Sheet Metal Forming in a Virtual Reality Environment using LS-DYNA and Neural Networks

Author:

Ashwini S. Gokhale
Industrial and Manufacturing Engineering Department
Wichita State University
1845 Fairmount, Wichita, Kansas, 67260
E-mail: ashwini_gokhale@hotmail.com

Keywords:
Sheet Metal Forming, Neural Networks, Virtual Reality, Parametric Analysis, Manufacturing Process Design and Optimization, Process Sensitivity Analysis, VRForm, VRwave

ABSTRACT

Manufacturing process simulation using finite element techniques has immensely contributed to ensuring the success of concurrent design methodologies. However, Finite Element Methods (FEM) is computationally expensive and consequently unsuitable for design and manufacturing optimization in a production environment. In this research, a coupled Artificial Intelligence (AI) and FEM technique was developed to simulate and predict process response to changes in part design. Generic process models of part families are developed using Artificial Neural Networks (ANNs) and FEM. The generic models are used to predict the response of the manufacturing process to variations in geometric, material and process parameters, in real time. The predicted results are graphically displayed in a Virtual Reality environment. Standalone software *VRF*orm© was developed based on this methodology. *VRForm*© can be used to optimize component, tool and process designs.

INTRODUCTION

Customer driven economy of recent times has greatly reduced product lifecycles, and increased the stress on manufacturers to tune their product development times to respond to rapid market changes. Shorter product life cycles also imply lower product development costs. Cost effective risk free virtual prototyping using finite element methods for manufacturing process simulation has played an invaluable role in the ability of automobile and aircrafts industries to respond to this impetus. However, finite element methods involve rigorous computation and hence, are time intensive. Consequently, market demands restrict the scope of application of this technology as an optimization tool in analysis of designs for manufacturability, and performance of manufacturing processes. This builds a strong case for the development of a more responsive technology.

The compelling need for a responsive industrial structure has forced industries into agile alliances with global partners. The ability to share information and resources with the global partners in real-time is key to the success of these geographically distributed enterprises. Most finite element softwares are packaged with 3D CAD systems to provide data visualization capabilities. Current CAD systems generate visually realistic 3D graphics, and make it possible to visualize the physically unseeable. For example, the progressive deformation of a metal sheet, into final part geometry, between a pair of matched dies, in a draw-forming operation. However, most CAD systems do not provide distributed multi-user capability, the CAD models are bulky for real time transmission over public networks amongst geographically separated users, and the use of proprietary data structures by different CAD systems leads to reliability concerns of shared CAD data. Hence, there is an urgent need for the identification of a multi-user technology with a standard framework, for visualization of manufacturing process analyses.

In this research *VRF*orm© a standalone software was developed. *VRF*orm© integrates artificial neural networks with virtual reality to enable real time prediction and visualization of process sensitivity and performance. Many researchers have employed FEM for manufacturing process simulation, and combined it with neural networks as an efficient control mechanism for monitoring the processes. However, in these studies, FEM has been used as a technique to acquire large amount of data in a quick and inexpensive manner, and neural networks have been used to control the physical processes [Ruffin, et. al., 1998]. Inamadar, et. al. [2000] have developed a neural network to predict springback in air V-bending. Little research has been done in developing manufacturing process simulations in a virtual reality environment. Nicholson, et. al. [2000], have converted FEM simulations into VRML 1.0 to present results of ballistic impact on vehicles. Ryken, et. al. [2000], analyzed a tractor lift arm in MSC/ NASTRAN and viewed the results in IGRIP. However, the analysis time was very large and results were not available in real time. The use of neural networks to at least partly take over the role of FEM as an optimization tool and the use of virtual reality as a visualization environment, for analyses: of designs for manufacturability, and performance of manufacturing processes, is the key contribution of this research. An overview of relevant technologies is provided in the following section.

INTEGRATING TECHNOLOGIES – Artificial Neural Networks

Neural networks are an artificial intelligence technique, conceptually similar to the construction and functioning of the human brain. Artificial neural networks have proven to be a great tool for solving a wide variety of problems, because of their ability to approximate non-linear functions in the absence of closed form solutions [Fausett, 1994]. Neural networks are a parallel processing multi-processor system. Simple Scalar messages and adaptive interaction is used to link the multiple processors. In general, neural networks are modeled by training and testing. Neural network modeling is a data driven process and hence, the quality of the network depends on the quantity and quality of available data. A variety of neural network architectures and training algorithms have been developed. The choice of architecture and training algorithm is essentially application specific. However, the most popular network architecture is the fully connected feed forward network, trained using the back-propagation algorithm.

