



INSTITUT FÜR ANGEWANDTE GEOMETRIE
Univ.-Prof. Dr. Bert Jüttler

Skriptum zur Vorlesung
Mathematische Grundlagen des CAD
Sommersemester 2007

Vortragender

Univ.-Prof. Dr. B. Jüttler (Vorlesung)
Johannes Kepler Universität Linz
Institut für Angewandte Geometrie

Büro: Kopfgebäude, Zi. 517, Tel. (0732) 2468 - 9178
Email: bert.juettler@jku.at
<http://www.ag.jku.at>

Sekretariat: Frau M. Bayer, Zi. 518, Tel. 2468-9162

Organisatorisches

- Termine: 7.3.2007 Vorbesprechung,
Freitag 27.4.2007 8:30 - 14:30 K 123 A
11.5., 25.5., 1.6.: 8:30 - 14:30 T 212

- Literatur:

– Kunwoo Lee: Principles of CAD/CAM/CAE Systems, Addison-Wesley

“In writing the book, my goal was to explain fundamental concepts with an appropriate number of figures and examples without getting bogged down in too many details. I have seen several textbooks fail to

get their main points across by trying to explain too many details. This overreliance on detail also makes a book so thick that it scares students away from using it effectively.”

- P. Benkő, R. Martin, T. Varady: Algorithms for reverse engineering boundary representation models, *Computer-Aided Design* 33 (2001), 839–851.

- Ergänzende Literatur:

- J. Hoschek und D. Lasser: Grundlagen der geometrischen Datenverarbeitung, Teubner, Stuttgart (auch in englischer Sprache: AK Peters).
- G. Farin: Kurven und Flächen im Computer Aided Geometric Design Vieweg, Braunschweig (auch in englischer Sprache: Academic Press);
- J. Gallier: Curves and surfaces in geometric modeling, Kaufmann, San Francisco.

Ziele, geplanter Inhalt

Die Vorlesung bietet Einführung in die heute verfügbare CAD-Technologie. Unter anderem sollen folgende Themen behandelt werden:

- Einleitung: CAD, CAM und CAE (–)
- Geometrisches Modellieren
- Grundlegende Beschreibungen für Kurven und Flächen
- Reverse Engineering

1 CAD, CAM und CAE

Diese Abkürzungen stehen für Computer Aided Design, Computer Aided Manufacturing und Computer Aided Engineering. Während diese Prozesse in der Vergangenheit häufig isoliert betrachtet wurden, sind sie heute in zunehmenden Maße miteinander verzahnt.

1.1 Fallstudie: Entwicklung eines Teiles eines Mobiltelefons

[Lee, Abschnitt 1.3 und 1.4]

Bei der Produktentwicklung wirken die verschiedenen CAx-Technologien eng zusammen:

- Styling und konzeptionelles Design (Entwurf der Geometrie, weitgehend nach ästhetischen Gesichtspunkte): CAD
- Analyse des Designs (FEM-basierte Berechnung der Festigkeit usw.): CAE
- Erzeugung eines physischer Prototyps (zum Beispiel zur Simulation des Zusammenbaus): CAM, “rapid prototyping”

- Erstellung von technischen Zeichnungen (soweit noch notwendig)
- Entwurf der Spritzgußform: CAD (häufig ein speziell angepaßtes System), Hinzufügen von für den Gießprozeß notwendigen Kanälen, usw.
- Numerische Simulation des Gießvorgangs: CAE
- Herstellung der Gußform: CAM (M=Manufacturing) (Erzeugung der Bahnen für die Steuerung von NC-Fräsmaschinen).
- Produktherstellung (CAM).

Alle Prozesse laufen zum Teil parallel ab bzw. müssen miteinander kommunizieren. Im Idealfall greifen alle diese Bearbeitungsschritte auf eine gemeinsame Datenbasis zu. Das dieser Basis zugrundeliegende Geometrie-Modell muß natürlich den verschiedenen Ansprüchen der einzelnen Schritte gerecht werden.

1.2 CAD, CAM und CAE-Software

[Lee, Abschnitt 2.3]

Die verschiedenen Teilschritte des Entwurfsprozesses werden durch verschiedene Softwarekomponenten realisiert.

