

# Zur Nutzung von CAS bei mathematischen Anwendungsprojekten

Burkhard Alpers  
CAS-Maple-Treffen 2007

# Überblick

- Rahmenbedingungen
- Generelle Nutzung von CAS
- Probleme und spezielle Nutzungserfahrungen, Beispiele
- Einstellungen
- Zusammenfassung

# Rahmenbedingungen

- Semester 1+2: Konventionelle Mathematikausbildung
- Semester 3: Numerik vorgezogen, ca. 9 Wochen für Projekte
- Gruppen mit 3-4 Studenten, Zeit: 20h pro Student
- Anwendungsproblemstellung häufig aus den Bereichen Bewegungsdesign, Mechanismenanalyse, Geometrienachbildung
- Bearbeitung des mathematischen Teils mit Maple
- Vorstellung von Maple in Semestern 1+2
- Mathematik III-Skript in Maple, Auffrischungsworksheets, Maple-Tutor.

# Generelle CAS-Nutzung

- Prä-CAS-Aktivitäten: Modellierung (eigene Modelle und Verstehen existierender Modelle)
- Modelleingabe bzw. –umsetzung im CAS
  - Direkte Eingabe von Modellobjekten
  - Erstellung mit Prozedur
  - Berechnung von Modellobjekten
  - Modell=Algorithmus
- Modellberechnungen (Möglichkeiten, Alternativen)
- Interpretation der Ergebnisse
- Experimentieren

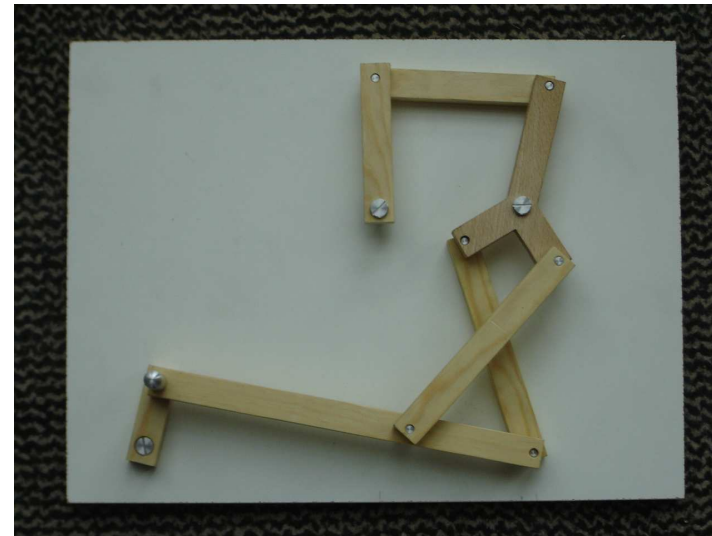
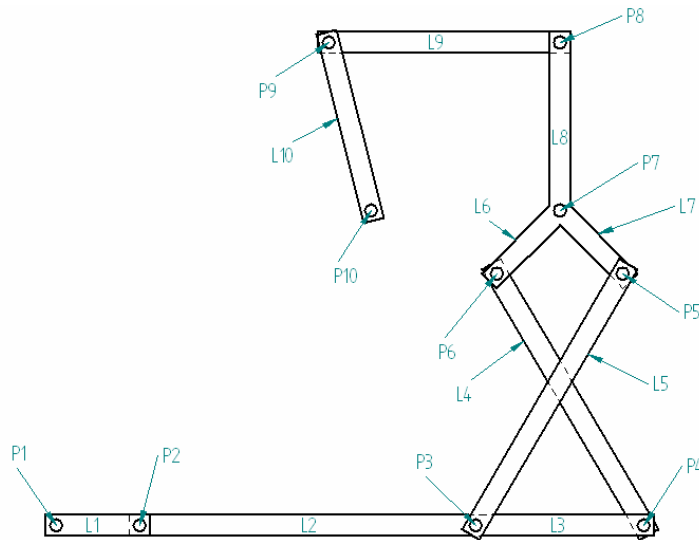
# Probleme mit CAS-Spezifika

- Genaue Beachtung der Datentypen (z.B. Unterschied zwischen  $f$  und  $f(x)$ )
- Auswertereihenfolge: z.B.  
> `abl:=x->diff(f(x),x);`
- undefiniertheit bei Spline-Ableitungen
- Und Ähnliches ...

