Key Engineering Materials
ISSN: 1662-9795, Vol. 1050, pp 137-144
doi:10.4028/p-wKWw4q

Submitted: 2026-01-19
Revised: 2026-02-18
Accepted: 2026-02-18
Online: 2026-04-16

Graph Neural Network for Draw-in Prediction in Sheet Metal Stamping

Hui Wang[1,a], Taegyun Ahn[1,b], Lu Huang[2,c], Seog-Chan Oh[2,d], Liang Huang[3,e], Andrey Ilinich[3,f], Sathya Dev[4,g], Jeong Whan Yoon[1,5,h*]

[1]Department of Mechanical Engineering, Korea Advanced Institute of Science and Technology (KAIST), 291 Daehak-ro, Yuseong-gu, Daejeon 305-701, Republic of Korea

[2]Research & Development, General Motors Global Technical Center, 30470 Harley Earl Boulevard, Warren, MI 48092, USA

[3]Stamping Business Unit, Ford Motor Company, 2101 Village Rd, Dearborn, MI 48124, USA

[4]Stamping Engineering, Stellantis, 1000 Chrysler Dr, CIMS 485-04-03 Auburn Hills MI 48326, USA

[5]School of Engineering, Deakin University, 75 Pigdons Rd., Waurn Ponds, VIC 3216, Australia

[a]huiwang@kaist.ac.kr, [b]t.ahn@kaist.ac.kr, [c]lu.huang@gm.com, [d]seog-chan.oh@gm.com, [e]lhuang6@ford.com, [f]ailinich@ford.com , [g]sathya.dev@stellantis.com, [h]j.yoon@kaist.ac.kr

Keywords: Graph Neural Networks, Draw-in prediction, Sheet metal forming, Machine learning.

Abstract. Draw-in distance is a key index for evaluating the quality of sheet metal stamping. Its accurate prediction is therefore required for tool design and process control. Traditional finite element (FE) simulations, while accurate, are computationally intensive and time-consuming for iterative design optimization. In this study, a graph neural network (GNN) method is proposed to predict draw-in during sheet metal forming. A dataset was built from FE simulations with different process settings, including blank holder force and draw bead force. The GNN model uses node coordinates and edge features to describe the spatial relations in the sheet. A multi-level loss function was applied. The coordinate error and edge distance error were included. In this way, the shape of the sheet is better preserved. The trained GNN can be used as a fast model for draw-in prediction. It can also be used for inverse analysis, where the process parameters are found from a given draw-in result. This provides an efficient tool for sheet metal forming design and optimization.

Introduction

Accurate prediction of sheet flow during stamping is required to obtain high-quality formed parts and to control part size [1,2]. In industry, finite element (FE) simulation is widely used to study material flow, strain, and springback in metal forming [3–5] for its reliable prediction ability. However, high computing costs are required especially when many design variables, materials, or tool shapes are involved. As a result, the use of FE simulation is limited in process design to meet the needs of fast digital manufacturing.

In recent years, data-driven methods have been used as an alternative to FE-based prediction [6,7]. Jansson et al. [1] utilized the response surface method (RSM) based on FE simulations for draw-in optimization. They reported that the restraining force predicted by the RSM differed by up to approximately 15% from experimental results, and that the CPU time for a single FE simulation was about 35 minutes. Bae et al. [8] applied FE simulations into the optimization of draw-bead forces, by comparing with the measured results, the correlation coefficient of 0.9977 shows excellent accuracy.

Neural networks and surrogate models can learn the nonlinear mapping between process parameters and forming outcomes, significantly accelerating design iterations [9,10]. However, most existing models, such as multilayer perceptrons (MLPs) and convolutional neural networks (CNNs), are only designed for grid-structured data. For example, complex spatial relations in deforming sheets cannot be accurately described, or the topological links between different regions are not fully considered by these models. As a result, the physical continuity of the sheet is not well captured and stress-strain transfer between neighboring areas is often lost. For example, Wollschlaeger et al. [11] employed transfer learning with MLPs to predict the draw-in distance in sheet forming. Although

high prediction accuracy of the distance was achieved, the full draw-in shape was not provided since only scalar values were predicted.

