R. Brigola, TH Nürnberg Georg Simon Ohm, 2019

Mathematica - Notebooks als Bonusmaterial zum Lehrbuch

[1] Rolf Brigola Fourier-Analysis und Distributionen,
Eine Einführung mit Anwendungen,
edition swk, Hamburg 2019

Einige grafische Darstellungen von Äquipotentialflächen von Lösungen der Potentialgleichung $\Delta u = -\rho/\epsilon_0$, ρ zu 2 Punktladungen gehörig

1. Nachfolgend sieben Äquipotentialflächen, erzeugt von
   2 Ladungen an den Stellen x1, x2.
   Die im Unendlichen verschwindende Lösung der Potentialgleichung erhält man durch die Faltung der regulären Distribution $1/(4\pi r)$, r der Abstand zum Nullpunkt, mit der verallgemeinerten Ladungsdichte $\rho/\epsilon_0 = q1\ \delta(x-x1) + q2\ \delta(x-x2)$,
   (x1,x2 die 2 Raumpunkte, in denen die Ladungen sitzen).
   Im Ergebnis erhält man die bekannte Coulomb-Formel für Punktladungen (vgl. [1], Kapitel 8). Das Potential hat gerade an den Stellen, an denen die Ladungen sitzen, Singularitäten. Es ist mathematisch als reguläre Distribution aufzufassen.
   Wir setzen nachfolgend , abgesehen von konstanten Faktoren und physikalischen Einheiten: q1= -2 bei (-1,0,0), q2=1 bei (1,0,0)
   und plotten die Äquipotentialflächen zu den Niveaus
   u = {-0.75,-0.25,-0.1,0,0.1,0.25,0.75}.
   Im zweiten Bild nur die Äquipotentialfläche zum Niveau u = -0.1
   Wir betrachten zunächst den Nahbereich im Halbraum y > 0 und verwenden den Mathematica-Befehl ContourPlot3D

In[1]:=
```
electroStaticPotential [q_, p_, r_] := Sum[q[[i]] / Norm[r - p[[i]]], {i, Length[q]}]
```

Bei 3 Ladungen q1, q2, q3 würde das Potential lauten:

In[2]:=
```
electroStaticPotential [{q₁, q₂, q₃}, {{x₁, y₁, z₁}, {x₂, y₂, z₂}, {x₃, y₃, z₃}}, {x, y, z}] //
 TraditionalForm
```

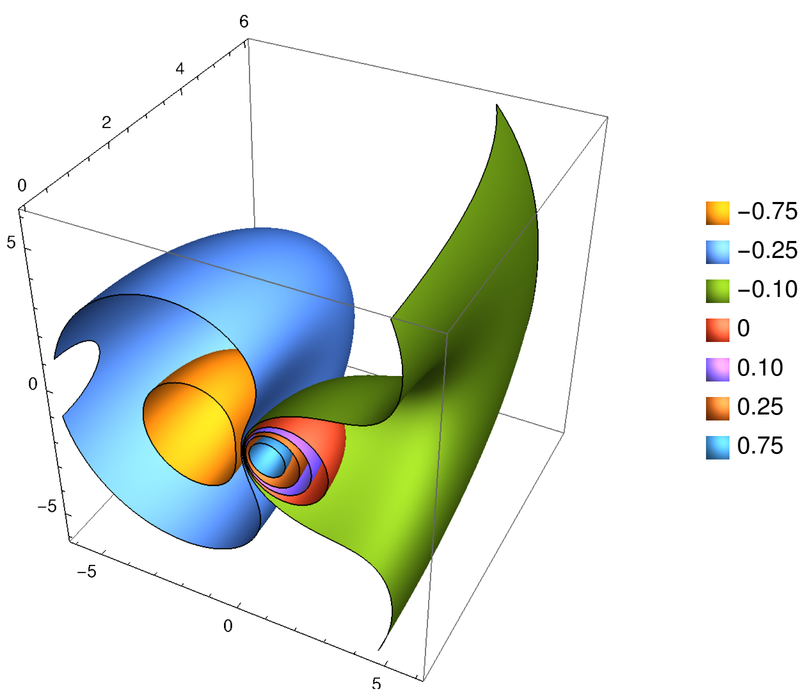
Out[2]//TraditionalForm=

$$\frac{q_1}{\sqrt{|x-x_1|^2+|y-y_1|^2+|z-z_1|^2}}+\frac{q_2}{\sqrt{|x-x_2|^2+|y-y_2|^2+|z-z_2|^2}}+\frac{q_3}{\sqrt{|x-x_3|^2+|y-y_3|^2+|z-z_3|^2}}$$

