engineering proceedings

MDPI

Proceeding Paper

Initial Study on Machine Learning Algorithm Selection for Crash Box Performance Prediction Using AutoGluon †

Zoja Barančoková [1,*], Petr Koňas [2], Kamil Řehák [1] and Radka Janečková [1]

[1] Institute of Automotive Engineering, Faculty of Mechanical Engineering, Brno University of Technology, 616 69 Brno, Czech Republic; rehak@fme.vutbr.cz (K.Ř.); radka.janeckova@vutbr.cz (R.J.)
[2] Energy Institute, Faculty of Mechanical Engineering, Brno University of Technology, 616 69 Brno, Czech Republic; konas@fme.vutbr.cz
* Correspondence: zoja.barancokova@vutbr.cz
† Presented at the Sustainable Mobility and Transportation Symposium 2026, Győr, Hungary, 8 October 2026.

Abstract

This study proposes an AutoML-based surrogate-modeling framework for crash box design optimization. A dataset consisting of 96 explicit finite-element crash simulations was generated for crash boxes with a wall thickness of 1.0–2.5 mm and six cross-sectional profiles: square, rectangular, circular, elliptical, hexagonal, and octagonal. AutoGluon-TimeSeries was used to benchmark machine learning models for predicting absorbed crash energy and crush displacement from geometric design variables. Among the evaluated models, Weighted Ensemble achieved the highest accuracy for absorbed energy prediction, with an R^2 = 0.975, while Temporal Fusion Transformer achieved an R^2 = 0.9863 for deformation-history prediction. The selected surrogate models were then used to support iterative sampling of the design space, significantly reducing the number of expensive finite-element evaluations required for optimization. These results show that AutoML-based surrogate modeling can accelerate crash box design and provide a practical foundation for data-efficient crashworthiness optimization.

Keywords: surrogate modeling; crashworthiness optimization; automated machine learning; crash box

1. Introduction

Nowadays, the maximization of occupant safety is one of the main objectives in vehicle design. Each vehicle consists of specific deformation zones, known as crumple zones. Their function is to absorb as much impact energy as possible to prevent the kinetic energy from transferring to the rest of the vehicle's structure and the passenger cabin.

A Crash Management System (CMS) is a critical component of the energy-absorbing structure. It includes a crash box, which is located at the front and the rear part of the vehicle and connects the bumper to the longitudinal beams. Its main function is to absorb kinetic impact energy, reduce energy transfer, and minimize the risk of injury. During the designing process, the goal is to find a balance between the geometric parameters of the crash box and its material properties to utilize the full energy absorption capacity and the progressive deformation mode. Furthermore, engineers must consider different designs based on the type of vehicle, manufacturing costs, and weight minimization of the overall structure.

Check for updates

Academic Editors: Gábor Dogossy, Szabolcs Kocsis-Szürke, Dániel Csikor, Dániel Serfőző, Miklós Kuczmann, András Lajos Nagy, László Lendvai and Ambrus Zelei

Published: 22 September 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

Given the large number of variables, this is a highly complex engineering problem. The integration of machine learning (ML) algorithms into the design process facilitates a significant reduction in computational time, which is required for geometric design optimization. Several studies have investigated the application of ML in crash box optimization. Lu et al. [1] integrated polynomial Response Surface Methodology (RSM) with a Non-Dominated Sorting Genetic Algorithm (NSGA-II) for tailor-rolled tubes. Wang et al. [2] applied Least-Squares Support Vector Regression (LS-SVR) and an Improved Particle Swarm Optimization (IPSO) algorithm to a crash box with a re-entrant auxetic core. As Artificial Neural Networks (ANNs) have become a dominant surrogate-modeling approach, Li et al. [3] combined an ANN with NSGA-II for 3D-printed origami cells, and Tesfaye et al. [4] integrated ANNs, Taguchi methods, and genetic algorithms (GAs) for foam-filled crash box structures. Additionally, Wang et al. [5] used RSM and evolutionary algorithms for biomimetic crash box designs. Advanced architectures and data-driven methods have further expanded these capabilities. Sudirja et al. [6] employed a Gated Recurrent Unit (GRU) for corner reinforcements, and Jusuf et al. [7] utilized Gaussian Process Regression (GPR) with Bayesian optimization for design space exploration. To address the challenge of limited training data, Colella et al. [8] demonstrated the effectiveness of fine-tuning a pre-trained Multilayer Perceptron (MLP). For highly complex frameworks, Borse et al. [9] combined Deep Convolutional Generative Adversarial Network (DC-GAN) data augmentation, an XGBoost surrogate, and Deep Q-Network (DQN) reinforcement learning (RL) for inverse crash box design, while Wang et al. [10] introduced an MOEA/D-DAE algorithm for many-objective Pareto optimization.

