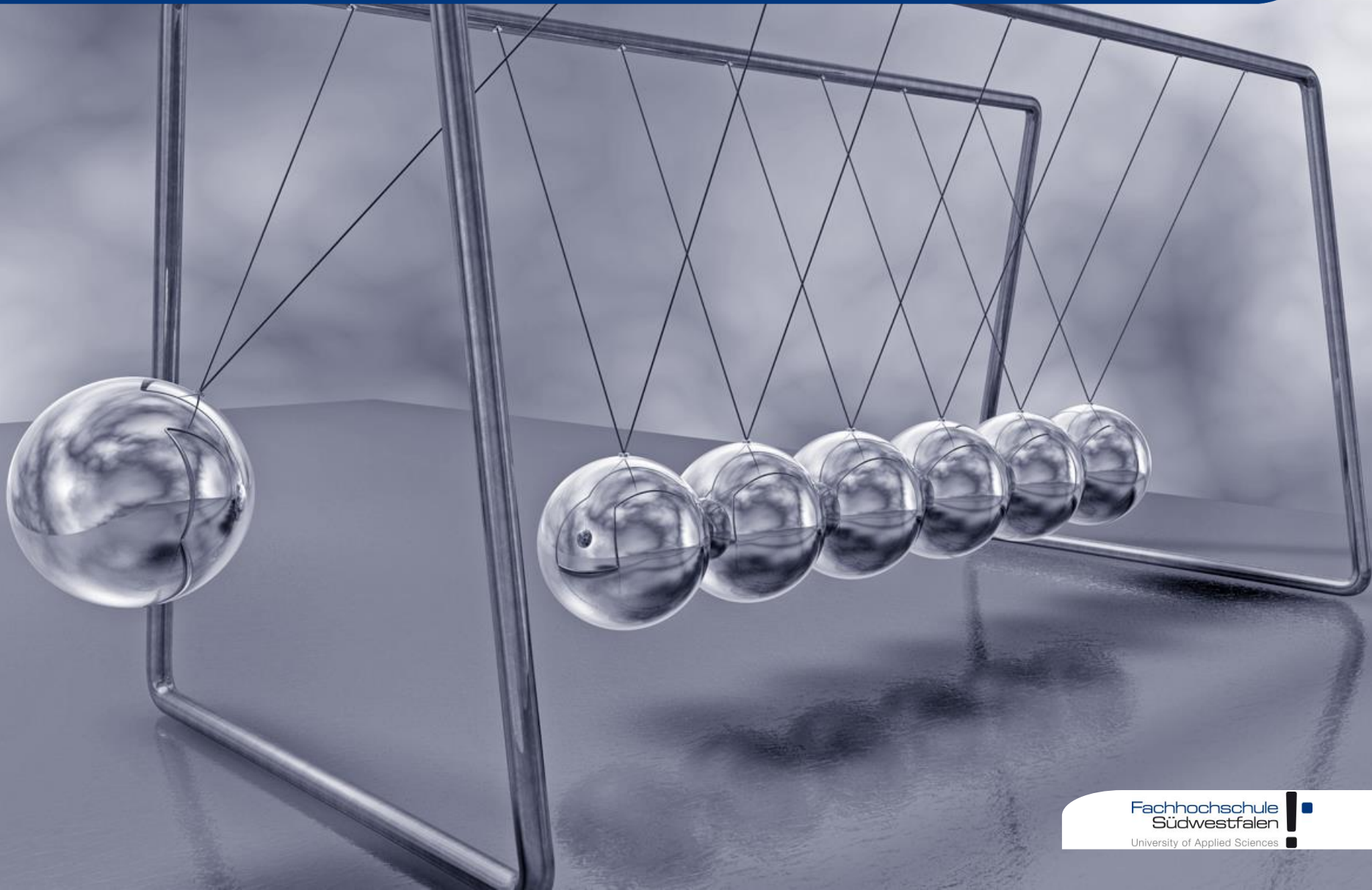


Lichtausbreitung in Räumen

Skizzen zur Monte-Carlo-Simulation von Jendrik Kraft und Jenny Kuhn



Agenda

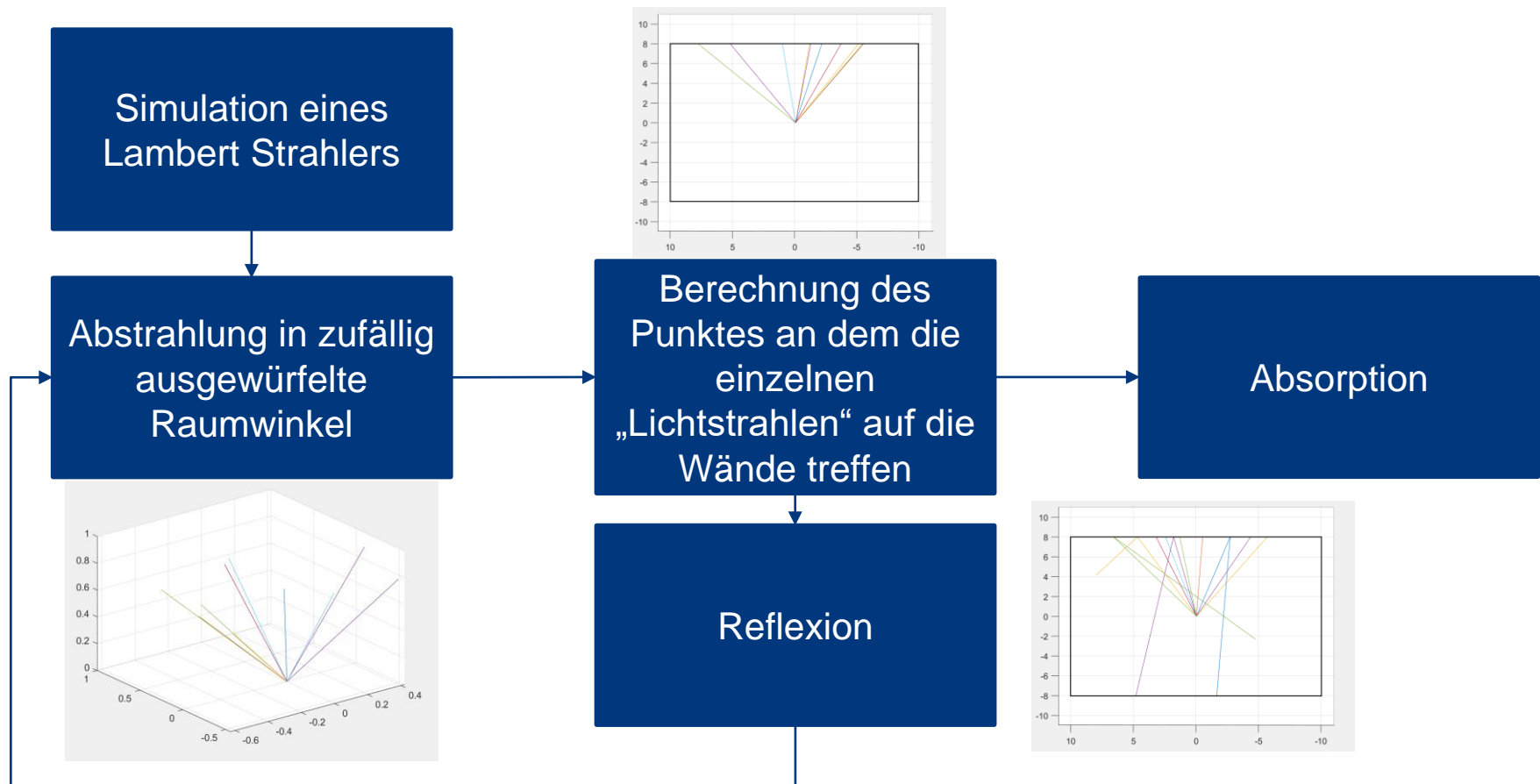


- Aufgabenstellung
- Monte-Carlo-Simulation
- Lambert-Strahler
- Schnittpunkte zwischen Grade und Ebene
- Drehmatrix
- Überlebensmatrix
- Auswertung & Visualisierung
 - Path Tracing
 - Heatmap
- Nächste Schritte

Aufgabenstellung

Lichtausbreitung in Räumen: Skizzen zur Monte-Carlo-Simulation

Ziel: Ausleuchtung von (quadratischen) Räumen simulieren



Monte-Carlo-Simulation

- In der Vorlesung: Schätzung von π und e
- Mathematisches Verfahren um mögliche Ergebnisse eines ungewissen Ereignisses abschätzen zu können
- Wahrscheinlichkeitsverteilung für jede Variable
- Wiederholung der Simulation
- **Vorteile:** bietet Sensitivitätsanalysen oder berechnen der Korrelation von Eingaben
- **Anwendungen:** KI, Klimamodelle, Aktienkurse, Preisgestaltung

Quellen: [Was ist die Monte-Carlo-Simulation? - Deutschland | IBM](#), Zugriff: 23.12.2021 um 10:00
[Monte Carlo Simulation - Definition, Methods, Examples \(wallstreetmojo.com\)](#), Zugriff: 23.12.2021 um 10:15

Lambert-Strahler

Kugelkoordinaten

- Modell für diffusen Strahler / Reflexion an Wänden
- Richtungsvektor der Photonen zufällig verteilt
- Verwendung von Kugelkoordinaten
- φ über 2π gleichverteilt
- θ zwischen 0 und π
 - Für unseren Strahler nur 0 bis $\pi/2$
 - Verteilung: $\theta = \arccos(\sqrt[m+1]{u})$

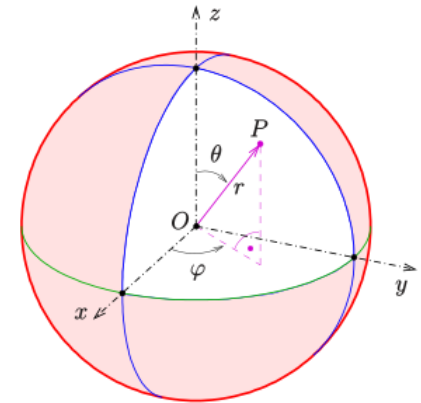


Abbildung 1: Kugelkoordinaten

Quelle: <https://de.wikipedia.org/wiki/Kugelkoordinaten>

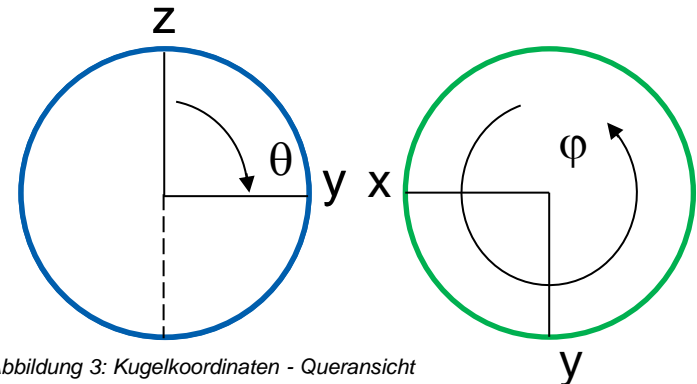


Abbildung 3: Kugelkoordinaten - Queransicht

Abbildung 2: Kugelkoordinaten - Draufsicht

Quellen: „Lichtausbreitung in Räumen: Skizzen zur Monte-Carlo-Simulation“, von Henrik Schulze, 3.11.2021

Lambert-Strahler

Beispielsimulation für $m=1$

- Verteilung: $\theta = \arccos(\sqrt[m+1]{u})$

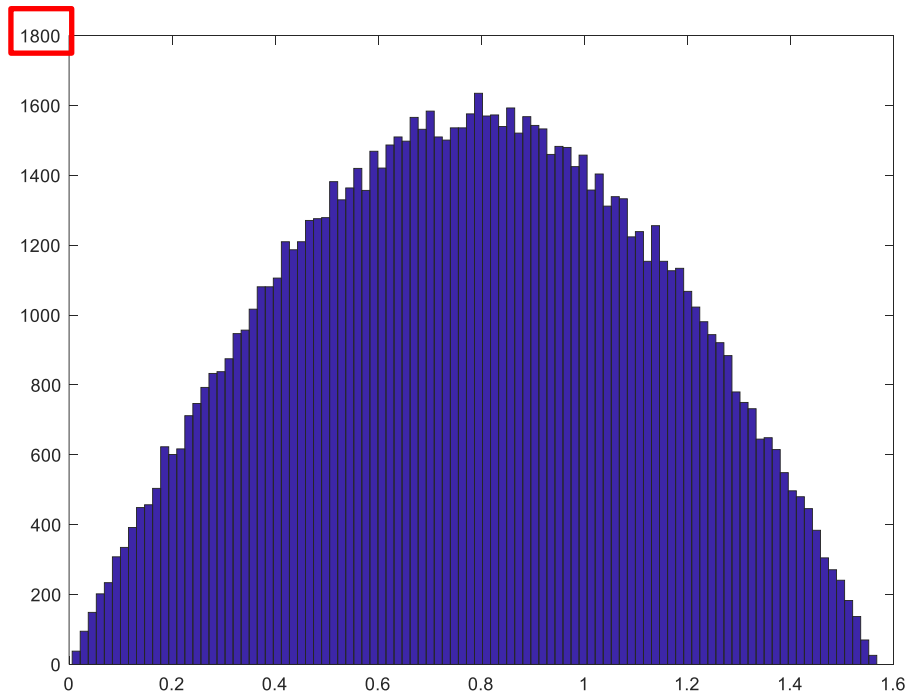


Abbildung 4: Zufällige Häufigkeitsverteilung von θ , $N=100000$, $m=1$ (Lambertstrahler)

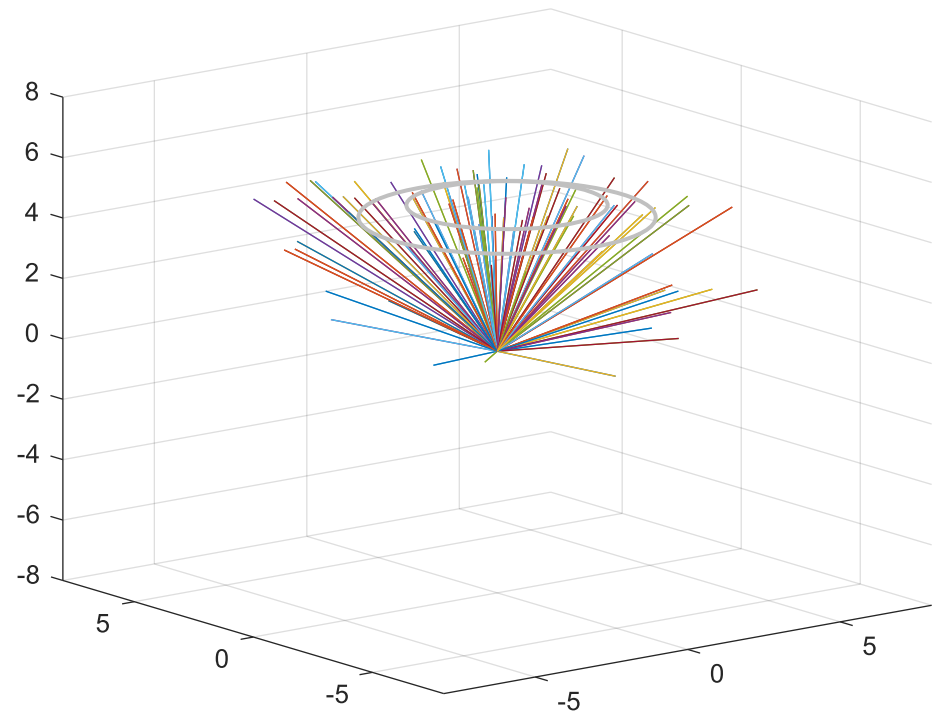


Abbildung 5: Simulation von 100 Lichtstrahlen mit $m=1$

(verallgemeinerter) Lambert-Strahler

Beispielsimulation für $m=10$

- Verteilung: $\theta = \arccos({}^{m+1}\sqrt{u})$

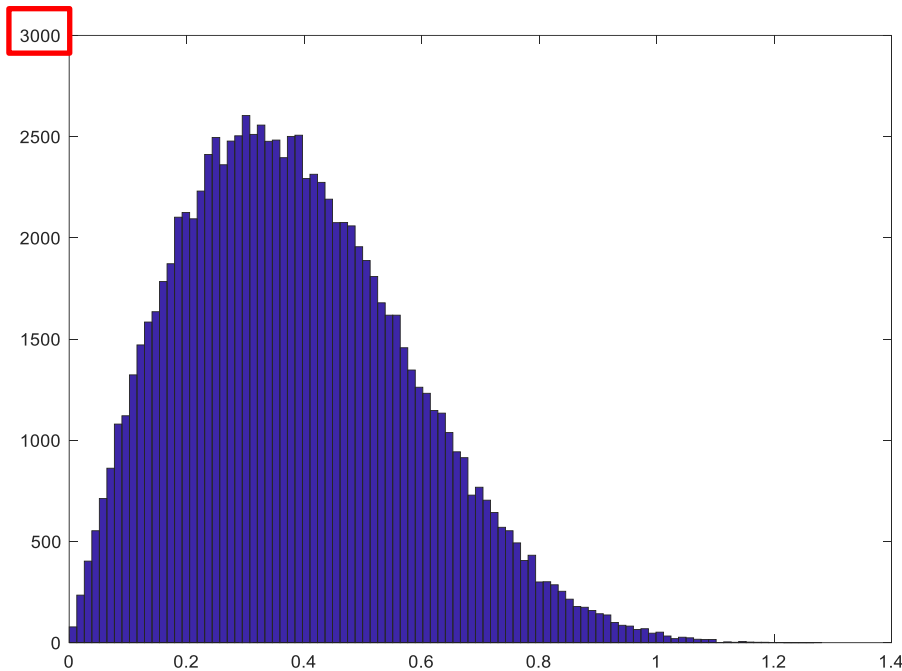


Abbildung 6: Zufällige Häufigkeitsverteilung von θ , $N=100000$, $m=10$ (gerichteter Strahler)

