

# Simulation kontinuierlicher Systeme unter Verwendung diskreter ereignisorientierter Algorithmen: ein Paradigmenwandel

François E. Cellier  
Departement Informatik  
ETH Zürich  
[FCellier@Inf.ETHZ.CH](mailto:FCellier@Inf.ETHZ.CH)

Wenn ein kontinuierliches System, welches durch ein Differentialgleichungssystem modelliert werden kann, auf einem Digitalrechner simuliert werden soll, muss der Lösungsraum auf die eine oder andere Weise diskretisiert werden. Die analytische Lösung ändert sich kontinuierlich mit der Zeit. Da jedoch der Digitalrechner in endlicher Zeit nur eine endliche Anzahl Berechnungen durchführen kann, muss der kontinuierliche Lösungsraum durch einen diskreten Lösungsraum angenähert werden.

Diese Diskretisierung kann auf unterschiedliche Weise vorgenommen werden. Beinahe alle heute auf dem „Markt“ verfügbaren Differentialgleichungslöser tun dies aber auf dieselbe Weise: sie diskretisieren die Zeitachse („time slicing“), während die Zustandsvariablen selbst beliebige Werte annehmen können. Somit wird die folgende Frage gestellt und beantwortet:

Gegeben der Wert der Zustandsvariablen und deren erste Ableitungen zur jetzigen Zeit  $t$  und zu einer Reihe früherer Zeitpunkte,  $t-h$ ,  $t-2h$ , etc., wobei  $h$  die Schrittlänge bezeichnet, welchen Wert werden die Zustandsvariablen zum Zeitpunkt  $t+h$  annehmen?

Alle Differentialgleichungslöser, die auf dieser Methodik aufgebaut sind, tun eigentlich dasselbe. Sie legen ein Interpolationspolynom durch die bekannten Stützwerte der Gegenwart und Vergangenheit und verwenden dieses dann in einem Extrapolationsschritt, um den Wert der Zustandsvariablen zu einem zukünftigen Zeitpunkt abzuschätzen. Dies trifft zumindest auf die expliziten Integrationsverfahren zu. Bei den impliziten Verfahren wird der numerisch problematische Extrapolationsschritt durch einen Interpolationsschritt ersetzt, indem der unbekannte Wert der zukünftigen Zustandsableitung als Stützwert mitverwendet wird. Dadurch ergibt sich in jedem Integrationsschritt ein Gleichungssystem, welches iterativ im Allgemeinen unter Einsatz des Newton Iterationsverfahrens gelöst wird.

Wenn nun doch alle Differentialgleichungslöser eigentlich dasselbe tun, warum gibt es dennoch verschiedene Löser auf dem „Markt“? Jeder Differentialgleichungslöser muss zwei Eigenschaften in den Griff bekommen: die Genauigkeit der Approximation

und die numerische Stabilität. Leider eignet sich nicht jeder Algorithmus gleichermaßen für die Simulation jedes Systems. Es gibt steife und nicht-steife Systeme; es gibt auch marginal stabile Systeme; wir müssen unterscheiden zwischen linearen und hoch nichtlinearen Systemen; und schliesslich müssen auch Systeme integriert werden, die häufige und heftige Diskontinuitäten aufweisen. Für jede Klasse dieser Systeme wurden im Laufe der letzten paar Jahrzehnte Löser entwickelt, die optimal für die Simulation dieser bestimmten Systemklasse geeignet sind.

In den letzten Jahren hat sich die Simulationslandschaft wieder ein wenig gewandelt. Moderne Rechner sind so leistungsfähig geworden, dass die Effizienz des Verfahrens gegenüber deren Robustheit in den Hintergrund getreten ist. Der Anwender einer Modellierungs- und Simulationsumgebung, der im Allgemeinen recht wenig von numerischer Mathematik versteht, will sich mit diesen Problematiken überhaupt nicht auseinandersetzen müssen. Er will einen Differentialgleichungslöser zur Verfügung haben, der fast alle Systeme simulieren kann, die in seinen numerischen Rachen geworfen werden.

Der robusteste Simulationscode, der zur jetzigen Zeit auf dem „Markt“ angeboten wird, ist DASSL, ein lineares implizites Mehrschrittverfahren mit variabler Schrittlänge und von variabler Ordnung. DASSL wurde für die Simulation linearer und/oder quasi-linearer steifer Differentialgleichungssysteme entwickelt. Bei der Simulation nicht-steifer Systeme ist DASSL ineffizient (da dort ein explizites Verfahren ebenfalls funktionieren würde) und unnötig ungenau (da der Fehlerkoeffizient unnötig gross ist), und auch bei der Simulation stark nichtlinearer und/oder diskontinuierlicher Systeme ist DASSL ineffizient, da dort die Schrittlänge häufig angepasst werden muss und da die Schrittsteuerung bei DASSL teuer ist.