This paper focuses on the development of a surrogate model through the application of an AutoML framework, specifically the AutoGluon-TimeSeries, to develop a robust surrogate model for crash box optimization. Based on an initially generated dataset, the framework evaluates predefined global neural models, ranks their performance, and selects the best-performing candidates for predicting absorbed crash energy and crush displacement directly from geometric design variables. This approach could minimize the need to generate a large number of simulations manually, which could lead to a significant reduction in overall computational time.

2. Materials and Methods

2.1. Dataset Description

For surrogate model training, the dataset consisted of 96 types of crash box geometries, and it represents a baseline that will be further augmented. Therefore, only fundamental geometric variations were designed, namely, changes in cross-sectional geometry (Figure 1) with wall thickness varying between 1 and 2.5 mm. The dataset was generated using Finite-Element (FE) simulations performed in PAM-CRASH solver v2020.5. From the obtained results, energy–time and displacement–time curves were extracted and added to the dataset. Each simulation was identified by a unique ID and had the end time set to 150 ms with a time step of 2 ms. This configuration resulted in the creation of the dataset, where each simulation consisted of 76 rows. The dataset was structured in long format, where each row represented a single time step of a given simulation run.

2.2. Surrogate Model Configuration

AutoGluon-TimeSeries v1.5.0 [11,12] was used as the AutoML surrogate-modeling framework for identifying the most suitable model. It is designed to predict future values of multiple time series based on historical data and related covariates. The fit() function trains multiple models to generate accurate probabilistic forecasts. The module includes simple statistical forecasting models, deep learning models, tree-based models, and an ensemble

https://doi.org/10.3390/engproc2026157005

that integrates individual model predictions to generate multi-step-ahead probabilistic forecasts for univariate time series data.

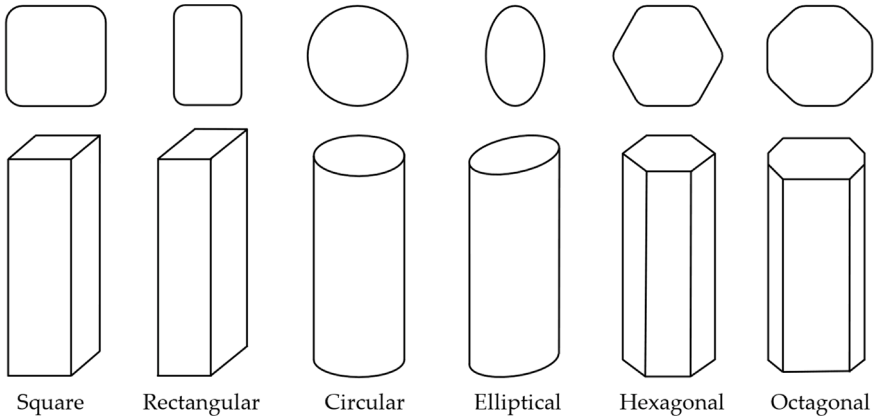


Figure 1. Cross-sectional variants of the crash boxes included in the dataset.

The following global neural models were selected for training:

- DeepAR [13]—a Recurrent Neural Network-based (RNN) autoregressive model;
- PatchTST [14]—a patch-based transformer model;
- DLinear [15]—a lightweight trend-decomposition network;
- TiDE [16]—an encoder–decoder model for long-horizon prediction;
- Simple Feed Forward (SFF)—a direct multi-step feedforward network;
- Temporal Fusion Transformer (TFT) [17]—an LSTM–transformer hybrid with probabilistic outputs.

A Weighted Ensemble model, which combined all trained models, was constructed after the training of all individual models. The dataset was split into 69 samples for training, 12 for validation, and 15 for testing. The simulations used in the validation subset were selected using stratified sampling. This approach ensured that each of the six cross-section designs was included in validation. The time axis was determined as a known covariate, because this value is fixed for each simulation. Crash energy and crush displacement, which are time-dependent curves that increase throughout the crash event, were defined as target variables. The static features included cross-sectional geometry type, wall thickness, corner radius, and width and height of the profile.

