

TU WIEN
Einführung in die Methode der finiten Elemente
Mathematik Fachvertiefung

von

Benjamin Kulnik
1425900
benjamin.kulnik@student.tuwien.ac.at

20. Januar 2018

Inhaltsverzeichnis

1	Finite Differenzen Methode	2
1.1	Analytische Lösung	2
1.2	Numerische Lösung	3
1.3	Implementierung in Matlab	4
1.4	Ergebnisse	5
2	Finite Elemente Methode	7
2.1	Angabe	7
2.2	Herleiten der schwachen Formulierung	7
2.2.1	Allgemeine Idee	7
2.3	Festlegen des Intervalls	9
2.4	Ansatzfunktion	9
2.5	Newmark Zeitschrittverfahren	11
2.6	Implementierung in Matlab	12
2.7	Ergebnis	14
3	Simulation mit Netgen und Matlab	16
3.1	Plattenkondensator Simulation	16
3.2	Einfluss des fernen Randes	16

1 Finite Differenzen Methode

Als Beispiel sei das elektrische Skalarpotential V eines Zylinderkondensators zu bestimmen, also das Lösen der Differentialgleichung:

$$-\frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial}{\partial r} u(r) \right) = 0 \quad (1)$$

was dem elektrischen Skalarpotential entspricht.

$$\vec{\nabla} \cdot \epsilon_0 \vec{\nabla} V = 0 \quad (2)$$

Aufgrund der Symmetrie - es wird passenderweise ein Zylinderkoordinatensystem gewählt - kann das Problem eindimensional betrachtet werden, also:

$$V(r, \phi, z) = V(r) \quad (3)$$

Als Randbedingung sind $V(r = 1) = 0$ und $V(r = 5) = 100$ vorgegeben.

Aufgaben

1. Bestimmen Sie die analytische Lösung $u(r)$.
2. Bestimmen Sie eine numerische Lösung mittels Methode der finiten Differenzen indem Sie das Intervall $(1,5)$ gleichmäßig in 4 Teilintervalle teilen und den Differentialoperator diskretisieren.
3. Vergleichen Sie die analytische Lösung mit der numerischen Lösung. Wie groß ist der relative Fehler in % in den Knotenpunkten?

1.1 Analytische Lösung

Die Differentialgleichung kann analytisch gelöst werden. Hierzu wird zuerst das erste Differenzial aufgelöst und die Gleichung auf folgende Form gebracht:

$$\frac{1}{r} \frac{\partial}{\partial r} u(r) + \frac{\partial^2}{\partial r^2} u(r) = 0 \quad (4)$$

umformen und integrieren:

$$-\frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial}{\partial r} u(r) \right) = 0 \quad (5)$$

$$\frac{\partial}{\partial r} \left(r \frac{\partial}{\partial r} u(r) \right) = 0 \quad (6)$$

$$r \frac{\partial}{\partial r} u(r) = C_1 \quad (7)$$

$$\frac{\partial}{\partial r} u(r) = \frac{C_1}{r} \quad (8)$$

$$(9)$$

Nochmaliges integrieren nach r bringt uns schließlich auf

$$u(r) = \int \frac{\partial u}{\partial r} dr = C_1 \ln r + C_2 \quad (10)$$

mit noch zu bestimmenden Konstanten C_1 und C_2 . Diese werden so gewählt, dass die Lösung unsere Randbedingungen erfüllt. Für $u(1) = 0$ ergibt sich:

$$u(1) = C_1 \ln(1) + C_2 = 0 \quad \Rightarrow \quad C_2 = 0 \quad (11)$$

Und für die zweite Randbedingung ergibt sich:

$$u(5) = C_1 \ln(5) = 100 \quad \Rightarrow \quad C_1 = \frac{100}{\ln 5} \approx 62.13349... \quad (12)$$

1.2 Numerische Lösung

Für die numerische Berechnung wurde das Intervall in 4 Teilintervalle aufgeteilt und jeweils das Potential an den Stützstellen $r = \{1, 2, 3, 4, 5\}$ berechnet. Die Gleichung 2 ist ausgeschrieben und in Zylinderkoordinaten:

$$-\frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial}{\partial r} u(r) \right) = 0 \quad (13)$$

Durch Anwenden der Produktregel lässt sich das Gleichungssystem auf

$$\frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial}{\partial r} u(r) \right) = \frac{1}{r} \frac{\partial}{\partial r} u(r) + \frac{\partial^2}{\partial r^2} u(r) = 0 \quad (14)$$

Da wir $u(r=1) = 0$ und $u(r=5) = 100$ bereits kennen, konstruieren wir einen Vektor mit den gesuchten Werten der Knotenpunkte 2, 3, 4:

$$\mathbf{u} = \begin{bmatrix} u_2 \\ u_3 \\ u_4 \end{bmatrix} \quad (15)$$

Das Differential wird numerisch mit dem zentralen Differenzen Quotient angenähert:

$$\frac{\partial u(r)}{\partial r} \approx \frac{u(r+h) - u(r-h)}{2h} \quad (16)$$

und entsprechend für die zweite Ableitung wurde

$$\frac{\partial^2 u(r)}{\partial r^2} \approx \frac{u(r+h) - 2u(r) + u(r-h)}{h^2} \quad (17)$$

gewählt. Die Herleitung erfolgt über die Taylorreihen-Entwicklung.

