Time-based metamodeling technique for vehicle crashworthiness optimization

Hu Wang a,⁎, G.Y. Li a,⁎, Enying Li b

a Key Laboratory of Advanced Technology for Vehicle Body Design & Manufacture, College of Mechanical and Automobile Engineering, Hunan University, Changsha, 410082, PR China
b School of Logistics, Central South University of Forestry and Telegey, Changsha, 410004, P.R. China

A R T I C L E   I N F O

Article history:
Received 21 February 2009
Accepted 12 April 2010
Available online xxxx

Keywords:
Time-based metamodeling
Intelligent sampling
Support vector regression
Frontal collision
Crashworthiness

A B S T R A C T

In automotive industry, structural optimization for crashworthiness criteria is of special importance in the early design stage. To reduce the vehicle design cycle, metamodeling techniques have become so widespread... In this study, a time-based metamodeling technique is proposed for the vehicle design. The characteristics of the proposed method are the construction of a time-based objective function and establishment of a metamodel by support vector regression (SVR). Compared with other popular metamodel-based optimization methods, the design space of the proposed method is expanded to time domain. Thus, more information and features can be extracted in the expanded time domain. To validate the performance of the time-based metamodeling technique, cylinder impacting and full vehicle frontal collision are optimized by the proposed method. The results demonstrate that the proposed method has potential capability to solve the crashworthiness vehicle design.

© 2010 Elsevier B.V. All rights reserved.

1. Introduction

Vehicle safety can be measured by parameters such as the contact forces exerted on the occupants and the resulting accelerations during a vehicle crash procedure [1]. These safety parameters are largely determined by the amount of energy absorbed by the vehicle before the impact wave propagates to the occupants. With the aid of finite element (FE) analysis software designed for dynamic impact problems, such as LS-DYNA and PAM-CRASH, etc., it is possible to simulate the crashing behavior of structures under impact and evaluate these parameters.

However, numerical optimization often requires many evaluations to achieve an optimum design. It often becomes inapplicable for a full vehicle crashworthiness design problem, as each impact simulation/analysis requires extensive computation resources. For example, one run of an FE model for vehicle crashworthiness may take several hours to finish the calculation. A widely used strategy is to utilize approximation models which are often referred to as metamodels as they provide a finite model of the model [2]. This technique commonly is utilized to overcome this challenge in a crashworthiness design optimization.

Among the widely used metamodeling techniques in the crashworthiness optimization, there are various approximation methods such as response surface methodology (RSM) [3], Kriging (KG) [4,5], radial basis functions (RBF) [6], moving least square method (MLSM) [7] and multivariate adaptive regression splines (MARS) [8] for engineering design in the last 20 years. For further reading about corresponding techniques, see other literatures [9–12].

Unfortunately, it is very difficult for the above mentioned metamodeling techniques to capture all features in the course of the crash process. In our opinion, the causes of this problem are complicated and due to the three interrelated stages: construction of objective functions, design of experiments (DOEs) and modeling.

Commonly, objective functions need definition at the beginning stage of optimization. For crashworthiness optimization, peak values such as a maximum contact force and acceleration can be objective functions. Actually, the essence of the entire crash procedure cannot be presented by peak values simply. For example, if the peak acceleration only occurs in a very short-time, it will not affect occupants indeed. Such objective function has no real meaning in practical engineering problems. Thus, in order to construct a metamodel which can capture the features of an entire simulation process; it is necessary to establish a time-based metamodeling technique for representing dynamical process objectively.

DOE is important as a formal way of maximizing information gained while resources required. The efficiency and accuracy of the metamodel-based optimization largely depend on the DOE. To improve the efficiency of optimization, an intelligent DOE: boundary and best neighbor searching (BBNS) strategy introduced by Wang and Li [10,11] is employed in this study. The BBNS generates new samples based on the boundary samples and the best neighbor samples distributed in the design space. Therefore, the number of samples and size of design space can be controlled easily by the BBNS.

In addition, the accuracy of optimization is determined by the approximation method directly. For most of the approximate methods, the accuracy is assessed by statistical analysis such as...
2. Time-based metamodeling technique

The vehicle impact response is commonly described by the acceleration history with the peak acceleration typically used as a rough indicator of impact severity. The peak acceleration is determined by the amount of kinetic energy that can be absorbed by the vehicle and time that it takes for this energy to be absorbed. Therefore, we need to consider two key factors: energy absorption capacity and rate. In order to minimize the amount of energy transferred to occupants, the purpose of the crashworthiness optimization is to maximize the vehicles energy absorption capacity. Although the peak acceleration is often used as the key objective function, the peak acceleration-based objective function cannot capture the features of energy absorption capacity completely.

