

Ein CASE-Werkzeug zur hybriden Modellierung und Simulation intelligenter Maschinen

István Péter¹, Alexander Pretschner², Thomas Stauner²

¹Lehrstuhl für Informationstechnik im Maschinenwesen

Technische Universität München, Boltzmannstr. 15, 85748 Garching

² Institut für Informatik,

Technische Universität München, Arcisstr. 21, D-80290 München

Tel.: ¹ +49 (0)89 289-16427, ² +49 (0)89 289-28325

Fax: +49 (0)89 289-16410

E-Mail: {peter@itm, pretschn@in, stauner@in}.tum.de

Abstract: Im Rahmen der Toolpräsentation wird das auf einer hybriden Erweiterung der UML-RT basierende CASE-Werkzeug MaSiEd (**M**aschinen-**S**imulator/**E**ditor) vorgestellt. MaSiEd wurde für die Modellierung, Simulation und Analyse des E/A-Verhaltens moderner feldbusbasierter Fertigungsmaschinen mit gemischt diskret-kontinuierlichem Charakter entwickelt und anhand zahlreicher industrieller Fallstudien bewertet. Graphische Beschreibungstechniken für sowohl diskrete als auch kontinuierliche Systeme ermöglichen die integrierte Modellierung und Simulation hybrider Systeme.

Stichwörter: virtuelle Maschine, Modellierung und Simulation hybrider Systeme.

1 Einleitung

Rasante Fortschritte in Informations- und Mikrosystemtechnik und die durch zunehmenden Funktionsumfang bedingte Komplexitätssteigerung führen zu immer längeren Entwicklungszeiten für die Steuerungssoftware moderner Fertigungsmaschinen. Für die Unternehmen bedeutet dieser Trend eine dramatische Entwicklung, denn Test- und Optimierungsphase der Software sind erst nach Abschluß der Mechanik-Entwicklung in Kopplung mit der Maschine möglich und liegen auf dem kritischen Projektpfad. Simultaneous Engineering, d.h. Parallelisierung der Entwicklungsabläufe, verkürzt die wertvolle "Time-to-Market". Das CASE-Werkzeug MaSiEd (Maschinen-Simulator/Editor) ist ein PC-Simulationssystem, mit dessen Hilfe das Maschinenverhalten in Echtzeit mit gewünschter Präzision simuliert werden kann. Dadurch läßt sich der Steuerungstest in eine frühere Projektphase vorverlagern, während die Mechanikentwicklung parallel stattfindet. Die Steuerung wird an diesen Simulator, eine sog. virtuelle Maschine, schnittstellenkompatibel angeschlossen, so daß sich für die Steuerungssoftware der Unterschied zur realen Maschine nicht bemerkbar macht. Die moderne Feldbustechnik vereinfacht diese Kopplung drastisch.

2 MaSiEd (Maschinen-Simulator/Editor)

Der besondere Ansatz von MaSiEd liegt in einer komponentenbasierten Vorgehensweise, die die komfortable Erstellung eines Simulationsmodells der intelligenten Maschine ermöglicht. Die Nachbildung des gesamten Maschinenverhaltens erfolgt durch Komposition einzelner modellierter Maschinenkomponenten aus einer Bibliothek (Transportband, Greifer, Zylinder, Aktoren, Sensoren usw.). Entscheidend ist, daß bei typischer Variantenbildung einer Maschine nun die neue Simulation aus der vorhandenen einfach durch Hinzufügen, Entfernen oder Modifizieren von Einzelkomponenten abgeleitet werden kann.