Wir konstruieren unser lineares Gleichungssystem folgendermaßen:

$$\frac{1}{2rh} \mathbf{A} \mathbf{u} + \frac{1}{h^2} \mathbf{B} \mathbf{u} = \mathbf{f} \quad (18)$$

wobei die Matrizen \mathbf{A} und \mathbf{B} die Koeffizienten für die Differenzenquotienten enthalten. Der Vektor \mathbf{f} enthält die rechte Seite der Differenzialgleichung (die hier allerdings immer 0 ist) und außerdem die entsprechenden Randbedingungen die zum Lösen erforderlich sind.

$$\mathbf{f} = \begin{bmatrix} \frac{1}{rh} u_1 - \frac{1}{h^2} u_1 \\ 0 \\ -\frac{1}{2rh} u_5 - \frac{1}{h^2} u_5 \end{bmatrix} \quad (19)$$

Ausgeschrieben erhält man mit $h = 1$ und $r = \{2, 3, 4\}$:

$$\frac{1}{2} \begin{bmatrix} 0 & \frac{1}{2} & 0 \\ -\frac{1}{3} & 0 & \frac{1}{3} \\ 0 & -\frac{1}{4} & 0 \end{bmatrix} \begin{bmatrix} u_2 \\ u_3 \\ u_4 \end{bmatrix} + \begin{bmatrix} -2 & 1 & 0 \\ 1 & -2 & 1 \\ 0 & 1 & -2 \end{bmatrix} \begin{bmatrix} u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} \frac{1}{2}u_1 - u_1 \\ 0 \\ -\frac{1}{8}u_5 - u_5 \end{bmatrix} \quad (20)$$

Die Matrizen können leicht zusammengefasst werden und anschließend gelöst werden.

1.3 Implementierung in Matlab

```

1 %% Analytisches L?sen
2 N = 3; % Stützstellen
3 r_start = 1;
4 r_end = 5;
5
6 h = (r_end - r_start) / (N+1); % Schrittweite
7 % r: Stützstellen an denen u(r) gesucht wird
8 r = (r_start+h:h:r_end-h)';
9
10 % U = C_1 log(r) + C_2
11 V = 62.13349 * log(r) ;
12
13 %% Numerische Berechnung
14 % (sp?rliche) Matrizen anlegen (bessere performance, da
    Matrizen
15 % sowieso aus fast nur 0en bestehen
16
17 A = sparse(2:N,1:N-1,-1,N,N)+ ...
    sparse(1:N-1,2:N,1,N,N);
18 B = sparse(1:N,1:N,-2) + ...
    sparse(2:N,1:N-1,1,N,N)+ ...
    sparse(1:N-1,2:N,1,N,N);
19 b = sparse(1:N,1,0);
20
21 A = 1/(2*h) * (1./r) .* A;
22 B = (1/h)^2 * B;
23
24 % Randbedingungen
25 b(1) = 0;
26 b(N) = -100 * 1/(2*h) * 1/r(N) - 100 * (1/h)^2;
27
28 % Au + Bu = b
29 % -> (A+B)u = Cu = b
30 % -> b =inv(C)*b
31
32 % Matlab nutzt automatisch ein geeignetes L?
    sungsverfahren
33 C = A+B;
34 Vnum = C \ b ;

```

```

38
39 %% Hinzufügen der Randwerte zum Lösungsvektor
40 V = [0; V(:); 100]';
41 Vnum = [0; Vnum(:); 100]';
42 r = [1;r;5]';
43 %% Vergleichen
44 figure()
45 plot(r,V,r,Vnum);
46 legend('Analytische Lösung', 'Numerische Lösung', ...
47       'Location','southeast');
48 xlabel('Radius r'); ylabel('Potential V'); grid on;
49 set(gca, 'FontSize', 18);
50 print('cylinder.png','-dpng','-r300');
51
52 figure()
53 plot(r,abs((Vnum-V)./V*100), r, abs(Vnum-V));
54 xlabel('Radius r'); ylabel('Fehler'); grid on;
55 legend('Relativer Fehler [%]', 'Absoluter Fehler',...
56       'Location','northeast');
57 set(gca, 'FontSize', 18);
58 print('error.png','-dpng','-r300');

```

Listing 1: Implementierung der FDM für die Berechnung des Potentials in einem Zylinderkondensators. Eine Anpassung der Anzahl der Stützstellen ist möglich über die Variable N

1.4 Ergebnisse

Die Methode der finiten Differenzen erlaubt eine zuverlässige Berechnung einer einfachen Differentialgleichung. Der relative Fehler ist selbst bei wenigen Stützstellen sehr gering.