A neural network consists of many simple processors called neurons. Structurally each neuron consists of an input, an output and an activation function. The activation function controls the contribution of the input to the output. The network output is the weighted sum of outputs generated by the individual neurons. The neurons are arranged in layers. In general, for a neural network to be able to process a non-linear data set it must have at least three layers, namely; the input layer, hidden layer, and the output layer. The input layer consists of as many neurons as the number of independent variables in the data set, and the output layer as the number of dependent variables. The hidden layer consists of one or more neurons, and a network of one or more hidden layers. The neurons in a network may be fully connected, i.e., all neurons in one layer are connected to all of the neurons in the next, or are selectively connected by the analyst. In neural network terminology the structure and layout of the network is termed as *topology*. Feed forward networks are trained by passing data from the input to the output end of the network. Convergence of the network is ensured by the back propagation algorithm. The *error* between the predicted output and the actual value of the output is used to adjust the connection weights. The network is said to have converged when the error is less than a preset value, for a batch of data not included in the training data set.

INTEGRATING TECHNOLOGIES – Virtual Reality

In recent years computer-rendered 3D virtual environments, popularly known as 'Virtual Reality', which respond to user actions in real-time, are gaining momentum. A virtual reality environment is essentially a multi-user environment, in which users can interact with and manipulate objects in the virtual world, and interact with and modify other user's actions in the virtual world [McCarthy et. al, 1998]. Interactive and natural navigational modes like walk and fly, stereoscopic vision, and user interaction with objects and users in the virtual environment through tactile and auditory feedback, have made it feasible to explore the unexplorable. From a simulation of manufacturing processes perspective: virtual reality not only facilitates visualization of the unseeable but also the ability to touch, feel and listen.

Virtual reality systems can be divided into two major groups fully Immersive and Non-immersive. Immersive systems, replace the user’s view of the real world with computer-generated images that react to the position and orientation of the user's head. A non-immersive system, on the other hand, leaves the user visually aware of the real world but still be able to observe the virtual world through some display device such as a graphics workstation. Interaction with the virtual world is accomplished through a variety of feedback devices of which the most common are head-mounted displays and gloves.

Virtual Reality Modeling Language (VRML) is widely used as a scene description language for developing virtual reality scenes. The design of VRML started in the First World Wide Web (WWW) conference in May 1994 [Pesce, et. al., 1994] and was based on a subset of Open Inventor [McCarthy et. al, 1998]. VRML 2.0 is based on VRML 97 specification. VRML 97 includes specification for interaction, avatar behavior and multimedia extension. VRML 2.0 uses a scene graph to define the structure of the virtual world. The scene graph is a node tree, with each node representing a feature of the virtual world. Optimization of the scene graph is key to developing responsive worlds, as the rendering engine runs off the scene graph. VRML 2.0 provides a convenient method to build 3D geometric worlds using primitives, and freeform surface geometry can be constructed using finite regular surfaces. VRML 2.0 facilitates texturing, audio and video support, animation, morphing, and event triggers in the form of sensors. VRML 2.0 is compatible only with Java Programming language and Java script, and Java is the only programming language that supports VRML 2.0. VRML 2.0 can be embedded in JAVA and vice versa, to facilitate external control of events in the virtual world or control of the external world through events in the virtual world. Two different mechanisms script nodes and External Authoring Interface (EAI) can be used to interface JAVA with VRML 2.0 [Baerten et. al, 1998]. Use of script nodes facilitates use of a java class, which can be called when the node is initialized. EAI provides a flexible approach to link VRML with java. Java applets can be used to send events to the VRML nodes to change node attributes directly. Virtual reality typically implies an immersive Three Dimensional (3D) environment. However, the use of VRML does not ensure an immersive environment. It depends on the capabilities of the browser and supporting hardware. The interpretation, execution and presentation of VRML files is typically done by a browser. Performance of a VRML world depends upon the choice of browser. Rendering speed and image quality are the comparative features of browsers [Nadeau, 1997]. Some browsers also support use of separate hardware for navigation, which enhance the feeling of immersiveness [Broll et al., 1996]. Some of the popular browsers are Blaxxun 3D, Cortona, Cosmo player 2.1, Microsoft VRML 2.0 viewer, VRwave, and VRweb.