For instance, the peak acceleration is extracted at time A as shown in Fig. 1 (left plot); the value of the corresponding objective can be reduced at time A. When the optimization is completed, the acceleration might be decreased at time A. Because such objective function doesn’t consider the entire time series of crash process, the acceleration at time B might be increased and the peak acceleration increases accordingly. It is urgent to establish a strategy to capture the features of the entire crash process.

For this reason, the time-based metamodeling technique is suggested in this study. The purpose of this scheme is to obtain an overall decline curve based on the entire crash process as shown in Fig. 1 (right plot). For this reason, a time-based metamodeling technique is proposed and the basic idea of this strategy is illustrated in Fig. 2. Compared with other metamodeling techniques, the distinctive characteristic of the proposed scheme is to establish metamodel-based on history of an entire simulation process. For most of the commonly used metamodeling techniques, the extremum response value of each evaluation should be the objective function and the objective function can be represented in design space demonstrated in the bottom-right plot of Fig. 2. In the proposed strategy, the time-based objective function can be shown as bottom-left plot of Fig. 2. The design space of the time-based strategy is expanded to the time domain. It is observed that much more information can be extracted from the expanded time-based design space.

Although the design space is expanded, it brings another challenge, how to extract useful information from the time domain. Obviously, it is impossible to use all discrete time points to build time-based objective functions directly. Otherwise, a large number of metamodels need to be constructed. It is an inapplicable strategy to obtain optimum results with the superabundance of metamodels. Furthermore, such strategy might weaken features of the time-based objective function. In our opinion, the objective function should be established based on some sensitive time series extracted from the entire simulation process, based on this principle, the proposed strategy is established and presented in Fig. 3. According to Fig. 3, there are three key issues need to be discussed:

1. Which approximation method can be used to generate the robust metamodel?
2. Which DOE method should be employed to build highly accurate metamodel efficiently?
3. How to extract sensitive time series from the entire simulation procedure?

In the following sections, corresponding methods are suggested for the above three issues. Sections 2.1, 2.2 and 2.3 answer the above three questions respectively.
Fig. 2. An illustration of basic ideal of a time-based metamodeling technique.

Fig. 3. The flowchart of proposed time-based metamodeling technique.
2.1. Least square support vector regression (LS-SVR)

To obtain a robust and accurate metamodel, the LS-SVR is engaged in this study. Consider the problem of approximating the set of \( N \) samples \( \{(x_i, y_i)\}_{i=1}^N \) with input data \( x_i \) and output data with \( y_i \) and corresponding response as presented in Eq. (1)

\[
D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}. x_i \in \mathbb{R}^p, y_i \in \mathbb{R}
\]  

(1)

The optimization problem in primal weight space can be expressed as

\[
\min_{w, \varepsilon} f(w, \varepsilon) = \frac{1}{2} w^T w + \frac{1}{2} \sum_{i=1}^{N} \varepsilon_i^2
\]  

(2)

subjected to

\[
y_i = w^T \varphi(x_i) + b + \varepsilon_i, i = 1, 2, \ldots, N
\]  

(3)

for \( \varphi(\cdot) = \mathbb{R}^p \rightarrow \mathbb{R}^p \), a kernel that maps the input space into a so-called higher dimensional design space. Weight vector \( w \in \mathbb{R}^p \) in primal space, error variable \( \varepsilon_i \in \mathbb{R} \) and bias term is \( b \). The cost function \( f \) consists of an SSE fitting error and regulation term. The relative importance of the EMR and SRM terms is determined by the positive constant \( \gamma \).

The model of primal space can be presented as follows

\[
y(x) = w^T \varphi(x) + b
\]  

(4)

The weight vector \( w \) can be of infinite dimension, which makes a calculation of \( w \) from Eq. (2) impossible in general. Therefore, the model in the dual space is used instead of the primal space. Then, the Lagrangian multiplier expression applied for Eqs. (4) and (5) is given by

\[
\left\{ \begin{array}{l}
\partial \ell (w, \varepsilon, a) = f(w, \varepsilon) - \frac{1}{2} \sum_{i=1}^{N} a_i \left( w^T \varphi(x_i) + b + \varepsilon_i - y_i \right) \\
\partial w \ell (w, \varepsilon, a) = 0 \Rightarrow w = \sum_{i=1}^{N} a_i \varphi(x_i) \\
\partial \varepsilon \ell (w, \varepsilon, a) = 0 \Rightarrow \varepsilon_i = 0 \\
\partial a \ell (w, \varepsilon, a) = 0 \Rightarrow a_i = \gamma \varepsilon_i \\
\partial a \ell (w, \varepsilon, a) = 0 \Rightarrow w^T \varphi(x_i) + b + \varepsilon_i - y_i = 0
\end{array} \right.
\]  

