

# Evaluierung von Simulationsverfahren für allgemeine, physikalisch basierte Animationssysteme

Friedrich Wagner, Dietmar Jackèl\*  
Universität Rostock

## Zusammenfassung

In dieser Arbeit werden die gängigen Simulationsverfahren für physikalisch basierte Animationssysteme im Überblick präsentiert. Im besonderen werden die unterschiedlichen Ansätze für eine Einbindung der zur Animationssteuerung dienenden Constraints behandelt und mit ihren Vorzügen und Nachteilen diskutiert. Neben dem Penalty-Verfahren und der Reduktionsmethode steht hier vor allem die Lagrange-Faktoren-Methode im Mittelpunkt dieser Betrachtungen, deren effiziente Umsetzung in das konkret realisierte Animationssystem (EMPHAS) vorgestellt wird.

## 1 Einführung

Eine schwierige Aufgabe stellt bei der physikalisch basierten Erstellung von Computeranimationen die geeignete *Steuerung* der generierten Bewegungen dar. Besonders wichtig sind in diesem Zusammenhang *Constraints*, also allgemeine Bedingungen an die Zustandsvariablen eines dynamischen Systems, die auch für die Modellierung komplexer Objekte benötigt werden. Die oftmals sehr komplizierte Anwendung von Verfahren zur Simulation Constraint-behafteter Systeme stellt aber immer noch einen großen Hinderungsgrund für die allgemeine Verbreitung physikalisch basierter Techniken dar. Insbesondere stellt sich das Problem, wie die Verfahren in ein allgemeines Animationssystem integriert werden können, das Anforderungen wie Effektivität, Flexibilität, Intuitivität und Generalität zu erfüllen hat.

In dem nachfolgenden Abschnitt 2 werden die Simulationsverfahren zur Constraint-Einbindung in einer komprimierten Form vorgestellt. Zentraler Teil dieser Arbeit ist der Abschnitt 3, in dem die zuvor präsentierten Verfahren miteinander verglichen und bewertet werden. In Abschnitt 4 wird in Umrissen das Implementierungskonzept eines physikalisch basierten Animationssystems (EMPHAS) auf der Grundlage der von uns favorisierten Lagrange-Faktoren-Methode vorgestellt. Zum Abschluss, folgt im Abschnitt 5, eine Zusammenfassung der wichtigsten Ergebnisse der Evaluierung der betrachteten Simulationsverfahren.

---

\*Fachbereich Informatik, Institut für Computergraphik, D-18051 Rostock

## 2 Simulationsverfahren zur Constraint-Einbindung

Ein freies (nicht Constraint-behaftetes) mechanisches System lässt sich durch einen  $k$ -dimensionalen Zustandsvektor  $\mathbf{x}$  und einen  $k$ -dimensionalen Geschwindigkeitsvektor  $\mathbf{v}$  beschreiben, die der Zeitentwicklung

$$\mathbf{M}(\mathbf{x}) \dot{\mathbf{v}} = \mathbf{F}(\mathbf{x}, \mathbf{v}, t) \quad (1)$$

unterworfen sind. In der  $(k \times k)$ -dimensionalen Matrix  $\mathbf{M}$  sind dabei die Masseneigenschaften und in dem  $k$ -dimensionalen Vektor  $\mathbf{F}$  die Kraftwirkungen festgelegt. Es soll nun ein Überblick über die wichtigsten Ansätze zur Einbindung von Constraints in ein solches System, d.h. für ihre Berücksichtigung bei physikalischen Simulationen, gegeben werden.

Eine besonders wichtige Klasse von Constraints stellen dabei Gleichungen der Form  $\mathbf{J}(\mathbf{x}, \mathbf{v}, t) \cdot \dot{\mathbf{v}} + \mathbf{c}(\mathbf{x}, \mathbf{v}, t) = 0$  dar. Zu ihnen gehören sowohl *holonome* Constraints, die sich als Gleichung  $\mathbf{C}(\mathbf{x}, t) = 0$  formulieren lassen und z.B. sämtliche Gelenkverbindungen umfassen, als auch *nichtholonome* Constraints in der Form  $\mathbf{C}(\mathbf{x}, \mathbf{v}, t) = 0$ , die als *Geschwindigkeits-Constraints* bezeichnet werden sollen.

**Penalty-Verfahren.** Der Penalty-Ansatz stellt die einfachste Möglichkeit zur Constraint-Einbindung dar, weshalb er z.B. in [MW88] für die Behandlung von Ruhekontakten eingesetzt wurde. Bei holonomen Constraints wird dem System eine künstliche Kraft in der Form

$$\mathbf{F}^{Penalty} := -\alpha \mathbf{J}^T (\Omega^2 \mathbf{C} + 2\Omega \mu \dot{\mathbf{C}})$$

hinzugefügt. Dabei stellt  $\mathbf{J} = \left(\frac{\partial \mathbf{C}}{\partial \mathbf{x}}\right)$  die Jacobi-Matrix von  $\mathbf{C}$  und  $\alpha$ ,  $\Omega$  und  $\mu$  konstante Parameter in Form von Skalaren oder Diagonalmatrizen dar (siehe z.B. [GB94]). Diese Kraft wirkt einer Constraint-Verletzung entgegen, wobei der  $\dot{\mathbf{C}}$ -Term aus Stabilisierungsgründen eingefügt wird<sup>1</sup>. Auch nichtholonome Geschwindigkeits-Constraints  $\mathbf{C}(\mathbf{x}, \mathbf{v}, t) = 0$  können mit dem Penalty-Verfahren eingebunden werden, indem eine Kraft

$$\mathbf{F}^{Penalty} := -\alpha \left(\frac{\partial \mathbf{C}}{\partial \mathbf{v}}\right)^T \mu \mathbf{C}$$

mit den Parametern  $\alpha$  und  $\mu$  definiert wird. Diese Penalty-Kräfte lassen sich dann als zusätzliche äußere Kräfte in die Bewegungsgleichungen (1) einfügen. Die Lösung dieses Systems aus gewöhnlichen Differentialgleichungen führt zu einer *näherungsweise* Erfüllung der Constraints, exakt wäre das Verfahren nur für den Grenzfall eines unendlich großen Penalty-Terms  $\alpha$ . Große Werte für  $\alpha$  führen allerdings zu sogenannten *steifen* Differentialgleichungen, die spezielle numerische Methoden erfordern und nur mit einem sehr großen Zeitaufwand gelöst werden können.

