

PARALLELE SIMULATIONSTECHNIK

L.P.Feldmann, V.A.Svjatnyj, V.V. Lapko

Fakultät für Rechentechnik und Informatik (FRTI)

Technische Universität Donezk

Artemstr. 58, 83000 Donezk, Ukraine

e-mail: {feldmann, lapko, svjatnyj} @cs.dgtu.donetsk.ua

E.-D. Gilles, M. Zeitz

Institut für Systemdynamik und Regelungstechnik (ISR)

Universität Stuttgart

Postfach 801140, 70511 Stuttgart

e-mail: {gilles, zeitz} @isr.uni-stuttgart.de

A.Reuter

International University in Germany Bruchsal (IU)

76646 Bruchsal, Germany

e-mail: Andreas.Reuter@i-u.de

K.Rothermel

Institut für parallele und verteilte Höchstleistungsrechner (IPVR)

Universität Stuttgart

Breitwiesenstr. 20 – 22, 70565 Stuttgart

e-mail: Kurt.Rothermel@informatik.uni-stuttgart.de

Abstract

Feldmann L.P., Svjatnyj V.A., Lapko V.V., Gilles E.-D., Zeitz M., Reuter A., Rothermel K. Parallel Simulation Technique. Nowadays simulation application areas have essentially grown and the necessity to use parallel computer resources occurs in the simulation procedure of physical and engineering system models of real complexity. The article considers problems of effective design and use of parallel simulation models as a contribution to the new research area of „Parallel Simulation Technique“. Moreover, the concept of a parallel simulation environment is presented and implemented by means of parallel computer systems of SIMD and of MIMD architecture.

1. Einführung

Die Anwendungen der modernen Simulationstechnik reichen von den "traditionellen" physikalischen und technischen dynamischen Systemen (DS) mit konzentrierten und mit verteilten Parametern (DSKP und DSVP) über biotechnologische und ökologische Prozesse bis zur Projektierung und Optimierung von Unternehmens-, Wirtschafts- und Gesellschaftsstrukturen. Die modellgestützte Entwicklung von komplexen technischen und nichttechnischen DS in allen Anwendungsbereichen ist ein Schlüsselfaktor für die Qualitätssicherung und Konkurrenzfähigkeit von zukünftigen technischen und technologischen Problemlösungen. Als Folge der steigenden Anforderungen an Qualität, Sicherheit und Wirtschaftlichkeit der DS-Prozesse werden immer genauere und damit komplexere Simulationsmodelle benötigt. Bei dieser Entwicklung ergibt sich in einer natürlichen Weise eine Komplexitätsschwelle für die Implementierung und Anwendung der Simulationsmodelle, die nur durch den Einsatz von neuen Simulationstechniken überwunden werden kann. Zu dieser Technik gehören u.a. die in

diesem Artikel betrachteten parallelen Höchstleistungsrechner der SIMD- und MIMD-Architektur [1].

Die effiziente Nutzung von parallelen Rechentechniken für die Lösung von Simulationen erfordert eine benutzerfreundliche Simulationssoftware mit neuen Ansätzen zur Modellerstellung und für die Durchführung von Simulationsexperimenten. Nach den Erfahrungen der Autoren zählen die dabei zu lösenden theoretischen und praktischen simulationstechnischen Fragestellungen zum Gebiet der parallelen Simulationstechnik. Auf diesem Gebiet besteht schon seit längerer Zeit eine enge Kooperation zwischen Simulationstechnikern und Informatikern aus Donezk (FRTI), Stuttgart (ISR, IPVR) und Bruchsal (IU) [2,3,4].

2. Anforderungen an die parallele Simulationstechnik

Die parallele Simulationstechnik lässt sich in einen methodischen, einen implementierungstechnischen und einen anwendungsorientierten Bereich untergliedern. Die parallelen Simulationsmethoden hängen eng mit der Modellierung der betrachteten Prozesse zusammen. Im Hinblick auf eine parallele Simulation ist es sehr wichtig, bereits frühzeitig eine Strukturierung und Modularisierung der Prozessmodelle vorzunehmen. Dabei kann man sich meist an der inhärenten Parallelstruktur der betrachteten Prozesse orientieren. Deshalb sind Simulationsmodelle komplexer Prozesse in einer natürlicher Weise modular strukturiert und daher auch für eine parallele Simulation geeignet. Dabei ist zu unterscheiden, ob die Simulationsmodelle in einer rein gleichungsorientierten oder in einer gemischt-block-/gleichungsorientierten Form vorliegen. Deshalb kommen der Topologieanalyse und der Gleichungsgenerierung für komplexe DSKP und DSVP eine grosse Bedeutung zu und sind als besonders interessante Fragestellungen der parallelen Simulationstechnik einzustufen.