Wir plotten Äquipotentialflächen zum oben genannten Beispiel
mit 2 Ladungen q1 und q2

In[4]:=
```
p1 = ContourPlot3D [
  Evaluate[electroStaticPotential [{-2, +1}, {{-1, 0, 0}, {1, 0, 0}}, {x, y, z}]],
  {x, -6, 6}, {y, 0, 6}, {z, -6, 6}, Contours → {-0.75, -0.25, -0.1, 0, 0.1, 0.25, 0.75},
  Mesh → None, PlotLegends → Automatic]
```

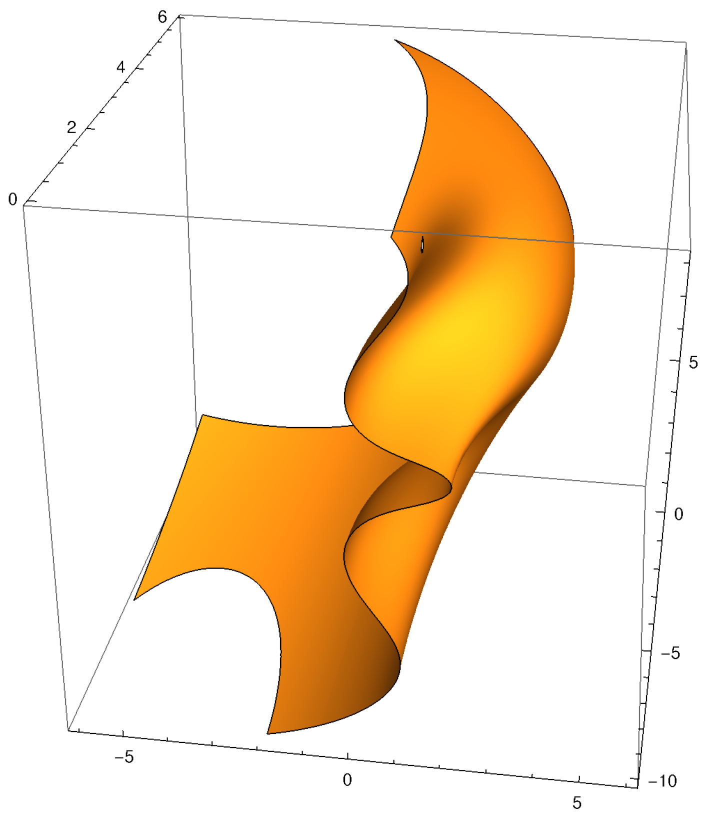
Out[4]=



Nur die Fläche zum Niveau u=-0.1
mit 3 Ladungen {-3,1,1} an den Stellen
{{-1,0,0},{1,0,-2},{1,0,2}}

In[11]:=
```
p2 = ContourPlot3D[Evaluate[
    electroStaticPotential [{-3, +1, +1}, {{-1, 0, 0}, {1, 0, -2}, {1, 0, 2}}, {x, y, z}]],
  {x, -6, 6}, {y, 0, 6}, {z, -10, 8}, Contours → {-0.1},  Mesh → None]
```