Das Verhalten solch komplexer technischer Systeme aus Mechanik, Mikroelektronik und Mikrosystemkomponenten mit minimalen Aufwand so genau wie nötig zu modellieren, um sie in Echtzeit zu simulieren, ist eine große Herausforderung. Dieser Aufgabe haben sich Industrie und Hochschulen im Verbundprojekt SEMI [Bend97] gestellt. Das Ergebnis der gemeinsamen Forschung und Entwicklung führte zu drei wesentlichen Ergebnissen: ein Vorgehensmodell, eine Methode und ein CASE-Werkzeug (Abb. 1). Das aus dem SEMI-Projekt hervorgegangene und im DFG-Schwerpunktprogramm KONDISK weiterentwickelte CASE-Tool MaSiEd wurde um Beschreibungstechniken für kontinuierliche Aktivitäten [Peter00] erweitert. Ähnlich den hybriden Automaten [Alur93] können kontinuierliche Vorgänge durch Simulink-Blockdiagramme oder Übertragungsfunktionen mit Zuständen assoziiert werden. Im Unterschied zu hybriden Automaten gewährleistet die Abstützung auf ROOM bzw. UML-RT allerdings eine modulare Entwicklung [Grosu98]. Darüberhinaus wurde ein Editor für hybride Sequenzdiagramme [Grosu00, UML00] für die Anforderungsspezifikation in MaSiEd integriert.

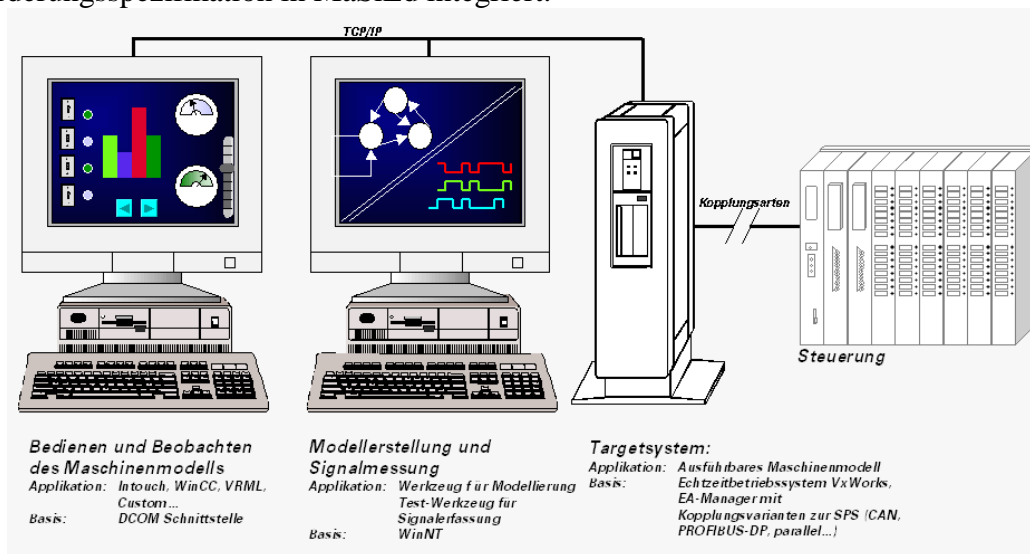


Abbildung 1: Architektur des Simulationsystems

In bezug auf die Modellierungsmöglichkeiten läßt sich auf den ersten Blick eine gewisse Ähnlichkeit zwischen MaSiEd und der in der Industrie populären Werkzeugkopplung Matlab/Simulink/Stateflow ausmachen. Im Gegensatz zu Matlab/Stateflow erlaubt HyROOM eine bessere Wiederverwendung von Komponenten durch das abstrakte Schnittstellenkonzept mit Protokollverwendung.

Ein weiterer entscheidender Unterschied besteht in der Granularität der Integration kontinuierlicher Beschreibungen. MaSiEd gestattet die Definition kontinuierlicher Vorgänge auf Zustandsebene. In Matlab hingegen können kontinuierliche Aktivitäten nur auf der Ebene von Komponenten