Zum methodischen Bereich der parallelen Simulation zählen auch noch die numerischen Verfahren zur Approximation bzw. Rechnerlösung von dynamischen Simulationsmodellen. Diese Modelle umfassen neben algebraischen Gleichungen vor allem zeitabhängige gewöhnliche und partielle Differentialgleichungen sowie Integrodifferentialgleichungen. Die bisher verfügbaren numerischen Verfahren wurden in erster Linie für eine sequentielle Anwendung entwickelt und eignen sich deshalb nur bedingt für eine parallele Implementierung. Aus diesem Grund kommt der Untersuchung und Entwicklung von parallelen Lösungsalgorithmen für DSKP und DSVP eine besondere Bedeutung zu.

Für die Anwendung der parallelen numerischen Lösungsalgorithmen spielen auch die Implementierungstechniken eine wichtige Rolle. Hierfür werden benutzerfreundliche parallele Simulationssprachen benötigt. Diese müssen einerseits die bewährten Eigenschaften der bekannten block- und gleichungsorientierten Sprachen besitzen und andererseits dem Benutzer die Mittel und Werkzeuge zur Topologieanalyse, Gleichungsgenerierung, Parallelnumerik und Ergebnisvisualisierung zur Verfügung zu stellen. In diesem Artikel werden die Anforderungen an eine parallele Simulationssprache im Hinblick auf die optimale Nutzung von SIMD- und MIMD-Parallelrechnerstrukturen betrachtet.

3. Konzeption einer parallelen Simulationsumgebung

Ein wesentliches Ziel der von den Autoren getragenen Kooperation auf dem Gebiet der Simulationstechnik umfasst die Entwicklung einer parallelen Simulationsumgebung für dynamische Systeme mit konzentrierten (DSKP) und mit verteilten (DSVP) Parametern. Mit dieser Simulationsumgebung soll untersucht werden, inwieweit die bereits angesprochenen Anforderungen an eine parallele Simulationstechnik mit den verfügbaren Hard- und Software-

Ressourcen realisiert werden können. Weitere Fragestellungen dieser Untersuchung befassen sich mit der Effizienz und Akzeptanz von einem solchen Werkzeug. Daraus ergeben sich die folgenden Anforderungen und Bedingungen für die Konzeption einer parallelen Simulationsumgebung:

- Hochentwickelte Benutzeroberfläche (BOF) für die rechnergestützte Modellerstellung und das Modell-Debugging für DSKP und DSVP
- Simulationssprache mit einer domänenabhängigen Modellspezifikation und leichter Erlernbarkeit
- Objektorientierte Modellspezifikation für die Struktur, das Verhalten und die Parameter von DSKP und DSVP
- Numerische Lösungsverfahren für die parallele Simulation von grossen und komplexen DSKP und DSVP
- Visualisierung und Dokumentation der Simulationsmodelle und der Simulationsergebnisse
- Echtzeitfähigkeit zur Kopplung von Simulationsmodellen mit realen DS-Komponenten und Automatisierungskomponenten (Hardware- und Software-in-the-Loop-Simulation)
- Schnittstellen für Werkzeuge zur rechnergestützten Projektierung von DSKP und DSVP
- Gleichartige Benutzung der Simulationswerkzeuge für die Bearbeitung von DS-Problemen auf der Simulationsebene und auf der Projektierungsebene
- Realisierung von anwendungsspezifischen Simulationsumgebungen

Zur Untersuchung von komplexen DSKP- und DSVP-Simulationsaufgaben wird eine massiv-parallele Simulationsumgebung (MPSU) vorgeschlagen, entwickelt und implementiert. Nach Abb.1 umfasst die MPSU eine benutzerfreundliche Organisation der Hardware (parallele Rechnerressourcen, Arbeitsplätze, Peripheriegeräte, Kommunikationsnetze), der Systemsoftware (Betriebssysteme, parallele Programmiersprachen, Netzsoftware, Peripheriesoftware, BOF-Systemteil) und der Simulationssoftware (parallele Simulationssprachen, Numerik-Bibliotheken, BOF-Simulationsteil, Hilfsprogramme). Auf diese Weise sollen alle Etappen der Modellierung und Simulation unter Berücksichtigung der obigen Anforderungen unterstützt werden.

Nach Abb.2 wird die MPSU mit parallelen Hard- und Softwareressourcen und mit Arbeitsplatzrechnern (APR) in der Architektur eines verteilten Rechnersystems realisiert. Diese Architektur erfordert die Entwicklung einer Simulationssoftware für die SIMD- und MIMD-Komponenten entsprechend den vorgegebenen DS-Anforderungen. Dabei werden für DSKP und für ortsdiskretisierte DSVP dieselben numerischen Lösungsalgorithmen verwendet. Ausserdem besitzt die entwickelte parallele Simulationssoftware ähnliche Eigenschaften wie die block- und gleichungsorientierten Simulationssprachen. Die Entwicklung der MPSU bildet eine zentrale Aufgabenstellung für die parallelen Simulationstechnik und wird von den beteiligten Instituten unter verschiedenen Gesichtspunkten verfolgt. Darüber hinaus umfasst die MPSU Module, die auch für die Entwicklung anderer Softwarewerkzeuge verwendet werden können.

