

Erzeugen Stöße die Feinstrukturkonstante?

In Fortsetzung von [2015-Feinstrukturkonstante.pdf](#) soll in diesem **Arbeitsblatt** die Berechtigung zur Annahme der Existenz diskreter Planckobjekte (**Postulat** dort) erhärtet werden.

Ein Hinweis, dass einfache diskrete Objekte (Planckobjekte) die Feinstrukturkonstante erzeugen, erfordert die Stoßtransformationen. Die Verbindung zum schnellen und genauen de Vries-Algorithmus kann durch Verwendung von Mittelwerten und den erreichten Wert der vorhergehenden Iteration erzielt werden. Dafür sollen die Stoßtransformationen mit kartesischen Koordinaten durch einen beliebigen Ausgangswert des Relativ-Geschwindigkeitsbetrags in Kugelkoordinaten gefüllt werden. In deren beiden Winkeln sind demnach zu findende Zusammenhänge zu kodieren. Dazu kommen die Parameter der Stoßachsenwinkel. Der Spin ist ein Hinweis auf die zu suchenden Parameter.

Eine räumliche Ausdehnung kleinster Objekte führt zwangsweise zu Stößen. Im einfachsten Fall können diese lokal betrachtet werden, ohne ein ausgehntes Feld mit seinen raumzeitlichen Veränderungen mit zu untersuchen. Ob daraus bereits ein Erkenntnisgewinn im Hinblick auf die gewünschte Erklärung von elementaren Naturkonstanten zu erzielen ist, kann nur mit Hilfe einer Untersuchung vieler Stöße und der dabei stattfindenden Veränderungen überprüft werden. Dafür sind die Stoßtransformationen, also Bewegungsgleichungen zur Bestimmung der Geschwindigkeiten nach dem Stoß, erforderlich. Vektoren (3 Komponenten) werden hier (wie in Mathcad üblich) ohne Pfeil geschrieben, meistens **fett**.

Zuerst wird die Relativgeschwindigkeit der Stoßpartner bestimmt:

$$\mathbf{w}(\mathbf{u}, \mathbf{v}) := \mathbf{v} - \mathbf{u} \quad (1)$$

Die Richtung der Relativgeschwindigkeit wird mit der Kugelkoordinaten-Transformation ermittelt, für die hier die in Mathcad eingebaute Funktion verwendet wird:

$$\Phi(\mathbf{u}, \mathbf{v}) := \text{xyz2sph}(\mathbf{w}(\mathbf{u}, \mathbf{v})_0, \mathbf{w}(\mathbf{u}, \mathbf{v})_1, \mathbf{w}(\mathbf{u}, \mathbf{v})_2)_1 \quad (2)$$

$$\Theta(\mathbf{u}, \mathbf{v}) := \text{xyz2sph}(\mathbf{w}(\mathbf{u}, \mathbf{v})_0, \mathbf{w}(\mathbf{u}, \mathbf{v})_1, \mathbf{w}(\mathbf{u}, \mathbf{v})_2)_2 \quad (3)$$

Bei den Ergebnissen mit der ausführlichen Transformation gemäß dem Artikel über **Kugelkoordinaten** in Wikipedia, nimmt Φ Werte von 0 bis 2π an.

Die Stoßachsenwinkel ergeben sich i.A. zufallsabhängig, wobei gleichwahrscheinliche parallele Bahnen zur Richtung der Relativgeschwindigkeit angenommen werden. Das ist auf gleichwahrscheinliche parallele Bahnen bei den Stoßpartnern zurückzuführen. Damit ergibt sich in kartesischen Koordinaten der Stoßachsenvektor:

$$\mathbf{S}_z(\theta_s, \phi_s) := \begin{pmatrix} \cos(\phi_s) \cdot \sin(\theta_s) \\ \sin(\phi_s) \cdot \sin(\theta_s) \\ \cos(\theta_s) \end{pmatrix} \quad (4)$$