- CAD Software erlaubt es dem Designer, ein geometrisches Objekt interaktiv am Monitor zu entwerfen und zu manipulieren. Man unterscheidet zwischen (weitgehend) zweidimensional arbeitenden Zeichenprogramme (Computer Aided Drafting) einerseits und echt dreidimensional arbeitenden Systemen A für die Modellierung von Volumina ("Solid Modelling". Letztere finden in zunehmenden Maße Verbreitung.
- CAM Software dient zur Vorbereitung und Vereinfachung des Herstellungsprozesses. Hierzu zählt insbesondere Software zur Konstruktiven von Gußformen, zur Berechnung von Fräsbahnen, zum Entwurf von Tiefzieh-Vorgängen u.v.a.m.
- CAE Programme dienen zur Analyse von entworfenen Produkten, zu deren numerischer Simulation und Optimierung. Diesen Programme liegt in der Regel eine numerische Simulation mittels finiter Elemente zugrunde. Die Umwandlung einer konstruierten Geometrie in ein FEM-Modell ist nichttrivial und führt immer wieder zu Problemen.

Neben spezialisierter Software für die einzelnen Schritte existieren integrierte Systeme (z.B. CATIA), die die verschiedenen Funktionalitäten integrieren und bündeln. Neben diesem Trend zu immer größeren integrierten Systemen ist eine parallele Entwicklung hin zu Windows-basierten spezialisierten Programmen zu beobachten.

2 Geometrisches Modellieren

[Lee, Kapitel 5] CAD Software soll den Designer beim Entwurf geometrischer Objekte unterstützen. Wir beschäftigen uns hier ausschließlich mit dreidimensionalen Objekten.

2.1 “Wireframe modelling”

Bei diesen Systemen wird das Objekt durch charakteristische Linien und Punkte modelliert. Dabei entsteht ein „Drahtgittermodell“ des konstruierten Objekts. Das Modell besteht aus

- der Liste der Kurvengleichungen,
- der Koordinaten der Eckpunkte, und
- der Zusammenhangsinformation für Ecken und Kurven.

Bemerkung 2.1 Diese Drahtgittermodelle waren besonders in der Frühzeit des CAD populär, da sie relativ leicht zu bedienen sind. Andererseits führt diese einfache Darstellung relativ schnell zu Mehrdeutigkeiten, da im Modell keinerlei Informationen über die Flächen sowie über Innen- und Außenbereiche enthalten ist. Heute sind diese Systeme weitgehend vom Markt verschwunden.

2.2 “Surface Modelling”

Bei dieser Art von CAD Systemen werden – zusätzlich zu den Informationen über Ecken und Kanten – noch mathematische Beschreibungen der Flächen zwischen den Kanten verwendet (implizit oder parametrisch). Das Modell besteht damit aus

- der Liste der Flächengleichungen,
- der Liste der Kurvengleichungen,
- der Koordinaten der Eckpunkte, und
- der Zusammenhangsinformation für Ecken, Kurven und Flächen.

Insbesondere Zusammenhangsinformationen für Flächen (z.B., zwei Flächen stoßen entlang einer Kante stetig oder C^1 (glatt) zusammen) sind für die weitere Behandlung des Modells (CAM) extrem wichtig. Oft ist diese Information jedoch nicht in der Datenbasis enthalten (zum Beispiel werden von einem Zylinder nur Achse und Radius gespeichert) und muß erst rekonstruiert werden.

Typischerweise stehen drei Verfahren zur Erzeugung der Flächen zur Verfügung:

- Interpolation von Punkten,
- Interpolation von Kurvennetzen, sowie

- durch Verschiebung oder Drehung einer Profilkurve.

Flächenmodelle werden einerseits verwendet, um die Ästhetik eines Objektes visuell zu bewerten (etwa bei Karosserien im Automobilbau), und andererseits, um NC-Programme zur Flächenherstellung zu erzeugen. [Lee Abb. 5.2 und 5.3]

2.3 “Solid Modelling”

Diese Systeme stellen gegenwärtig die am weitesten fortgeschrittene CAD-Technologie dar. Mit diesen Systemen werden Volumina, die sog. Solids, modelliert. Eine Menge von Flächen oder charakteristischen Kurven und Punkte kann nur dann modelliert werden, wenn diese den Rand eines solchen Volumens bilden. Die mathematische Beschreibung eines Solids enthält Informationen darüber, welche Punkte sich innerhalb und außerhalb des Körpers befinden.

