Proceedings of the Design Society, Volume 5: ICED25
https://doi.org/10.1017/pds.2025.10168

Enhancing numerical simulation analysis with the use of explainable artificial intelligence and large language models: a case study on vehicle side crash optimization

Janis Mathieu[1,2,✉], Stefan Kronwitter[3], Johannes Pfahler[1], Michael Di Roberto[1] and Michael Vielhaber[2]

[1] Porsche Engineering Group GmbH, Germany, [2] Saarland University, Germany,
[3] Dr. Ing. h.c. F. Porsche AG, Germany

✉ janis.mathieu@porsche-engineering.de

ABSTRACT: Substantial engineering efforts are dedicated to reducing injury risks in crash scenarios during the development of new vehicles. This is achieved by performing crash simulations to optimize the nonlinear behavior of systems. However, the complexity makes their behavior difficult and time-consuming for engineers to understand. To reduce the analysis time, this study introduces a modular framework combining Explainable Artificial Intelligence and Large Language Models (LLM). Shapley Additive Explanation values allow for simulation-wise feature importance attribution and generate a data-driven understanding. An LLM assists by making result data interactively accessible and supports technical report generation. Validated through a real-world vehicle side crash optimization use case, the framework demonstrates enhanced and accessible insights into system behavior within virtual engineering.

KEYWORDS: artificial intelligence, virtual engineering (VE), computational design methods, optimisation

---

1. Introduction

Road traffic injuries cause more than one million deaths worldwide every year. As a result, substantial engineering efforts are dedicated to reducing injury risks for both occupants and pedestrians in crash scenarios during the development of new vehicles (Gonter et al., 2021). Protection mechanisms include mechanical energy absorption structures, like side sills and longitudinal members, as well as restraint systems such as seats, seatbelts, and airbags. These systems undergo dynamic loads resulting in large deformations within a short period of time. The behavior is nonlinear due to large deformations, material fracture, as well as contact interactions (Beyer et al., 2021). By using virtual investigations based on numerical Finite Element (FE) crash simulations, flexibility during development is increased compared to purely physical crash-testing-based development processes. Nowadays, physical crash tests are used primarily for final system approval and validation of virtual models during development.

However, the design process remains complex and poses significant challenges to the engineers involved, especially due to growing demands from legal regulations and customer expectations, which are leading to increased system complexity. The design can thus be described as a substantial multi-objective optimization problem. To systematically handle optimization procedures, approaches include the use of less complex data-driven surrogate models, which are trained using data from high-fidelity FE simulation models (Horii, 2017; Büttner et al., 2023), Reinforcement Learning (RL)-agents (Mathieu et al., 2024; Trilling et al.; 2024) or specifically defined heuristics (Beyer et al., 2021). The results generated with these methods are often complex and a considerable proportion of development time is spent on analyzing and understanding these systems. Approaches that use global sensitivity metrics, such as Sobol Indices (Büttner et al., 2023), or

comprehensible heuristics for layout definition (Beyer et al., 2021) can provide a superficial understanding of why a certain system configuration appears optimal. Frameworks for the Artificial Intelligence (AI)-based analysis of numerical simulation results in general show great potential in increasing development process quality and efficiency as well as improving system understanding. Proposed methods include rule mining for bumper deformations (Diez et al., 2018), anomaly detection in crash-loaded structures (Kracker et al., 2023) and in crash sensor signals (Mathieu et al., 2023), as well as deep-learning plausibility checks (Bickel et al., 2023) and modelling cause-and-effect relationships (Conti and Kaijima, 2020).
However, all these approaches lack detailed datapoint-specific insights, the possibility for intuitive user-interaction and a direct link to generate technical documentation. Subsequently, this work centers on the aspect of gaining global and local insights with the use of eXplainable AI (xAI) as well as the interactive result representation and documentation with Large-Language-Models (LLM). This will be discussed in the context of mechanical system optimization and validated within an industry use case.

2. State of the art – optimizing and understanding system behavior

The related work for the proposed framework involves optimization of crash-loaded mechanical systems (Section 2.1.), AI-driven analysis of FE simulation results (Section 2.2.), and utilization of xAI methodologies and LLMs to generally explain and understand complex behavior (Section 2.3.).

2.1. Optimization of crash-loaded systems

During the development process of new vehicle generations, simultaneous optimization of the vehicle structure and the restraint system is performed (Gonter et al., 2021). This ensures structural integrity and the lowest possible loads on occupants represented by Anthropomorphic Test Devices (ATD) in crash tests, commonly known as crash test dummies. Occupant safety as discussed in Horii (2017), or Mathieu et al. (2024) typically focuses on the design of airbags and seatbelts with the aim of reducing loads on the ATD. Structural safety as discussed in Beyer et al. (2021), Büttner et al. (2023) or Borse et al. (2024) focuses on optimizing energy absorption, maximum force or intrusion depths. Approaches are similar for both fields and include the use of global optimization algorithms either with surrogate models or computationally efficient simulation models. Especially for the topology optimization of the 3D layout of structural components graph-heuristic approaches have shown great potential (Beyer et al. (2021). The latter approach within two dimensions can be extended by a RL-based heuristic (Trilling et al., 2024). Mathieu et al. (2024) also include the use of RL techniques to minimize occupant loads by design adjustments in the restraint system, and Borse et al. (2024) increase the structural performance of a crash box structure by adjusting the wall thicknesses.