All hyperparameters, such as learning rate and maximum number of epochs, were tuned automatically by the framework. The training time for each model was set to 1800 s. The number of historical points that the model has access to during the training is defined by the context length parameter. This was the only parameter which was set manually through a grid search of context lengths ranging from 15 to 33. The optimal value was determined based on the highest predictive accuracy on the test set, and it was obtained for the context_length = 32. It ensured that the peaks of both the crash energy and crush displacement curves fell outside this window. With context length fixed, the prediction length was set to 44.

Model performance was evaluated using the coefficient of determination (R^2), Mean Absolute Error (MAE), and Root Mean Square Error (RMSE), calculated as follows [18,19]:

$$R^2 = 1 - \frac{\sum_{i=1}^{N}\sum_{t=T+1}^{T+H}(f_{i,t} - y_{i,t})^2}{\sum_{i=1}^{N}\sum_{t=T+1}^{T+H}(\overline{y} - y_{i,t})^2}, \quad (1)$$

$$\mathrm{MAE} = \frac{1}{N}\frac{1}{H}\sum_{i=1}^{N}\sum_{i=1}^{T+H}|f_{i,t} - y_{i,t}|, \quad (2)$$

https://doi.org/10.3390/engproc2026157005

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\frac{1}{H}\sum_{i=1}^{N}\sum_{i=1}^{T+H}(f_{i,t} - y_{i,t})}, \quad (3)$$

where $y_{i,t}$ is the observed value of time series i at time t, $f_{i,t}$ is the corresponding predicted value, $\overline{y}$ is the mean of all observed values within the forecast horizon, N denotes the number of time series in dataset, H is the forecast horizon length, and T is the length of the observed time series.

3. Results and Discussion

The evaluation results for the trained global neural architectures are presented in Table 1. Among the seven trained architectures, only three (Weighted Ensemble, TiDE, and TFT) reached a coefficient of determination R^2 above 0.95, which was reached for both target values. The remaining architectures (DLinear, DeepAR, PatchTST, and SFF) demonstrated low accuracy for absorbed crash energy, for which R^2_{E} values ranged from 0.6 to 0.74. However, their predictions of crush displacement curves were considerably more accurate, with R^2_{D} ranging between 0.88 and 0.96. The difference in their performance shows that the prediction of crash energy absorption curves is a more challenging task. Prediction of these types of curves requires utilization of more complex models that can capture the nonlinear response and differentiation caused by geometric parameters. Approaches that rely only on simple temporal decomposition, such as DLinear, or provide weak combinations of static geometric features with time−dependent signals (PatchTST, SFF, and DeepAR) appear to be inaccurate for these types of predictions. In contrast, architectures such as TFT or TiDE include architectural mechanisms that are better suited for capturing the influence of geometric parameters on the predicted response.

Table 1. Comparison of the performance of the selected global neural models for absorbed crash energy and crush displacement curves.

Model	R^2_{E} (−)	$\mathrm{MAE_E}$ (J)	$\mathrm{RMSE_E}$ (J)	R^2_{D} (−)	$\mathrm{MAE_D}$ (J)	$\mathrm{RMSE_D}$ (J)
Weighted Ensemble	0.9750	424	506	0.9811	6.49	8.13
TiDE	0.9673	457	578	0.9806	5.80	8.24
TFT	0.9538	497	687	0.9863	5.60	6.91
DLinear	0.7350	1377	1647	0.9574	7.82	12.19
DeepAR	0.7186	1017	1697	0.8849	11.98	20.04
PatchTST	0.6715	1255	1833	0.9387	10.95	14.63
SFF	0.6047	1508	2011	0.9512	10.08	13.05

The mean value of the maximum absorbed energy for all simulation runs in the dataset was approximately 15,500 J, while the mean value of the maximum crush displacement reached approximately 229 mm. This indicates that the values of MAE and RMSE reported in Table 1 are relatively small compared to the overall values.

The composition of the Weighted Ensemble model is reported in Tables 2 and 3 for crush displacement and crash energy prediction, respectively. The ensemble weights were obtained using a greedy algorithm that minimizes the Mean Absolute Error (MASE) on the validation set. For the energy prediction task, TiDE dominates the ensemble, with a weight of 92%, while the remaining architectures make only minor contributions. The ensemble for displacement prediction is more balanced. TiDE is the larger contributor, with a weight of 49.2%, followed by SFF (33.9%) and DLinear (16.9%). Despite the strong performance in the testing of individual models, TFT is absent in the ensemble. This is because of the difference between the validation MASE used for ensemble selection and the R^2 used for the test set. The limited validation data likely amplified this difference.

https://doi.org/10.3390/engproc2026157005

Table 2. Composition of the Weighted Ensemble for crush displacement prediction.