Solange aber die Rechner nur schnell genug sind, ist die Bequemlichkeit des Anwenders wichtiger als die Effizienz des Differentialgleichungslösers, und so hat es sich eingebürgert, bei Modellierungs- und Simulationsumgebungen wie z.B. Dymola DASSL als Defaultlöser anzubieten. Es werden schon auch noch andere Löser angeboten; der durchschnittliche Anwender einer solchen Umgebung wird diese aber kaum je anwählen, es sei denn, DASSL schaffe es überhaupt nicht, ein gegebenes System zu simulieren. Dann wird der Anwender vielleicht und unter Protest auf einen anderen Löser ausweichen.

Wo ist denn hier noch Handlungsbedarf? Wenn die Anwender mit DASSL restlos glücklich sind, warum lohnt es sich dann überhaupt noch, nach anderen Lösern zu suchen als die, die bereits entwickelt wurden?

Es gibt immer noch zwei Problemklassen, bei denen die Bequemlichkeit gegenüber der Effizienz zurücktreten muss. Erstens gilt dies für Systeme mit verteilten Parametern, d.h. Systeme, die durch partielle Differentialgleichungen beschrieben werden, da dort die Effizienzeinbusse bei einem ungeeigneten Lösungsverfahren enorm gross sein kann. Zweitens gilt dies für Echtzeitsimulationen, da dort die Simulation zwangsläufig mit der Echtzeit Schritt halten muss.

Welche anderen Möglichkeiten gibt es denn noch, die exploriert werden können? Einige Forscher versuchten, die Diskretisation im Frequenzbereich vorzunehmen, d.h., statt dass die Zeitachse diskretisiert und die kontinuierliche Lösung durch eine Taylorreihe approximiert wird, wird die Frequenzachse diskretisiert. In diesem Fall wird die Lösung durch eine Fourierreihe approximiert, was besonders bei der Simulation harmonischer Schwingungen attraktiv sein kann. Solche Löser haben sich aber nie als allgemeine Differentialgleichungslöser durchsetzen können. Sie waren immer zu ineffizient, ausser für die Simulation einer recht kleinen Klasse ausgewählter Probleme.

Es gibt aber noch eine andere Klasse von Differentialgleichungslösern, die in den letzten Jahren Aufmerksamkeit auf sich gezogen hat. Bei diesen Lösern wird die Diskretisierung ebenfalls im Zeitbereich vorgenommen, aber statt der Zeitachse wird die Zustandsachse diskretisiert („state quantization“). Wir stellen und beantworten hier eine andere Frage:

Gegeben der Wert einer Zustandsvariablen und deren Ableitungen zum jetzigen Zeitpunkt  $t$ , zu welchem Zeitpunkt wird die Zustandsvariable zum ersten Mal um einen Wert  $\pm Q$  vom jetzigen Wert abweichen, wobei  $Q$  das Zustandsquantum bezeichnet?

Es wurden in den letzten Jahren eine ganze Reihe solcher Differentialgleichungslöser sowohl für nicht-steife wie auch für steife Systeme entwickelt. Im Gegensatz zu den Fourierlösern sind diese Verfahren durchaus mit den klassischen Lösern konkurrenzfähig, und es gibt grosse Klassen von Problemen, für die sich diese Verfahren besonders gut eignen. Sie sind besonders gut geeignet für die Simulation stark diskontinuierlicher Systeme, wie sie z.B. in der Leistungselektronik häufig anzutreffen sind – diese Systeme sind ausserdem häufig auch noch steif. Sie sind ebenfalls besonders gut geeignet für Systeme mit echten Verzögerungsgliedern („delay differential equations“), sogar wenn die Verzögerung selbst zeit- und vielleicht sogar zustandsabhängig ist. Schliesslich eignen sich diese Löser sehr gut für die Echtzeitsimulation, wo die klassischen Löser beinahe vollständig versagen.

Im Vortrag werden die Eigenschaften dieser Löser aufgezeigt, und deren Effizienz wird an Hand von Beispielsimulationen demonstriert. Es wird ebenfalls gezeigt, warum sich diese Löser besonders gut für bestimmte Problemklassen eignen, und was in den kommenden Jahren von diesen Lösern erwartet werden kann. Ein besonderes Augenmerk gilt dabei den Problemen der Echtzeitsimulation.