VRwave is a non-immersive virtual reality browser developed by the Institute for Information Processing and Computer Supported New Media (IICM) [Wagenbrunn, 1998]. VRwave source code is available at the IICM web page, and is free for non-commercial applications. VRwave is largely authored in java, and it uses a customized version of Opengl for 3D graphics and rendering. The customized version is compatible only with Java 1.2.x. It is recommended that JDK/JRE 1.2.2 be used to execute VRwave. Further, VRwave does not support stereoscopic viewing. *VRForm*© uses an embedded version of VRwave consequently; these technical limitations apply to *VRForm*© also.

METHODOLOGY

VRForm© evolved in response to the question *'how do we convert the data generated by manufacturing process simulations using FEM into accessible design and manufacturing knowledge that can be readily used by design and manufacturing engineers?'* *VRForm*© was conceived on the premise that in an industrial environment, there is always a need to modify existing designs in successive generations of the product, and consequently there exists a need to evaluate the potential impact of design changes on the manufacturability of the part. Similarly, at the manufacturing end there exists a need to evaluate potential consequences of changing process parameters on the manufacturability of components. As an example in the event of break down of a sheet metal forming press, would a similar press with a lower pressure head form components of competitive quality? *VRForm*© was developed to address this class of problems and to present a real time solution strategy that can be used by all personnel with minimal training.

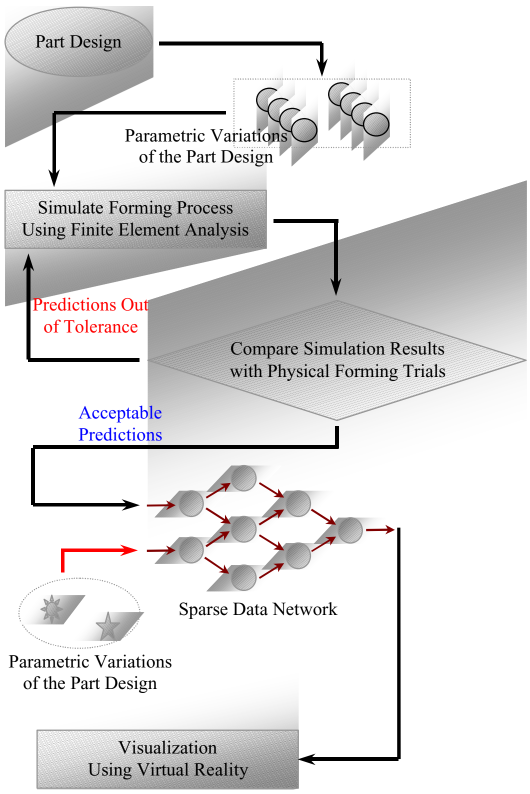


Figure 1 Development of Generic Neural Network Models for Part Families and Visualization of Neural Network based Simulations in a Virtual Reality Environment

VRForm© couples finite elements with artificial neural networks to capture and analyze patterns in the data generated by finite element simulations, and employs a virtual reality environment to present the results of the simulations. The process flow chart for *VRForm*© is shown in Figure 1. Parametric manufacturability analysis of a family of parts is conducted using finite elements. Manufacturability response of the part family with respect to various parameters like: part geometry, material, manufacturing process, friction conditions, etc are simulated. *VRForm*© parses the results of the simulations and trains embedded artificial neural networks, consequently developing a non-linear mapping between the various parameters influencing formability, of the part family. The trained networks are employed to make *real time predictions and generate manufacturing process simulations for parametric variations of the design*. Thus, part and process specific results of finite element simulations are used to develop generalized design criteria based on manufacturability. The algorithm used in the development of *VRForm*© is shown in Table 1.

Step 1.	FEM simulation of the manufacturing process
a.	Define input parameters (*factors*) – geometric, material and process parameters
b.	Define the bounding box for the parameters
c.	Define output parameters of interest – von Mises stress, plastic strain, displacement, sheet thickness, etc.
d.	Design a set of FEM simulations to determine the effect of variation of the *factors* – determine appropriate number of *levels* for the experiment
Step 2.	Select a suitable neural network algorithm to model the effect of input parameters on the output parameters
Step 3.	Model networks for each output parameter
Step 4.	Simulate the manufacturing process for input parameter values not in the designed set of experiments using the neural networks
Step 5.	Visualize the neural network simulated manufacturing process in the virtual environment

Table 1 Algorithm used to Develop Neural Network Models using *VRForm*©

CASE STUDY

Aircraft and automobile industries depend largely on sheet metal forming processes for manufacturing components. Variants of channel type geometry, shown in Figure 2, are one of the most commonly encountered part shapes, and hence, was used to demonstrate the neural network based simulations developed in this research. Aluminum 2024 alloy (ISO AlCu4Mg1) in T3 temper was used as the material for forming the channels. The forming of the parts was simulated using the Hydroforming process, at a pressure of 4,000 psi.