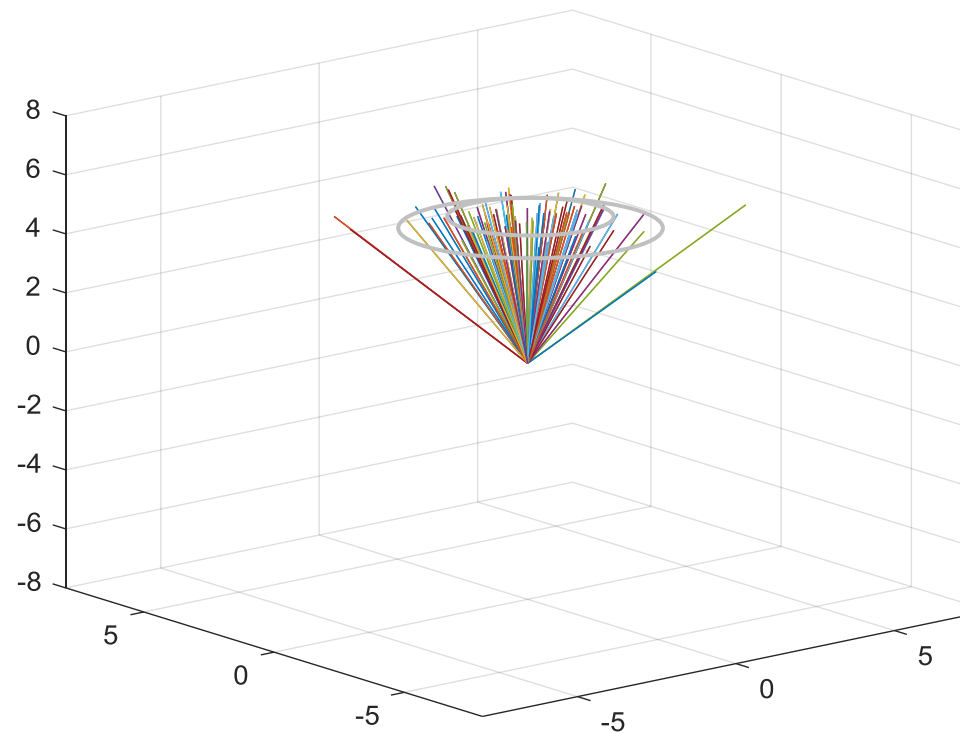


Abbildung 7: Simulation von 100 Lichtstrahlen mit $m=10$

- Ziel: Richtungsvektoren
- $\theta = \arccos({}^{m+1}\sqrt{u})$
 - $\cos(\theta) = {}^{m+1}\sqrt{u}$
- φ über 2π gleichverteilt
- Berechnen des Richtungsvektors

```
function [u_x, u_y, u_z]=  
BerZufRiVekt(n, m, Schnittpunkt, groesse)  
  
% Gleichverteilte Zufallsvariable  
u = rand(n,1);  
cos_theta = nthroot(u,m+1);  
sin_theta = sqrt(1-cos_theta.^2);  
  
phi = 2*pi()*rand(n,1);  
cos_phi = cos(phi);  
sin_phi = sin(phi);  
  
%% Vektor Berechnung der Photonen  
u_x = sin_theta .* cos_phi;  
u_y = sin_theta .* sin_phi;  
u_z = cos_theta;
```

Quellen: „Lichtausbreitung in Räumen: Skizzen zur Monte-Carlo-Simulation“, von Henrik Schulze, 3.11.2021

Schnittpunkte zwischen Grade und Ebene

Definition der Wände

- „Photonenflug“ als Geradengleichung
- Wände als Ebenengleichung
- Definieren von 6 x 3 Punkten um 6 Ebenen aufspannen zu können, für jede Wand eine
- Berechnen der Koordinatenform für jede Wand

```
%% Wand Matrizen Definition
%           Decke   Boden
P1 = groesse*[0,0,1; 0,0,-1; ...
P2 = groesse*[1,0,1; 1,0,-1; ...
P3 = groesse*[0,1,1; 0,1,-1; ...

%   Vorne   Hinten
... 1,0,0; -1,0,1; ...
... 1,0,1; -1,0,0; ...
... 1,1,1; -1,1,1; ...

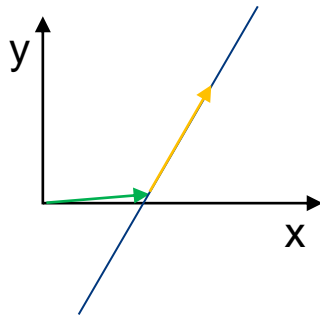
%   Links   Rechts
... 0,1,0; 1,-1,1;];
... 1,1,1; 1,-1,0;];
... 1,1,0; 0,-1,1;];

function [A,B,C,D] =
BerKoordinatenform(P1,P2,P3)
for n=1:6
    normal(:,n) =
        cross(P1(n,:)-P2(n,:),
              P1(n,:)-P3(n,:));
    A(n) = normal(1,n);
    B(n) = normal(2,n);
    C(n) = normal(3,n);
    D(n) = -dot(normal(:,n),
                 P1(n,:));
end
end
```

Schnittpunkte zwischen Gerade und Ebene

Photonenvektor

- Ausgangspunkt und Richtung der einzelnen Photonen (i)



- Schnittpunkt berechnen auf Basis von Ausgangspunkt, Richtung und Koordinatenform der Wände

```
%% Berechnen der Schnittpunkte mit  
%% Wand  
stuetz_v = startpunkte(i,:);  
richtungs_v = [u_x(i) u_y(i)  
               u_z(i)];
```

```
function [richtiger_Schnittpunkt] =  
    BerSchnittpunkt(stuetz_v,  
    richtungs_v, A, B, C, D, n,  
    groesse)  
    ...  
end
```

Schnittpunkte zwischen Gerade und Ebene

Vektorrechnung

- Gradengleichung:

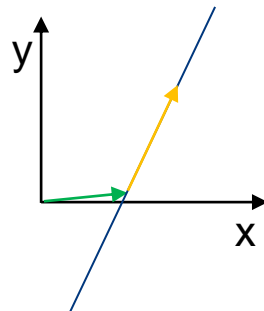
$$p = \text{stuetz_v} + t \cdot \text{richtungs_v}$$

- Ebenengleichung in Koordinatenform:

$$\text{Wand} = Ax + By + Cz + D$$

- Schnittpunkt: Geraden- und Ebenengleichung gleichsetzen und nach t auflösen:

- t in die Gradengleichung einsetzen ergibt den Schnittpunkt



```
t(photonenzahl) =  
    (-A(wand)*stuetz_v(1)  
    - B(wand)*stuetz_v(2)  
    - C(wand)*stuetz_v(3)  
    - D(wand)) /  
    A(wand)*richtungs_v  
    photonenzahl,1) +  
    B(wand)*richtungs_v  
    photonenzahl,2) +  
    C(wand)*richtungs_v  
    photonenzahl,3)  
);
```

```
Schnittpunkt(photonenzahl,:,wand) =  
    stuetz_v + t(photonenzahl)*  
    richtungs_v(photonenzahl,:);
```

Schnittpunkte zwischen Gerade und Ebene

Schnittpunktproblematik

- Aktuelles Problem: Schnittpunkte liegen nicht (genau) auf den definierten Wänden!

schnittpunkt			
100x3 double			
	1	2	3
1	0.0031	-5	0.4729
2	-1.2947	-5	1.1211
3	-2.4268	-5	1.1154
4	0.7869	-5	-1.4128
5	0.2875	-5	-0.5674
6	-0.6807	-5	0.2071
7	-1.3765	-5.0000	1.2835
8	1.7851	-5	-0.8973
9	-0.1397	-5	-2.4252
10	-0.3230	-5	-4.1488
11	1.0849	-5	-0.3759
12	-3.3294	-5	-0.2116
13	0.5912	-5.0000	-2.1946
14	-2.0459	-5	0.6340
15	0.4363	-5	-0.6070
16	-3.8890	-5	3.1995
17	-0.0645	-5	-0.8822
18	2.4920	-5	-0.3716
19	-2.6469	-5	-0.3903
20	-1.6873	-5	0.7951

Abbildung 11: Berechnete Schnittpunkte mit verschiedenen Darstellungen

6	-0.6807	-5	0.2071
7	-1.3765	-5.0000000000000001	1.2835
8	1.7851	-5	-0.8973

12	-3.3294	-5	-0.2116
13	0.5912	-4.9999999999999999	-2.1946
14	-2.0459	-5	0.6340

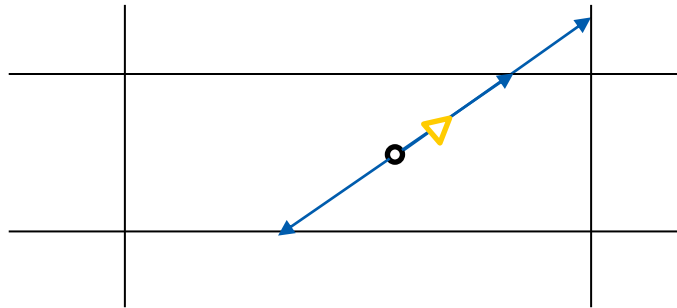
Abbildung 12: Problematische Schnittpunkte in voller Länge

Rundungsfehler ?!

Schnittpunkte zwischen Gerade und Ebene

Sonderfälle

1. Ebenen sind unendlich lang, Photon hat Schnittpunkt mit jeder Wand, die nicht parallel zum Richtungsvektor verläuft → es muss geprüft werden, dass der Schnittpunkt im Raum liegt
2. Bei 2 Schnittpunkten mit Wänden innerhalb des Raumes muss die Entfernung überprüft werden um richtige Richtung zu wählen (z.B. bei Lichtquelle im Raummittelpunkt $[0\ 0\ 0]$)
 - Welcher Schnittpunkt liegt näher am ursprünglich für die Richtung ausgewürfelten Punkt (Richtungsvektor)



- Rückgabe: `richtiger_Schnittpunkt`

Drehmatrix

Notwendigkeit

- Variable Strahlrichtung des Strahlers
- Reflexion an Wänden
 - Idee: Drehen um eine Achse im 90°
 - Wenn Wand bei $y = -8$ getroffen $\rightarrow 90^\circ$ Drehung um x-Achse

- Drehung um die x -Achse:

$$R_x(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{pmatrix}$$

- Drehung um die y -Achse:

$$R_y(\alpha) = \begin{pmatrix} \cos \alpha & 0 & \sin \alpha \\ 0 & 1 & 0 \\ -\sin \alpha & 0 & \cos \alpha \end{pmatrix}$$

- Drehung um die z -Achse:

$$R_z(\alpha) = \begin{pmatrix} \cos \alpha & -\sin \alpha & 0 \\ \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Abbildung 10: Drehmatrizen um x -, y - und z -Achse

Quelle: <https://de.wikipedia.org/wiki/Drehmatrix>

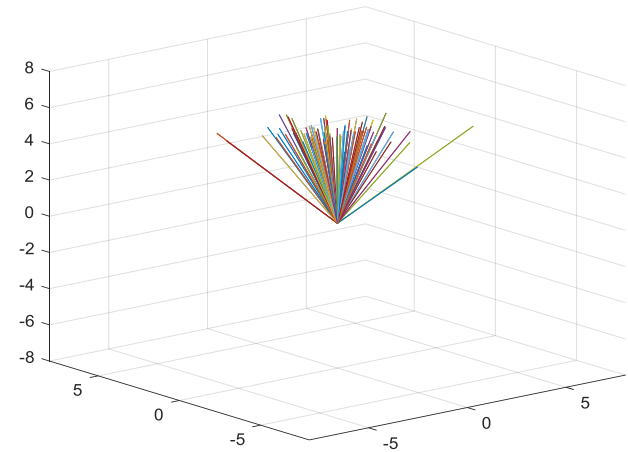


Abbildung 8: Simulation von 100 Lichtstrahlen mit $m=10$

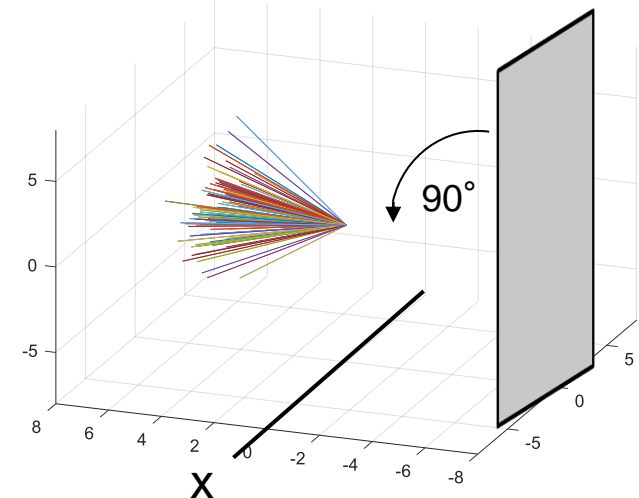


Abbildung 9: Simulation von 100 Lichtstrahlen mit $m=10$, um 90° gedreht

Drehmatrix

MATLAB Implementierung

- Ermitteln der getroffenen Wand
 - Sonderfall für Startpunkt [0 0 0]
- Definieren der Drehmatrix
- Berechnen der gedrehten Richtungsvektoren

```
groesse = groesse - 0.000001; %FÜR RUNDUNGSFEHLER
if abs(Schnittpunkt(1)) > groesse
    if Schnittpunkt(1) >= groesse
        %Wenn die Wand bei x = groesse
        %getroffen wurde muss um die y-Achse
        %gedreht werden, weil diese Wand
        %parallel zur y-Achse liegt
        drehmatrix = BerDrehmatrix(-pi/2,2);
    else
        drehmatrix = BerDrehmatrix(pi/2,2);
    end
    ...
function [drehmatrix] =
    BerDrehmatrix(alpha, dimension)
    ...
end

%% Drehen des Richtungsvektors
%% mit der bestimmten Drehmatrix
i=1:n;
gewuerfelte_richtung=[u_x,u_y,u_z]';
richtungsvektor =
drehmatrix*gewuerfelte_richtung(:,i);
```

Überlebensmatrix

Hintergrund

Beim auftreten des Lichtpaketes auf die Wand wird der Anteil $1 - \rho$ der Leistung absorbiert, die Wand hat eine Reflektivität ρ ($0 < \rho < 1$)

Umsetzungsmöglichkeiten:

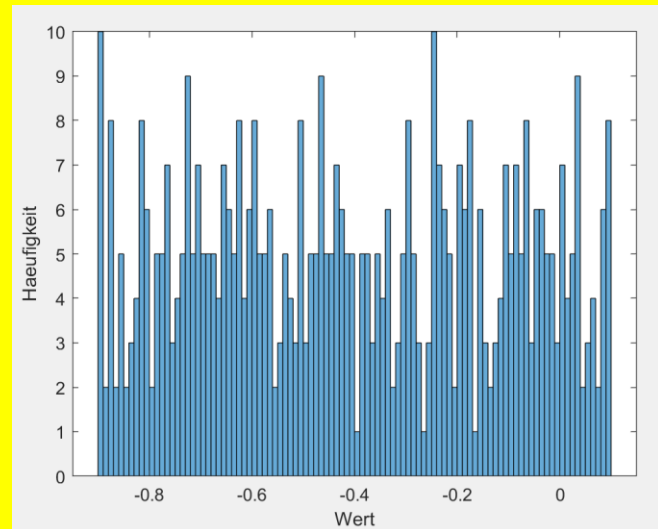
1. Das Lichtpaket hat zu Beginn eine Leistung von 1 und wird bei jeder Reflexion um ρ schwächer.
 2. **Das Photon hat eine konstante Leistung und wird mit der Wahrscheinlichkeit $1 - \rho$ von der Wand absorbiert und mit der Wahrscheinlichkeit ρ reflektiert**
- Umsetzung als „Überlebensmatrix“

Quellen: Lichtausbreitung in Räumen: Skizzen zur Monte-Carlo-Simulation", von Henrik Schulze, 3.11.2021

