0.99 for predicting compressive strength and $R^2_{splitting\,.tensile}$ = 0.99 for predicting splitting tensile strength in testing performance). These R^2 values were close to 1, indicating the EMARS models may be employed to accurately approximate the underlying function of concrete strength. In terms of *RMSE*, the EMARS model yielded very small values, with average *RMSE* values for predicting compressive strength and splitting tensile strength in terms of testing performance of 0.99 and 0.07, respectively. There was no significant difference between *RMSE* training and testing results, indicating that the EMARS model is generalizable. In terms of *MAPE*, the average testing and training values for predicting compressive strength were 2.16% and 1.87%, respectively, and for splitting tensile strength were 2.46% and 2.08%, respectively. Training and testing values of *MAPE* smaller than 2.5% ensure proposed model performance reliability.

As mentioned earlier, one of MARS' most important advantages is its ability to verify the importance of input variables. Table 3 displays the ANOVA decomposition of the proposed models for predicting fold 1 compressive strength. The first column lists the ANOVA function number. The second column gives an indication of importance of the corresponding ANOVA function by listing the GCV score for a model with all BFs corresponding to that particular ANOVA function removed. This GCV score can be employed to assess whether the ANOVA function contributes significantly to the model or only slightly improves the global GCV score. The third column gives the number of BFs in the ANOVA function. The last column gives the particular input variables associated with the ANOVA function.

There were a total of 13 basis functions with a maximum of 3 interactions among variables in the final model. Global GCV score was 0.000 in this case. Variable 8 (tire rubber) was the most important parameter, followed by variable 6 (fine aggregate). The presence of variable 2 (silica fume) in 6 of the 13 basis functions demonstrates the important contribution of this variable to modeling

Table 2. EMARS testing and training results

		fold	1	2	3	4	5	6	7	8	9	10	Avg.
Compressive strength	Training	RMSE	0.872	0.786	0.927	0.819	1.041	0.949	0.888	0.680	0.830	1.005	0.880
		MAPE	1.972	1.691	2.059	1.426	2.864	1.927	1.420	1.290	1.984	2.099	1.873
		R^2	0.998	0.999	0.998	0.999	0.998	0.998	0.998	0.999	0.999	0.998	0.998
	Testing	RMSE	0.701	1.256	0.920	0.387	1.281	0.532	0.899	1.448	1.436	1.060	0.992
		MAPE	1.499	1.735	2.776	1.045	2.184	1.081	3.549	2.267	2.246	3.258	2.164
		R^2	0.999	0.990	0.998	1.000	0.996	0.999	0.997	0.994	0.995	0.997	0.996
Splitting tensile strength	Training	RMSE	0.054	0.056	0.059	0.045	0.056	0.060	0.070	0.054	0.055	0.063	0.057
		MAPE	1.891	2.063	2.196	1.410	1.822	2.333	2.627	1.947	2.144	2.398	2.083
		R^2	0.997	0.997	0.997	0.998	0.997	0.997	0.996	0.998	0.998	0.996	0.997
	Testing	RMSE	0.072	0.044	0.077	0.053	0.062	0.077	0.083	0.060	0.096	0.081	0.070
		MAPE	1.973	1.667	2.522	1.931	1.996	2.191	3.803	1.366	2.536	4.623	2.461
		R^2	0.997	0.995	0.994	0.999	0.998	0.992	0.989	0.987	0.972	0.997	0.992

Table 3. ANOVA decomposition of the EMARS model

Function	GCV	Basis function	Variable (s)
1	0.001	1	X_2
2	8.888	2	X_6
3	11.789	2	X_8
4	0.001	1	$X_1 X_2$
5	0.002	1	$X_2 X_7$
6	0.001	1	$X_2 X_8$
7	0.009	1	$X_3 X_6$
8	0.016	2	$X_7 X_8$
9	0.001	2	$X_1 X_2 X_5$

rubberized concrete strength. The absence of variable 4 (super-plasticizer) from all basis functions demonstrates that super-plasticizer does not influence the EMARS rubberized concrete strength model.