Als ein Vorteil dieses Vorgehens können Anwendungsprogramme diese Volumeninformation nutzen. Dies ist beispielsweise bei der Berechnung von Bahnkurven für Fräsmaschinen (und der Simulation des Fräsprozesses), oder bei der Erzeugung von finiten Elementen von Vorteil.

Andererseits ist die Erzeugung und Beschreibung eines Solids relativ aufwendig und nicht immer besonders intuitiv. Deshalb werden meist einfache Erzeugungs- und Modellierungsfunktionen bereitgestellt, die dem Anwender die Feinheiten der Beschreibung abnehmen sollen.

2.3.1 Verfügbare Funktionen

Diese Funktionen lassen sich grob in fünf Gruppen einteilen.

1. Grundobjekte und Boolesche Operationen. Typische Grundobjekte sind Quader, Keile, Kreiskegel und -zylinder, Kugeln, und Tori. Boolesche Operationen sind Vereinigung, Durchschnitt, und Differenz. Mit diesen Operationen lassen sich aus den Grundobjekten bereits recht komplizierte Geometrien erzeugen.

Bemerkung 2.2 Bei der Implementation und Anwendung dieser Operationen ist Vorsicht geboten, da unter Umständen keine gültigen Solids entstehen. Beispielsweise ist der Schnitt zweier Würfel, die eine Facette gemeinsam haben, nicht wieder ein Solid! Derartige Situationen müssen ausgeschlossen werden.

2. Bewegte Flächen. Wird ein ebener Bereich verschoben oder um eine Achse gedreht, so spricht man von *sweeping* (im Falle einer Drehung auch von *swinging*). Die dabei entstehenden Solids sind von Translations- bzw. Drehflächen begrenzt (translational sweep, rotational sweep). Manchmal sind auch allgemeinere Bewegungen zugelassen, dann erhält man sog. Bewegflächen.

Beim *skinning* wird aus mehreren gegebenen Profilkurven durch Interpolation eine Fläche erzeugt. Dies ist ähnlich zur mechanischen Konstruktion einer Struktur durch Bespannen

von einzelnen frames mit einem stoffartigen elastischen Material. Durch Hinzunahme der beiden Endflächen entsteht ein Solid, andernfalls ein Flächenmodell.

3. Modifikationsfunktionen. Beim *rounding* oder *blending* wird eine Kante oder Ecke eines Modells durch eine glatte gekrümmte Fläche ersetzt, die glatt an die benachbarten Flächen anschließt. Falls dazu Material hinzugefügt werden muß, so spricht man von *filleting*. Die Konstruktion von solchen Blendflächen (insbesondere für Ecken) ist ein nicht-triviales Problem. Insbesondere muß darauf geachtet werden, daß verschiedene Blend-Operationen miteinander vertauschbar sind.

Bemerkung 2.3 Eine weit verbreitete Konstruktion für Blendflächen sind sog. rolling ball blends, die durch das Abrollen einer Kugel entstehen. Die Hüllfläche der bewegten Kugel dient als Blendfläche (prozedurale Definition).

Beim *lifting* wird ein Teil oder eine Facette eines Solids in eine Richtung verschoben und das Objekt entsprechend modifiziert. Dabei können unter Umständen ungültige Solids entstehen (Lee Fig. 5.15)

4. Boundary modelling. Hier werden verschiedene Funktionen zur Erzeugung und Modifikation der Randpunkte, Kanten und Facetten bereitgestellt. Damit kann beispielsweise ein Objekt aus Facetten zusammengesetzt werden, die ihrerseits wieder durch Kanten und Eckpunkte erzeugt worden sind (Lee Fig. 5.16). Anders als bei Flächenmodellierungssystemen muß beim Solid Modelling stets ein gültiges Solid entstehen.

