Combined Analysis of LS-DYNA Crash-Simulations and Crash-Test Scans

Dominik Borsotto[1], <u>Lennart Jansen</u>[1], Clemens-August Thole[1]

[1]SIDACT GmbH, Grantham Allee 2-8, 53757 Sankt Augustin, Germany

1 Abstract

In robustness campaigns and optimization processes metamodels are created out of a set of crash-simulations. With the help of such analyses the models used for the simulations can be improved. For example, instabilities can be found and explained or the needed material can be minimized under certain safety restrictions.

An important question in this context is: How good can these metamodels represent the reality? To answer this question, one can compare the crash-simulations to the real crash-tests, which were recorded by camera systems after the crash. To be able to compare the test-data with the LS-DYNA crash-simulations, we first need to convert the test-data by matching the geometries and transferring the part information from the simulation to the crash-test. Afterwards one can calculate the combination of the simulations, which approximates geometry and deformation behavior of the test-data as close as possible. The distance and difference in behavior between this calculated Best Fit and the actual crash-test can be used to measure the quality of the simulation model. Once the evaluation of the model is finished, the test-data can also be added to a robustness campaign as an additional simulation and used for further analysis.
This allows us to answer questions such as: How does the test fit into the simulation subspace? Which simulation runs are similar to the test for a certain crash event? Which of the dominating crash events found in the simulation can also be found in the test?

Thus, the described matching procedure combined with exemplary further analysis methods on the one hand allow for a quick and automated matching between test and simulation and on the other hand a more detailed validation of the simulation model in comparison to the actual test. Due to the conversion of the test-data, the same post-processors can be used for both the simulations and the test-data, resulting in a smoother workflow.

2 Background and Objectives

Since the past few years the overall awareness of variability and scatter for CAE predictions is steadily increasing. Giving the fact that variability is inherent in nature it is also a major task to master it during product- and in this case especially vehicle-development. As a matter of fact in car industry for many load cases there is only provision for a single performance confirmation test to verify the CAE model. As such a test is influenced by a series of potential variability sources like e.g. production tolerances and crash test parameter settings, the chance to run into unpredictable crash results rises. In case of unforeseen results this usually leads to expensive and inefficient design changes, at a late vehicle development phase. To counteract the above mentioned the CAE model should already have a robust design which is not sensitive to small variations and still delivers predictable results. Thus before applying design optimizations, the overall robustness of the model needs to be ensured. Taking a deeper look into the complex event of a car crash many reasons can be discovered why small variations actually lead to a big spread among the results. Just to mention a view, consider parts kinking in one direction or the other or parts passing each other instead of hooking up. As a consequence one approach to generate a robust design is to find these events (often referenced to as bifurcations) and derive design suggestions that can handle the variations and still deliver a deterministic crash behavior. One way to achieve this is mainly based on Principle Component Analysis methods and standard statistics. Our main objective is to add the information contained in crash test scans to this robustness analysis. Previous results can be found in [1].

3 Simulations and Test data

In the following we will present an example for a set of simulations and one for the test data. The simulation set consists of 30 runs. As a model we use the Chevrolet Silverado, which has been developed by The National Crash Analysis Center (NCAC) of The George Washington University under a contract with the FHWA and NHTSA of the US DOT. Each simulation has 152 time steps and 679 Parts with a total of 929,131 Finite-Elements.
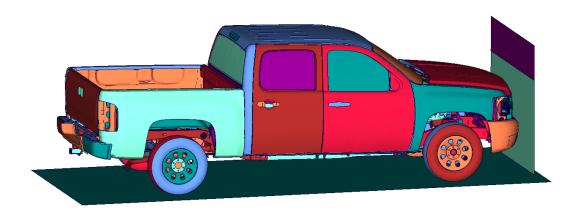

Fig.1:   Snapshot of a simulation run of the Chevrolet Silverado before the crash.

In contrast to the simulation runs the scan of the actual crash test, which we will also call test data, has no part information and only one time step. The time step may be after the last time step of the simulation runs since the car is scanned after the crash event. Even so the scan captures only a section of left part of the driver's cabin it consists of 415456 Finite-Elements. Hence, the grid of the scan may be much finer than the grid of the simulation runs.
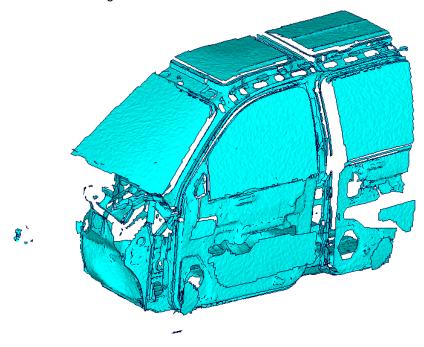

Fig.2:   Crash test scan in STL format without any part information. View from the outside.

