

DETC2006-99485

AN EFFICIENT ALGORITHM FOR VEHICLE CRASHWORTHINESS DESIGN VIA CRASH MODE MATCHING

Karim Hamza and Kazuhiro Saitou[§]
Department of Mechanical Engineering
University of Michigan
Ann Arbor, Michigan, USA
{khamza, kazu}@umich.edu

ABSTRACT

This paper presents an efficient algorithm for developing vehicle structures for crashworthiness, based on the analyses of *crash mode*, a history of the deformation of the different structural zones during a crash event. It emulates a process called crash mode matching where structural crashworthiness is improved by manually modifying the design until its crash mode matches the one the designers deem as optimal. Given an initial design and a desired crash mode, the algorithm iteratively finds a new design whose crash mode is increasingly closer to the desired one. At each iteration, a new design is chosen as the best among the normally distributed samples near the current design, whose mean and standard deviation are adjusted by a set of fuzzy rules. Each fuzzy rule encapsulates elementary knowledge of manual crash mode matching, as a mapping from the differences between the current and desired crash modes, to the changes in mean and standard deviation for sampling a sizing parameter in a structural zone. A case study on a vehicle frontal crash demonstrated the algorithm outperformed the conventional methods both in design quality and computational time.

1. INTRODUCTION

Despite the recent advent of technologies such as anti-lock braking, active steer, and collision mitigation, enhancing vehicle crashworthiness remains a high priority in helping to protect occupants during crashes. Designing vehicle bodies for superior crashworthiness is a difficult task, since a body structure needs to be strong in some parts to minimize deformation near the passenger compartment, yet compliant in other parts to absorb the impact energy. Oftentimes, the

outcome of design modifications can be difficult to predict due to the inherent high nonlinearity of a dynamic crash event.

Instead of costly and time consuming crash tests using physical prototypes, finite element (FE) crash simulations are widely used for exploring design alternatives. FE crash simulations, however, require massive computational resources, making them difficult to use with optimization algorithms. Although computer speed continues to increase, higher detailed FE models are being constructed for better accuracy, creating ever increasing demands for more computer resources. Nowadays, detailed FE models for crash simulations commonly exceed one million DOFs. Tuning such complex nonlinear systems for a desired dynamic behavior (*i.e.*, crashworthiness) poses a truly unique challenge. The use of surrogate models [1] only partially solve the issue, since training a high fidelity surrogate requires a large number of samples FE runs. As such, the need is imminent for an efficient algorithm that can achieve high crash performance with a far fewer number of FE crash simulations than the conventional optimization algorithm.

Experienced vehicle designers, on the other hand, can vastly improve the crash performance of a vehicle structure with fewer FE crash simulations, by observing the *crash modes* of the structure. A crash mode is a time history of the deformation (such as axial crushing, twisting, and transversal bending) in the different structural zones during a crash event. Viewing crash mode as a strategy for energy absorption, a process commonly called *crash mode matching* can be employed where the design is manually modified until its crash mode matches the one the designers deem as optimal based on their experiences.

[§] Corresponding Author

As an example of the crash mode matching, consider a front frame of a vehicle subject to frontal impact in Fig. 1, where the crash modes of two different designs are illustrated as sequences of figures. The objective is to minimize the deformation in zone 2, close to the passenger compartment. Design A in Fig. 1 (b) exhibits CM_1 , where zone 1 fully deforms first and then zone 2 partially deforms. Design B in Fig. 1 (c) exhibits CM_2 , where zone 1 only partially deforms, followed by severe bending in zone 2. Between the two crash modes, CM_1 absorbs the same energy with less deformation in zone 2 than CM_2 , due to the occurrence of axial crushing (which tends to absorb more energy than twisting or transversal bending) immediately after the impact. Without additional FE simulations, the designer would simply try to match the crash mode of design B to CM_1 , by allocating less material to zone 1 (so it would deform more easily), and more material to zone 2 (so it would not bend as much). To demonstrate how these changes to design B indeed improves its crash performance, the deformation in zone 2 was plotted in Fig. 2 as a function of the material fraction in zone 1, using LS-Dyna [2]. The plot confirms design A is indeed better than design B. With the division between CM_1 and CM_2 at about 30% mass to zone 1, the plot also confirms the above changes to design B will bring its crash mode closer to CM_1 and improve its crash performance.

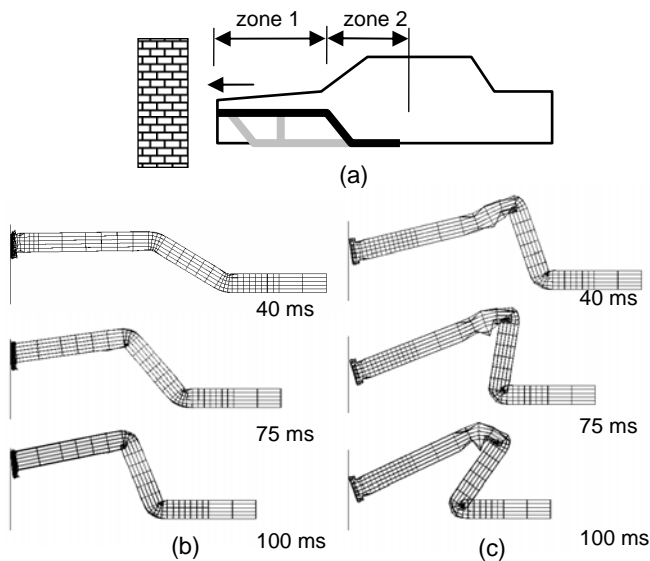


Fig. 1 Vehicle mid-rail subject to frontal impact. (a) vehicle schematic, (b) Design A exhibiting crash mode CM_1 and (c) Design B exhibiting crash mode CM_2

Since one FE crash simulation of a vehicle takes hours if not days to complete, the crash mode matching process occurs over several meetings. At a meeting, a team of designers analyze the animation of FE crash simulation and suggest several design changes, and the next meeting is called when the simulation results with the suggested changes are completed. The recurrent meetings are held until a high quality design is obtained, which typically happens after several meetings.

Visual observation of crash mode in the animation is essential to the crash mode matching process, as it gives the designers clues as to where to make changes (strengthen or weaken) in different zones of the structure.

Despite its effectiveness in practice, the crash mode matching remains more of an art than a systematic procedure, due to the difficulty in formalizing 1) the identification of the desired crash mode and 2) the design changes to attain the desired crash mode. As our first step towards the automated crash mode matching, we have previously proposed the computational identification of a high-quality crash mode using a reduced order dynamic model [3-6], followed by the manual (non-automated) crash mode matching to the identified crash mode. Although the results were very promising, the automation of the crash mode matching remained as future work in [3-6], which is addressed in this paper.