Überlebensmatrix

- Anzahl der gewürfelten Photonen: $n = 500$
- Reflexionswahrscheinlichkeit: $p = 0.1$
- $n \times 1$ Matrix von standard-gleichverteilten Zufallszahlen zwischen 0 und 1
- Verschieben des Wertebereiches entsprechend der Reflexionswahrscheinlichkeit

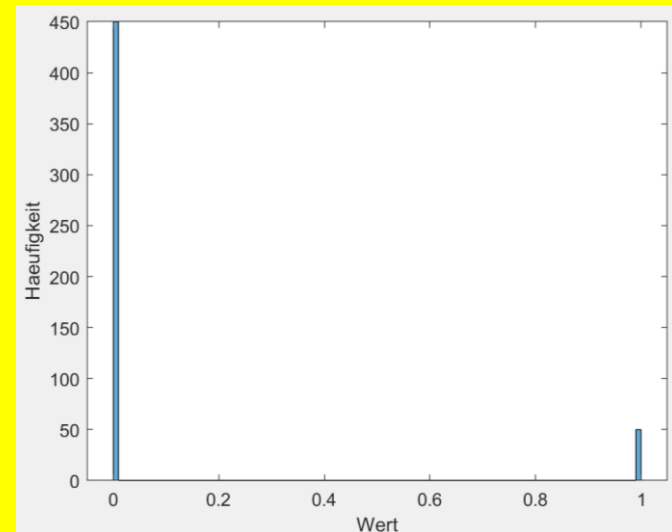
```
ueberlebens_matrix = ceil(rand(n,1) -  
(1-p)).*ueberlebens_matrix;
```



Überlebensmatrix

- Anzahl der gewürfelten Photonen: $n = 500$
- Reflexionswahrscheinlichkeit: $p = 0.1$
- Aufrunden schreibt 0 (nicht reflektieren) oder 1 (reflektieren) in die Matrix
- Nächsten Schnittpunkt berechnen für reflektierte Photonen
- Absorbierte Photonen werden eliminiert

```
ueberlebens_matrix = ceil(rand(n,1) -  
(1-p)).*ueberlebens_matrix;
```



```
if ueberlebens_matrix(i)==1  
    naechster_schnittpunkt(i,:) =  
        BerSchnittpunkt(stuetz_v, ...);  
else  
    naechster_schnittpunkt(i,:) =  
        nan(1,3); ...
```

Abschließen eines Durchlaufes

- Schnittpunkt wird zum neuen Startpunkt
- Für Analyse und Visualisierung werden alle Schnittpunkte gespeichert

```
startpunkte=naechster_schnittpunkt;
```

```
alle_schnittpunkte =  
[alle_schnittpunkte;  
rmmissing(naechster_schnittpunkt)];
```

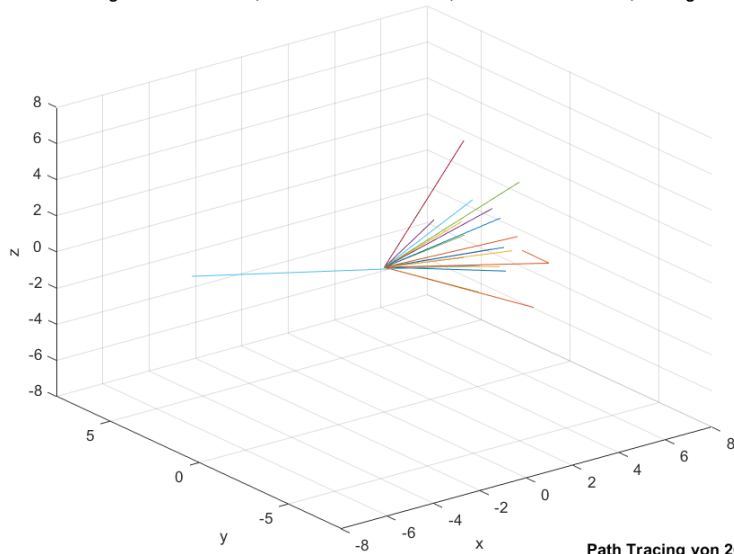
- Visualisierung als:
 - Path Tracing
 - HeatmapMit 10000 Photonen
- Parameter:
 - Reflexionswahrscheinlichkeit: **0,8** (weiße Wand); 0,3 (mittelgraue Wand); 0,125 (dunkle Eiche)
 - Raumgröße: 1; **5**; 10
 - Lambert-Parameter: 1; **5**; 100

Quellen: https://archive.arch.ethz.ch/bph/Filep/Licht/Licht_Bauwerk/Kennwerte/ReflexionsgradBaumat.html; Zugriff am 06.01.2022 um 18:07

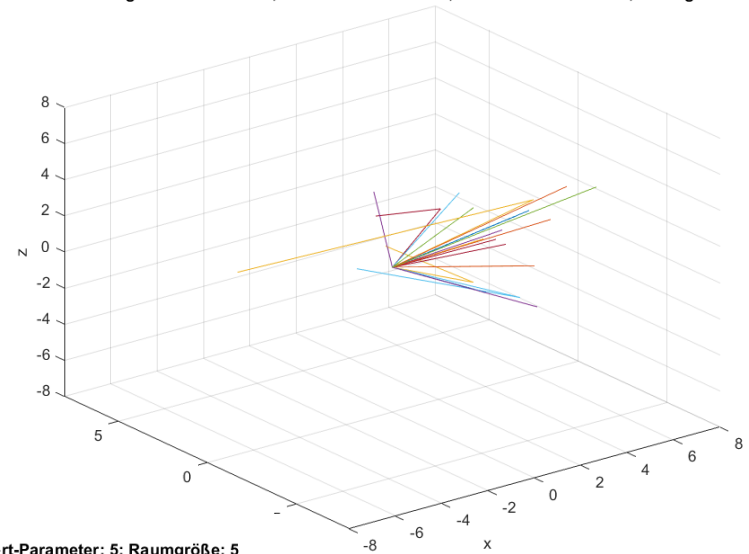
Variation des Reflexionsfaktors

Path Tracing

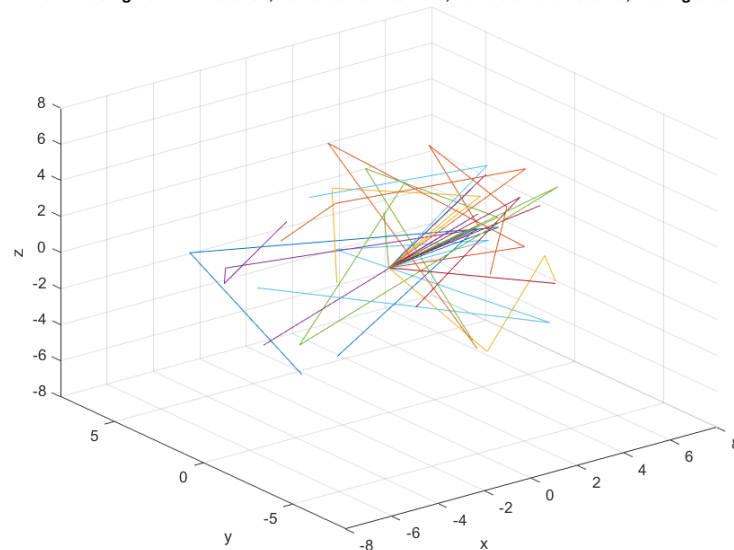
Path Tracing von 20 Photonen, Reflexionsfaktor: 0.125; Lambert-Parameter: 5; Raumgröße: 5



Path Tracing von 20 Photonen, Reflexionsfaktor: 0.3; Lambert-Parameter: 5; Raumgröße: 5



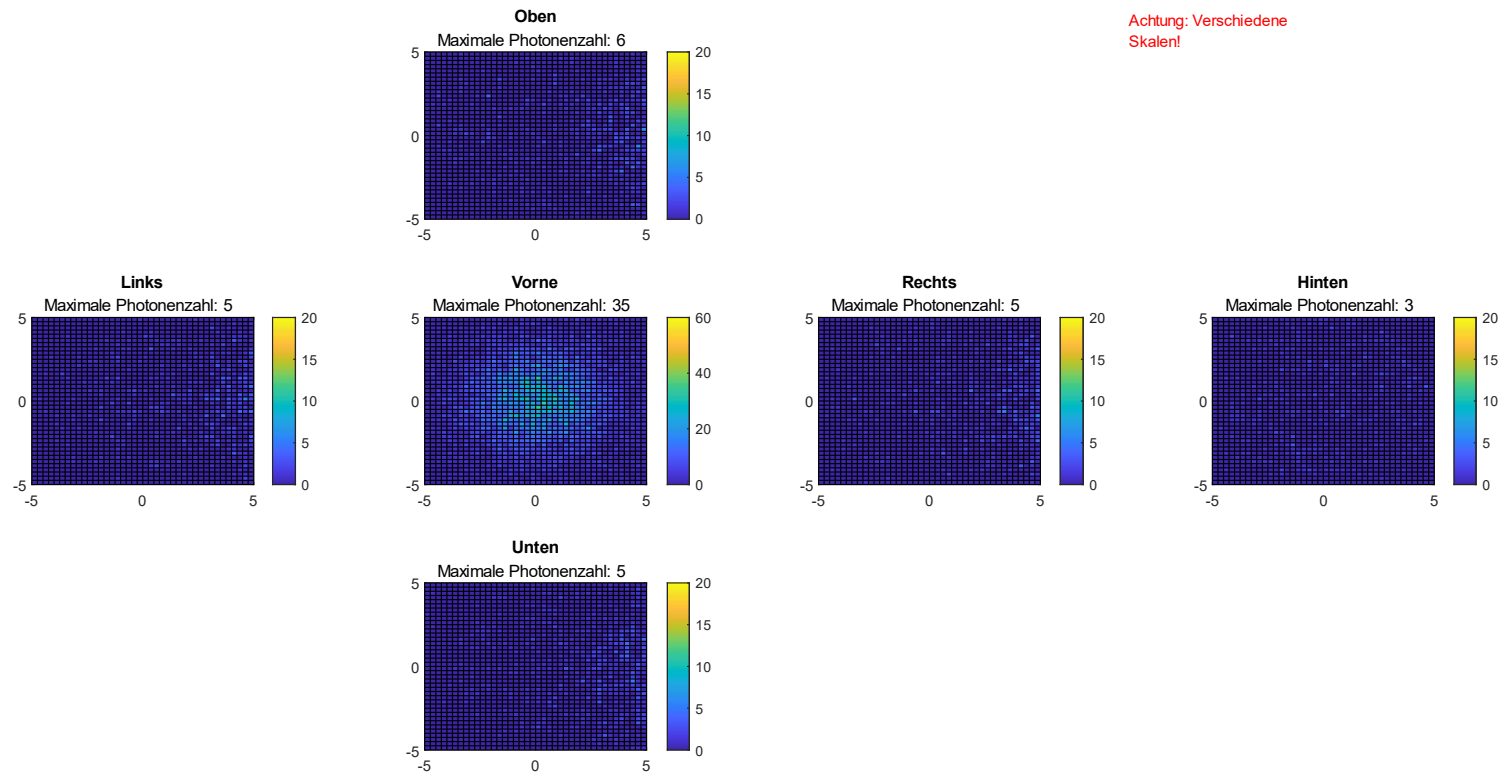
Path Tracing von 20 Photonen, Reflexionsfaktor: 0.8; Lambert-Parameter: 5; Raumgröße: 5



Variation des Reflexionsfaktors

Heatmap I

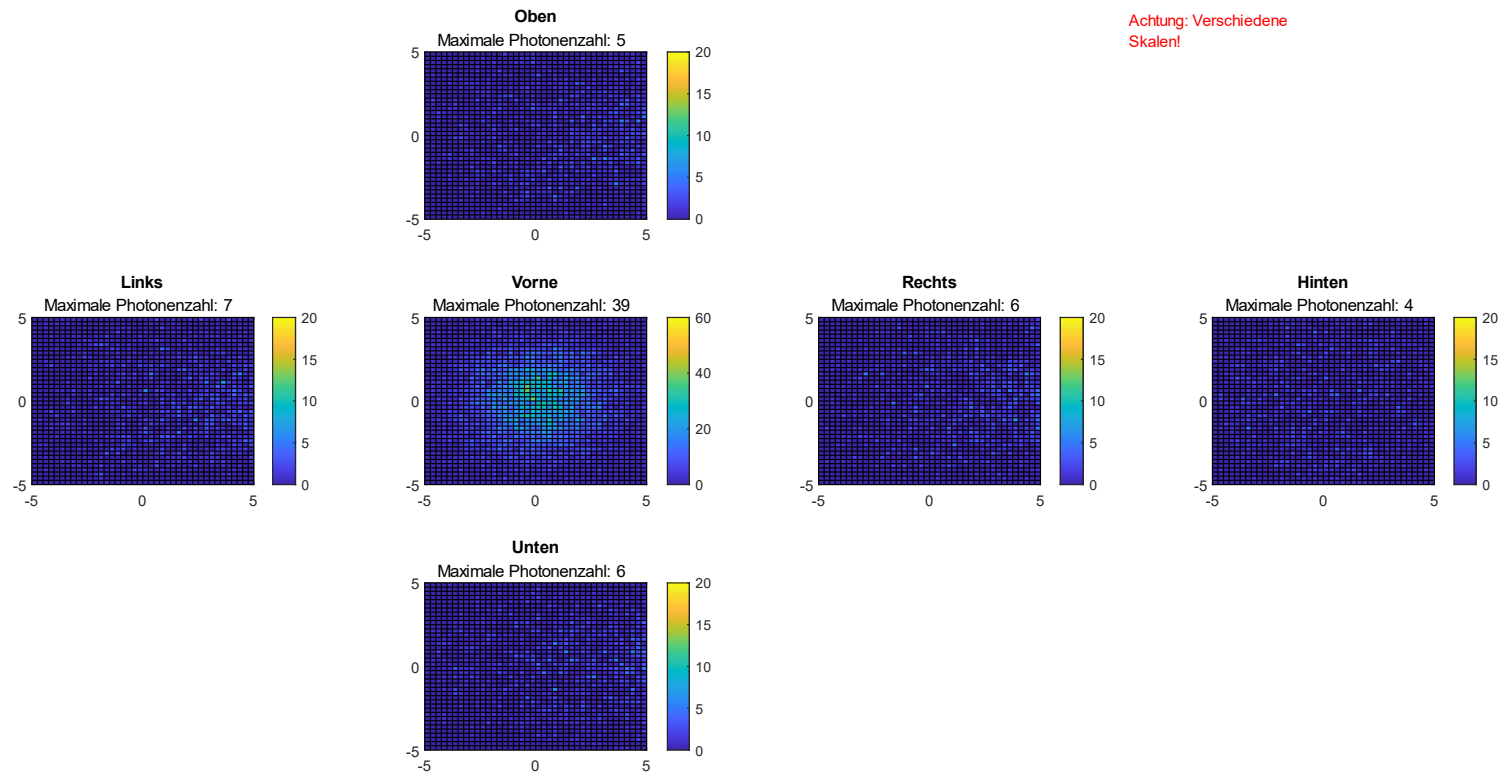
Heatmap von 10000 Photonen, Reflektionsfaktor: 0.125; Lambert-Parameter: 5; Raumgröße: 5