Model	Weight (%)
TiDE	49.2
SFF	33.9
DLinear	16.9

Table 3. Composition of the Weighted Ensemble for absorbed-energy prediction.

Model	Weight (%)
TiDE	92
DLinear	3
SFF	3
PatchTST	1
TFT	1

Figure 2 shows the predictions of the best-performing models (Weighted Ensemble, TiDE, and TFT) for one representative test simulation characterized by a progressive deformation mode. All three approaches captured the shape of both target curves, including the rise, the peak, and the rebound.

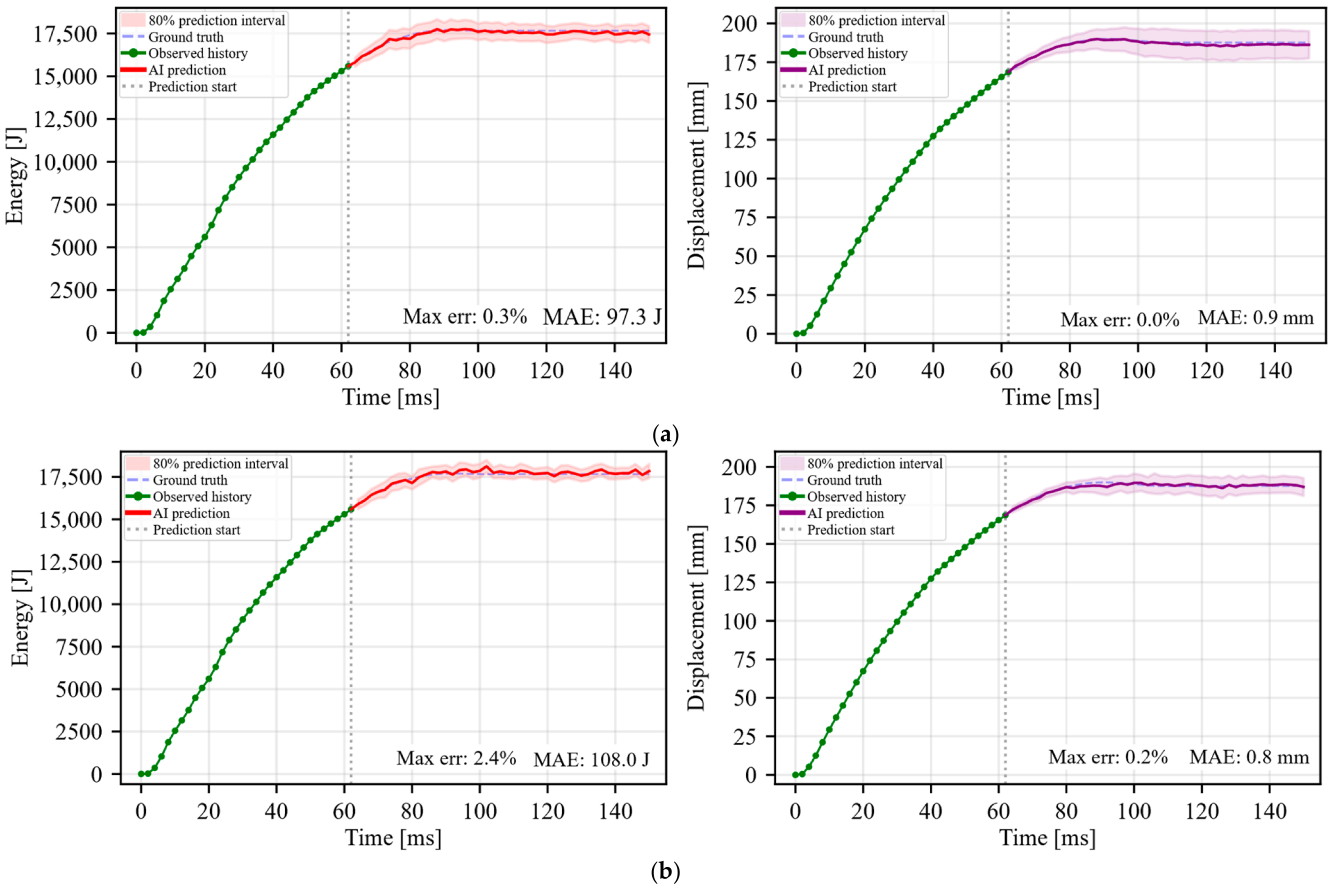
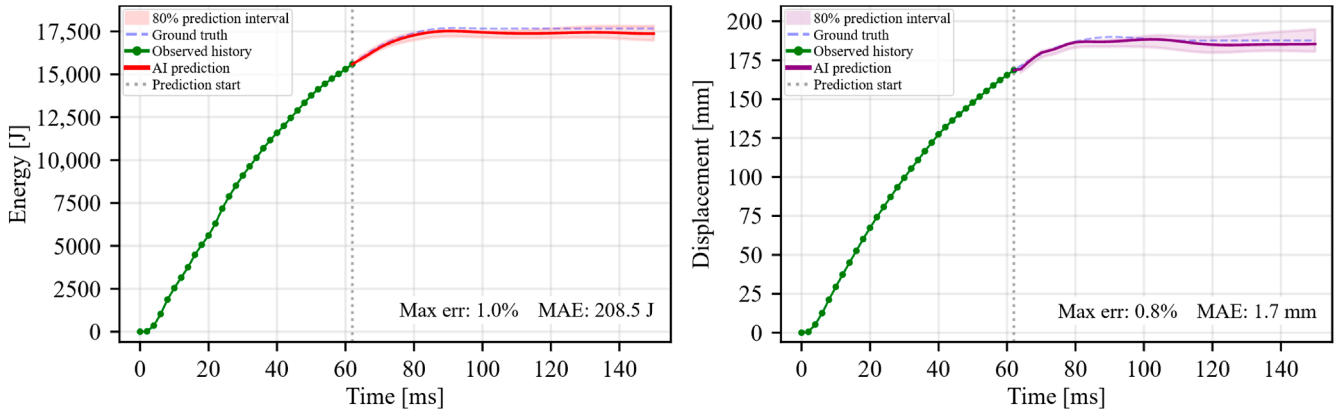