Springback, wrinkling, and excessive thinning or tearing of the sheet during forming are the most common reasons for rejection of formed components. Of these, reliable prediction of springback using FEM, especially for the 2024 aluminum T3 temper alloy, is extremely difficult. This is mainly due to the fact that the 2xxx series aluminum alloys are strain rate sensitive. In this case study, material properties obtained from uniaxial tension tests performed under static conditions according to ASTM-E8, shown in Table 2 [Kalpakjian, et.al., 2001], have been used. Even though prediction of springback was not the focus of this research, every effort was made to ensure that the FEM predicted springback values exhibited similar trends as expected by theory. However, not much emphasis was laid on the numerical correctness of the FEM predicted springback. Hence, a similar behavior must be expected even from the neural network models.

R	Outside Bend Radius	r	Inside Bend Radius
α	Inside Bend Angle	β	Outside Bend Angle
t	Sheet Thickness	Size	20 x 1 Sq in flat sheet, Depth 1 in

Constraints

Inside Bend Radius is equal to Outside Bend Radius for Forming
Inside Bend Angle is equal to Outside Bend Angle for Forming

Figure 2 Geometry of Channel Simulated using FEM and Neural Networks

2024 Aluminum T3 Temper Alloy		
Young's Modulus [E]	(psi)	9.97E+06
Poisson's Ratio [ν]	(psi)	0.35
Density [ρ]	(lb/in^3)	1.010e-01
Strength Coefficient [K]	(MPa)	690
Hardening Exponent [n]		0.16

Table 2 Static Material Properties for 2024 Aluminum T3 Temper Alloy

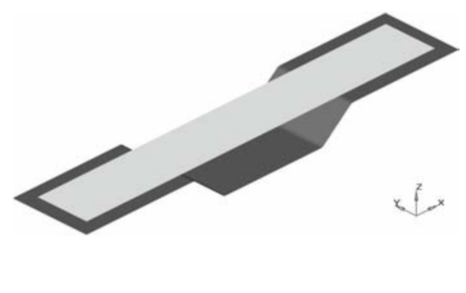
IMPLEMENTATION – Finite Element Modeling

Finite element model for hydroforming of the channel shaped component was developed using HyperMesh 3D, shown in Figure 3, and LS-Dyna 950d solver was used to analyze the manufacturing processes. The simulation parameters are shown in Table 3. The Coefficient of friction, maximum forming pressure and forming depth were maintained a constant over all simulations. The FEM simulation results were analyzed using HyperView.

Geometric design parameters or factors for the part family were selected based on their influence on springback. The levels for the finite element simulations were based on most commonly used industrial values, shown in Table 4. Finite element simulations were run on HP-J5600 dual processor machines with 2.0 GB of RAM, using HP-UX 11.0. The run time for each simulation was approximately 3.0 CPU hrs.

Simulation Matrix						
Factors		Simulation Levels				
Bend Angle [α, β]	(degree)	30	45	60	90	
Bend Radius [R, r]	(inch)	0.125	0.1875	0.25		
Sheet Thickness [t]	(inch)	0.032	0.040	0.050	0.063	0.080

Table 4 Design factors and Levels for FEM Experiments



FE Model Specifications	
Parameter	Value
Material	
Sheet (deformable)	Al2024-T3
Female Die (rigid)	Tool Steel
Number of Nodes	
Sheet	2121
Female Die	2000
Friction Coefficient	0.15
Element Type	3d Quad Shells

Figure 3 Finite Element Model of Channel Forming, showing Sheet and Female Die

Table 3 Finite Element Simulation Parameters

IMPLEMENTATION – *VRForm*©

VRForm© provides a common integrated virtual environment for both analysis and visualization. It was developed using Java 1.2.x, in order to facilitate compatibility with VRML 2.0, and has a modular architecture. The three main modules are the browser, the FEM to VRML translator, and the Artificial Neural Network module.

