

Krümmungsapproximation auf 3D-Modellen

Mathematische Grundlagen und Approximation

Christian Cyrus
Matrikelnummer 157707

Fakultät für Informatik
Otto-von-Guericke-Universität Magdeburg

Diese Ausarbeitung enthält mathematische Grundlagen zur Krümmungsberechnung und einige Methoden zur Approximation der Krümmung von 3D-Modellen. Beginnend mit einer Motivation zum Thema Krümmung und deren Approximation, werden zunächst die mathematischen Grundlagen wie Ableitungen eindimensionaler und zweidimensionaler Funktionen, sowie die Berechnung zusätzlicher Krümmungsparameter erläutert. Im Anschluss werden einige Methoden zur Approximation vorgestellt, welche auf drei verschiedene Weisen eine Schätzung der Krümmungsparameter beschreiben. Darauf folgt ein Vergleich der Methoden in Hinblick auf Robustheit, Genauigkeit und Skalierbarkeit. Zum Ende erfolgt eine Zusammenfassung der Erkenntnisse.

Inhaltsverzeichnis

1	Einleitung	3
2	Mathematische Grundlagen	3
3	Methoden	4
3.1	Surface Fitting	5
3.2	Curve Fitting	6
3.3	Tensor Averaging	7
4	Vergleich	9
5	Zusammenfassung	10
	Literaturverzeichnis	10

1 Einleitung

In der Computergrafik ist die Krümmung eine sehr wichtige Komponente in vielen Bereichen, z.B. zur Glättung, Segmentierung, Visualisierung oder *Remeshing* von 3D-Modellen, sowie dem *Non-Photorealistic* Rendering. Direkte Beispiele sind in anderen Arbeiten dieses Seminars nachzulesen. Da die 3D-Modelle in Form von Polygonnetzen vorliegen, es also nur einzelne Stützstellen mit entsprechenden Verbindungen gibt, sind nur zwei verschiedene Krümmungswerte vorhanden. Zum einen ist die Krümmung auf den Teilflächen des Modells 0 und an den Kanten und Ecken ∞ . Da diese Werte nicht sehr sinnvoll für die Weiterverarbeitung sind und auch nicht den Werten des realen Objektes entsprechen, müssen Krümmungsinformationen mit Hilfe der Umgebung eines Punktes approximiert werden.

2 Mathematische Grundlagen

In diesem Abschnitt soll es nun um die mathematischen Grundlagen zur Krümmung gehen. Mathematisch definiert ist die Krümmung (κ) als Änderung der Richtung einer Funktion. Da die Richtung die erste Ableitung einer Funktion ist, ist es nötig diese nochmals abzuleiten. Es gilt also

$$\kappa = \frac{\partial^2}{\partial x^2}.$$

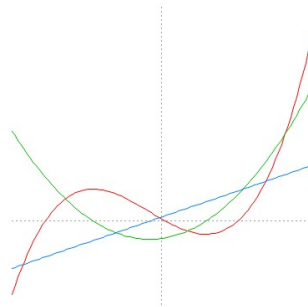


Abbildung 1: Eine kubische Funktion mit ihrer 1. Ableitung (Richtung) und 2. Ableitung (Krümmung)

Ein Beispiel für die Krümmung einer Funktion, welche aus dem täglichen Leben bekannt ist, wäre das Fahren mit einem Auto. Hierbei ist die Stammfunktion die momentan zurückgelegte Strecke, die Richtung die gegenwärtige Geschwindigkeit und die Krümmung schließlich die Beschleunigung, die zum entsprechenden Zeitpunkt auf das Fahrzeug wirkt.

Ein paar Beispiele von Krümmungen sieht man gut in Abbildung 2(a) auf der nächsten Seite.

Eine alternative Interpretation der Krümmung ist der Krümmungskreis, siehe Abbildung 2(b) auf der nächsten Seite. Es wird versucht, einen Kreis an die Funktion

anzuschmiegen. Der Radius dieses Kreises ist indirekt proportional zur berechneten Krümmung. Aufgrund dieser Betrachtung sieht man, dass ein Knick in der Funktion eine Krümmung von ∞ haben muss, weil der Kreis den Durchmesser 0 hat. Genauso ist es umgekehrt bei geraden Stücken, der Radius ist ∞ und die Krümmung dementsprechend 0.