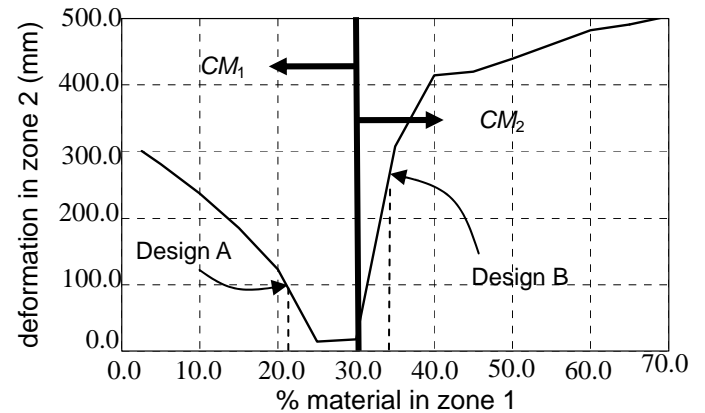


Fig. 2 Deformation in zone 2 in Fig. 1 (a) as function of percentage material allocated to zone 1

The crash mode matching algorithm presented in this paper takes as inputs an initial design and a desired crash mode, and iteratively finds a new design whose crash mode is increasingly closer to the desired crash mode. At each iteration, a new design is chosen as the best one among a small number of normally distributed samples near the current design, whose mean and standard deviation are adjusted by a set of fuzzy rules [7]. Each fuzzy rule encapsulates elementary knowledge of manual crash mode matching, as a mapping from the differences between the current and desired crash modes, to the changes in mean and standard deviation for sampling a sizing parameter in a structural zone. The algorithm stops upon attaining a desired crash performance or after a pre-set number of iterations.

The paper started with a motivation behind the development of the proposed algorithm. The following sections provide a review of relevant literature, details of the algorithm and a case study of a vehicle model subjected to a full-overlap frontal crash condition.

2. RELATED WORK

The algorithm presented in this paper is for sizing design of vehicle structures for crashworthiness. While numerous works are found in this area, no previous work has addressed computational algorithms for crashworthiness design driven by crash modes.

2.1. Optimization with surrogate models

For sizing optimization for crashworthiness, methods involving the use of phenomenological surrogate models are the most dominant (eg., [8-17]). Surrogate models can provide a convenient way of approximating the mapping of inputs (design variables) to outputs (objectives and/or constraints), when the nonlinearity of the underlying physics is modest to minimum. Since crash phenomena are highly nonlinear, the use of a surrogate model is limited to small ranges of design variables where “jumps” in the outputs, such as the one at the boundary of CM_1 and CM_2 in Fig.2, do not occur. If design changes are large enough to span multiple crash modes, the accuracy deteriorates at the boundaries of crash modes, where there is always “jumps” in crash performances. Since the crash mode matching always involves the design changes spanning multiple crash modes, surrogate models are unsuitable for use with the manual crash mode matching and the present algorithm.

2.2. Optimization with reduced physics-based models

Another contender to sizing optimization for crashworthiness is the family of reduced order physics-based models. Examples include coarse-mesh FE, lumped parameter, and lattice models [18-25]. While these models can be computationally inexpensive and also bear some physical roots in the underlying crash phenomena, their level of accuracy does not allow them to be a complete substitute to detailed FE models. With a “right” level of abstraction preserving general geometries, however, these models would be capable of simulating the crash modes of vehicle structures with realistic topology. In our previous work [4-6], we demonstrated that a reduced order linkage model of vehicle structure, called an equivalent mechanism (EM) model, could in fact simulate the crash modes with a reasonable accuracy. We also demonstrated the efficiencies of manual (non-automated) crash mode matching processes guided by the crash mode of the optimized EM model. The automated crash mode matching algorithm presented in this paper assumes a desired crash mode as a given input, which can be obtained by using the optimized EM model and/or based on the designer’s experience.

3. CRASH MODE MATCHING ALGORITHM

Given a parameterized geometry of a structure, an initial design and a desired crash mode, the proposed algorithm iteratively finds a new design whose crash mode is increasingly closer to the desired crash mode. At each iteration, a new design is chosen as the best one among a small number of normally distributed samples near the current design, whose

mean and standard deviation are adjusted based on the differences between the current and desired crash modes by using fuzzy rules. The details of the algorithm are presented in the following subsections.

3.1. Crash mode

A crash mode of a structure is a time history of deformation (such as axial crushing, twisting, and transversal bending) in different structural zones during a crash event. It is defined as a $3 \times m$ matrix of functions of time:

$$CM(\mathbf{x}) = (cm_{ij}(t, \mathbf{x})); \quad i=1, 2, 3; \quad j=1, \dots, m \quad (1)$$

where $cm_{ij}(t, \mathbf{x}) \in [0,1]$ is the normalized deformation of collapsing type i at structural zone j of structure with size \mathbf{x} , as a function of normalized time $t \in [0, 1]$. The collapsing type can be either axial crush (= 1), bend (= 2), and side squish (= 3). The structural zones are typically chosen to coincide with major structural members. Fig. 3 shows an example of $cm_{ij}(t, \mathbf{x})$ corresponding to bending of a structural member, obtained by a FE crash simulation. During a crash event, the collapsing of a structural member typically remains minimal until it suddenly rises at some point and quickly reaches a steady state. This behavior is well approximated by a step function, as also shown in Fig. 3.

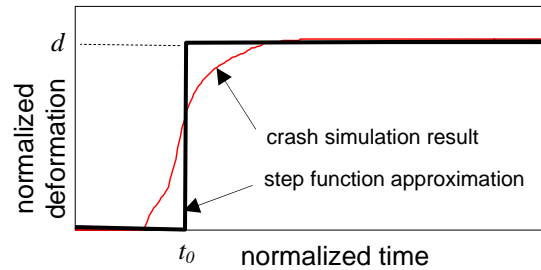


Fig. 3 Example of crash mode $cm_{ij}(t, \mathbf{x})$ obtained by FE crash simulation and its approximation as a step function.