Variation des Reflexionsfaktors

Heatmap II

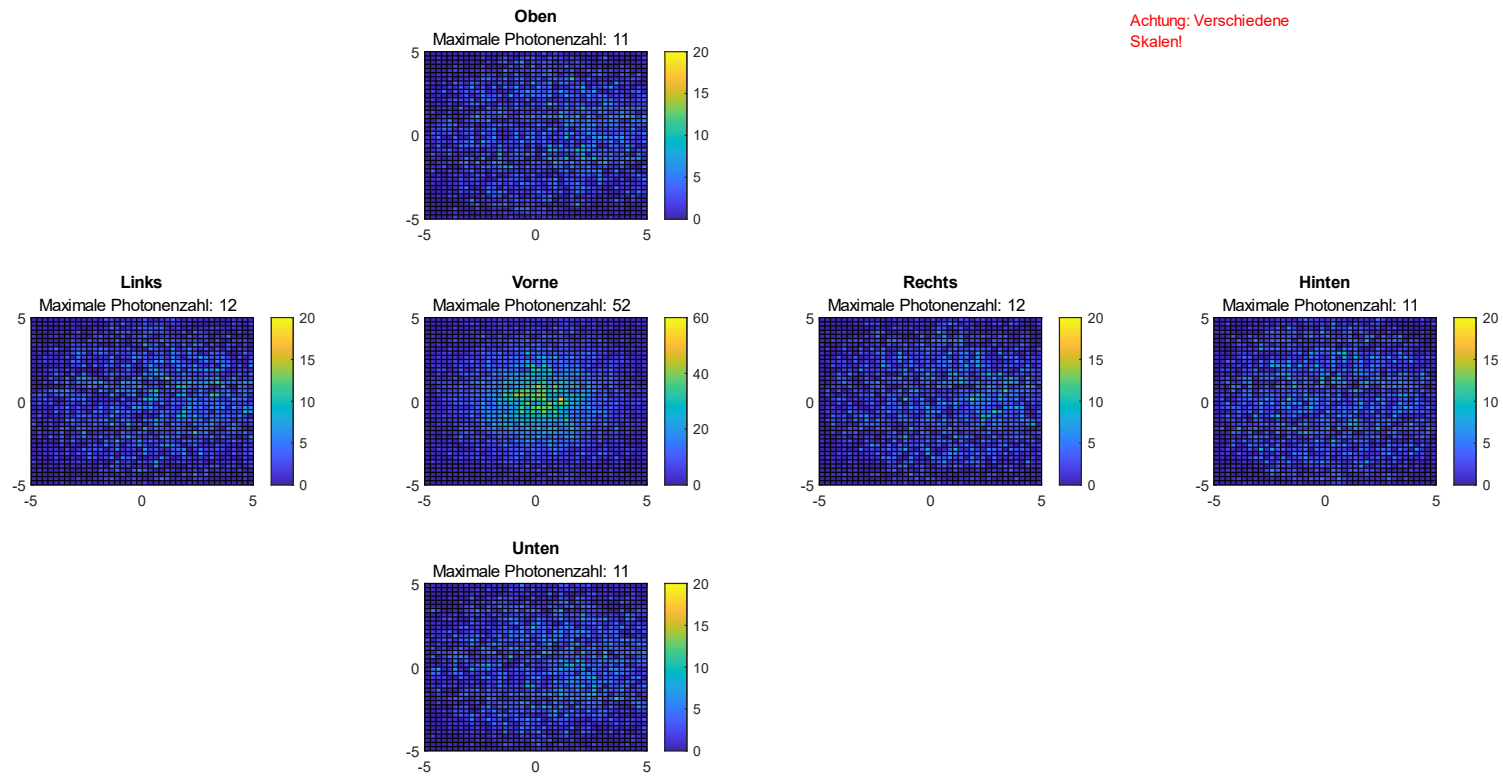
Heatmap von 10000 Photonen, Reflektionsfaktor: 0.3; Lambert-Parameter: 5; Raumgröße: 5



Variation des Reflexionsfaktors

Heatmap III

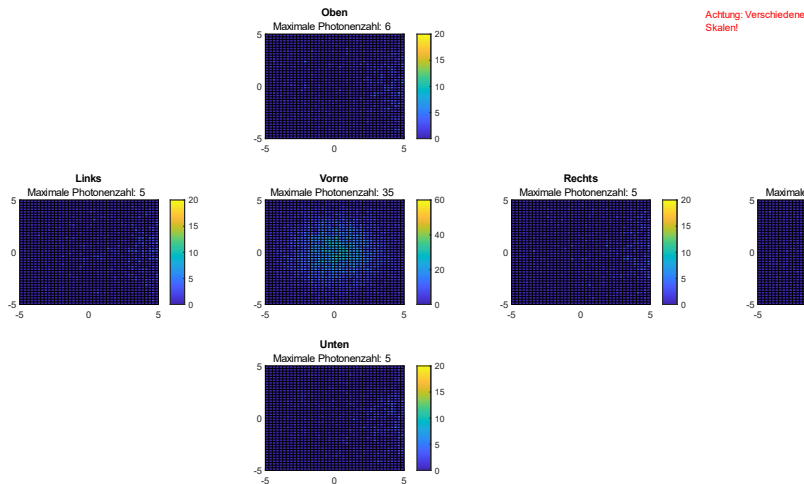
Heatmap von 10000 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 5; Raumgröße: 5



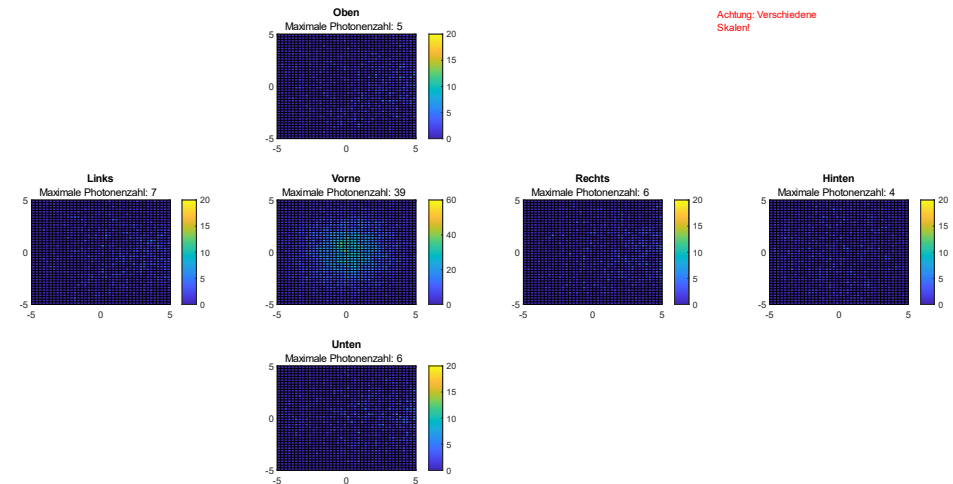
Variation des Reflexionsfaktors

Heatmap Übersicht

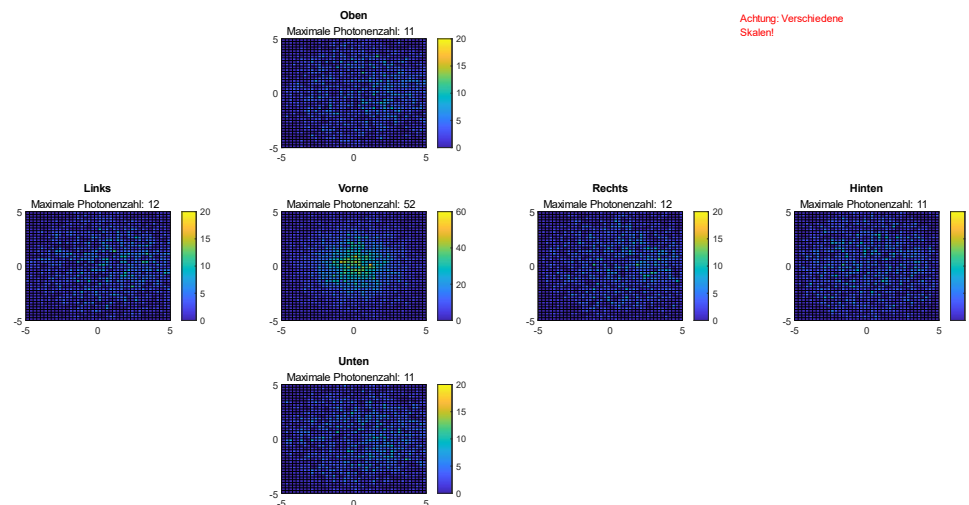
Heatmap von 10000 Photonen, Reflexionsfaktor: 0.125; Lambert-Parameter: 5; Raumgröße: 5



Heatmap von 10000 Photonen, Reflexionsfaktor: 0.3; Lambert-Parameter: 5; Raumgröße: 5



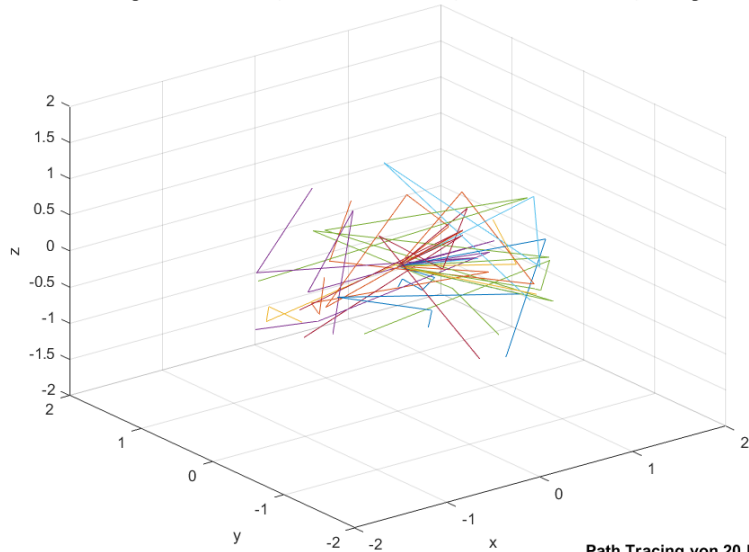
Heatmap von 10000 Photonen, Reflexionsfaktor: 0.8; Lambert-Parameter: 5; Raumgröße: 5



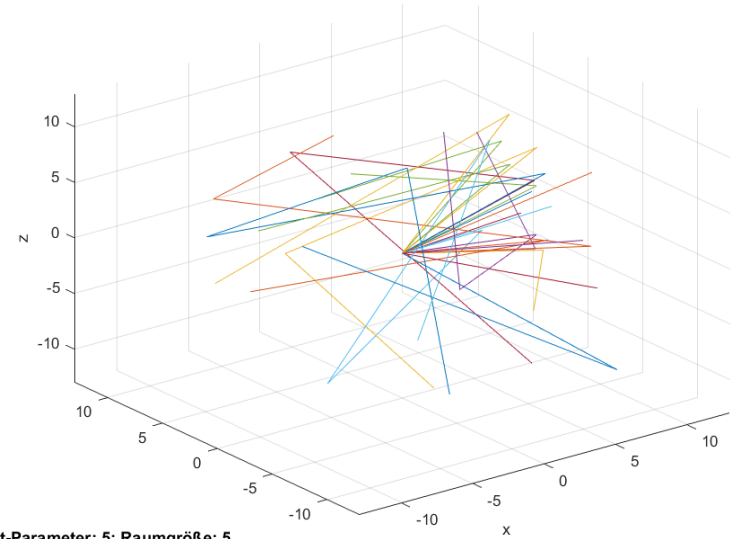
Variation der Raumgröße

Path Tracing

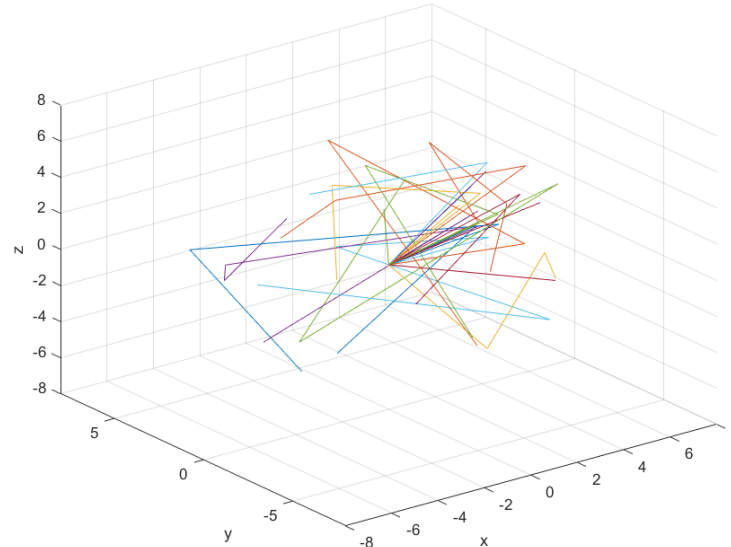
Path Tracing von 20 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 5; Raumgröße: 1



Path Tracing von 20 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 5; Raumgröße: 10



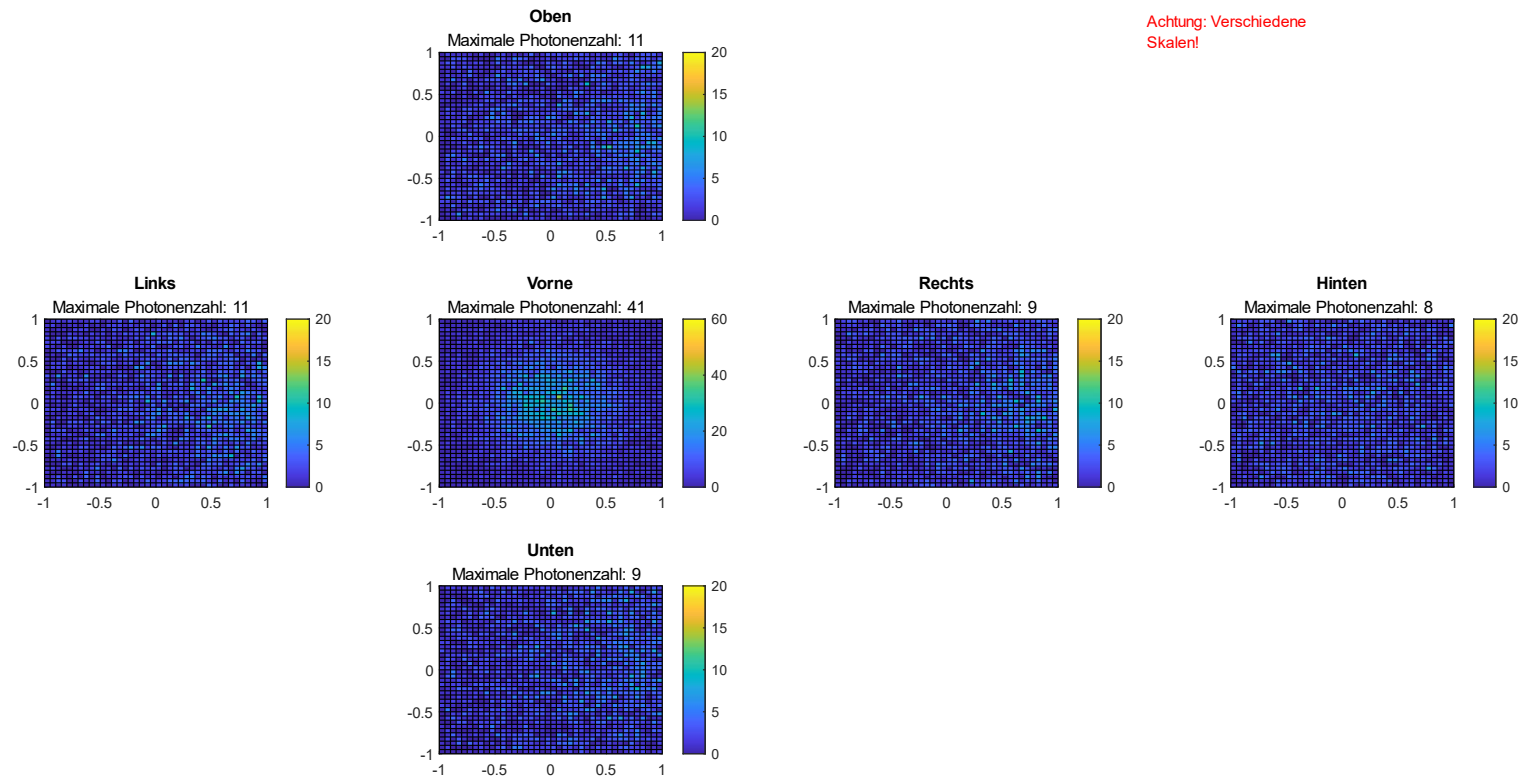
Path Tracing von 20 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 5; Raumgröße: 5



Variation der Raumgröße

Heatmap I

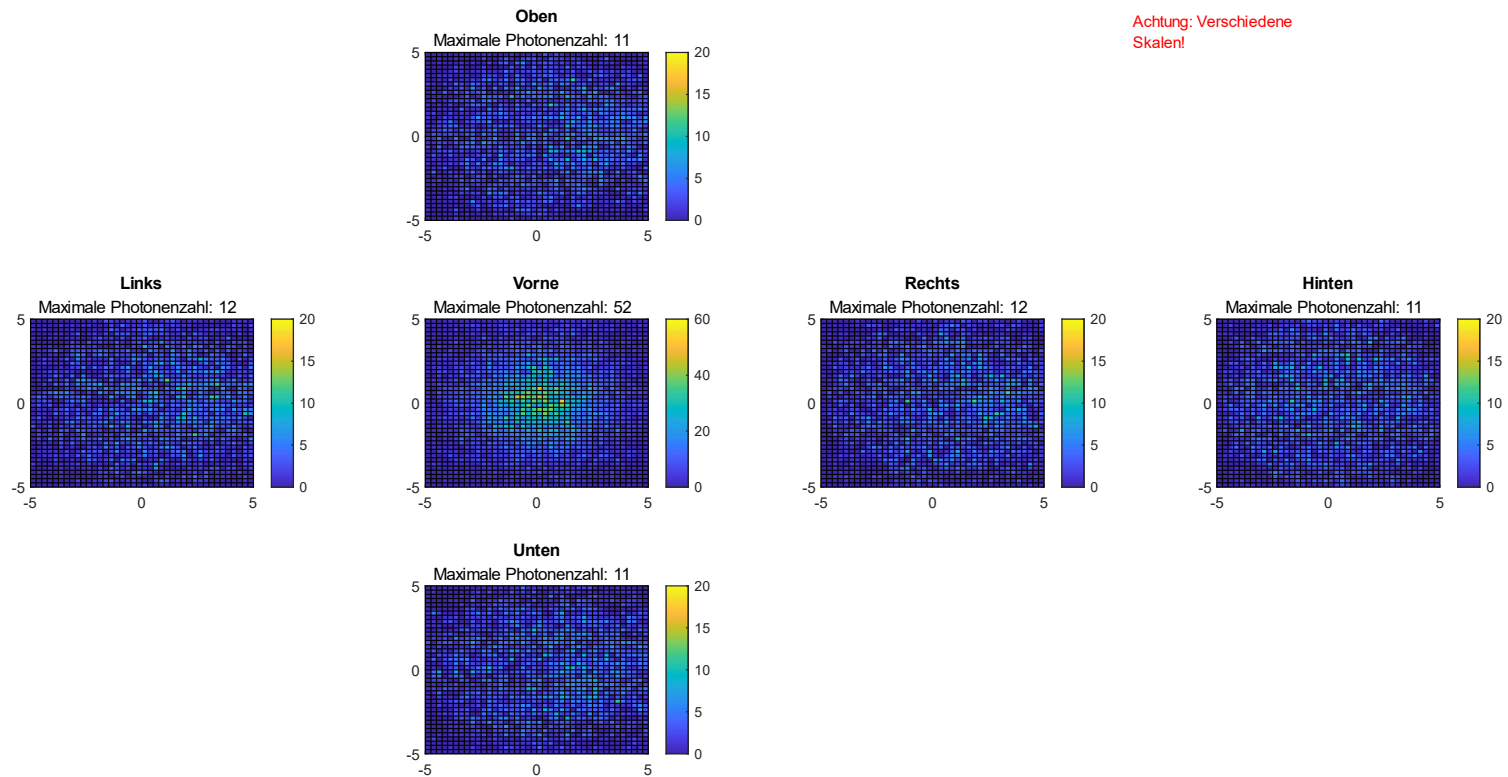
Heatmap von 10000 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 5; Raumgröße: 1



