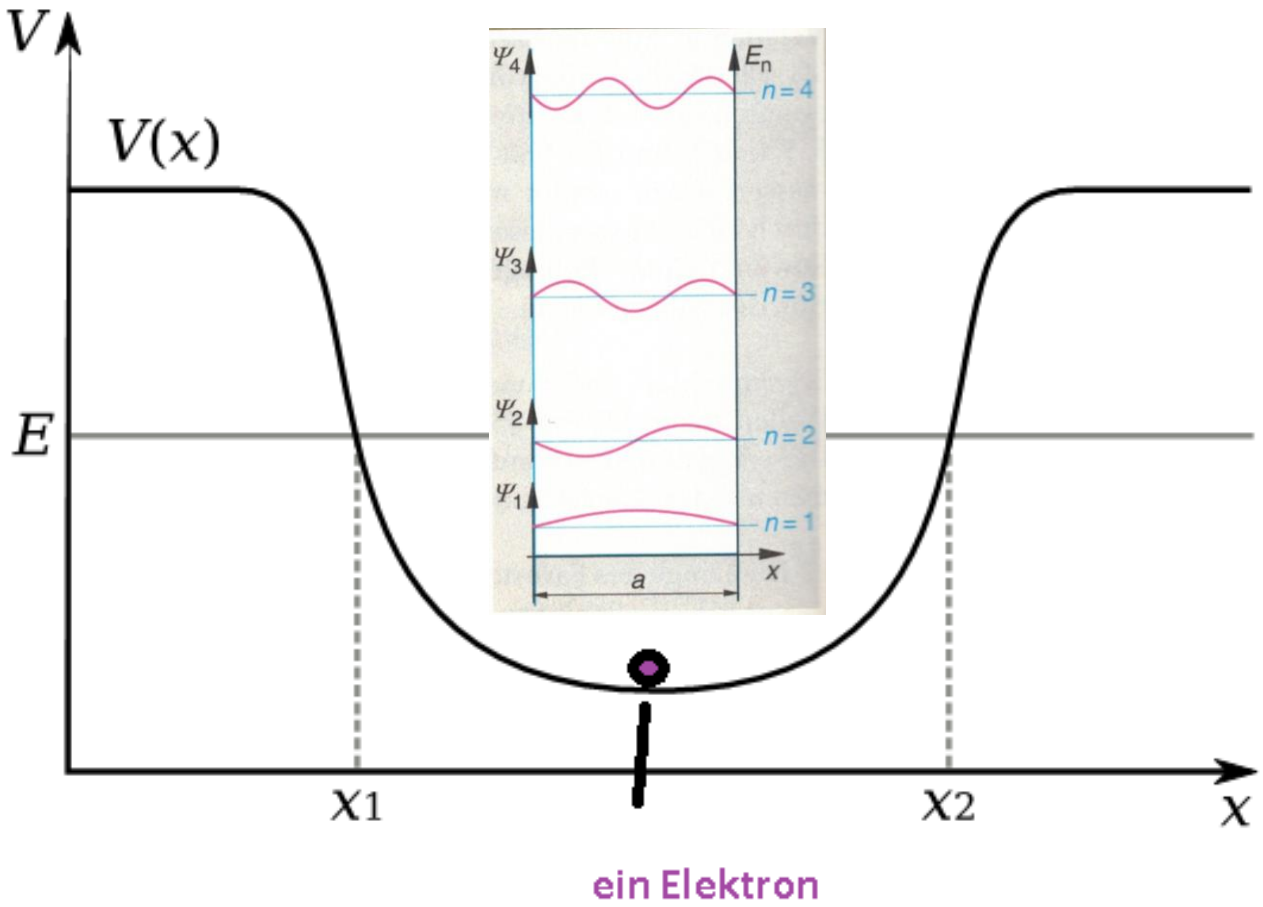
Abschlussprojekt:
Die numerischen Lösungen der Schrödingergleichung für ein Elektron im Potentialtopf am Beispiel von Python

1.12.2023

Das Modell:

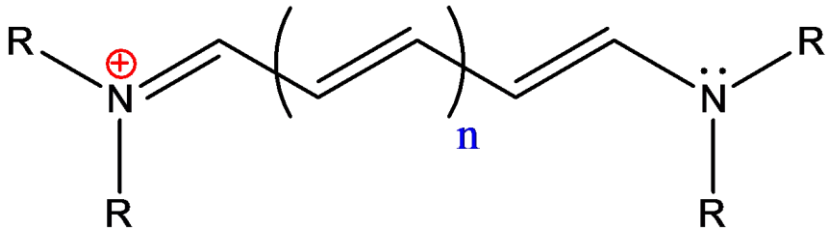


Bildquelle: wikipedia.org: Eintrag: Potentialtopf (1.12.2023)