stattfinden – eine Einschränkung, die u.a. für die Probleme mit der Anwendung eines konventionellen Entwicklungsprozesses auf den Entwurf hybrider Systeme verantwortlich ist [Pretsch00]. Das führt gezwungenermaßen zum Kopieren von Zustandsstrukturen: Zustandsübergänge werden in Matlab nicht nur in den diskreten Stateflow-Komponenten spezifiziert, sondern zusätzlich in den assoziierten Blockdiagrammen gespiegelt. Dieses geschieht innerhalb der Blockdiagramme durch die Einführung von Switches oder durch die Einführung diskreter Variablen. Eine Änderung im Blockdiagramm oder in der Stateflow-Komponente muß von Hand immer auch in die jeweils verknüpfte Komponente übertragen werden: Ein zusätzlicher Zustand in Stateflow führt zu einem weiteren Switch in den Blockdiagrammen und umgekehrt. In MaSiEd unterbleibt eine solche fehleranfällige Verpflichtung. MaSiEd erlaubt außerdem die Definition sog. Datenflußports für die direkte Verbindung kontinuierlicher und diskreter Komponenten. Ausgehend von existierenden, auf Mikroprozessorsteuerungen, Mikrosystem- und Feldbustechnik basierenden Maschinen wurden exemplarisch für diese realen Maschinen Simulationsmodelle entwickelt. Dabei wurden wiederkehrende Arbeiten identifiziert und systematisiert. Das Ergebnis ist ein *Vorgehensmodell*, das es erlaubt, virtuelle Maschinen auf Basis der hybriden Erweiterung der UML-RT zu beschreiben und daraus automatisch ablauffähigen Code für die Simulation für die Betriebssysteme VxWorks, WindowsNT und Linux zu generieren.