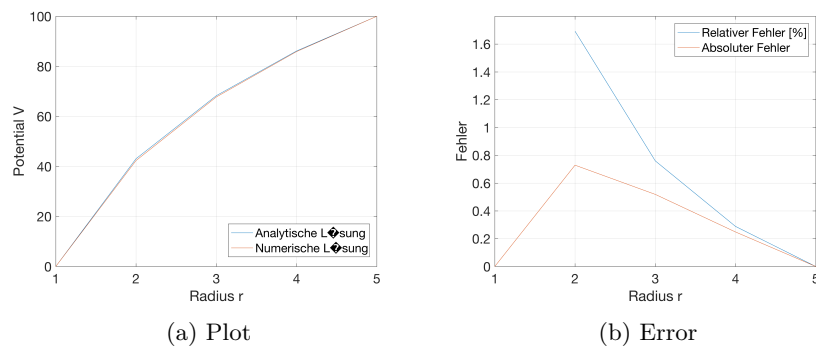


Abbildung 1: Ergebnisse für 3 Stützstellen im Intervall $r = [1, 5]$

Tabelle 1: Ergebnisse der FDM für $N = 3$

	Analytisch	Numerisch	Relativer Fehler [%]
u_2	43.0677	42.3387	1.6926
u_3	68.2606	67.7419	0.7599
u_4	86.1353	85.8871	0.2882

2 Finite Elemente Methode

2.1 Angabe

Es soll die Methode der finiten Elemente (FEM) auf folgende Gleichung angewendet werden.

$$\frac{\partial^2 u(x, t)}{\partial^2 x} - \mu_0 \epsilon_0 \frac{\partial^2 u(x, t)}{\partial^2 t} = 0 \quad (21)$$

Mit den Randbedingungen

$$u(a, t) = 0 \quad (22)$$

$$-\frac{\partial}{\partial x} u(b, t) + \sqrt{\mu_0 \epsilon_0} \frac{\partial}{\partial t} u(b, t) = 0 \quad (23)$$

Mit den Anfangsbedingungen

$$u(x, 0) = w(x) = e^{-k(2x-a-b)^2} \quad (24)$$

$$\frac{\partial}{\partial t} u(x, 0) = 0 \quad (25)$$

soll das Newmarksche Zeitschrittverfahren durchgeführt werden.

Aufgaben

1. Leiten Sie dazu die schwache Formulierung für die Abhängigkeit vom Ort her.
2. Nehmen Sie eine geeignete Länge des Intervalls $[a, b]$ an und unterteilen Sie es gleichmäßig in n finite Elemente der Länge h .
3. Verwenden Sie Hutfunktionen als Basisfunktionen. Berücksichtigen Sie die Randbedingungen und erstellen Sie die Steifigkeitsmatrix \mathbf{S}_h und die Massenmatrix \mathbf{M}_h .
4. Lösen Sie das gewöhnliche Differentialgleichungssystem

$$\mathbf{S}_h \mathbf{u}_h + \mathbf{M}_h \frac{\partial^2}{\partial t^2} \mathbf{u}_h = \mathbf{0}$$

mit dem Newmark-Zeitschritt-Verfahren

2.2 Herleiten der schwachen Formulierung

2.2.1 Allgemeine Idee

Wir betrachten Differentialgleichungssysteme auf einem Gebiet Ω (etwa ein Teil der reellen Zahlen \mathbb{R}) mit vorgegebenen Werten (Randbedingungen) an $\partial\Omega$ die sich der Art

$$\mathbf{A} \mathbf{u} = \mathbf{f} \quad (26)$$

anschreiben lassen. Dabei ist \mathbf{u} die gesuchte Funktion. Alle Faktoren und Differentialoperatoren werden durch \mathbf{A} ausgedrückt. Die Idee ist nun \mathbf{u} durch eine Ansatzfunktion u_h zu approximieren. Diese Ansatzfunktion soll in etwa die Gestalt einer Summe von (Laplace'schen) Ansatzfunktion multipliziert mit einem