**Reduktionsmethode.** Allgemein werden  $n$  Freiheitsgrade eines freien Systems durch  $m$  (skalare) Constraints auf  $n - m$  Freiheitsgrade reduziert. Man kann da-

<sup>1</sup>In [GB94] wird zusätzlich ein  $\ddot{\mathbf{C}}$ -Term eingeführt, was zu einer höheren Stabilität führt, aber zusätzlich die Auflösung eines linearen Gleichungssystems erfordert.

her ein solches System durch  $n-m$  unabhängige, generalisierte Koordinaten  $q_i$  beschreiben. Der Satz der Zustandsvariablen  $x_i$  des freien Systems läßt sich dann als Funktion dieser Größen formulieren:

$$x_1 = x_1(q_1, \dots, q_{n-m}), \dots, x_n = x_n(q_1, \dots, q_{n-m}). \quad (2)$$

Die Reduktionsmethode besteht nun aus den folgenden Schritten:

- Auswahl von  $n-m$  unabhängigen, generalisierten Koordinaten  $q_i$ .
- Explizite Parametrisierung des Systems durch die unabhängigen Koordinaten gemäß Gleichung (2).
- Aufstellung von  $n-m$  Bewegungsgleichungen bezüglich der  $q_i$ , z.B. mit Hilfe des *Lagrange-Formalismus* (siehe z.B. [Gol89]).
- Standardmäßige Lösung des resultierenden  $(n-m)$ -dimensionalen Systems gewöhnlicher Differentialgleichungen.

Die durch die Constraints verursachten Kräfte tauchen dabei im Gegensatz zu anderen Verfahren nicht explizit in den Bewegungsgleichungen auf, was eine sehr elegante Beschreibung ermöglicht. Für die explizite Parametrisierung in Form von Gleichung (2) muß allerdings ein nichtlineares Gleichungssystem aufgelöst werden. Durch Singularitäten kann ein Satz unabhängiger Koordinaten zudem auch während der Simulation inadäquat werden.

In [AG85], [A<sup>+</sup>87], [IC87] ist die Reduktionsmethode erstmals für die Animation von Gelenkkörpern angewandt worden. Mit der sogenannten *Articulated Body Method* wurde in [Fea87] eine rekursive Umsetzung für hierarchische Systeme mit einer linearen Zeitkomplexität beschrieben, die u.a. in [MZ90] und [VC91] Anwendung fand. Eine Erweiterung auf nichthierarchische Systeme wurde dann in [SZ90] für die Animation mechanischer Systeme eingesetzt.

**Lagrange-Faktoren-Methode (LFM).** Physikalisch betrachtet werden Constraints durch das Wirken von Kräften erfüllt. Die LFM beruht nun auf der expliziten Bestimmung dieser Zwangskräfte (Constraint-Kräfte)  $\mathbf{F}^C$ , die anschließend dem System als zusätzliche Kräfte hinzugefügt werden. Als Constraints können dabei alle Beziehungen in der Form  $\mathbf{J}(\mathbf{x}, \mathbf{v}, t) \cdot \dot{\mathbf{v}} + \mathbf{c}(\mathbf{x}, \mathbf{v}, t) = 0$  berücksichtigt werden. Die Constraint-Kräfte  $\mathbf{F}^C$  lassen sich mit Hilfe der sogenannten *Lagrange-Faktoren* ausdrücken. Ein physikalisches Prinzip sagt aus, daß derartige Zwangskräfte keine Arbeit leisten können ([Gol89]). Die mathematische Entsprechung dieser Forderung ist die Beziehung  $\mathbf{F}^C = \mathbf{J}^T \lambda$ , wobei  $\lambda$  einen (zunächst noch unbestimmten) Vektor darstellt, dessen Komponenten als *Lagrange-Faktoren* bezeichnet werden. Die Bewegungs- und Constraint-Gleichungen bilden somit das differentiellalgebraische Gleichungssystem

$$\begin{aligned} \mathbf{M} \dot{\mathbf{v}} - \mathbf{F} - \mathbf{J}^T \lambda &= 0 \\ \mathbf{J} \dot{\mathbf{v}} + \mathbf{c} &= 0, \end{aligned} \quad (3)$$

das bei  $n$  Zustandsvariablen und  $m$  Constraints aus  $n + m$  Gleichungen besteht. Zur Lösung dieses Systems geht man bei der LFM in zwei Schritten vor:

- Bestimmung der Lagrange-Faktoren  $\lambda$  als Lösung des folgenden Gleichungssystems, das sich durch eine einfache Umformung aus Gleichung (3) ergibt:

$$(\mathbf{J} \mathbf{M}^{-1} \mathbf{J}^T) \lambda = -(\mathbf{J} \mathbf{M}^{-1} \mathbf{F} + \mathbf{c}) . \quad (4)$$

- Einsetzen von  $\lambda$  in die Bewegungsgleichungen und standardmäßiges Lösen des  $n$ -dimensionalen Systems gewöhnlicher Differentialgleichungen

$$\dot{\mathbf{v}} = \mathbf{M}^{-1}(\mathbf{F} + \mathbf{J}^T \lambda) . \quad (5)$$

Eine grundsätzliche Schwierigkeit aller auf Gleichung (3) beruhenden Verfahren ist das sogenannte Driftproblem. Holonome Constraints werden nur über die Gleichung  $\dot{\mathbf{C}} = 0$  berücksichtigt, deren allgemeine Lösung  $\mathbf{C}(t) = a t + b$  einer ständig wachsenden Verletzung der Constraint-Bedingung entspricht. Eine in [BB88, Pla92] beschriebene Lösung dieses Problems besteht in der Anwendung eines Stabilisierungsverfahrens, bei dem der Term  $\mathbf{c}$  in Gleichung (3) durch den Term

$$\tilde{\mathbf{c}} = \mathbf{c} + 2 \alpha \dot{\mathbf{C}} + \beta^2 \mathbf{C}$$

mit zwei skalaren Parametern  $\alpha$  und  $\beta$  ersetzt wird. Als alternatives Verfahren werden die Lagrange-Faktoren in [W<sup>+</sup>90] zunächst wie üblich berechnet, während der Term  $2 \alpha \dot{\mathbf{C}} + \beta^2 \mathbf{C}$  als zusätzliche Kraft in das System eingeht.