(6)

Eliminating \( w \) and \( \varepsilon \), the solution is obtained as

\[
0 = \begin{bmatrix} 0 & I^T \\ I & \Gamma + \frac{1}{\gamma} I \end{bmatrix} \begin{bmatrix} b \\ a \end{bmatrix} = \begin{bmatrix} 0 \\ y \end{bmatrix}
\]  

(7)

where

\[
y = [y_1, y_2, \ldots, y_N]
\]  

(8.1)

\[
l = [1, 1, \ldots, 1]
\]  

(8.2)

\[
an = [a_1, a_2, \ldots, a_N]
\]  

(8.3)

\[
\Gamma_{ij} = \varphi(x_i)^T \varphi(x_j) \text{ for } i, j = 1, 2, \ldots, N
\]  

(8.4)

Based on the Mercer’s condition [15], there exists a mapping \( \varphi(\cdot) \) and an expression can be written as

\[
K(x,y) = \sum_i \varphi_i(x)^T \varphi_i(y).
\]  

(9)

If and only if, for any \( g(x) \) such that \( \int g(x)^2 dx \) is finite

\[
\int K(x,y)g(x)g(y)dx dy > 0
\]  

(10)

As a result, the kernel \( K(\cdot, \cdot) \) such that

\[
K(x_i, x_j) = \varphi(x_i)^T \varphi(x_j), \text{ for } i, j = 1, 2, \ldots, N
\]  

(11)

The LS-SVR model for the function estimation is obtained as

\[
y(x) = \sum_{i=1}^{N} a_i K(x_i, x) + b
\]  

(12)

where \( a_i \) and \( b \) are the solutions of Eq. (7). In this study, the RBF kernel \( K(x_i, x_j) = \exp \left( \frac{-(\|x_i - x_j\|_2^2)}{\sigma} \right) \) is employed.

2.2. Intelligent DOE

The training sample set for the LS-SVM is often carried out by neural network (NN). Although the NN-based sample generator can reduce the range of design variables in considerable small intervals, the NN training procedure might not be easy to converge for large-scale nonlinear problems due to the expensive computational consumption of evaluations, such as crashworthiness design. Hence, an efficient intelligent DOE strategy—BBNS [10,11] is introduced to generate samples for the LS-SVR.

For most of the real-world engineering problems, bounds of design variables can be given based on engineering experiences. Therefore, boundary information may assist use to generator efficiently and control the size of design space. In the BBNS, initial samples should be generated by the space filling sampling strategies, such as Latin hypercube sampler (LHS), etc. To save computational expenses of evaluation procedure, the initial samples should be sparsely distributed. Next, the initial samples are sorted in descending order according to the response values of the objective functions. And then, we collect several better samples on the top of sorted sample list as new initial samples. The selected samples are used to generate the new sample based on the nearest boundary sample and best sample in neighbor region until criterion for convergence is reached. The details of the BBNS algorithm are described as follows and corresponding diagram is illustrated in Fig. 6.

Step 1. Generate the initial sparsely distributed samples in design space by LHS (any kind of popular DOEs can be used to generate the initial samples theoretically);

Step 2. Evaluate objective functions by fitful toolkit, such as the FE analysis;

Step 3. Sort generated samples in descending order according to the response values of objective functions. Gather the top of several samples on the sorted list. A collected sample and a set of better samples are called a “better sample” and “better-sample-set”, respectively (as Fig. 4 shows).

Step 4. A new sample is generated by Eqs. (13) and (14).
Step 4.1. The location of the new sample is given by

\[ X_{1.2-n} = \frac{X_{\text{Current}} + X_{\text{Nearest}}}{m_1} + \frac{X_{\text{Current}}}{m_2}, \]

where \( X \) is the coordinate vector of sample, \( n \) denotes the number of design variables, \( m_1, m_2 \) are determined by

\[
\begin{align*}
    m_{1.2-n} &= \frac{R(X_{\text{Current}}) + R(X_{\text{Boundary/Nearest}})}{R(X_{\text{Current}})}, \\
    m_{2.1-n} &= \frac{R(X_{\text{Current}}) + R(X_{\text{Best/Nearest}})}{R(X_{\text{Best/Nearest}})}.
\end{align*}
\]

\( c_1, c_2 \) denote the acceleration weight coefficients, they are determined by Eq. (15)

\[
\begin{align*}
    c_1 &= \frac{X_{\text{Best/Nearest}} + X_{\text{Boundary/Nearest}}}{X_{\text{Current}}}, \\
    c_2 &= 1 - c_1.
\end{align*}
\]

The meanings of superscripts are defined as:

- Current: the current assigned sample in better-sample-set;
- Nearest: the nearest sample from current assigned sample;
- Boundary: the boundary of bounds;
- Best: the sample which possesses the best value of the objective function;
- Boundary (nearest): the nearest boundary sample from the current assigned sample;

Best (nearest): the nearest sample of better-sample-set from current assigned sample.