Graph neural networks (GNNs) have be well captured and developed to address these limits [12,13]. By representing the sheet as a graph, where nodes correspond to discrete points on the sheet surface and edges describe their mechanical interactions, GNNs can propagate information through the material structure in a physically meaningful way [14]. This enables the model to learn both local deformation patterns and global flow characteristics directly from data, while preserving geometric relationships.

In this work, a GNN-based framework is developed to predict the blank flow shape during stamping based on finite element simulation data. The dataset is built from FE results and each sample is treated as a graph. The global features are given by the blank holder force and the draw-bead forces. These values control the overall material flow. The nodes correspond to the sheet coordinates, representing the spatial positions of material points, while the edges encode the distances between adjacent nodes, describing the local geometric connectivity of the sheet. The mapping from the global process settings to the final node positions is learned by the model, thus the final flow shape can then be predicted without running a full FE simulation. The results show that the use of physical graph features helps the GNN capture the deformation of sheet metal under different forming conditions with good accuracy and high efficiency.

Finite Element Model and Dataset

The material used in this study is AA 6016-T4 sheet with a thickness of 0.9 mm. A Coulomb friction coefficient of 0.1 was adopted. The stamping process was simulated using the explicit solver in LS-DYNA at a forming speed of 1 m/s. A cross-sectional view of the finite element (FE) model and the initial blank geometry is shown in Fig. 1. The forming assembly consists of the upper die, upper die reform, upper pad, post, and binder. The blank holder force (BHF) applied by the binder increased linearly from 0 to a target value (e.g., 20 kN) to clamp the sheet against the upper die, after which it remained constant at the target level. The value of 20 kN is used here only as an example to illustrate the loading procedure, in the dataset generation, the target BHF was varied as a design parameter. Once the target BHF was reached, the upper die advanced to deform the sheet, operating in coordination with the upper die reform, upper pad, and the binder to obtain the final geometry. Draw bead forces (DBFs) were incorporated to regulate material flow during drawing.

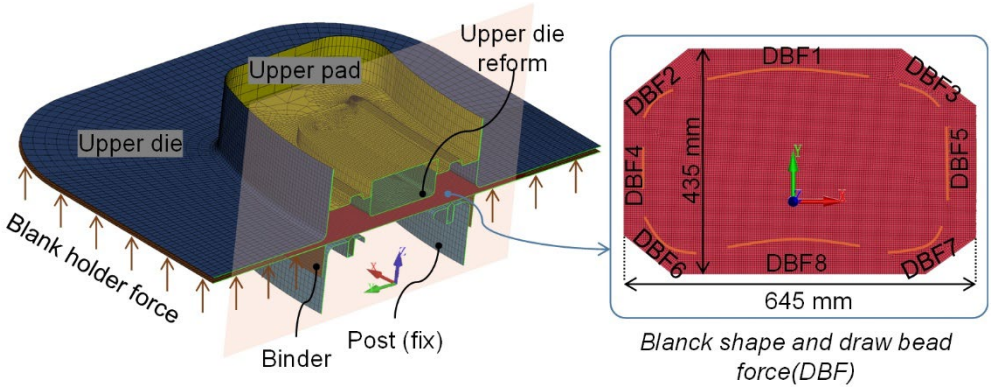

Fig. 1. Cross-sectional finite element representation of the stamping assembly and initial blank geometry.

The constitutive behavior of the sheet was described using the Hockett–Sherby hardening law combined with the Barlat–89 yield criterion [15,16], and parameters are shown in Table 1. The Hockett–Sherby model expresses the evolution of the flow stress as a function of the equivalent plastic strain according to

$$\sigma = \sigma_{\text{sat}} - (\sigma_{\text{sat}} - \sigma_{\text{yield}}) \exp(-c\varepsilon_p^n) \tag{1}$$

where, σ_{sat} is the saturation stress, σ_{yield} is the initial yield stress, ε_p denotes the equivalent plastic strain, c controls the rate of hardening and saturation, and n is strain hardening coefficient. Shell elements were used for all components, and all tooling parts other than the blank were modeled as rigid bodies. Eight draw bead regions were defined around the periphery of the blank, with their positions and shapes illustrated in Fig. 1.

For each simulation case, the nodal coordinates of the blank were extracted and used to construct graph data samples as shown in Fig. 2. The node features are given by the spatial coordinates of each material point, and the edges are defined by the Euclidean distance between neighboring nodes, thereby preserving the geometric connectivity of the blank. The global graph features consist of the blank holder force and eight draw-bead force, which govern the material flow in the forming process. All features were normalized before training to enhance numerical stability.