5. “Feature-based modeling”. Diese Funktionalität ermöglicht es dem Designer, vordefinierte Grundobjekte zur Modifikation eines Solids zu verwenden. Typische Features sind Löcher, Schlitze, Taschen, gebrochene Kanten u.ä. Diesen Features sind jeweils Herstellungsgänge wie Bohren, Fräsen, usw. zugeordnet. Die Verwendung von Features erleichtert damit die Planung des Herstellungsprozesses, da bereits beim Design entsprechende Information (implizit) mit abgelegt wird.

Beim *parametrischen Modellieren* verwendet der Designer geometrische Nebenbedingungen (constraints) und Abmessungen zur Konstruktion (vorwiegend für zweidimensionale Objekte). Die Nebenbedingungen beschreiben Relationen zwischen den Objekten, wie Parallelität, Kontakt zwischen gekrümmter Kante und gerader Nachbarkante, usw. Die Abmessungen sind sowohl Abmessungen zwischen den Objekten wie auch Beziehungen zwischen diesen. Die gegebenen Informationen liefern ein Gleichungssystem, das zur Ermittlung der Geometrie nach den freien Parametern aufgelöst wird. Durch Variieren der Parameter und Nebenbedingungen lassen sich so rasch verschiedene Modelle (Design-Alternativen) erzeugen.

2.3.2 Datenstrukturen

Zur Beschreibung von Solids stehen drei Datenstrukturen zur Verfügung: CSG Modell, B-Rep-Modell und Zerlegungsmodell.

1. CSG–Modell. CSG ist die Abkürzung für Constructive Solid Geometry. Beim CSG–Modell wird in einer Baumstruktur die Geschichte der Konstruktion des Objektes (Erzeugung von Grundobjekten und Anwendung Boolescher Operationen) gespeichert. Der entstehende Baum heißt CSG–tree.

Das CSG–Modell besitzt folgende Vorteile:

- Es verfügt über eine einfache und kompakte Datenstruktur
- Das entstehende geometrische Objekt ist stets ein gültiges Solid.
- Die CSG–Darstellung kann stets in ein B–Rep Modell konvertiert werden.
- Parametrisches Modellieren ist relativ leicht durch Modifikation der Abmessungen der Grundobjekte realisierbar.

Daneben existieren auch zwei wesentliche Nachteile:

- Die Konstruktion ist durch die Verwendung Boolescher Operationen beschränkt. Insbesondere können lokale Operationen wie *lifting* oder *rounding* nicht oder nur schwer verwendet werden.
- Die Ermittlung der Informationen über Randflächen, deren Kanten (*edges*) und Eckpunkt (*vertices*) kann relativ aufwendig sein. Diese Informationen werden aber für zahlreiche Anwendungen (angefangen bei der Visualisierung bis hin zur Erzeugung von Fräsbahnen) benötigt.

Aufgrund dieser Nachteile wird die CSG–Darstellung in der Regel nicht ausschließlich / selbständig verwendet, sondern in der Regel von einem B–Rep–Modell begleitet (hybride Darstellung).

2. B–Rep–Modell. B–Rep steht für Boundary representation. Die Grundelemente dieser Darstellung sind Ecken (*vertices*), Kanten (*edges*) und Facetten (*faces*). Im B–Rep–Modell werden die Daten dieser Grundobjekte sowie die Inzidenz Beziehungen zwischen ihnen gespeichert.

Realisierung durch Tabellen. Im einfachsten Modell werden dafür einfach drei Tabellen benötigt. Die Tabelle der Eckpunkte enthält deren Koordinaten. Die Tabelle der Kanten enthält jeweils die entsprechenden Eckpunkte. In die Tabelle der Facetten sind schließlich die anliegenden Kanten eingetragen. Dabei werden die Ecken jeder Facette jeweils entgegen dem Uhrzeigersinn durchlaufen. Durch diese Konvention wird die Information über das Innere des Solids in das Modell integriert. Dieses Tabellenmodell verfügt jedoch über einige Nachteile und wird daher nicht verwendet:

- Das Tabellenmodell ist vor allem für Polyeder geeignet. Im Falle allgemeiner Solids müssen auch die Gleichungen der Kanten und Facetten in das Modell integriert werden.