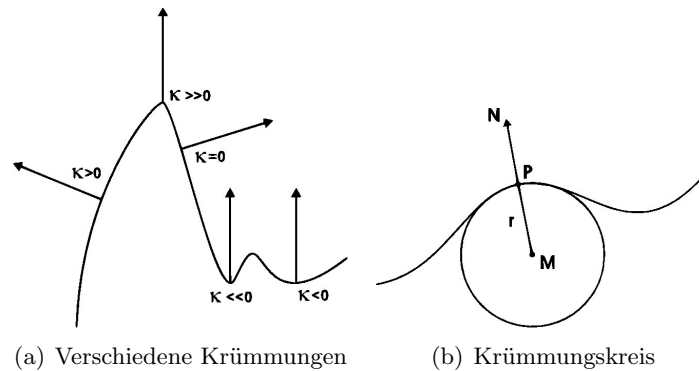


Abbildung 2: Krümmung entlang einer Kurve und Krümmungskreis [Sch05]

Dies waren die Grundlagen für eine eindimensionale Funktion. Bei unseren 3D-Modellen benötigen wir aber die Krümmung der Oberfläche an einem bestimmten Punkt. Die Krümmung ist auch hier die zweite Ableitung, wobei diese in eine Matrix resultiert. Diese Matrix wird *Weingartenmatrix* (W) [dC93] genannt.

Die zweite Ableitung einer zweidimensionalen Funktion ist somit

$$W = \begin{bmatrix} \frac{\partial^2}{\partial x^2} & \frac{\partial^2}{\partial y \partial x} \\ \frac{\partial^2}{\partial x \partial y} & \frac{\partial^2}{\partial y^2} \end{bmatrix}.$$

Die Eigenwerte dieser Matrix werden Hauptkrümmungen (κ_1, κ_2) genannt. Diese Hauptkrümmungen verlaufen entlang der Eigenvektoren der *Weingartenmatrix*. Man nennt diese Vektoren dementsprechend Hauptkrümmungsrichtungen ($t_{\kappa_1}, t_{\kappa_2}$).

Mit Hilfe der beiden Hauptkrümmungen lassen sich nun noch weitere Maße, wie die Gaußsche Krümmung $K = \kappa_1 \cdot \kappa_2$ und die Mittlere Krümmung $H = \frac{\kappa_1 + \kappa_2}{2}$, bestimmen. Diese errechneten Maße erlauben eine grobe Kategorisierung der Punkte (siehe Abb. 3 auf der nächsten Seite) auf Grund der Gaußschen Krümmung in elliptische ($K > 0$), hyperbolische ($K < 0$) und parabolische Punkte ($K = 0$).

3 Methoden

In [Sch05] und [Rus04] werden einige Methoden erläutert, welche man grundlegend in drei verschiedene Kategorien gruppieren kann. Auf den folgenden Seiten wird für jede Kategorie ein Grundalgorithmus skizziert und einige Beispiele der Verwendung mit Parametern gegeben.

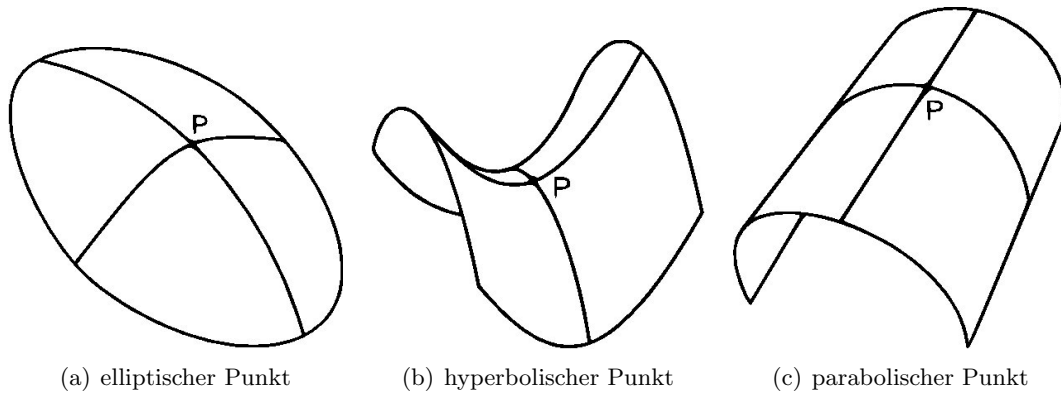


Abbildung 3: Punktkategorien: $K > 0$ elliptischer Punkt, d.h. beide Hauptkrümmungen besitzen gleiches Vorzeichen, $K < 0$ hyperbolischer Punkt, d.h. negative und positive Hauptkrümmung, $K = 0$ parabolischer Punkt mindestens eine Hauptkrümmung ist 0 [Sch05]