2.2. AI-based analysis of numerical simulations

Aiming for increased quality and faster understanding of mechanical systems, a variety of approaches have been proposed in the literature to support engineers in analyzing simulation results. These focus either on the analysis of time-variant crash-loaded systems (Diez et al., 2018; Iza Teran et al., 2019; Kracker et al., 2023) or statically loaded systems (Conti and Kaijima, 2020; Bickel et al., 2023). The purposes of these investigations are diverse, including the detection of outliers (Kracker et al., 2023), the identification of cause-and-effect relationships (Conti and Kaijima, 2020), as well as finding rules of how outcomes compile (Diez et al., 2018). Kracker et al. (2023) propose an approach to automatically specify outliers in structural behavior of vehicle components subjected to crash load. Outlier scores are calculated using the kth-nearest-neighbor method within a dimensionally reduced representation of each component. Thole et al. (2010) analyze multiple crash simulations, identifying the components that contribute most significantly to the dominant differences in the results. Global understanding is gained by applying Principal Component Analysis (PCA) to detect deformation modes, similar to Kracker et al. (2023). Additionally, to identify potential origins of scatter for a given state of interest, correlation clustering is applied, and each component at each time step is analyzed with the so-called difference PCA. Both approaches have been validated on large-scale crash simulation models. An alternative for representing numerical crash simulations in a low-dimensional space is proposed by Iza Teran and Garcke (2019), who use the Laplace-Beltrami and Fokker-Planck operator on FE mesh data. Diez et al. (2018) utilize instead a supervised learning technique, specifically decision trees, to extract rules in an IF-THEN format for deformation modes of a bumper in a frontal crash scenario. Hence, information from the simulation input and output is used for processing. In the work of Hahner et al. (2020), the temporal behavior of crash-loaded vehicle structures represented as oriented bounding boxes is learned using Long

Short-Term Memory Networks. Two decoders are incorporated: one for reconstruction and one for predicting future time steps of the analyzed sequence. In addition to analysis of the latent space, predictions can also be performed. In the above-mentioned approaches, no direct classification is made regarding the plausibility of obtained simulation results. This aspect, however, is covered in the work of Bickel et al. (2023), who use a multilabel classification technique incorporating Convolutional Neural Networks. The multilabel classifier, trained on 60,000 simulations including different models, allows for defining the plausibility of the mesh, geometrical shape, or load values. To increase the interpretability of static numerical FE simulations and define what may have caused a certain simulation result, Conti and Kaijima (2020) trained a Bayesian Network on design parameters and corresponding deflections of a beam under static load. The resulting probabilities can thus help to estimate an outcome and improve understanding of relationships within the design space.

2.3. Explanation and understanding of systems with artificial intelligence

For gaining insights, why machine learning (ML) models make certain predictions given specific input data, xAI techniques, such as counterfactuals or model-agnostic techniques, including SHapley Additive exPlanations (SHAP) and Local Interpretable Model-Agnostic Explanations (LIME), can be used (Molnar et al., 2022). These techniques provide valuable insights into the rationale of complex AI systems and improve trustworthiness. However, xAI techniques serve to gain transparency of the trained ML model, whereby explainability is only achieved by including domain knowledge (Roscher et al., 2020). In the field of production engineering, Feldkamp and Strassburger (2023) apply different model-agnostic techniques (SHAP, LIME, Anchor) to investigate the robustness in control factors of a plant simulation. Multiple relationships are modeled, including those between control factors and a robustness measure, revealing what factors contribute to potential robustness issues. Other promising results include the work of Lin et al. (2023), who propose a robust optimization framework that integrates a ML model to forecast tunnel-induced damage. The model is trained on static structural FE simulation data and is used to solve multi-objective optimization problems. Global importances based on SHAP values provide transparency, enhancing understanding and trust in the recommendations. It is important to highlight that what constitutes an interpretable model can vary based on the situation and target user group, making it challenging to determine the level of interpretability required (Molnar et al., 2022).

Apart from the xAI field, large language models (LLM) have demonstrated substantial capabilities in reasoning over data and explaining certain behaviors in a human-relatable manner. Extensive pre-training equips these models with a large knowledge base, enabling the contextualization of data and generation of insightful interpretations. By presenting recommendations in natural language, accessibility and interpretability of information, especially for non-technical users, is increased. Hegselmann et al. (2024) analyze few-shot classification of tabular data with LLMs, highlighting their ability by benchmarking against classical approaches, including gradient-boosted trees. Their so-called TabLLM outperforms baseline models in the very-few-shot setting. However, when many training samples are available, classical ML models remain competitive, emphasizing the effective utilization of prior knowledge in the LLM. The actual use of the feature names and their relationships by the LLM is demonstrated by the observed performance drops when column names are removed. Bordt et al. (2024) confirm these findings, revealing that LLMs have memorized many popular tabular datasets verbatim. This aspect is particularly promising as data understanding forms the basis of reasoning and explanation. Roy et al. (2024) explore the use of LLMs to reduce the efforts of on-call engineers by automatically investigating the causes of incidents. The proposed ReAct agent is equipped with retrieval tools and evaluated on an out-of-distribution dataset of production incidents collected at Microsoft. Results show good retrieval and reasoning performances, with increased factual accuracy. Notably, the LLM agent is not fine-tuned as outlined in Ahmed et al. (2023) due to cost and time constraints. Hsu et al. (2024) combine xAI and LLM, where global importances based on SHAP values are explained in textual format to be understood by non-experts. This approach successfully transforms complex SHAP plots into short and easy-to-understand text outputs. In the field of mechanical engineering, Jadhav et al. (2024) propose the use of an LLM for mechanical design. The system specification is provided in natural language, and the LLM designer tests new designs by executing FE simulations to evaluate the performance. A success rate of up to 90 % for optimizing simple truss structures highlights future potential.