Trägerfunktion $p_i(x)$ sein. Diese Trägerfunktion soll im gesamten Gebiet mit Ausnahme eines kleinen Bereichs (eines *Trägers*) verschwinden, also konstant 0 sein. Dadurch teilt man

$$u_h = \sum_{i=0}^N u_i p_i(x) \quad (27)$$

Diese Ansatzfunktion verändert unsere Funktion u , deshalb wird sie im Allgemeinen die Gleichung **nicht** erfüllen. Der linke Teil der Gleichung minus dem rechten Teil wird nicht exakt 0 sondern ein gewissen Wert r ergeben. Dieser Fehler $r(x)$ wird *Residuum* genannt.

$$\mathbf{r} = \mathbf{f} - \mathbf{A}\mathbf{u}_h \neq 0 \quad (\text{allgemein})$$

und in unserem Beispiel

$$r(x) = \frac{\partial^2 u_h(x, t)}{\partial^2 x} - \mu_0 \epsilon_0 \frac{\partial^2 u_h(x, t)}{\partial^2 t} \neq 0 \quad (28)$$

Der tatsächliche Fehler (also der Differenz $e = u - u_h$) kann, im Gegensatz zum Residuum, nicht bestimmt werden, da die Funktion u nicht bekannt ist. Das Residuum kann jedoch bestimmt werden und dieses gilt es zu minimieren. Gesucht ist nun also:

$$\min \mathbf{r} = \min \mathbf{f} - \mathbf{A}\mathbf{u}_h = 0 \quad (\text{hoffentlich})$$

Bisher war es die Aufgabe die Funktion u_h zu finden (A und f sind ja bekannt), welche die Residuuen verschwinden lässt. Wenn u_h Dazu wählen wir für die schwache Formulierung eine bekannte Gewichtsfunktion $w(x)$ und multiplizieren damit unsere Gleichung (*Methode der gewichteten Residuuen*). Anschließend integrieren wir über das gesamte Gebiet Ω (was in unserem eindimensionalen Fall dem Intervall $[a, b]$ entspricht). Damit ergibt sich:

$$\int_{\Omega} r(x) w(x) dx = 0 \quad (29)$$

$$\int_a^b u_h''(x, t) w(x) dx - \mu_0 \epsilon_0 \int_a^b \frac{\partial^2}{\partial t^2} u_h(x, t) w(x) dx = 0 \quad (30)$$

Mit der verallgemeinerten Ableitung der Distributionen in H (dem Sobolev-Raum, welcher Funktionen, für die eine verallgemeinerte Ableitung existiert umfasst):

$$\int_a^b u_h''(x) w(x) dx = - \int_a^b u_h'(x) w'(x) dx + u_h'(x) w(x) \Big|_a^b \quad (31)$$

An $x = a$ (allgemeiner $\Gamma_D = \{a\}$) liegt eine Dirichlet'sche Randbedingung vor, was so viel bedeutet wie das $u(a, t) = \text{const}$ ist, damit muss die Weg-Ableitung $u'(a, t)$ verschwinden. Generell gilt hier, dass durch das Einsetzen der Ansatzfunktionen N Gleichungen erzeugt werden. Da der wert von u an der Stelle $\Gamma_D = \{a\}$ allerdings bereits bekannt ist, wird diese Gleichung nicht benötigt damit kann $w(a) = 0$ gewählt werden. Außerdem ist in unserem konkreten Fall noch eine Cauchy'sche Randbedingung an $\Gamma_C = \{b\}$ vorgegeben:

$$u'(b, t) = u_x(b, t) = \sqrt{\mu_0 \epsilon_0} u_t(b, t) \quad \forall t \in [0, t_0]$$

daraus folgt also

$$u'(x, t) w(x) \Big|_a^b = \sqrt{\mu_0 \epsilon_0} u_t(b, t) w(b) - \underbrace{u'(a, t) w(a)}_{=0} \quad (32)$$

2.3 Festlegen des Intervalls

Bei der Methode der finiten Elemente wird eine stetige Funktion auf N Teilintervalle (Elemente) zerlegt. Dazu werden Funktionswerte an Knotenpunkten berechnet und diese mit auf kleine Intervalle beschränkten Funktionen multipliziert. Für das Intervall $x = \{a, \dots, b\}$ kann die Zerlegung folgendermaßen aussehen:

$$x_i = a + ih \quad (33)$$

wobei die Schrittweite $h = \frac{b-a}{N}$ ist. Im Prinzip kann diese Zerlegung aber beliebig gewählt werden.