3.1 Surface Fitting

Als erste Methode ist das *Surface Fitting* zu nennen. Hierbei wird versucht eine Passfläche für den entsprechenden Punkt und dessen Umgebung zu erzeugen. Diese Fläche ist eine Funktion mit zwei Parametern $f(x, y)$. Sie bildet eine Ebene auf ihre Höhenwerte ab. Dafür müssen der Punkt und seine Umgebung, je nach Methode nur die direkten Nachbarn oder auch weitere, in diese Ebene überführt werden. Als Ebene kommt die Tangentialebene in Betracht. Sind die Nachbarpunkte und ihre Nebenbedingungen (z.B. Normalen) nun in das neue Koordinatensystem der Ebene überführt worden, kann man mit Hilfe dieser Werte ein mehr oder weniger komplexes Gleichungssystem aufstellen. Die gesuchten Parameter werden mittels einer *Least Squares*-Approximation gefunden. Nun hat man die komplette Funktion der Passfläche. Diese lässt sich leicht an der entsprechenden Stelle ableiten und man gewinnt somit die *Weingartenmatrix*, aus der sich die Krümmungen extrahieren lassen.

Bei der Projektion des Netzes in die Tangentialebene eines Punktes kann es zu Überschneidungen kommen und somit die Topologie zerstört werden. Um dies zu vermeiden, gibt es Ansätze, die eine Überführung in Polarkoordinaten vorsehen [Sch05], um die Topologie zu erhalten.

Ein Beispiel für das *Surface Fitting* wäre die *Quadratische Approximation*. Hier wird eine quadratische Funktion als Passfläche angenommen.

$$z = \frac{A}{2}x^2 + Bxy + \frac{C}{2}y^2$$

Die Koordinaten, der in die Tangentialebene transformierten Punkte, werden nun eingesetzt. Somit erhält man folgendes Gleichungssystem:

$$\begin{bmatrix} \vdots \\ \frac{1}{2}x_i^2 & x_i y_i & \frac{1}{2}y_i^2 \\ \vdots \end{bmatrix} \begin{bmatrix} A \\ B \\ C \end{bmatrix} = \begin{bmatrix} \vdots \\ z_i \\ \vdots \end{bmatrix}$$

Wird dieses Gleichungssystem gelöst und die Flächenfunktion abgeleitet, erhält man nun die *Weingartenmatrix*

$$W = \begin{bmatrix} A & B \\ B & C \end{bmatrix}.$$

Eine weitere Parametrisierung wäre die Methode der *Kubische-Adjazenz-Normalen*. Hierbei wird eine kubische Funktion für die Passfläche angenommen.

$$z = \frac{A}{2}x^2 + Bxy + \frac{C}{2}y^2 + Dx^3 + Ex^2y + Fxy^2 + Gy^3$$

Auch die Normalen der umgebenden Punkte werden betrachtet. Sie werden in das neue Koordinatensystem der Tangentialebene überführt und haben nun die Form $N(P_i) = (a_i, b_i, c_i)$. Aus diesen Informationen lässt sich dieses Gleichungssystem erstellen:

$$\begin{bmatrix} \vdots \\ \frac{1}{2}x_i^2 & x_i y_i & \frac{1}{2}y_i^2 & x_i^3 & x_i^2 y_i & x_i y_i^2 & y_i^3 \\ x_i & y_i & 0 & 3x_i^2 & 2x_i y_i & y_i^2 & 0 \\ 0 & x_i & y_i & 0 & x_i^2 & x_i y_i & 3y_i^2 \\ \vdots \end{bmatrix} \begin{bmatrix} A \\ B \\ C \\ D \\ E \\ F \\ G \end{bmatrix} = \begin{bmatrix} \vdots \\ z_i \\ -\frac{a_i}{c_i} \\ -\frac{b_i}{c_i} \\ \vdots \end{bmatrix}$$

Es wird gelöst, und aus der entstandenen Funktion sind die gewünschten Krümmungsparameter errechenbar.