2.4. Derivation of research objectives

Based on the previously discussed literature, relevant gaps addressed in the present work are highlighted. Global optimization approaches focus on directly providing the best possible design but often lack insights of why this design is optimal. Currently, this can be achieved by manual comparison of the optimal design and initial design, which is complex, especially if many parameters, such as the 38 wall thicknesses in Büttner et al. (2023), are considered simultaneously. Global design space analysis as realized based on SHAP values in Lin et al. (2023) or Sobol indices, as in Büttner et al. (2023), improve understanding but lack datapoint-specific information. Unsupervised learning approaches are utilized by analysis methods for crash simulations to detect outliers (Thole et al., 2010; Iza Teran and Garcke, 2019; Kracker et al., 2023). The question of what may cause the outlier is partially covered by Diez et al. (2018) and Thole et al. (2010). However, the latter does not allow intuitive traceback to design parameters, which are relevant for understanding system behavior. The decision trees used in Diez et al. (2018) do not allow for an attribution to certain features, and the approach is marginally feasible for simultaneous analysis of many parameters. Additionally, none of the approaches named offers the possibility of providing results in textual format for interaction and technical report generation. While Hsu et al. (2024) propose the idea of explaining SHAP values with a LLM, the encoding into the model is only in tabular format. Furthermore, the local representation for analyzing individual data points is not discussed, despite SHAP enabling this. Additionally, there is no integration of the trained predictive models that can provide on-demand predictions and contribute to system understanding. Subsequently, the proposed framework will cover three main objectives:

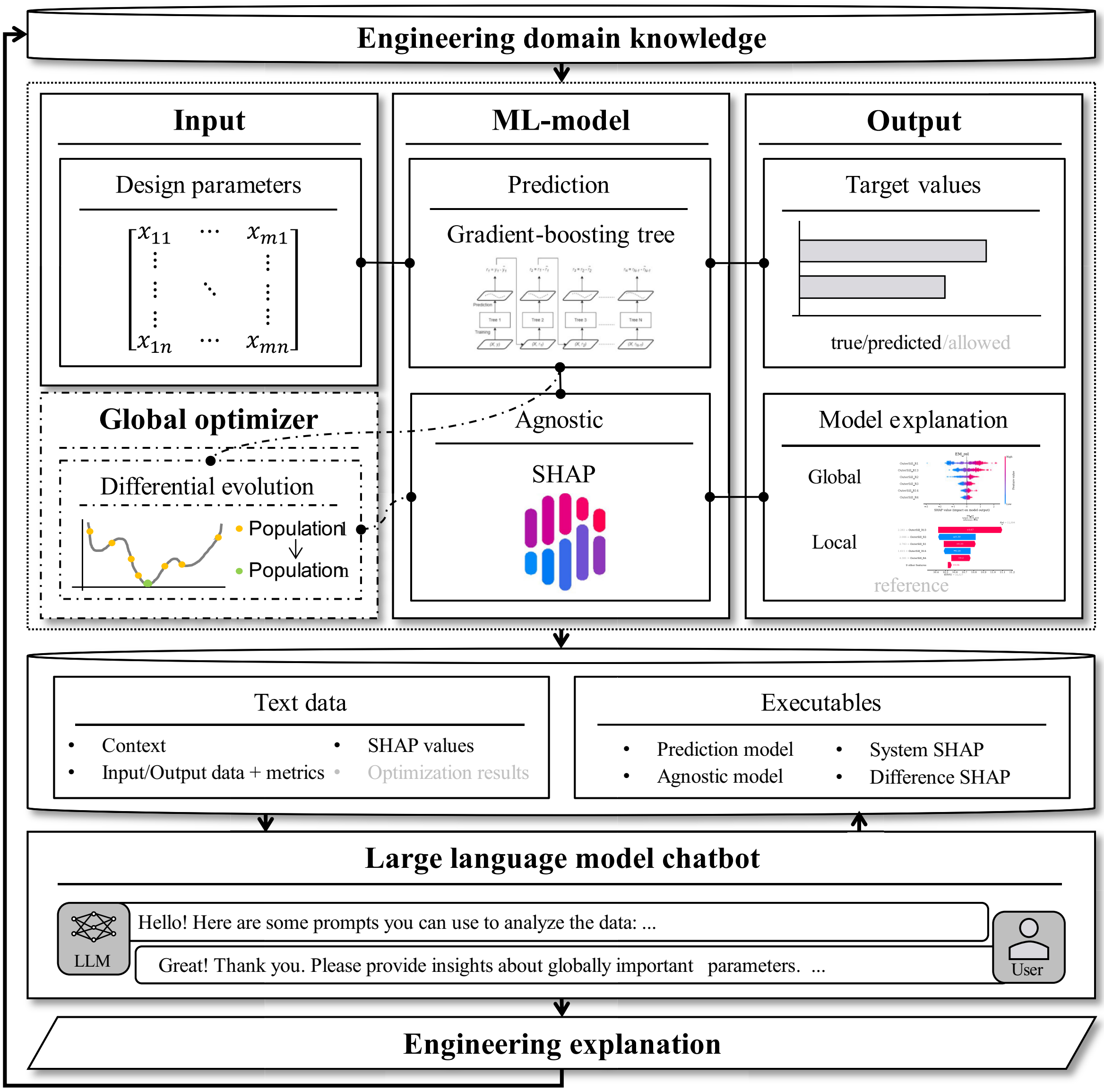
1) Generating global importance and local design parameter contributions of crash-loaded systems through SHAP analysis of a data-driven surrogate model for system behavior understanding.
2) Providing the obtained data and trained model executables of the SHAP analysis to an LLM enabling interactive exploration for engineers or automated technical report generation.
3) Validating the usefulness of the framework using the example of an optimization for a crash-loaded side sill.