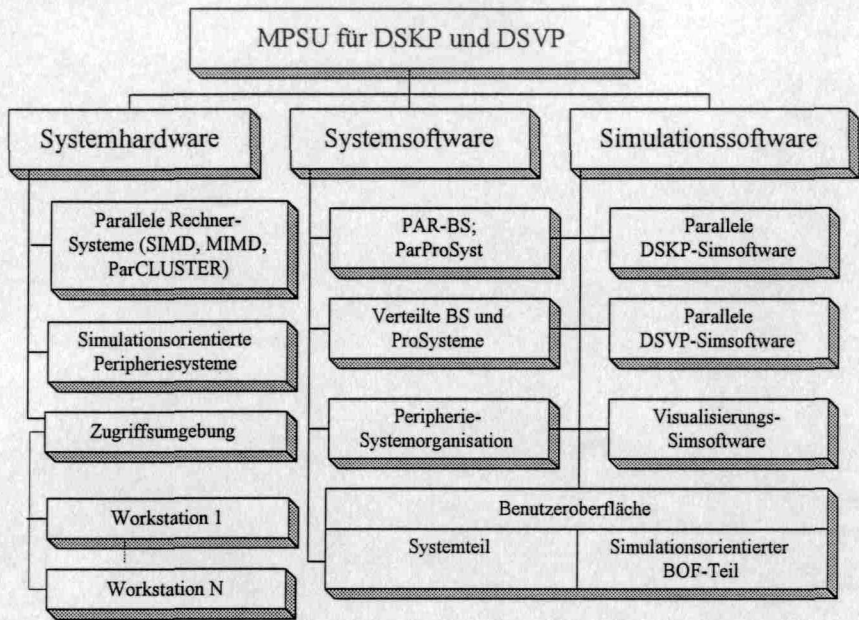


Abb.1 Architektur der massiv-parallelen Simulationsumgebung (MPSU) für dynamische Systeme mit konzentrierten (DSKP) und mit verteilten (DSVP) Parametern

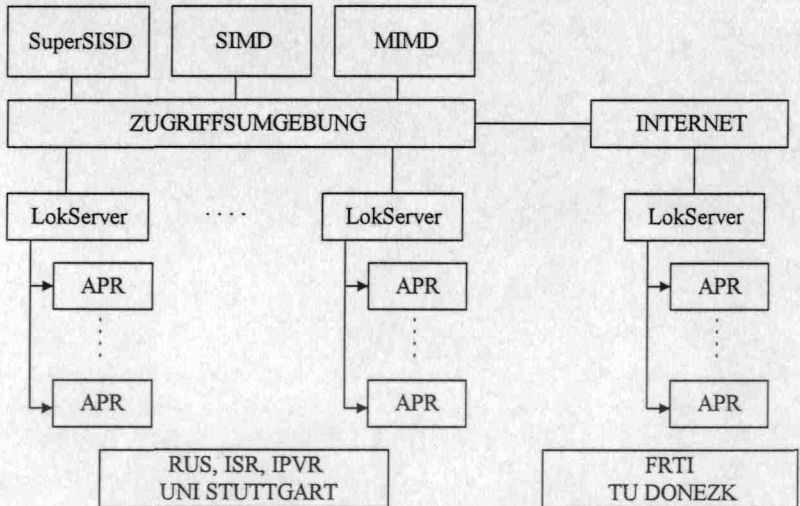


Abb.2 Realisierung der MPSU als verteiltes Rechnersystem mit parallelen Ressourcen an der Universität Stuttgart und der Technischen Universität Donetsk

4. Realisierung der SIMD-Komponente der MPSU

In [2] wird die Realisierung der MPSU mit einer SIMD-Hardware vorgestellt und für die Simulation von DSKP- und DSVP untersucht. Diese MPSU funktioniert in zwei verschiedenen Versionen. Am IPVR in Stuttgart wurde die MPSU mit der Hardware "MasPar – IPVR-Netz – Arbeitsplatzrechner" aufgebaut und an der FRTI in Donetsk mit "Arbeitsplatzrechner – InterNet – IPVR-Netz – MasPar" realisiert. Zur Zeit wird eine MPSU-Implementierung mit der folgenden Konfiguration "FRTI-APR – InterNet – IPVR-Netz-PARALLAXIS-Debugger" und "FRTI-APR – LAB-Netz – PARALLAXIS-Debugger" verwendet.