Accordingly, a desired crash mode CM^* , an input to the proposed algorithm, is defined as a $3 \times m$ matrix of step functions:

$$CM^* = (cm^*_{ij}(t)); \quad i=1, 2, 3; \quad j=1, \dots, m \quad (2)$$

where $cm^*_{ij}(t) = d \cdot \theta(t-t_0)$, d and t are the magnitude and start time of step as in Fig 3, respectively, and $\theta(t)$ is the unit step function:

$$\theta(t) = \begin{cases} 0 & \text{if } t < 0 \\ 1 & \text{otherwise} \end{cases} \quad (3)$$

Crash mode metric, the differences between $CM(\mathbf{x})$ and CM^* , are defined as a $3 \times m$ matrix of the normalized differences:

$$\Delta(\mathbf{x}) = (\delta_{ij}(\mathbf{x})); \quad i=1, 2, 3; \quad j = 1, \dots, m \quad (4)$$

where $\delta_{ij}(\mathbf{x}) \in [0,1]$ is the normalized difference between $cm_{ij}(t, \mathbf{x})$ and $cm_{ij}^*(t)$ over $t \in [0,1]$, given as:

$$\delta_{ij}(\mathbf{x}) = \frac{\int_0^1 \{cm_{ij}(t, \mathbf{x}) - cm_{ij}^*(t)\} dt}{\int_0^1 cm_{ij}^*(t) dt} \quad (5)$$

Fig. 4 shows examples of crash mode $cm_{ij}(t, \mathbf{x})$ and desired crash mode $cm_{ij}^*(t)$ corresponding to various values of crash mode metric $\delta_{ij}(\mathbf{x})$. With the deformation starting approximately at $t = t_0$, a zero value of $\delta_{ij}(\mathbf{x})$ means the amount of deformation of the current design matches well with the desired crash mode (Fig. 4 (a)), and negative and positive values mean smaller (Fig. 4 (b)) and larger (Fig. 4 (c)) deformations than the desired crash mode, respectively.

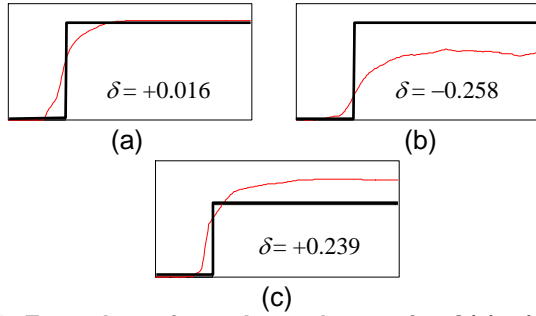


Fig. 4 Examples of crash mode metric $\delta_{ij}(\mathbf{x})$: (a) well matched deformation, (b) too little deformation, (c) too much deformation.

Although crash mode metric $\Delta(\mathbf{x})$ contains $3m$ elements $\delta_{ij}(\mathbf{x})$ for each deforming type and each structural zone, some of them can be ignored or combined if their contributions to the crash performance are minor. For example, the axial crush and bend of an insignificant structural member may be combined into one equivalent value for “deformation.” Another example is side-squish, which can be often ignored except for certain structural zones such as the bumper area. Appendix A lists the reduced $\delta_{ij}(\mathbf{x})$ used in the following case study.

3.2. Algorithm

The crush mode matching algorithm solves the following optimization problem:

$$\begin{aligned} & \text{minimize} \quad \{f(\mathbf{x}), \Delta(\mathbf{x})\} \\ & \text{subject to} \quad \mathbf{x}_l \leq \mathbf{x} \leq \mathbf{x}_u \end{aligned}$$

where $f(\mathbf{x})$ is a vector of the design objectives to be minimized (eg., structural weight) and the constraint violations (eg., intrusion beyond maximum allowable limit), $\Delta(\mathbf{x})$ is the crash mode metric as defined in Equations 4 and 5, and \mathbf{x}_l and \mathbf{x}_u are upper and lower bounds of design variable \mathbf{x} , respectively.

The algorithm is essentially adaptive neighborhood sampling. Starting with an initial design \mathbf{x}_0 , it iteratively samples the new design within a neighborhood of the current design \mathbf{x} . At each iteration, the new design is chosen as the best design with respect to $f(\mathbf{x})$ and $\Delta(\mathbf{x})$, among n_s samples $\mathbf{x}_s \sim \mathbf{N}(\boldsymbol{\mu}, \boldsymbol{\sigma})$ whose mean $\boldsymbol{\mu}$ and standard deviation $\boldsymbol{\sigma}$ are adjusted based on $\Delta(\mathbf{x})$. The following outlines each step of the algorithm:

match-crash-mode(\mathbf{x}_0)

1. $\mathbf{x} \leftarrow \mathbf{x}_0$ and $O \leftarrow \{\mathbf{x}_0\}$.
2. evaluate $f(\mathbf{x})$ and $\Delta(\mathbf{x})$ by FE crash simulation.
3. $(\boldsymbol{\mu}, \boldsymbol{\sigma}) \leftarrow \text{adjust-neighborhood}(\mathbf{x}, \Delta(\mathbf{x}))$.
4. for each $i = 1, \dots, n_s$, sample $\mathbf{x}_i^s \sim \mathbf{N}(\boldsymbol{\mu}, \boldsymbol{\sigma})$ and $\mathbf{x}_l \leq \mathbf{x}_i^s \leq \mathbf{x}_u$, and evaluate $f(\mathbf{x}_i^s)$ and $\Delta(\mathbf{x}_i^s)$ by FE crash simulation.
5. $O \leftarrow O \cup \{\mathbf{x}_1^s, \mathbf{x}_2^s, \dots, \mathbf{x}_{n_s}^s\}$ and remove from O all dominated and/or infeasible designs with respect to $f(\mathbf{x})$.
6. $\mathbf{x} \leftarrow \arg \min_{k \in \{1, \dots, n_s\}} \sum_{ij} \delta_{ij}(\mathbf{x}_k^s)$
7. if the maximum number of iteration has reached, return O . Otherwise, go to step 2.

In line 6, if all samples $\mathbf{x}_1^s, \mathbf{x}_2^s, \dots, \mathbf{x}_{n_s}^s$ becomes virtually indistinguishable in term of $\sum \delta_{ij}(\mathbf{x}_k^s)$ towards the end of the run, the next iterate can be selected based on the values of $f(\mathbf{x})$.

Subroutine `adjust-neighborhood`($\mathbf{x}, \Delta(\mathbf{x})$) in line 3 adjusts $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ as outlined below:

`adjust-neighborhood`($\mathbf{x}, \Delta(\mathbf{x})$)

8. $\boldsymbol{\mu} \leftarrow \mathbf{x}$, $\boldsymbol{\sigma} \leftarrow \mathbf{0}$.
9. for each $k = 1, \dots, n_r$, $(l, \tilde{a}) \leftarrow r_k(\Delta(\mathbf{x}))$, $\mu_l \leftarrow \mu_l + \tilde{a}$ and $\sigma_l \leftarrow \sigma_l + |\tilde{a}|$.
10. $\boldsymbol{\sigma} \leftarrow \max\{\boldsymbol{\sigma}_{min}, \boldsymbol{\sigma}\}$.
11. return $(\boldsymbol{\mu}, \boldsymbol{\sigma})$.