Dieser wurde relativ zur Richtung der Relativgeschwindigkeit $\mathbf{w}(\mathbf{u}, \mathbf{v})$ erzeugt und muss nun im ursprünglichen Koordinatensystem (dem Laborsystem von \mathbf{u} und \mathbf{v}) ausgedrückt werden, was durch zwei hintereinander ausgeführte Drehungen erreicht wird:

$$\mathbf{D}_z(\mathbf{u}, \mathbf{v}) := \begin{pmatrix} \cos(\Phi(\mathbf{u}, \mathbf{v})) & \sin(\Phi(\mathbf{u}, \mathbf{v})) & 0 \\ -\sin(\Phi(\mathbf{u}, \mathbf{v})) & \cos(\Phi(\mathbf{u}, \mathbf{v})) & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (5)$$

$$\mathbf{D}_y(\mathbf{u}, \mathbf{v}) := \begin{pmatrix} \cos(\Theta(\mathbf{u}, \mathbf{v})) & 0 & -\sin(\Theta(\mathbf{u}, \mathbf{v})) \\ 0 & 1 & 0 \\ \sin(\Theta(\mathbf{u}, \mathbf{v})) & 0 & \cos(\Theta(\mathbf{u}, \mathbf{v})) \end{pmatrix} \quad (6)$$

Damit ergibt sich die Stoßachse im ursprünglichen Koordinatensystem durch das zweifache Zurückdrehen zu:

$$S(\mathbf{u}, \mathbf{v}, \theta_s, \phi_s) := Dz(\mathbf{u}, \mathbf{v})^{-1} \cdot Dy(\mathbf{u}, \mathbf{v})^{-1} \cdot Sz(\theta_s, \phi_s) \quad (7)$$

Dieses S entspricht beim Zentralstoß auf eine ruhende Kugel dem ursprünglichen \mathbf{u} bzw. beim Zentralstoß auf ein beliebiges \mathbf{v} allgemeiner dem Relativgeschwindigkeitsvektor \mathbf{w} normiert auf 1.

Beim Stoß werden nun die zur Stoßachse parallelen Geschwindigkeiten der beiden beteiligten Kugeln ausgetauscht. **Das ist die elementare Wechselwirkung, welche durch das Postulat eingeführt wurde.** Alle Vektoren sollen jedoch weiterhin im ursprünglichen Koordinatensystem betrachtet werden.

$$u_p(\mathbf{u}, \mathbf{v}, \theta_s, \phi_s) := S(\mathbf{u}, \mathbf{v}, \theta_s, \phi_s) \cdot (S(\mathbf{u}, \mathbf{v}, \theta_s, \phi_s) \cdot \mathbf{u}) \quad (8)$$

$$v_p(\mathbf{u}, \mathbf{v}, \theta_s, \phi_s) := S(\mathbf{u}, \mathbf{v}, \theta_s, \phi_s) \cdot (S(\mathbf{u}, \mathbf{v}, \theta_s, \phi_s) \cdot \mathbf{v}) \quad (9)$$

parallele
Geschwindigkeiten

$$u_o(\mathbf{u}, \mathbf{v}, \theta_s, \phi_s) := \mathbf{u} - u_p(\mathbf{u}, \mathbf{v}, \theta_s, \phi_s) \quad (10)$$

$$v_o(\mathbf{u}, \mathbf{v}, \theta_s, \phi_s) := \mathbf{v} - v_p(\mathbf{u}, \mathbf{v}, \theta_s, \phi_s) \quad (11)$$

orthogonale
Geschwindigkeiten

$$u_s(\mathbf{u}, \mathbf{v}, \theta_s, \phi_s) := v_p(\mathbf{u}, \mathbf{v}, \theta_s, \phi_s) + u_o(\mathbf{u}, \mathbf{v}, \theta_s, \phi_s) \quad (12)$$

$$v_s(\mathbf{u}, \mathbf{v}, \theta_s, \phi_s) := u_p(\mathbf{u}, \mathbf{v}, \theta_s, \phi_s) + v_o(\mathbf{u}, \mathbf{v}, \theta_s, \phi_s) \quad (13)$$