Die SIMD-parallelen Simulationsalgorithmen für DSVP wurden für parabolische, elliptische und hyperbolische partielle Differentialgleichungen und spezielle DSVP entwickelt, implementiert und erprobt [3,4]. Für DSKP wurden die in konventionellen Simulations-sprachen vorhandenen Algorithmen parallelisiert und für die benötigte Topologieanalyse und Gleichungsgenerierung weiterentwickelt.

Die Simulation komplexer DSKP erfordert die numerische Lösung von gewöhnlichen Differentialgleichungssystemen hoher Ordnung. Die Effizienz der Gleichungslöser in einer massiv-parallelen Simulationsumgebung hängt wesentlich von den verwendeten parallelen Integrationsalgorithmen ab. Die Analyse von solchen Algorithmen basiert auf der Betrachtung des allgemeinen Cauchy- Problems für lineare DS

$$\frac{d\bar{x}}{dt} = A\bar{x} + \bar{f}(t), \quad \bar{x}(t_0) = \bar{x}^0 = (x_1^0, x_2^0, \dots, x_m^0)^T \quad (1)$$

mit der Koeffizientenmatrix $A = (a_{ij})$, dem Zustandsvektor $\bar{x}(t)$ und der vorgegebenen zeitabhängigen Vektorfunktion $\bar{f}(t)$. Die Berechnungsformeln des Runge-Kutta-Verfahrens 4. Ordnung mit der Integrations-schrittweite τ

$$\begin{aligned} \bar{x}^{n+1} &= \bar{x}^n + \frac{1}{6}(\bar{k}_1^n + 2\bar{k}_2^n + 2\bar{k}_3^n + \bar{k}_4^n), \quad n=0,1,2,\dots, \text{hier } \bar{k}_1^n = \tau[A\bar{x}^n + \bar{f}^n], \\ \bar{k}_2^n &= \tau[A\bar{x}^n + 0.5\bar{k}_1^n + \bar{f}^{n+0.5}], \quad \bar{k}_3^n = \tau[A\bar{x}^n + 0.5\bar{k}_2^n + \bar{f}^{n+0.5}], \\ \bar{k}_4^n &= \tau[A\bar{x}^n + \bar{k}_3^n + \bar{f}^{n+1}] \end{aligned} \quad (2)$$

werden in einem SIMD-Rechner mit m^2 Prozessorelementen (PE) implementiert. Die Analyse der DSKP-SIMD-Algorithmen ist in [5] beschrieben. Ausserdem wurden in [6] die Benutzeroberfläche mit verschiedenen Ansätzen für die Modellerstellung sowie in [7] eine problemorientierte SIMD-MPSU untersucht.

5. Realisierung der MIMD-Komponente

Die vorgeschlagene MIMD-Komponente der MPSU besteht aus Hardware-Komponenten (MIMD-Rechner, lose gekoppelte Rechnercluster, simulationsorientierte Peripheriesysteme), aus der Systemsoftware (MIMD-Betriebssysteme, Programmiersysteme, Netzsoftware, MPI-Standard-Software), aus der Simulationssoftware (MIMD-parallele DSKP-/DSVP-Simulationssoftware, Visualisierungssoftware) sowie aus der Benutzeroberfläche mit den system- und simulationsorientierten Bedienfunktionen. Die MIMD-Komponente ist nach Abb.2 am Rechenzentrum der Universität Stuttgart (RUS) mit dem Rechnersystem Intel Paragon XP/S und in Donetsk mit 10 XP/E-Prozessorknoten aufgebaut.

Die vorhandene MIMD-parallele Programmiersprache und Systeme wie MPI und PVM ermöglichen dem Modellentwickler, parallelisierte numerische Verfahren zu programmieren und verschiedene DSKP-/DSVP-Modelle zu realisieren. Die Gleichungssysteme für DSKP und

örtlich diskretisierte DSVP werden als Cauchy-Problem (1) formuliert. Für die MIMD-Parallelisierung wird das SPMD-Schema empfohlen und jede Vektorkomponente X_j als Prozess T_j mit dem Prozessor P_j und mit ausgewählten numerischen Verfahren berechnet. Dabei erhält man wegen der Verwendung derselben Berechnungsformel (2) praktisch die gleichen Belastungen für alle MIMD-Prozessoren. Für gemischte Algebra- und Differentialgleichungen n-ter Ordnung wird das SPMD-Schema mit n Prozessoren und einer zusätzlichen Aufteilung des Algebra-Teils zwischen den Prozessoren benutzt. Für dynamische Systeme mit einer algebraischen Kontinuitätsbedingung kann man eine Kommunikationsmatrix bestimmen. Dabei erfolgt in jedem Iterationsschnitt ein Datenaustausch. Als Testbeispiele werden die MIMD-parallele Modelle von dynamischen Netzobjekten mit konzentrierten und mit verteilten Parametern untersucht, wie diese beispielsweise bei Bewetterungssystemen von Kohlegruben vorkommen.