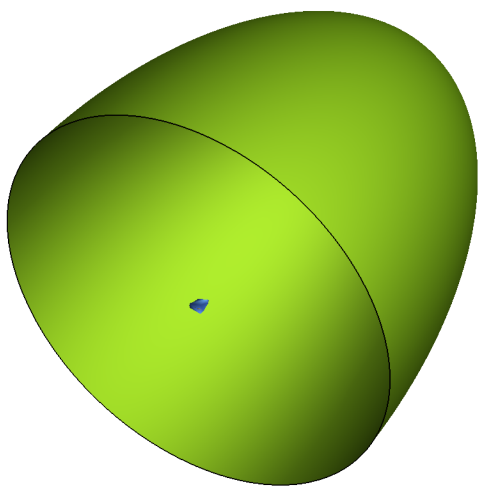
Out[11]=



Nachfolgend eine Ansicht im Fernbereich. Für wachsende r>>1 sieht es immer mehr so aus, als ob das Potential zu einer einzigen Ladung mit der Summe aller Ladungsmengen im Nullpunkt gehört, was man der Coulomb-Formel auch ansehen kann (vgl. auch Multipol-Entwicklung in der theoretischen Physik). Bedenkt man, dass in uns und um uns im Außenbereich viele Ionen fließen und elektrische Felder herrschen, ist das kein unerheblicher physikalischer Sachverhalt.

Wir betrachten Niveauflächen im Halbraum y > 0, abfallend bis zum Niveau u = - 0.008 (blau)

In[15]:=
```
p3 = ContourPlot3D[Evaluate[electroStaticPotential[{-3, +1, +1},
    {{-1, 0, 0}, {1, 0, -2}, {1, 0, 2}}, {x, y, z}]], {x, -220, 220}, {y, 0, 220},
  {z, -220, 220}, Contours → {-0.75, -0.25, -0.008, 0, 0.25, 0.75},
  Mesh → None, Axes → None, Boxed → False, ImageSize → Large]
```

Out[15]=


Für das folgende Bild wird das Vektorfeld
zum Potential von Bild 1 berechnet und es werden
einige Feld-Vektoren zusammen mit den
Äquipotentialflächen geplottet.
Man kann die Grafik mit der Maus drehen und sie
damit aus verschiedenen Richtungen betrachten.
Ggf. können Sie den "ViewPoint" dann passend

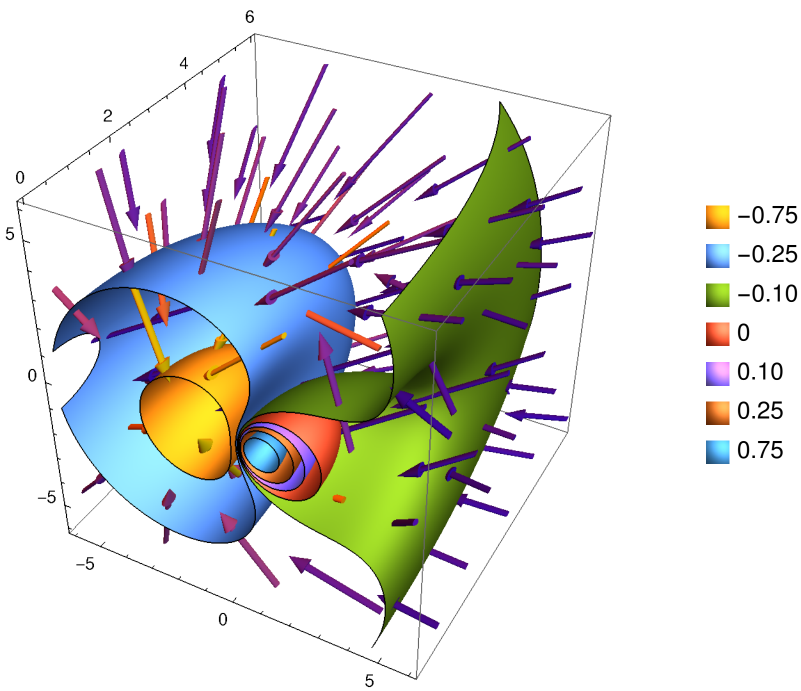
in das Mathematica-Kommando übernehmen.