These objectives aim to increase the interpretability of crash simulation results. Therefore, the engineers involved understand the system better and in less time. As this builds the foundation for developing innovative products and improving efficiency in the long term by automated documentation processes, the present work has high relevance to the field of AI-enhanced engineering.

3. Explanation of FE crash simulation results through xAI and LLMs

3.1. Framework

The foundation of the framework, depicted in Figure 1, is given by engineering domain knowledge. This includes the situational deployment of the framework in general, the definition of input and output data, as well as simulation data generation. The model input consists of design parameters, such as wall thicknesses or restraint force levels. The ML model is then trained to represent relations to so-called target values. A target value can be a single value such as a certain load on an ATD but can also be an objective function weighing multiple values such as maximum force and energy absorption. Due to the high performance on tabular data of medium sized datasets and the efficient computation of SHAP values with TreeSHAP (Lundberg et al., 2020) a gradient-boosting tree ML model according to Chen and Guestrin (2016) is used. The ML model is trained to serve as a surrogate model for the FE simulation model and can optionally be passed to a differential evolution (DE) optimizer for optimization of the target value, which is known as metamodel-based optimization (Büttner et al., 2023). As a next step, the trained prediction model is analyzed using TreeSHAP to obtain the SHAP values as contributions of each design parameter to the target value for all simulation results. A background dataset, i.e., the training data, is used for SHAP value calculation to allow dependencies between features to be handled according to the rules dictated by causal inference (Lundberg et al., 2020; Janzing et al., 2020). The input and output data, SHAP values and the executables for the prediction and agnostic model are stored in a database. Input and output data as well as the SHAP values are serialized for each simulation and transformed into a natural-language string (Hegselmann et al., 2024), as the LLM understands natural language text input best. This is also done for the results obtained by the DE optimizer and statistical measures for the input and output data, e.g., mean and standard deviation. If too many text units (tokens) are required to fit the whole input at once, information is provided simulation-wise depending on user demand. The LLM functionality within the framework is realized with an Application Programming Interface (API) provided by OpenAI. The user can give the instructions, “Please predict the results for the target value for the input data XYZ!”, to locally call the trained prediction and agnostic model executables. The concepts of system and difference SHAP later introduced in Section 3.2 can also be executed to improve design comparisons and further analyze subsystem importance.



Hello! Here are some prompts you can use to analyze the data: ...

Great! Thank you. Please provide insights about globally important parameters. ...

Figure 1. Framework for explaining results from numerical finite element simulations

The LLM used is a company custom OpenAI Chat-GPT-4o version. This aspect is important, as simulation data for new products is confidential and data security issues must be considered. Aside from data and models, a context statement describing the general task is provided to the LLM. Within the context statement predefined tasks such as making a standard technical report of comparing the optimal with the initial variant are specified. In contrast to existing approaches, where the obtained results must be explained and documented manually, a substantial amount of time is saved. As previously highlighted by Hegselmann et al. (2024), LLMs indeed rely on the given data, which allows the engineer to collaborate with the AI and interactively chat with the data and models exploiting advanced reasoning capabilities. The capabilities have been shown in other fields such as production error explanation as in Roy et al. (2024), mechanical design in Jadhav et al. (2024), as well as explanation of global feature importances based on SHAP values in Hsu et al. (2024). The obtained engineering explanation can be documented in a knowledge storage system and help to set up further analysis.

3.2. Exploiting SHAP value additivity for improved result comparison

For an improved system understanding and comparison of simulation configurations the additivity of SHAP values (Lundberg et al., 2020) is exploited. On one side, this property allows for the aggregation of features describing certain subsystems according to the paradigm of systems engineering. The contributions specified by SHAP value φ_i for each feature belonging to the subsystem S can be summed up to an overall subsystem attributionφ_S (Eq. 1). From an engineering perspective, these importances can be traced back to subsystems that may be manufactured by a particular supplier. Subsequently, changes can be implemented to improve overall system behavior. On the other, it permits the subtraction of features contributions for gaining insights into why results from two simulations differ (Eq. 2). In the context of optimization, this may involve comparing the optimized system design with the initial system design. Note that N is the overall number of input features. These functionalities are also integrated in the above framework. System SHAP and difference SHAP values can be calculated on demand by calling the corresponding executable.

$$\phi_s = \sum_{i \in S} \phi_i \tag{1}$$

$$f(x^A) - f(x^B) = \sum_{i=1}^{N} (\phi_i^A - \phi_i^B) \tag{2}$$

4. Validation – sill optimization in pole side crash

The validation of the framework is demonstrated using an industrial use case, an optimization of a side sill in a vehicle pole side crash as per the FMVSS-214 regulation. In the crash, the vehicle, initially moving sideways at a constant velocity, impacts a fixed pole at location of the driver's head center-of-gravity. This results in significant intrusions of the lateral structure of the vehicle. The side sill absorbs much of the vehicle's kinetic energy through plastic deformation. Due to the high computing time required for full vehicle crash simulations, which can take up to several days on high-performance clusters, and the associated costs, a submodel setup is generated as depicted on top of Figure 2.
This model contains only the side sill and connected point masses and inertias representing the vehicle and its movement. The computing time of the submodel is approximately 400 times faster than the full vehicle crash simulation, allowing for efficient data generation. Such submodel structures are regularly used in optimization, where a feasible correspondence to the original model must be maintained (Büttner et al., 2023). As depicted in the lower left side of Figure 2, 14 wall thicknesses of this side sill profile are included as design parameters with defined boundaries for the optimization. A Latin Hypercube Sampling scheme is used to generate 556 to be computed design variants, ensuring comprehensive coverage of the permissible design range for each parameter. The target value to be optimized in this example is the mass-specific energy absorption at 80 mm of intrusion EM_rel.

