

# Hardwarebeschleunigte Simulation von Drahtantennen

Lukas Brumm (S553370)

March 8, 2019

# 1 Einleitung

Die vorliegende Arbeit "Hardwarebeschleunigte Simulation von Drahtantennen" wurde im Rahmen des Studienganges 'Informations- und Kommunikationstechnik' an der Hochschule fuer Technik und Wirtschaft Berlin (HTW) verfasst. Ziel der Arbeit ist es die Methode der finiten Differenzen im Zeitbereich (FDTD-Methode) nachzuvollziehen und fuer eine hardwarebeschleunigte Simulation von Drahtantennen zu verwenden. Da numerische Loesungsverfahren fuer Differentialgleichungen nicht Teil meines Studiums waren hiehl ich mich bei der Herleitung der FDTD-Methode an die Vorlesung 'Electromagnetic Analysis Using Finite-Difference Time-Domain' samt veroeffentlichtem Skript Raymond Rumpfs der Universitaet Texas in El Paso, wobei ich mich entschloss die Maxwellgleichungen fuer ein normalisiertes H-Feld zu loesen und nur auf dem E- und H-Feld zu arbeiten. Dadurch laesst sich die numerische Loesung der Maxwellgleichungen durch nur zwei Formeln gaenzlich beschreiben.

# 2 Motivation

Jede drahtlose Kommunikation benoetigt eine Antenne auf der Send- und Empfangsseite. Durch die Fortschreitende Miniaturisierung der Elektronik und eine daraus folgende Vernetzung wie man sie zum Beispiel im Internet of Things (IoT) beobachten kann werden immer spezifischere Charakteristiken an die Send- und Empfangsantenne gestellt. In solchen Einsatzgebiete sind konventionelle Antennendesigns haeufig nicht anwendbar, sondern erfordern spezielle Antennen. Der Entwurf solcher Antennen passiert interaktiv mit einem Simulationsprogramm, wobei auch gezeigt wurde, dass evolutionaere Algorithmen (EA) zum automatisierten Antennendesign verwendet werden koennen. [Globus et al.()Globus, Linden, and John] Ein solcher EA muss mitunter hunderttausende Simulationen durchfuehren [Globus et al.()Globus, Linden, and John, p.6], weshalb jede Simulation so performant wie moeglich sein sollte.

# 3 Stand der Technik

Das numerische Loesen von Differentialgleichungen wird seit 1928 untersucht [R. Kourant(1928)] und hat sich seitdem kontinuierlich weiterentwickelt. [Hagness(2005)] Neben der FDTD existieren noch andere Verfahren um Differentialgleichungen zu loesen wie zum Beispiel die Finite-Elemente-

Methode, die Finite-Volumen-Methode, oder die Randelement-Methode, die haeufig fuer Differentialgleichungen besonderer Form entwickelt wurde. [Stenfeld()] Die FDTD-Methode hingegen kann auf beliebige Differentialgleichungen angewandt werden und wird im Verlaufe der Arbeit eingehend behandelt. Auch wenn die FDTD-Methode seit 1928 erforscht wird werden fuer die Einsatzgebiete der FDTD neue Erkenntnisse gewonnen, so ist die Simulation einer Antenneneinspeisung noch nicht vollstaendig geklaert und wird weiterhin untersucht. [Luebbers and H.S()]

## 4 Loesen der Maxwellgleichungen durch die FDTD-Methode

Die FDTD-Methode, auch Yee-Methode genannt, loest Differentialgleichungen indem die als infinitesimal klein angenommenen Differentialquotienten durch finite Differenzenquotienten approximiert werden. [R. Kourant(1928)] Die Maxwellgleichungen werden daher nur in der Differentialform betrachtet. Die FDTD-Methode ist zu komplex und problemabhaengig, als dass eine Einschaetzung der Simulationsergebnisse gelingen koennte ohne die FDTD-Methode mathematisch fuer ein Problem nachvollzogen zu haben. Es werden daher nun kurz die Maxwellgleichungen eingefuehrt, allgemeine Stabilitaetsbedingungen der FDTD-Methode behandelt und die FDTD-Methode genutzt um eine numerische Loesung der Maxwellgleichungen sowie des 'Perfectly Matched Layer' (PML) herzuleiten.

### 4.1 Maxwellgleichungen

Unter den Maxwellgleichungen versteht man die folgenden Gleichungen, die zusammen alle elektromagnetischen Phaenomene beschreiben. Das Gausz-Gesetz fuer Elektrizitaet [Rumpf(a)],

$$\nabla \cdot \vec{D} = \rho_v \quad (1)$$

wobei  $\vec{D}$  die elektrische Flussdichte und  $\rho_v$  die Ladungsdichte ist. Elektrische Felder divergieren bei positiven Ladungen und konvergieren bei negativen Ladungen. Das Gausz-Gesetz fuer Magnetismus [Rumpf(b)],

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

wobei  $\vec{B}$  die magnetische Flussdichte ist. Es gibt keine magnetischen Monopole, Magnetfelder formen immer Schleifen. Das Amperesches Gesetz [Rumpf(c)],

$$\nabla \times \vec{H} = \vec{J} + \frac{\partial \vec{D}}{\partial t} \quad (3)$$

wobei  $\vec{H}$  das magnetische Feld und  $\vec{J}$  die Stromdichte ist. Zirkulierende magnetische Felder induzieren Stroeme sowie zeitabhaengige elektrische Felder. Das Induktionsgesetz [Rumpf(d)],

$$\nabla \times \vec{E} = -\frac{\partial \vec{B}}{\partial t} \quad (4)$$

wobei  $\vec{E}$  das elektrische Feld ist. Zirkulierende elektrische Felder induzieren zeitabhaengige magnetische Felder. Weiterhin ist das D-Feld ueber die Permittivitaet mit dem E-Feld verbunden, [Rumpf(e)]

$$\vec{D}(t) = [\epsilon(t)] * \vec{E}(t) \quad (5)$$

sowie das B-Feld ueber die Permeabilitaet mit dem H-Feld verbunden ist. [Rumpf(e)]

$$\vec{B}(t) = [\mu(t)] * \vec{H}(t) \quad (6)$$

In beiden Faellen handelt es sich um eine Faltung mit einem Tensor. Im Verlauf der Arbeit wird angenommen, dass die Permittivitaet und die Permeabilitaet zeitlich konstant sind. Damit wird die Faltung zu einer Multiplikation. Verwendet man die Gleichungen (5) und (6) um das D- und B-Feld zu eliminieren und gibt man die Permittivitaet und Permeabilitaet relativ zur Vacuumpermittivitaet und -permeabilitaet an so ergeben sich folgende vier Formeln.

$$\nabla * ([\mu] \vec{H}(t)) = 0 \quad (7)$$

$$\nabla * ([\epsilon] \vec{E}(t)) = \rho_v \quad (8)$$

$$\nabla \times \vec{H} = \epsilon_0 [\epsilon_r] \frac{\partial \vec{E}}{\partial t} \quad (9)$$

$$\nabla \times \vec{E} = -\mu_0 [\mu_r] \frac{\partial \vec{H}}{\partial t} \quad (10)$$

Aus den beiden verschraenkten Gleichungen (9) und (10) werden spaeter die sog. Updategleichungen hergeleitet, also die Gleichungen die das E- bzw. H-Feld des naechsten Zeitschrittes liefern. Die Stromdichte  $\vec{J}$  kann genutzt werden um Verlust innerhalb von Materialien darzustellen. Im Verlauf der Arbeit wird angenommen, dass die Antennengeometrie nicht verlustbehaftet ist.

$$\vec{J} = 0 \quad (11)$$

Das E- und H-Feld sind durch die Materialimpedanz  $\eta$  mit einander verknuepft, wobei jede Impedanz relativ zur Freiraumimpedanz  $\eta_0$  angegeben werden kann:

$$\eta = \frac{|\vec{E}|}{|\vec{H}|}$$

$$\eta = \eta_0 * \sqrt{\frac{\mu_r}{\epsilon_r}}, \eta_0 = \pi * 119,9169832 \Omega \quad (12)$$

Im freien Raum ist das E- Feld also um den Faktor  $\eta_0$  groeszer als das H-Feld. Diese Skalierung fuehrt bei numerischen Loesungsverfahren zu Rundungsfehlern, weshalb das H-Feld normalisiert wird und somit die gleiche Groeszenordnung wie das E-Feld besitzt. [Rumpf(f)]

$$\begin{aligned} \tilde{\vec{H}} &= \eta_0 \vec{H} \\ \vec{H} &= \frac{1}{\eta_0} \tilde{\vec{H}} \end{aligned} \quad (13)$$