2.4 Ansatzfunktion

Für die Funktion u wird eine Interpolation u_h folgender Gestalt gewählt:

$$u \approx u_h = \sum_{i=0}^N u_i p_i(x) \quad (34)$$

Dies entspricht einer Summe von Ansatzfunktion gewichtet mit noch zu bestimmenden konstanten Faktoren u_i . Diese Ansatzfunktion können verschiedene Gestalten haben, sollten jedoch die Eigenschaft aufweisen einen kompakten Träger zu besitzen, d.h. $p_i(x) = 0$ für alle $x > a + ih$ und $x < a - ih$. Zwischen zwei Knotenpunkten ist der Wert der Funktion nicht bekannt und muss entsprechend interpoliert werden und da sich zwischen den Knotenpunkten mehrere Ansatzfunktionen i.A. überlagern, sollte darauf geachtet werden, dass die Summe immer gleich 1 ist, also

$$p(x) = \sum_{i=0}^N p_i(x) = 1 \quad \forall x \in [a, b] \quad (35)$$

Da die Koeffizienten u_i nun konstante Faktoren sind, wird die (Weg-)Ableitung auf $p(x)$ übertragen. Das bedeutet:

$$u'_h(x, t) = \sum_{i=0}^N u_i p'_i(x) \quad (36)$$

In diesem Beispiel wird als Ansatzfunktionen verschobene Dreiecksfunktionen ('Hutfunktionen') verwendet. Selbstverständlich muss bei der Auswahl darauf geachtet werden, dass die gewählte Ansatzfunktion hinreichend oft differenzierbar ist.

Setzen wir dies nun in die Herleitung der schwachen Formulierung 30 ein, erhalten wir mit etwas umformen und unter Verwendung der verallgemeinerten Ableitung aus Formel 31. Da sich unser Intervall zeitlich nicht ändert, darf die Zeit-Ableitung vor das Integral gezogen werden.

$$\sum_{i=0}^N u_i \int_a^b p'_i(x) w'(x) dx - \mu_0 \epsilon_0 \frac{\partial^2}{\partial t^2} \sum_{i=0}^N u_i \int_a^b p_i(x) w(x) dx = 0 \quad (37)$$

Hier gelten noch u_1, u_2, \dots, u_N , also N Unbekannte Koeffizienten, zu finden. Für N unbekannte Koeffizienten werden bekanntlich N unabhängige Gleichungen

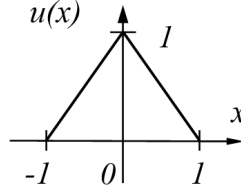


Abbildung 2: Hutfunktion

benötigt. Ein Weg diese zu erhalten ist über die Ansatzfunktion $v(x)$. Für diese Ansatzfunktion wird ebenfalls eine ähnliche Zerlegung wie für $p(x)$ festgelegt, sollte es sogar die selbe sein, handelt es sich um die *Galerkin Methode*. Man setzt also

$$w(x) = p(x) = \sum_{j=0}^N p_j(x) \quad (38)$$

und setzt dies dann in die Gleichung (37) ein.

$$\sum_{i=0}^N u_i \sum_{j=0}^N \int_a^b p'_i p'_j dx + \sqrt{\mu\epsilon} \frac{\partial}{\partial t} u_i \sum_{j=0}^N p_i(B) p_j = \mu\epsilon \sum_{i=0}^N \frac{\partial^2}{\partial t^2} u_i \sum_{j=0}^N \int_a^b p_i p_j dx \quad (39)$$

Da wir Hutfunktionen verwenden können, welche durch die Dreiecksfunktion $p_i = \text{tri}(\frac{x+ih}{h})$ beschrieben werden, können die Integrale gelöst werden:

$$\int_a^b p_i p_j dx = \begin{cases} \frac{4}{6}h, & j = i \\ \frac{1}{6}h, & j = \{i-1, i+1\} \\ 0 & \text{sonst} \end{cases} \quad (40)$$

und für $p'_i p'_j$ gilt

$$\int_a^b p'_i p'_j dx = \begin{cases} \frac{2}{h}, & j = i \\ -\frac{1}{h}, & j = \{i-1, i+1\} \\ 0 & \text{sonst} \end{cases} \quad (41)$$

Ausgenommen sind hier die Ergebnisse die direkt am Rand liegen. Bei $\Gamma_D = \{a\}$ ist der Wert von u bereits bekannt, die Ansatzfunktion kann also ohne weiteres Null gesetzt werden. Bei $\Gamma_C = \{b\}$ ist nur der Wert der Ableitung von u bekannt. Hier muss der Wert erst bestimmt werden. Bei Betrachtung der Situation am Rand sieht man, dass hier nur über die *halbe* Ansatzfunktion integriert wird, also das auch genau halb so groß ist.