Die LFM stellt ein allgemeines und einfach anwendbares Verfahren dar und ist daher schon in [IC88] und [A<sup>+</sup>89] für die Animation von Gelenkkörpern eingesetzt worden. Als sehr nützlich hat sich diese Methode auch für die Animation von deformierbaren Körpern ([PB88], [WW90]) und für kombinierte Systeme aus starren und deformierbaren Körpern erwiesen ([Met95], [C<sup>+</sup>95]). In [BW97] wurde eine Kapselung und effiziente Kombination von Simulationsverfahren für unterschiedliche Anwendungsbereiche auf der Grundlage der LFM vorgestellt.

Eine besonders interessante Anwendung stellt das in [BB88] eingeführte und in [W<sup>+</sup>90] und [Pla92] erweiterte *Dynamic Constraints*-Konzept dar. Da Constraints durch das Wirken von Kräften gelöst werden, geschieht der Prozess der Constraint-Erfüllung bei der LFM in einer stetigen, physikalisch basierten Weise. In diesen Arbeiten wurde gezeigt, wie sich diese Eigenschaft für eine intuitivere Modellierung einsetzen lässt.

Die effiziente Anwendung der LFM galt allerdings lange als sehr kompliziert. Einen großen Fortschritt stellt daher die einfach umsetzbare und effiziente Umsetzung von David Baraff ([Bar96]) dar. Constraints, die Paare von Körpern beeinflussen, können mit diesem Verfahren mit einer linearen Zeitkomplexität bezüglich der Anzahl der Freiheitsgrade gelöst werden.

**Direkte numerische Verfahren.** Das differentiell-algebraisches Gleichungssystem (3) lässt sich auch direkt mit Hilfe numerischer Verfahren lösen, d.h. ohne eine

vorherige Umformung in ein gewöhnliches Differentialgleichungssystem. Diese Verfahren sind allerdings weniger effizient und stabil als die anderen hier vorgestellten Lösungsansätze (vergl. [Val93] und [GB94], S. 163).

**Koordinatenaufteilung.** Ausgangspunkt dieses Ansatzes sind die Bewegungsgleichungen in nichtreduzierter Form wie Gleichung (3). Mit Hilfe dieser Gleichungen wird dann eine Aufspaltung in  $n-m$  unabhängige Koordinaten  $\mathbf{q}_u$  und  $m$  abhängige Koordinaten  $\mathbf{q}_v$  vorgenommen. Die Integration geschieht dann nur bezüglich  $\mathbf{q}_u$ , während die  $\mathbf{q}_v$ -Werte in einem anschließenden Schritt aus den Constraint-Gleichungen bestimmt werden. Im Unterschied zur Reduktionsmethode kann das Verfahren auch auf die Einbindung von Geschwindigkeits- und anderen nichtholonomen Constraints erweitert werden und ermöglicht die Aufstellung der Bewegungsgleichungen in beliebigen Koordinaten. Ein schwieriges Problem stellt allerdings wie bei der Reduktionsmethode die automatische Bestimmung und Auswahl der unabhängigen Koordinaten dar.

Obwohl die Koordinatenaufteilung nicht den Effizienzgrad der mit der Baumgart-Stabilisierung ergänzten LFM erreicht ([Val93]), stellt dieser Ansatz ein effizientes und in der Technischen Simulation häufig angewandtes Verfahren dar.

**Projektionsmethode.** Durch die Projektion des Zustandsraums auf den Unterraum der erlaubten Bewegungen mit Hilfe einer Projektionsmatrix läßt sich eine Umformulierung des Gleichungssystems (3) durchführen, bei der die Zwangskraft-Terme eliminiert werden und ein System von  $n$  gewöhnlichen Differentialgleichungen resultiert. Die Projektionsmatrix muß dabei bei jedem Simulationsschritt neu bestimmt werden, bei holonomen Constraints z.B. durch Triangulierung der Jacobi-Matrix  $\mathbf{J}$ . Im Bereich der Technischen Simulation werden diese Verfahren sehr häufig eingesetzt, um den dynamischen und kinematischen Anteil der Bewegungsgleichungen voneinander trennen zu können. Für den Einsatz im Bereich der Computera-nimation ergeben sich aber keine grundsätzlichen Vorteile gegenüber den anderen Verfahren, so daß auf diesen Ansatz nicht weiter eingegangen werden soll.

**Gibbs-Appel-Formalismus.** Der Gibbs-Appel-Formalismus hat in der Computergraphik eine gewisse Bekanntheit erlangt, da er in den frühen Arbeiten [WB85], [Wil87] zur Animation von Gelenkkörpern angewandt wurde. Er ist aber durch eine Zeitkomplexität von  $\mathcal{O}(n^4)$  bezüglich der Anzahl der Freiheitsgrade gekennzeichnet und eignet sich daher nicht für eine effektive Constraint-Einbindung ([Wil88]).

**Constraint-Entkopplung.** Ein völlig anderer Ansatz wurde in [vO90], [vO91] sowie in [GG94] mit der Entkopplung von freier Bewegung und Constraint-Erfüllung vorgeschlagen. Für jeden Constraint müssen hierfür aber passende Korrekturterme (wie in [GG94]) bzw. Reaktionskräfte (wie in [vO90], [vO91]) konstruiert werden und selbst mit diesen läßt sich nur ein „plausibles“ Bewegungsverhalten erzielen. Als Grundlage von allgemeinen Animationssystemen ist dieser Ansatz daher weniger geeignet.