Table 1. Material parameters of 6016-T4 aluminum alloy

Model	Barlat-89 yield criterion				Hockett-Sherby hardening model			
Parameters	m	R_0	R_{45}	R_{90}	σ_{sat} (MPa)	c	n	σ_{yield} (MPa)
	8	0.61	0.49	0.70	349.40	3.87	0.67	117.40

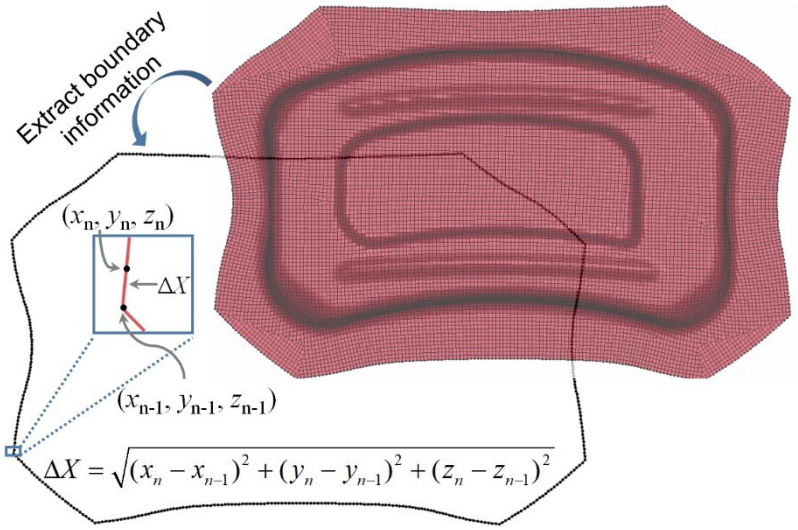

Fig. 2. Schematic of node and edge definition for the deformed sheet.

Methodology

The proposed graph neural network (GNN) is employed to predict sheet-flow deformation by integrating nodal geometry, edge connectivity, and global forming parameters within a unified framework. The overall architecture of the model is illustrated in Fig. 3. The input graph is built from the initial blank shape. Each node represents a material point and is defined by its initial (x, y) coordinates. Each edge stores the initial distance between neighboring nodes. The global forming settings include the blank holder force (BHF) and eight draw bead forces (DBFs). These values are used as graph-level features.

The embedded node and edge features are then passed through four graph attention-based (GAT-based) message passing layers. In each GAT block, multi-head attention is applied. Neighboring information is aggregated with the use of edge features. As a result, deformation behavior controlled by local connections can be learned by the model.

A multi-head attention module is employed to integrate the encoded global parameters with the node embeddings in order to capture the effect of forming loads. This mechanism enables nodes to selectively attend to process parameters, thereby ensuring that forming forces meaningfully modulate the learned representations.

Following message passing and parameter–node attention, the merged node–parameter features are passed to a coordinate prediction head that produces the deformed nodal positions. In parallel, edge features derived from the source and target nodes together with encoded edge attributes are fed into an edge prediction head to estimate relative distances between adjacent nodes. Both outputs are jointly optimized, with the edge-distance prediction serving as an auxiliary geometric constraint. This constraint helps preserve the local shape of the sheet and thereby improves the accuracy of the predicted sheet geometry.

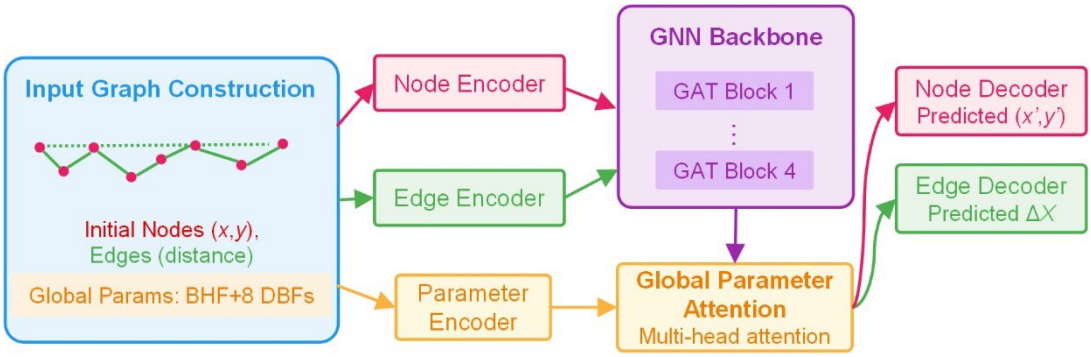

Fig. 3. Architecture of the proposed GNN model.