4 Conversion

To realise the inclusion of the test data into a robustness analysis we first have to convert the test data.

4.1 Position and Part information

The first step of the conversion is the matching of the nodes of the test data and the simulation runs. Therefore we translate and rotate the geometry of the test data such that it matches the geometry of the simulations as much as possible. As soon as the nodes of the test data can be identified with nodes from the simulation runs, the part information of the simulation runs can be transferred to the test data.
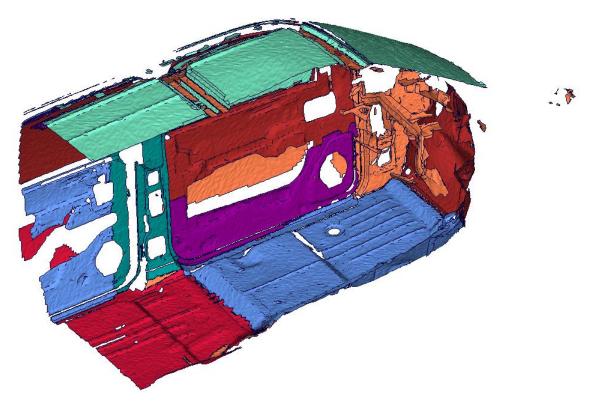

Fig.3: Crash test scan with part information. View from the inside.

4.2 Best Fit

Based on the test data with the already transferred part information, we calculate an artificial simulation run which matches the test data on a selection of parts as good as possible. Therefore let n be the number of simulations and let X_i contain the coordinates of the i-th simulation. Then the coefficients β_i arise of the optimal matching process. We define

$$X_{Test} := \sum_{i=1}^{n} \beta_i X_i, \quad with \quad \sum_{i=1}^{n} \beta_i = 1$$

and call X_{Test} the Best Fit of the simulation set regarding the test data. By calculating the distance between the nodes of the test data and their counterpart in the Best Fit, we can measure how well the test data can be described by the simulation set. In case of overall low distances we would say that the simulation set is able to approximate the scan of the crash test. Then one can assume that the simulation model is able to represent the effects which occur in a real crash.
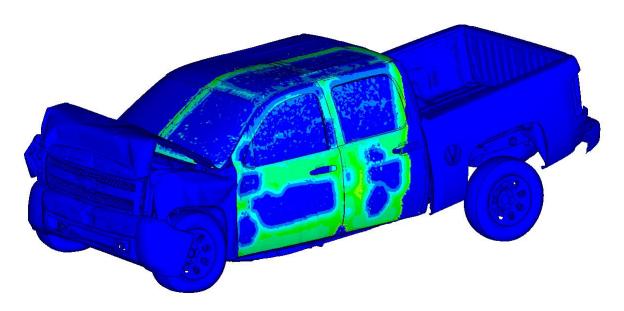
Fig.4: Visualisation of the distance between the test data and the Best Fit with low distances.

In case of overall high distances it is the other way around. The set of simulation seems not to be able to describe the effect occurring in the test data.

5 Robustness Analysis

Before we integrate the test data in the robustness analysis we shortly revise the basics of the Principle Component Analysis. For more information see [2].

5.1 Principle Component Analysis

Let X_i contain the coordinates of the i-th simulation run and let n be the number of simulations again. Then we define the average of all simulations as

$$\overline{X} := \frac{1}{n} \sum_{i=1}^{n} X_i.$$

Futhermore we define the difference of each simulation and the average of all simulations and gather these values in a matrix

$$\hat{X}_i := X_i - \overline{X}, \quad \hat{X} := \left(\cdots \; \hat{X}_i \; \cdots \right).$$

Let C be the covariance matrix defined by

$$C_{ij} := \left\langle \hat{X}_i, \hat{X}_j \right\rangle.$$

The covariance matrix is positive definite and therefore there is a orthogonal matrix V and a diagonal matrix Λ such that

$$C = V^T \Lambda^2 V = A^T A, \quad with \quad A := \Lambda V.$$

We define the modes of the set of simulations as

$$M := A^T \hat{X}.$$

Now each simulation can be expressed in terms of the modes:

$$M = A^T \hat{X} \quad \Rightarrow \quad \hat{X}_i = \alpha^i M \quad \forall i \quad \Rightarrow \quad X_i = \overline{X} + \sum_{j=1}^{n} \alpha^i_j M_j \quad \forall i$$

With α^i being the i-th row of A.