Die zeitunabhängige Schrödinger-Gleichung und die Randbedingungen:

$$-\frac{\hbar^2}{2m}\frac{d^2\psi}{dx^2} + V(x)\psi = E\psi$$

$$\psi(0) = \psi(L) = 0$$

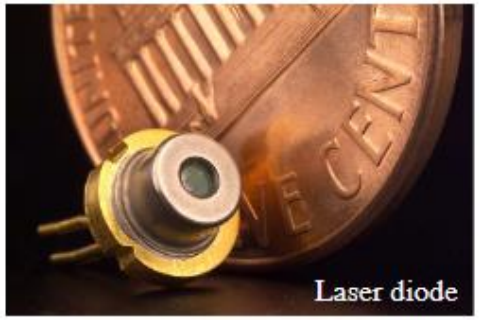


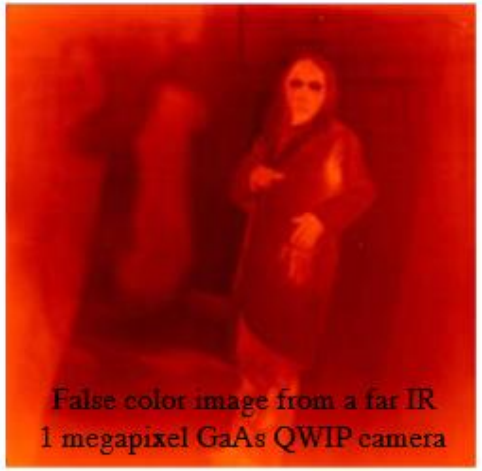
Bildquelle: abiweb.de; Eintrag: Cyanin-Farbstoffe (1.12.2023)

Praktische Anwendungen:

Quantum wells:

- Semiconductor material with small energy gap (e.g. GaAs) is sandwiched between energy barriers from material with a larger energy gap (e.g. AlGaAs). A quantum well is formed between the barriers.
- Typical layer thicknesses ~ *1-10 nm*.
- Quantization effects result in allowed energy bands, whose energy positions are dependent on the height and width of the barrier.
- This is used in the fabrication of specialised semiconductor devices such as: laser diodes, high electron mobility transistors (HFET or MODFET), quantum well infrared photodetectors (QWIP arrays).





Quelle: Quantum Mechanics(P304H) Lectures- University of Glasgow

Die Funktionsweise des Algorithmus:

Die zeitunabhängige Schrödinger-Gleichung und die Randbedingungen:

$$-\frac{\hbar^2}{2m}\frac{d^2\psi}{dx^2} + V(x)\psi = E\psi$$

$$\psi(0) = \psi(L) = 0$$

Euler-Cromer (Runge-Kutta 2nd)

$$\ddot{f} = \frac{d^2 f}{dt^2}$$

$$\boxed{\ddot{f} = \ddot{f}(t)} = \frac{\Delta \dot{f}}{\Delta t} = \frac{\dot{f}_2 - \dot{f}_1}{\Delta t}$$

$$\boxed{\dot{f}_2 = \dot{f}_1 + \ddot{f}\Delta t}$$

$$\dot{f} = \frac{\Delta f}{\Delta t} \qquad \boxed{f_2 = f_1 + \dot{f}_2 \Delta t}$$

Quelle: Allain Rhett: Solving the schroedinger equation with finite difference methods, https://trinket.io/glowscript/a068e60f42

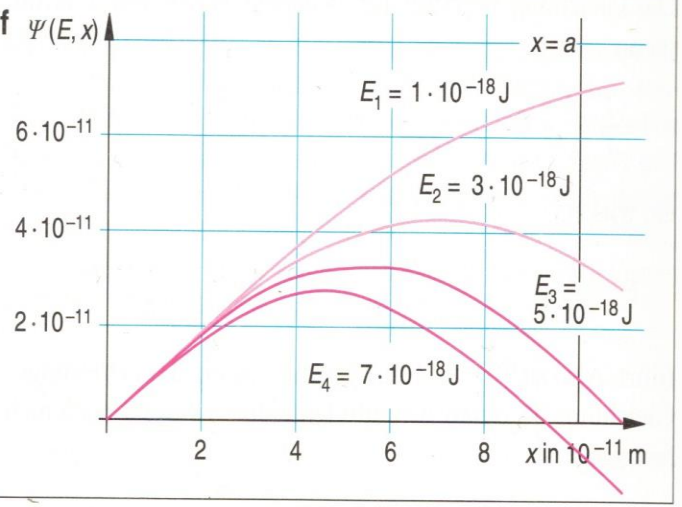
Iterative Berechnung der Wellenfunktion im Potentialtopf

