
Implementierung einer Netzwerkschnittstelle in LS-DYNA zur gekoppelten Simulation

S. Kriechenbauer¹

¹Fraunhofer IWU, Nöthnitzer Str.44, 01187 Dresden

1 Einleitung

Zur Simulation von Blechumformprozessen werden im industriellen Einsatz einfache Modelle verwendet. Damit werden Machbarkeitsstudien durchgeführt und Blechzuschnitte ausgelegt. Mit der Simulation werden Tiefziehwerkzeuge entwickelt, die nach der Fertigung eingearbeitet werden müssen. Da in den vereinfachten Modellen nicht alle qualitätsbeeinflussenden Faktoren berücksichtigt werden, entstehen Defizite bei der Entwicklung der Werkzeuge [1]. Beispielsweise werden die Durchbiegung der Werkzeuge und die Auffederung der Pressmaschine nicht abgebildet. Diese Nachgiebigkeiten führen beim Tiefziehen zu einer Druckverteilung am Niederhalter, die den Werkstofffluss bestimmt. In der Simulation werden nur starre Wirkflächen abgebildet, womit abweichende Flächenpressungen im Flansch prognostiziert werden. Bei hohem Druck wird das Blech am Hineinfließen in die Umformzone gehindert. Bei der Einarbeitung werden die Kräfte durch manuelles Ab- oder Auftragen der Niederhalterfläche korrigiert. Dieser Prozess muss mehrmals wiederholt werden bis das Ergebnis stimmt, da nur Erfahrungen und Handarbeit zum Erfolg führen.

Mit modernen Servo-Pressen können die Bewegungen von Stößel und Ziehkissen frei programmiert werden. Diese Flexibilität kann mit den starren Randbedingungen in den einfachen Prozessmodellen nicht abgebildet werden. Bei der Modellierung werden keine realen Massenverhältnisse abgebildet, um die Rechenzeit zu beschleunigen. Die dynamischen Verhältnisse werden somit falsch beschrieben. Neuere Entwicklungen von Prozessen auf modernen Servo-Pressen, die z.B. eine deutliche Abhängigkeit von der Bewegung der Werkzeugteile zeigen, können mit den einfachen Modellen nicht abgebildet werden.

In vorhergehenden Arbeiten wurden bereits Ansätze entwickelt, um die Prozessmodelle um charakteristische Maschineneigenschaften zu erweitern. Das Spektrum der Erweiterungen reicht in der Simulation mit finiten Elementen von vereinfachten Ersatzmodellen wie z.B. Feder-Dämpfer-Elementen bis hin zur Modellierung gesamter Pressen [2]. Dadurch erhöht sich die Komplexität der Modelle deutlich. Es wird schwieriger, die Modelle aufzubauen und zu parametrieren. Häufig dauert der Modellfindungsprozess länger. Die Berechnung der Umformvorgänge verlängert sich bei den erweiterten Modellen um ein Vielfaches. Die Interaktion von Prozess und Maschine lässt sich auch durch die Kopplung zweier unterschiedlicher Simulationsumgebungen abbilden [3]. Dabei werden die Vorteile der zwei Umgebungen wie z.B. FEM (Finite-Elemente-Methode) und DBS (Digitale Blocksimulation) verbunden und Nachteile der einzelnen Programme beseitigt. Mit der zuletzt genannten Simulationsumgebung kann das dynamische mechatronische Systemverhalten erfasst werden. Durch die Kopplung erweitert sich auch der Funktionsumfang. Die Prognosegenauigkeit nimmt bei gekoppelten Simulationen zu, da Prozessgrößen wie z.B. die Kräfte während der Umformung in beiden Umgebungen aktualisiert und ausgetauscht werden. Beispielsweise können durch diesen Austausch asymmetrische Effekte aufgrund der Stößelkipfung wie einseitige Faltenbildung oder lokal stärkerer Blecheinzug erklärt werden [1].

2 Softwarekonzept zur gekoppelten Simulation

In der vorliegenden Veröffentlichung wird beispielhaft eine Kopplung zwischen Matlab-Simulink (DBS) und LS-DYNA (FEM) vorgestellt. Die entwickelte Software kann auch mit anderen Simulationsumgebungen gekoppelt werden. Zunächst soll das Konzept der Kopplung näher erläutert werden. Der Aufbau der gekoppelten Simulation ist beispielhaft für die zwei Clients in Fig.1: dargestellt. Die Kommunikation mit dem Server kann um beliebig weitere Simulationsumgebungen erweitert werden. Die Clients arbeiten unabhängig voneinander in verschiedenen Threads.