Variation der Raumgröße

Heatmap II

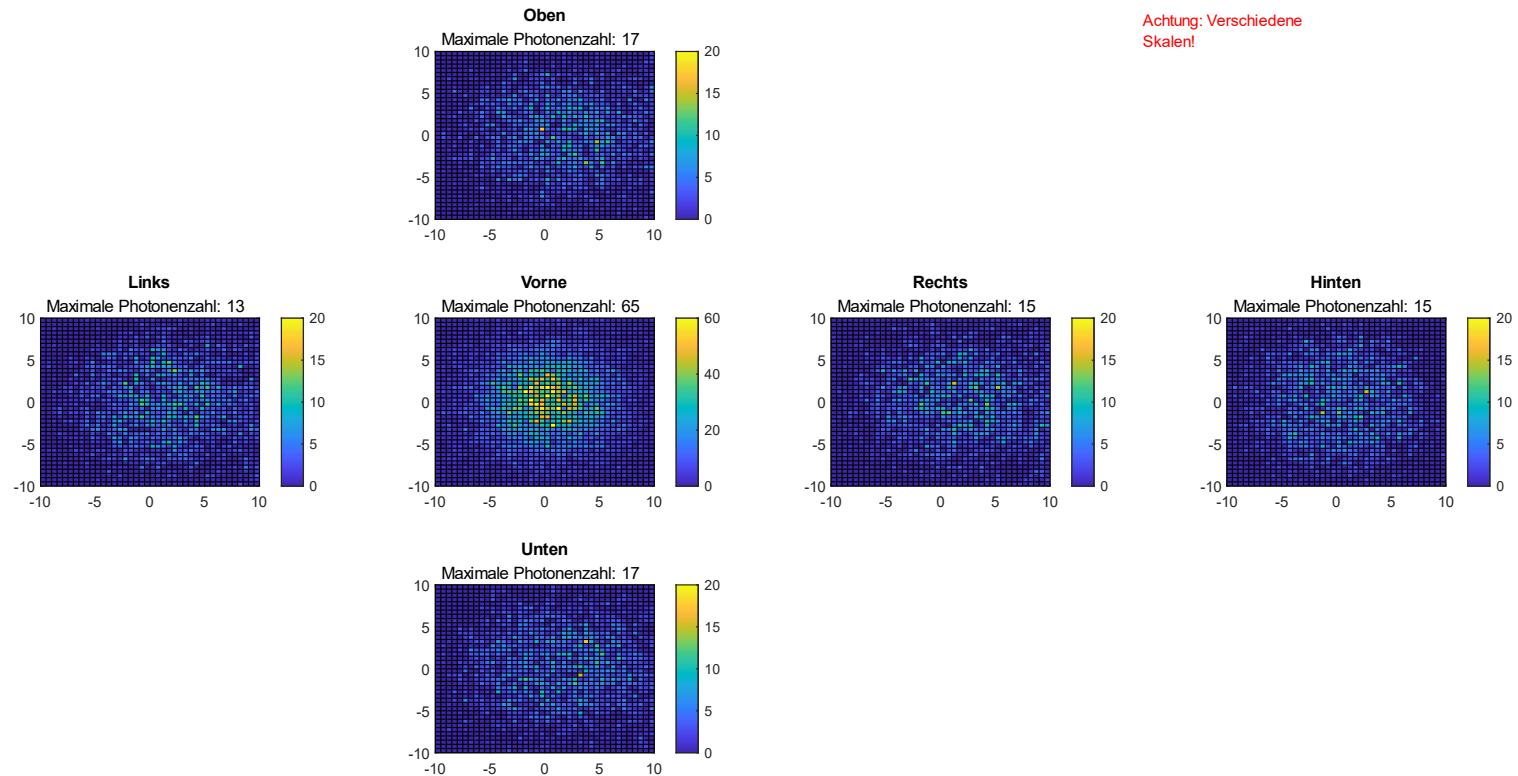
Heatmap von 10000 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 5; Raumgröße: 5



Variation der Raumgröße

Heatmap III

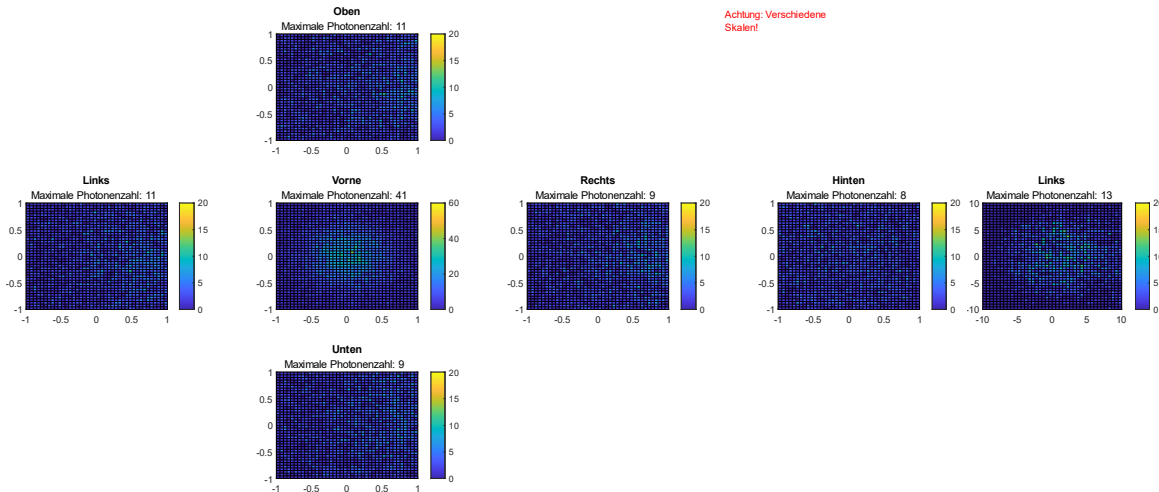
Heatmap von 10000 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 5; Raumgröße: 10



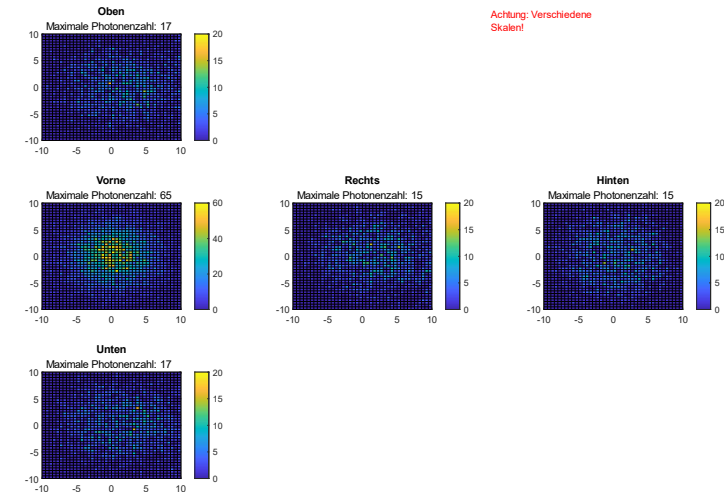
Variation der Raumgröße

Heatmap Übersicht

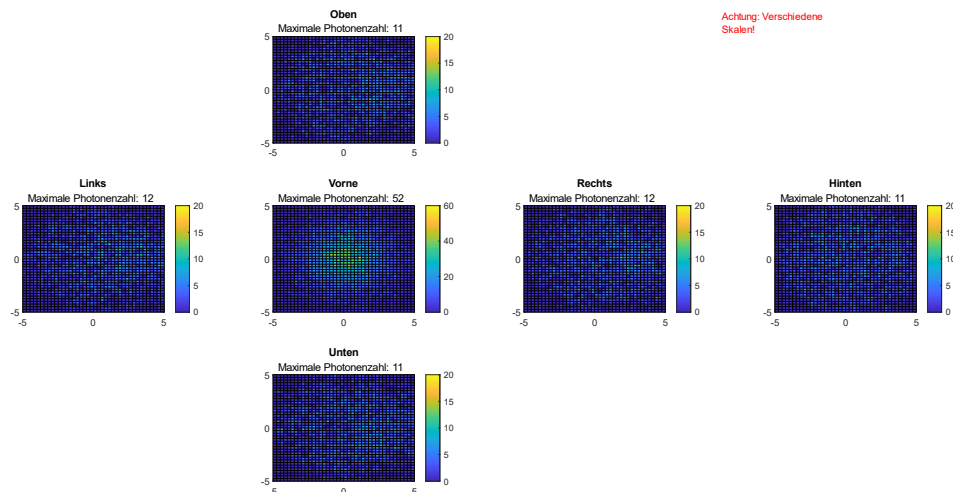
Heatmap von 10000 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 5; Raumgröße: 1



Heatmap von 10000 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 5; Raumgröße: 10



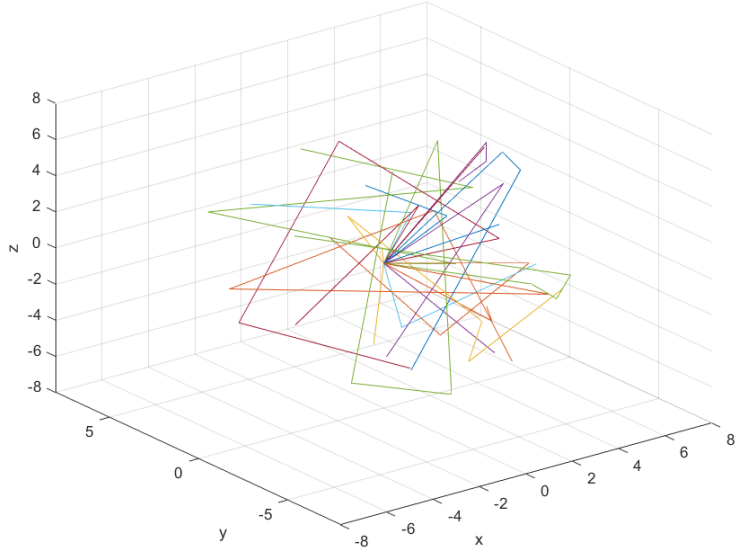
Heatmap von 10000 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 5; Raumgröße: 5



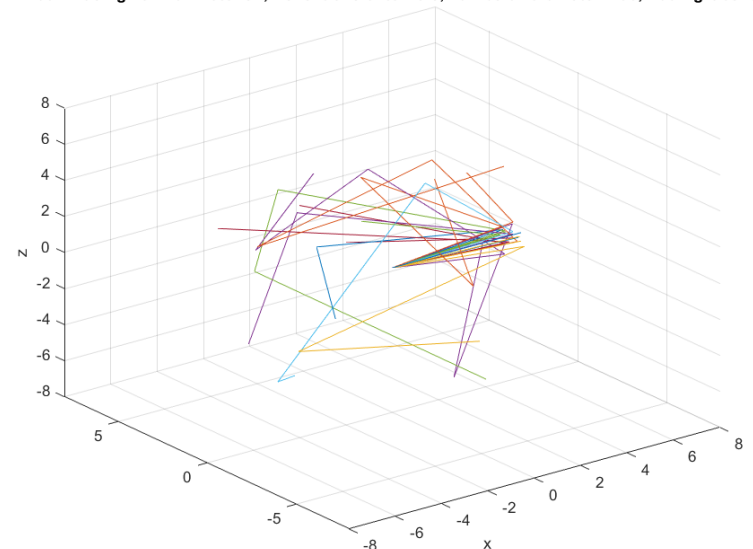
Variation des Lambert-Parameters

Path Tracing

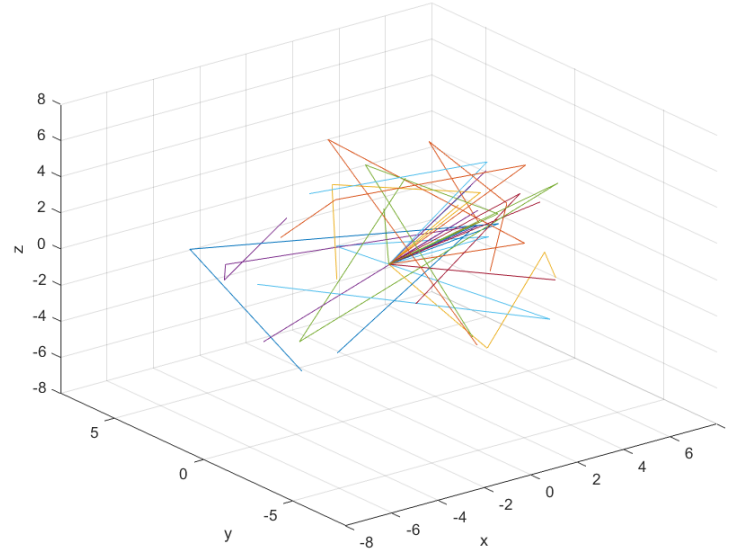
Path Tracing von 20 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 1; Raumgröße: 5



Path Tracing von 20 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 100; Raumgröße: 5



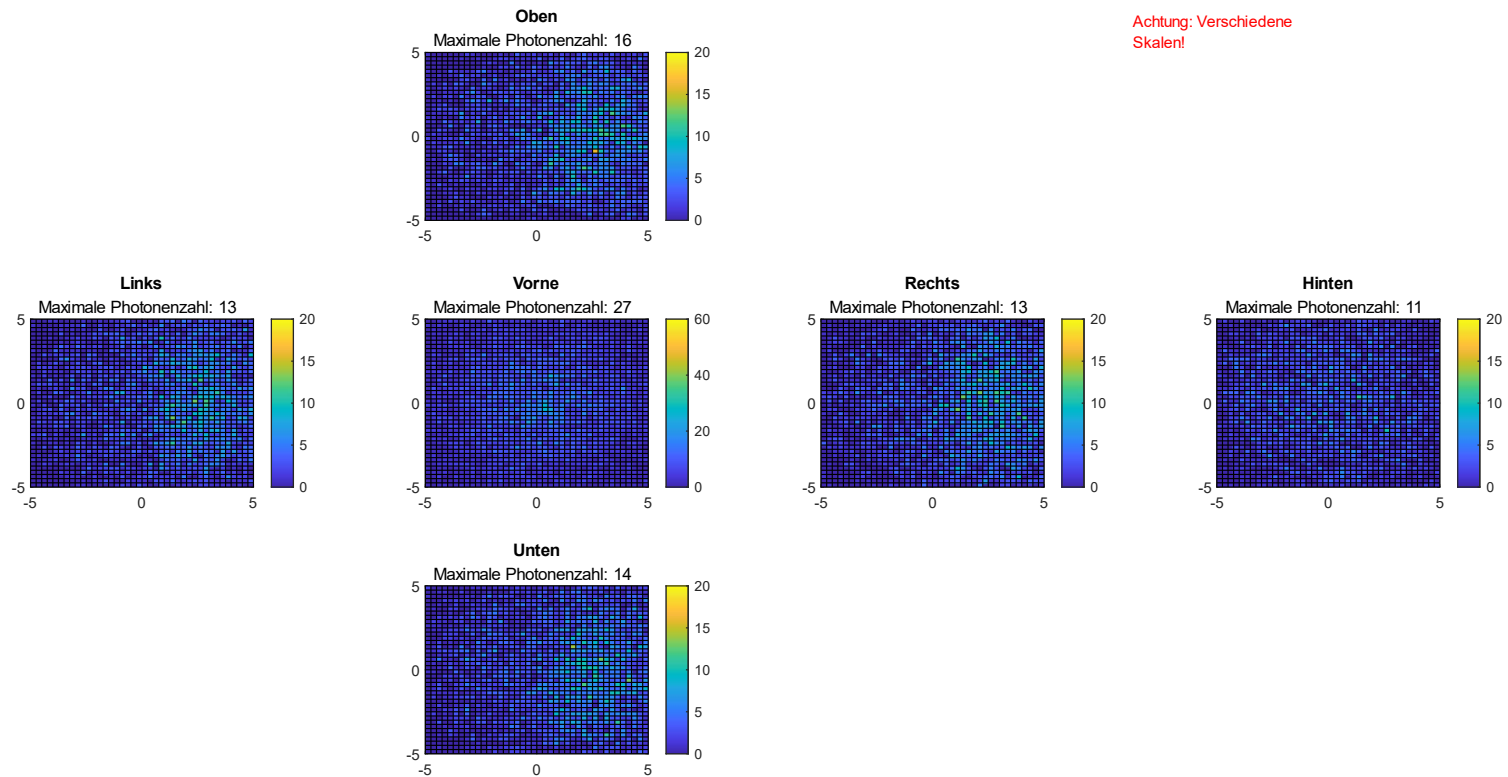
Path Tracing von 20 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 5; Raumgröße: 5