The searching pattern is demonstrated in Fig. 4. Compared with other intelligent sampling methods, the new sample is generated by considering the boundary and neighbor information.

Step 4.2. If the new sample is repeatedly generated or outside of the design space, the best sample should be substituted by the present assigned sample, and goes back to Step 4.1;

Step 4.3. Evaluate the objective functions with the new generated samples;

Step 5. If

\[
\frac{|v_{\text{new}} - v_{\text{old}}|}{|v_{\text{best new}}|} \leq \eta \quad \eta \in (0, 1)
\]

then procedure ends, else goes to step 3, where \( \eta \) is threshold which can be assigned by the user and the default value is 10%.

Although engineering experiences help us reduce design space, it might be a potential drawback. If optimum solutions aren’t involved in the initial design space, the sequential optimum results must be error. Thus, it is necessary to validate design space when the optimum solutions are on or near the boundaries of design space. The boundary-extension validation strategy is briefly introduced as follows.

If a design variable linearly varies with an objective function, a corresponding interval of design variable should be extended in both positive and negative directions as shown in Fig. 5. And then, test samples should be distributed in extended intervals uniformly or randomly and should be performed with the test samples. If a design variable is also linear, the constructed metamodel can be promised. Otherwise, the initial interval of design variable needs modifications (Fig. 6).

2.3. Cumulation extraction strategy

As mentioned in previous sections, we need to extract sensitive time series from the entire simulation procedure. Therefore, the cumulation extraction strategy is suggested to generate sensitive time clusters. The corresponding strategy can be accomplished through the following five steps:

1. Normalization: response values of objective functions should be normalized first, and the maximum response function value should be set to 1;

![Boundary samples](image)

Fig. 5. Boundary-extension validation strategy.
2. Clustering: data of each sample should be clustered by fuzzy c-mean (FCM) [20] clustering method. Theoretically, any partition methods can be applied in this study. We have tested several methods, such as greedy, SVM. Due to the huge number of samples, these methods are not sensitive to final results. Thus, the FCM, a simple and efficient partition method is used in this step. In addition, how to determine the number of clusters \( n \) is an open issue. Theoretically, with the increasing of the number of clusters, more detailed information of the entire time history should be gathered and more metamodels need to be constructed. Conversely, small \( m \) might not capture all features of the objective functions but fewer metamodels need to be built. For this reason, the default cluster number (experience value) is 5 in the developed system and we suggest to user define this value according to the complexity of the problem.

3. Simplification: the historic time curve of the objective functions should be simplified as shown in Fig. 7, the response function value in each of the cluster should be the mean value in the corresponding cluster given by Eq. (17):

\[
x_{\text{mean}}^j = \frac{\sum_{i=1}^{m} x_i^j}{m}
\]  

(17)

where \( x_i^j \) denotes the value of the \( i \)th time points in the \( j \)th time cluster, \( x_{\text{mean}}^j \) is the mean value of the \( j \)th time series.

4. Cumulation: response values of all samples should be cumulated and averaged at each time point, the updated cumulation plot is illustrated as shown in Fig. 8 (upper plot);
be established in this phase. Consequently, the objective functions should be determined and updated. And then the BBNS is employed to generate samples. In the second phase, the metamodel is constructed by the LS-SVR based on the updated samples as shown in Fig. 3. Finally, the optimization should be performed with the metamodel.