Geschwindigkeiten nach
Stoß

Sind demnach die erforderlichen **Stoßtransformationen** (ohne Differentiale),

Für die folgenden Überlegungen soll ein Vektor zuerst in Kugelkoordinaten definiert werden:

$$V(r, \theta, \phi) := (r \ \theta \ \phi) \quad V(1.7, 0, 0) = (1.7 \ 0 \ 0)$$

Anstelle des willkürlichen Wertes 1.7 könnte 3.67 oder -2.085 oder ... stehen. Kartesisch wird das:

$$v(r, \theta, \phi) := \begin{pmatrix} r \cdot \sin(\theta) \cdot \cos(\phi) \\ r \cdot \sin(\theta) \cdot \sin(\phi) \\ r \cdot \cos(\theta) \end{pmatrix} \quad v(1.7, 0, 0) = \begin{pmatrix} 0 \\ 0 \\ 1.7 \end{pmatrix} \quad \theta_s := \frac{\pi}{4} \quad \phi_s := \frac{\pi}{4}$$

r und Vektorwinkel
in Kugelkoordinaten

Stoßachsenwinkel

$$\mathbf{u} := v\left(-1.7, 0, \frac{\pi}{4}, 0, \frac{\pi}{4}\right) = \begin{pmatrix} 0 \\ 0 \\ -1.7 \end{pmatrix} \quad \mathbf{v} := v\left(1.7, \frac{\pi}{4}, \frac{\pi}{4}\right) = \begin{pmatrix} 0.85 \\ 0.85 \\ 1.202081528017131 \end{pmatrix}$$

$$|\mathbf{u}| = 1.7$$

$$|\mathbf{v}| = 1.7$$

Für den Beginn der Überlegungen wird die Struktur der Simulation vieler Stöße aus 2015-Feinstrukturkonstante verwendet. Dabei werden erst einmal Beispielswerte angeschaut, um auch den Formalismus von Mathcad zu testen.

$$\begin{aligned}
z_a(z, \theta, \phi) := & \left. \begin{aligned}
& \theta_s \leftarrow \frac{\pi}{4} \\
& \phi_s \leftarrow \frac{\pi}{4} \\
& \theta_0 \leftarrow \theta \\
& \phi_0 \leftarrow \phi \\
& a_0 \leftarrow z \\
& \mathbf{u}_0 \leftarrow \begin{pmatrix} 0 \\ 0 \\ -z \end{pmatrix} \\
& \mathbf{v}_0 \leftarrow \begin{pmatrix} z \cdot \sin(\theta_0) \cdot \cos(\phi_0) \\ z \cdot \sin(\theta_0) \cdot \sin(\phi_0) \\ z \cdot \cos(\theta_0) \end{pmatrix} \\
& \text{for } i \in 1..8 \\
& \left. \begin{aligned}
& \mathbf{u}_i \leftarrow \begin{pmatrix} 0 \\ 0 \\ -a_{i-1} \end{pmatrix} \\
& \mathbf{v}_i \leftarrow v_s(\mathbf{u}_{i-1}, \mathbf{v}_{i-1}, \theta_s + \theta_{i-1}, \phi_s + \phi_{i-1}) \\
& \mathbf{w}_i \leftarrow \mathbf{v}_{i-1} - \mathbf{u}_{i-1} \\
& \theta_i \leftarrow \text{xyz2sph}[(\mathbf{w}_{i-1})_0, (\mathbf{w}_{i-1})_1, (\mathbf{w}_{i-1})_2]_1 \\
& \phi_i \leftarrow \text{xyz2sph}[(\mathbf{w}_{i-1})_0, (\mathbf{w}_{i-1})_1, (\mathbf{w}_{i-1})_2]_2 \\
& a_i \leftarrow |\mathbf{v}_i|
\end{aligned} \right. \\
& a \leftarrow a
\end{aligned}
\right.
\end{aligned}$$

\mathbf{v}_0 ist der erste Stoßpartner aus der Richtung mit θ und ϕ in kartesische Koordinaten transformiert.