(1) $\Psi(x) = \Psi(x-\Delta x) + \Psi'(x-\Delta x)\Delta x$
(2) $\Psi'(x) = \Psi'(x-\Delta x) + \Psi''(x-\Delta x)\Delta x$
(3) $\Psi''(x) = -1{,}6382 \cdot 10^{38}\, E\, \Psi(x)$

Startwerte mit $E = 1 \cdot 10^{-18}$ *und* $\Delta x = 10^{-11}$:
$x = 0; \Psi(0) = 0; \Psi'(0) = 1;\ \Psi''(0) = 0$

Ergebnisse des 1. Iterationsschrittes:
$x = \Delta x;\ \Psi(\Delta x) = 1 \cdot 10^{-11};\ \Psi'(\Delta x) = 1;\ \Psi''(\Delta x) = -1{,}6382 \cdot 10^{9}$

Ergebnisse des 2. Iterationsschrittes:
$x = 2\Delta x;\ \Psi(2\Delta x) = 2 \cdot 10^{-11};\ \Psi'(2\Delta x) = 0{,}9836;$
$\Psi''(2\Delta x) = -3{,}2764 \cdot 10^{9}$



419.1 Mit den Startwerten für die Stelle $x = 0$ gibt die Iterationsvorschrift die Berechnung von $\Psi(x)$ an der Stelle Δx vor. Das Ergebnis ist $\Psi(10^{-11}) = 10^{-11}$. Die Iteration liefert weitere Werte von $\Psi(x)$, mit denen der Graph der Funktion $\Psi(x)$ für die Energie $E = 1 \cdot 10^{-18}$ J gezeichnet wird. Andere Werte von E führen zu weiteren Graphen. Ein Wert von E wie z. B. $E_1 = 6 \cdot 10^{-18}$ J, der zu einem Graphen durch den Punkt $(a|0)$ führt, beschreibt einen Quantenzustand. (Die Iteration wird natürlich nur mit den Zahlenwerten und ohne Einheiten durchgeführt.)

Quelle: J. Grehn, J. Krause: Metzler Physik, Schroedel-Verlag (2006): Seite: 419, 436