3. Particle swarm optimization (PSO)

In this study, PSO is applied to optimize the constructed metamodel and briefly described in this section. The PSO is a population based optimization approach, first developed by Kennedy and Eberhart in 1995 [22,23]. The PSO shares many similarities with evolutionary computation techniques such as Genetic Algorithms (GA). In the PSO, the potential solutions, called particles, fly through the problem space by following the current optimum particles. The position, \( x_i \), of the \( i \)th particle is adjusted by a stochastic velocity \( v_i \) which depends on the distance that the particle is from its own best solution and that of its neighborhood. For the original PSO, the velocity and position are determined by

\[
v_i(t + 1) = w v_i(t) + \phi_1 (y_i(t) - x_i(t)) + \phi_2 (\hat{y}_i(t) - x_i(t))
\]

\[
x_i(t + 1) = x_i(t) + v_i(t + 1)
\]

for \( i = 1, ..., m \) and \( j = 1, ..., n \), where \( \phi_1 = a c_1 r_1(t) \) and \( \phi_2 = a c_2 r_2(t) \), \( m \) is the total number of particles in the swarm, \( n \) is the dimension of the problem, i.e., the number of parameters of the function being optimized, \( a c_1 \) and \( a c_2 \) are acceleration coefficients, \( r_1(t) \) and \( r_2(t) \) are the random numbers between 0 and 1, \( x_i(t) \) is the position of particle \( i \) of \( j \)th design parameter at time step \( t \), \( v_i(t) \) is the velocity of particle \( i \) of \( j \)th design parameter at time step \( t \), \( y_i(t) \) is the best fitness of particle \( i \) at time step \( t \), \( \hat{y}_i(t) \) is the best fitness found by the neighborhood of particle \( i \) at time step \( t \), and \( \phi \) and \( \kappa \) are acceleration coefficients.

Empirical results have shown that a constant inertia of \( w = 0.7298 \) and acceleration coefficients with \( a c_1 = a c_2 = 1.49618 \) provide good convergent behavior [21]. Clerc and Kennedy [22] provided a theoretical analysis of particle trajectories to ensure convergence to a stable point. As a result of this study, the velocity equation changes to

\[
v_i(t + 1) = \chi (v_i(t) + \phi_1 (y_i(t) - x_i(t)) + \phi_2 (\hat{y}_i(t) - x_i(t)))
\]

with \( \chi \) is the constriction coefficient calculated by

\[
\chi = \frac{2\kappa}{2 - \phi - \sqrt{\phi^2 - 4\phi}}
\]

with \( \phi = \phi_1 + \phi_2 \geq 4 \) and \( \kappa \in [0,1] \). The constant \( \kappa \) controls the speed of convergence.

<table>
<thead>
<tr>
<th>Table 1</th>
<th>Optimization results and DOE information of case 1.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DOE</strong></td>
<td><strong>Metamodelling techniques</strong></td>
</tr>
<tr>
<td><strong>LHS</strong></td>
<td><strong>LHS</strong></td>
</tr>
<tr>
<td>Number of samples</td>
<td>x1</td>
</tr>
<tr>
<td>Optimal design (mm)</td>
<td>x2</td>
</tr>
<tr>
<td>FE simulation result</td>
<td>60.3</td>
</tr>
<tr>
<td>Internal force (kJ)</td>
<td>Approximation value</td>
</tr>
<tr>
<td>FE simulation result</td>
<td>8.3</td>
</tr>
</tbody>
</table>
4. Crashworthiness optimization

4.1. Problem 1: optimization of a cylindrical tube impacting a rigid wall

4.1.1. Problem description

The optimization of a cylindrical tube impacting a rigid wall with the initial velocity of 10 m/s is considered as shown in Fig. 9. The von-Mises yielding condition is used, but the strain rate effect is ignored. Young’s modulus $E = 70$ GPa, Poisson’s ratio $\nu = 0.33$, the yield stress $\sigma_y = 280$ MPa and material density $\rho = 2.82 \times 10^3$ kg/m$^3$ are also referred in [23]. This case, although simple, presents some challenging features encountered in vehicle crashes such as nonlinearity, buckling, and dynamics. The cylindrical tube has a constant mass of 0.54 kg with the radius and thickness as design variables. The length of the cylinder is thus dependent on the design variables because of the mass constraint. A concentrated mass of 500 times the cylinder weight is attached to the end of the cylinder not impacting the rigid wall in order to supply enough energy for crushing. The cylindrical tube is then optimized to maximize the energy absorption subjected to an axial impact force (average normal impact force on the rigid wall) within 20 ms.

The optimization problem in this case is given:

Maximize: internal energy $f_1(X)$

Subject to: rigid wall force $f_2(X) \leq 80$ KN

And cylinder length $L = \frac{0.54}{2\pi\rho x_1 x_2}$ with $x_1 \in [20,100]$ mm and $x_2 \in [2,6]$ mm where the design parameters $x_1$ and $x_2$ are the radius and thickness of the cylinder respectively.

4.1.2. Optimization procedure and results

To assess the performance of the suggested strategy, KG, RBF, 2nd order polynomial RSM (2nd PRS) and the proposed metamodeling technique are implemented for this case. The LHS is used to generate...
10 initial samples for the BBNS, and sizes of samples for other test metamodeling techniques are 50 presented in Table 1. The KG, RBF, 2nd PRS are well known and widely used techniques. Thus, the procedure of the proposed metamodeling technique is only discussed in this section.