The choice of Java as the programming language sets constraints on the availability of VR browsers. Most popular browsers cannot be embedded in a Java application, due to various incompatibilities. VRwave developed by IICM is authored in Java 1.2.x and is available as free ware source code. Consequently, VRwave was adopted as the VR browser in spite of several deficiencies. VRwave was stripped of its stand-alone features and integrated into the *VRForm*© environment, to provide browsing capability.

VRForm© provides support for direct translation of LS-Dyna ASCII output to VRML 2.0. One of the key features of the direct translator is automatic scene graph optimization. The scene graph optimizer builds the VR world based on the LS-Dyna key files used to develop the finite element model, and the ASCII output of the LS-Dyna analysis. In building the VR world, the translator maintains a one to one correspondence with the finite element model. The various parts in the finite element model are identified by their part numbers in the analysis. Appropriate parent child relationships are established and the VR world is constructed in layers. The VR models are constructed using finite regular surfaces. Unlike most CAD softwares, *VRForm*© does not use an arbitrary surface tessellation for building the VR world, but instead uses the mesh used for the finite element analysis. This ensures bi-directional compatibility with the finite element model for mapping of results, and leads to a more compact and human readable model than ones generated by most CAD softwares. Further, it provides support for: multiple deformable bodies, the use of texture maps, color, lighting, and animation. Rigid body masking and dynamic animation control are some of the more important features that were added to VRwave in order to facilitate convenient viewing of the manufacturing simulation.

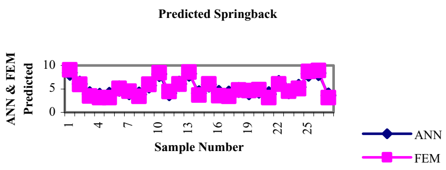
A concern in the development of the artificial neural network module was the vast amount of data needed to develop reliable networks, and consequently, the number of finite element simulations needed to generate the data. The obvious solution was to use a training strategy that would enable development of reliable networks even under conditions of sparse data [Twomey, 1998]. It is generally accepted that the data is sparse if the number of available data sets is less than five times the number of independent variables in the data set. The two main problems encountered in training networks under conditions of sparse data are non-convergence of the network due to inadequate data, and over training, both of which lead to erroneous models. It has been demonstrated that 0.632e stop-training algorithm effectively controls over training under conditions of sparse data and ensures the best fit [Mardana, 2001]. The artificial neural network module was designed to use the back propagation algorithm for network training and the 0.632e stop-training algorithm to control training. This strategy ensured reduced number of finite element simulations while ensuring reliable network models.

Developing artificial neural network models to simulate a manufacturing process involves training the network to track the movement of material overtime in the three spatial directions as a function of the input parameters. As an example modeling a network to simulate forming of a sheet metal component would mean predicting the movement of the sheet material and values of the output parameters, over the area of the sheet, over the duration of the forming process. However, at a given instant in time the forming process exhibits localized differences over the area of the sheet, depending on the geometry of the component being formed. Consequently, the problem is highly non-linear. *VRFo*rm© employs separate networks for each output parameter in order to reduce the complexity of the problem. *VRFo*rm© provides a convenient auto-training function, a semi-automatic training module that employs a 'teach and train' algorithm, which, largely reduces the frustrations involved in determining a suitable network topology. *VRFo*rm© supports training of multiple networks in parallel, thus reducing overall training time. Support is provided for modeling simple feed-forward and feed-forward recurrent back-propagation networks. *VRFo*rm© uses an encoded proprietary format to save the network models hence; networks developed in *VRFo*rm© are not compatible with other commercial neural network applications.

Current implementation of *VRFo*rm© provides capabilities for the analysis and simulation of von Mises stress, plastic strain, displacements, and nodal coordinates for prediction of geometric changes to the deformable body due to the manufacturing process.

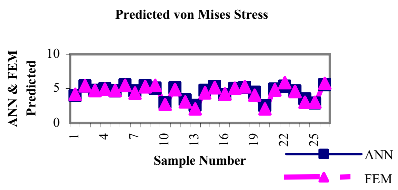
RESULTS AND DISCUSSION

Springback being a one step phenomenon was modeled using simple feed-forward networks. Independent networks were modeled to predict springback at each of the bends in the channel. Figure 4a shows the performance of a springback network in testing. Topologies of the networks are not presented because of proprietary issues.