### 3 Evaluierung der Simulationsverfahren

Im Bereich der Computeranimation wurden die Besonderheiten der jeweiligen Verfahren zur Constraint-Einbindung bislang nur sehr selten betrachtet und ausgenutzt. Lediglich in [BB88], [W<sup>+</sup>90], [Pla92], [Bar96] und [BW98] wird explizit auf derartige Eigenschaften verwiesen, wobei nur in die drei letztgenannten Arbeiten vergleichende Betrachtungen zu anderen Simulationsverfahren enthalten. Die systematische Untersuchung der Eignung der Simulationsverfahren für die Realisierung allgemeiner Animationssysteme, die den in der Einführung genannten Anforderungen gerecht werden, ist dagegen noch nicht durchgeführt worden. Eine solche Analyse soll nun vorgestellt werden.

Grundlage dieser Analyse, die in ausführlicher Form in [Wag01] wiedergegeben ist, ist die Ausarbeitung der folgenden Anwendungsaspekte. Ihre Realisierbarkeit durch die wichtigsten der in Abschnitt 2 beschriebenen Simulationsverfahren ist in Tabelle 1 aufgeführt.

	Penalty	LFM	Reduktion	Aufteilung	Direkt	Relevanz
1a	niedrig	hoch	hoch	mittel	niedrig	hoch
1b	mittel	mittel	mittel	mittel	niedrig	hoch
2a	ja	ja	nein	zusätzl.	ja	hoch
2b	zusätzl.	zusätzl.	nein	zusätzl.	zusätzl.	hoch
3a	(ja) <sup>a</sup>	ja	zusätzl.	zusätzl.	zusätzl.	mittel
3b	(ja) <sup>a</sup>	ja	zusätzl.	zusätzl.	zusätzl.	niedrig
4a	ja	ja	zusätzl.	zusätzl.	zusätzl.	mittel
4b	(ja) <sup>b</sup>	(ja) <sup>b</sup>	zusätzl.	zusätzl.	(ja) <sup>b</sup>	niedrig
4c	zusätzl.	zusätzl.	ja	ja	ja	mittel
5a	ja	ja	(nein) <sup>c</sup>	(nein) <sup>c</sup>	(nein) <sup>c</sup>	hoch
5b	ja	ja	(nein) <sup>c</sup>	(nein) <sup>c</sup>	(nein) <sup>c</sup>	hoch
5c	ja	ja	(nein) <sup>c</sup>	(nein) <sup>c</sup>	(nein) <sup>c</sup>	mittel
5d	(ja) <sup>d</sup>	(ja) <sup>d</sup>	ja	ja	ja	niedrig
6a	ja	ja	nein	ja	ja	mittel
6b	ja	ja	nein	ja	ja	hoch
6c	(nein) <sup>e</sup>	(nein) <sup>e</sup>	ja	(nein) <sup>e</sup>	(nein) <sup>e</sup>	mittel
7a	ja	ja	nein	nein	ja	mittel
7b	ja	ja	zusätzl.	ja	ja	hoch
7c	nein	nein	ja	ja	ja	mittel

<sup>a</sup>Keine exakten Constraint-Kräfte, sondern künstlich eingeführte Federkräfte.

<sup>b</sup>Bis auf unstetige Zustandsänderungen bei Kollisionssimulationen.

<sup>c</sup>Nur durch wesentliche Erweiterungen der Verfahren realisierbar.

<sup>d</sup>Nahezu instantane Constraint-Erfüllung durch geeignete Stabilisierungsparameter.

<sup>e</sup>Nur durch wesentliche Einschränkungen anderer Eigenschaften sinnvoll anwendbar.

Tabelle 1: Realisierbarkeit (ja, nein oder nur durch zusätzliche Aufwendungen möglich) und Relevanz der Anwendungsaspekte. Für die Punkte 1a und 1b ist der Grad der Effizienz bzw. Robustheit angegeben.

## 1. Effizienz und Robustheit

- (a) **Effizienz.** Die Interaktivität stellt eine wesentliche Anforderung an allgemeine Animationssysteme dar und kann nur durch ein hinreichend effizientes Simulationsverfahren erreicht werden.
- (b) **Robustheit.** Eine geringe Robustheit äußert sich vor allem in der Anfälligkeit gegenüber singulären oder numerisch ungünstigen Konfigurationen.

## 2. Einbindung nichtholonomer Constraints

- (a) **Einbindung von Geschwindigkeits-Constraints.** Geschwindigkeits-Constraints in der Form  $C(\mathbf{x}, \dot{\mathbf{x}}, t) = 0$  können vor allem für die flexible Bewegungskontrolle eine wichtige Rolle spielen.
- (b) **Einbindung von Ungleichungen.** Auch Bedingungen an die Zustandsvariablen in Form von Ungleichungen, z.B. die Forderung der Undurchdringlichkeit für dreidimensionale Körper, stellen wichtige Einflußmöglichkeiten dar.

## 3. Explizite Bestimmung der Zwangskräfte

Der physikalische Ausdruck eines festgelegten Constraints ist die Wirkung seiner zugehörigen Zwangskraft, die bei der LFM explizit berechnet wird. Dieses Kenntnis kann in zweierlei Weise ausgenutzt werden:

- (a) **Anwendung für spezielle Steuertechniken.** Die Größe der Zwangskräfte stellt eine zusätzliche dynamische Systemeigenschaft dar, die von speziellen Steuertechniken ausgenutzt werden kann. Für bestimmte Ansätze zur Kontaktbehandlung muß außerdem die Richtung dieser Kräfte bekannt sein ([Pla92]).
- (b) **Realisierung einer Kraftrückkopplung.** Durch die Umwandlung in reale Kräfte, die dem Benutzer über ein Force Feedback- Eingabegerät erfahrbar gemacht werden, kann die Kenntnis der Zwangskräfte für eine sehr intuitive Interaktion mit den Szenenobjekten ausgenutzt werden. In [NC99] wurde eine solche Technik im Rahmen von Modellierungsaufgaben beschrieben, bei der das Problem der Bewegungsgenerierung unberücksichtigt blieb.