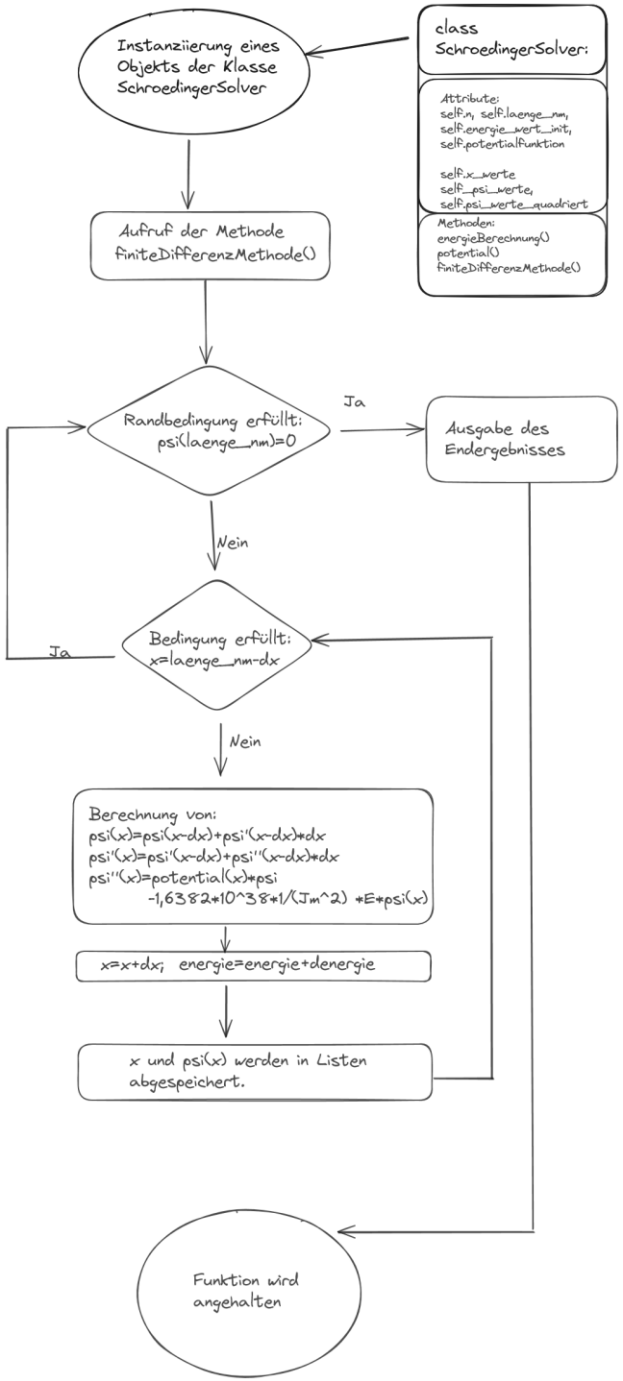
Der Algorithmus:

```
while psi_final >0:
  x = 0
  psi = 0
  dpsi = 1
  while x < self.laenge_nm-dx:
    psi = psi + dpsi * dx
    dpsi = dpsi + ddpsi * dx
    ddpsi = self.potential(x)*psi- self.__konstanten_dict["8mepi2/h2"] * energie_wert * psi
    x = x + dx
    psi_final = psi
    energie_wert = energie_wert + denergie
    x_wert.append(x)
    psi_wert.append(psi)
    psi_wert_quadriert.append(psi**2)
    self.x_werte=x_wert
    self.psi_werte=psi_wert
    self.psi_werte_quadriert=psi_wert_quadriert
    print(" Der numerisch berechnete Energiewert ausgehend von einem Anfangsenergiewert E=",se
  return self.x_werte, self.psi_werte, self.psi_werte_quadriert
```

(1) $\Psi(x) = \Psi(x-\Delta x) + \Psi'(x-\Delta x)\,\Delta x$

(2) $\Psi'(x) = \Psi'(x-\Delta x) + \Psi''(x-\Delta x)\,\Delta x$

(3) $\Psi''(x) = -1{,}6382 \cdot 10^{38}\, E\, \Psi(x)$



Die nächsten Schritte:

- Eigenvektoren und Eigenwerte mit der Funkton **eigh_tridiagonal** berechnen:

$$\Rightarrow \psi''(x_i) = \frac{\psi(x_{i+1}) - 2\psi(x_i) + \psi(x_{i-1})}{d^2}$$

$$\begin{bmatrix} \frac{1}{\Delta y^2} + mL^2V_1 & -\frac{1}{2\Delta y^2} & 0 & 0\ldots \\ -\frac{1}{2\Delta y^2} & \frac{1}{\Delta y^2} + mL^2V_2 & -\frac{1}{2\Delta y^2} & 0\ldots \\ \ldots & \ldots & \ldots & -\frac{1}{2\Delta y^2} \\ \ldots 0 & 0 & -\frac{1}{2\Delta y^2} & \frac{1}{\Delta y^2} + mL^2V_{N-1} \end{bmatrix} \begin{bmatrix} \psi_1 \\ \psi_2 \\ \ldots \\ \psi_{N-1} \end{bmatrix} = mL^2E \begin{bmatrix} \psi_1 \\ \psi_2 \\ \ldots \\ \psi_{N-1} \end{bmatrix}$$

$$\psi_0 = \psi_N = 0$$

```
from scipy.linalg import eigh_tridiagonal
d = 1/dy**2 + mL2V(y)[1:-1]
e = -1/(2*dy**2) * np.ones(len(d)-1)
```

```
w, v = eigh_tridiagonal(d, e)
```

- Zeitabhängige Schrödingergleichung berechnen in 2D, 3D

$$i\hbar\frac{\partial}{\partial t}\psi(x,t) = -\frac{\hbar^2}{2m}\frac{\partial^2}{\partial x^2}\psi(x,t) + V(x)\psi(x,t) \qquad \psi(0,t) = \psi(L,t) = 0$$