3.2 Curve Fitting

Beim *Curve Fitting* werden zunächst die Krümmungen entlang der ausgehenden Kanten bestimmt. Diese sieht man am besten auf Abbildung 4 auf der nächsten Seite. Man stellt sich einen Kreis vor, welcher durch die zwei Endpunkte der Strecke $\overline{P_i P_j}$ geht und die Normalen des Punktes (N_i) berücksichtigt. Anhand dieses entstehenden Kreises kann man nun die Krümmungsinformationen entlang der Kante bestimmen. Die Formel zur Ermittlung dieser Kantenkrümmung lautet

$$\kappa_{N_i} = \frac{2N_i \cdot (P_j - P_i)}{(P_j - P_i)^2}.$$

Die nun gewonnenen Informationen werden auf die Tangentialebene abgebildet. Danach wird für die Krümmungsfunktion

$$\kappa = Ax^2 + Bxy + Cy^2$$

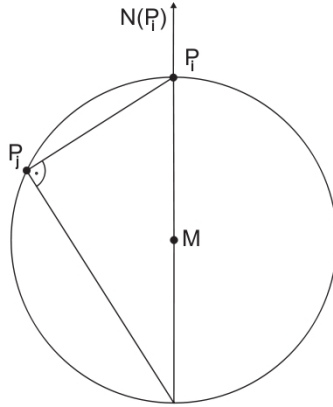


Abbildung 4: geometrische Deutung eines Krümmungskreis der Kante $\overline{P_i P_j}$ mit Normale $N(P_i)$ [Sch05]

ein Gleichungssystem

$$\begin{bmatrix} \vdots \\ x_i^2 & x_i y_i & y_i^2 \\ \vdots \end{bmatrix} \begin{bmatrix} A \\ B \\ C \end{bmatrix} = \begin{bmatrix} \vdots \\ \kappa_{N_i} \\ \vdots \end{bmatrix}$$

aufgestellt, welches mittels *Least Squares* gelöst wird. Nach dem Ableiten der Krümmungsfunktion erhält man dann wieder die *Weingartenmatrix*, aus der man die Krümmungsinformationen ablesen kann.

3.3 Tensor Averaging

Die Grundidee beim *Tensor Averaging* ist ein Krümmungstensor Π , auch als zweite Fundamentalform bezeichnet. Dieser Tensor wird für den untersuchten Punkt erstellt. Er ist ein gewichtetes Mittel der Krümmungstensoren der benachbarten Polygone. Aus diesem Tensor kann man dann die gewünschten Krümmungsparameter errechnen.

Ein Krümmungstensor ist so definiert, dass, wenn man einen Richtungsvektor mit ihm multipliziert, man die Änderung der Normalen in diese Richtung erhält. Diese Definition kann man sich zunutze machen bei der Berechnung dieses Tensors. Man erstellt hierfür ein Gleichungssystem, welches die Änderungen der Normalen der Eckpunkte (n_i) an ihren jeweiligen angrenzenden Kanten (e_j) enthält. Hierbei ist zu beachten, dass die Werte eines Polygons in ein Koordinatensystem (\vec{u}, \vec{v}) zu überführen sind, welches in der Ebene des Polygons liegt.

Für ein Dreieck (Abb. 5 auf der nächsten Seite) sieht das zu lösende Gleichungssystem

zur Bestimmung des Tensors wie folgt aus:

$$\begin{aligned} \Pi \cdot \begin{pmatrix} e_0 \cdot u \\ e_0 \cdot v \end{pmatrix} &= \begin{pmatrix} (n_2 - n_1) \cdot u \\ (n_2 - n_1) \cdot v \end{pmatrix} \\ \Pi \cdot \begin{pmatrix} e_1 \cdot u \\ e_1 \cdot v \end{pmatrix} &= \begin{pmatrix} (n_0 - n_2) \cdot u \\ (n_0 - n_2) \cdot v \end{pmatrix} \\ \Pi \cdot \begin{pmatrix} e_2 \cdot u \\ e_2 \cdot v \end{pmatrix} &= \begin{pmatrix} (n_1 - n_0) \cdot u \\ (n_1 - n_0) \cdot v \end{pmatrix} \end{aligned}$$