## 4. Erfüllung der Constraints durch Kraftwirkungen

Bei der LFM und bei der Penalty-Methode werden Constraints durch das Wirken von Kräften erfüllt, woraus sich neben dem physikalisch basierten Ablauf der Constraint-Erfüllung (Punkt 5) interessante Aspekte ergeben.

- (a) **Erhaltung der Stabilität bei elastischen Körpern.** Durch die stetigen Kraftanwendungen werden extrem große Ableitungen der Zustandsvariablen vermieden, was die stabile Simulation elastischer Körper erleichtert ([Pla92]).
- (b) **Standardmäßige Einbindung von numerischen Integrationsroutinen.** Im Gegensatz zu Verfahren, die zu sprunghaften Zustandsänderungen führen oder sogar einen Wechsel der Zustandskoordinaten während der Simulation erfordern können, ermöglicht die stetige Constraint-Erfüllung die standardmäßige Einbindung von numerischen Integrationsroutinen ([Pla92], [GB94]).

- (c) **Vermeidung ungewollter Bewegungen.** Ein Aspekt, der durch die Erfüllung der Constraints durch Kraftwirkungen erschwert wird, ist die Vermeidung von Bewegungen, die als Artefakte des Lösungsverfahrens entstehen. Durch zusätzliche Techniken lassen sich diese Artefakte aber weitestgehend unterdrücken, wie in [Wag01] genauer ausgeführt wird.

## 5. Physikalisch basierter Prozess der Constraint-Erfüllung

Die Constraint-Erfüllung wird bei der LFM und bei der Penalty-Methode nicht instantan, sondern durch einen Prozess, der wie die generierten Bewegungsverläufe aus der Auswertung der physikalischen Bewegungsgleichungen resultiert, durchgeführt. Im Rahmen der interaktiven Animationserstellung ist dieser Aspekt sehr wichtig, da der Systemzustand beliebig weit von der Erfüllung eines Constraints entfernt sein kann.

- (a) **Keine Bewegungssprünge bei der Animationserstellung.** Constraints lassen sich während der Simulation aktivieren oder deaktivieren, ohne unnatürlich wirkende Bewegungssprünge zu verursachen, wie auch in [Pla92] angemerkt wurde. Dieser Aspekt kann für eine höhere Flexibilität der Bewegungssteuerung ausgenutzt werden.
- (b) **Intuitive Anwendung von Constraints.** Dem Animator muß eine visuelle Rückkopplung über das Systemverhalten auf der Grundlage seiner Festlegungen ermöglicht werden. Mit der stetigen, physikalisch basierten Constraint-Erfüllung wird diese Anforderung in natürlicher Weise erfüllt. Dieser Aspekt bezieht sich dabei nicht nur auf Modellierungsaufgaben wie bei dem in [BB88] eingeführten *Dynamic Constraint*-Konzept, sondern auch auf die Anwendung von Constraints zur Bewegungssteuerung.
- (c) **Einheitliche Methoden für Modellierung und Bewegungssteuerung.** Bei der Anwendung von Constraints läßt sich jeder Modellervorgang aufgrund der physikalisch basierten Erfüllung auch als visuell plausible Animation einsetzen. Diese Einheitlichkeit kann das Erlernen und Anwenden der Techniken erheblich vereinfachen.
- (d) **Möglichkeit zur instantanen Erfüllung der Constraints.** In bestimmten Fällen kann eine instantane Constraint-Erfüllung auch erwünscht sein. Auch bei der LFM und der Penalty-Methode läßt sich aber über geeignet gewählte Stabilisierungsparameter eine nahezu instantane Erfüllung erreichen.

## 6. Bewegungsgleichungen bezüglich beliebiger Zustandskoordinaten

Bis auf die Reduktionsmethode, die auf unabhängigen Koordinaten beruht, können bei den Verfahren beliebige Koordinaten für die Aufstellung der Bewegungsgleichungen zum Einsatz kommen, woraus sich einige wichtige Aspekte ergeben:

- (a) **Schematische Aufstellung der Bewegungsgleichungen.** Das Aufstellen der Bewegungsgleichungen läßt sich bei der Verwendung beliebiger Koordinaten sehr schematisch durchführen und ist daher leicht umzusetzen.



- (b) **Unabhängige Formulierung von Körpern und Constraints.** Durch die Verwendung beliebiger Zustandskoordinaten lassen sich Körper und Constraints unabhängig voneinander formulieren. Dieser Umstand erleichtert nicht nur die Aufstellung der Bewegungsgleichungen, sondern ermöglicht auch eine *Kapselung* von Körpern und Constraints in separate Objekte, wie u.a. in [Bar96] angemerkt wurde. Diese Eigenschaft läßt sich für eine größere Modularität von Animationsystemen und für eine einfachere Anwendung von Constraints ausnutzen, wie in Abschnitt 4 genauer erläutert wird.
- (c) **Integrierte Bestimmung relativer Koordinaten.** Relative Koordinaten, wie z.B. der Auslenkungswinkel eines Scharniergelenks, stellen einfach interpretierbare Zustandsgrößen dar, die für die Bewegungssteuerung sehr nützlich sein können. Sie lassen sich bei Verfahren auf der Grundlage beliebiger Zustandskoordinaten durch einen zusätzlichen Berechnungsschritt bestimmen. Neben dem dafür anfallenden Zeitaufwand wären damit allerdings die unter den Punkten 6a, 6b und 7a genannten Vorteile hinfällig, da sie gerade auf der Vermeidung einer solchen Koordinatentransformation beruhen. Wie in [Wag00] und [Wag01] gezeigt wird, läßt sich aber auch ohne relative Koordinaten eine flexible und intuitive Bewegungssteuerung erzielen.