Avg. Error ≈ 5.4%, Max. Error ≈ 13.2%

a. Plot of Predicted Springback for the Angle between the Base and Slant Flange



Avg. Error ≈ 4.19%, Max. Error ≈ 13.8%

b. Plot of Predicted von Mises Stress

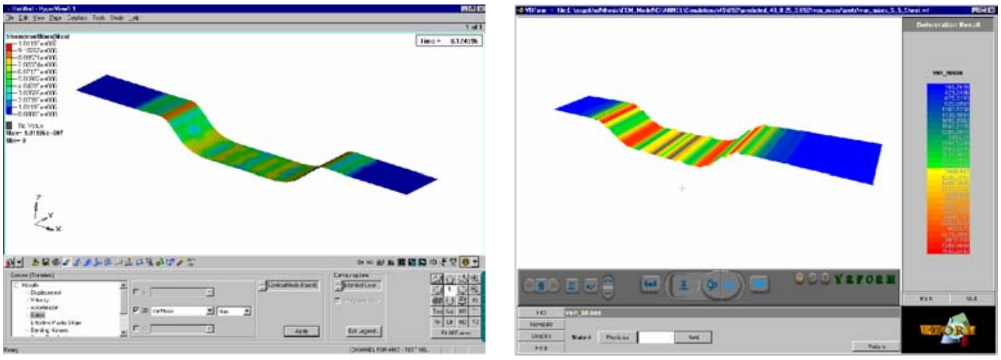
Figure 4 Neural Network Test Results – Comparison of ANN & FEM Predictions
Note: Numerical Values Indicated along the Y-Axis are *VRForm*©'s internal scaled Representation

The network models were used to predict springback in one, two and three factor experiments, shown in Table 5, to evaluate their performance against FEM simulations. The network predictions demonstrated acceptable correspondence with FEM simulations.

Predicted Springback						
Test Type	Geometric Parameters			Predicted Springback		
	Angle (degree)	Radius (inch)	Sheet Thickness (inch)	FEM (degree)	ANN (degree)	Deviation (percent)
Single Factor	40	0.125	0.050	42.28	43.34	2.51
	40	0.1875	0.050	41.30	41.19	0.27
	40	0.25	0.050	40.41	39.33	2.67
Two Factor	50	0.125	0.045	52.42	52.51	0.17
	50	0.1562	0.063	52.42	49.59	5.40
	65	0.1562	0.032	67.21	66.85	0.54
Three Factor	65	0.2187	0.0715	66.53	61.02	8.28

Table 5 Comparison of Springback Predicted using FEM and Neural Networks

Prediction of the current state of a material with respect to stress, strain and displacement of the material involves knowledge of the materials previous state of stress, strain and position. Feed-forward recurrent back-propagation networks were modeled to predict von Mises stress, plastic strain, displacements, and nodal coordinates for the channel forming process. The use of feed-forward recurrent back-propagation networks to model the stress, strain and displacement states enables analysis of pre-stressed and pre-strained materials, as in a multi-stage forming process. The channel forming process was analyzed using 15 samples in time. The sampling frequency was non-linear and based on the rate of deformation of the sheet. Figure 4b shows the performance of a von Mises stress network in testing. Topology of the network is not presented because of proprietary issues. Figures 5a & b show plots of von Mises stress distribution in the channel as obtained from the finite element simulation and using neural networks respectively. The network predicted von Mises stresses were within 20% of the finite element solutions. Simulation time using neural networks was in the range of 30 - 60 seconds on a Pentium IV 2.0 GHz processor with 512 MB of RAM, running Windows XP. Comparable results were obtained for plastic strain and displacements using neural networks. However, prediction of coordinates using neural networks resulted in a jagged or non-smooth surface, shown in Figure 5b, in the initial and final stages of the process. It is hypothesized that the networks were unable to track the process in the unsteady state due to large deformations, while they performed acceptably in the steady state zones of the process. Possible solutions to over come this drawback are to increase the sampling frequency during the unsteady state of the process, thus rendering incremental deformation data for training, and using a best-fit technique to smooth the predicted deformed surface.



a. FEM Predicted Plot of von Mises Stress

b. ANN Predicted Plot of von Mises Stress

Figure 5 von Mises Stress - Bend Angle 40°, Bend Rad. 0.25” & Sheet Thk. 0.050”