6. MIMD-parallele Simulationssoftware

Die voll funktionsfähige DSVP-Simulationssoftware für die MIMD-Komponente umfasst die in Abb.3 gezeigten Module. Die DSKP-Simulationssoftware besitzt eine ganz ähnliche Struktur [5]. Die Simulationssoftware gliedert sich in die folgenden vier Bereiche: Benutzeroberfläche, Spezifikationsprache, interaktive Modellerstellung und Modellbibliothek. Mit Hilfe der in diesen Bereichen verfügbaren Module und Funktionen wird der Benutzer bei der Modellierung, der Auswahl der numerischen Verfahren und der Durchführung der Simulation unterstützt. Auf diese Weise wird der von den sequentiellen Simulationswerkzeugen bekannte Standard auch bei der parallelen Simulation erreicht.

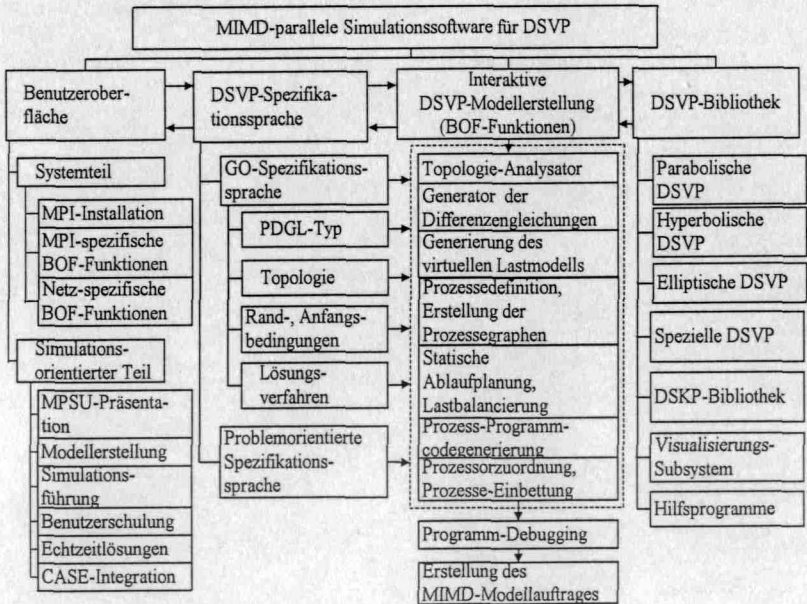


Abb.3. Aufbau der MIMD-parallelen Simulationssoftware für DSVP

7. Ansätze zur Entwicklung der MPSU- Simulationssprachen

Die bisherigen Erfahrungen mit der Modellerstellung für die SIMD- und MIMD-Komponenten zeigen, daß der Modellentwickler mit den vorhandenen parallelen Programmiersprachen eine Unterstützung auf dem Niveau der ehemaligen konventionellen Simulationssysteme der zweiten und dritten Generation [8] erhält. Deshalb werden die block- (BO), gleichungs- (GO) und objektorientierten (OO) Ansätze zur Entwicklung der parallelen Simulationssprachen für MPSU analysiert und zusammengeführt. Nach Abb.4 besteht damit auch für die parallele Simulationssprache ein für den Benutzer transparenter Zusammenhang zwischen den Anwendungsbereichen und den mathematischen und topologischen Beschreibungsformen für die dynamischen Systeme.

Eine wesentliche Bedeutung für die Erstellung von parallelen Simulationsmodellen haben die in Abb.5 dargestellten Entwicklungsebenen mit einer Dialogunterstützung für den Benutzer. Die vorgeschlagenen BO- und GO-Ansätze beinhalten im Unterschied zu den sequentiellen Spracheimplementierungen auch Werkzeuge zum Aufstellen der Modellgleichungen. Nach Abb.6 werden drei Arten von DS-Topologien unterschieden und entsprechenden Gleichungen generiert. Die Modellerstellung beinhaltet auch Algorithmen zur Lösung der aufgestellten Gleichungssystemen mit Hilfe des BO- oder GO-Modellprogramms. Die Gleichungslöser sind entsprechend der BO- und GO- Spezifikation der Modelle aufgebaut.

Der objektorientierte Modellierungsansatz basiert auf den vorhandenen BO- und GO-Erfahrungen und verallgemeinert entsprechend der Darstellung in Abb.7 die beiden anderen Ansätze.