## 7. Verfahrensspezifische Aspekte

- (a) **Vermeidung einer Trennung von abhängigen und unabhängigen Koordinaten.** Die Reduktionsmethode und die Koordinatenaufteilung erfordern die explizite Bestimmung der unabhängigen Koordinaten<sup>2</sup>. Die (i.a. nichtlinearen) Constraint-Gleichungen müssen dazu nach den Zustandsvariablen aufgelöst werden, was eine sehr schwer zu automatisierende Aufgabe darstellt und auch während der Simulation notwendig werden kann.
- (b) **Anwendung auf beliebige Systemtopologien.** Im Gegensatz zu den anderen Verfahren läßt sich die Reduktionsmethode standardmäßig nur auf hierarchische Systeme anwenden. Systeme mit geschlossenen Schleifen (zyklischen Verbindungen), die bei der Animationserstellung sehr häufig vorliegen und oftmals erst während der Simulation entstehen, erfordern daher bei dieser Methode zusätzliche Bearbeitungsschritte.
- (c) **Vermeidung zusätzlicher Stabilisierungsparameter.** Bei der LFM stellt sich das Problem der geeigneten Festlegung der Stabilisierungsparameter  $\alpha$  und  $\beta$  (vergl. Abschnitt 2), die einen großen Einfluß bei der Verfahrensanwendung haben können, aber relativ unintuitive Größen darstellen. Das gleiche Problem ergibt sich auch bei dem Penalty-Verfahren, das mit dem Penalty-Faktor sogar die Festlegung eines zusätzlichen Parameters erfordert.

Es soll nun eine Zuordnung dieser Anforderungsaspekte zu den in der Einführung genannten Anforderungen an allgemeine Animationssysteme vorgenommen werden

---

<sup>2</sup>Bei der Reduktionsmethode ist diese Bestimmung dabei schon vor der Aufstellung der Bewegungsgleichungen erforderlich (vergl. Punkt 6).

und auf dieser Grundlage eine zusammenfassende Einschätzung der Simulationsverfahren erfolgen.

**Effektivität.** Ein hoher Grad an zeitlicher Effektivität (Punkt 1a) stellt einen äußerst wichtigen Gesichtspunkt dar. Bezüglich dieser Eigenschaft haben die LFM, die Reduktionsmethode und mit gewissen Einschränkungen auch die Koordinatenaufteilung Vorteile gegenüber den beiden anderen Verfahren.

**Flexibilität.** Durch die Möglichkeit zur Einbindung nichtholonomer Constraints (Punkte 2a und 2b), die bei der Reduktionsmethode nicht besteht, läßt sich eine große Klasse von Constraints für die Modellierung und Bewegungssteuerung einsetzen. Die Kenntnis der Zwangskräfte kann zudem wie unter Punkt 3a beschrieben von speziellen Steuertechniken ausgenutzt werden. Durch die Vermeidung von Bewegungssprüngen (Punkt 5a) lassen sich Constraints außerdem sehr flexibel einsetzen, da sie auch während einer laufenden Simulation aktiviert und deaktiviert werden können, was einen wichtigen Vorteil der LFM und des Penalty-Verfahrens darstellt.

**Intuitivität.** Für die einfache und intuitive Bedienung von Animationssystemen bieten die stetige Constraint-Erfüllung (Punkt 5b) und die hiermit verknüpfte Einheitlichkeit bezüglich Modellierungs- und Steueraufgaben (Punkt 5c) sowie die unter Punkt 3b genannte Verwendung der Zwangskräfte interessante Möglichkeiten. Auch die unabhängige Formulierung von Constraints und Körpern (Punkt 6b) kann für eine einfache Anwendung von Constraints ausgenutzt werden. Für den Gesichtspunkt der Intuitivität bieten die LFM und das Penalty-Verfahren daher wesentliche Vorteile. Einen gewissen Nachteil dieser Verfahren stellen lediglich die Anpassung der Stabilisierungsparameter (Punkt 7c) und das Auftreten ungewollter Bewegungen (Punkt 4c) dar.

**Generalität und einfache Umsetzbarkeit.** Wichtige Beispiele für die einfache Umsetzbarkeit der Lösungsverfahren im Rahmen eines allgemeinen Animationssystems stellen die schematische Aufstellung der Bewegungsgleichungen (Punkt 6a), die Vermeidung der Bestimmung unabhängiger Koordinaten (Punkt 7a), die standardmäßige Einbindung von Integrationsroutinen (Punkt 4b) sowie die einfache Einbindung von elastischen Körpern (Punkt 4a) und Systemen mit geschlossenen Schleifen (Punkt 7b) dar. Auch die Robustheit der Lösungsverfahren (Punkt 1b) und die modulare Konzeption von Körpern und Constraints (Punkt 6b) trägt entscheidend für eine einfache Umsetzbarkeit bei. Die LFM und das Penalty-Verfahren bieten für diese Aspekte die größten Vorteile, wie Tabelle 1 entnommen werden kann.

Wie diese Ausführungen gezeigt haben, bietet die LFM somit hervorragende Voraussetzungen für den Einsatz im Rahmen eines allgemeinen, insbesondere interaktiven Animationssystems. Das Penalty-Verfahren teilt sehr viele Anwendungsaspekte mit der LFM, läßt sich aber – wie auch die direkte numerische Lösung – bei weitem nicht so effizient umsetzen. Die anderen hier betrachteten Verfahren, insbesondere die Reduktionsmethode, führt dagegen zu wesentlichen Einschränkungen und zusätzlichen Aufwendungen bei der Umsetzung eines allgemeinen Animationssystems.

## 4 Physikalisch basierte Animation auf der Grundlage der LFM

Im folgenden werden einige wichtige Konzepte für die physikalisch basierte Animation auf der Grundlage der LFM besprochen, die sich aus der oben dargestellten Analyse der Simulationsverfahren ergeben. Praktisch umgesetzt wurden diese Konzepte in dem von uns entwickelten Animationssystem EMPHAS (**E**asy-to-use, **M**odular, **P**Hysically-based **A**nimation **S**ystem), das auch „Nicht-Experten“ die einfache und vollständig interaktive Erstellung physikalisch basierter Animationen ermöglichen soll. Die modulare Architektur sowie die graphische Oberfläche des Systems zeigt Abbildung 1.