CONCLUSION AND FUTURE WORK

VRForm©’s real time analysis capability makes parametric analysis of manufacturability a realizable idea, thus rendering a better exploration of the design space from both engineering and manufacturing perspectives. *VRForm*© makes on the fly risk free analysis of manufacturing process response to variations in geometric, material and process parameters feasible. Further, *VRForm*©’s capability to process sparse data makes it possible to develop a reliable knowledgebase based on a limited number of finite element simulations. The current implementation of *VRForm*© supports only LS-Dyna 3D, and 3D shell element models. Support for LS-Dyna binary output format and adaptive meshes are desirable features. *VRForm*©‘s integrated virtual reality environment facilitates multi-user interaction. However, the current implementation is limited to a single active user and multiple-passive users. The passive users will be able to view the VR worlds over the Internet using any VR browser, while only the active user can run neural network based simulations using *VRForm*©. *VRForm*©‘s scene graph optimization capability renders real time communication amongst geographically distributed users across the Internet feasible.

ACKNOWLEDGEMENT

The author gratefully acknowledges and thanks her friends Mr. Shashikiran Prabhakar, Mr. Srinivasu Mardana and Ms. Kavitha Gorantla and for their valuable assistance during the course of the research.

COPYRIGHTS AND PATENTS

REFERENCES

Ames, A. L., Nadeau, D.R., and Moreland, J.L., VRML 2.0 Sourcebook, John Wiley & Sons, Inc., (1997).

Baerten, H., and Reeth, F.V., “Using VRML and JAVA to Visualize 3D Algorithms in Computer Graphics Education”, Computer Networks and ISDN Systems, Vol. 30, pp. 1833-1839, (1998).

Broll, W., and Koop, T., “VRML: Today and Tomorrow, Computer and Graphics”, Vol. 20, No. 3, pp. 427-434, (1996).

Fausett, L., Fundamentals of Neural Networks Architectures, Algorithms, and Applications, Prentice-Hall, (1994).

Hallquist, J.O., Ls-Dyna Manuals, Livermore Software Tech. Co., (2001).

Inamadar, M.V., Date, P.P., and Desai, U.B., “Studies on the Prediction of Springback in Air Vee Bending of Metallic Sheets using an Artificial Neural Network”, Journal of Materials Processing Technology, Vol. 108, pp. 45-54, (2000).

Kalpakjian, S., Schmid, S.R., and Schmidt, S., Manufacturing Engineering and Technology, Prentice-Hall, Ed. 4, pp. 2-9, (2001)

Twomey, J., and Smith, A.E., “Bias and Variance of Validation Methods for Function Approximation Neural Networks Under Conditions of Sparse Data”, IEEE Transactions on Systems, Man, and Cybernetics, Vol. 28 (3), pp. 417-430 (1998).

Mardana, S., 0.632E Stop-Training Method for Neural Networks under the Conditions of Sparse Data, Master’s Thesis, Wichita State University, (2001).

Martin McCarthy, and Alligator Descartes, Building 3D worlds with Java and VRML, Prentice-Hall Europe, (1998).

Nadeau, D.R., “How to Optimize the Performance of VRML Worlds, part II," VRML Technique column, NetscapeWorld, August-September, (1997).

Nicholson, D.W., Moraes, R. F., Divo, E., and Cahill, B., “Virtual Reality Visualization (VRV) of Realistic Weapons Effects Predicted Using Ls-Dyna”, 6th International LS-DYNA Users Conference, April 9-11, Dearborn, MI, Session 15, pp. 15.47-15.58, (2000).

Pesce, M.D., Kennard, P., and Parisi, A.S., “Cyberspace”, First International Conference on the World Wide Web, May 25-27, CERN, Geneva, Switzerland, (1994).

Ruffin, R., and Cao, J., “Using Neural Network for Springback Minimization in a Channel Forming Process”, SAE Special Publications Developments in Sheet Metal Stamping Proceedings of the 1998 SAE International Congress & Exposition, Feb 23-26, Detroit, MI, Vol. 1322, pp. 77-85, (1998).

Ryken, M. J., and Vance, J.M., “Applying Virtual Reality Techniques to the Interactive Stress Analysis of a Tractor Lift Arm”, Finite Elements in Analysis and Design, Vol. 30, pp. 141-155, (2000).

Wagenbrunn, K.H., Rendering and External Authoring Functionality for VRwave, Master’s Thesis, Institute for Information Processing and Computer Supported New Media (IICM), Graz University of Technology, Graz, Austria, (1998).