In line 8, $r_k : [0,1]^{[1,2,3] \times \{1, \dots, m\}} \mapsto \{1, \dots, n\} \times \mathbf{R}$ is the k -th fuzzy rule [7] of the following form:

$$\begin{aligned} & \text{If} \quad \bigwedge_{(i,j) \in C_k} \delta_{ij}(\mathbf{x}) \in FS_{ijk} \\ & \text{then adjust the } l_k\text{-th variable by } a_k \end{aligned}$$

where $C_k \subseteq \{1,2,3\} \times \{1, \dots, m\}$, $l_k \in \{1, \dots, n\}$, $a_k \in \mathbf{R}$, and FS_{ijk} is one of five fuzzy sets, *NH* (highly negative), *NL* (lightly negative), *Z* (almost zero), *PL* (lightly positive), or *PH* (highly positive), whose membership functions $u_{FS} : [0,1] \mapsto [0,1]$ are shown in Fig. 5. For all five fuzzy sets, standard sigmoid membership functions [7] are used. Based on the membership values of $\delta_{ij}(\mathbf{x})$ in the “If” part of the rule, adjustment a_k is scaled to produce the actual adjustment value \tilde{a}_k as follows:

$$\begin{aligned} \tilde{a}_k &= w_k a_k \\ w_k &= \min_{(i,j) \in C_k} \{u_{FS_{ijk}}(\delta_{ij}(\mathbf{x}))\} \end{aligned} \quad (6)$$

For example, consider one of the fuzzy rules used in the following case study:

If $\delta_{12}(\mathbf{x}) \in NH \wedge \delta_{22}(\mathbf{x}) \in PL$
then adjust the 2nd variable by +0.15.

Since the 2nd variable x_2 is the plate thickness in zone 2, this rule translates to:

“**If** axial crush in zone 2 is too little and bend in zone 2 is much, **then** adjust the plate thickness in zone 2 by +0.15.”

If $\delta_{12}(\mathbf{x}) = 0.0$ and $\delta_{22}(\mathbf{x}) = 0.3$, for example, Fig. 5 tells the membership value of $\delta_{12}(\mathbf{x})$ to *NH* is $u_{NH}(\delta_{12}(\mathbf{x})) \approx 0.05$ and the membership value of $\delta_{22}(\mathbf{x})$ to *PL* is $u_{PL}(\delta_{22}(\mathbf{x})) \approx 0.8$. Since $\min\{0.05, 0.8\} = 0.05$, the actual adjustment value is $\tilde{a} = 0.05 \times 0.15 = 0.0075$. Note that this value is used to adjust μ_2 and σ_2 , rather than x_2 itself.

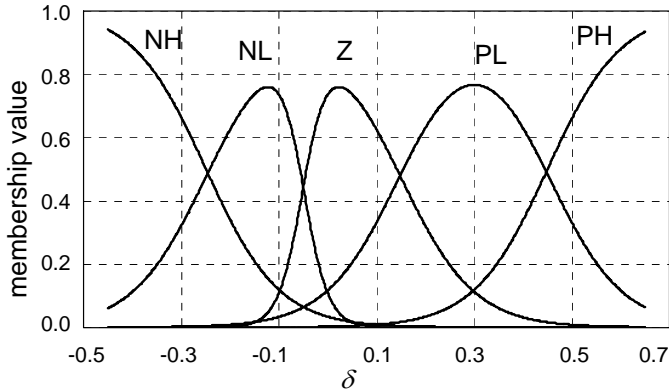


Fig. 5. Membership functions for fuzzy sets.

As seen above, each fuzzy rule encapsulates elementary knowledge of manual crash mode matching with the number of terms in the if-clause of a rule, $|C_k|$, typically less than 2. The rules can be classified to two types:

- If a zone deforms too much (little) relative to the desired crash mode, increase (decrease) the plate thicknesses in the zone.
- If a zone is bending too much relative to the desired crash mode, increase the height of the member cross sections in the zone.

Given structural geometry, defining rules of these types is relatively a straightforward task, and does not require significant expert knowledge (which of course would help).

In line 4, the candidates for the next iterate are sampled from normal distribution in order to guarantee the connectivity

of the neighborhoods during the iterations. This, together with the preservation of the best encountered solutions in line 5, satisfies the global convergence criteria for a local neighborhood search [26], *i.e.*, the algorithm has a non-zero probability of finding a global solution within a finite number of iterations. The actual efficiency of the algorithm, of course, depends on the quality of the fuzzy rules.

4. CASE STUDY

4.1. Scenario

This section describes a case study on a front half-body model of a mid-sized vehicle, subjected to full-overlap frontal crash against a rigid barrier (Fig. 6). The model has the following specifications:

- All main structural members are hollow box-section
- The engine and power train are represented as a rigid box of mass 250 kg, connected to the engine mounting points via stiff beams.
- The rest of the vehicle mass (600 kg) is represented as a lumped mass connected to the structure via stiff beams.
- Crash speed is 15.6 m/s (35 mph)
- Coefficient of friction at the rigid barrier is 0.3
- Material model is elastic-plastic for mild steel

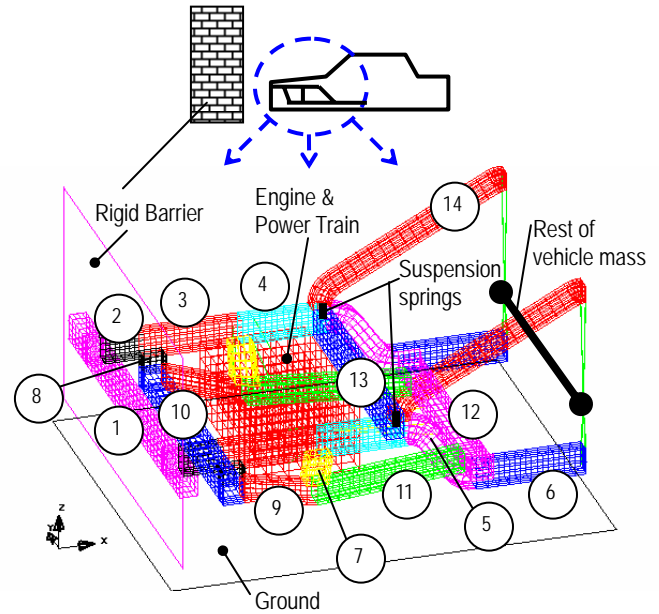


Fig. 6. FE model of a vehicle subjected to full-overlap frontal crash.