- Facetten mit inneren und äußeren Rändern (beispielsweise mit einem Loch im Inneren) können nicht ohne weiteres in der Facettenliste gespeichert werden. Ein möglicher Ausweg ist die Verwendung von „Brückenkanten“.
- Die Anzahl der Kanten je Facette kann variieren. Daher muß mit einer Tabelle variabler Größe gearbeitet werden.
- Die Ermittlung der Zusammenhangsinformationen aus den drei Tabellen ist unter Umständen aufwendig. Um etwa die Facetten zu ermitteln, die eine gegebene Kante gemeinsam haben, muß die gesamte Facetten-Tabelle durchsucht werden. Ähnlich aufwendig ist die Ermittlung aller Kanten, die eine Ecke gemeinsam haben.

Anstelle von Tabellen wird daher in der Regel mit zeigerbasierten (verpointerten) Datenstrukturen gearbeitet.

Eine naheliegende Idee ist die Verwendung einer zeigerbasierten Datenstruktur mit Kanten, Ecken und Facetten. Die zu einer Facette gehörenden Kanten könnten dabei jeweils im Uhrzeigersinn auf jeweils die ihnen folgende Kante zeigen. Dieses Vorgehen ist jedoch nicht ausreichend, da jede Kante in der Regel zu zwei Facetten gehört. Man verwendet daher häufig die sog. Halbkantendarstellung:

Realisierung durch Halbkanten (“half edges”). Wie bereits gesagt, wird das Solid im wesentlichen als Liste seiner Facetten dargestellt, wobei jede Facette durch ihre Kanten beschrieben wird. Diese Datenstruktur besteht aus folgenden Datentypen (* steht für „Zeiger auf“), die jeweils doppelt verkettete Listen bilden:

- solid (identifizier, *faces, *edges, *vertices, *next solid, *previous solid)
- face (identifizier, *solid, *outer loop, *list of loops, face equation, *next face, *previous face)
- loop (*ring of halfedges, *face, *next loop, *previous loop)
- edge (*right halfedge, *left halfedge, *next edge, *previous edge)
- halfedge (*parent edge, *starting vertex, *loop, *next halfedge, *previous halfedge).
- vertex (identifizier, *one of the halfedges, vertex coordinates, *next vertex, *previous vertex)

Die Loops werden eingeführt, um auch innere Löcher beschreiben zu können, ohne dafür redundante Brückenkanten einführen zu müssen. Die zu einer Facette gehörenden Halbkanten werden jeweils im Uhrzeigersinn durchlaufen.

Beispiel 2.4 Frage: Wie findet man alle Kanten, die in einer Ecke zusammenlaufen?

Bemerkung 2.5 Für die Kanten wird (hier) keine Gleichung gespeichert, diese ergeben sich als Schnitt der benachbarten Facetten. Dies kann jedoch problematisch sein, etwa bei tangentialen Übergängen zwischen Nachbarfacetten. In der Praxis verwendet man häufig Facetten, die als sog. getrimmte Flächen beschrieben sind. Diese stoßen dann nur näherungsweise in den Kanten zusammen.

Bei dieser Datenstruktur spielen die Facetten die wesentliche Rolle. Eine andere mögliche Datenstruktur, die um die Kanten organisiert ist, ist folgende:

Realisierung durch “winged edges”. Bei dieser Datenstruktur werden für jede Kante die beiden benachbarten Facetten, die 4 benachbarten Kanten und die Eckpunkte gespeichert. Die Zusammenhangsinformation der Ecken, Kanten und Facetten ist etwas direkter aus dieser Datenstruktur ersichtlich, da die Halbkanten vermieden werden. Diese Datenstruktur besteht aus folgenden Datentypen (* steht wieder für „Zeiger auf“), die jeweils einfach verkettete Listen bilden:

- solid (identifizier, *next solid, *faces)
- face (identifizier, *solid, *outer loop, *list of loops, face equation, *next face)
- loop (identifizier, *face, *next loop, *edge, loop type)
- edge (*left loop, *right loop, *ne edge, *nw edge, *se edge, *sw edge, *n vertex, *s vertex, *geometry data)
- vertex (identifizier, *edge, vertex coordinates)