Somit ergibt sich aus Gleichung (10):

$$\nabla \times \vec{E} = [-\mu] \frac{\partial \tilde{\vec{H}}}{\partial t} * \frac{1}{\eta_0} \quad |\eta_0 = \mu_0 * c \quad (14)$$

$$= -\mu_0 [\mu_r] \frac{1}{\mu_0 c} \frac{\partial \tilde{\vec{H}}}{\partial t} \quad (15)$$

$$= -\frac{[\mu_r]}{c} \frac{\partial \tilde{\vec{H}}}{\partial t} \quad (16)$$

Und aus Gleichung (9) ergibt sich:

$$\nabla \times \vec{H} = \eta_0 \epsilon_0 [\epsilon_r] \frac{\partial \vec{E}}{\partial t} \quad (17)$$

$$= c \mu_0 \epsilon_0 [\epsilon_r] \frac{\partial \vec{E}}{\partial t} \quad |\mu_0 = \frac{1}{c^2 \epsilon_0} \quad (18)$$

$$= \frac{1}{c^2 \epsilon_0} c \epsilon_0 [\epsilon_r] \frac{\partial \vec{E}}{\partial t} \quad (19)$$

$$= \frac{[\epsilon_r]}{c} \frac{\partial \vec{E}}{\partial t} \quad (20)$$

## 4.2 Mathematische Stabilitätsbedingungen fuer finite Differenzen

Immer wenn die FDTD-Methode auf Differentialgleichungen angewendet wird muss jede finite-Differenz am selben Punkt im Raum und in der Zeit existieren um die Differentialgleichung korrekt zu approximieren. [Rumpf(g)]

$$\frac{\partial f(x)}{\partial x} + f(x) = 0 | FDTD - Methode \quad (21)$$

$$\underbrace{\frac{x + \Delta x - f(x)}{\Delta x}}_{\text{Existiert am Punkt } x + \frac{\Delta x}{2}} + \underbrace{f(x)}_{\text{Existiert am Punkt } x} = 0 \quad | x = n\Delta x, x \in \mathbb{N} \quad (22)$$

Diese Gleichung waere instabil, der Term  $f(x)$  muss interpoliert werden um ebenfalls an den Punkten  $x + \frac{\Delta x}{2}$  zu existieren:

$$\frac{f(x + \Delta x) - f(x)}{\Delta x} + \frac{f(x + \Delta x) + f(x)}{2} = 0 \quad (23)$$

## 4.3 Anwenden der FDTD auf die Maxwellgleichungen

Die Gleichungen (9) und (10) beinhalten eine offensichtliche Ableitung nach der Zeit und eine Ableitung nach allen drei Dimensionen, die im  $\nabla$ -Operator steckt:

$$\nabla = \left( \frac{\partial}{\partial x_1}, \dots, \frac{\partial}{\partial x_n} \right) \quad (24)$$

Als erstes wird die Ableitung nach der Zeit betrachtet und mit der FDTD-Methode approximiert:

$$\nabla \times \vec{E}(t) = -\frac{[\mu_r]}{c} \frac{\partial \tilde{\vec{H}}(t)}{\partial t} \quad |FDTD - Methode \quad (25)$$

$$\underbrace{\nabla \times \vec{E}(t)}_{\text{Zeitpunkt } t} = -\frac{[\mu_r]}{c} \underbrace{\frac{\tilde{\vec{H}}(t + \Delta t) - \tilde{\vec{H}}(t)}{\Delta t}}_{\text{Zeitpunkt } t + \frac{\Delta t}{2}} \quad (26)$$

$$\nabla \times \tilde{\vec{H}}(t) = \frac{\epsilon_r}{c} \frac{\partial \vec{E}(t)}{\partial t} \quad |FDTD - Methode \quad (27)$$

$$\underbrace{\nabla \times \tilde{\vec{H}}(t)}_{\text{Zeitpunkt } t} = \frac{\epsilon_r}{c} \underbrace{\frac{\vec{E}(t + \Delta t) - \vec{E}(t)}{\Delta t}}_{\text{Zeitpunkt } t + \frac{\Delta t}{2}} \quad (28)$$

$$(29)$$

Diese Gleichungen sind aus den oben genannten Gruenden instabil. Das Problem wird geloest, indem man das H-Feld um  $\Delta t/2$  verschoben definiert. [Rumpf(h)] Daraus ergibt sich:

$$\nabla \times \vec{E}(t) = -\mu \frac{\vec{H}(t + \frac{\Delta t}{2}) - \vec{H}(t - \frac{\Delta t}{2})}{\Delta t} \quad (30)$$

$$\nabla \times \vec{H}(t + \frac{\Delta t}{2}) = \epsilon \frac{\vec{E}(t + \frac{\Delta t}{2}) - \vec{E}(t)}{\Delta t} \quad (31)$$

Nun existieren alle Terme einer Gleichung zu selben Zeitpunkten. Das gleiche Problem tritt bei der Ableitung nach dem Ort auf, doch erst muss der  $\nabla$ -Operator aufgeloeset werden, was zu den folgenden sechs Gleichungen fuehrt:

$$\frac{\partial E_z}{\partial y} - \frac{\partial E_y}{\partial z} = -\frac{1}{c}(\mu_{xx}\frac{\partial \widetilde{H}_x}{\partial t} + \mu_{xy}\frac{\partial \widetilde{H}_y}{\partial t} + \mu_{xz}\frac{\partial \widetilde{H}_z}{\partial t}) \quad (32)$$

$$\frac{\partial E_x}{\partial z} - \frac{\partial E_z}{\partial x} = -\frac{1}{c}(\mu_{yx}\frac{\partial \widetilde{H}_x}{\partial t} + \mu_{yy}\frac{\partial \widetilde{H}_y}{\partial t} + \mu_{yz}\frac{\partial \widetilde{H}_z}{\partial t}) \quad (33)$$

$$\frac{\partial E_y}{\partial x} - \frac{\partial E_x}{\partial y} = -\frac{1}{c}(\mu_{zx}\frac{\partial \widetilde{H}_x}{\partial t} + \mu_{zy}\frac{\partial \widetilde{H}_y}{\partial t} + \mu_{zz}\frac{\partial \widetilde{H}_z}{\partial t}) \quad (34)$$

$$\frac{\partial \widetilde{H}_z}{\partial y} - \frac{\partial \widetilde{H}_y}{\partial z} = \frac{1}{c}(\epsilon_{xx}\frac{\partial E_x}{\partial t} + \epsilon_{xy}\frac{\partial E_y}{\partial t} + \epsilon_{xz}\frac{\partial E_z}{\partial t}) \quad (35)$$

$$\frac{\partial \widetilde{H}_x}{\partial z} - \frac{\partial \widetilde{H}_z}{\partial x} = \frac{1}{c}(\epsilon_{yx}\frac{\partial E_x}{\partial t} + \epsilon_{yy}\frac{\partial E_y}{\partial t} + \epsilon_{yz}\frac{\partial E_z}{\partial t}) \quad (36)$$

$$\frac{\partial \widetilde{H}_y}{\partial x} - \frac{\partial \widetilde{H}_x}{\partial y} = \frac{1}{c}(\epsilon_{zx}\frac{\partial E_x}{\partial t} + \epsilon_{zy}\frac{\partial E_y}{\partial t} + \epsilon_{zz}\frac{\partial E_z}{\partial t}) \quad (37)$$

Es ist immer moeglich ein Koordinatensystem zu waehlen, sodass alle nicht diagonalen Anteile der relativen Permittivitaet und Permeabilitaet null sind, sie koennen daher als null angenommen werden, wodurch sich die Gleichungen vereinfachen [Rumpf(i)]:

$$\frac{\partial E_z}{\partial y} - \frac{\partial E_y}{\partial z} = -\frac{1}{c}\mu_{xx}\frac{\partial \widetilde{H}_x}{\partial t} \quad (38)$$

$$\frac{\partial E_x}{\partial z} - \frac{\partial E_z}{\partial x} = -\frac{1}{c}\mu_{yy}\frac{\partial \widetilde{H}_y}{\partial t} \quad (39)$$

$$\frac{\partial E_y}{\partial x} - \frac{\partial E_x}{\partial y} = -\frac{1}{c}\mu_{zz}\frac{\partial \widetilde{H}_z}{\partial t} \quad (40)$$

$$\frac{\partial \widetilde{H}_z}{\partial y} - \frac{\partial \widetilde{H}_y}{\partial z} = \frac{1}{c}\epsilon_{xx}\frac{\partial E_x}{\partial t} \quad (41)$$

$$\frac{\partial \widetilde{H}_x}{\partial z} - \frac{\partial \widetilde{H}_z}{\partial x} = \frac{1}{c}\epsilon_{yy}\frac{\partial E_y}{\partial t} \quad (42)$$

$$\frac{\partial \widetilde{H}_y}{\partial x} - \frac{\partial \widetilde{H}_x}{\partial y} = \frac{1}{c}\epsilon_{zz}\frac{\partial E_z}{\partial t} \quad (43)$$