There are 4 continuous and 14 discrete design variables:

- h_1, b_1 [mm]: height and width of the box-section of upper rails and cross members (continuous in [50, 150]).
- h_2, b_2 [mm]: height and width of the box-section of lower rails and cross members (continuous in [50, 150]).

- t_1, \dots, t_{14} [mm]: sheet metal thickness of the box-section, for structural zones 1 through 14 as indicated in Fig. 6 (discrete in $\{0.6, 0.8, 1.0, \dots, 4.2, 4.4\}$).

The design objective is to minimize the structural mass, subject to the constraints on the passenger safety:

- f [kg]: structural weight, to be minimized
- $g_1 < 100$ [mm]: intrusion into passenger compartment
- $g_2 < 30$ [G]: acceleration at passenger compartment

Attempts to tackle this problem were performed in [4, 6] via direct application genetic algorithm (GA) [27] to the FE model, and via manual (non-automated) crash mode matching. For manual crash mode matching, the desired crash mode is obtained by the optimization of a reduced order dynamic model (equivalent mechanism model [3-6]). This case study uses the desired crash mode in [6] as CM^* , in order to compares the results of manual and automated crash mode matching.

A total of 19 crash mode metrics are chosen as listed in Appendix A, which is summarized as:

- Side squish in zones 1 and 10 (bumper and front cross bar).
- Axial and bending deformations in zones 2, 3, 5, 9 and 11 (zones with significant deformations in CM^*).
- Combined values for axial and bending for zones 4, 6, 7, 8, 12, 13 and 14 (zone with small deformations in CM^*).

In the rest of the section the 19 crash mode metrics are simply denoted by a single subscript as $\delta_1, \delta_2, \dots, \delta_{19}$ for notational simplicity. Additionally, a total of 238 fuzzy rules are defined, which are listed in Appendix B in a tabulated form.

The initial design x_0 is the same as in [6]. Fig. 7 shows CM^* , $CM(x_0)$, and $\Delta(x_0)$. The minimum value of standard deviation σ_{\min} in adjust-neighborhood (line 11) was set to 2.0 mm for h_1, b_1, h_2 and b_2 and to 0.2 mm for t_1, \dots, t_{14} .

4.2. Results

Table 1 provides a comparison of the results of direct GA, manual crash mode matching, optimization via a response surface model (RMS), and the proposed algorithm. Four runs of the proposed algorithm were performed, of which the best and worst are shown in Table 1. The CM^* and the $CM(x)$ and $\Delta(x)$ of the best design found by the proposed algorithm are shown in Fig. 8. The best obtained result by the proposed algorithm has good match to the design targets of the desired crash mode (Fig. 8), which is reflected both in graphic observation of deformation history, as well as the crash mode metrics δ_1 through δ_{19} . An overall summary of results is provided in Fig. 9.

Despite the anticipation of poor performance, a response surface model was attempted anyway to complete the study since it is currently a dominant automation approach. Two randomized L_{54} orthogonal arrays [28] (total 108 samples) were

used to construct the RSM model, and optimization was performed on the constructed RSM. However, the result obtained from RSM was infeasible to both constraints.

Directly linking the FE model to genetic algorithm (GA) was successful in attaining a fairly good and feasible design. However, the computational time required for the direct application of GA were enormous (350 hours) compared to other methods (55-75 hours), rendering it impractical if more detailed vehicle models were considered.

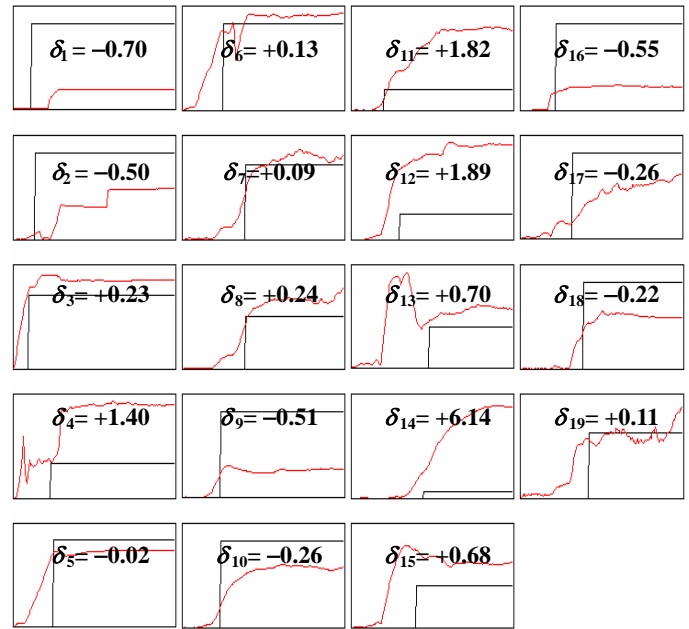


Fig. 7 CM^* , $CM(x_0)$, and $\Delta(x_0)$.

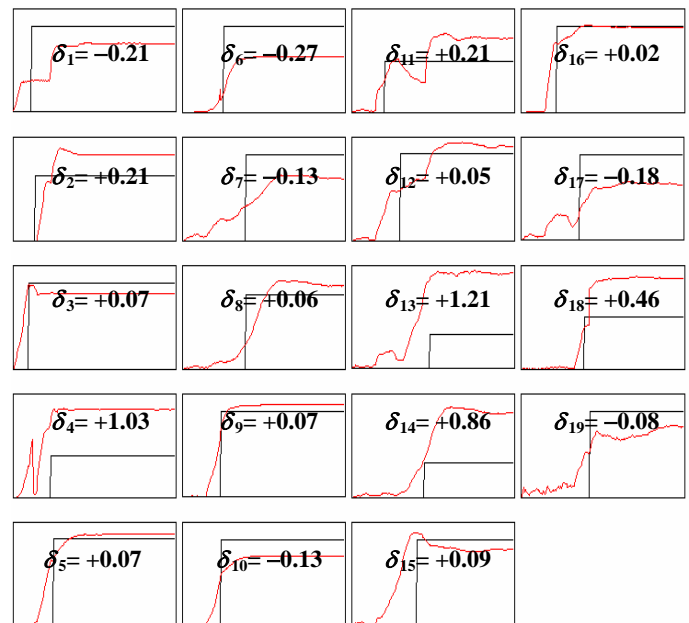


Fig. 8. CM^* and $CM(x)$ and $\Delta(x)$ of the best design found by the proposed algorithm.