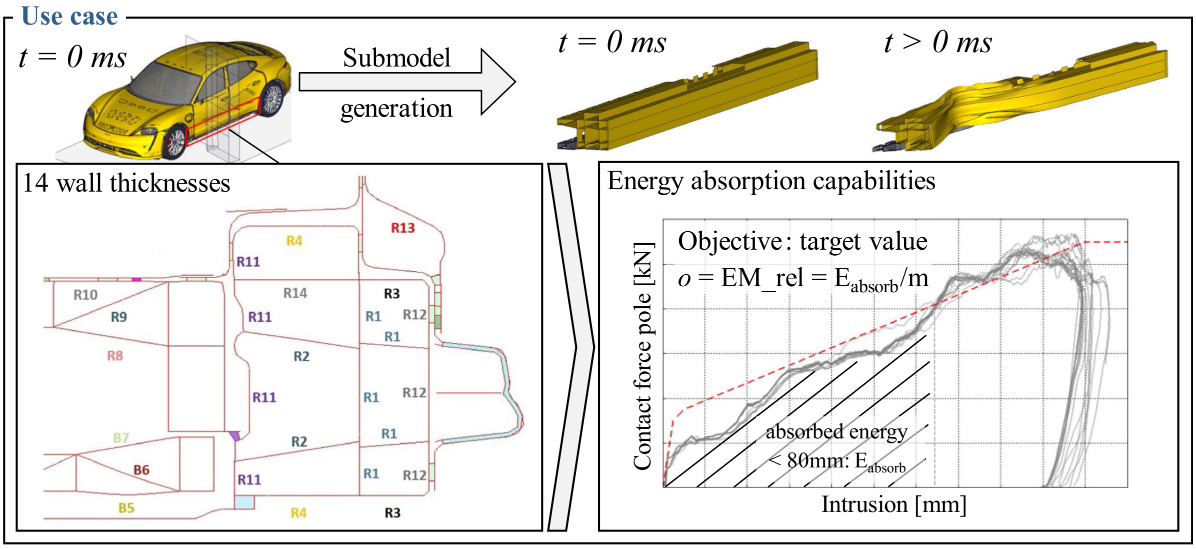


Figure 2. Sill optimization in pole side crash using a submodel with point-masses

4.1. Learning and optimizing system behavior

A gradient-boosting tree ML model is trained to predict EM_rel based on the simulation results of 556 variants, using the 14 wall thickness values as input features. The model accuracy, measured by the coefficient of determination (R^2), for the unseen test data is 0.80, as shown on the left side of Figure 3. If necessary, further evaluation of the model can be conducted using k-fold cross-validation. The slight overfitting of the prediction model may primarily be attributed to the high nonlinearity inherent in the crash event, evidenced by buckling of the walls. For the present case, accuracy is considered sufficient. Figure 3 also shows on the right side the optimization results of EM_rel. It increased by about 20 %, from 13.4 J/kg in the initial variant to 16.5 J/kg, and approximately doubled compared to the worst variant, which shows only 8.0 J/kg