Wendet man nun die FDTD-Methode erneut an um die Ableitung nach dem Ort zu approximieren ergeben sich:



$$\frac{E_z^{i,j+1,k}|_t - E_z^{i,j,k}|_t}{\Delta y} - \frac{E_y^{i,j,k+1}|_t - E_y^{i,j,k}|_t}{\Delta z} = -\frac{\mu_{xx}^{i,j,k}}{c} \frac{\widetilde{H}_x^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_x^{i,j,k}|_{t-\frac{\Delta t}{2}}}{\Delta t} \quad (44)$$

$$\frac{E_x^{i,j,k+1}|_t - E_x^{i,j,k}|_t}{\Delta z} - \frac{E_z^{i+1,j,k}|_t - E_z^{i,j,k}|_t}{\Delta x} = -\frac{\mu_{yy}^{i,j,k}}{c} \frac{\widetilde{H}_y^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_y^{i,j,k}|_{t-\frac{\Delta t}{2}}}{\Delta t} \quad (45)$$

$$\frac{E_y^{i+1,j,k}|_t - E_y^{i,j,k}|_t}{\Delta x} - \frac{E_x^{i,j+1,k}|_t - E_x^{i,j,k}|_t}{\Delta y} = -\frac{\mu_{zz}^{i,j,k}}{c} \frac{\widetilde{H}_z^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_z^{i,j,k}|_{t-\frac{\Delta t}{2}}}{\Delta t} \quad (46)$$

$$\frac{\widetilde{H}_z^{i,j+1,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_z^{i,j,k}|_{t+\frac{\Delta t}{2}}}{\Delta y} - \frac{\widetilde{H}_y^{i,j,k+1}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_y^{i,j,k}|_{t+\frac{\Delta t}{2}}}{\Delta z} = -\frac{\epsilon_{xx}^{i,j,k}}{c} \frac{E_x^{i,j,k}|_{t+\Delta t} - E_x^{i,j,k}|_t}{\Delta t} \quad (47)$$

$$\frac{\widetilde{H}_x^{i,j,k+1}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_x^{i,j,k}|_{t+\frac{\Delta t}{2}}}{\Delta z} - \frac{\widetilde{H}_z^{i+1,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_z^{i,j,k}|_{t+\frac{\Delta t}{2}}}{\Delta x} = -\frac{\epsilon_{yy}^{i,j,k}}{c} \frac{E_y^{i,j,k}|_{t+\Delta t} - E_y^{i,j,k}|_t}{\Delta t} \quad (48)$$

$$\frac{\widetilde{H}_y^{i+1,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_y^{i,j,k}|_{t+\frac{\Delta t}{2}}}{\Delta x} - \frac{\widetilde{H}_x^{i,j+1,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_x^{i,j,k}|_{t+\frac{\Delta t}{2}}}{\Delta y} = -\frac{\epsilon_{zz}^{i,j,k}}{c} \frac{E_z^{i,j,k}|_{t+\Delta t} - E_z^{i,j,k}|_t}{\Delta t} \quad (49)$$

$$(50)$$

Wobei  $i, j, k$  den diskreten Punkt im Raum angeben und  $\Delta x, \Delta y, \Delta z$  sowie  $\Delta t$  die Groesse der finiten Differenzen sind. Betrachtet zum Beispiel Gleichung (44), so faellt auf, dass auch hier die Stabilitaetsbedingung der FDTD-Methode nicht eingehalten wird:

$$\underbrace{\frac{E_z^{i,j+1,k}|_t - E_z^{i,j,k}|_t}{\Delta y}}_{\text{Existiert am Ort } i, j+\frac{\Delta y}{2}, k} - \underbrace{\frac{E_y^{i,j,k+1}|_t - E_y^{i,j,k}|_t}{\Delta z}}_{\text{Existiert am Ort } i, j, k+\frac{\Delta z}{2}} = -\frac{\mu_{xx}^{i,j,k}}{c} \underbrace{\frac{\widetilde{H}_x^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_x^{i,j,k}|_{t-\frac{\Delta t}{2}}}{\Delta t}}_{\text{Existiert am Ort } i, j, k} \quad (51)$$

Dieses Problem kann geloest werden indem man definiert, dass das H-Feld dem E-Feld um den Vector  $\left(\frac{\Delta x}{2}, \frac{\Delta y}{2}, \frac{\Delta z}{2}\right)$  verschoben ist. Man bezeichnet die zeitlich und raeumlich verschobenen Bezugsrahmen nach dem Mathematiker Kane S. Yee Yee-Gitter. [Yee(1966)] Durch die Verwendung des

Yee-Gitters werden die Gleichungen (1) und (2) sowie die Kontinuitaetsbedingung an Mediumsuebergaengen implizit eingehalten und muessen nicht mehr explizit behandelt werden. Fuer die raeumliche Ableitung muss nun einmal ein positiver und einmal ein negativer Offset verwendet werden. Es handelt sich dabei jedoch weiterhin um eine rechtsseitige Ableitung.

$$\frac{E_z^{i,j+1,k}|_t - E_z^{i,j,k}|_t}{\Delta y} - \frac{E_y^{i,j,k+1}|_t - E_y^{i,j,k}|_t}{\Delta z} = -\frac{\mu_{xx}^{i,j,k}}{c} \frac{\widetilde{H}_x^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_x^{i,j,k}|_{t-\frac{\Delta t}{2}}}{\Delta t} \quad (52)$$

$$\frac{E_x^{i,j,k+1}|_t - E_x^{i,j,k}|_t}{\Delta z} - \frac{E_z^{i+1,j,k}|_t - E_z^{i,j,k}|_t}{\Delta x} = -\frac{\mu_{yy}^{i,j,k}}{c} \frac{\widetilde{H}_y^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_y^{i,j,k}|_{t-\frac{\Delta t}{2}}}{\Delta t} \quad (53)$$

$$\frac{E_y^{i+1,j,k}|_t - E_y^{i,j,k}|_t}{\Delta x} - \frac{E_x^{i,j+1,k}|_t - E_x^{i,j,k}|_t}{\Delta y} = -\frac{\mu_{zz}^{i,j,k}}{c} \frac{\widetilde{H}_z^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_z^{i,j,k}|_{t-\frac{\Delta t}{2}}}{\Delta t} \quad (54)$$

$$\frac{\widetilde{H}_z^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_z^{i,j-1,k}|_{t+\frac{\Delta t}{2}}}{\Delta y} - \frac{\widetilde{H}_y^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_y^{i,j,k-1}|_{t+\frac{\Delta t}{2}}}{\Delta z} = -\frac{\epsilon_{xx}^{i,j,k}}{c} \frac{E_x^{i,j,k}|_{t+\Delta t} - E_x^{i,j,k}|_t}{\Delta t} \quad (55)$$

$$\frac{\widetilde{H}_x^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_x^{i,j,k-1}|_{t+\frac{\Delta t}{2}}}{\Delta z} - \frac{\widetilde{H}_z^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_z^{i-1,j,k}|_{t+\frac{\Delta t}{2}}}{\Delta x} = -\frac{\epsilon_{yy}^{i,j,k}}{c} \frac{E_y^{i,j,k}|_{t+\Delta t} - E_y^{i,j,k}|_t}{\Delta t} \quad (56)$$

$$\frac{\widetilde{H}_y^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_y^{i-1,j,k}|_{t+\frac{\Delta t}{2}}}{\Delta x} - \frac{\widetilde{H}_x^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_x^{i,j-1,k}|_{t+\frac{\Delta t}{2}}}{\Delta y} = -\frac{\epsilon_{zz}^{i,j,k}}{c} \frac{E_z^{i,j,k}|_{t+\Delta t} - E_z^{i,j,k}|_t}{\Delta t} \quad (57)$$

$$(58)$$

Nun koennen die Gleichungen nach den Feldstaerkewerten des jeweils naechsten Zeitschritts umgestellt werden. Es ergeben sich folgende Updategleichungen:

$$\widetilde{H}_x^{i,j,k}|_{t+\frac{\Delta t}{2}} = \widetilde{H}_x^{i,j,k}|_{t-\frac{\Delta t}{2}} - \frac{c\Delta t}{\mu_{xx}^{i,j,k}} \left( \frac{E_z^{i,j+1,k}|_t - E_z^{i,j,k}|_t}{\Delta y} - \frac{E_y^{i,j,k+1}|_t - E_y^{i,j,k}|_t}{\Delta z} \right) \quad (59)$$

$$\widetilde{H}_y^{i,j,k}|_{t+\frac{\Delta t}{2}} = \widetilde{H}_y^{i,j,k}|_{t-\frac{\Delta t}{2}} - \frac{c\Delta t}{\mu_{yy}^{i,j,k}} \left( \frac{E_x^{i,j,k+1}|_t - E_x^{i,j,k}|_t}{\Delta z} - \frac{E_z^{i+1,j,k}|_t - E_z^{i,j,k}|_t}{\Delta x} \right) \quad (60)$$

$$\widetilde{H}_z^{i,j,k}|_{t+\frac{\Delta t}{2}} = \widetilde{H}_z^{i,j,k}|_{t-\frac{\Delta t}{2}} - \frac{c\Delta t}{\mu_{zz}^{i,j,k}} \left( \frac{E_y^{i+1,j,k}|_t - E_y^{i,j,k}|_t}{\Delta x} - \frac{E_x^{i,j+1,k}|_t - E_x^{i,j,k}|_t}{\Delta y} \right) \quad (61)$$

$$E_x^{i,j,k}|_{t+\Delta t} = E_x^{i,j,k}|_t - \frac{c\Delta t}{\epsilon_{xx}^{i,j,k}} \left( \frac{\widetilde{H}_z^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_z^{i,j-1,k}|_{t+\frac{\Delta t}{2}}}{\Delta y} - \frac{\widetilde{H}_y^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_y^{i,j,k-1}|_{t+\frac{\Delta t}{2}}}{\Delta z} \right) \quad (62)$$

$$E_y^{i,j,k}|_{t+\Delta t} = E_y^{i,j,k}|_t - \frac{c\Delta t}{\epsilon_{yy}^{i,j,k}} \left( \frac{\widetilde{H}_x^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_x^{i,j,k-1}|_{t+\frac{\Delta t}{2}}}{\Delta z} - \frac{\widetilde{H}_z^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_z^{i-1,j,k}|_{t+\frac{\Delta t}{2}}}{\Delta x} \right) \quad (63)$$

$$E_z^{i,j,k}|_{t+\Delta t} = E_z^{i,j,k}|_t - \frac{c\Delta t}{\epsilon_{zz}^{i,j,k}} \left( \frac{\widetilde{H}_y^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_y^{i-1,j,k}|_{t+\frac{\Delta t}{2}}}{\Delta x} - \frac{\widetilde{H}_x^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_x^{i,j-1,k}|_{t+\frac{\Delta t}{2}}}{\Delta y} \right) \quad (64)$$

Einige Probleme, wie zum Beispiel das Simulieren eines Hornstrahlers, lassen sich aufgrund von Symmetrien im zwei- oder eindimensionalen Raum hinreichend genau beschreiben. Das reduziert die noetige Rechenleistung und wird wann immer moeglich angewandt. Fuer eine Reduktion auf den zwei- oder eindimensionalen Fall nimmt man an, dass die Differenzenquotienten der nichtbetrachteten Dimensionen gleich null sind. Jede Geometrie wird dadurch in den nichtbetrachteten Dimensionen unendlich ausgedehnt. [Rumpf(j)] Fuer die Simulation von Drahtantennen ist diese Art von Vereinfachung leider nicht moeglich, da die Geometrie eines Drahts in jede Dimension begrenzt ist und keine Symmetrie aufweist, die eine solche Annaeherung erlauben wuerde.