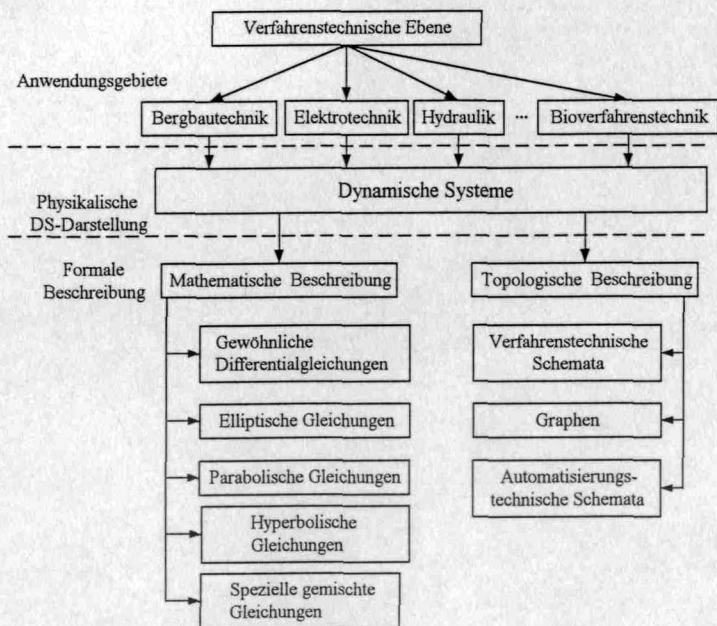


Abb.4. Mathematische und topologische Beschreibungsformen von dynamischen Systemen aus verschiedenen Anwendungsgebieten

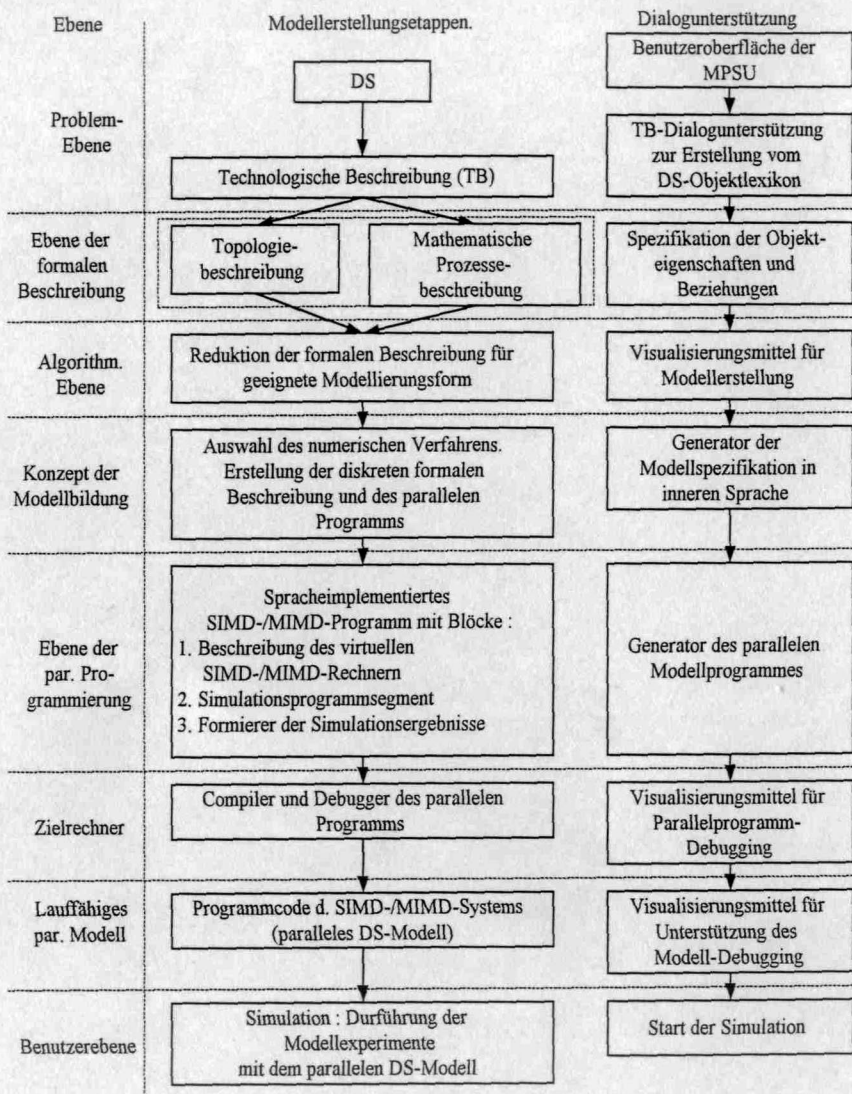


Abb.5. Entwicklungsebenen zum Aufstellen von parallelen Simulationsmodellen mit einer Dialogunterstützung für den Benutzer