4.3. Result comparison

To better evaluate the performance of the EMARS model, this study compared its performance with results obtained by four other approach techniques, including MARS, BPNN, RBFNN, and GP. BPNN and RBFNN were chosen as they are both popular and effective approaches for conducting regression analysis in high dimensional space (Samarasinghe 2006). GP was chosen because this technique was highly regarded by Gesoğlu *et al.* (2010) for its ability to predict the dynamic properties of rubberized concrete. Finally, the paper included MARS as an index to verify the improvements achieved by EMARS. Furthermore, these techniques represent different data mining techniques that are worth investigating in determining the strength of rubberized concrete.

In terms of parameters, MARS was set as follows: the maximum number of basis functions is 30, the maximum interaction among variable is 3, and the penalty parameter is 3; BPNN was set as follows: the number of hidden layers is 1, the number of neurons in the hidden layer is 8, and the learning rate is 1; RBFNN was set as follows: the maximum number of neurons is 16 and the number of neurons to add between displays is 25; These parameters of BPNN and RBFNN were identified by a large number of trails that provided the best performances. GP was set as follows: size of population is 100, the number of generations is 200, and the tournament size is 7. Table 4 shows the average results of testing *10*-fold cross validation for all five models.

For compressive strength prediction errors, average *RMSE* training data values for EMARS, MARS, BPNN, RBFNN, and GP were 0.880, 0.923, 3.303, 1.031, and 1.781, respectively, and average *RMSE* testing values of testing for the same were 0.992, 1.150, 3.612, 1.597, and 1.934, respectively. Results show EMARS to be the fittest model in terms of minimizing *RMSE* values, with a value roughly 16% below the second best model (MARS). In terms of *MAPE*, EMARS yielded the smallest prediction error for both training and testing data: $MAPE_{training}^{EMARS} = 2.164\%$ and $MAPE_{training}^{EMARS} = 1.873\%$. RBFNN earned the second smallest *MAPE* prediction errors, which were significantly larger than EMARS. For evaluating performance based on R^2, EMARS achieved results very close to 1 for both training and testing data in predicting both compressive and splitting tensile strength.

In terms of determining splitting-tensile-strength prediction performance, EMARS was the best model, yielding the smallest values for both training and testing data in terms of *RMSE* and *MAPE*. In comparison with the second best model RBFNN for testing data, EMARS delivered roughly 28% and 40% improvement in terms of *RMSE* and *MAPE*. Furthermore, EMARS also achieved very good values for R^2 (0.992 and 0.997 for testing and training, respectively). All statistical values in Table 4 demonstrate that the proposed EMARS model is suitable to predict rubberized concrete strength very close to observed values.

Table 4. Average testing error

	Models	RMSE	MAPE (%)	R2	RMSE	MAPE (%)	R2
		Compressive strength			Splitting tensile strength		
Training	EMARS	0.880	1.873	0.998	0.057	2.083	0.997
	MARS	0.923	2.061	0.998	0.073	2.714	0.996
	BPNN	3.303	7.651	0.938	0.223	8.307	0.895
	RBFNN	1.031	2.325	0.998	0.063	2.205	0.997
	GPs	1.781	5.658	0.992	0.119	4.883	0.983
Testing	EMARS	0.992	2.164	0.996	0.070	2.461	0.992
	MARS	1.150	2.844	0.996	0.098	3.492	0.986
	BPNN	3.612	9.720	0.917	0.277	11.529	0.840
	RBFNN	1.597	3.682	0.991	0.090	3.449	0.988
	GPs	1.934	6.101	0.989	0.137	5.424	0.964

Conclusions

Earlier and accurate estimation of concrete strength is valuable to the construction industry. This study fused an ABC algorithm with MARS to create a new model, EMARS. In EMARS, MARS chiefly addresses learning and curve fitting by mapping input and output and ABC primarily copes with global optimization. The purpose of developing this new model was to concurrently determine optimal M_{max}, mi and d parameters. The EMARS model integrated the superior characteristics of the two constituent AI techniques to attain comparatively better prediction performance and virtually eliminate the weakness of traditional techniques. The developed model has the capability to reduce the need for human intervention in rubberized concrete strength prediction.