#### 4.4 Physikalische Stabilitaetsbedingungen fuer finite Differenzen

Wie bei der Einfuehrung der FDTD-Methode erwaeht approximiert die FDTD-Methode Differentialgleichungen indem die Differentialquotienten durch Differenzenquotienten ersetzt werden. Sie kann daher nur dann korrekte Ergebnisse liefern, wenn die Differenzenquotienten hinreichend klein sind, so sollte die Gittergroesze in jede Dimension so klein gewaehlt werden, dass die kleinste vorkommende Wellenlaenge noch mindestens 10mal abgetastet wird. Fuer die räumliche Abtaste  $N_\lambda$  muss  $N_\lambda \geq 10$  gelten [Rumpf(k)]. Fuer die Gittergroeszen  $\Delta x, \Delta y, \Delta z$  gilt daher

$$\lambda_{min} = \frac{c}{f_{max} * \eta_{max}} \quad (65)$$

$$\Delta x, \Delta y, \Delta z \leq \frac{\lambda_{min}}{N_\lambda} \quad (66)$$

$$\Delta x, \Delta y, \Delta z \leq \frac{c}{f_{max} \eta_{max} N_\lambda} \quad (67)$$

Weiterhin muss die Gittergroesze klein genug gewaehlt sein dass die minimalste Strukturbreite mindestens viermal abgetastet wird [Rumpf(k)]. Aus der ersten Bedingung heraus folgt, dass die Breite des Yee-Gitters die maximale Frequenz vorgibt die korrekt simuliert werden kann. Das wird spaeter fuer die Wahl einer geeigneten Quellenfunktion relevant werden. Da das E- und H-Feld zeitversetzt berechnet werden ist es in der numerischen Loesung nicht moeglich, dass sich eine Welle schneller als eine Zelle pro Zeitschritt ausbreitet. Der Zeitschritt  $\Delta t$  muss so klein gewaehlt werden, dass sich auch eine physikalische Welle in einem Zeitschritt nicht weiter als eine Zelle ausbreiten wuerde. Fuer den dreidimensionalen Fall wird das durch die Courant-Stabilitaetsbedingung sichergestellt [Rumpf(1)]:

$$\Delta t \leq \frac{1}{c \sqrt{\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} + \frac{1}{\Delta z^2}}} \quad (68)$$

#### 4.5 Geometrie in der FDTD

Geometrien koennen durch die ortsabhaengige relative Permittivitaet und Permeabilitaet gesetzt werden, sie geben die Materialeigenschaften an einem Gitterpunkt an. Fuer Metalle sind keine relativen Permittivitaeten definiert,

sie werden als unendlich gross angenommen. Daraus ergibt sich, dass die elektrische Feldstaerke an Metallen null ist:

$$\lim_{\epsilon \rightarrow \infty} \frac{c\Delta t}{\epsilon} = 0 \quad (69)$$

$$(70)$$

Metallgeometrien koennen platziert werden, indem die elektrischen Feldstaerkewerte nach der Berrechnung eines Zeitschrittes wieder auf null gesetzt werden, oder man einen Metallgeometriefaktor  $m \in \{0, 1\}$  einfuehrt:

$$E_x^{i,j,k}|_{t+\Delta t} = m * (E_x^{i,j,k}|_t - \frac{c\Delta t}{\epsilon_{xx}^{i,j,k}} (\frac{\widetilde{H}_z^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_z^{i,j-1,k}|_{t+\frac{\Delta t}{2}}}{\Delta y} - \frac{\widetilde{H}_y^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_y^{i,j,k-1}|_{t+\frac{\Delta t}{2}}}{\Delta z})) \quad (71)$$

$$E_y^{i,j,k}|_{t+\Delta t} = m * (E_y^{i,j,k}|_t - \frac{c\Delta t}{\epsilon_{yy}^{i,j,k}} (\frac{\widetilde{H}_x^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_x^{i,j,k-1}|_{t+\frac{\Delta t}{2}}}{\Delta z} - \frac{\widetilde{H}_z^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_z^{i-1,j,k}|_{t+\frac{\Delta t}{2}}}{\Delta x})) \quad (72)$$

$$E_z^{i,j,k}|_{t+\Delta t} = m * (E_z^{i,j,k}|_t - \frac{c\Delta t}{\epsilon_{zz}^{i,j,k}} (\frac{\widetilde{H}_y^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_y^{i-1,j,k}|_{t+\frac{\Delta t}{2}}}{\Delta x} - \frac{\widetilde{H}_x^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_x^{i,j-1,k}|_{t+\frac{\Delta t}{2}}}{\Delta y})) \quad (73)$$

Dieser Faktwor wird in der Implementierung der FDTD-Methode genutzt, der Uebersichtlichkeit halber aber hier nicht weiter mitgefuehrt.

#### 4.6 Quellen in der FDTD-Methode

Neben der Frage wie sich Geometrien in der FDTD-Methode setzen lassen stellt sich die Frage nach Quellen in der FDTD-Methode. Eine Quelle ist eine, wie auch immer geartete Erregung, des elektrischen oder magnetischen Feldes. Dabei kann eine Erregung als Addition passieren, man spricht dann von einer sog. ‘Soft Source’ oder als Zuweisung, man spricht dann von einer sog. ‘Hard Source’. [Rumpf(m)] Da bei einer Hard Source, aehnlich wie bei einem Metall, ein Feldstaerkewert vorgegeben wird kommt es an ihr zu Reflektionen. Hauefig wird eine Quelle auch als Spannungsdifferenz zwischen zwei Gitterpunkten implementiert, man spricht dann von einer sog. ‘Gap-Source’, die sich fuer eine Punktquellle in X-Richtung des E-Feldes an der Stelle q wie folgt berrechnet [Luebbers and H.S()]:

$$E_x^{q_x, q_y, q_z} \big|_t = u_q(t) / \Delta x \quad (74)$$

Wobei  $u_q(t)$  die Quellspannung ist. Es wurde gezeigt, dass sich die noetigen Zeitschritte fuer einen eingeschwungenen Zustand einer Antenne durch die Wahl einer geeigneten Quelle reduzieren lassen [Luebbers and H.S()], etwas worauf ich der Einfachkeit halber verzichtet habe. Neben der Art der Quelle beeinflusst auch das Quellensignal die Simulation. Da die FDTD nur bis zu einer maximalen Frequenz stabil ist darf kein Quellensignal verwendet werden, dass Frequenzen ueberhalb der maximalen Frequenz aufweist. Zeitlich stark begrenzte Signale mit einem unendlich ausgedehnten Spektrum wie zum Beispiel ein Rechteck- oder Deltaimpuls fuehren immer zu instabilem Verhalten. Soll betrachtet werden wie sich ein System bei unterschiedlichen Frequenzen verhaelt kann ein Gauzimpuls genutzt werden, dessen Fourier-transformierte, wieder ein Gauszimpuls, fuer die maximale Frequenz eine Amplitude nahe null angenommen hat. Fuer die Simulation von Antennen bietet sich ein Sinus mit der gewuenschten Sendefrequenz an. Da die FDTD-Methode ein numerisches Loesungsverfahren darstellt beginnt die Berrechnung an einem diskreten Zeitpunkt. Fuer die sinusfoermige Quellenspannung bedeutet das  $q(t) = 0, \forall t \in [-\infty, 0]$ . Analytisch bedeutet das eine Multiplikation mit der Heaviside-Funktion, die im Frequenzbereich einer Faltung mit  $\frac{1}{j\omega}$  entspricht. Um die Verzerrung des Quellenspektrums moeglichst gering zu halten wird daher meist ein Sinus verwendet, dessen Amplitude sich kontinuierlich auf die gewuenschte Zielamplitude vergroesert [Rumpf(n)].

## 4.7 Numerisches Randwertproblem

Eine numerische Loesung einer Differentialgleichung ist immer nur in einem begrenzten Problembereich moeglich. Betrachtet man jedoch die Update-funktionen, so faellt auf, dass die rauemliche Ableitung fuer einen Feldstaerkewert am Rand des Problembereichs einen Feldstaerkewert von ausserhalb des Problembereichs benoetigt. Die einfachste Moeglichkeit dieses Randwertproblem zu loesen ist die sog. Dirichlet-Randbedingung. Es wird angenommen, dass die Feldstaerkewerte ausserhalb des Problembereichs null sind [Rumpf(o)]. Das entspricht wie oben beschrieben jedoch der Eigenschaft eines perfekten Leiters. Durch die Dirichlet-Randbedingungen gibt es eine Totalreflektion am Rand des Problembereichs. Neben der Dirichlet-Randbedingung gibt es noch sog. Absorbierende Randbedingungen (ABCs), die versuchen die Ausbreitung der EM-Wellen vorrauszusagen und Feldstaerken ausserhalb des Problembereichs annaehnern koennen.

