

In dieser Arbeit wollen wir aus einem gegebenem SurfaceNet eine möglichst glatte Oberfläche berechnen. Auf die Definition von Glattheit wollen wir an dieser Stelle nicht genauer eingehen.

Sei $x^j = (x_1^j, x_2^j, x_3^j) \in \mathbb{R}^3$ die Position des j -ten Vertex, $j = 1, 2, \dots, N$ mit der Anzahl an Vertices N . Das Potential

$$V[x] = \frac{1}{4} \sum_{i,j=1}^N |x^i - x^j|^2 \chi_{ij}, \quad x = (x^1, x^2, \dots, x^N) \quad (1)$$

ist die „Abstandsenergie“ mit der Nachbarindikatorfunktion

$$\chi_{ij} = \begin{cases} 1 & , i \text{ und } j \text{ sind Nachbarn} \\ 0 & , \text{sonst} \end{cases} . \quad (2)$$

Für die Bewegung der Vertices x^i gilt nun die Newtonsche Bewegungsgleichung

$$d_t^2 x^j = -\partial_{x^j} V[x], \quad x^j \in \Omega^j, j = 1, \dots, N, \quad (3)$$

mit der Zelle $\Omega^j = \left\{ x \in \mathbb{R}^3 : |x_k - \bar{x}_k^j| \leq \frac{1}{2}, k = 1, 2, 3 \right\}$ und Ausgangsposition \bar{x}^j . Wir verwenden zur Lösung dieses Systems gewöhnlicher, linearer, gekoppelter Differentialgleichungen das implizite Eulerverfahren und diskretisieren in der Zeit wie folgt

$$d_t^2 x = \frac{1}{t^2} (x^{(n+1)} - 2x^{(n)} + x^{(n-1)}), \quad (4)$$

mit der Zeitschrittweite t . Damit lässt sich die Bewegungsgleichung auch in der allgemeinen Form

$$Ax^{(n+1)} = 2x^{(n)} - x^{(n-1)} \quad (5)$$

schreiben. Dabei besteht die Matrix A aus Blöcken A_i auf der Diagonalen, B_{ij} auf den Nebendiagonalen der Form

$$A_i = \begin{pmatrix} 1 + t^2 \xi_i & 0 & 0 \\ 0 & 1 + t^2 \xi_i & 0 \\ 0 & 0 & 1 + t^2 \xi_i \end{pmatrix}, \quad (6)$$

$$B_{ij} = \begin{pmatrix} -t^2 \chi_{ij} & 0 & 0 \\ 0 & -t^2 \chi_{ij} & 0 \\ 0 & 0 & -t^2 \chi_{ij} \end{pmatrix}, \quad (7)$$

mit der Anzahl an Nachbarn ξ_i . Als Anfangsbedingung haben wir $x(0) = \bar{x}$ und $d_t x(0) = 0$, also $x^{(0)} = x^{(1)} = 0$. Zu beachten ist, dass die numerische Lösung in der Nähe des Randes zu Fehlern führen kann. Die Zeitschrittweite kann mit $t = 1$ gewählt werden, aber weitere Tests sind u.U. notwendig. Die Anzahl der Iterationen n kann in Abhängigkeit des Potential $V[x]$ (genauer der Änderung des Potentials $V[x^{(n+1)}] - V[x^{(n)}]$), der Kräfte auf die Vertices, einer maximalen Zeitdauer oder maximalen Anzahl an Iterationen gewählt werden.

Zur Lösung des Gleichungssystem haben wir 2 Möglichkeiten:

1. Einfach die inverse Matrix berechnen lassen, mit einem Aufwand $\mathcal{O}(N^{2.373})$,
2. Die Struktur und Eigenschaften der Matrix analysieren und die inverse Matrix per Hand berechnen.

Alternativ kann auch das explizite Eulerverfahren verwendet werden. Dies ist jedoch instabil und im allgemeinen langsamer als das implizite Verfahren bzgl. gleicher Resultate. Das explizite Verfahren lautet

$$x^{j(n+1)} = \frac{1}{1 + D \frac{t}{2}} \left(x^{j(n)} (2 - t^2 K \xi_j) - x^{j(n-1)} (1 - D \frac{t}{2}) \right) + t^2 \sum_{i=1}^N x^{i(n)} \chi_{ij}, \quad (8)$$

wobei wir eine Federstärke K und Dämpfung D eingebaut haben.

Nachdem die neuen Vertexpositionen berechnet wurden bleibt noch zu überprüfen, ob diese außerhalb der Zelle liegen. Sollte eine Vertex die Zelle verlassen, so wird die zugehörige Position auf dem Zellenrand berechnet. In dem Fall wird die korrigierte neue Position $x^{j(n+1)}$, wie folgt berechnet

$$x^{j(n+1)} = x^{j(n)} + \lambda(x^{j(n+1)} - x^{j(n)}), \quad (9)$$

wobei λ gegeben ist mit

$$\lambda = \left(\frac{\Delta_i}{2\Delta_i} - (x_i^{j(n)} - \bar{x}_i^j) \right) \frac{1}{x_i^{j(n+1)} - x_i^{j(n)}}. \quad (10)$$

Der Index i ist dabei derjenige Index, bei dem $\Delta = x^{j(n+1)} - \bar{x}^j$ den größten Wert annimmt.

Es gibt noch weitere Möglichkeiten das Surfacenet zu berechnen:

1. Andere Kräfte verwenden, z.B. Krümmung der Oberfläche Δx^j ,
2. die Vertices als Fluid betrachten, wobei Druck und Viskosität zu modellieren sind,
3. bestimmte Energiefunktionale, z.B. potentielle Energie basierend auf der Entfernung der Nachbarn, minimieren.

Autor: Paul Th.