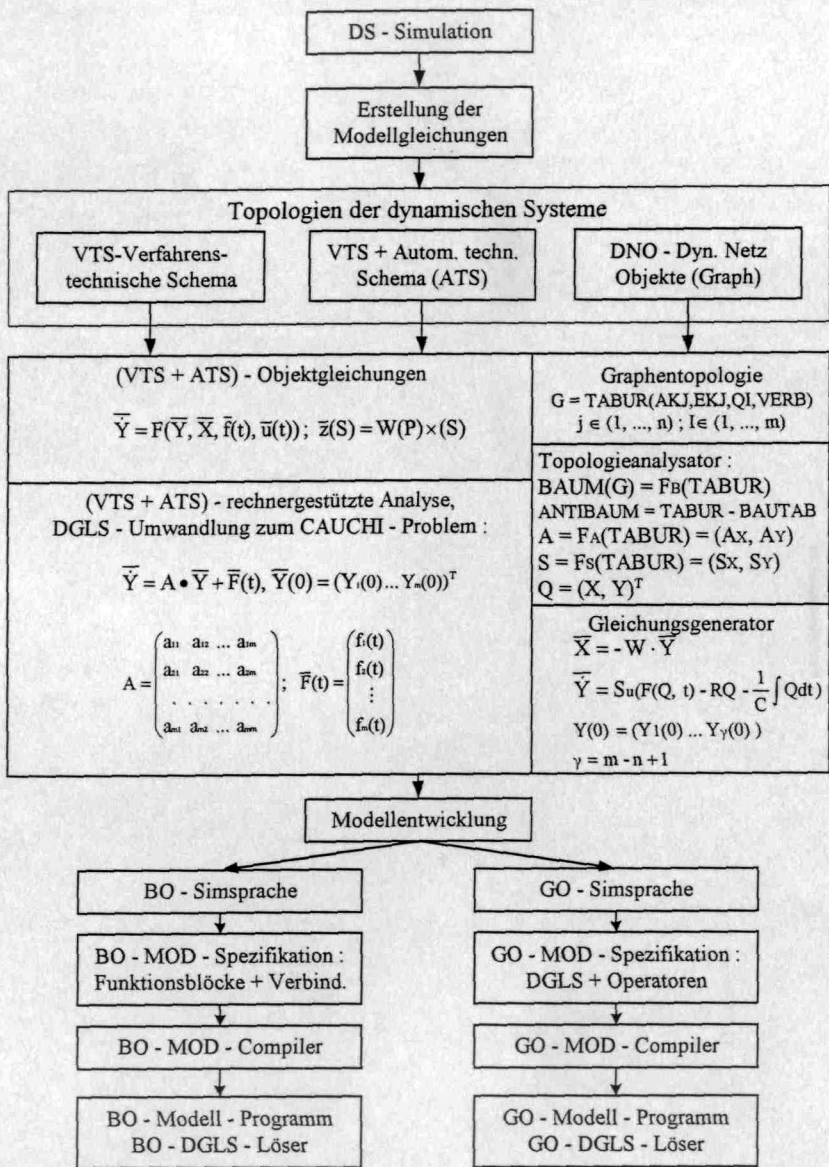


Abb.6. Block- und gleichungsorientierte Modellierungsansätze für die Modellentwicklung von dynamischen Systemen (DS)