The design obtained by manual crash mode matching in [6] remains the current best known solution to this case study. The best result by the proposed algorithm among the four runs is feasible and only 0.9 kg heavier than the best known result. The worst result by the proposed algorithm was still better than the one obtained by GA. The computational time for the proposed algorithm (75 hours) is approximately one fifth of those of GA (350 hours) and about the same as RSM (75 hours).

5. CONCLUSION AND FUTURE WORK

This paper presented a computer algorithm for crush mode matching, a process which was previously possible only by using qualitative judgment and visual observations of the history of structural deformation. The performance of the algorithm was demonstrated by a case study on a vehicle model subjected to full overlap frontal crash. Future research would include testing the algorithm on more detailed vehicle models, and the automatic generation of baseline fuzzy rules through the optimization history of an equivalent mechanism model.

ACKNOWLEDGMENTS

The vehicle model in Fig 6 was developed under the support of Nissan Technical Center North America (NTCNA), Inc. The authors acknowledge the discussions and suggestions offered by Mr. Ken Jimbo and Mr. Akira Toyama at NTCNA during the model development.

Table 1. Case study results by different techniques (results with asterisk * are infeasible).

	Base-line*	Direct GA	RSM*	Manual CM [6]	Auto CM – Best	Auto CM – Worst
h_1 [mm]	60.0	114.0	94.0	88.0	83.0	94.0
b_1 [mm]	96.0	67.0	66.0	80.0	77.0	77.0
h_2 [mm]	67.0	69.0	50.0	60.0	54.0	68.0
b_2 [mm]	50.0	95.0	75.0	50.0	56.0	105.0
t_1 [mm]	4.0	2.2	2.2	3.2	2.6	3.2
t_2 [mm]	1.0	2.0	2.0	2.4	1.6	2.2
t_3 [mm]	1.0	1.8	1.8	2.0	2.2	2.0
t_4 [mm]	3.4	2.8	3.0	2.8	4.6	3.4
t_5 [mm]	3.8	3.0	3.2	4.2	4.4	3.0
t_6 [mm]	2.2	3.0	3.0	2.8	3.2	2.6
t_7 [mm]	4.0	1.2	1.2	3.2	3.4	2.0
t_8 [mm]	2.2	2.2	2.2	2.0	1.6	1.6
t_9 [mm]	4.0	2.6	2.6	2.4	2.2	1.8
t_{10} [mm]	1.8	2.8	2.8	2.6	2.6	3.2
t_{11} [mm]	1.8	3.0	2.8	1.2	1.8	1.8
t_{12} [mm]	3.8	2.4	2.4	2.8	3.0	3.0
t_{13} [mm]	3.0	2.0	2.0	1.8	2.0	0.6
t_{14} [mm]	2.0	2.2	2.4	2.2	2.2	2.0
f [kg]	69.2	73.1	61.7	66.9	68.8	72.6
g_1 [mm]	324.0	62.0	142.0	76.0	97.0	89.0

g_2 [G]	28.0	25.9	31.0	29.4	26.8	27.4
# FE runs		500	108	10	50	50
# EM runs		–	–	500	500	500
Comp. time [hr]		350	75	55	75	75

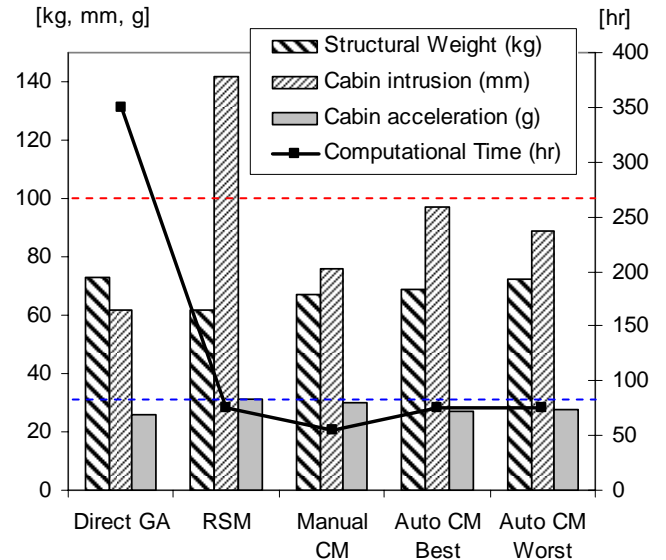


Fig. 9. Summary of case study results. Two horizontal dashed lines indicate the maximum allowable intrusion (100 mm) and maximum cabin acceleration (30 g).

REFERENCES

- [1] Simpson, T., Booker, A., Ghosh, D., Giunta, A., Koch, P. and Yang, R., 2004, "Approximation Methods in Multidisciplinary Analysis and Optimization: A Panel Discussion," *Structural and Multidisciplinary Optimization*, **27**: 302-313.
- [2] LSTC, 2001, *LS-DYNA Software Manuals*, Livermore Software Technology Corporation, Livermore, CA, USA.
- [3] Hamza, K. and Saitou, K., 2005, "Design for Structural Crashworthiness using Equivalent Mechanism Approximations," *Journal of Mechanical Design*, **127**(3): 485-492.
- [4] Hamza, K. and Saitou, K., 2004, "Crash Mode Analysis of Vehicle Structures based on Equivalent Mechanism Approximations," Proc. *5th International Symposium on Tools and Methods of Competitive Engineering*, Lausanne, Switzerland, April 13 - 17, pp. 277-287.
- [5] Hamza, K. and Saitou, K., 2004, "Design for Crashworthiness of Vehicle Structures via Equivalent Mechanism Approximations and Crash Mode Matching," Proc. *ASME 2004 International Mechanical Engineering Congress and Exposition*, November 13-20, Anaheim, CA, IMECE2004-62226.
- [6] Hamza, K. and Saitou, K., "Design for Crashworthiness of Vehicle Structures via Crash Mode Matching and