Because the objective function of this case is to maximize the internal energy, it cannot be regarded as the objective function directly in Step 2 described in Section 2.3. The increment of the internal energy based on the sensitive time series is the objective function and given by

\[ \text{Max} \{ f_{T_i - T_{i-1}}(X) - f_{T_i}(X) \} \]  

where \( T_i \) denotes the start time of a time series.

The optimum design variables achieved by the KG, RBF, 2nd PRS and BBNS-based LS-SVR are presented in Table 1. The history of the absorbed energy and rigid wall force with respect to time for the initial and optimum design of the cylinder by different metamodeling techniques are plotted in Fig. 10.

In addition, three sensitive time series are extracted as 0.0–0.178 ms, 0.202–0.364 ms, and 0.382–0.632 ms. According to Fig. 10, we notice that most of the internal energy is absorbed in the extracted sensitive time series. The optimum combination of design variables are achieved by the BBNS-based LS-SVR as shown in Table 1. According to Table 1 and Fig. 10, the proposed strategy only uses 34 samples to achieve the best optimum result. To validate the accuracy of the proposed method, the optimum design variables are simulated by the commercial FE solver, LS-DYNA. Compared with the simulation result, the predicted response value is almost the same.

### 4.2. Problem 2: vehicle frontal crashworthiness optimization

#### 4.2.1. Finite element model

In this case, crashworthiness optimization problem is based on a full-scale FE model of 1998 CHEVROLET S10 pickup truck. The original
FE model of this vehicle was developed at FHWA/NHTSA National Crash Analysis Center at the George Washington University [25]. It consists of 20 types of materials and 346 parts for a total mass of 1126 kg. It has 220,499 nodes and 220,409 (the number of shell elements is 203,158) elements. The original FE model before and 150 ms after a frontal impact at 55.69 km/h are shown in Fig. 11. The time history of the internal energy of this case is illustrated in Fig. 11.

4.2.2. Design variables and objective functions

The preliminary vehicle frontal impact simulation shows that more than 40% of the kinetic energy is absorbed within the first 40 ms and more than 90% within 80 ms. As indicated by the time history curves in Fig. 12, of 346 parts, 10 components are found to be responsible for 59% of the energy absorption at 40 ms and for 52% at 80 ms even though they make up only 3.7% of the total vehicle mass.

For crash simulations, LS-DYNA970 on an IBM P690 with a total of 32 1.33 GHz Power4 and 64 g RAM is applied. A single simulation of 100 ms frontal impact (as Fig. 11 shows) takes approximately 9 h using 32 processors.

The thicknesses of the components in the frontal impact event are selected as the design variables. As shown in Fig. 13, the thicknesses of the 8 reinforced members around the frontal structure are chosen as design variables which significantly affect the crash safety. Consequently, we then focused our attention on thirteen parts at three specific instances into the crash, as shown in Fig. 14.

The proposed cumulation extraction strategy proposed in Section 2.3 is employed to cluster and extract the sensitive time series. In this case, 24 initial samples are generated by the LHS. Based on 24 evaluations (simulations), three sensitive time series are extracted as

![Fig. 11. FE model before and after a frontal impact at 55.69 km/h.](image1)

![Fig. 12. An illustration of history of energy.](image2)

![Fig. 13. Selected members need optimization.](image3)

![Fig. 14. FE model of thirteen selected parts at (a) initial state, (b) 20 ms, (c) 40 ms, and (d) 100 ms after impact.](image4)
The baseline of component thickness and response characteristic of case 2.