A composite loss function is employed to train the proposed GNN. The loss jointly supervises node coordinates and edge distances, thereby enabling the model to learn both the global sheet shape and the local geometric connectivity simultaneously. For a graph with N nodes, the coordinate loss is defined as the discrepancy between the predicted node positions $\hat{y}_i$ and the ground-truth FE results y_i:

$$L_{\text{coord}} = \frac{1}{N}\sum_{i=1}^{N}\left\|\hat{y}_i - y_i\right\|^2 \tag{2}$$

The coordinate loss encourages the network to reconstruct the final sheet shape, while local geometric relations are preserved through an edge-based loss applied across all edges. Specifically, the loss penalizes the discrepancy between the predicted edge distance $\hat{e}_{ij}$ and the target FE-computed distance e_{ij}:

$$L_{\text{edge}} = \frac{1}{E}\sum_{(i,j)\in\text{edges}}\left\|\hat{e}_{ij} - e_{ij}\right\|^2 \tag{3}$$

The edge loss helps the network learn consistent deformation patterns and mitigates local distortions.

As coordinate values and edge distances differ in scale, both losses are normalized by their respective standard deviations, σ_c and σ_e. Thereby preventing one term from dominating the overall loss and the normalized loss terms are expressed as:

$$\text{scaled}_{\text{coord}} = L_{\text{coord}}/\sigma_c^2, \text{scaled}_{\text{edge}} = L_{\text{edge}}/\sigma_e^2 \tag{4}$$

The total loss is defined as a weighted sum of the normalized coordinate and edge losses:

$$L_{\text{total}} = \alpha\cdot\text{scaled}_{\text{coord}} + (1-\alpha)\cdot\text{scaled}_{\text{edge}} \tag{5}$$

where the weight factor $\alpha \in [0,1]$ is introduced to balance global shape prediction against local distance prediction. By adjusting weight factor, the contribution of the two loss terms can be optimized.

Table 2. Ranges of BHF (kN) and DBFs (N/mm) used for dataset generation.

	BHF	DBF1	DBF2	DBF3	DBF4	DBF5	DBF6	DBF7	DBF8
Min.	100	58.92	2.63	11.06	37.89	2.63	2.63	1.58	37.89
Max.	300	132.57	26.30	44.22	113.67	26.3	26.30	14.22	113.67

The dataset used to train and evaluate the proposed GNN model was generated entirely through finite element (FE) simulations. For the nine process parameters, including the blank holder force (BHF) and eight draw-bead forces (the ranges are illustrated in Table 2), Latin hypercube sampling was applied. Considering the computing cost and the size of the parameter space, 200 sample points were generated within the allowed ranges of these parameters. The sampled parameter sets were then used as inputs for batch simulations in LS-DYNA3D. The corresponding draw-in shapes were obtained from these simulations. For every FE simulation, the nodal coordinates after deformation were extracted and converted into graph representations consistent with the GNN input format. The 200 graph samples were randomly split into three groups. An 80/10/10 ratio was used for training, validation, and test sets. The training set was used to learn the model parameters. The validation set was employed for hyperparameter tuning, including the number of GAT layers, hidden feature dimension, learning rate, and dropout rate. The test set was kept completely separate and was not used during training or hyperparameter tuning. It was used only to evaluate the final model performance and to analyze the effect of different loss-weighting strategies.

Results and Discussion

The proposed GNN model was trained using the dataset introduced in Methodology section by adopting an early-stopping strategy with a patience of 50 epochs and a checkpointing mechanism to automatically save the best-performing model based on the validation loss. A grid search was conducted to tune the main hyperparameters. After the optimal configuration was identified, the weight factor α—which balances the coordinate loss and the edge-distance loss—was systematically varied. The influence of α on prediction accuracy was evaluated using the test set, while all other hyperparameters were kept fixed. This strategy ensures that the observed performance trends are not biased by parameter tuning and that the contribution of the auxiliary task is assessed in a consistent and objective manner.