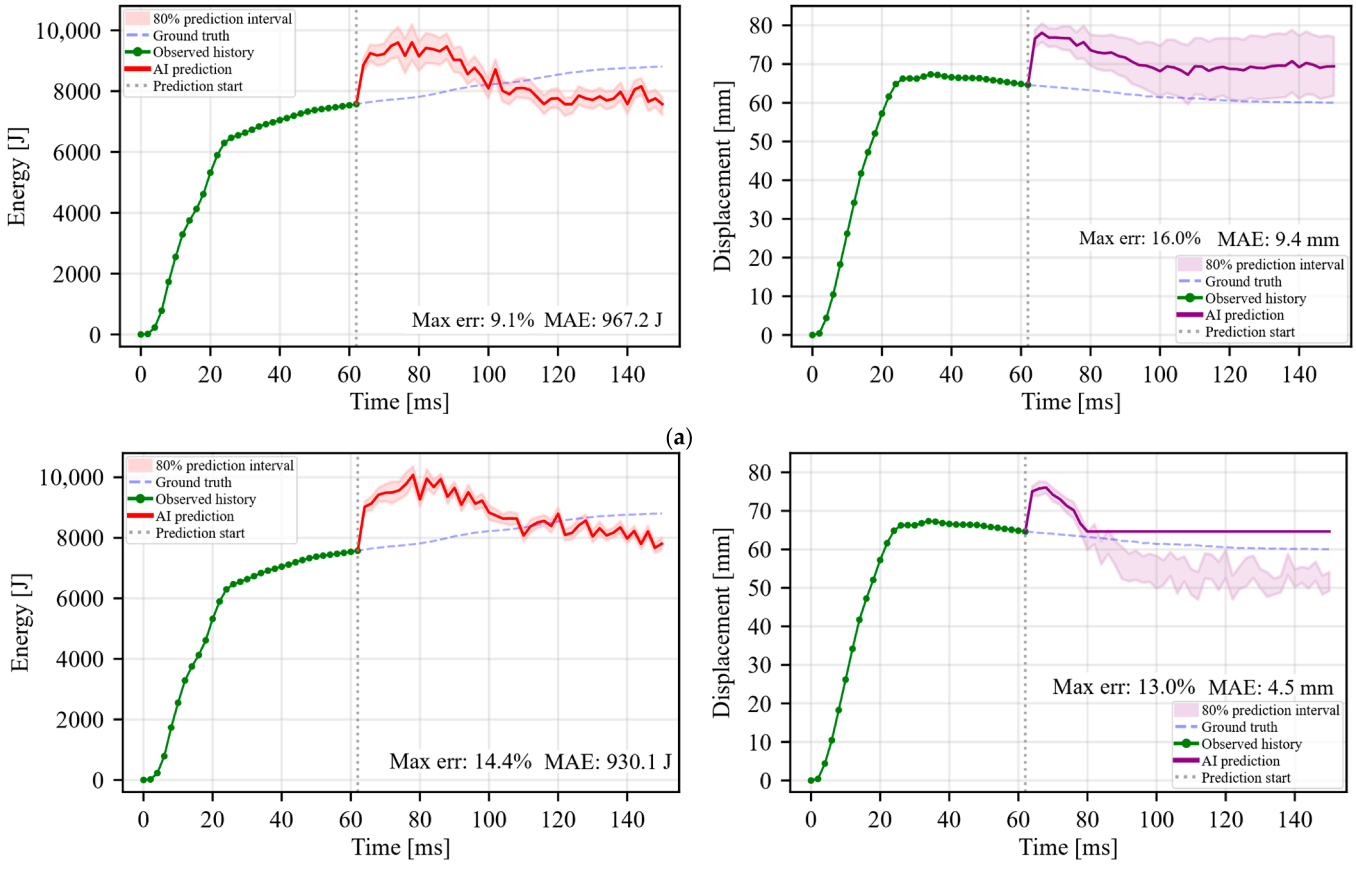
Figure 2. *Cont.*



(**c**)

Figure 2. Surrogate model predictions of crash energy and crush displacement for progressive deformation mode: (**a**) Weighted Ensemble; (**b**) TiDE; (**c**) Temporal Fusion Transformer.

Despite achieving similar evaluation scores, TFT behaved noticeably differently relative to the other dominant models. The prediction curves obtained by TFT indicated continuous and smooth behavior