Table 2
<table>
<thead>
<tr>
<th>Design variables</th>
<th>Component name</th>
<th>Component ID</th>
<th>Thickness (mm)</th>
<th>Internal energy I (kJ)</th>
<th>Internal energy II (kJ)</th>
<th>Mass (kg)</th>
</tr>
</thead>
<tbody>
<tr>
<td>x1</td>
<td>BMP-BUMPER-FT</td>
<td>2000001</td>
<td>2.28</td>
<td>1.442</td>
<td>1.675</td>
<td>1.04</td>
</tr>
<tr>
<td>x2</td>
<td>RF-BOTTOMFRAME</td>
<td>2000052</td>
<td>2.40</td>
<td>1.642</td>
<td>1.792</td>
<td>0.63</td>
</tr>
<tr>
<td>x3</td>
<td>RL-FCROSS-TOP</td>
<td>2000050</td>
<td>3.20</td>
<td>1.442</td>
<td>1.792</td>
<td>0.63</td>
</tr>
<tr>
<td>x4</td>
<td>RF-TOPFRAME</td>
<td>2000014</td>
<td>3.00</td>
<td>1.442</td>
<td>1.792</td>
<td>0.63</td>
</tr>
<tr>
<td>x5</td>
<td>RL-RAIL-INN CFRONT</td>
<td>2000026</td>
<td>3.16</td>
<td>1.442</td>
<td>1.792</td>
<td>0.63</td>
</tr>
<tr>
<td>x6</td>
<td>RL-RAIL-INN CFRONT</td>
<td>2000052</td>
<td>3.16</td>
<td>1.442</td>
<td>1.792</td>
<td>0.63</td>
</tr>
<tr>
<td>x7</td>
<td>RL-RAIL-OUT FRAME</td>
<td>2000001</td>
<td>2.28</td>
<td>1.442</td>
<td>1.792</td>
<td>0.63</td>
</tr>
<tr>
<td>x8</td>
<td>RL-RAIL-OUT FRAME</td>
<td>2000001</td>
<td>2.28</td>
<td>1.442</td>
<td>1.792</td>
<td>0.63</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td>6.571</td>
<td>8.881</td>
<td>6.24</td>
<td></td>
</tr>
</tbody>
</table>

The acceleration values of each time series of case 2.

Table 3
<table>
<thead>
<tr>
<th>Time series (s)</th>
<th>Acceleration I (km/s²)</th>
<th>Acceleration II (km/s²)</th>
<th>Acceleration III (km/s²)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.256–0.375</td>
<td>0.793</td>
<td>0.345</td>
<td>0.112</td>
</tr>
<tr>
<td>0.401–0.582</td>
<td>0.345</td>
<td>0.112</td>
<td>0.256</td>
</tr>
<tr>
<td>0.645–0.832</td>
<td>0.345</td>
<td>0.112</td>
<td>0.256</td>
</tr>
</tbody>
</table>

The intervals of design variables and corresponding reduced intervals.

Table 4
<table>
<thead>
<tr>
<th>Design variables</th>
<th>Component name</th>
<th>Component ID</th>
<th>Thickness (mm)</th>
<th>Internal energy I (kJ)</th>
<th>Internal energy II (kJ)</th>
<th>Mass (kg)</th>
</tr>
</thead>
<tbody>
<tr>
<td>x1</td>
<td>BMP-BUMPER-FT</td>
<td>20000001</td>
<td>2.28</td>
<td>1.442</td>
<td>1.675</td>
<td>1.04</td>
</tr>
<tr>
<td>x2</td>
<td>RF-BOTTOMFRAME</td>
<td>2000052</td>
<td>2.40</td>
<td>1.642</td>
<td>1.792</td>
<td>0.63</td>
</tr>
<tr>
<td>x3</td>
<td>RL-FCROSS-TOP</td>
<td>2000050</td>
<td>3.20</td>
<td>1.442</td>
<td>1.792</td>
<td>0.63</td>
</tr>
<tr>
<td>x4</td>
<td>RF-TOPFRAME</td>
<td>2000014</td>
<td>3.00</td>
<td>1.442</td>
<td>1.792</td>
<td>0.63</td>
</tr>
<tr>
<td>x5</td>
<td>RL-RAIL-INN CFRONT</td>
<td>2000026</td>
<td>3.16</td>
<td>1.442</td>
<td>1.792</td>
<td>0.63</td>
</tr>
<tr>
<td>x6</td>
<td>RL-RAIL-INN CFRONT</td>
<td>2000052</td>
<td>3.16</td>
<td>1.442</td>
<td>1.792</td>
<td>0.63</td>
</tr>
<tr>
<td>x7</td>
<td>RL-RAIL-OUT FRAME</td>
<td>2000001</td>
<td>2.28</td>
<td>1.442</td>
<td>1.792</td>
<td>0.63</td>
</tr>
<tr>
<td>x8</td>
<td>RL-RAIL-OUT FRAME</td>
<td>2000001</td>
<td>2.28</td>
<td>1.442</td>
<td>1.792</td>
<td>0.63</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td>7.229</td>
<td>9.711</td>
<td>5.85</td>
<td></td>
</tr>
</tbody>
</table>

where $W_i$ represents the weight for the $i$th objective function with the additional requirements. Furthermore, the objective function is composed of two types, energy and acceleration. Thus, $W_i$ is subject to

$$
W_1 + W_2 = 0.5
$$

$$
W_3 + W_4 + W_5 = 0.5
$$

(26)

The relationship of weights are determined by

$$
W_1 : W_2 = f_1(X) : f_2(X)
$$

$$
W_3 : W_4 : W_5 = f_3(X) : f_4(X) : f_5(X)
$$

(27)

The acceleration values of each time series of case 2.