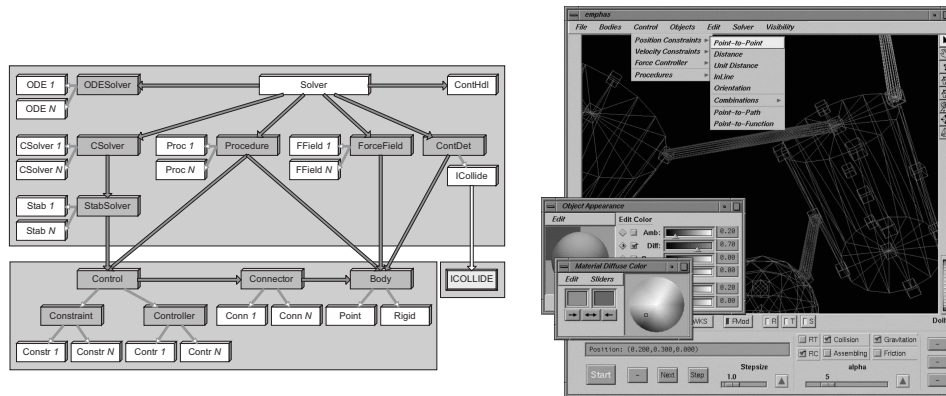


Abbildung 1: Modulare Architektur und graphische Oberfläche von EMPHAS.

**Modulare Konzeption von Körpern und Constraints.** Die LFM ermöglicht in natürlicher Weise die modulare Konzeption von Körpern und Constraints als „black box“-Entitäten (vergl. Punkt 6b in Abschnitt 3). Um die Terme  $\mathbf{J}$  und  $\mathbf{c}$ , die für die Durchführung der LFM benötigt werden (Gleichung (3)), automatisch aus den Constraint-Gleichungen berechnen zu können, ohne dabei die Unabhängigkeit von Körpern und Constraints aufzuheben, wurde in [W<sup>+</sup>90] das sogenannte Konnektoren-Konzept eingeführt. Konnektoren repräsentieren körperfeste Punkte oder andere allgemeine Anknüpfungspunkte und fungieren als Bindeglieder zwischen Körpern und Constraints. Dieses Konzept wurde von uns in zweierlei Hinsicht verallgemeinert (für Details siehe [Wag01]):

- Neben holonomen Constraints werden auch Geschwindigkeits-Constraints  $\mathbf{C}(\mathbf{x}, \dot{\mathbf{x}}, t) = 0$  in das Konnektoren-Konzept integriert.
- Es wird eine explizite Zeitabhängigkeit der Konnektoren zugelassen.

Während Geschwindigkeits-Constraints ein wichtiges Hilfsmittel zur Bewegungssteuerung darstellen, können mit Hilfe explizit zeitabhängiger Konnektoren z.B. auch Punkte, die sich auf einer festgelegten raumzeitlichen Bahn bewegen, als Anknüpfungsobjekte dienen, ohne eine Sonderbehandlung zu erfordern.

**Effiziente Umsetzung der LFM.** Die LFM ermöglicht die schematische und daher leicht automatisierbare Aufstellung und Lösung der Bewegungsgleichungen. In [Bar96] wurde zudem ein einfach umsetzbarer Algorithmus zur effizienten Umsetzung dieser Methode vorgestellt. Dieser Algorithmus wurde von für die robuste Simulation überbestimmter Systeme erweitert, die bei der interaktiven Animationserstellung sehr leicht entstehen können. Hierzu wurde die in [Bar96] vorgeschlagene Unterscheidung von „primären“ und „zusätzlichen“ (*auxiliary*) Constraints ausgenutzt. Primäre Constraints wirken auf genau zwei Körper und bilden keine geschlossenen Schleifen, so daß sie zu keinen überbestimmten Systemen führen können. Wir wenden daher ausschließlich für die Lösung der zusätzlichen Constraints ein Singular Value Decomposition-Verfahren an, das eine robuste Lösung singulärer oder schlecht konditionierter Gleichungssysteme ermöglicht. Auf diese Weise konnte ein optimaler Kompromiß zwischen Effizienz und Robustheit gefunden werden.

**Das „Tool Box“-Konzept.** In EMPHAS wurde ein modulares „Tool Box“-Konzept auf der Grundlage der LFM realisiert. Eine Reihe verschiedener Steuereinheiten in Form von Constraints, Controllern und ereignisbasierten Prozeduren können wie in einem Baukastensystem auf punktförmige und dreidimensionale starre Körper angewandt werden, um komplexe Systeme zu modellieren und zu steuern. In [Wag00] und [Wag01] werden die in EMPHAS realisierten Techniken zur Interaktion und Bewegungssteuerung, die auch eine automatische Behandlung von Kollisionen und Ruhekontakten umfassen, genauer erläutert. Darin wird insbesondere gezeigt, daß die beliebige Kombinierbarkeit der verschiedenen Steuertechniken von den speziellen Eigenschaften der LFM sehr begünstigt wird.

## 5 Schlussfolgerungen

Das Ergebnis unserer vergleichenden Untersuchung zeigt, daß speziell die LFM für interaktive, physikalisch basierte Animationssysteme hervorragend geeignet ist. Mit ihr läßt sich eine Vielzahl von unterschiedlichen Steuertechniken realisieren. Da diese Methode eine stetige Erfüllung der Constraints zuläßt, ist sie besonders für Interaktionen geeignet.

Die Penalty-Methode ist in vielen Anwendungsaspekten zwar gleichwertig, sie ist jedoch bekanntermaßen nur erheblich ineffizienter zu implementieren. Alle weiteren hier diskutierten Verfahren, insbesondere die Reduktionsmethode, weisen zusätzlich erhebliche Einschränkungen hinsichtlich ihrer Verwendbarkeit für ein interaktives Animationssystem auf.