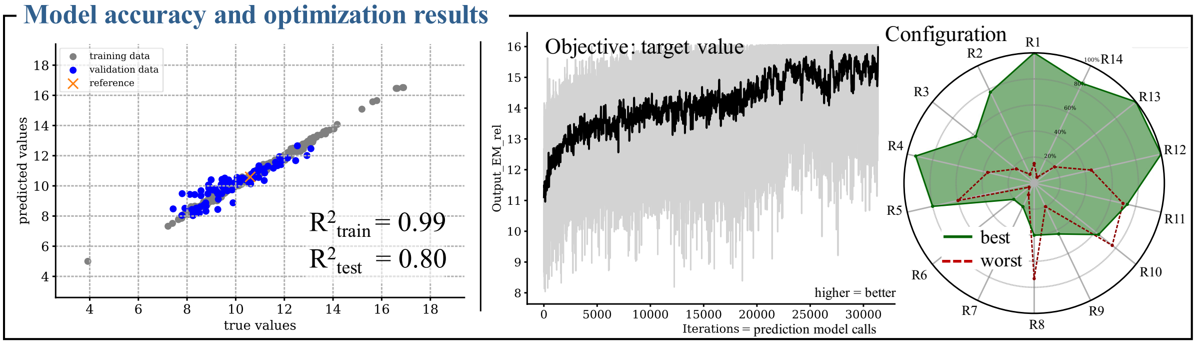
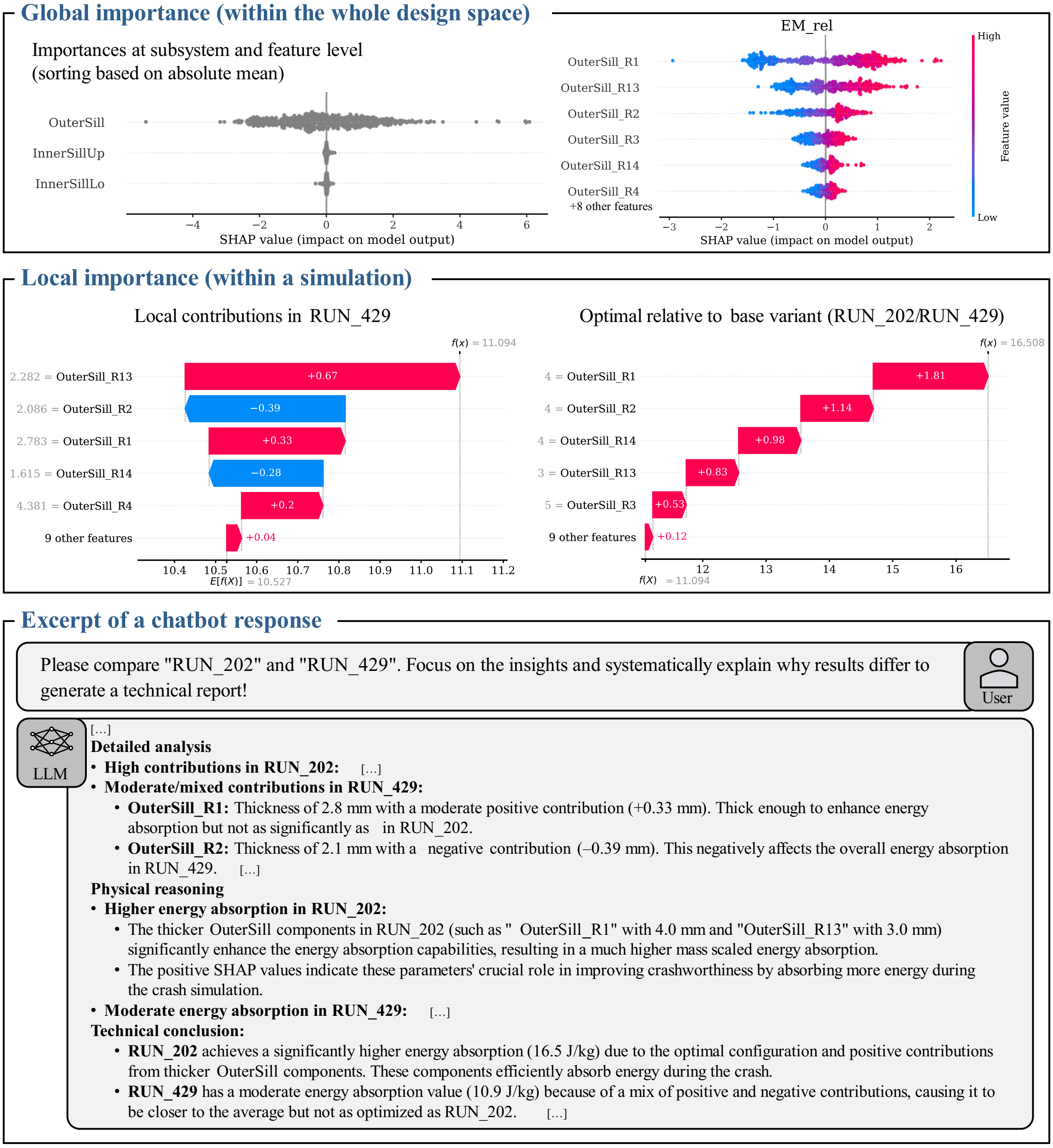


Figure 3. Scatter plot for true vs. predicted values and optimization results

4.2. Explaining the system behavior using SHAP and an LLM

At the top of Figure 4, the global importance analysis, derived from SHAP values, helps to identify significant design parameters (right) and subsystems (left) influencing the optimized target value EM_rel. Furthermore, it enables the verification of the learned relationships within the prediction model for reasonableness. The analysis reveals a higher importance of the outer sill wall thicknesses compared to the those of the inner sill. This is reasonable, as the specific energy absorption EM_rel is evaluated at an intrusion of 80 mm. Globally important wall thicknesses are R1, R13, and R2, indicated by the large range of SHAP values for each design parameter. In the center of Figure 4, the SHAP values for a single simulation result are presented, illustrating the local contribution and, consequently, the importance of the parameters for that specific outcome. These waterfall plots depict the cumulative summation of each design parameter's SHAP value, starting from the expected value at the bottom and progressing upwards, resulting in the predicted value at the top. This enables a detailed understanding of the prediction model's reasoning behind its output for the given design parameter values. For clarity, only the five most influential design parameters are displayed in the plots, along with the aggregated influence of the remaining nine. For an arbitrarily selected reference variant RUN_429, counteracting contributions of R13/R1 and R2/R14 become evident. Relative to this reference variant, difference SHAP values are calculated in relation to the optimal configuration RUN_202 and are shown on the right plot. The main differences can be attributed to R1 and R2. As no counteracting influences are visible, the optimal solution seems plausible as all design parameters contribute positively regarding EM_rel. Combining general engineering judgment with these analyses, which reveal the global importance and the local contributions of design parameters on the target value, allow for the rapid development of a comprehensive system understanding. However, engineers may find these plots and analyses unintuitive, or they may lack the necessary expertise for their proper interpretation. These concerns are addressed by using the LLM, as this interface allows engineers to analyze results in the form of text in a collaborative and interactive manner. An excerpt of a response comparing two simulation runs is visualized at the lower end of Figure 4. Text and figures similarly as displayed can directly be placed on slides providing the engineer with the basis for a technical report of the optimization campaign. Aside from the analysis itself, synthesis in the form of recommendations for actions to perform certain changes, similarly to the work of Jadhav et al. (2024) may be realized but is discussed as part of future work. The total execution time of the framework, encompassing the training of the ML model to the point where the LLM is ready for interaction with text data and executables, is 85 seconds on a workstation computer for this example.

The LLM is interfaced on the workstation but resides and executes on a separate machine. This execution time allows easy integration of the framework into the development process.



Please compare "RUN_202" and "RUN_429". Focus on the insights and systematically explain why results differ to generate a technical report!

[...]

Detailed analysis

- **High contributions in RUN_202:** [...]
- **Moderate/mixed contributions in RUN_429:**
  - **OuterSill_R1:** Thickness of 2.8 mm with a moderate positive contribution (+0.33 mm). Thick enough to enhance energy absorption but not as significantly as in RUN_202.
  - **OuterSill_R2:** Thickness of 2.1 mm with a negative contribution (–0.39 mm). This negatively affects the overall energy absorption in RUN_429. [...]