- Benutzerfreundliche GUI

Quelle: Luke Polson: Eigenvalue_eigenfunction_solution,
https://github.com/lukepolson/youtube_channel/blob/main/Python%20Metaphysics%20Series/vid3.ipynb

Die nächsten Projekte:

Iterative Berechnung der Wellenfunktion im H-Atom

(1) $\Psi(r) = \Psi(r-\Delta r) + \Psi'(r-\Delta r)\,\Delta r$

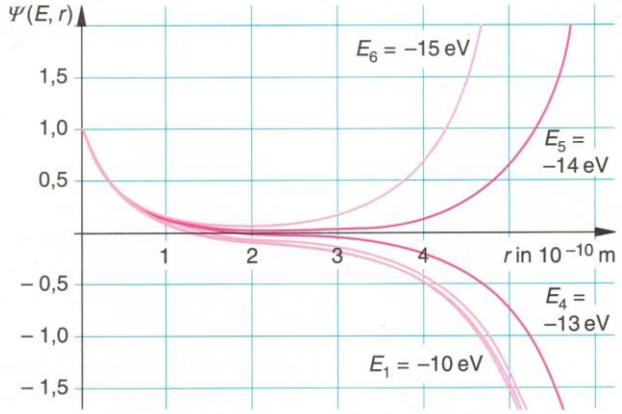
(2) $\Psi'(r) = \Psi'(r-\Delta r) + \Psi''(r-\Delta r)\,\Delta r$

(3) $\Psi''(r) = -2\Psi'(r)/r - 1{,}6382 \cdot 10^{38}\,(E + 2{,}30708 \cdot 10^{-28}/r)\,\Psi(r)$

Startwerte mit $E = -15 \cdot 1{,}602 \cdot 10^{-19}$ *und* $\Delta r = 10^{-12}$:
$r_0 = 10^{-12}$; $\Psi(r_0) = 1$; $\Psi'(r_0) = -10^{10}$; $\Psi''(r_0) = -1{,}7401 \cdot 10^{22}$

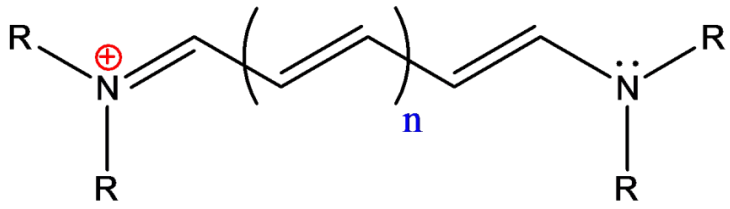
Ergebnisse des 1. Iterationsschrittes:
$r = r_0 + \Delta r$; $\Psi(r_0 + \Delta r) = 0{,}99$; $\Psi'(r_0 + \Delta r) = -2{,}7401 \cdot 10^{10}$;
$\Psi''(r_0 + \Delta r) = 9{,}0823 \cdot 10^{21}$

Ergebnisse des 2. Iterationsschrittes:
$r = r_0 + 2\Delta r$; $\Psi(r_0 + 2\Delta r) = 0{,}9626$; $\Psi'(r_0 + 2\Delta r) = -1{,}8319 \cdot 10^{10}$;
$\Psi''(r_0 + 2\Delta r) = 4{,}6432 \cdot 10^{20}$



419.2 Mit den Startwerten für die Stelle $r_0 = 10^{-12}$ m gibt die Iterationsvorschrift die Berechnung von $\Psi(r)$ an der Stelle $r_0 + \Delta r$ vor. Die Iteration liefert die Werte von $\Psi(r)$, mit denen der Graph für die Energie $E = -15 \cdot 1{,}6 \cdot 10^{-19}$ J gezeichnet wird. Die Iteration wird mit unterschiedlichen Werten von E wiederholt. Zwischen $E = -14 \cdot 1{,}6 \cdot 10^{-19}$ J und $E = -13 \cdot 1{,}6 \cdot 10^{-19}$ J liegt bei etwa $-13{,}6 \cdot 1{,}6 \cdot 10^{-19}$ J der Wert von E_1, der zur ersten Lösungsfunktion mit $\Psi(\infty) = 0$ führt.

Absorptionsmaxima von Farbstoff-molekülen (Parameter: C-Atome n in der konjugierten Ketten- Länge)



Quelle: J. Grehn, J. Krause: Metzler Physik, Schroedel-Verlag (2006): Seite: 419

Bildquelle: abiweb.de;
Eintrag: Cyanin-Farbstoffe (1.12.2023)