## 4.8 Perfectly Matched Layer (PML)

ABCs fordern dass der Winkel oder die Frequenz der EM-Welle bekannt ist um einen Anhaltspunkt zu haben anhand dessen die Ausbreitung der EM-Welle vorausgesagt werden kann. Fuer die Simulation von Antennen koennen solche Angaben jedoch nicht getroffen werden, es koennen daher keine ABCs verwendet werden. Um zu verhindern, dass abgestrahlte Wellen in den Problembereich zurueckreflektiert werden und die Abstrahlcharakteristik verfaelschen muss der unendlich ausgedehnte Freiraum simuliert werden. Das geschieht durch ein sog. 'Pefectly Matched Layer' (PML), ein nur in der Theorie existierendes anisotropes Material, das Wellen aller Frequenzen und unter jedem Winkel reflektionsfrei passieren laesst und dennoch verlustbehaftet ist [Schneider()]. Abgestrahlte Wellen werden absorbiert bevor sie am Rand des Problembereichs reflektiert werden. Folgend werde ich das PML herleiten, dafuer werden die Maxwellgleichungen zuerst im Frequenzbereich betrachtet und anschliessend wieder in den Zeitbereich zuruecktransformiert [Rumpf(p)]. Im Frequenzbereich lassen sich verlustbehaftete Materialien durch eine komplexe Permittivitaet ausdruecken. [Rumpf(q)] Die Herleitung wird nur fuer X-Komponente des E- und des H-Feldes passieren da sie auf gleiche Weise fuer die anderen Komponenten durchgefuehrt werden kann. Sie werden folgend mit I und II nummeriert. Damit die keine Reflexion und keine Brechung stattfindet muss die Impedanz des PML gleich der Impedanz des umliegenden Materials sein, da der unendliche Freiraum simuliert werden soll wird es zu einer Mediumsgrenze von Vacuum zu PML geben. Die Impedanz des PML muss daher ueberall 1 sein. Aus der Formel der Impedanz folgt, dass dafuer die relative Permittivitaet gleich der relativen Permeabilitaet sein muss. [Rumpf(r)]

$$\eta = \sqrt{\mu_r \epsilon_r} \stackrel{!}{=} 1 \quad (75)$$

$$\Rightarrow \mu_r = \epsilon_r \quad (76)$$

Da  $\mu_r$  und  $\epsilon_r$  gleich sind wird ein neuer Parameter  $s$  eingefuehrt. Dieser hat die wie  $\mu_r$  und  $\epsilon$  die Form einer Diagonalmatrix:

$$[s] = [\mu_r] = [\epsilon_r] \quad (77)$$

$$[s] = \begin{bmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \end{bmatrix} \quad (78)$$

Aus dem Selliuschen Brechungsgesetz fuer anisotrope Medien in Z-Richtung und der Bedingung  $\eta_1 = 1$  folgt:

$$\eta_1 \sin \theta_1 = \sqrt{bc} \sin \theta_2 \quad , \eta_1 = 1 \quad (79)$$

$$\sin \theta_1 = \sqrt{bc} \sin \theta_2 \quad (80)$$

Da keine Brechung vorliegen soll muss  $\theta_1 = \theta_2$  gelten, wodurch:

$$\sqrt{bc} = 1 \quad (81)$$

$$c = \frac{1}{b} \quad (82)$$

gelten muss. Mit  $\theta_1 = \theta_2$  vereinfacht sich das Fresnel-Gesetz fuer anisotrope Medien in Z-Richtung zu einer winkelunabhaengigen Form:

$$r_{TE} = \frac{\sqrt{a} \cos \theta_1 - \sqrt{b} \cos \theta_2}{\sqrt{a} \cos \theta_1 + \sqrt{b} \cos \theta_2} \quad (83)$$

$$= \frac{\sqrt{a} - \sqrt{b}}{\sqrt{a} + \sqrt{b}} \quad (84)$$

$$r_{TM} = \frac{\sqrt{a} \cos \theta_2 - \sqrt{b} \cos \theta_1}{\sqrt{a} \cos \theta_1 + \sqrt{b} \cos \theta_2} \quad (85)$$

$$= \frac{-\sqrt{a} + \sqrt{b}}{\sqrt{a} + \sqrt{b}} \quad (86)$$

Da  $r_{TE} = 0$  und  $r_{TM} = 0$  gelten soll muss:

$$\sqrt{a} - \sqrt{b} = 0 \quad (87)$$

$$\sqrt{a} = \sqrt{b} \quad (88)$$

$$a = b \quad (89)$$

gelten. Mit Gleichung (82) ergibt sich nun fuer  $[s_z]$ :

$$[s_z] = \begin{bmatrix} a & 0 & 0 \\ 0 & a & 0 \\ 0 & 0 & \frac{1}{a} \end{bmatrix} = \begin{bmatrix} b & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & \frac{1}{b} \end{bmatrix} \quad | s_z = a = b \in \mathbb{C} \quad (90)$$

$$[s_z] = \begin{bmatrix} sz & 0 & 0 \\ 0 & sz & 0 \\ 0 & 0 & \frac{1}{sz} \end{bmatrix} \quad (91)$$



Durch Rotation erhaelt man die Parameter  $[s_x]$  und  $[s_y]$  die eine Medium mit relfektionsfreier Mediumsgrenze in X- und Y-Richtung beschreibt.

$$[s_x] = \begin{bmatrix} \frac{1}{s_x} & 0 & 0 \\ 0 & s_x & 0 \\ 0 & 0 & s_x \end{bmatrix} \quad (92)$$

$$[s_y] = \begin{bmatrix} s_y & 0 & 0 \\ 0 & \frac{1}{s_y} & 0 \\ 0 & 0 & s_y \end{bmatrix} \quad (93)$$

Durch eine Multiplikation erhaelt man einen Parameter  $[s]$ , der ein reflektionsfreies Material fuer alle Richtungen beschreibt:

$$[s] = [s_x] * [s_y] * [s_y] \quad (94)$$

$$= \begin{bmatrix} \frac{s_y s_z}{s_x} & 0 & 0 \\ 0 & \frac{s_x s_z}{s_y} & 0 \\ 0 & 0 & \frac{s_x s_y}{s_z} \end{bmatrix} \quad (95)$$

Die Parameter  $s_x$ ,  $s_y$  und  $s_z$  sind komplex und werden nur innerhalb ihrer jeweiligen PML-Schichten ungleich 1. Im nun etwas kleiner gewordenen Problemereich gilt:  $s_x = s_y = s_z = 1$ . Innerhalb des PML-Materials steigen die Parameter kontinuierlich an:

$$s_x(x) = 1 + \frac{\sigma'(x)}{j\omega\epsilon_0}, \quad \sigma'(x) = \frac{\epsilon_0}{2\Delta t} \frac{x^3}{L_x}, \quad x \in [0, L_x] \quad (96)$$

Wobei  $L_x$  die Laenge des PML-Materials in X-Richtung ist. Die Gleichung gilt analog fuer  $s_y$  und  $s_z$ . Dieser kontinuierliche Anstieg ist nur aufgrund des Diskretisierungsfehlers noetig. Der nun vollstaendig beschriebene Parameter  $[s]$  wird in den Maxwellgleichungen als zusaetzlzliche Materialeigenschaft verwendet. Im Frequenzbereich ergibt sich dadurch:

$$\nabla \times \vec{H} = j\omega\epsilon_0[\epsilon_r][s]\vec{E} \quad (97)$$

$$\nabla \times \vec{E} = -j\omega\mu_0[\mu_r][s]\vec{H} \quad (98)$$

Normalisiert man das H-Feld ergibt sich:

$$\nabla \times \vec{H} = j\omega \frac{[\epsilon_r][s]}{c} \vec{E} \quad (99)$$

$$\nabla \times \vec{E} = -j\omega \frac{[\mu_r][s]}{c} \vec{H} \quad (100)$$

Fuer die Ruecktransformation umgestellt ergibt sich:

$$\text{I} : j\omega \left(1 + \frac{\sigma'_x}{j\omega\epsilon_0}\right)^{-1} \left(1 + \frac{\sigma'_y}{j\omega\epsilon_0}\right) \left(1 + \frac{\sigma'_z}{j\omega\epsilon_0}\right) E_x = \frac{c}{\epsilon_{xx}} \left(\frac{\partial H_z}{\partial y} - \frac{\partial H_y}{\partial z}\right) \quad | * \left(1 + \frac{\sigma'_x}{j\omega\epsilon_0}\right) \quad (101)$$

$$\text{II} : j\omega \left(1 + \frac{\sigma'_x}{j\omega\epsilon_0}\right)^{-1} \left(1 + \frac{\sigma'_y}{j\omega\epsilon_0}\right) \left(1 + \frac{\sigma'_z}{j\omega\epsilon_0}\right) H_x = -\frac{c}{\mu_{xx}} \left(\frac{\partial E_z}{\partial y} - \frac{\partial E_y}{\partial z}\right) \quad | * \left(1 + \frac{\sigma'_x}{j\omega\epsilon_0}\right) \quad (102)$$