Bei der Iteration sollen die bereits erzielten Geschwindigkeitsbetragsänderungen, also a_i als neue Geschwindigkeitsbeträge beider Vektoren verwendet werden.

Dafür sind Kugelkoordinaten-Transformationen nötig. Zuerst wird der Winkel der Relativgeschwindigkeit bestimmt und dann die neuen (kartesischen) Vektoren. Bei \mathbf{u} ändert sich nur die Länge a_i .

Die Übertragung des allgemeinen Formalismus der Simulationen stellte sich als überflüssig heraus, weshalb dessen gleiche Resultate hier nicht nötig sind. Vereinfacht auf einen Vektor ergibt sich mit der Stoßtransformation, in der \mathbf{u} im Ursprung liegt und \mathbf{v} bereits mit dem willkürlichen Zahlenwert [4.13897129](#), der hier **durch Probieren** ermittelt wurde, für einen (durchschnittlichen axialen?) Vektor beginnt. Die Zahl der Iterationen entspricht der Zahl von Durchläufen in der ursprünglichen Simulation. Hier sind wegen der Verwendung von Durchschnittswerten bis zur Konvergenz zur FSK viel weniger erforderlich.

$$\begin{aligned}
z_a(z) := & a_0 \leftarrow z \\
& \mathbf{v}_0 \leftarrow \begin{pmatrix} 0 \\ 0 \\ a_0 \end{pmatrix} \\
& \text{for } i \in 1..8 \\
& \left| \mathbf{v}_i \leftarrow v_s \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \left(\begin{array}{c} \frac{\pi}{4.13897129} \\ 0 \\ a_{i-1} \end{array} \right), \frac{\pi}{4}, 0 \right. \\
& \left. a_i \leftarrow \left(\frac{|\mathbf{v}_i|}{2 \cdot \pi} \right)^2 \right. \\
& a \leftarrow a
\end{aligned}$$

Zum Vergleich die Fixpunktiteration nach de Vries (umgeformt auf Summendarstellung):

Das Quadrat der Taylorreihe ist eine Beschreibung für zwei stoßende Objekte. Der Iterationsfaktor kann als Wurzel des Abstands in das Quadrat eingefügt werden. Die Iteration ersetzt möglicherweise die Differentiation

$$\begin{aligned}
\alpha x(x) := & g_0 \leftarrow x \\
& \text{for } i \in 1..8 \\
& g_i \leftarrow \left[1 + g_{i-1} + \frac{(g_{i-1})^2}{2 \cdot \pi} \right]^2 \cdot e^{-\frac{\pi^2}{2}} \\
& g \leftarrow g
\end{aligned}$$

Werden von den **Stößen** nur die Mittelwerte berücksichtigt, konvergieren a und g gegen die FSK. Im willkürlichen Zahlenwert bzw. dem Iterationsfaktor stecken möglicherweise die aus vielen Einzelstößen stammenden Standardabweichungen. Die Formel für die FSK enthält auch noch $h/2\pi$. **Überlegungen dazu:** Der Winkel β des zweiten Planckobjekts kann die Information über lokale freie Weglängen enthalten. Wird er durch die konstante **Elementarladung** bestimmt, kann das auch vom Einfluss des **Spin**, also einer Drehung der Relativgeschwindigkeit bei jedem Stoß, abhängen.