Damit kann das Gleichungssystem in der Form

$$\mathbf{S}_h \mathbf{u}_h + \mathbf{C}_h \frac{\partial}{\partial t} \mathbf{u}_h + \mathbf{M}_h \frac{\partial^2}{\partial t^2} \mathbf{u}_h = \mathbf{0} \quad (42)$$

folgendermaßen angeschrieben werden:

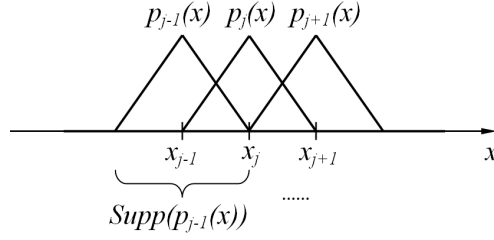


Abbildung 3: Überlagerte Hutfunktionen

$$\begin{aligned}
& \frac{1}{h} \begin{bmatrix} 2 & -1 & 0 & \cdots & 0 \\ -1 & 2 & -1 & \cdots & 0 \\ 0 & -1 & 2 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & -1 \\ 0 & 0 & \cdots & -1 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_N \end{bmatrix} + \\
& + \sqrt{\mu_0 \epsilon_0} \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \frac{\partial}{\partial t} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_N \end{bmatrix} + \\
& + \mu_0 \epsilon_0 \frac{h}{6} \begin{bmatrix} 4 & 1 & 0 & \cdots & 0 \\ 1 & 4 & 1 & \cdots & 0 \\ 0 & 1 & 4 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 1 \\ 0 & 0 & \cdots & 1 & 2 \end{bmatrix} \frac{\partial^2}{\partial t^2} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_N \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}
\end{aligned} \tag{43}$$

2.5 Newmark Zeitschrittverfahren

Hier werden die partiellen Zeitableitung durch die Variablen u, v, a dargestellt.

$$u = u(t_j), \quad \frac{\partial u}{\partial t} \approx v(t_j), \quad \frac{\partial^2 u}{\partial t^2} \approx a(t_j)$$

Diese sind eine Anlehnung an die Mechanik, wo die erste Ableitung des Weges nach der Zeit die Geschwindigkeit v , und die zweite Ableitung nach der Zeit die Beschleunigung a bezeichnet. Das Newmark-Verfahren beruht auf einer Taylor-Entwicklung für u und v , wobei zweite Ableitungen durch Beschleunigungen zum Zeitpunkt t_j und t_{j+1} approximiert werden. Für den Zeitpunkt $t_{j+1} = t_j + \Delta t$ ergibt sich das Gleichungssystem

$$\mathbf{S}u_{j+1} + \mathbf{C}v_{j+1} + \mathbf{M}a_{j+1} = \mathbf{f}_{j+1} \tag{44}$$

$$v_{j+1} = v_j + \Delta t \left[\left(\frac{1}{2} - \gamma \right) a_j + \gamma a_{j+1} \right] \quad (45)$$

$$u_{j+1} = u_j + \Delta t v_j + \frac{(\Delta t)^2}{2} \left[\left(\frac{1}{2} - \beta \right) a_j + \beta a_{j+1} \right] \quad (46)$$

mit $\beta = \frac{1}{4}$ und $\gamma = \frac{1}{2}$.

Die Gleichung 44 kann auf einen der drei Parameter umgeformt werden und es ergibt sich

$$(\mathbf{S} + \gamma \Delta t \mathbf{C} + \mathbf{M}) a_{j+1} = \mathbf{f}_{j+1} \quad (47)$$

wobei die Größen u_j , v_j und a_j in den Vektor \mathbf{f}_{j+1} sind. Herleitung siehe Skriptum.

2.6 Implementierung in Matlab

```

1 N = 99; % Stützstellen - 1
2 x0 = 0; % Intervall a-b
3 xend = 10;
4 h = (xend-x0) / N;
5 x = (x0:h:xend)';
6 N = N+1;
7 f = zeros(N,1);
8
9
10
11 mu0 = 4*pi*10^(-7);
12 eps0 = 8.85*10^(-12);
13
14 %mu0 = 1;
15 %eps0 = 1;
16
17 f(1) = 0;
18 f(N) = 0;
19
20 M = sparse(1:N,1:N,4) + sparse(1:N-1,2:N,1,N,N) + sparse
    (2:N,1:N-1,1,N,N);
21 S = sparse(1:N,1:N,2) + sparse(1:N-1,2:N,-1,N,N) + sparse
    (2:N,1:N-1,-1,N,N);
22 C = sparse(1:N,1:N,0);
23 %%
24 S(N,N) = 1;
25 M(N,N) = 2;
26 S = 1/h * S;
27 M = mu0*eps0*h/6*M;
28 C(N,N) = sqrt(mu0*eps0); % du/dt = 0
29
30 %C(1,1) = sqrt(mu0*eps0);
31
32