$$\text{I} : j\omega \left(1 + \frac{\sigma'_y}{j\omega\epsilon_0}\right) \left(1 + \frac{\sigma'_z}{j\omega\epsilon_0}\right) E_x = \frac{c}{\epsilon_{xx}} \underbrace{\left(\frac{\partial H_z}{\partial y} - \frac{\partial H_y}{\partial z}\right)}_{C_x^H} \left(1 + \frac{\sigma'_x}{j\omega\epsilon_0}\right) \quad (103)$$

$$\text{II} : j\omega \left(1 + \frac{\sigma'_y}{j\omega\epsilon_0}\right) \left(1 + \frac{\sigma'_z}{j\omega\epsilon_0}\right) H_x = -\frac{c}{\mu_{xx}} \underbrace{\left(\frac{\partial E_z}{\partial y} - \frac{\partial E_y}{\partial z}\right)}_{C_x^E} \left(1 + \frac{\sigma'_x}{j\omega\epsilon_0}\right) \quad (104)$$

$$\text{I} : j\omega E_x + \frac{\sigma'_z + \sigma'_y}{\epsilon_0} E_x + \frac{1}{j\omega} \frac{\sigma'_y \sigma'_z}{\epsilon_0^2} E_x = \frac{c}{\epsilon_{xx}} C_x^H + \frac{1}{j\omega} \frac{c\sigma'_x}{\epsilon_{xx}\epsilon_0} C_x^H \quad (105)$$

$$\text{II} : j\omega H_x + \frac{\sigma'_z + \sigma'_y}{\epsilon_0} H_x + \frac{1}{j\omega} \frac{\sigma'_y \sigma'_z}{\epsilon_0^2} H_x = \frac{c}{\mu_{xx}} C_x^E - \frac{1}{j\omega} \frac{c\sigma'_x}{\mu_{xx}\epsilon_0} C_x^E \quad (106)$$

Die Ruecktransformation kann aufgrund der Linearitaetseigenschaft der Fouriertransformation fuer jeden Term einzeln durchgefuehrt. Der Übersichtlichkeit halber wird die Ruecktransformation fuer die beiden Gleichungen getrennt vorgenommen. Die Korrespondenztabelle fuer I ist:

$$j\omega E_x(\omega) \circ \bullet \frac{\partial E_x(t)}{\partial t} \quad (107)$$

$$\frac{\sigma_z' + \sigma_y'}{\epsilon_0} E_x(\omega) \circ \bullet \frac{\sigma_z' + \sigma_y'}{\epsilon_0} E_x(t) \quad (108)$$

$$\frac{1}{j\omega} \frac{E_x(\omega) \sigma_y' \sigma_z'}{\epsilon_0^2} \circ \bullet \int_{-\infty}^t \frac{\sigma_y' \sigma_z'}{\epsilon_0^2} E_x(\tau) d\tau \quad (109)$$

$$\frac{c}{\epsilon_{xx}} C_x^H(\omega) \circ \bullet \frac{c}{\epsilon_{xx}} C_x^H(t) \quad (110)$$

$$\frac{1}{j\omega} \frac{c}{\epsilon_{xx}} \frac{\sigma_x'}{\epsilon_0} C_x^H(\omega) \circ \bullet \int_{-\infty}^t \frac{c \sigma_x'}{\epsilon_{xx} \epsilon_0} C_x^H(\tau) d\tau \quad (111)$$

$$(112)$$

Daraus ergibt sich fuer I:

$$\frac{\partial E_x(t)}{\partial t} + \frac{\sigma_z' + \sigma_y'}{\epsilon_0} E_x(t) + \int_{-\infty}^t \frac{\sigma_y' \sigma_z'}{\epsilon_0^2} E_x(\tau) d\tau = \frac{c}{\epsilon_{xx}} C_x^H(t) + \int_{-\infty}^t \frac{c \sigma_x'}{\epsilon_{xx} \epsilon_0} C_x^H(\tau) d\tau \quad (113)$$

Fuer II ist die Korrespondenztabelle:

$$j\omega H_x(\omega) \circ \bullet \frac{\partial H_x(t)}{\partial t} \quad (114)$$

$$\frac{\sigma_z' + \sigma_y'}{\epsilon_0} H_x(\omega) \circ \bullet \frac{\sigma_z' + \sigma_y'}{\epsilon_0} H_x(t) \quad (115)$$

$$\frac{1}{j\omega} \frac{H_x(\omega) \sigma_y' \sigma_z'}{\epsilon_0^2} \circ \bullet \int_{-\infty}^t \frac{\sigma_y' \sigma_z'}{\epsilon_0^2} H_x(\tau) d\tau \quad (116)$$

$$-\frac{c}{\mu_{xx}} C_x^E(\omega) \circ \bullet -\frac{c}{\mu_{xx}} C_x^E(t) \quad (117)$$

$$-\frac{1}{j\omega} \frac{c}{\mu_{xx}} \frac{\sigma_x'}{\epsilon_0} C_x^E(\omega) \circ \bullet -\int_{-\infty}^t \frac{c \sigma_x'}{\epsilon_{xx} \epsilon_0} C_x^E(\tau) d\tau \quad (118)$$

$$(119)$$

Daraus ergibt sich fuer II:

$$\frac{\partial H_x(t)}{\partial t} + \frac{\sigma'_z + \sigma'_y}{\epsilon_0} H_x(t) + \int_{-\infty}^t \frac{\sigma'_y \sigma'_z}{\epsilon_0^2} H_x(\tau) d\tau = -\frac{c}{\mu_{xx}} C_x^E(t) - \int_{-\infty}^t \frac{c \sigma'_x}{\epsilon_{xx} \epsilon_0} C_x^E(\tau) d\tau \quad (120)$$

$$(121)$$

Die transformierten Terme koennen nun separat ueber die FDTD-Methode angenaehert werden. Dadurch ergibt sich fuer I:

$$\frac{\partial E_x(t)}{\partial t} \approx \frac{E_x^{i,j,k}|_{t+\Delta t} - E_x^{i,j,k}|_t}{\delta t} \quad (122)$$

$$\frac{\sigma'_z + \sigma'_y}{\epsilon_0} E_x(t) \approx \frac{\sigma'_z + \sigma'_y}{\epsilon_0} E_x^{i,j,k}|_t \quad (123)$$

$$\int_{-\infty}^t \frac{\sigma'_z \sigma'_y}{\epsilon_0^2} E_x(\tau) d\tau \approx \frac{\sigma'_z \sigma'_y}{\epsilon_0^2} \sum_{T=0}^t E_x^{i,j,k}|_T \Delta t \quad (124)$$

$$\frac{c}{\epsilon_{xx} i} C_x^H(t) \approx \frac{c}{\epsilon_{xx}} \left( \frac{\widetilde{H}_z^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_z^{i,j-1,k}|_{t+\frac{\Delta t}{2}}}{\Delta y} - \frac{\widetilde{H}_y^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_y^{i,j,k-1}|_{t+\frac{\Delta t}{2}}}{\Delta z} \right) \quad (125)$$

$$\int_{-\infty}^t \frac{c \sigma'_x}{\epsilon_{xx} \epsilon_0} C_x^H(\tau) d\tau \approx \frac{c \sigma'_x \Delta t}{\epsilon_{xx} \epsilon_0} \sum_{T=0}^t \left( \frac{\widetilde{H}_z^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_z^{i,j-1,k}|_{t+\frac{\Delta t}{2}}}{\Delta y} - \frac{\widetilde{H}_y^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_y^{i,j,k-1}|_{t+\frac{\Delta t}{2}}}{\Delta z} \right) \quad (126)$$

$$\frac{c}{\epsilon_{xx}} \left( \frac{\widetilde{H}_z^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_z^{i,j-1,k}|_{t+\frac{\Delta t}{2}}}{\Delta y} - \frac{\widetilde{H}_y^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_y^{i,j,k-1}|_{t+\frac{\Delta t}{2}}}{\Delta z} \right) + \frac{c \sigma'_x \Delta t}{\epsilon_{xx} \epsilon_0} \sum_{T=0}^t \left( \frac{\widetilde{H}_z^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_z^{i,j-1,k}|_{t+\frac{\Delta t}{2}}}{\Delta y} \right) \quad (127)$$

Insgesamt also:

$$\frac{E_x^{i,j,k}|_{t+\Delta t} - E_x^{i,j,k}|_t}{\delta t} + \frac{\sigma'_z + \sigma'_y}{\epsilon_0} E_x^{i,j,k}|_t + \frac{\sigma'_z \sigma'_y}{\epsilon_0^2} \sum_{T=0}^t E_x^{i,j,k}|_T \Delta t = \quad (128)$$

$$\frac{c}{\epsilon_{xx}} \left( \frac{\widetilde{H}_z^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_z^{i,j-1,k}|_{t+\frac{\Delta t}{2}}}{\Delta y} - \frac{\widetilde{H}_y^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_y^{i,j,k-1}|_{t+\frac{\Delta t}{2}}}{\Delta z} \right) \quad (129)$$

$$+ \frac{c \sigma'_x \Delta t}{\epsilon_{xx} \epsilon_0} \sum_{T=0}^t \left( \frac{\widetilde{H}_z^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_z^{i,j-1,k}|_{t+\frac{\Delta t}{2}}}{\Delta y} - \frac{\widetilde{H}_y^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_y^{i,j,k-1}|_{t+\frac{\Delta t}{2}}}{\Delta z} \right) \quad (130)$$