Zum Vergleich noch einmal die Fixpunktiteration. Ein Unterschied nach der zehnten Nachkommastelle rührt vermutlich von der Rechengenauigkeit des CAS und wird hier ausgeblendet.

$$z_a(1.7) = \begin{pmatrix} 1.7 \\ 0.04389895565 \\ 0.0073210853 \\ 0.00729735689 \\ 0.0072973525 \\ 0.0072973525 \\ 0.0072973525 \\ 0.0072973525 \\ 0.0072973525 \\ 0.0072973525 \end{pmatrix}$$

$$\alpha x(1.7) = \begin{pmatrix} 1.7 \\ 0.0718133517 \\ 0.0082745782 \\ 0.0073115534 \\ 0.0072975588 \\ 0.0072973555 \\ 0.0072973526 \\ 0.0072973525 \\ 0.0072973525 \\ 0.0072973525 \end{pmatrix}$$

Zum weiteren Vergleich auch noch die Produkt- und Summenform der Fixpunktiteration. Das Weglassen des zweiten Gliedes, welches die zweite Ableitung benötigt, führt auf Werte, welche experimentell überprüft werden sollten. Die geringe Abweichung bei Verzicht auf zweite Ableitungen könnte Hinweis auf die von Bjorken/ Drell angedeutete Möglichkeit bzw. Notwendigkeit eines neuartigen Gedanken sein (siehe Bjorken/ Drell, Relativistische Quantenfeldtheorie, BI Mannheim, Wien, Zürich 1990, [S.15](#)).

$$\alpha\text{Pr}(x) := \begin{cases} a_0 \leftarrow x \\ \text{for } i \in 1..8 \\ a_i \leftarrow \left[1 + a_{i-1} \cdot \left(1 + \frac{a_{i-1}}{2 \cdot \pi} \right) \right]^2 \cdot e^{-\frac{(\pi)^2}{2}} \\ a \leftarrow a \end{cases}$$

Produktform, der Original De Vries Formel zum Nachweis, dass sie der Summendarstellung entspricht:

$$\alpha\text{Su}(x) := \begin{cases} a_0 \leftarrow x \\ \text{for } i \in 1..8 \\ a_i \leftarrow \left(1 + a_{i-1} \right)^2 \cdot e^{-\frac{(\pi)^2}{2}} \\ a \leftarrow a \end{cases}$$

Summenform nach Wiese

mit ohne
Glied zweiter Ableitung

$$\alpha\text{Pr}(1.7) = \begin{pmatrix} 1.7 \\ 0.07181335169 \\ 0.00827457821 \\ 0.00731155342 \\ 0.00729755878 \\ 0.00729735554 \\ 0.00729735259 \\ 0.00729735255 \\ 0.00729735255 \end{pmatrix}$$

$$\alpha x(1.7) = \begin{pmatrix} 1.7 \\ 0.07181335169 \\ 0.00827457821 \\ 0.00731155342 \\ 0.00729755878 \\ 0.00729735554 \\ 0.00729735259 \\ 0.00729735255 \\ 0.00729735255 \end{pmatrix}$$

$$\alpha\text{Su}(1.7) = \begin{pmatrix} 1.7 \\ 0.05242882966 \\ 0.00796577633 \\ 0.00730691758 \\ 0.00729736834 \\ 0.00729722998 \\ 0.00729722797 \\ 0.00729722795 \\ 0.00729722794 \end{pmatrix}$$

Bei der Iteration, welche aus der Stoßsimulation durch Vereinfachung entsteht, kann auch die gesamte Stoßtransformation weggelassen werden. Der Zahlenwert $\pi/4.13897 = 0.75902740886$ spielt eine Schlüsselrolle:

$$y(z) := \begin{cases} a_0 \leftarrow z \\ \mathbf{v}_0 \leftarrow \begin{pmatrix} 0 \\ 0 \\ a_0 \end{pmatrix} \\ \text{for } i \in 1..8 \\ \mathbf{v}_i \leftarrow \begin{pmatrix} \frac{\pi}{4.13897129} \\ 0 \\ a_{i-1} \end{pmatrix} \\ a_i \leftarrow \frac{\left(\frac{|\mathbf{v}_i|}{2 \cdot \pi} \right)^2}{2} \\ a \leftarrow a \end{cases}$$

$$\alpha\text{DV}_{\text{sum}}(x) := \begin{cases} a_0 \leftarrow x \\ \text{for } i \in 1..8 \\ a_i \leftarrow \left[1 + \sum_{j=0}^{27} \left[\frac{(a_{i-1})^{j+1}}{\sum_{j=0}^j (2 \cdot \pi)^j} \right] \right]^2 \cdot e^{-\frac{(\pi)^2}{2}} \\ a \leftarrow a \end{cases}$$