Physical reasoning

- **Higher energy absorption in RUN_202:**
  - The thicker OuterSill components in RUN_202 (such as " OuterSill_R1" with 4.0 mm and "OuterSill_R13" with 3.0 mm) significantly enhance the energy absorption capabilities, resulting in a much higher mass scaled energy absorption.
  - The positive SHAP values indicate these parameters' crucial role in improving crashworthiness by absorbing more energy during the crash simulation.
- **Moderate energy absorption in RUN_429:** [...]

Technical conclusion:

- **RUN_202** achieves a significantly higher energy absorption (16.5 J/kg) due to the optimal configuration and positive contributions from thicker OuterSill components. These components efficiently absorb energy during the crash.
- **RUN_429** has a moderate energy absorption value (10.9 J/kg) because of a mix of positive and negative contributions, causing it to be closer to the average but not as optimized as RUN_202. [...]

Figure 4. Results of explaining and documenting the system behavior in the validation use case

4.3. Discussion of results

The discussion of the results is based on the main objectives formulated in Section 2.4. Regarding the first objective, both local and global feature importances have been successfully extracted with the proposed framework using a gradient-boosting tree ML model and SHAP values within the validation example. This allows the rapid identification of important design parameters. Furthermore, the SHAP value for a design parameter represents the influence its parameter value has on the target variable in the simulation of interest. Such an importance measure is very intuitive to engineers. Consequently, this highlights the potential of data-driven prediction models and SHAP values for improving the analysis of numerical simulation in general, as this aspect also has not been comprehensively covered in literature yet. The second objective, namely, to pass data from the SHAP analysis, the prediction and agnostic model executables to an LLM, for enabling an intuitive and interactive analysis for the engineer with the LLM, is also achieved. This confirms the findings of Hegselmann et al. (2024), who observed that additionally provided data can be used as a basis for text generation with an LLM. The added value of incorporating the LLM in this context is the improved result accessibility either as a collaborative chat or automated generation of technical reports, which the latter has not been further explored in this research. Furthermore, the proposed framework was successfully applied in an industry-relevant example of the side sill wall thickness optimization.

However, the proposed framework also has its limitations. First, the prediction model is unaware of physical relationships such as energy conservation. Although the training data is physically meaningful, this cannot be guaranteed for the rationale of the prediction model. Tracking accuracy and reasonableness, as well as preventing overfitting and bias in the training of the model through careful evaluation of the results, is crucial. Uncertainty measures like those used in Büttner et al. (2023) for predictions may further improve trustworthiness. Additionally, literature highlights pitfalls, e.g., feature dependence or unjustified causal interpretation, when using SHAP (Molnar et al., 2022). These need to be considered to ensure proper application of the framework also from an AI-perspective making the use potentially more complex than it might initially appear. Although the LLM reasons correctly about the system in the validation example without additional training in analogy to Roy et al. (2023), which was spot-checked, hallucinations cannot be excluded for other applications or unchecked regions. This should be comprehensively evaluated in the next step, which can only be achieved by including domain knowledge for ensuring correct context and consistency as emphasized in Roscher et al. (2020).

5. Summary and future work

Modern passive vehicle safety systems are primarily developed virtually, resulting in a high number of simulations due to system complexity and nonlinearity. Hence, analyzing simulation results in multiple crash scenarios is time-consuming and requires experienced engineers. To reduce development time and increase system understanding, this work proposes a new approach incorporating xAI and LLMs to enhance the analysis of numerical simulations. Determined SHAP values based on a trained prediction model serve as feature importance measures, indicating the contribution of design parameters to the model's outcome. To further enhance result comparison, the additivity of SHAP values is exploited to introduce system and difference SHAP values. Available data of this analysis is transformed into a natural language string and provided along with executable functions to an LLM, which generates human-relatable textual explanations. This enables interactive analysis or can support technical report generation and thus makes complex results more accessible. The proposed framework was validated using a side sill optimization scenario in a vehicle pole side crash. The approach is modular and may be applied to similar problems in other domains such as thermal system development.

Future work can cover incorporating physical laws, such as energy conservation, directly into the prediction models to enhance the accuracy and trustworthiness of the results. To improve user interaction and trust, a systematic analysis of the limitations, the correctness and the usefulness of the LLM explanations need to be conducted. An exploration of different validation examples, such as the analysis of static structural simulations, can further demonstrate generalizability.

References

Ahmed, T., Ghosh, S., Bansal, C., Zimmermann, T., Zhang, X., & Rajmohan, S. (2023). Recommending root-cause and mitigation steps for cloud incidents using large language models. *In Proceedings of the 45th International Conference on Software Engineering (ICSE '23)* (pp. 1737-1749). IEEE Press.

Beyer, F., Schneider, D., & Schumacher, A. (2021). Finding three-dimensional layouts for crashworthiness load cases using the graph and heuristic based topology optimization. *Structural and Multidisciplinary Optimization*, 63, 1-15.

Bickel, S., Goetz, S., & Wartzack, S. (2023). Detection of plausibility and error reasons in finite element simulations with deep learning networks. *Algorithms*, 16(209).

Bordt, S., Nori, H., Rodrigues, V., Nushi, B., & Caruana, R. (2024). Elephants never forget: Memorization and learning of tabular data in large language models. *arXiv Preprint, abs/2404.06209.*

Borse, A., Gulakala, R., & Stoffel, M. (2024). Development of a machine learning-based design optimization method for crashworthiness analysis. *Archive of Mechanical Engineering*, 76(1-2), 61-92.