Der Datenaustausch ist bei gekoppelten Simulationen nicht standardisiert. Daher muss jeweils eine Schnittstelle geschaffen werden, die den Ablauf der gekoppelten Simulation kontrolliert. Sinnvollerweise wird für den Datenaustausch auf das Netzwerkprotokoll TCP/IP (Transmission Control Protocol/Internet Protocol) zurückgegriffen. Dieses Konzept ermöglicht maximale Flexibilität bei der Wahl der Programme und Standorte der Simulation. So lassen sich über das Internet Daten zwischen mehreren Simulationsumgebungen austauschen. LS-DYNA und MATLAB melden sich als Clients am

Server an, der den Datenaustausch zentralisiert und überwacht. In der jetzigen Form wurde der Server in C/C++ programmiert und als Konsolenprogramm gestartet. Eine Benutzeroberfläche soll in weiteren Arbeitsschritten folgen. Die Funktionalität ist so verpackt, dass die Programmerroutinen als Bibliothek exportiert und in andere Programmiersprachen leicht eingebunden werden können. Der Umfang ist auf wenige Funktionen für das Senden und Empfangen von Daten oder Anweisungen beschränkt. Daher können leicht Schnittstellen zu anderen Programmiersprachen wie z.B. Python oder Fortran realisiert werden. Falls eine Simulationsumgebung keine offenen Programmierschnittstellen bietet, kann ein externes Programm die Austauschgrößen erfassen und an den Server weiterleiten. Die Steuerung der Simulation wird dadurch allerdings erschwert.

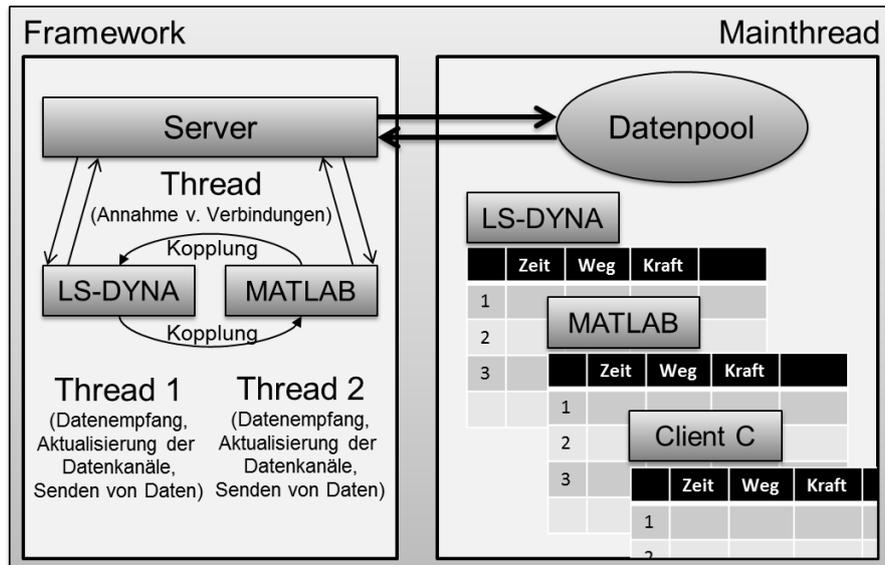


Fig.1: Softwarekonzept zur gekoppelten Simulation basierend auf dem Netzwerkprotokoll TCP/IP am Beispiel zwischen LS-DYNA und Matlab Simulink

In LS-DYNA wurde das Netzwerkprotokoll in die User-Subroutine uctrl1 in der Programmierschnittstelle dyn21.F implementiert. Vorteil ist dabei, dass LS-DYNA in jedem Berechnungsschritt mit dem Server kommuniziert und während der Simulation auch auf Freigaben des Servers wartet. In der expliziten Version von LS-DYNA werden die Austauschgrößen für den nächsten Zeitschritt direkt in die Vorgaben der Randbedingungen geschrieben. Diese Vorgehensweise funktioniert für kinematische Größen und Kräfte. Zur Aktivierung und Steuerung der Schnittstelle wird eine separate Datei beim Start des Solvers eingelesen. Damit wird sichergestellt, dass bestehende Umformsimulationen nicht beeinflusst werden. In der Datei wird die IP-Adresse des Servers hinterlegt, die Zeitschrittweite für das Empfangen und Senden von Daten und die ID's für ***DEFINE_CURVE** bzw. ***DEFINE_CURVE_FUNCTION**. Der Datenaustausch ist asynchron, d.h. die Zeitschrittweite für das Senden kann sich von der für das Empfangen unterscheiden. Die Daten aus der Empfangsrichtung werden direkt in die Arrays für die entsprechenden Kurven geschrieben bzw. in der Senderichtung ausgelesen. Bei jedem Austauschschritt zum Datenempfang wartet der Client auf Antwort vom Server, um die Simulation mit den aktuellen Daten eines anderen Clients fortsetzen zu können. Durch diese Abfolge wird gewährleistet, dass alle Simulationsumgebungen mit aktuellen Datensätzen weiterrechnen. Die berechneten Austauschdaten werden im Server in Datenkanälen zwischengespeichert. Fordert nun ein Simulationsprogramm zu einem bestimmten Zeitschritt Daten an, werden diese interpoliert und weitergegeben. Die Zeitschritte für den Datenaustausch dürfen nicht zu weit auseinander liegen, um die Stabilität der gekoppelten Simulation zu gewährleisten.