Diese verallgemeinerte Summendarstellung konvergiert gleich wie die Original de Vries Fixpunktiteration. Das CAS steigt aber beim Wert > 27 der ersten Summe aus. In Python wurde übrigens die Präzision des Grenzwertes von Gottfried Helms am 11.04.2014 im Thread: "Ist die Feinstrukturkonstante Ergebnis von Stößen (Fortsetzung von 2005)" auf beliebig viele Stellen erweitert.

FSK - Iteration ohne
Stöße

FSK - Iteration mit Stößen und $\pi/4$
Stoßachsenwinkeln

normale
Fixpunktiteratio
n

liefern vollkommen identische Ergebnisse

$$\begin{array}{ccc}
 \begin{array}{c}
 y(1.7) = \begin{pmatrix} 1.7 \\ 0.04389895565 \\ 0.0073210853 \\ 0.00729735689 \\ 0.0072973525 \\ 0.0072973525 \\ 0.0072973525 \\ 0.0072973525 \\ 0.0072973525 \end{pmatrix} &
 z_a(1.7) = \begin{pmatrix} 1.7 \\ 0.04389895565 \\ 0.0073210853 \\ 0.00729735689 \\ 0.0072973525 \\ 0.0072973525 \\ 0.0072973525 \\ 0.0072973525 \\ 0.0072973525 \end{pmatrix} &
 \alpha_x(1.7) = \begin{pmatrix} 1.7 \\ 0.0718133517 \\ 0.0082745782 \\ 0.0073115534 \\ 0.0072975588 \\ 0.0072973555 \\ 0.0072973526 \\ 0.0072973525 \\ 0.0072973525 \end{pmatrix}
 \end{array}
 \end{array}$$

Ohne Stoß geht es demnach auch, der Faktor 1/2 deutet aber auf die **Impulsaufteilung**, welche wohl doch Stöße erfordert.

Im betrachteten Algorithmus stecken aber auch eine Wurzel mit dem Satz von Pythagoras und die Größen 2π , welche hier nur von Hand iteriert werden:

$$\text{ysqrt}(x) := \frac{1}{2} \cdot \left[\frac{\sqrt{\left(\frac{\pi}{4.13897129}\right)^2 + x^2}}{2 \cdot \pi} \right]^2$$

$$\text{ysqrt}(0.04389895565) = 0.0073210853$$

$$\text{ysqrt}(0.00729735689) = 0.0072973525$$

$$\text{ysqrt}(1.7) = 0.04389895565$$

$$\text{ysqrt}(0.0073210853) = 0.00729735689$$

$$\text{ysqrt}(0.0072973525) = 0.0072973525$$

$$\frac{1}{2} \cdot \left[\frac{\sqrt{\left(\frac{\pi}{4.13897129}\right)^2 + x^2}}{2 \cdot \pi} \right]^2$$

ergibt vereinfacht $0.01266514795529222143 \cdot x^2 + 0.0072966780630964478135$

was ebenfalls in einer **Iteration**, sogar mit einem Taschenrechner, die gleichen Resultate liefert.

Steckt darin ebenfalls der **Satz von Pythagoras**? Ist dieser im ganz Kleinen mit **Stößen** verbunden?

Wichtig wird nun die Suche nach einer Erklärung der darin steckenden willkürlich erscheinenden Zahlen. Dafür sollen selbständig entstehende **Strukturen** im Substrat der Planckobjekte untersucht werden. Damit wird die Dunkle Phase des Universums verlassen.