The coordinate prediction errors for different values of α are summarized in Fig. 4. When $\alpha = 0$, the model is trained solely on the edge-distance loss and the coordinate mean absolute error (MAE) reaches 105.6 mm, indicating that edge supervision alone is insufficient to guide the deformation prediction. As α increases from 0.1 to 0.6, the coordinate MAE rapidly decreases and reaches its minimum value of 1.4648 mm at $\alpha = 0.3$. This demonstrates that combining coordinate loss with the auxiliary edge-distance loss provides more stable and physically consistent supervision, allowing the model to learn both global shape deformation and local geometric continuity. When α approaches 1.0, the coordinate MAE increases again (2.77 mm), showing that excessive reliance on the coordinate loss weakens the beneficial geometric constraints provided by edge supervision. These observations confirm that the auxiliary task enhances the model's inductive bias and improves robustness, particularly when local deformation patterns play a significant role in sheet-flow evolution.

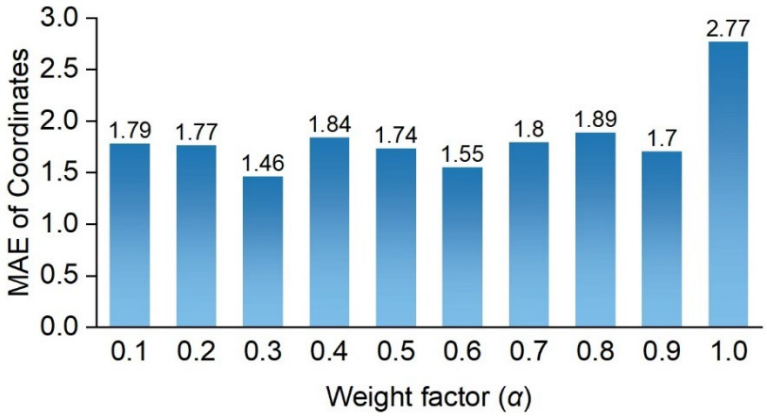

Fig. 4. Influence of weight factors on draw-in shape prediction.

Representative prediction results for α = 0.0, 0.3, and 1.0 are shown in Fig. 5. The predicted nodal coordinates and edge distances are shown in each case. When α = 0.0, non-physical sheet contours are obtained. Although the edge distances remain reasonable, the nodal positions are poorly reconstructed. This indicates that the sheet geometry cannot be recovered from edge constraints alone.

When α = 0.3, the predicted flow shape agrees well with the FE reference. Smooth deformation and uniform spacing between neighboring nodes are observed. These results suggest that the auxiliary edge-distance task provides an effective geometric constraint during message passing. As a result, the spatial continuity of the sheet is better preserved.

When α =1.0, the global shape remains acceptable. However, local distortions and irregular node spacing appear. This behavior reflects the loss of geometric regularization when the edge-distance term is removed from training.

Overall, the results show that the inclusion of the edge-distance task improves both the accuracy and the physical consistency of the GNN model. The best performance is achieved at α = 0.3. This indicates that the use of multi-task learning is important for capturing global deformation behavior and local geometric relations at the same time. The proposed framework therefore provides a useful basis for future graph-based simulation of sheet metal forming.