In LS-DYNA steht mit dieser Implementierung in Fortran eine voll funktionsfähige Netzwerkschnittstelle zum Datenaustausch bereit. In Matlab Simulink wurde die Funktionalität mit einer S-Function implementiert. Die S-Function ist eine C/C++ Programmierschnittstelle. In Fig.2: ist die Schnittstelle in einem Matlab-Simulink-Modell zusammen mit einem Beispielprozess dargestellt. Die aktuellen Daten von LS-DYNA kommen an der Client-Schnittstelle an. Im Beispiel wird die Kraft als Austauschgröße weitergeleitet an das Matlab-Simulink-Modell. Im Beispiel wird ein Kraftsprung simuliert. Die Reaktion des Systems gemessen an den kinematischen Größen wird wieder über die Client-Schnittstelle und den Server weitergeleitet an LS-DYNA oder an andere Simulationsumgebungen. Ähnlich wie beim Tiefziehen wird die Prozesskraft aus dem Umformmodell der FEM-

Umgebung ausgelesen und im Modell der Pressmaschine in der DBS-Umgebung vorgegeben. Die resultierende Verlagerung wird zurück an das Umformmodell gegeben. Diese Art der Kopplung basiert auf dem Austausch von Kraft- und kinematischen Größen wie Weg, Geschwindigkeit oder Beschleunigung.

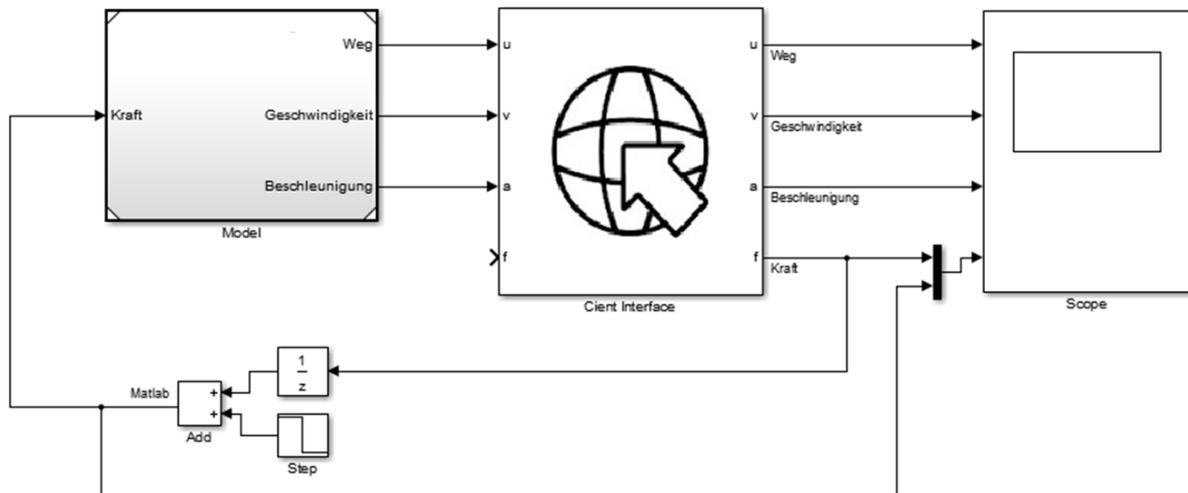


Fig.2: Darstellung des Matlab-Simulink-Modells mit Beispielprozess in der FEM (links) der Client-Schnittstelle (mitte) und der Auswertung (rechts)

3 Zusammenfassung

Mit einer gekoppelten Simulation kann die Wechselwirkung zwischen einem Umformprozess und der Pressmaschine berechnet werden. Der Werkstofffluss ist beim Tiefziehen abhängig von der Druckverteilung am Niederhalter, die wiederum von der Verformung des Werkzeuges und der Auffederung der Pressmaschine bestimmt wird. Bei Berücksichtigung der Effekte in einer gekoppelten Simulation können Tiefziehwerkzeuge entwickelt werden, die eines geringeren Einarbeitungsaufwands bedürfen.

Zur Kopplung mehrerer Programme entstand eine Software, die zum Zweck des Datenaustausches auf dem Netzwerkprotokoll TCP/IP basiert. Damit können verschiedenste Simulationsprogramme gekoppelt werden. Am Beispiel von LS-DYNA und Matlab Simulink erfolgte eine direkte Implementierung in die Programme. Das Netzwerkprotokoll kann dabei durch eine Eingabedatei aktiviert und gesteuert werden.

4 Literatur

- [1] Hardtmann, A.: " Entwicklung und Bewertung eines erweiterten Blechumformprozessmodells unter besonderer Berücksichtigung der elastostatischen Wechselwirkungen zwischen Maschine und Prozess", Dissertation TU Dresden, 2010
- [2] Geimer, M.; Krüger, T.; Linsel, P.: „Co-Simulation, gekoppelte Simulation oder Simulatorkopplung“, O+P 11-12, 2006, S. 572-576
- [3] Weber, J.; Schulze, T.; Großmann, K.; Penter, L.; Schenke, C.-C.: Simulationsgestützte Abstimmung von Ziehkissen, EFB-Forschungsbericht Nr. 412, Hannover, 2015