Table 5
<table>
<thead>
<tr>
<th>Time series (s)</th>
<th>Acceleration I (km/s²)</th>
<th>Acceleration II (km/s²)</th>
<th>Acceleration III (km/s²)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.256–0.375</td>
<td>0.793</td>
<td>0.345</td>
<td>0.112</td>
</tr>
<tr>
<td>0.401–0.582</td>
<td>0.345</td>
<td>0.112</td>
<td>0.256</td>
</tr>
<tr>
<td>0.645–0.832</td>
<td>0.345</td>
<td>0.112</td>
<td>0.256</td>
</tr>
</tbody>
</table>

Table 6
<table>
<thead>
<tr>
<th>Time series (s)</th>
<th>Mean value Acceleration I (km/s²)</th>
<th>Mean value Acceleration II (km/s²)</th>
<th>Mean value Acceleration III (km/s²)</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Estimated by metamodel</td>
<td>−0.672</td>
<td>−0.243</td>
<td>0.117</td>
<td>7.824</td>
</tr>
<tr>
<td>Simulated by FE model</td>
<td>−0.625</td>
<td>−0.231</td>
<td>0.112</td>
<td>7.229</td>
</tr>
<tr>
<td>Error</td>
<td>7.52%</td>
<td>5.19%</td>
<td>4.47%</td>
<td>8.23%</td>
</tr>
</tbody>
</table>

4.2.3. Optimization procedure and results

Firstly, 24 initial samples are generated by the LHS. The initial design variables and corresponding intervals are listed in Table 4. Twenty-two samples are generated by the BBNS sampling strategy. To verify the optimum solution, the full FE model is performed with the optimum design variable, the corresponding objective function values evaluated by the metamodel and FE simulation are listed in Table 6.

The internal energy of optimized components at 20 ms, 40 ms are given in Table 4 along with each component thickness. A comparison of the initial and optimum design variables with respect to mass, the internal energy absorption and acceleration in sensitive series are shown in Fig. 14. A comparison of acceleration histories for the original and optimum designs is shown in Fig. 15. According to Fig. 15, maximum acceleration occurs in the time period 0.256–0.375 ms as mentioned before. The proposed method successfully decreases the maximum acceleration in this sensitive time series. We notice that the acceleration of other sensitive time series is also well controlled. The optimized acceleration time curve is smoother than the original one. Even if the maximum acceleration in 0.401–0.582 ms increases, this increase can be accepted compared with the decrease of the other time periods. Additionally, the optimum design has an increased energy absorption capacity at all measured instances and decreased peak acceleration without any increase in the total vehicle mass (Fig. 16).

5. Conclusions

In this study, a time-based framework for the crashworthiness optimization of vehicle structures has been presented and demonstrated with the frontal crash problem with full FE model. The major advantages of the proposed method are summarized as follows.

1. The distinctive characteristic of the proposed method is time-based mode. Objective functions of the time-based strategy are based on time history of the entire process and more useful information can be collected and involved in the constructed metamodels. The cumulation extraction strategy is proposed to extract the sensitive time series based on the entire simulation procedure. Consequently, single-objective functions such as the maximum acceleration and internal energy should be transformed to multi-objective functions based on the time series.

2. To improve the efficiency of the proposed metamodeling technique, the BBNS intelligent sampling strategy is used to generate samples. The BBNS generates new samples according to the boundary information and best neighbor, size of samples and design space can be well controlled.

3. According to the time-based strategy, more information of the entire time history can be employed to construct the metamodel. In order to construct more robust and accurate results, the LS-SVR is used to construct the metamodels in this study. Compared with other popular metamodeling techniques, the LS-SVR is based on the ERM and SRM. Thus, the robustness metamodel and the result of the crashworthiness design should be achieved.

Finally, a successful implementation of a multi-objective optimization by the proposed method demonstrates that the vehicle performance is improved without an increase in the vehicle mass, which is a major consideration in design and manufacturing of automobile bodies. Additionally, the proposed optimization method needs to be studied further and should be extended to apply for other nonlinear optimization problems.

Acknowledgments

This work is supported by the National 973 Program of China under the grant number 2010CB328005; the Project of National Science Foundation of China (NSFC) under the grant number 2010CB328005; the Program for Changjiang Scholars and Innovative Research Team in University and the State Key Laboratory of Advanced Technology for vehicle design and manufacture.

References