In[12]:=
```
electricField[{q1_, q2_}, {{x1_, y1_, z1_}, {x2_, y2_, z2_}}] =
  -D[electroStaticPotential[{q1, q2}, {{x1, y1, z1}, {x2, y2, z2}}, {x, y, z}], {{x, y, z}}] /.
    Abs'[x_] :> x/Sqrt[x^2];
```

In[13]:=
```
v = VectorPlot3D[electricField[{-2, 1}, {{-1, 0, 0}, {1, 0, 0}}],
    {x, -6, 6}, {y, 0, 6}, {z, -6, 6}, VectorStyle → "Arrow3D", VectorPoints → 5,
    VectorScale → {.2, Scaled[0.25]}, Axes → True, Boxed → True, AxesStyle → Black];
```

In[16]:=
```
p4 = Show[p1, v]
```

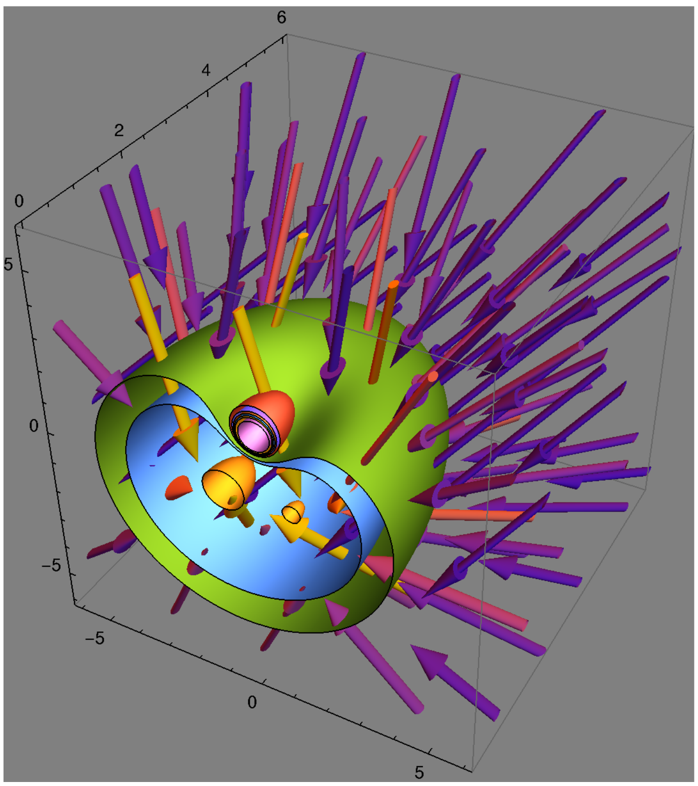
Out[16]=



Nachfolgend ein analoges Beispiel mit 3 Ladungen

In[17]:=
```
p5 = ContourPlot3D[Evaluate[electroStaticPotential[{-3, -1, 1},
      {{-1, 0, 0}, {1, 0, 0}, {0, 0, 2}}, {x, y, z}]], {x, -6, 6}, {y, 0, 6},
   {z, -6, 6}, Contours → {-3. - 2, -1, -0.75, -0.25, -0.1, 0, 0.1, 0.2, 0.5},
   Mesh → None, Axes → True, Boxed → True, ImageSize → Medium,
   Background → Gray, AxesStyle → Black];
electricField[{q1_, q2_, q3_}, {{x1_, y1_, z1_}, {x2_, y2_, z2_}, {x3_, y3_, z3_}}] =
  -D[electroStaticPotential[{q1, q2, q3},
      {{x1, y1, z1}, {x2, y2, z2}, {x3, y3, z3}}, {x, y, z}], {{x, y, z}}] /. Abs'[x_] :> x/Sqrt[x^2];
v2 = VectorPlot3D[electricField[{-3, -1, 1}, {{-1, 0, 0}, {1, 0, 0}, {0, 0, 2}}],
   {x, -6, 6}, {y, 0, 6}, {z, -6, 6}, VectorStyle → "Arrow3D", VectorPoints → 5,
   VectorScale → {.3, Scaled[0.25]}, Axes → True, Boxed → True, AxesStyle → Black];
Show[p5, v2, ViewPoint → {1.3`, -2.4`, 2.`}]
```

Out[20]=



Sie können selbst analoge Veranschaulichungen mit einigen Ladungen mehr erzeugen oder auch wie in [1] einmal ein Potential zu einer kontinuierlichen Ladungsdichte berechnen und analog veranschaulichen, um damit ein Gefühl für die zugehörige Physik an ganz konkreten Beispielen entwickeln.