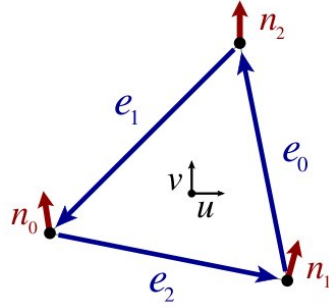


Abbildung 5: Dreieck ($e_{0..2}$) mit Normalen ($n_{0..2}$) und Koordinatensystem (\vec{u}, \vec{v}) [Rus04]

Dieses Gleichungssystem wird nun mittels *Least Squares* gelöst und man erhält den Tensor für das untersuchte Polygon.

Um nun das gewichtete Mittel der umgebenden Tensoren für einen Punkt zu ermitteln, müssen die Tensoren in die Ebene des Punktes überführt werden. Hierfür wird die Normale des Polygons in die Normale des untersuchten Punktes gedreht. Als Gewicht der Tensoren wird in [Rus04] der Anteil an der *Voronoi*-Umgebung (Abb. 6) verwendet.

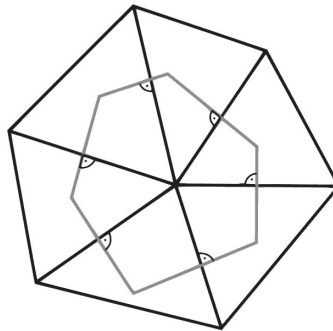


Abbildung 6: Voronoi-umgebung zur Wichtung der Krümmungstensoren der angrenzenden Flächen eines Punktes [Sch05]

4 Vergleich

Im Anschluss werden die in Kapitel 3 auf Seite 4 aufgezeigten Methoden verglichen in Hinsicht auf Robustheit, Genauigkeit und Skalierbarkeit. Beginnend mit der Robustheit ist als erstes das Verhalten der Methoden bei degenerierten Konfiguration wie in Abbildung 7(a) zu betrachten. Hierbei haben die *Fitting* Methoden Probleme, da ihre Gleichungssysteme meist unterbestimmt sind. Wie man in Abbildung 8 sieht reagieren die Methoden, welche nur kleine Umgebungen betrachten, anfälliger auf Rauschen. Methoden, welche die Normalen nutzen, sind dagegen stabiler.

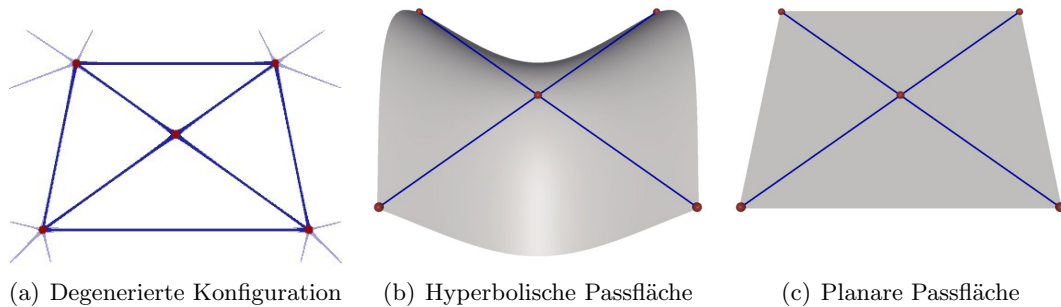


Abbildung 7: Entartete Konfiguration für Algorithmen, welche die Krümmung anhand einer eingepassten Fläche schätzen. Bei dieser Konfiguration liegen die Punkte auf zwei sich kreuzenden Linien. Dies ermöglicht keine eindeutige Bestimmung einer Passfläche. Hier zwei mögliche Passflächen mit verschiedenen Krümmungen. [Rus04]