3. Zerlegungsmodell . Hier unterscheidet man zwischen Voxels, Octrees und Zell-Darstellung.

Voxel-Darstellung. Bei diesem Modell wird ein dreidimensionales Würfelraster über das zu beschreibende Solid gelegt und die Würfel danach klassifiziert, ob sie zum Inneren oder zum Äußeren gehören. Damit lassen sich beliebige Solids mit hinreichender Genauigkeit beschreiben. Weiter sind Eigenschaften wie die Masse des Solids leicht zugänglich, und die Datenstruktur beschreibt auch den das Solid umgebenden Raum. Nachteilig ist die schnelle Zunahme des Datenvolumens bei höherer Genauigkeit und natürlich die Tatsache, daß das Voxelmodell stets nur eine Approximation des Solids darstellt.

Octree-Darstellung. Diese ist ähnlich zur Voxel-Darstellung, jedoch wird eine hierarchische Datenstruktur aus ineinander geschachtelten Würfeln zur Beschreibung verwendet. Damit wird der Speicherbedarf verringert.

Beide Darstellungen (Voxel und Octree) werden in der Regel nicht allein, sondern als Ergänzung eines exakten Modells verwendet.

Zell–Darstellung. Ähnlich wie bei der Voxel–Darstellung wird auch hier das Solid in kleinere Einheiten (Zellen) zerlegt. Dabei sind jetzt jedoch allgemeinere Zellen zugelassen. Ein typisches Beispiel für eine solche Zellenzerlegung sind die finiten Elemente zur numerischen Simulation.

2.3.3 Euler–Formel und Euler–Operatoren

Eine exakte Beschreibung eines Solids besteht aus Facetten, Kanten und Ecken. Zwischen der Anzahl dieser Grundobjekte besteht ein Zusammenhang, der durch die Euler–Poincaré–Formel beschrieben wird:

$$v - e + f - h = 2(s - p) \quad (1)$$

Dabei sind v , e , und f die Anzahl der Ecken (“vertices”), Kanten (“edges”) und Facetten (“faces”), h ist die Anzahl der Löcher in den Facetten (“holes”), s die Anzahl der “shells” (also der Zusammenhangskomponenten des Randes) und p die Anzahl der Löcher im Solid (“passages”).

Beispiel 2.6 Bei einem Würfel mit einem quadratischen Loch gilt $v = 16$, $f = 10$, $h = 2$, $s = 1$ und $p = 1$.

Die Anzahl der Facetten, Kanten und Ecken eines Solids können daher nicht unabhängig voneinander verändert werden. Zur Modifikation eines Solids werden daher sog. Euler–Operatoren implementiert, die – für den Anwender nicht sichtbar – stets Gruppen von Objekten verändern. Durch die Verwendung dieser Operatoren ist sichergestellt, daß das Solid stets der Euler–Poincaré Formel genügt.

Beispiel 2.7 Wird beispielsweise eine Facette eines Würfels in zwei Dreiecke erlegt, so ist dabei der Operator „Erzeuge eine Kante und einen Loop“ anzuwenden.

2.3.4 Ermittlung volumetrischer Eigenschaften

Für ein Solid in B–Rep–Darstellung lassen sich volumetrische Eigenschaften (Masse, Momente, usw.) ermitteln, in dem die auftretenden Volumenintegrale mit Hilfe des Gaußschen Integralsatzes in Oberflächenintegrale umgeformt werden.

2.4 Systeme zur Modellierung allgemeinerer Objekte

Durch Flächenmodellierungs–Systeme lassen sich nur solche: Objekte modellieren, die homöomorph zu einer Kreisscheibe sind. Beim Solid Modelling wird dies noch weiter eingeschränkt: es können nur Flächen modelliert werden, die Rand eines Volumens sind. Dies garantiert zwar einerseits eine Korrektheit des Modells, ist aber andererseits für viele Anwendungen zu einschränkend. Insbesondere während der Phase des konzeptionellen Designs ist es für den Anwender oft unbequem, stets ein korrektes Modell entwerfen zu müssen. Hier ist wünschenswert auch allgemeinere Objekte zugelassen, bei denen unter Umständen auch von der Dicke abstrahiert wird. Dies ist in manchen System realisiert. Dafür sind allgemeinere Datenstrukturen und Euler–Operatoren notwendig (siehe Lee: Anhang E).