In simulations run to predict rubberized concrete strength under various rubber and silica fume ratios, this study compared EMARS model performance against four other data mining techniques, including MARS, BPNN, RBFNN, and GP. A *10*-fold cross-validation model was utilized to diminish prediction bias. Comparisons were based on the three criteria of R^2, *MAPE*, and *RMSE*. Result comparisons have shown that the proposed approach can achieve roughly 16% and 28% reduction in *RMSE* for compressive strength prediction and splitting-tensile strength prediction compared to other data-mining techniques, respectively. Experimental results amply demonstrate EMARS as a reliable and proper tool for civil engineers to predict rubberized concrete strength.

Because EMARS allows civil engineers to determine rubberized concrete strength without running costly and time-consuming confirmatory experiments, it should be considered an effective tool to manage concrete quality and reduce project time. Civil engineers can employ EMARS to strike an optimal balance between quality and cost by creating the ideal mix proportion in the concrete aggregation that meets but does not unnecessarily exceed project concrete strength requirements.

References

Abdollahzadeh, A.; Masoudnia, R.; Aghababaei, S. 2011. Predict strength of rubberized concrete using artificial neural network, *WSEAS Transactions on Computers* 10(2): 31–40.

Andalib, A.; Atry, F. 2009. Multi-step ahead forecasts for electricity prices using NARX: a new approach, a critical analysis of one-step ahead forecasts, *Energy Conversion and Management* 50(3): 739–747.
http://dx.doi.org/10.1016/j.enconman.2008.09.040

ASTM C39 / C39M-16, Standard Test Method for Compressive Strength of Cylindrical Concrete Specimens, ASTM International, West Conshohocken, PA, 2016.

ASTM C136 / C136M-14, Standard Test Method for SieveAnalysis of Fine and Coarse Aggregates, ASTM International, West Conshohocken, PA, 2014.

ASTM C143 / C143M-15a, Standard Test Method for Slump of Hydraulic-Cement Concrete, ASTM International, West Conshohocken, PA, 2015.

ASTM C192 / C192M-16, Standard Practice for Making and Curing Concrete Test Specimens in the Laboratory, ASTM International, West Conshohocken, PA, 2016.

ASTM C496 / C496M-11, Standard Test Method for Splitting Tensile Strength of Cylindrical Concre-te Specimens, ASTM International, West Conshohocken, PA, 2004.

Bentur, A.; Cohen, M. D. 1987. Effect of condensed silica fume on the microstructure of the interfacial zone in portland cement mortars, *Journal of the American Ceramic Society* 70(10): 738–743.
http://dx.doi.org/10.1111/j.1151-2916.1987.tb04873.x

Bishop, C. M. 2006. *Pattern recognition and machine learning (information science and statistics)*. New York: Springer-Verlag, Inc. 740 p.

Cheng, M.-Y.; Wu, Y.-W. 2009. Evolutionary support vector machine inference system for construction management, *Automation in Construction* 18(5): 597–604.
http://dx.doi.org/10.1016/j.autcon.2008.12.002

Davidson, J. W.; Savic, D. A.; Walters, G. A. 2003. Symbolic and numerical regression: experiments and applications, *Information Sciences* 150(1–2): 95–117.
http://dx.doi.org/10.1016/S0020-0255(02)00371-7

Eldin, N. N.; Senouci, A. B. 1994. Measurement and prediction of the strength of rubberized concrete, *Cement and Concrete Composites* 16(4): 287–298.
http://dx.doi.org/10.1016/0958-9465(94)90041-8