across all evaluated simulations. This contrast was especially visible in simulations with a buckling deformation mode, as shown in Figure 3.

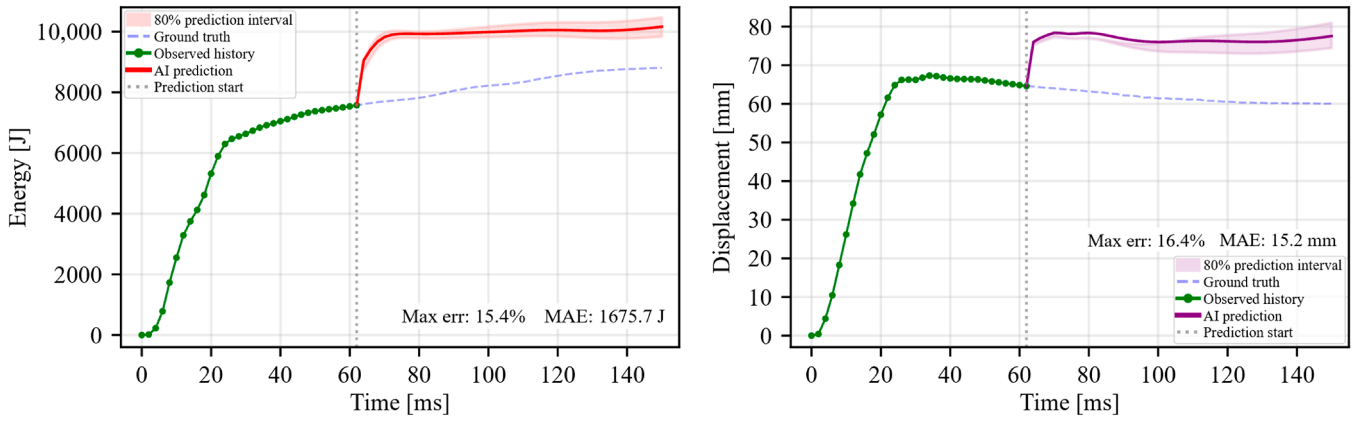


(**a**)

(**b**)

Figure 3. *Cont.*

https://doi.org/10.3390/engproc2026157005



(**c**)

Figure 3. Surrogate model predictions of crash energy and crush displacement for buckling deformation mode: (**a**) Weighted Ensemble; (**b**) TiDE; (**c**) Temporal Fusion Transformer.