```

```

33
34 %% Newmark Zeitschrittverfahren
35
36 % Anfangsbedingungen
37 k = 1/2;
38 u = exp(-k*(2*x-x0-xend).^2);
39 v = zeros(N,1);
40 a = M\(f-S*u-C*v); % calculate initial a
41
42 % Zeitschritte
43 dt = 1/2 * h * sqrt(mu0*eps0);
44
45 gamma = 1/2;
46 beta = 1/4;
47
48 % Time Loop
49 max_time_steps = 340;
50 U = zeros(length(u),max_time_steps);
51
52 for i = 1:max_time_steps
53
54     U(:,i) = u;
55
56     A = (M + gamma*dt* C + beta* dt^2 * S);
57     f_ = f - C * (v+(1-gamma)*dt*a) - S * (u + dt*v+ (dt
        ^2)/2 * (1-2*beta)*a);
58
59     a_ = A \ f_;
60
61     v_ = v + dt*( (1-gamma)* a + gamma*a_ );
62     u_ = u + dt*v + dt^2/2 * ((1/2-beta)*a + beta*a_);
63
64     u = u_;
65     v = v_;
66     a = a_;
67
68 end
69
70
71 %% Plot Data
72 t = 0:dt:dt*(size(U,2)-1);
73 surf(t,x,U,'EdgeAlpha',0.3)
74 set(gca,'FontSize',14);
75 xlabel('Zeit t');
76 ylabel('Ort x');
77 print('wave.png','-dpng','-r300')
78
79 %%

```

```
80 makeMovie(U,x,x0,xend);
```

Listing 2: Implementierung der FEM für ein eindimensionales Lösen der Wellengleichung

2.7 Ergebnis

Die Methode der finiten Elemente in Kombination mit dem Newmark'schen Zeitschrittverfahren bringt sehr brauchbare Ergebnisse hervor. Jedoch neigt das Verfahren, trotz einhalten der Stabilitätsbedingung nach ausreichend vielen Zeitschritten

$$\Delta t < \frac{\Delta x^2}{v} \quad (48)$$

zur Instabilität.

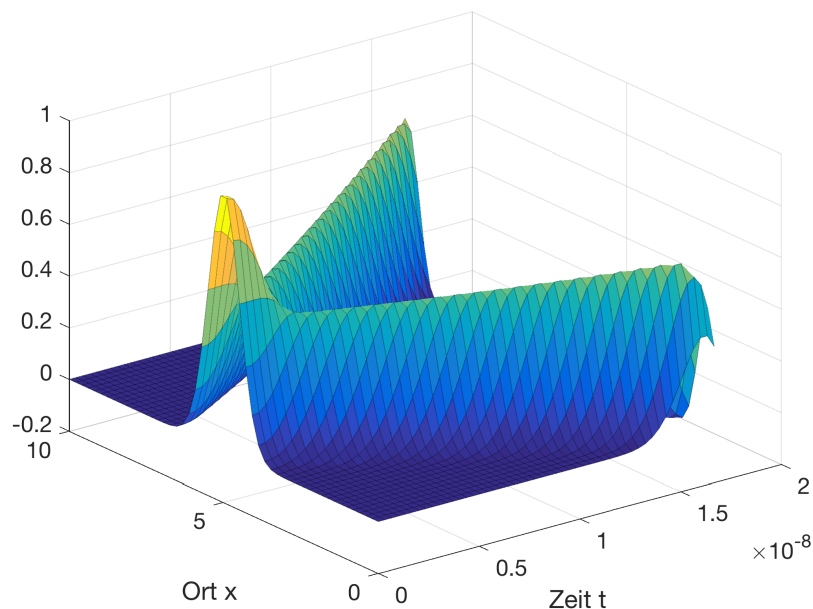


Abbildung 4: Ergebnisse der FEM mit dem Newmark-Zeitschrittverfahren

FEM Ablauf (Zusammenfassung)

1. Stelle das Gleichungssystem in der Form $Lu = f$ dar, wobei L ein linearer Differentialoperator beliebiger Gestalt ist.
2. Ersetze deine Funktion u durch u_h , eine gewichtete Summe von Ansatzfunktionen, mit noch zu bestimmenden Gewichten.
3. Multipliziere die Gleichung mit einer Gewichtsfunktion w und integriere über das gesamte Gebiet.
4. Für das Galerkin-Verfahren wähle die selbe Ansatzfunktionen für u_h und w .

5. Löse die Integrale und stelle das Gleichungssystem in einer Matrix Schreibweise dar. Anschließend muss nur noch das lineare Gleichungssystem gelöst werden.

3 Simulation mit Netgen und Matlab

Alle Unterlagen für den 3ten Übungsteil finden Sie im TUWEL-Kurs. Installieren Sie Netgen 6.1. Das Simulationsbeispiel finden Sie in Simulations.zip im Verzeichnis Electrostatics. Falls Sie nach ernsthaften Anstrengungen immer noch ein Problem mit der Installation oder den Simulationen haben, dann schreiben Sie einfach ein Email an karl.hollaus@tuwien.ac.at und schildern kurz das Problem.