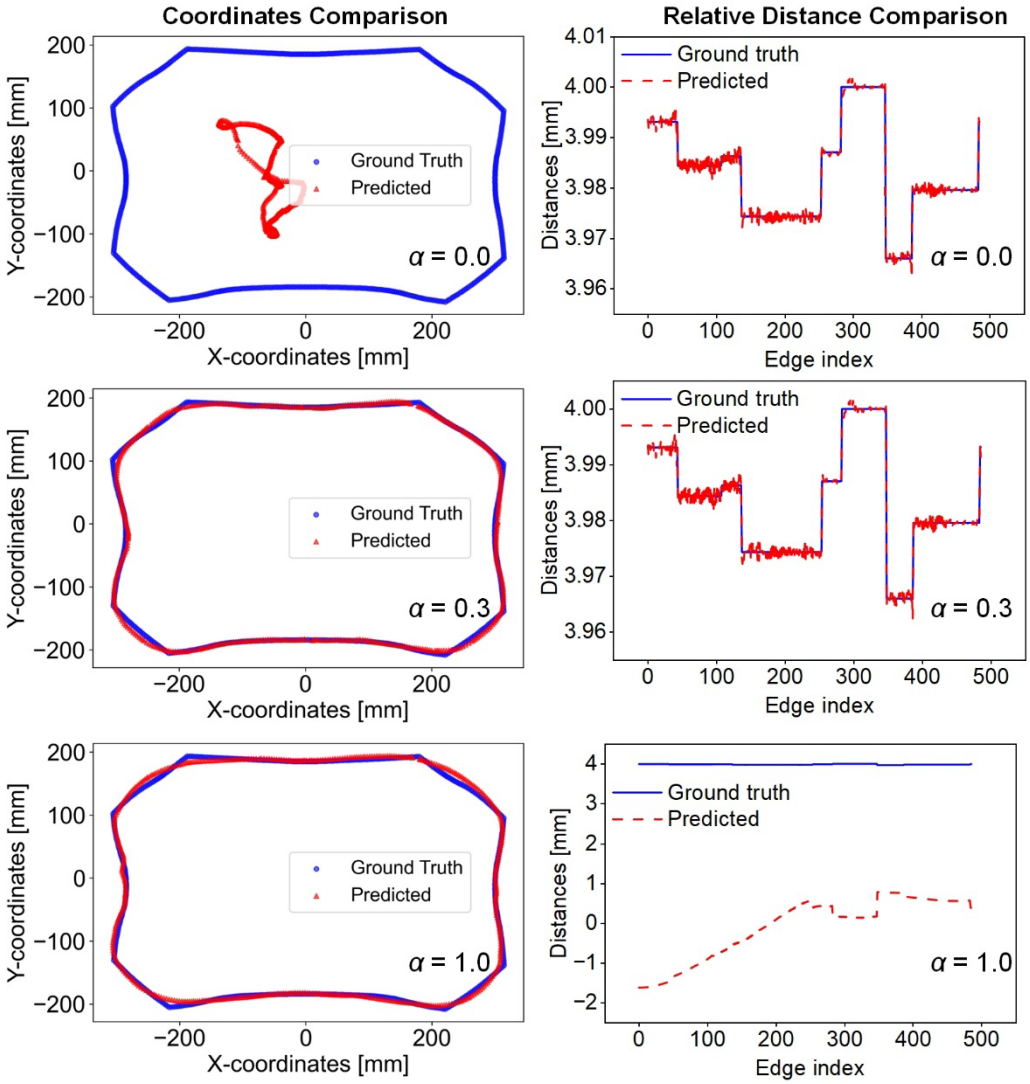

Fig.5 Coordinates and relative distance comparison with the weight factor of 0.0, 0.3, and 1.0.

Conclusions

A graph neural network framework is proposed for predicting sheet flow in stamping. The model includes node geometry, edge links, and global forming forces such as BHF and DBFs. GAT-based message passing is used together with a parameter–node attention module. A multi-task scheme is also applied. Both deformed node positions and edge distances are predicted at the same time.

The results show that the edge-distance task improves the accuracy and geometric quality of the coordinate prediction. A study of the weight factor is performed. The best balance between the two tasks is found at $\alpha = 0.3$. At this value, the lowest coordinate error and the most consistent deformation shape are obtained.

Overall, the proposed GNN provides a fast and flexible option for deformation prediction. It serves as an effective alternative to FE simulation. It also offers a basis for future physics-based graph learning in sheet metal forming.

Acknowledgement

The authors are thankful for USCar Program (Material testing and development of a self-trained physics-based ML model) supported by DoE in USA. Especially, the authors are thankful for Drs. Stanley Wang and Kaiping Li for their technical supports on USCar Program and critical reviews of this manuscript. This research was also supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (RS-2025-25406725) and supported by the InnoCORE program of the Ministry of Science and ICT (N10250154).

References

[1] T. Jansson, A. Andersson, L. Nilsson, Optimization of draw-in for an automotive sheet metal part: An evaluation using surrogate models and response surfaces, J. Mater. Process. Technol. 159 (2005) 426–434. https://doi.org/10.1016/j.jmatprotec.2004.06.011.