5.2 Mode composition of the Test data

With the help of the expression of the simulation runs in terms of modes we are also able to formulate the Best Fit run in terms of modes:

$$X_{Test} = \sum_{i=1}^{n} \beta_i X_i = \sum_{i=1}^{n} \beta_i \left(\overline{X} + \sum_{j=1}^{n} \alpha_j^i M_j \right) = \overline{X} + \sum_{i=1}^{n} \sum_{j=1}^{n} \beta_i \alpha_j^i M_j = \overline{X} + \sum_{j=1}^{n} \left(\sum_{i=1}^{n} \beta_i \alpha_j^i \right) M_j \, .$$

We denote the contribution of the j-th mode to the Best Fit by γ_j:

$$\gamma_j := \sum_{i=1}^{n} \beta_i \alpha_j^i$$

Hence, we obtain:

$$X_{Test} = \overline{X} + \sum_{j=1}^{n} \gamma_j M_j \, .$$

With the help of this representation of the test data we are able to compare the behavior of the test data with the behavior of the simulations in terms of modes.

5.2.1 Visualisation in the Scatterplot

We can visualise the modes in a scatterplot. In the following figure the two dominating modes are plotted with the modes of the test data being highlighted in red. The modes of the test data are part of a cluster of the modes of the simulations. Hence one can deduce that the behaviour is similar to that of the simulations in this particular cluster.
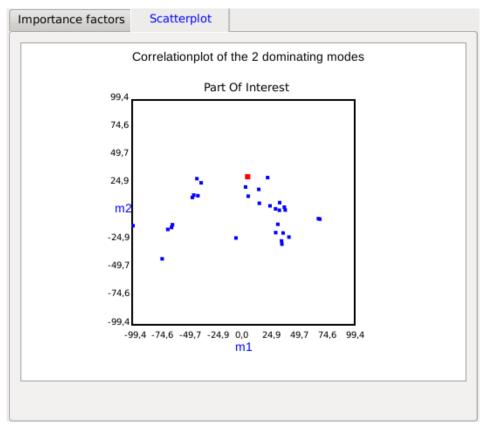

Fig.5: Representation of the test data as modes in the scatterplot. The qualitative and quantitative behaviour of the simulations and the test data match.

If that would not be the case, it is probable that the test data does not behave like a simulation from the set.

5.2.2 Relation to the Importance Factors

To decrease scatter in the simulations, one is interested to diminish the mode with the most impact on the simulations. But it may be that the mode that dominates the behavior of the simulations is not necessarily the most important one for the test data as we see in Figure 6. Hence, reducing the modes

with the highest impact on the simulations may not be enough to reduce the scatter in the actual crash test. And therefore the modes dominating the behavior of the test data should also be taken in account.
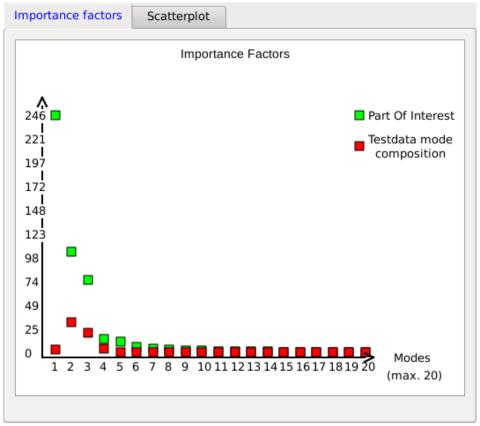

Fig.6:   The mode with the highest importance factor does almost contribute nothing to the test data.

6 Summary

The driving question of this work is: How good can a set of simulation represent the behavior of an actual crash test? To answer this question, we added the test data to a robustness analysis of the crash-simulations. After converting the test data we were able to compare the behavior of the crash test with the simulations in terms of modes with the help of the Principle Component Analysis.

7 Literature

[1]   Automotive Crash Test & Simulation Comparisons of a Pickup Truck & a Small Car Oblique Impacts Into a Concrete Barrier, D. Marzougui, C.D. Kan, and K.S. Opiela: 13th International LS-DYNA Users Conference, 2014
[2]   Advanced Mode Analysis for Crash Simulation Results, C.A. Thole, L. Nikitina, I. Nikitin, T.Clees: 9. LS-DYNA Forum, 2010