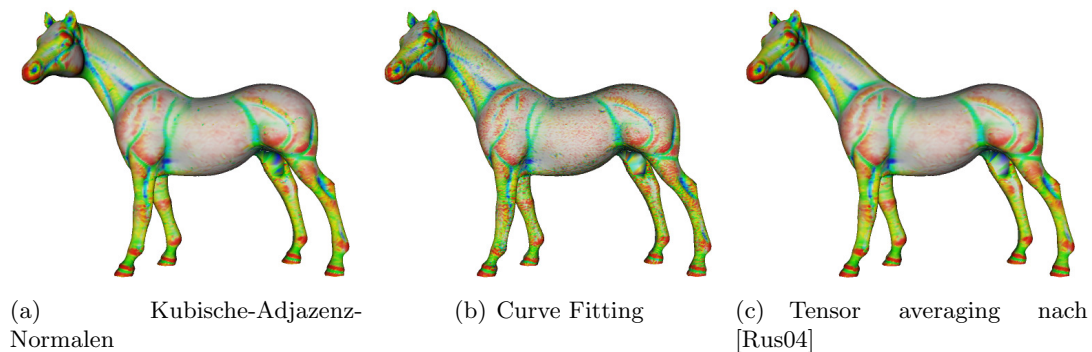


Abbildung 8: Farbcodierte Visualisierung der berechneten Krümmungen auf einem unregelmäßig tesselierten Modell mit verschiedenen Methoden. Zu Beachten ist das Vorhandensein von Ausreißern in den Schätzungen. [Rus04]

Im Hinblick auf die Genauigkeit der Methoden ist zu erkennen, dass die Fitting-Methoden in Abhängigkeit der Nachbarschaftsgröße entweder zu rauschanfälligen oder zu glatten Ergebnissen führen. Bei einer klein gewählten Nachbarschaft hat Rauschen

mehr Einfluss auf das Ergebnis. Bei einer groß gewählten Nachbarschaft ist hingegen die Verallgemeinerung der Ergebnisse hinderlich.

Zur Skalierbarkeit der vorgestellten Methoden ist zu sagen, dass die Fitting-Methoden für jeden einzelnen Punkt des Meshes die Lösung eines Gleichungssystems erfordern. Die Methode aus [Rus04] hingegen bestimmt für jeden Eckpunkt die Normale und für jede Fläche den Tensor. Somit skaliert die Methode aus [Rus04] bei Vergrößerung der Datenmengen besser.

5 Zusammenfassung

Da die Krümmung in der Computergraphik einen hohen Stellenwert einnimmt, wurde erklärt, was Krümmung ist bzw. welche mathematische Bedeutung (zweite Ableitung einer Funktion) sie hat. Da auf 3D-Modellen die Krümmung rechnerisch nur die Werte 0 und ∞ annimmt, muss sie geschätzt werden. Zur Schätzung wurden drei verschiedene Herangehensweisen vorgestellt. Beim *Surface Fitting* wird versucht, eine Passfläche durch einen Punkt und dessen Umgebung zu legen. Beim *Curve Fitting* wird die Krümmung an den ausgehenden Kanten errechnet und mit Hilfe dieser Krümmungen eine Flächenkurve ermittelt. Für die Kategorie des *Tensor Averaging* werden die Krümmungen der angrenzenden Teilflächen geschätzt und diese dann gewichtet gemittelt.

Im Vergleich hat die Methode der dritten Kategorie, vorgestellt in [Rus04], am besten abgeschnitten. Sie ist robust gegenüber Rauschen und degenerierten Teilstücken und wird auch mit zunehmender Komplexität des Modelles nicht übermäßig langsam.

In [Rus04] wird auch die dritte Ableitung betrachtet und nach dem selben Schema gebildet. Diese lässt sich gut zur Visualisierung von Objekten nutzen. Die Ausgabe ist hierbei mit den Methoden der *Feature Edges* vergleichbar.

Literatur

- [dC93] CARMO, MANFREDO PERDIGÃO DO: *Differentialgeometrie von Kurven und Flächen*. Vieweg, Wiesbaden, 1993.
- [Rus04] RUSINKIEWICZ, SZYMON: *Estimating Curvatures and Their Derivatives on Triangle Meshes*. In: *Proceeding of the 2nd International Symposium on 3D Data Processing, Visualization and Transmission*, Seiten 486–493. IEEE Computer Society, September 2004.
- [Sch05] SCHULZ, CHRISTIAN: *Approximation von Krümmungsinformation für die medizinische Visualisierung*. Diplomarbeit, Otto-von-Guericke-Universität Magdeburg, Oktober 2005.
- [Wik09] WIKIPEDIA: *Krümmung* — *Wikipedia, Die freie Enzyklopädie*, 2009. [Online; Stand 18. April 2009].