Fuer II ergibt sich:

$$\frac{\partial H_x(t)}{\partial t} \approx \frac{\widetilde{H}_x^{i,j,k}|_{t+\frac{\Delta t}{2}} - \widetilde{H}_x^{i,j,k}|_{t-\frac{\Delta t}{2}}}{\Delta t} \quad (131)$$

$$\frac{\sigma'_z + \sigma'_y}{\epsilon_0} H_x(t) \approx \frac{\sigma'_z + \sigma'_y}{\epsilon_0} \frac{\widetilde{H}_x^{i,j,k}|_{t+\frac{\Delta t}{2}} + \widetilde{H}_x^{i,j,k}|_{t-\frac{\Delta t}{2}}}{2} \quad (132)$$

$$\int_{-\infty}^t \frac{\sigma'_z \sigma'_y}{\epsilon_0^2} H_x(\tau) d\tau \approx \frac{\sigma'_z \sigma'_y \Delta t}{\epsilon_0^2} \left( \sum_{T=\frac{\Delta t}{2}}^{t+\frac{\Delta t}{2}} \widetilde{H}_x^{i,j,k}|_T \right) \quad (133)$$

$$- \frac{c}{\mu_{xx}} C_x^E(t) \approx - \frac{c}{\mu_{xx}} \left( \frac{E_z^{i,j,k}|_{t+\Delta t} - E_z^{i,j-1,k}|_{t+\Delta t}}{\Delta y} - \frac{E_y^{i,j,k}|_{t+\Delta t} - E_y^{i,j,k-1}|_{t+\Delta t}}{\Delta z} \right) \quad (134)$$

$$- \int_{-\infty}^t \frac{c \sigma'_x}{\mu_{xx} \epsilon_0} C_x^E(\tau) d\tau \approx - \frac{c \sigma'_x \Delta t}{\mu_{xx} \epsilon_0} \sum_{T=0}^t \left( \frac{E_z^{i,j,k}|_t - E_z^{i,j-1,k}|_t}{\Delta y} - \frac{E_y^{i,j,k}|_t - E_y^{i,j,k-1}|_t}{\Delta z} \right) \quad (135)$$

Betrachtet man Gleichung (133) so faellt auf, dass die magnetischen Feldstaerkewerte bis zum Zeitpunkt  $t + \frac{\Delta t}{2}$  aufsummiert werden, es wird einen halben Zeitschritt zu weit integriert. Das Problem wird geloest, indem man bis zum Zeitpunkt  $t - \frac{\Delta t}{2}$  integriert und den letzten halben Zeitschritt interpoliert.

$$\int_{-\infty}^t \frac{\sigma'_z \sigma'_y}{\epsilon_0^2} H_x(\tau) d\tau \approx \frac{\sigma'_z \sigma'_y \Delta t}{\epsilon_0^2} \left( \sum_{T=\frac{\Delta t}{2}}^{t+\frac{\Delta t}{2}} \widetilde{H}_x^{i,j,k} \Big|_T \right) \quad (136)$$

$$= \frac{\sigma'_z \sigma'_y \Delta t}{\epsilon_0^2} \sum_{T=\frac{\Delta t}{2}}^{t-\frac{\Delta t}{2}} \widetilde{H}_x^{i,j,k} \Big|_T + \frac{\widetilde{H}_x^{i,j,k} \Big|_{t+\frac{\Delta t}{2}} + \widetilde{H}_x^{i,j,k} \Big|_{t-\frac{\Delta t}{2}}}{2} \frac{\Delta t}{2} \quad (137)$$

$$= \frac{\sigma'_z \sigma'_y \Delta t}{\epsilon_0^2} \left( \sum_{T=\frac{\Delta t}{2}}^{t-\frac{\Delta t}{2}} \widetilde{H}_x^{i,j,k} \Big|_T + \frac{\widetilde{H}_x^{i,j,k} \Big|_{t+\frac{\Delta t}{2}} + \widetilde{H}_x^{i,j,k} \Big|_{t-\frac{\Delta t}{2}}}{4} \right) \quad (138)$$

Insgesamt ergibt sich also:

$$\frac{\widetilde{H}_x^{i,j,k} \Big|_{t+\frac{\Delta t}{2}} - \widetilde{H}_x^{i,j,k} \Big|_{t-\frac{\Delta t}{2}}}{\Delta t} + \frac{\sigma'_z + \sigma'_y}{\epsilon_0} \frac{\widetilde{H}_x^{i,j,k} \Big|_{t+\frac{\Delta t}{2}} + \widetilde{H}_x^{i,j,k} \Big|_{t-\frac{\Delta t}{2}}}{2} \quad (139)$$

$$+ \frac{\sigma'_z \sigma'_y \Delta t}{\epsilon_0^2} \left( \sum_{T=\frac{\Delta t}{2}}^{t-\frac{\Delta t}{2}} \widetilde{H}_x^{i,j,k} \Big|_T + \frac{\widetilde{H}_x^{i,j,k} \Big|_{t+\frac{\Delta t}{2}} + \widetilde{H}_x^{i,j,k} \Big|_{t-\frac{\Delta t}{2}}}{4} \right) \quad (140)$$

$$= -\frac{c}{\mu_{xx}} \left( \frac{E_z^{i,j,k} \Big|_{t+\Delta t} - E_z^{i,j-1,k} \Big|_{t+\Delta t}}{\Delta y} - \frac{E_y^{i,j,k} \Big|_{t+\Delta t} - E_y^{i,j,k-1} \Big|_{t+\Delta t}}{\Delta z} \right) \quad (141)$$

$$- \frac{c \sigma'_x \Delta t}{\mu_{xx} \epsilon_0} \sum_{T=0}^t \left( \frac{E_z^{i,j,k} \Big|_t - E_z^{i,j-1,k} \Big|_t}{\Delta y} - \frac{E_y^{i,j,k} \Big|_t - E_y^{i,j,k-1} \Big|_t}{\Delta z} \right) \quad (142)$$

Loest man nun I nach  $E_x^{i,j,k} \Big|_{t+\Delta t}$  und II nach  $\widetilde{H}_x^{i,j,k} \Big|_{t+\frac{\Delta t}{2}}$  auf erhaelt man die folgenden Updategleichungen:

$$\text{I: } E_x^{i,j,k}|_{t+\Delta t} = E_x^{i,j,k}|_t \left(1 - \frac{\Delta t(\sigma'_z + \sigma'_y)}{\epsilon_0}\right) - \frac{\sigma'_z \sigma'_y \Delta t^2}{\epsilon_0^2} \sum_{T=0}^t E_x^{i,j,k}|_T \quad (143)$$

$$+ \frac{c\Delta t}{\epsilon_{xx}} C_x^H|_{t+\frac{\Delta t}{2}}^{i,j,k} + \frac{c\sigma'_x \Delta t^2}{\epsilon_{xx}\epsilon_0} \sum_{T=0}^t C_x^H|_{T+\frac{\Delta t}{2}}^{i,j,k} \quad (144)$$

$$\text{II: } \widetilde{H}_x^{i,j,k}|_{t+\frac{\Delta t}{2}} = \frac{-\frac{\sigma'_y + \sigma'_z}{2\epsilon_0} + \frac{\sigma'_y \sigma'_z \Delta t}{4\epsilon_0} + \frac{1}{\Delta t}}{\frac{\sigma'_y + \sigma'_z}{2\epsilon_0} + \frac{\sigma'_y \sigma'_z \Delta t}{4\epsilon_0} + \frac{1}{\Delta t}} \widetilde{H}_x^{i,j,k}|_{t-\frac{\Delta t}{2}} - \frac{\frac{x}{\mu_{xx}}}{\frac{\sigma'_y + \sigma'_z}{2\epsilon_0} + \frac{\sigma'_y \sigma'_z \Delta t}{4\epsilon_0} + \frac{1}{\Delta t}} C_x^E|_t^{i,j,k} \quad (145)$$

$$- \frac{\frac{c\sigma'_x \Delta t}{\mu_{xx}\epsilon_0}}{\frac{\sigma'_y + \sigma'_z}{2\epsilon_0} + \frac{\sigma'_y \sigma'_z \Delta t}{4\epsilon_0} + \frac{1}{\Delta t}} \sum_{T=0}^t C_x^E|_T^{i,j,k} - \frac{\frac{\sigma'_y \sigma'_z \Delta t}{\epsilon_0^2}}{\frac{\sigma'_y + \sigma'_z}{2\epsilon_0} + \frac{\sigma'_y \sigma'_z \Delta t}{4\epsilon_0} + \frac{1}{\Delta t}} \sum_{T=\frac{\Delta t}{2}}^{t-\frac{\Delta t}{2}} \widetilde{H}_x^{i,j,k}|_T \quad (146)$$

$$(147)$$

Um zu ueberpruefen ob die hergeleiteten Gleichungen (144) und (145) korrekt sind kann man alle  $\sigma'$ -Terme null setzten. Sie muessen sich dann zu den Gleichungen (59) und (62) vereinfachen, was sie tun.

## 5 Implementierung der FDTD

Die FDTD habe ich einmal sequentiell in C und einmal parallelisiert in C und OpenCL implementiert. Die Referenzimplementierung in C wurde genutzt um die Ergebnisse der parallelen Implementierungen zu ueberpruefen und die Performance der Implementierungen vergleichen zu koennen. Ich habe die FDTD auf zwei unterschiedliche Weisen implementiert, einmal wird nur die Berrechnung eines Zeitschrittes auf die GPU ausgelagert und einmal alle. Beide Implementierungen sind fuer Flieszkommazahlen einfacher und doppelter Genauigkeit definiert.