Büttner, J., Schumacher, A., Bäck, T., Schwarz, S., & Krause, P. (2023). Making multidisciplinary optimization fit for practical usage in car body development. *Structural and Multidisciplinary Optimization*, 66(62).

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. *In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.*

Conti, Z., & Kaijima, S. (2021). Explainable ML: Augmenting the interpretability of numerical simulation using Bayesian networks. In I. As & P. Basu (Eds.), *The Routledge Companion to Artificial Intelligence in Architecture* (1st ed., pp. 21). Routledge.

Diez, C., Kunze, P., Toewe, D., Wieser, C., Harzheim, L., & Schumacher, A. (2018). Big-data based rule-finding for analysis of crash simulations. In A. Schumacher, T. Vietor, S. Fiebig, K.-U. Bletzinger, & K. Maute (Eds.), *Advances in Structural and Multidisciplinary Optimization* (pp. 396-410). Springer.

Feldkamp, N., & Strassburger, S. (2023). From explainable AI to explainable simulation: Using machine learning and XAI to understand system robustness. *In Proceedings of ACM SIGSIM Conference on Principles of Advanced Discrete Simulation (SIGSIM-PADS '23)* (pp. 197-198). ACM, New York, NY.

Gonter, M., Knoll, P., Leschke, A., Seiffert, U., & Weinert, F. (2021). Fahrzeugsicherheit. In S. Pischinger & U. Seiffert (Eds.), *Vieweg Handbuch Kraftfahrzeugtechnik* (pp. 1073-1160). Springer Vieweg.

Hahner, S., Iza Teran, R., & Garcke, J. (2020). Analysis and prediction of deforming 3D shapes using oriented bounding boxes and LSTM autoencoders. In I. Farkaš, P. Masulli, & S. Wermter (Eds.), *Artificial Neural Networks and Machine Learning – ICANN 2020* (pp. 253-270). Springer.

Hegselmann, S., Buendia, A., Lang, H., Agrawal, M., & Jiang, X. (2023). TabLLM: Few-shot classification of tabular data with large language models. *In Proceedings of the 26th International Conference on Artificial Intelligence and Statistics, Proceedings of Machine Learning Research*, 206, 5549-5581.

Horii, H. (2017). Multi-objective optimization of vehicle occupant restraint system by using evolutionary algorithm with response surface model. *International Journal of Computational Methods and Experimental Measurements*, 5(2), 163-173.

Hsu, C.-C., Wu, I.-Z., & Liu, S.-M. (2024). Decoding AI complexity: SHAP textual explanations via LLM for improved model transparency. *In Proceedings of the 2024 International Conference on Consumer Electronics - Taiwan (ICCE-Taiwan)* (pp. 197-198). Taichung, Taiwan.

Iza Teran, R., & Garcke, J. (2019). A geometrical method for low-dimensional representations of simulations. *Journal on Uncertainty Quantification*, 7(2), 472-496.

Jadhav, Y., & Farimani, A. B. (2024). Large language model agent as a mechanical designer. *arXiv Preprint, abs/2404.17525.*

Janzing, D., Minorics, L., Bloebaum, P. (2020). Feature relevance quantification in explainable AI: A causal problem. *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics, in Proceedings of Machine Learning Research* 108:2907-2916.

Kracker, D., Dhanasekaran, R. K., Schumacher, A., & Garcke, J. (2023). Method for automated detection of outliers in crash simulations. *International Journal of Crashworthiness*, 28(1), 96-107.

Lin, P., Zhang, L., & Tiong, R. L. K. (2023). Multi-objective robust optimization for enhanced safety in large-diameter tunnel construction with interactive and explainable AI. *Reliability Engineering & System Safety*, 234, 109172.

Lundberg, S. M., Erion, G., & Chen, H. *et al.* (2020). From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*, 2, 56-67.

Mathieu, J., Di Roberto, M., & Vielhaber, M. (2023). A data-driven methodology for the automated analysis and explanation of system behavior in crash simulations. *13th European LS-DYNA Users Conference*, Baden-Baden, Germany.

Mathieu, J., Gupta, P., Di Roberto, M., & Vielhaber, M. (2024). Minimizing occupant loads in vehicle crashes through reinforcement learning-based restraint system design: Assessing performance and transferability. *Proceedings of the Design Society*, 4, 2139-2148.

Molnar, C., Holzinger, A., Goebel, R., Fong, R., Moon, T., Müller, K. R., & Samek, W. (2022). General pitfalls of model-agnostic interpretation methods for machine learning models. In A. Holzinger, R. Goebel, R. Fong, T. Moon, K. R. Müller, & W. Samek (Eds.), *xxAI - Beyond Explainable AI* (vol. 13200, pp. 1-18). Springer.

Roscher, R., Bohn, B., Duarte, M. F., & Garcke, J. (2020). Explainable machine learning for scientific insights and discoveries. *IEEE Access*, 8, 42200-42216.

Roy, D., Zhang, X., Bhave, R., Bansal, C., Las-Casas, P., Fonseca, R., & Rajmohan, S. (2024). Exploring LLM-based agents for root cause analysis. *In Companion Proceedings of the 32nd ACM International Conference on the Foundations of Software Engineering (FSE Companion'24) (12 pages).* Porto de Galinhas, Brazil.

Thole, C.-A., Nikitina, L., Nikitin, I., & Clees, T. (2010). Advanced mode analysis for crash simulation results. *9th LS-DYNA Forum*, Bamberg, Germany.

Trilling, J., Schumacher, A., & Zhou, M. (2024). Reinforcement learning-based agents for improving layouts of automotive crash structures. *Applied Intelligence*, 54, 1751-1769.