This behavior can be explained by the architecture of the TFT model. The architecture contains the LSTM-based recurrent component that allows the network to retain and update information across time steps. TiDE, by contrast, follows an encoder–decoder design that compresses the context into a single, compact representation. It then generates the entire future horizon from that vector. As a result, nearby predicted points may be partially independent, and the reconstructed curve may exhibit local inconsistencies.

The Weighted Ensemble combines the strength of the individual models. Since TiDE carries the largest weight in the ensemble, the predictions inherit the encoder–decoder characteristic and therefore also show less continuity between predicted points.

Figure 3 shows a simulation exhibiting a buckling deformation mode. This simulation reaches the peak of the crash energy and crush displacement curves before the end of the context window, which differentiates it from the others with progressive deformation mode in the training set. All three evaluated models demonstrated increased prediction error for this deformation mode, which is expected given the limited representation of buckling simulations in the training set.

4. Conclusions

This study addressed the challenge of computationally expensive complex engineering task of crash box optimization by developing an AutoML-based surrogate-modeling framework. The findings demonstrate that the integration of global neural network architectures and hyperparameter tuning provides a practical and data-efficient foundation for accelerating crashworthiness optimization. This approach offers a practical step toward the integration of machine learning surrogates into engineering design workflows. The models selected for this task include DeepAR, PatchTST, DLinear, TiDE, SFF, and TFT.

The evaluated models demonstrated their capacity to predict the behavior of both target variables, namely, absorbed crash energy and crush displacement. For energy prediction, the Weighted Ensemble achieved the highest accuracy, with an $R^2_{\mathrm{E}} = 0.975$, while TFT obtained an $R^2_{\mathrm{D}} = 0.9863$ for crush displacement prediction. Visual analysis of the predicted curves further suggests that, despite the lower R^2 values for energy, the TFT architecture would be preferable for both target variables in future research. This is due to its LSTM-based design, which naturally produces more stable and continuous predictions that are more physically consistent and better suited for subsequent research and optimization tasks.

The main direction of future research is the augmentation of the existing dataset. Increasing the number of simulations should improve the representation of the buckling

deformation mode and enhance the ability of the neural models to capture the curve behavior. Additionally, a larger dataset is expected to improve alignment between the internal metrics and test set performance and to reduce the risk of overfitting on the validation dataset.

Overall, these developments are expected to improve the robustness and generalization of surrogate models. This should enable more reliable prediction of absorbed crash energy and crush displacement across a wider range of crash scenarios. The successful implementation of surrogate models is expected to significantly reduce the computational time required for crash box design. This can be achieved by replacing repeated finite-element simulations with surrogate evaluations, which could enable broader exploration of the design space.

Author Contributions: Conceptualization, Z.B. and K.Ř.; methodology, Z.B. and K.Ř.; software, Z.B.; validation, Z.B. and P.K.; formal analysis, Z.B.; investigation, Z.B.; resources, K.Ř.; data curation, Z.B.; writing—original draft preparation, Z.B. and R.J.; writing—review and editing, P.K. and K.Ř.; visualization, Z.B. and R.J.; supervision, P.K.; project administration, K.Ř.; funding acquisition, K.Ř. All authors have read and agreed to the published version of the manuscript.