Variation des Lambert-Parameters

Heatmap I

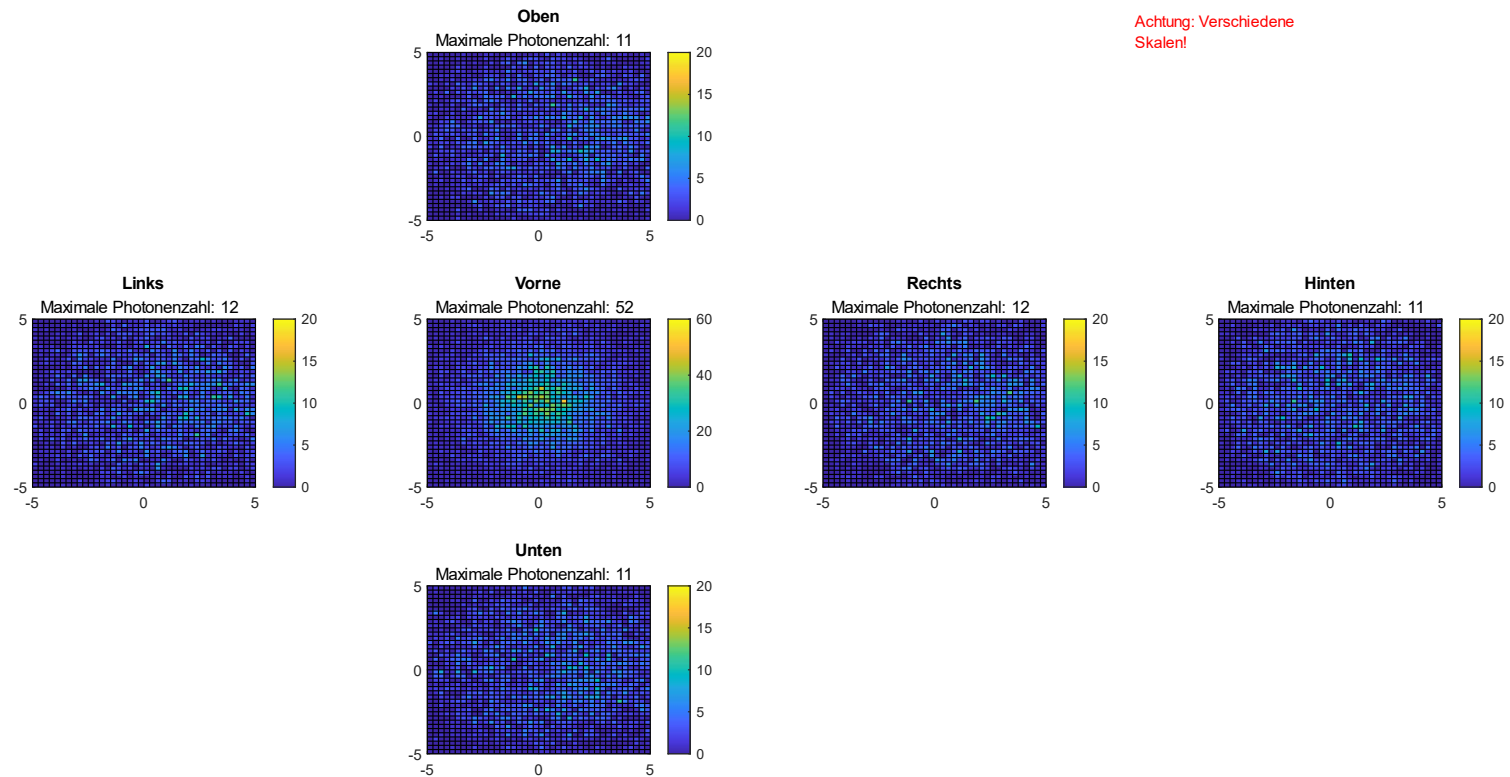
Heatmap von 10000 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 1; Raumgröße: 5



Variation des Lambert-Parameters

Heatmap II

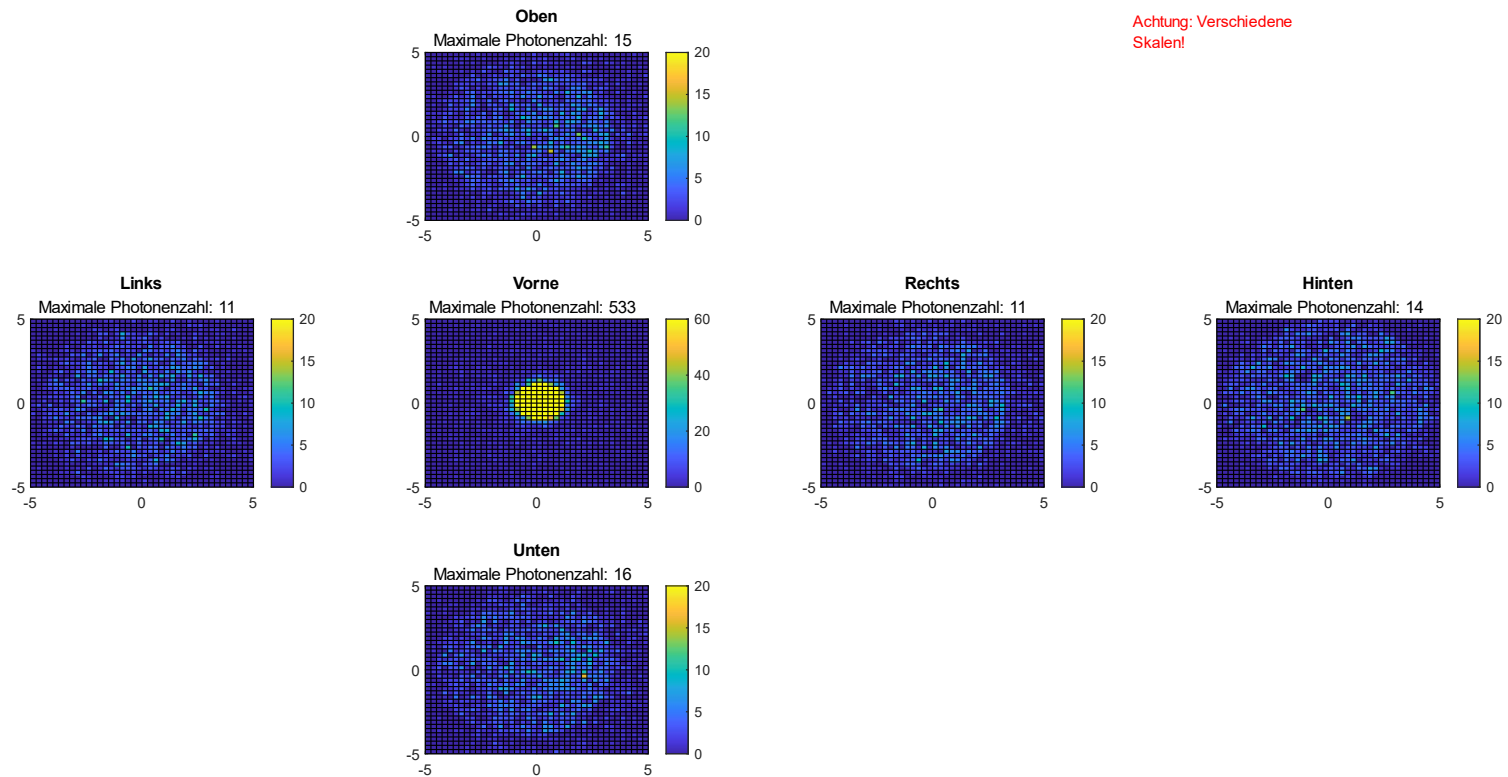
Heatmap von 10000 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 5; Raumgröße: 5



Variation des Lambert-Parameters

Heatmap III

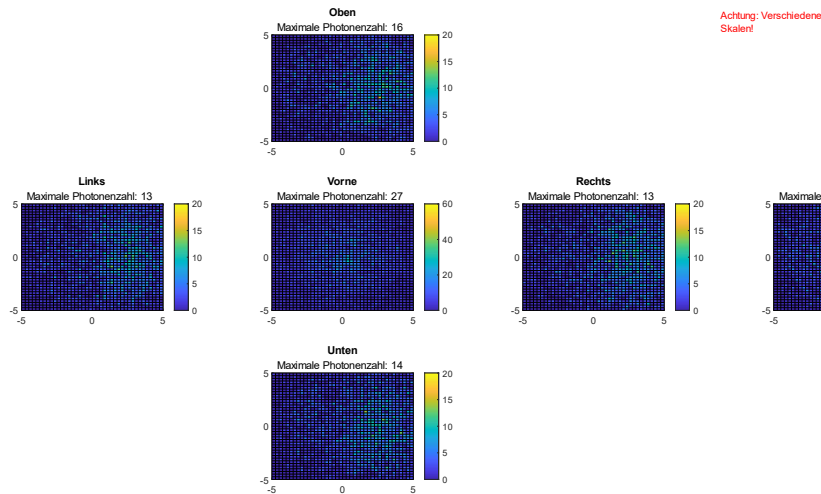
Heatmap von 10000 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 100; Raumgröße: 5



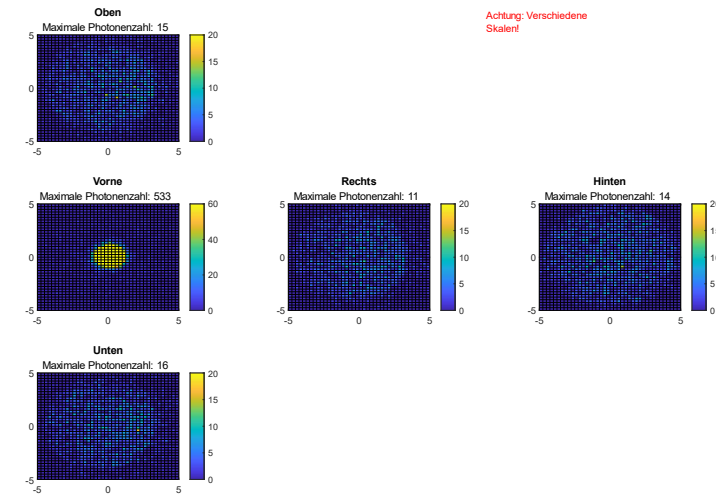
Variation des Lambert-Parameters

Heatmap Übersicht

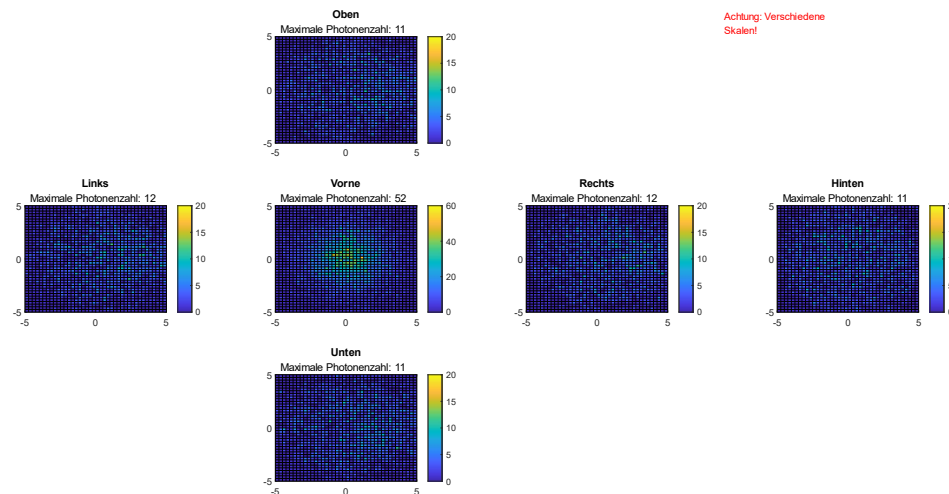
Heatmap von 10000 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 1; Raumgröße: 5



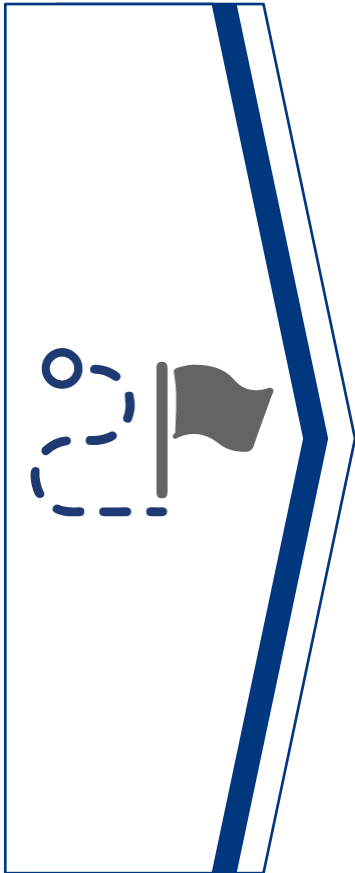
Heatmap von 10000 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 100; Raumgröße: 5



Heatmap von 10000 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 5; Raumgröße: 5



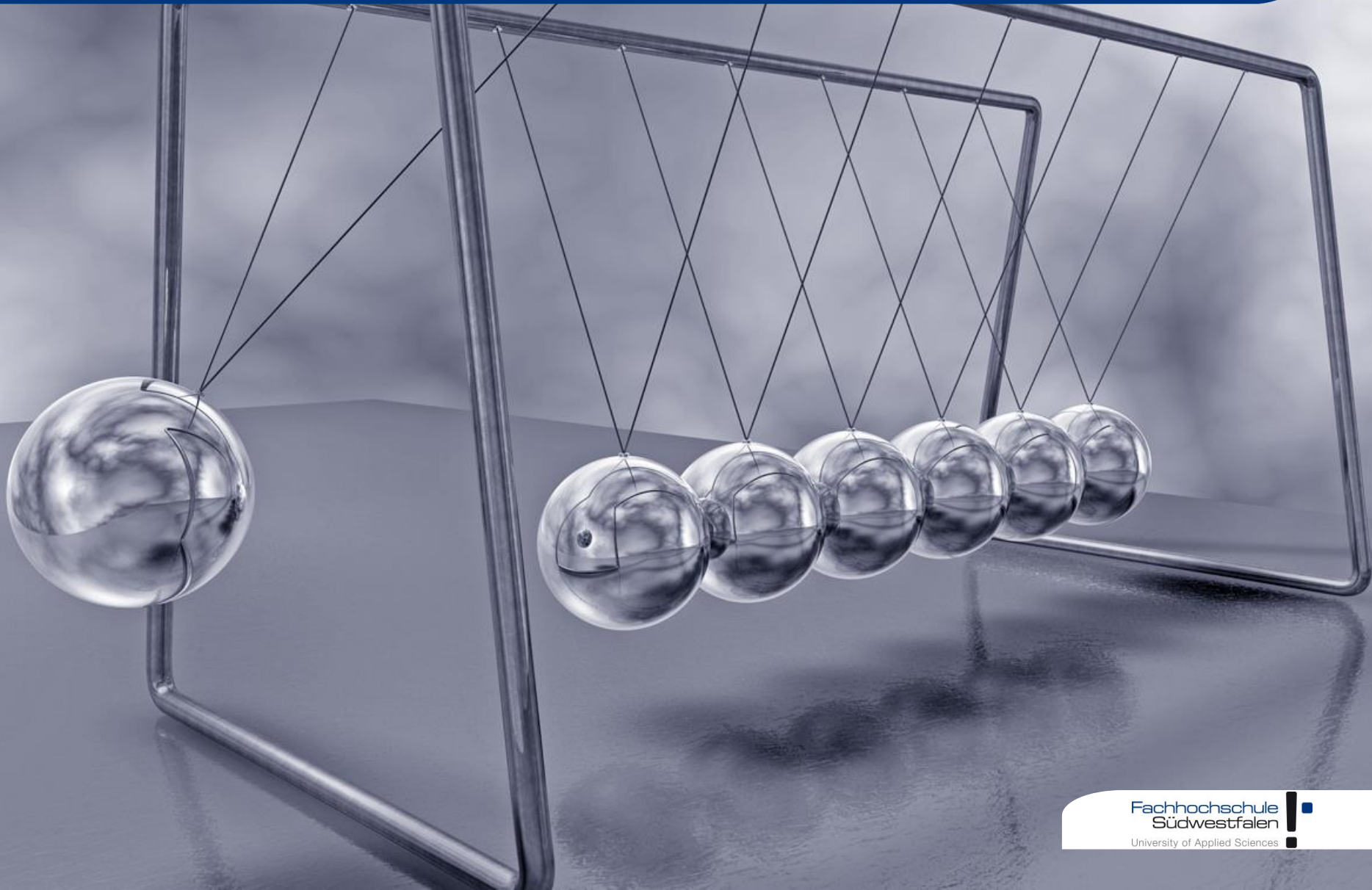
Zusammenfassung & Fazit



- Erfolgreiches Implementieren des Grundkonzeptes
- Visualisierung liefert nachvollziehbare Ergebnisse
- Einfluss der unterschiedlichen Variablen kann untersucht werden:
 - Raumgröße
 - Lambert-Parameter
 - Reflektivität