# Scheitern von Maple-Kommandos

- Berechnungsroutine liefert keine Lösung, z.B. solve oder fsolve
- Keine einfache Delegation der Rechenarbeit an das CAS
- Mathematische Qualifikation zur Ortung und Umgehung der Schwierigkeiten erforderlich
- Problem: Umgang mit Prozeduren, deren innere Arbeitsweise unbekannt ist
- Beispiel: Scheibenwischergetriebe

# Beispiel: Scheibenwischergetriebe



- „naive“ Modellierung: pro Länge eine Gleichung
- Keine Lösung mit solve und fsolve
- Überprüfung des Modells durch Einsetzen bekannter Lösung
- Aufteilung des Systems, Reduktion des Unbekanntenzahl

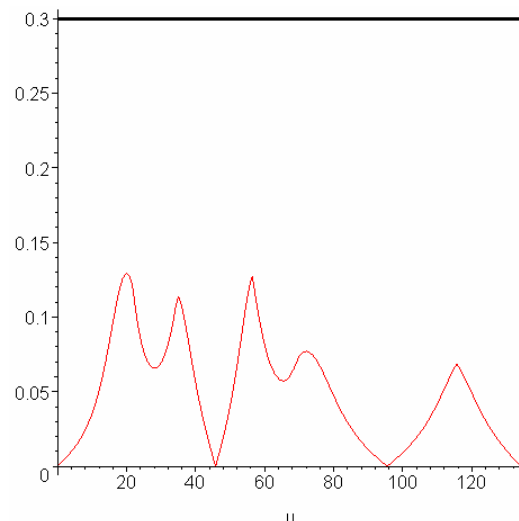
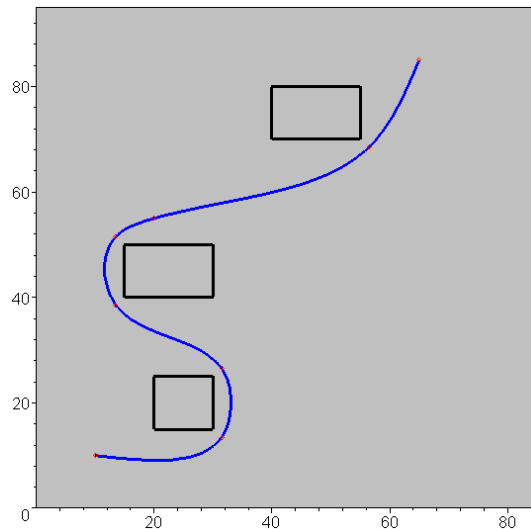
# Nutzen und Grenzen der Symbolik

- Einfacheres Arbeiten, wenn symbolische Vorgehensweise erfolgreich ist (z.B. direkte funktionale Abhängigkeit, direkt plotbar)
- Direkte Umsetzung von Papierroutinen
- Wiedererkennung math. Objekte im Output
- Erkennung von Fehlern (z.B.  $f:=x^2$ ,  $f(x)=x(x)^2$ )
- Grenzen bei nicht-linearen Modellen, Integralen (Bogenlängen, Umrechnung  $v(s) \rightarrow v(t)$ )
- Grenzen bei mehrdeutigen Lösungen (RootOf) und bei Lösungen mit begrenztem Definitionsbereich
- Unlesbarer symbolischer Output

# Von Rechenschritten zur Experimentierumgebung

- Anfängliche Nutzung: Rechnung mit konkreten Werten; Ziel: erste Ergebnisse
- Erkennen der Grenzen beim Experimentieren bzw. bei Anwendung auf andere Inputdaten
- Einführung von Symbolen für Inputdaten
- Automatische Anpassungen an flexible Inputdaten
- Optimieren durch Experimentieren
- CAS als „Autorensystem“ für die Erstellung einer eigenen Berechnungs- und Experimentierumgebung
- Beispiel: Bahn um Hindernisse

# Beispiel: Bahn um Hindernisse



- Bahn ist grob vorgeschrieben
- Länge möglichst gering, Krümmung  $< 0.3$
- Anfängliches Setzen von Punkten
- Berechnung der Splinekurve
- Ermittlung der Länge/Krümmung
- Experimentieren

# Nutzungstiefe und Differenzierung

- Große Unterschiede in der Nutzung:
  - Nutzung nur elementarer Möglichkeiten
  - Kopieren und Modifizieren bei Mehrfachberechnungen
  - Kopieren von Ergebnissen in Dateien zur weiteren Bearbeitung in anderen Programmen
  - Einbau von Schleifen
  - Nutzung der Programmierung von Prozeduren
  - Unterschiedliche Grade der Flexibilisierung
  - Nutzung von strukturierter Darstellung in Maple mit guter Dokumentation
- Jeder Student erzielt ein Resultat
- Begabte Studenten sind meistens animiert, die Möglichkeiten auszureizen.

# Einstellungen

- Beeindruckt von der Mächtigkeit
- Skepsis bzgl. der Bedienung bei schwächeren Studenten
- Zeitnahe Betreuung zur Vermeidung von Frust erforderlich
- Positive Einstellung bei begabten Studenten, Erkennen der Erweiterung der eigenen Möglichkeiten, „Nutzungsanreiz“
- Nutzung auch in späteren Semestern bei begabten Studenten
- Generelle Erkenntnis: CAS ist kein Allheilmittel

# Zusammenfassung

- Projekte ohne CAS-Umgebung nicht denkbar
- CAS ist gute Umgebung zur Erstellung von Experimentierumgebungen
- Bloße Delegation der Rechenarbeit ist nicht ausreichend, Problemlösung nur mit mathematischem und CAS-Verständnis

A fool  
with a tool  
is still a fool.