Funding: This publication was supported by the project "Innovative Technologies for Smart Low Emission Mobilities", funded as project No. CZ.02.01.01/00/23_020/0008528 by the Programme Johannes Amos Comenius, Intersectoral cooperation, and the Specific research on BUT FSI-S-26-8993.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The raw/processed data required to reproduce these findings cannot be shared at this time due to time limitations.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Lu, R.; Gao, W.; Hu, X.; Liu, W.; Li, Y.; Liu, X. Crushing analysis and crashworthiness optimization of tailor rolled tubes with variation of thickness and material properties. *Int. J. Mech. Sci.* **2018**, *136*, 67–84. [CrossRef]
2. Wang, T.; Li, Z.; Wang, L.; Hulbert, G.M. Crashworthiness analysis and collaborative optimization design for a novel crash-box with re-entrant auxetic core. *Struct. Multidisc. Optim.* **2020**, *62*, 2167–2179. [CrossRef]
3. Li, J.; Qu, M.; Jiang, Z. Multi-objective optimization for 3D printed origami crash box cell based on artificial neural networks and NSGA-II. *J. Mater. Process. Des.* **2023**, *7*, 1–13. [CrossRef]
4. Tesfaye, F.A.; Ali, A.N.; Fenta, E.W. Crashworthiness Optimization of Closed Cell–Sandwiched Aluminum Foam Crash Box Using FE and ANN Modeling. *Eng. Rep.* **2025**, *9*, e70353. [CrossRef]
5. Wang, C.; Li, Y.; Zhao, W.; Zou, S.; Zhou, G.; Wang, Y. Structure design and multi-objective optimization of a novel crash box based on biomimetic structure. *Int. J. Mech. Sci.* **2018**, *138–139*, 489–501. [CrossRef]
6. Sudirja; Ristiana, R.; Mardiati, R.; Kaleg, S.; Hapid, A.; Muharam, A.; Perdana, M.A.P.; Budiman, A.C.; Amin. Machine Learning-Based Crashworthiness Optimization of Crash Box Geometry by Adding Corner Shape Reinforcement Variations. In Proceedings of the 17th International Conference on Telecommunication Systems, Services, and Applications (TSSA), Lombok, Indonesia, 12–13 October 2023. [CrossRef]
7. Jusuf, A.; Jarwadi, M.H.; Hastungkorojati, D.G.; Gunawan, L.; Akbar, M.; Zakaria, K.; Izzaturrahman, M.F.; Palar, P.S. Design Exploration and Optimization of a Multi-Corner Crash Box under Axial Loading via Gaussian Process Regression. *Int. J. Technol.* **2024**, *15*, 1749–1770. [CrossRef]
8. Colella, G.; Lange, V.A.; Duddeck, F. Transfer learning for crash design. *Discov. Mech. Eng.* **2024**, *3*, 26. [CrossRef]
9. Borse, A.; Gulakala, R.; Stoffel, M. Multi-parameter design optimization of crash box for crashworthiness analysis. *Proc. Appl. Math. Mech.* **2024**, *24*, e202400096. [CrossRef]
10. Wang, W.; Dai, S.; Zhao, W.; Wang, C. Multi-objective optimization of hexahedral pyramid crash box using MOEA/D-DAE algorithm. *Appl. Soft Comput.* **2022**, *118*, 108481. [CrossRef]
11. Time Series Forecasting. Available online: https://auto.gluon.ai/stable/tutorials/timeseries/index.html (accessed on 11 May 2026).

12. Shchur, O.; Turkmen, C.; Erickon, N.; Shen, H.; Shirkov, A.; Hu, T.; Wang, Y. AutoGluon-TimeSeries: AutoML for Probabilistic Time Series Forecasting. *arXiv* **2023**, arXiv:2308.05566. [CrossRef]
13. Salinas, D.; Flunkert, V.; Gasthaus, J.; Januschowski, T. DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *Int. J. Forecast.* **2020**, *36*, 1181–1191. [CrossRef]
14. Nie, Y.; Nguyen, N.H.; Sinthong, P.; Kalagnanam, J. Time Series is Worth 64 Words: Long-term Forecasting with Transformers. *arXiv* **2023**, arXiv:2211.14730. [CrossRef]
15. Zeng, A.; Chen, M.; Zhang, L.; Xu, Q. Are Transformers Effective for Time Series Forecasting? *Proc. AAAI Conf. Artif. Intell.* **2023**, *37*, 11121–11128. [CrossRef]
16. Das, A.; Kong, W.; Leach, A.; Mathur, S.; Sen, R.; Yu, R. Long-term Forecasting with TiDE: Time-series Dense Encoder. *arXiv* **2023**, arXiv:2304.08424. [CrossRef]
17. Lim, B.; Arık, S.Ö.; Loeff, N.; Pfister, T. Temporal Fusion Transformers for interpretable multi-horizon time series forecasting. *Int. J. Forecast.* **2021**, *37*, 1748–1764. [CrossRef]
18. Forecasting TimeSeries—Evaluation Metrics. Available online: https://auto.gluon.ai/stable/tutorials/timeseries/forecasting-metrics.html (accessed on 11 May 2026).
19. Chicco, D.; Warrens, M.J.; Jurman, G. The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. *PeerJ Comput. Sci.* **2021**, *7*, e623. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.