- Equivalent Mechanism Approximations,” Submitted to *Journal of Mechanical Design*.
- [7] Hopgood, A. A., 2001, *Intelligent Systems for Engineers and Scientists*, 2nd Edition, CRC Press, New York, USA.
- [8] Yang, R. J., Tho, C. H., Wu, C. C., Johnson, D. and Cheng, J., 1999, “A Numerical Study of Crash Optimization,” *Proc. ASME 1999 Design Engineering and Technical Conferences*, September 12-15, Las Vegas, Nevada, DETC99-DAC8590.
- [9] Shi, Q. Hagiwara, I. and Takashima, F., 1999, “The Most Probable Optimal Design Method for Global Optimization,” *Proc. ASME 1999 Design Engineering and Technical Conferences*, September 12-15, Las Vegas, Nevada, DETC99-DAC8635.
- [10] Yang, R. J., Gu, L., Liaw, L., Gearhart, C., Tho, C. H., Liu, X. and Wang, B. P., 2000, “Approximations for Safety Optimization of Large Systems,” *Proc. ASME 2000 Design Engineering and Technical Conferences*, September 10-13, Baltimore, Maryland, DETC2000-DAC14245.
- [11] Yang, R. J., Wang, N., Tho, C. H., Bobineau, J. P. and Wang, B. P., 2001, “Metamodeling Development for Vehicle Frontal Impact Simulation,” *Proc. ASME 2001 Design Engineering and Technical Conferences*, September 9-12, Pittsburgh, PA, DETC2001-DAC 21012.
- [12] Redhe, M. and Nilsson, L., 2002, “Using Space Mapping and Surrogate Models to Optimize Vehicle Crashworthiness Design,” *Proc. 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, September 4-6, Atlanta, Georgia, AIAA-2002-5536.
- [13] Han, J. and Yamada, K., 2000, “Maximization of the Crushing Energy Absorption of the S-Shaped Thin-Walled Square Tube,” *Proc. 8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, September 6-8, Long Beach, CA, AIAA-2000-4750.
- [14] Kurtaran, H., Omar, T. and Eskandarian, A., 2001, “Crashworthiness Design Optimization of Energy-Absorbing Rails for the Automotive Industry,” *Proc. ASME 2001 International Mechanical Engineering Congress and Exposition*, November 11-16, New York, NY, IMECE2001-AMD25452.
- [15] Chen, S., 2001, “An Approach for Impact Structure Optimization using the Robust Genetic Algorithm,” *Finite Elements in Analysis and Design*, **37**: 431-446.
- [16] Mase, T., Wang, J. T., Mayer, R., Bonello, K. and Pachon, L., 1999, “A Virtual Bumper Test Laboratory for FMVR 581,” *Proc. ASME 1999 Design Engineering and Technical Conferences*, September 12-15, Las Vegas, Nevada, DETC99-DAC 8572.
- [17] Yang, R. J., Gu, L., Tho, C. H. and Sobieski, J., 2001, “Multidisciplinary Optimization of a Full Vehicle with High Performance Computing,” *Proc. 2001 American Institute of Aeronautics and Astronautics Conference*, pp. 688-698, AIAA-2001-1273.
- [18] Soto, C. A. and Diaz, A. R., 1999, “Basic Models for Topology Design Optimization in Crashworthiness Problems,” *Proc. ASME 1999 Design Engineering and Technical Conferences*, September 12-15, Las Vegas, Nevada, DETC99-DAC8591.
- [19] Song, J. O., 1986, “An Optimization Method for Crashworthiness Design,” *SAE Transactions*, Paper number: 860804, pp. 39-46.
- [20] Bennett, J. A., Lust, R. V. and Wang, J.T., 1991, “Optimal Design Strategies in Crashworthiness and Occupant Protection,” *Proc. ASME Winter Annual Meeting*, Atlanta, GA, **126**: 51-66.
- [21] Chellappa, S. and Diaz, A., 2002, “A Multi-Resolution Reduction Scheme for Structural Design,” *Proc. NSF 2002 Conference*, January 2002, pp. 98-107.
- [22] Ignatovich, C. L. and Diaz, A., 2002, “Physical Surrogates in Design Optimization for Enhanced Crashworthiness,” *Proc. 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, September 4-6, Atlanta, Georgia, AIAA Paper Number: AIAA-2002-5537.
- [23] Abramowicz, W., 2003, “Thin-Walled Structures as Impact Energy Absorbers,” *Thin Walled Structures*, **41**:91-107.
- [24] Takada, K. and Abramowicz, W., 2004, “Fast Crash Analysis of 3D Beam Structures Based on Object Oriented Formulation,” *Proc. 2004 SAE World Congress*, Detroit, Michigan, Paper number: 04B-119.
- [25] Abramowicz, W., 2004, “An Alternative Formulation of the FE Method for Arbitrary Discrete/Continuous Models,” *Int. J. of Impact Engineering*, **30**: 1081-1098.
- [26] Michalewicz, Z. and Fogel, D. B., 2000, *How to Solve it: Modern Heuristics*, Springer-Verlag Berlin Heidelberg, New York.
- [27] Goldberg, D., 1989, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley Inc., New York.
- [28] Phadke, M., 1989, *Quality Engineering using Robust Design*, Prentice Hall PTR, Englewood Cliffs, New Jersey.

APPENDIX A: CRASH MODE METRIC USED IN CASE STUDY

symbol	Description
δ_1	Side-squish in zone 1
δ_2	Side-squish in zone 10
δ_3	Axial crush in zone 2
δ_4	Bending in zone 2
δ_5	Axial crush in zone 3
δ_6	Bending in zone 3
δ_7	Axial crush in zone 5

δ_8	Bending in zone 5
δ_9	Axial crush in zone 9
δ_{10}	Bending in zone 9
δ_{11}	Axial crush in zone 11
δ_{12}	Bending in zone 11
δ_{13}	Combined axial crush and bending in zone 4
δ_{14}	Combined axial crush and bending in zone 6
δ_{15}	Combined axial crush and bending in zone 7
δ_{16}	Combined axial crush and bending in zone 8
δ_{17}	Combined axial crush and bending in zone 12
δ_{18}	Combined axial crush and bending in zone 13
δ_{19}	Combined axial crush and bending in zone 14

APPENDIX B: FUZZY RULES USED IN CASE STUDY

The following tables summarize the 238 fuzzy rules used in the case study in a tabulated form. Values in a table indicate adjustments to the respective design variable. For example, Table B.1 represents the following five fuzzy rules for adjusting t_1 :

- If $\delta_1 \in NH$, then adjust t_1 by -0.4.**
- If $\delta_1 \in NL$, then adjust t_1 by -0.2.**
- If $\delta_1 \in Z$, then adjust t_1 by +0.1.**
- If $\delta_1 \in PL$, then adjust t_1 by +0.2.**
- If $\delta_1 \in PH$, then adjust t_1 by +0.4.**

Table B.1 Fuzzy rules for adjusting t_1

Membership of δ_1				
NH	NL	Z	PL	PH
-0.4	-0.2	+0.1	+0.2	+0.4

Table B.2 Fuzzy rules for adjusting t_2

Membership of δ_3					
NH	NL	Z	PL	PH	
-0.2	-0.1	+0.05	+0.1	+0.2	
Membership of δ_4					
NH	NL	Z	PL	PH	
			+0.1	+0.2	
Membership of δ_5					
NH	NL	Z	PL	PH	
Memb. of δ_4	NH				
	NL				
	Z				
	PL	+0.15	+0.10		
	PH	+0.20	+0.15		