Friedman, J. H. 1991. Multivariate adaptive regression splines, *The Annals of Statistics* 19(1): 1–67.
http://dx.doi.org/10.1214/aos/1176347963

García Nieto, P. J.; Martínez Torres, J.; de Cos Juez, F. J.; Sánchez Lasheras, F. 2012. Using multivariate adaptive regression splines and multilayer perceptron networks to evaluate paper manufactured using Eucalyptus globulus, *Applied Mathematics and Computation* 219(2): 755–763. http://dx.doi.org/10.1016/j.amc.2012.07.001

Gesoğlu, M.; Güneyisi, E.; Özturan, T.; Özbay, E. 2010. Modeling the mechanical properties of rubberized concretes by neural network and genetic programming, *Materials and Structures* 43(1–2): 31–45. http://dx.doi.org/10.1617/s11527-009-9468-0

Güneyisi, E.; Gesoğlu, M.; Özturan, T. 2004. Properties of rubberized concretes containing silica fume, *Cement and Concrete Research* 34(12): 2309–2317. http://dx.doi.org/10.1016/j.cemconres.2004.04.005

Hong, W.-C. 2011. Electric load forecasting by seasonal recurrent SVR (support vector regression) with chaotic artificial bee colony algorithm, *Energy* 36(9): 5568–5578. http://dx.doi.org/10.1016/j.energy.2011.07.015

Karaboga, D. 2005. *An idea based on honey bee swarm for numerical optimization*. Technical Report-TR06. Erciyes University, Kayseri,Türkiye.

Karaboga, D.; Akay, B. 2009. A comparative study of Artificial Bee Colony algorithm, *Applied Mathematics and Computation* 214(1): 108–132. http://dx.doi.org/10.1016/j.amc.2009.03.090

Khaloo, A. R.; Dehestani, M.; Rahmatabadi, P. 2008. Mechanical properties of concrete containing a high volume of tire–rubber particles, *Waste Management* 28(12): 2472–2482. http://dx.doi.org/10.1016/j.wasman.2008.01.015

Khatib, Z.; Bayomy, F. 1999. Rubberized portland cement concrete, *Journal of Materials in Civil Engineering* 11(3): 206–213. http://dx.doi.org/10.1061/(ASCE)0899-1561(1999)11:3(206)

Kiranyaz, S.; Ince, T.; Yildirim, A.; Gabbouj, M. 2009. Evolutionary artificial neural networks by multi-dimensional particle swarm optimization, *Neural Networks* 22(10): 1448–14462. http://dx.doi.org/10.1016/j.neunet.2009.05.013

Leathwick, J. R.; Elith, J.; Hastie, T. 2006. Comparative performance of generalized additive models and multivariate adaptive regression splines for statistical modelling of species distributions, *Ecological Modelling* 199(2): 188–196. http://dx.doi.org/10.1016/j.ecolmodel.2006.05.022

Lee, T.-S.; Chen, I. F. 2005. A two-stage hybrid credit scoring model using artificial neural networks and multivariate adaptive regression splines, *Expert Systems with Applications* 28(4): 743–752. http://dx.doi.org/10.1016/j.eswa.2004.12.031

Lee, T.-S.; Chiu, C.-C.; Chou, Y.-C.; Lu, C.-J. 2006. Mining the customer credit using classification and regression tree and multivariate adaptive regression splines, *Computational Statistics & Data Analysis* 50(4): 1113–1130. http://dx.doi.org/10.1016/j.csda.2004.11.006

Li, H.; Liu, K.; Li, X. 2010. A comparative study of artificial bee colony, bees algorithms and differential evolution on numerical benchmark problems, in Z. Cai, H. Tong, Z. Kang, Y. Liu (Eds.). *Computational intelligence and intelligent systems*. Springer: Berlin Heidelberg, 198–207.