Nächste Schritte

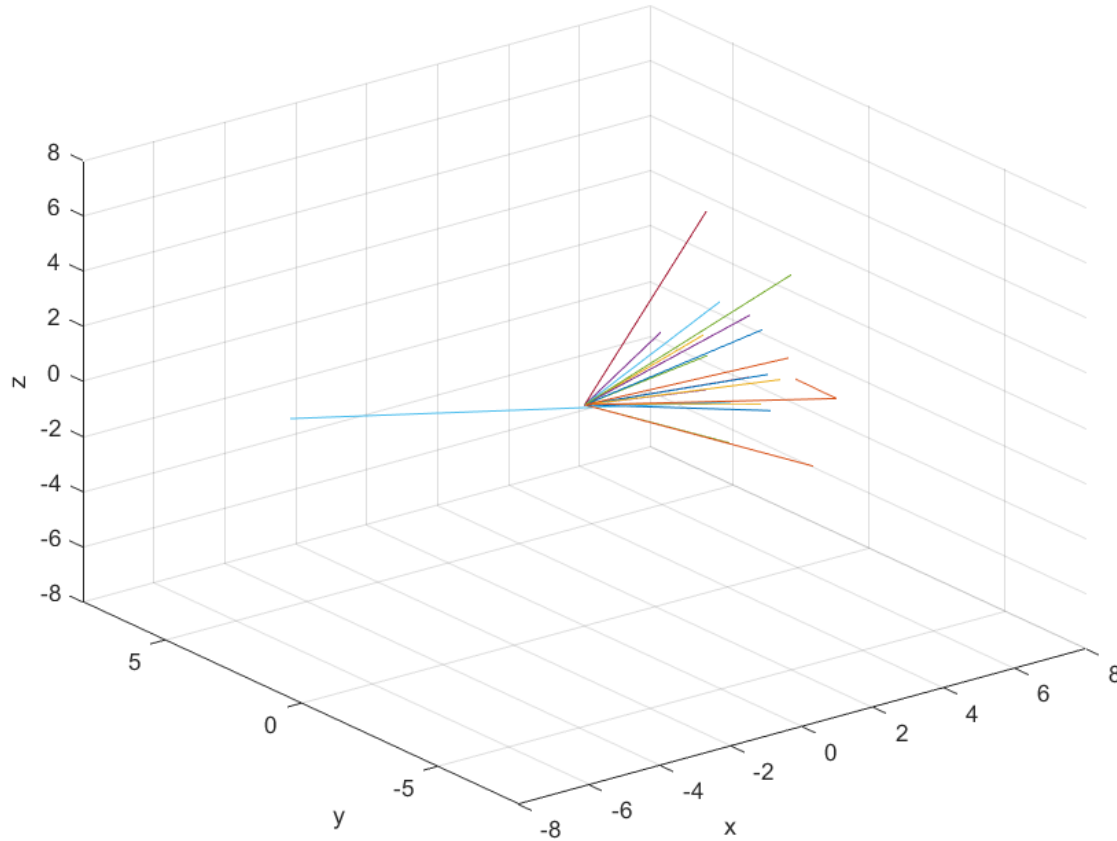
1. Rundungsfehler analysieren
2. Viele Durchläufe mitteln um repräsentative Ergebnisse zu erzielen (Monte-Carlo)
3. Laufzeit Optimieren, Schleifen eliminieren
4. Alternative Betrachtung der Reflektivität:
 - Leistungsverringerung als Alternative zur Überlebensmatrix
 - Wertung nach dem Ort des letzten Schnittpunktes
5. Variable Startposition und Richtung des Strahlers
6. Einheiten anpassen (Lichtstärke statt Anzahl der Photonen)



Variation des Reflexionsfaktors

Path Tracing I

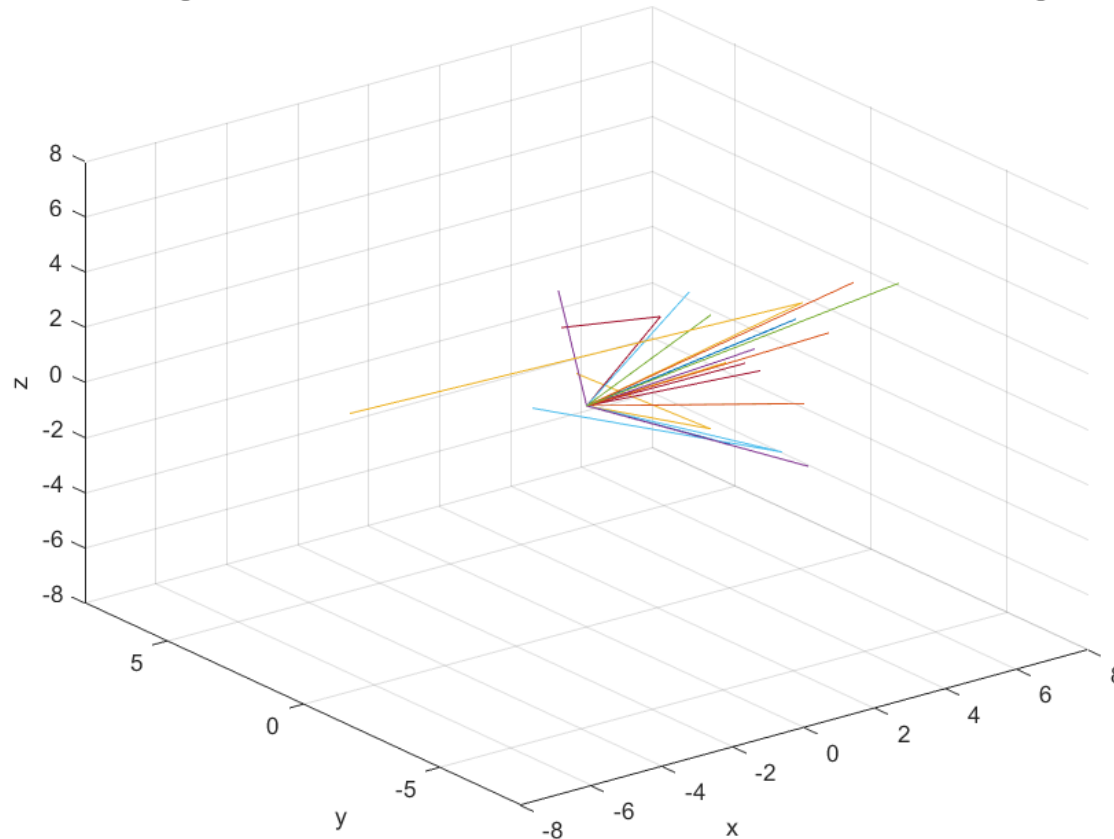
Path Tracing von 20 Photonen, Reflexionsfaktor: 0.125; Lambert-Parameter: 5; Raumgröße: 5



Variation des Reflexionsfaktors

Path Tracing II

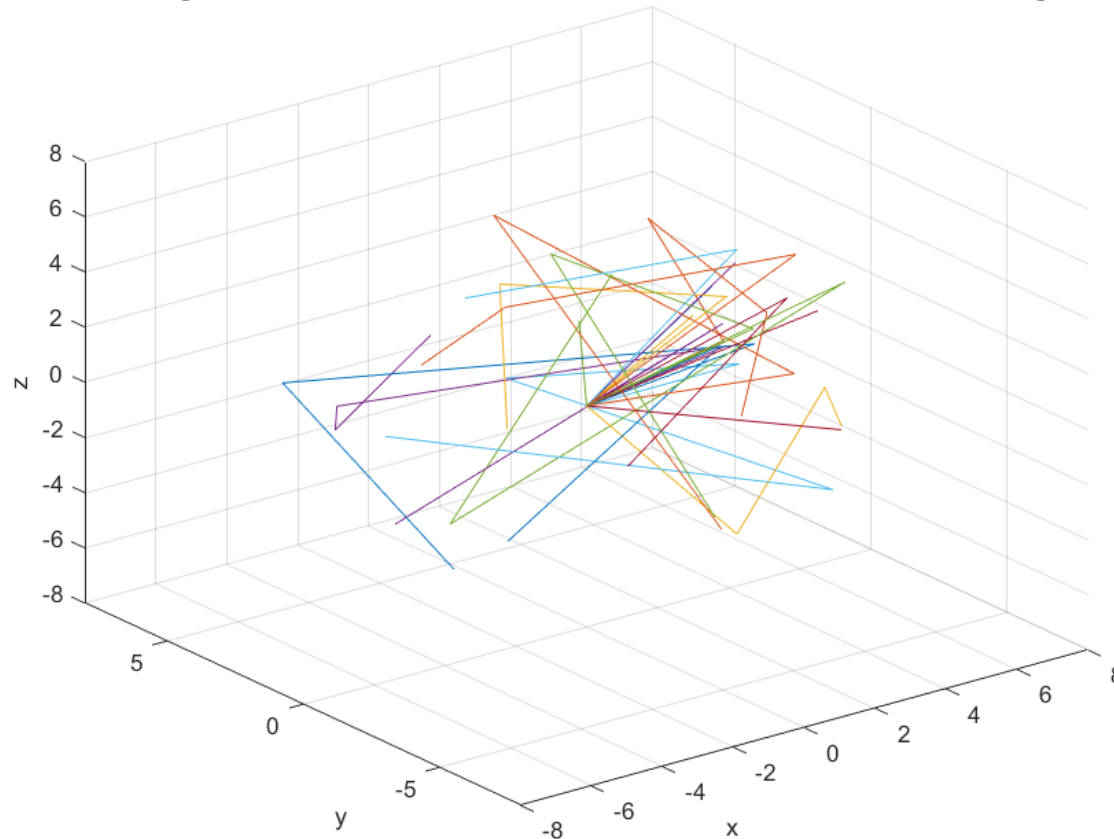
Path Tracing von 20 Photonen, Reflexionsfaktor: 0.3; Lambert-Parameter: 5; Raumgröße: 5



Variation des Reflexionsfaktors

Path Tracing III

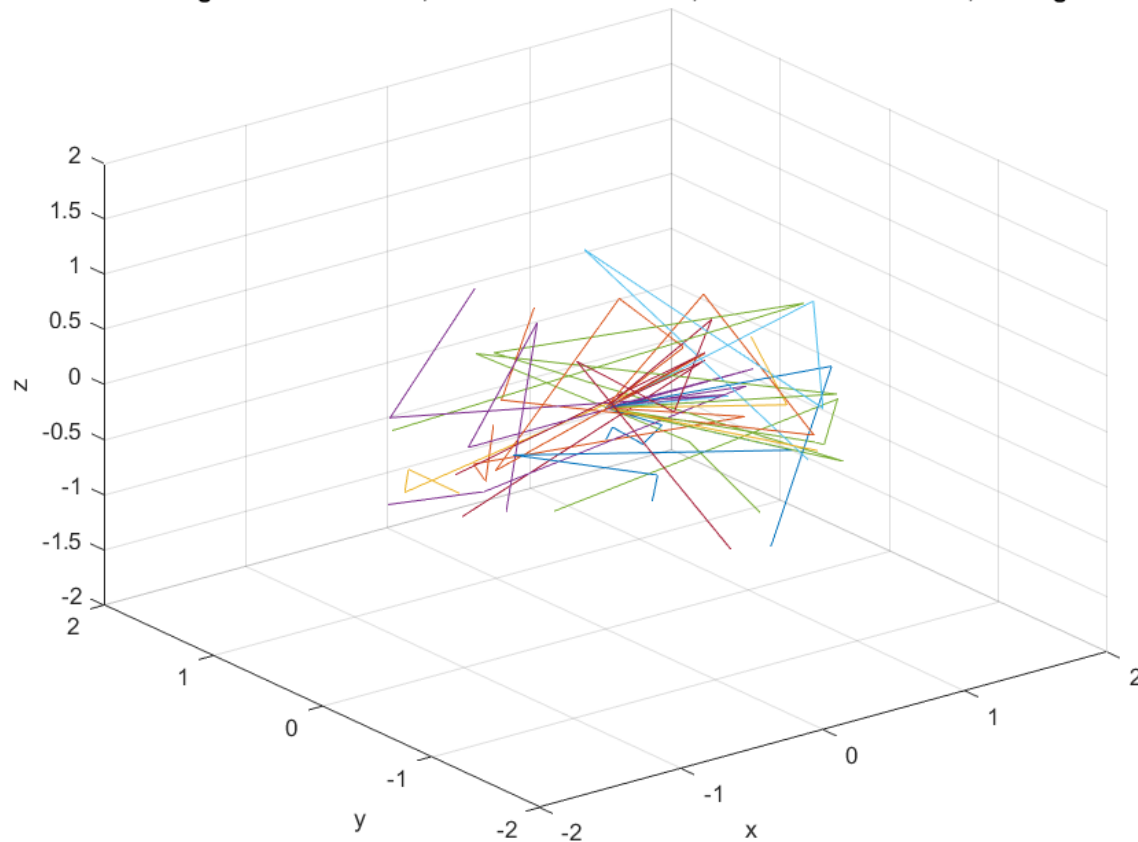
Path Tracing von 20 Photonen, Reflexionsfaktor: 0.8; Lambert-Parameter: 5; Raumgröße: 5



Variation der Raumgröße

Path Tracing I

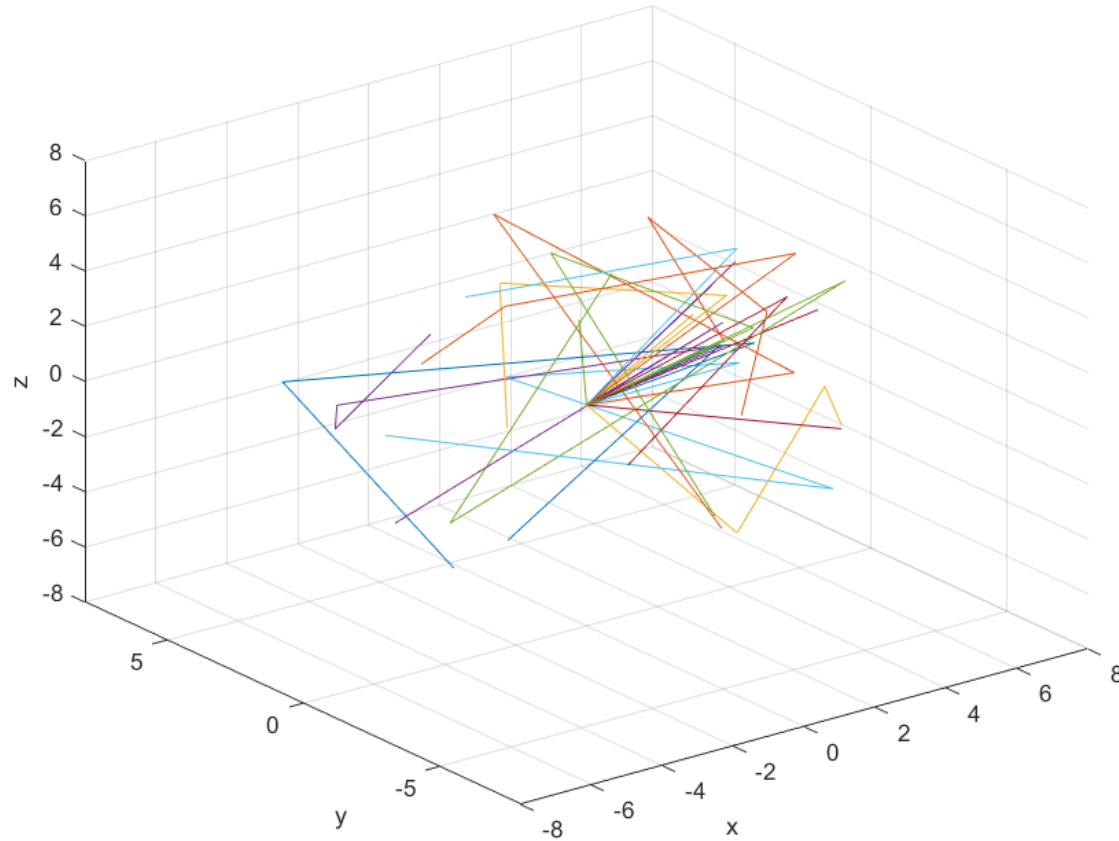
Path Tracing von 20 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 5; Raumgröße: 1