[2] S.-S. Han, H.-K. Kim, Optimum multistage deep drawing process design using artificial neural network-based forming quality evaluation function, J. Mater. Process. Technol. 341 (2025) 118881. https://doi.org/10.1016/j.jmatprotec.2025.118881.

[3] M. Joo, M.-S. Wi, S.-Y. Yoon, S.-Y. Lee, F. Barlat, C.N. Tomé, B. Jeon, Y. Jeong, A crystal plasticity finite element analysis on the effect of prestrain on springback, Int. J. Mech. Sci. 237 (2023) 107796. https://doi.org/10.1016/j.ijmecsci.2022.107796.

[4] S. Panich, M. Liewald, V. Uthaisangsuk, Stress and strain based fracture forming limit curves for advanced high strength steel sheet, Int. J. Mater. Form. 11 (2017) 643–661. https://doi.org/10.1007/s12289-017-1378-z.

[5] X. Song, L. Leotoing, D. Guines, E. Ragneau, Investigation of the forming limit strains at fracture of AA5086 sheets using an in-plane biaxial tensile test, Eng. Fract. Mech. 163 (2016) 130–140. https://doi.org/10.1016/j.engfracmech.2016.07.007.

[6] M.R. Jamli, N.M. Farid, The sustainability of neural network applications within finite element analysis in sheet metal forming: A review, Measurement 138 (2019) 446–460. https://doi.org/10.1016/j.measurement.2019.02.034.

[7] D. Tang, S. Qi, K. Zhou, M. Haggag, X. Sun, D. Li, H. Wang, P. Wu, A crystal plasticity-informed data-driven model for magnesium alloys, Int. J. Plast. 194 (2025) 104480. https://doi.org/10.1016/j.ijplas.2025.104480.

[8] G.H. Bae, J.H. Song, H. Huh, S.H. Kim, S.H. Park, Simulation-based prediction model of the draw-bead restraining force and its application to sheet metal forming process, J. Mater. Process. Technol. 187–188 (2007) 123–127. https://doi.org/10.1016/j.jmatprotec.2006.11.059.

[9] M.R. Jamli, A.K. Ariffin, D.A. Wahab, Incorporating feedforward neural network within finite element analysis for L-bending springback prediction, Expert Syst. Appl. 42 (2015) 2604–2614. https://doi.org/10.1016/j.eswa.2014.11.005.

[10] M. Flaschel, P. Steinmann, L. De Lorenzis, E. Kuhl, Convex neural networks learn generalized standard material models, J. Mech. Phys. Solids 200 (2025) 106103. https://doi.org/10.1016/j.jmps.2025.106103.

[11] L. Wollschlaeger, C. Heinzel, S. Thiery, M.Z. El Abdine, N.B. Khalifa, J. Heger, Increased reliability of draw-in prediction in a single stage deep-drawing operation via transfer learning, Procedia CIRP 130 (2024) 270–275. https://doi.org/10.1016/j.procir.2024.10.086.

[12] F. Scarselli, M. Gori, A.C. Tsoi, M. Hagenbuchner, G. Monfardini, The Graph Neural Network Model, IEEE Trans. Neural Netw. 20 (2009) 61–80. https://doi.org/10.1109/TNN.2008.2005605.

[13] S. Duan, D. Kozjek, E. Mehr, M. Anders, J. Cao, Forming force prediction in double-sided incremental forming via GNN-based transfer learning, J. Manuf. Process. 120 (2024) 867–877. https://doi.org/10.1016/j.jmapro.2024.04.093.

[14] Z. Wang, C. Wang, S. Zhang, L. Qiu, Y. Lin, J. Tan, C. Sun, Towards high-accuracy axial springback: Mesh-based simulation of metal tube bending via geometry/process-integrated graph neural networks, Expert Syst. Appl. 255 (2024) 124577. https://doi.org/10.1016/j.eswa.2024.124577.

[15] F. Barlat, K. Lian, Plastic behavior and stretchability of sheet metals. Part I: A yield function for orthotropic sheets under plane stress conditions, Int. J. Plast. 5 (1989) 51–66. https://doi.org/10.1016/0749-6419(89)90019-3.

[16] P.E. Armstrong, J.E. Hockett, O.D. Sherby, Large strain multidirectional deformation of 1100 aluminum at 300 K, J. Mech. Phys. Solids 30 (1982) 37–58. https://doi.org/10.1016/0022-5096(82)90012-6.