## Literatur

- [A<sup>+</sup>87] W. Armstrong et al. *Near-Real-Time Control of Human Figure Models*. IEEE Computer Graphics and Applications, 7(6):52–61, 1987.
- [A<sup>+</sup>89] B. Arnaldi et al. *Animation Control with Dynamics*. In: Proc. Computer Animation 1989, S. 113–123, 1989.
- [AG85] W. Armstrong und M. Green. *The Dynamics of Articulated Rigid Bodies for Purposes of Animation*. Visual Computer, 1(4):231–240, 1985.
- [Bar96] D. Baraff. *Linear-Time Dynamics using Lagrange Multipliers*. Computer Graphics (Proc. SIGGRAPH '94), 30:137–146, 1996.
- [BB88] R. Barzel und A. Barr. *A Modeling System Based On Dynamic Constraints*. Computer Graphics (Proc. SIGGRAPH '88), 22(4):179–188, 1988.
- [BW97] D. Baraff und A. Witkin. *Partitioned Dynamics*. Technical Report CMU-RI-TR-97-33, Carnegie Mellon University, Pittsburgh, Pennsylvania, march 1997.
- [BW98] D. Baraff und A. Witkin. *Large Steps in Cloth Simulation*. Computer Graphics (Proc. SIGGRAPH '98), 32:43–54, August 1998.
- [C<sup>+</sup>95] R. Cozot et al. *A Unified Model for Physically Based Animation and Simulation*. In: Applied modelling, simulation and optimization, S. 213–216, Anaheim, 1995. IASTED.
- [Fea87] R. Featherstone. *Robot Dynamics Algorithms*. Kluwer Academic Publishers, 1987.
- [GB94] J. Garcia de Jalon und E. Bayo. *Kinematic and dynamic simulation of multibody systems*. Springer-Verlag, New York, 1994.
- [GG94] J. Gascuel und M. Gascuel. *Displacement Constraints for interactive modeling and animation of articulated structures*. Visual Computer, 10(4):191–204, 1994.
- [Gol89] H. Goldstein. *Klassische Mechanik*. AULA-Verlag, Wiesbaden, 1989.
- [IC87] P. Isaacs und M. Cohen. *Controlling Dynamic Simulation with Kinematic Constraints, Behavior Functions and Inverse Dynamics*. Computer Graphics (Proc. SIGGRAPH '87), 21(4):215–224, 1987.
- [IC88] P. Isaacs und M. Cohen. *Mixed Methods for Complex Kinematic Constraints in Dynamic Figure Animation*. Visual Computer, 4(6):296–305, 1988.
- [Met95] D. Metaxas. *Articulated Figure Dynamics, Control and Shape Capture*. In: Siggraph Tutorials 1995, Course 11. ACM, 1995.
- [MW88] M. Moore und J. Wilhelms. *Collision Detection and Response for Computer Animation*. Computer Graphics (Proc. SIGGRAPH '88), 22(4):289–298, 1988.
- [MZ90] M. McKenna und D. Zeltzer. *Dynamic Simulation of Autonomous Legged Locomotion*. Computer Graphics (Proc. SIGGRAPH '90), 24(4):29–38, 1990.
- [NC99] D.D. Nelson und E. Cohen. *Interactive Mechanical Design Variation for Haptics and CAD*. Computer Graphics Forum, 18(3), September 1999.
- [PB88] J.C. Platt und A.H. Barr. *Constraint Methods for Flexible Models*. Computer Graphics (Proc. SIGGRAPH '88), 22(4):279–287, 1988.

- [Pla92] J. Platt. *A Generalization of Dynamic Constraints*. CVGIP - Graphical Models and Image Processing, 54(6):516–525, 1992.
- [SZ90] P. Schröder und D. Zeltzer. *The Virtual Erector Set: Dynamic Simulation with Linear Recursive Constraint Propagation*. Computer Graphics (Proc. 1990 Symposium on Interactive 3D Graphics), 24(2):23–31, 1990.
- [Val93] M. Valasek. *Integration of Recursive Equations of Motion for Multibody Systems with Loops*. In: W. Schiehlen, Hrsg., Advanced multibody system dynamics, S. 439–444. Kluwer, Dordrecht, 1993.
- [VC91] N. Vasilonikolidakis und G.J. Clapworthy. *Inverse Lagrangian dynamics for animating articulated models*. Journal of Visualization and Computer Animation, 2(3):106–113, 1991.
- [vO90] C. van Overveld. *A technique for motion specification in computer animation*. The Visual Computer, 6(2):106–116, 1990.
- [vO91] C. van Overveld. *An Iterative Approach to Dynamic Simulation of 3-D Rigid-Body Motion for Real-time Interactive Computer Animation*. The Visual Computer, 7:29, 1991.
- [W<sup>+</sup>90] A. Witkin et al. *Interactive Dynamics*. Computer Graphics (Proc. SIGGRAPH '90), 24(2):11–21, 1990.
- [Wag00] F. Wagner. *Physikalisch basierte Animation auf der Grundlage der Lagrange-Faktoren-Methode*. In: Proc. SimVis'2000, S. 239–252, Magdeburg, 2000.
- [Wag01] F. Wagner. *Konzepte und Methoden zu allgemeinen, physikalisch basierten Animationssystemen auf der Grundlage der Lagrange-Faktoren-Methode*. Phd, University of Rostock, 2001.
- [WB85] J. Wilhelms und B. Barsky. *Using Dynamic Analysis to Animate Articulated Bodies such as Humans and Robots*. In: Proc. Graphics Interface 1985, S. 97–104, 1985.
- [Wil87] J. Wilhelms. *Using Dynamic Analysis for Realistic Animation of Articulated Bodies*. IEEE Computer Graphics and Applications, 7(6):12–27, 1987.
- [Wil88] J. Wilhelms. *Dynamics for Computer Graphics: A Tutorial*. Computing Systems, 1(1):63–94, 1988.
- [WW90] A. Witkin und W. Welsh. *Fast Animation and Control of Nonrigid Structures*. Computer Graphics (Proc. SIGGRAPH '90), 24(4):243–252, 1990.