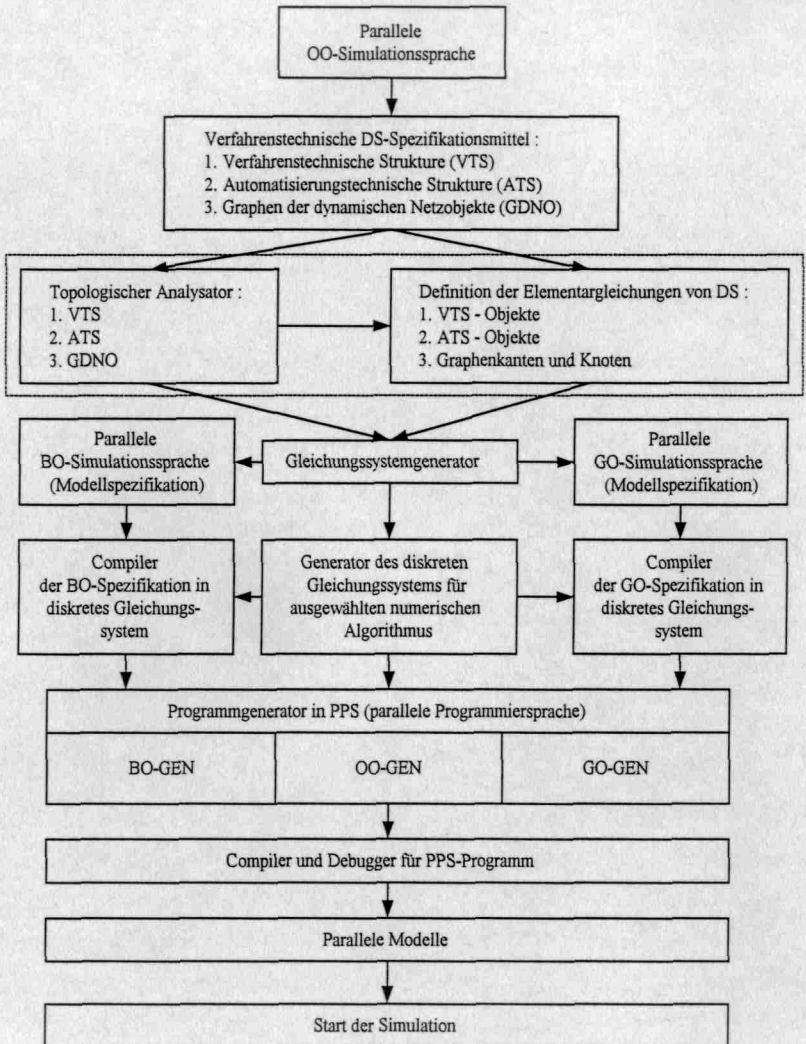


Abb.7. Funktionsebenen einer parallelen objektorientierten (OO) Simulationssprache für die MPSU

8. Zusammenfassung und Ausblick

Die vorgestellten Untersuchungen und Ergebnisse liefern einen Beitrag zur parallelen Simulationstechnik, der sich mit den algorithmischen, methodischen und implementierungstechnischen Problemen der Modellierung und Simulation von dynamischen Systemen mit konzentrierten (DSKP) und mit verteilten Parametern (DSVP) in verschiedenen Anwendungsbereichen befasst. Zur Zeit ist die parallele Simulationstechnik durch eine massiv parallele verteilte Simulationsumgebung (MPSU) mit SIMD- und MIMD-Komponenten in Donezk realisiert.

Haupteinsatzgebiete der parallelen Simulationstechnik sind die modellgestützte Projektierung von automatisierten DSKP und DSVP. Dabei hat es sich als zweckmässig erwiesen, die MPSU mit anwendungsspezifischen Modellierungs- und Simulationssprachen, Programm- und Modellbibliotheken sowie Benutzeroberflächen zu realisieren. Ein aktuelles Problem betrifft die Integration von den CASE-Werkzeuge mit der MPSU.

Die geplante weitere Entwicklung der parallelen Simulationstechnik konzentriert sich auf parallele Simulationssprachen, objektorientierte Ansätze, problemorientierte Simulationsumgebungen, Aspekte der verteilten Realisierung sowie auf industrielle und universitäre Anwendungen der MPSU.

Literatur

1. Bräunl, T.: Parallele Programmierung – Eine Einführung. Vieweg-Verlag 1993.
2. Anoprienko, A.J., Svjatnyj, V.A., Bräunl, T., Reuter, A., Zeitz, M.: Massiv parallele Simulationsumgebung für dynamische Systeme mit konzentrierten und verteilten Parametern. Tagungsband 9. Symposium ASIM'94, Vieweg-Verlag 1994, 183-188.
3. Feldmann, L.P., Lapko, V.V., Svjatnyj, V.A., Trub, I. I., Bräunl, T., Reuter, A., Zeitz, M.: Algorithmen einer massiv parallelen Simulationugebung für dynamische Systeme mit verteilten Parametern. Tagungsband 10. Symposium ASIM'96, Vieweg-Verlag 1996, 519-524.
4. Feldmann, L.P., Svjatnyj, V.A., Bräunl, T., Reuter, A., Zeitz, M.: Implementierung und Effizienzanalyse der parallelen Simulationsalgorithmen für dynamische Systeme mit verteilten Parametern. Tagungsband 11. Symposium ASIM'97, Vieweg-Verlag 1997, 38-47.
5. Feldmann, L.P., Svjatnyj, V.A., Gilles, E.-D., Zeitz, M., Reuter, A., Rothermel, K.: Parallele Simulationsalgorithmen für dynamische Systeme. Tagungsband 13. Symposium ASIM'99, SCS-Verlag 1999, 313-318
6. Anoprienko, A., Bazhenov, L., Bräunl, T. : The Development of the Interface Subsystem for the Massive Parallel Simulaton Environment. 11. Symposium ASIM'97, Tagungsband, Vieweg-Verlag 1997, 672-677.
7. Svjatnyj, V.A., Rasinkov, V.V., Bräunl, T., Reuter, A., Zeitz, M.: Problemorientierte massiv parallele Simulationugebung für dynamische Netzobjekte. Tagungsband 10. Symposium ASIM'96, Vieweg-Verlag1996, 515-518.
8. Schmidt, B.: Simulationssysteme der 5. Generation. SiP, Heft 1, 1994, 5-6.