Angabe

1. Plattenkondensator(Verwenden Sie das Problem Electrostatics__2Domains.pde)
 - (a) Bestimmen Sie die Kapazität des Plattenkondensators mit Hilfe einer FE-Simulation und vergleichen Sie diese mit der bekannten Näherungsformel. Die Plattenbreite ist $b = 6m$, der Plattenabstand ist $d = 2m$ und die Spannung an den beiden Platten beträgt $U = 2V$.
 - (b) Untersuchen Sie den relativen Fehler in Abhängigkeit vom Dielektrikum zwischen den Kondensatorplatten. Stellen Sie den relativen Fehler in Abhängigkeit von ϵ_r dar.
 - (c) Untersuchen Sie den Einfluss des fernen Randes auf die Lösung indem Sie die Geometrie des Problemgebietes vergrößern bzw. verkleinern. Dokumentieren Sie Ihre Erkenntnisse.

3.1 Plattenkondensator Simulation

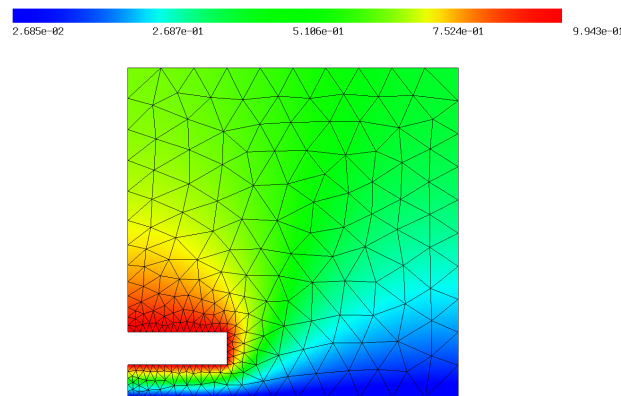


Abbildung 5: Simulation eines Plattenkondensators mit Netgen

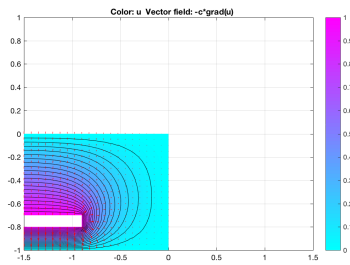
3.2 Einfluss des fernen Randes

Die Größe des berechneten Gebiets, und damit die "Nähe" der Randbedingungen zueinander haben einen großen Einfluss auf das sich um den Kondensator ergebende Streufeld. Es wurde aus Symmetrie-Gründen nur der linke, obere

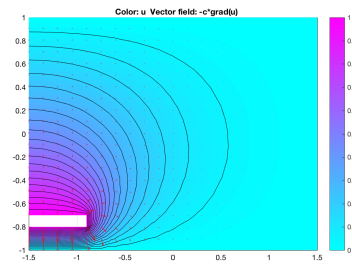
Teil eines Kondensators in 2D berechnet. Für die Berechnung in Matlab wurden folgende Werte verwendet:

- $1m = 0.1$ Skaleneinheiten
- PDGL: $-\nabla \cdot c \nabla u = 0$ wobei $c = 1$ ist.
- Dirichletsche Randbedingungen an den Kondenstorflächen, an der horizontalen Symmetrie-Achse, sowie am fernen Rand (rechts und oben).
- Neumansche Randbedingung an der vertikalen Symmetrie-Achse

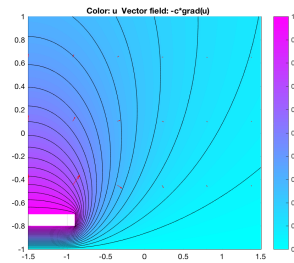
Info: Aufgrund anhaltender Installationsprobleme mit Python unter OS X wurden alle Simulationen mit dem *pdetool* von Matlab gemacht.



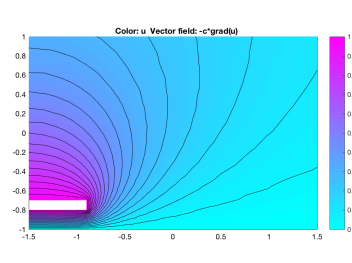
(a) Rand sehr nah, 1.5 x 1



(b) Rand nah, 3 x 2



(c) Großes Gebiet 10 x 10



(d) Riesen Gebiet 100 x 100

Abbildung 6: Untersuchen des Einflusses des fernen Randes

Abbildungsverzeichnis

1	Ergebnisse für 3 Stützstellen im Intervall $r = [1, 5]$	5
2	Hutfunktion	10
3	Überlagerte Hutfunktionen	11
4	Ergebnisse der FEM mit dem Newmark-Zeitschrittverfahren . . .	14
5	Simulation eines Plattenkondensators mit Netgen	16
6	Untersuchen des Einflusses des fernen Randes	17

Tabellenverzeichnis

1	Ergebnisse der FDM für $N = 3$	6
---	--	---