Lu, C.-J.; Lee, T.-S.; Lian, C.-M. 2012. Sales forecasting for computer wholesalers: a comparison of multivariate adaptive regression splines and artificial neural networks, *Decision Support Systems* 54(1): 584–596. http://dx.doi.org/10.1016/j.dss.2012.08.006

Orenstein, T.; Kohavi, Z.; Pomeranz, I. 1995. An optimal algorithm for cycle breaking in directed graphs, *Journal of Electronic Testing* 7(1–2): 71–81. http://dx.doi.org/10.1007/BF00993315

Samarasinghe, S. 2006. *Neural networks for applied sciences and engineering: from fundamentals to complex pattern recognition*. Boca Raton: Auerbach Publications.

Samui, P. 2012. Multivariate Adaptive Regression Spline (Mars) for prediction of elastic modulus of jointed rock mass, *Geotechnical and Geological Engineering* 31(1): 249–253. http://dx.doi.org/10.1007/s10706-012-9584-4

Sánchez-Lasheras, F.; de Andrés, J.; Lorca, P.; de Cos Juez, F. J. 2012. A hybrid device for the solution of sampling bias problems in the forecasting of firms' bankruptcy, *Expert Systems with Applications* 39(8): 7512–7523. http://dx.doi.org/10.1016/j.eswa.2012.01.135

Sekulic, S.; Kowalski, B. R. 1992. MARS: a tutorial, *Journal of Chemometrics* 6(4): 199–216. http://dx.doi.org/10.1002/cem.1180060405

Son, K. S.; Hajirasouliha, I.; Pilakoutas, K. 2011. Strength and deformability of waste tyre rubber-filled reinforced concrete columns, *Construction and Building Materials* 25(1): 218–226. http://dx.doi.org/10.1016/j.conbuildmat.2010.06.035

Topçu, İ. B.; Sarıdemir, M. 2008. Prediction of rubberized concrete properties using artificial neural network and fuzzy logic, *Construction and Building Materials* 22(4): 532–540. http://dx.doi.org/10.1016/j.conbuildmat.2006.11.007

Topçu, İ. B.; Avcular, N. 1997. Analysis of rubberized concrete as a composite material, *Cement and Concrete Research* 27(8): 1135–1139. http://dx.doi.org/10.1016/S0008-8846(97)00115-4

Vidoli, F. 2011. Evaluating the water sector in Italy through a two stage method using the conditional robust nonparametric frontier and multivariate adaptive regression splines, *European Journal of Operational Research* 212(3): 583–595. http://dx.doi.org/10.1016/j.ejor.2011.02.003

Yang, J.-B.; Yau, N.-J. 2000. Integrating case-based reasoning and expert system techniques for solving experience-oriented problems, *Journal of the Chinese Institute of Engineers* 23(1): 83–95. http://dx.doi.org/10.1080/02533839.2000.9670526

Yeh, I. C. 1998. Modeling of strength of high-performance concrete using artificial neural networks, *Cement and Concrete Research* 28(12): 1797–1808. http://dx.doi.org/10.1016/S0008-8846(98)00165-3

Zheng, L.; Sharon Huo, X.; Yuan, Y. 2008. Experimental investigation on dynamic properties of rubberized concrete, *Construction and Building Materials* 22(5): 939–947. http://dx.doi.org/10.1016/j.conbuildmat.2007.03.005

Zhou, Y.; Leung, H. 2007. Predicting object-oriented software maintainability using multivariate adaptive regression splines, *Journal of Systems and Software* 80(8): 1349–1361. http://dx.doi.org/10.1016/j.jss.2006.10.049

Min-Yuan CHENG. Distinguished Professor of Department of Construction Engineering at the National Taiwan University of Science and Technology, an Editorial Board member of Automation in Construction journal. His research interests include construction business process re-engineering, geographic information system, construction automation and E-business for construction management, project management information system, artificial intelligence applications in construction management.

Minh-Tu CAO. PhD candidate of Department of Construction Engineering at the National Taiwan University of Science and Technology. He received Master's degree in Construction Management (2012). His research interests include applications of artificial intelligence in construction management, construction economics, and decision-making support systems.