Table B.3 Fuzzy rules for adjusting t_3

Membership of δ_5				
NH	NL	Z	PL	PH
-0.2	-0.1	+0.05	+0.1	+0.2
Membership of δ_6				

Membership of δ_5					
NH	NL	Z	PL	PH	
			+0.1	+0.2	
Memb. of δ_6	NH				
	NL				
	Z				
	PL	+0.15	+0.10		
	PH	+0.20	+0.15		

Table B.4 Fuzzy rules for adjusting t_4

Membership of δ_3				
NH	NL	Z	PL	PH
-0.4	-0.2		+0.2	+0.4

Table B.5 Fuzzy rules for adjusting t_5

Membership of δ_7				
NH	NL	Z	PL	PH
-0.2	-0.1		+0.1	+0.2
Membership of δ_8				
NH	NL	Z	PL	PH
-0.2	-0.1		+0.1	+0.2

Table B.6 Fuzzy rules for adjusting t_6

Membership of δ_{14}				
NH	NL	Z	PL	PH
-0.4	-0.2		+0.2	+0.4

Table B.7 Fuzzy rules for adjusting t_7

Membership of δ_{15}				
NH	NL	Z	PL	PH
-0.4	-0.2		+0.2	+0.4

Table B.8 Fuzzy rules for adjusting t_8

Membership of δ_{16}				
NH	NL	Z	PL	PH
-0.4	-0.2		+0.2	+0.4

Table B.9 Fuzzy rules for adjusting t_9

Membership of δ_9				
NH	NL	Z	PL	PH
-0.2	-0.1		+0.1	+0.2
Membership of δ_{10}				
NH	NL	Z	PL	PH
-0.2	-0.1		+0.1	+0.2

Table B.10 Fuzzy rules for adjusting t_{10}

Membership of δ_2				
NH	NL	Z	PL	PH

-0.4	-0.2	+0.1	+0.2	+0.4
------	------	------	------	------

Table B.11 Fuzzy rules for adjusting t_{11}

Membership of δ_{11}				
NH	NL	Z	PL	PH
-0.2	-0.1		+0.1	+0.2
Membership of δ_{12}				
NH	NL	Z	PL	PH
-0.2	-0.1		+0.1	+0.2

Table B.12 Fuzzy rules for adjusting t_{12}

Membership of δ_{17}				
NH	NL	Z	PL	PH
-0.4	-0.2	-0.1	+0.2	+0.4

Table B.13 Fuzzy rules for adjusting t_{13}

Membership of δ_{18}				
NH	NL	Z	PL	PH
-0.4	-0.2	-0.1	+0.2	+0.4

Table B.14 Fuzzy rules for adjusting t_{14}

Membership of δ_{19}				
NH	NL	Z	PL	PH
-0.4	-0.2	-0.1	+0.2	+0.4

Table B.15 Fuzzy rules for adjusting t_{15}

		Membership of δ_3				
		NH	NL	Z	PL	PH
Memb. of δ_4	NH					
	NL					
	Z	+6.0	+3.0			
	PL	+9.0	+6.0	+3.0		
	PH	+12.0	+9.0	+6.0		
		Membership of δ_5				
		NH	NL	Z	PL	PH
Memb. of δ_6	NH					
	NL					
	Z	+6.0	+3.0			
	PL	+9.0	+6.0	+3.0		
	PH	+12.0	+9.0	+6.0		
		Membership of δ_7				
		NH	NL	Z	PL	PH
Memb. of δ_8	NH		-3.0	-6.0	-9.0	-12.0
	NL	+3.0		-3.0	-6.0	-9.0
	Z	+6.0	+3.0		-3.0	-6.0
	PL	+9.0	+6.0	+3.0		-3.0
	PH	+12.0	+9.0	+6.0	+3.0	

Table B.16 Fuzzy rules for adjusting b_1

		Membership of δ_3				
		NH	NL	Z	PL	PH
N	e	NH				

	NL				
	Z	-4.0	-2.0		
	PL	-6.0	-4.0	-2.0	
	PH	-8.0	-6.0	-4.0	

		Membership of δ_5				
		NH	NL	Z	PL	PH
Memb. of δ_6	NH					
	NL					
	Z	-4.0	-2.0			
	PL	-6.0	-4.0	-2.0		
	PH	-8.0	-6.0	-4.0		

		Membership of δ_7				
		NH	NL	Z	PL	PH
Memb. of δ_8	NH		+2.0	+4.0	+6.0	+8.0
	NL	-2.0		+2.0	+4.0	+6.0
	Z	-4.0	-2.0		+2.0	+4.0
	PL	-6.0	-4.0	-2.0		+2.0
	PH	-8.0	-6.0	-4.0	-2.0	

Table B.17 Fuzzy rules for adjusting h_2

		Membership of δ_9				
		NH	NL	Z	PL	PH
Memb. of δ_{10}	NH		-3.0	-6.0	-9.0	-12.0
	NL	+3.0		-3.0	-6.0	-9.0
	Z	+6.0	+3.0		-3.0	-6.0
	PL	+9.0	+6.0	+3.0		-3.0
	PH	+12.0	+9.0	+6.0	+3.0	

		Membership of δ_{11}				
		NH	NL	Z	PL	PH
Memb. of δ_{12}	NH		-3.0	-6.0	-9.0	-12.0
	NL	+3.0		-3.0	-6.0	-9.0
	Z	+6.0	+3.0		-3.0	-6.0
	PL	+9.0	+6.0	+3.0		-3.0
	PH	+12.0	+9.0	+6.0	+3.0	

Table B.18 Fuzzy rules for adjusting b_2

		Membership of δ_9				
		NH	NL	Z	PL	PH
Memb. of δ_{10}	NH		+2.0	+4.0	+6.0	+8.0
	NL	-2.0		+2.0	+4.0	+6.0
	Z	-4.0	-2.0		+2.0	+4.0
	PL	-6.0	-4.0	-2.0		+2.0
	PH	-8.0	-6.0	-4.0	-2.0	

		Membership of δ_{11}				
		NH	NL	Z	PL	PH
Memb. of δ_{12}	NH		+2.0	+4.0	+6.0	+8.0
	NL	-2.0		+2.0	+4.0	+6.0
	Z	-4.0	-2.0		+2.0	+4.0
	PL	-6.0	-4.0	-2.0		+2.0
	PH	-8.0	-6.0	-4.0	-2.0	