Variation der Raumgröße

Path Tracing II

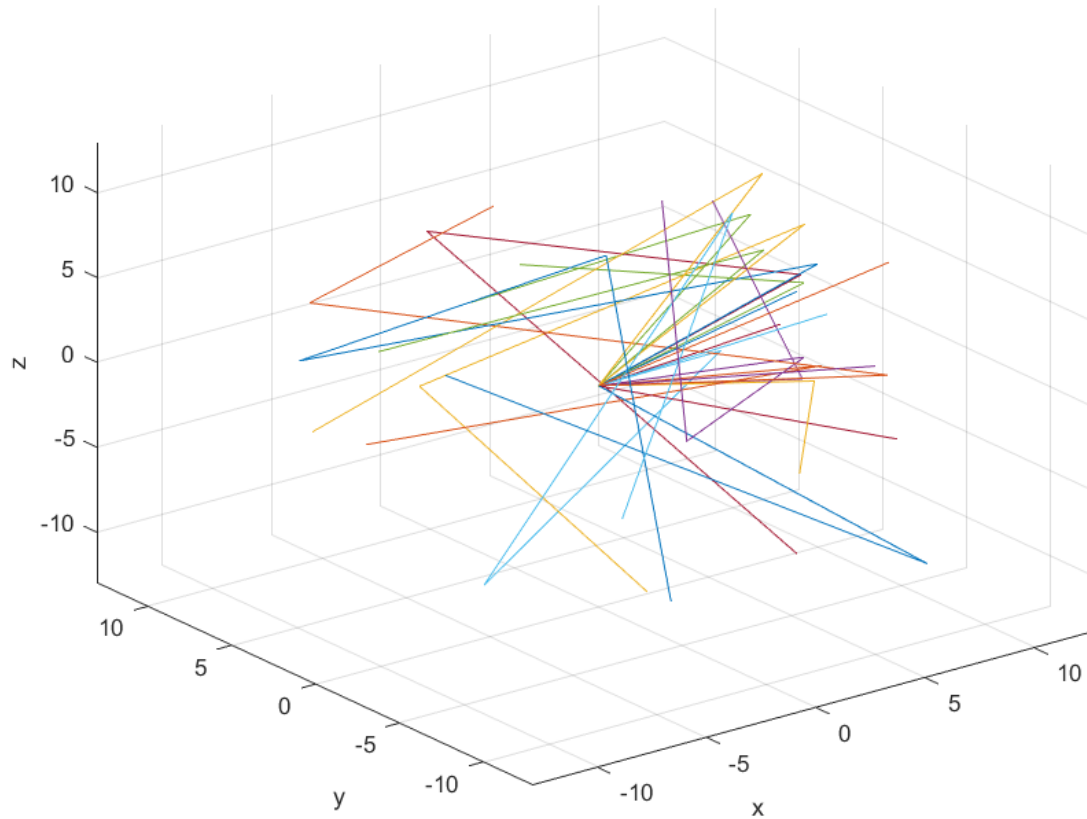
Path Tracing von 20 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 5; Raumgröße: 5



Variation der Raumgröße

Path Tracing III

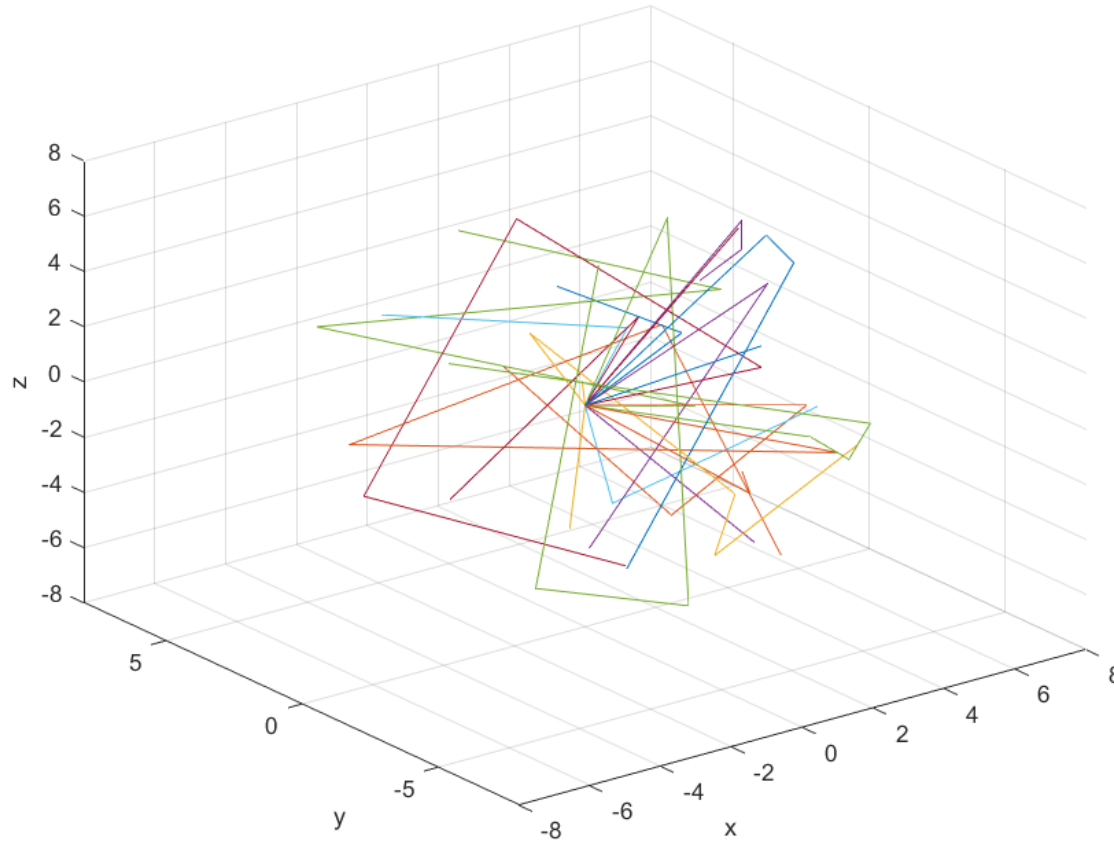
Path Tracing von 20 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 5; Raumgröße: 10



Variation des Lambert-Parameters

Path Tracing I

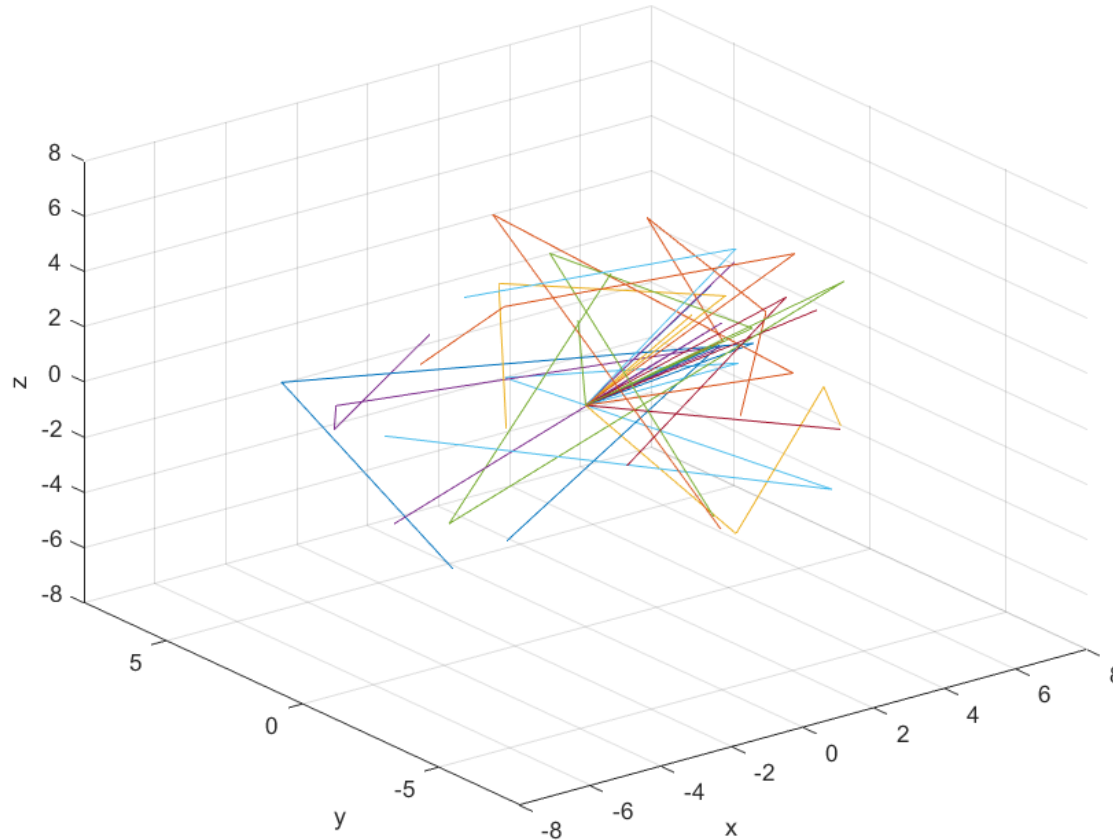
Path Tracing von 20 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 1; Raumgröße: 5



Variation des Lambert-Parameters

Path Tracing II

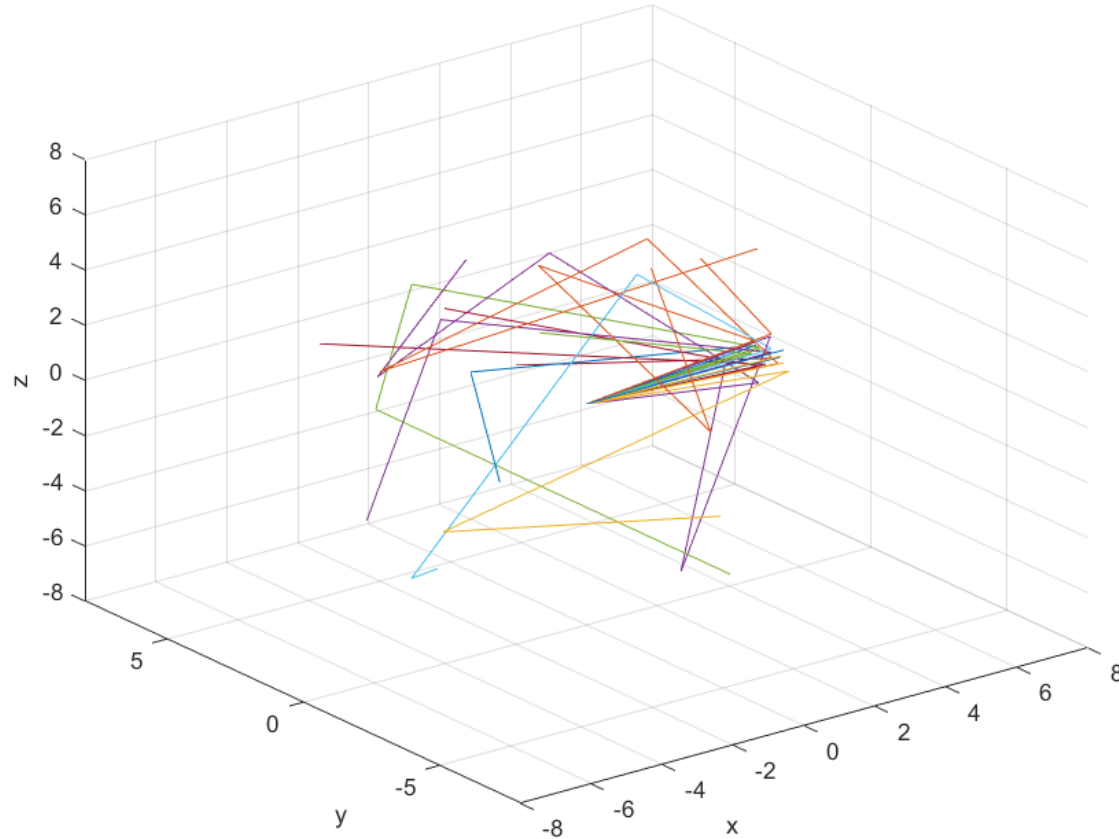
Path Tracing von 20 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 5; Raumgröße: 5



Variation des Lambert-Parameters

Path Tracing III

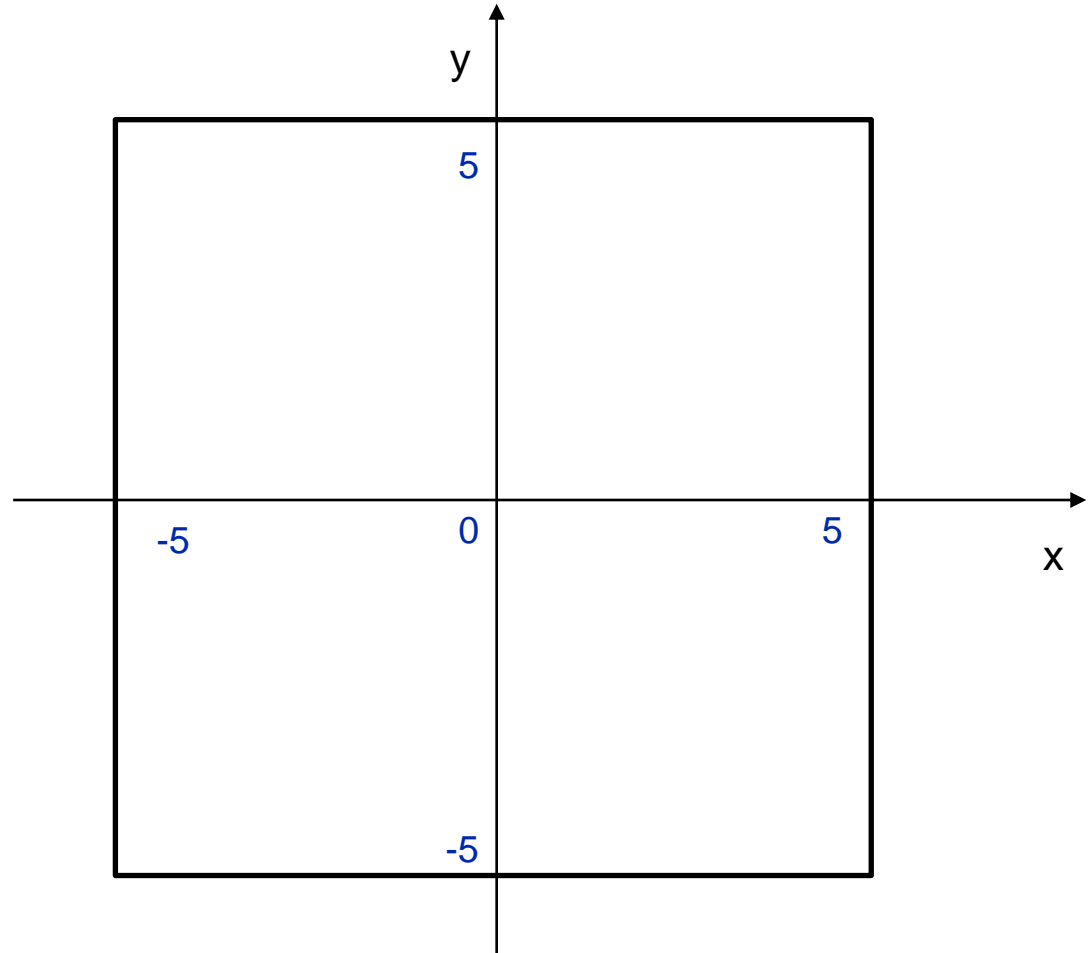
Path Tracing von 20 Photonen, Reflektionsfaktor: 0.8; Lambert-Parameter: 100; Raumgröße: 5



Definition Raumgröße

- Raumgröße: 5
- Wandlänge: 10
- Höhe: 10

→ Lichtquelle im
Raummittelpunkt (Ursprung
des Koordinatensystems)



Schnittpunkte zwischen Gerade und Ebene

Vektorrechnung

- Bewegung des Photons als Gradengleichung:

$$p = \text{stuetz_v} + t \cdot \text{richtungs_v}$$

- Stütz- und Richtungsvektor haben jeweils x, y und z Koordinaten
- Wände als Ebenengleichung in Koordinatenform:

$$Wand = Ax + By + Cz + D$$

- Schnittpunkt ist dort, wo Gerade und Ebene gleich sind