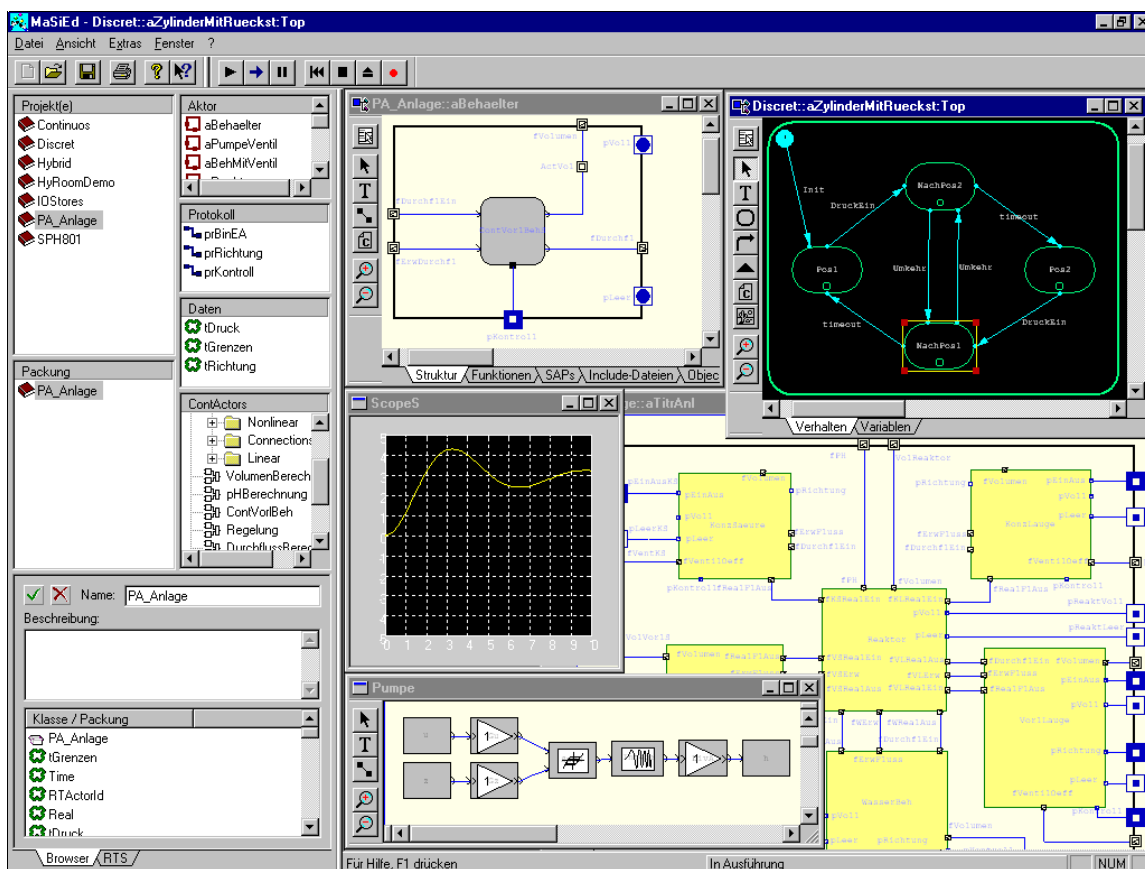


Abbildung 2: Screenshot des Simulationsystems

Anhand der erzeugten Maschinenmodelle konnte nachgewiesen werden, daß die *Methode* der komponentenbasierten Modellierung eine in der Praxis sehr gut anwendbare Vorgehensweise zur aufwandsarmen Erstellung von Maschinenmodellen (Abb. 2) darstellt. Die bei der Maschinenanalyse identifizierten Mechanik- und Mikrosystemkomponenten wurden in einer Modellbibliothek hinterlegt, so daß bei der Neuerstellung eines Maschinenmodells auf vorhandene

Elemente nach dem Baukastenprinzip zurückgegriffen werden kann, um ein Maximum an Wiederverwendbarkeit zu ermöglichen. Die Modellbibliothek kann von den Anwendern individuell stetig erweitert und auf die firmenspezifischen Belange zugeschnitten werden, so daß sich die Neuerstellung oder Modifizierung eines Maschinenmodells zunehmend vereinfacht.

Das Simulationssystem basiert auf *Standard*- PC-Hardware und Betriebssystem. Dadurch kann automatisch die Rechenleistungs- und Kostenentwicklung der PC-Welt ausgenutzt werden und Zukunftssicherheit garantiert werden. Die Möglichkeit zur Multi-Prozessor-Kopplung in modernen PCI-Systemen stellt darüberhinaus eine skalierbare Rechenleistung zur Verfügung, um auch wachsenden Anforderungen gerecht zu werden. Das PC-Simulationssystem wurde im Rahmen des Projekts bis zum Prototypenstadium entwickelt. Auf dieser Basis konnten mehrere lauffähige Maschinenmodelle entworfen werden (z.B. [Prinz98, Link99]), deren Funktionsfähigkeit durch den Test gegen bereits vorhandene funktionsfähige Steuerungen erfolgreich nachgewiesen werden konnte.

Literatur

- [Alur93] Alur, R.; Courcoubetis, C.; Henzinger, T.; Ho, P.: Hybrid automata: an algorithmic approach to the specification and verification of hybrid systems. Proc. Hybrid Systems I, Springer LNCS 736, 1993
- [Bend97] K. Bender, J. Tomaszunas. Erfahrungsbericht zur qualitativen Modellierung mechanischer Systeme zum Steuerungstest in Echtzeit. Entwurf komplexer Automatisierungssysteme 5. Fachtagung, TU Braunschweig, Institut für Regelungs- und Automatisierungstechnik, pp. 398-417 , 1997
- [Grosu98] Grosu, R.; Stauner, T.; Broy: A modular visual model for hybrid systems. In Proc. Formal Techniques in Real-Time and Fault-Tolerant Systems (FTRTFT 1998), Springer LNCS 1486, 1998.
- [Grosu00] Grosu, R.; Krüger, I.; Stauner, T.: Hybrid Sequence Charts. Proc. 3rd IEEE International Symposium on Object-oriented Real-time distributed Computing (ISORC 2000), IEEE, 2000
- [Link99] Link, R.: Systematische Maschinenmodellierung: Modellierung des Die-Bonders DB200 (Fa. ASEM) nach der ROOM-Methode, Diplomarbeit, TU-München, 1999
- [Peter00] Péter, I.; Pretschner, A.; Stauner, T.: Heterogeneous Development of Hybrid Systems. Proc. GI workshop Rigorose Entwicklung software-intensiver Systeme, Berlin, August 2000
- [Pretsch00] Pretschner, A.; Slotosch, O.; Stauner, T.: Developing Correct Safety Critical, Hybrid, Embedded Systems. Proc. New Information Processing Techniques for Military Systems, NATO Research and Technology Organization, Istanbul, October 2000
- [Prinz98] Prinz, O.: Systematische Maschinenmodellierung: Modellierung einer Transfer-Honmaschine (Fa. Nagel) nach der ROOM-Methode, Diplomarbeit, TU-München, 1998
- [UML00] Rational Software Corporation: Unified Modeling Language Documentation Resources v1.3G, 2000