### 5.1 OpenCL

OpenCL ist eine von der Khronos Group standardisierte Architektur fuer parallele Programmierung. [Group()] Eine OpenCL Applikation besteht aus einem Hostprogramm, der auf der CPU ausgefuehrt werden und einem oder mehrerer sog. Kernel, die parallelisiert auf der Beschleunigerhardware ausgefuehrt werden. [Munshi et al.(2011)] Munshi, Gaster, Mattson, Fung, and

Ginsburg, p.13] In der OpenCL Architektur existiert immer nur ein Host, der mit einem oder mehreren ‘Compute Devices’ verbunden ist. Ein Compute Device kann eine CPU, GPU oder ein Digitaler Signal Prozessor (DSP) sein. Ein Compute Device ist unterteilt in sog. Compute Units, die in sog. Processing Elements unterteilt sind, dort werden die Kernel tatsaechlich ausgefuehrt. [Munshi et al.(2011)Munshi, Gaster, Mattson, Fung, and Ginsburg, p.12] Moechte man einen Algorithmus in OpenCL implementieren so muss ein Kernel in OpenCL implementiert werden und ein globaler Arbeitsbereich definiert werden. Dieser globale Arbeitsbereich ist ein-, zwei-, oder dreidimensional und kann in lokale Arbeitsbereiche unterteilt werden. Lokale Arbeitsbereiche werden parallel von Compute Units ausgefuehrt, die ihrerseits Kernel konkurrierend auf ihren Processing Units ausfuehrt. Eine Instanz eines lokalen Arbeitsbereichs nennt man Workgroup, die Instanz eines Kernels nennt man Workitem. Jedem Workitem ist bekannt zu welcher Workgroup er gehoert und an welcher Stelle er im globalen Problembereich er steht. Da Workgroups auf getrennten Compute Units ausgefuehrt werden ist keine Synchronisation unter Kerneln unterschiedlicher Workgroups moeglich.

## 5.2 Implementieren der Dirichlet-Boundary-Condition

Die Dirichlet-Boundary-Condition loest den Spezialfall fuer Zellen am Rand des simulierten Raums. Mathematisch vereinfachen sich fuer Zellen am Rand die Updategleichungen, da Feldstaerkewerte ausserhalb des Simulationsbereichs als null angenommen werden. Fuer die Implementierung der Dirichlet-Boundary-Condition gibt es sequentiell und parallel mehrere Moeglichkeiten, die ich besprechen moechte, da sie die Form der letztendlichen Implementierung beeinflusst hat. Sequentiell kann man die Dirichlet-Boundary-Condition implementieren indem man die Berrechnung der Randfeldstaerkewerte auslagert und die vereinfachten Formeln nutzt. Das bietet sich fuer eine parallele Implementierung nicht an, da entweder fuer jeden Rand ein eigener Kernel implementiert und aufgerufen werden muesste, was umstaendlich ist, oder aber eine Kontrollstruktur genutzt werden muesste die Randfaelle erkennt und behandelt, was unperformant ist. Eine weitere Moeglichkeit ist es in der sequentiellen Implementierung die Iteration ueber den Simulationsraum nur von dem Index  $(1, 1, 1)$  bis  $N_x - 1, N_y - 1, N_z - 1$  laufen zu lassen. Somit ist der tatsaechliche Simulationsraum in jede Dimension um zwei kleiner, da der Rand nicht mit simuliert wird. Die Feldstaerkewerte am Rand bleiben bei ihrem Intialisierungswert null, wodurch die Dirichle-Boundary-Condition implizit eingehalten wird. Dieses Verfahren kann aehn-



lich auch in der parallelen Implementierung genutzt werden. Dafuer muss der globale Arbeitsbereich in jeder Dimension als zwei kleiner definiert werden als das zugrundeliegende Feldstaerkearray. Innerhalb des OpenCL-Kernels kann nun der globale Index mit einem Offset von eins fuer jede Dimension errechnet werden. Dadurch bleiben die Raender der Feldstaerkewerte unsimuliert und die Dirichlet-Boundary-Condition implizit implementiert. Zwischenzeitlich erschien es auch moeglich die Dirichlet-Boundary-Condition durch die OpenCL-Texturroutinen implementieren zu koennen. Dafuer haetten die alten und neuen E- und H-Felder als Texturen mit einem RGB-Farbkanal und einem float-Pixelformat dargestellt werden muessen. Fuer den Zugriff auf Texturen muss in OpenCL ein sog. Sampler definiert werden dem CLK\_ADDRESS\_CLAMP als Parameter uebergeben werden kann. Das veranlasst den Sampler bei Zugriffen auf Texturen ausserhalb des Texturbereichs einen definierten Randwert zurueckzugeben. Standardmaessig ist dieser als 0,0,0 definiert, was exakt der Dirichlet-Boundary-Condition entspraecht. Leider sind in OpenCL 1.1 Texturen als read- oder write-only definiert wodurch sie nicht fuer die Implementierung der FDTD genutzt werden koennen, da lesend und schreibend auf die Feldstaerkewerte zugegriffen wird. Ich habe mich daher bei der sequentiellen sowie der parallelen Implementierung dafuer entschieden den Simulationsraum kuenstlich zu verkleinern.

### 5.3 Referenzimplementierung in C

Wie erwaeht wird die Referenzimplementierung in C genutzt um die hardwarebeschleunigten Implementierungen auf Korrektheit zu ueberpruefen und die Performance zu vergleichen.

```
int main()
{
    //Dummy Code
    return 0;
}
```

### 5.4 Implementierung fuer einen Zeitschritt

Die erste, leichtere und wie sich herausstellen soll weniger performante Implementierungsmoeglichkeit lagert nur die Berrechnung eines Zeitschrittes auf die GPU aus. Fuer jeden Zeitschritt werden die Daten des E- und H-Feldes an die GPU gesendet, berechnet und das Ergebnis gelesen. Danach kann hostseitig eine oder mehrere Quellen injiziert werden und ein neuer

Zeitschritt beginnen. Das hat den Vorteil, dass die Quelle flexibler gestaltet werden kann und dass das Feld nach jedem Zeitschritt bekannt ist. Eine Visualisierung in einem Video waere damit moeglich. Die optimale Groesze des lokalen Problembereichs kann von OpenCL bestimmt werden und da keine Synchronisation noetig ist koennen alle Compute Units verwendet werden. Nachteilig ist, dass Daten von und auf die GPU kopiert werden muessen was im Vergleich zu Rechenoperationen ein langsamer Prozess ist. Der Kernel-code ist hier nur fuer Flieszkommazahlen einfacher Genauigkeit angegeben:

```
kernel void fdtd2d_noiter(global float* ez_old ,
                          global float* hx_old ,
                          global float* hy_old ,
                          global float* ez_current ,
                          global float* hx_current ,
                          global float* hy_current ,
                          global float* ez_integral ,
                          global float* hx_integral ,
                          global float* hy_integral ,
                          constant float* ez_factor1 ,
                          constant float* ez_factor2 ,
                          constant float* ez_factor3 ,
                          constant float* hx_factor1 ,
                          constant float* hx_factor2 ,
                          constant float* hx_factor3 ,
                          constant float* hy_factor1 ,
                          constant float* hy_factor2 ,
                          constant float* hy_factor3 ,
                          float grid_width_x ,
                          float grid_width_y ,
                          int uwidth)
{
    // Get the actual id
    const int row = uwidth;
    const int id = get_global_id(0) +
        get_global_id(1) * row;

    /*

    // Calculate the curl terms to simplify the
    actual calculation
```

```

const float hx_curl_term = ((ez_old[id + row]
    - ez_old[id]) / grid_width_y);
const float hy_curl_term = ((ez_old[id + 1] -
    ez_old[id]) / grid_width_x);

// Euler integration of the curl terms
hx_integral[id] = hx_curl_term;
hy_integral[id] = hy_curl_term;

// Update the magnetic field
hx_current[id] = hx_factor1[id] * hx_old[id]
    - hx_factor2[id] *
        hx_curl_term
    - hx_factor3[id] *
        hx_integral[id];
hy_current[id] = hy_factor1[id] * hy_old[id];
    + hy_factor2[id] *
        hy_curl_term
    + hy_factor3[id] *
        hy_integral[id];

// Calculate the curl term to simplify the
    actual calculation
const float ez_curl_term = ((hy_current[id] -
    hy_current[id-1]) / grid_width_x)
    - ((hx_current[id] - hx_current[id -
    row]) / grid_width_y);
// Euler integration of the curl term
ez_integral[id] += ez_old[id];
// Calculate the electric field
ez_current[id] = ez_factor1[id] * ez_old[id]
    - ez_factor2[id] *
        ez_integral[id]
    + ez_factor3[id] *
        ez_curl_term;

*/
ez_current[id] = 4;
}

```

## 5.5 Implementierung fuer mehrere Zeitschritte

Sollen gleich mehrere Zeitschritte auf der GPU verarbeitet werden muessen die Kernel synchronisiert werden um sicherzustellen, dass die Felder fuer einen Zeitschritt schon vollstaendig berechnet wurden bevor die naechste Iteration beginnt. Da eine Synchronisation nur zwischen Kerneln einer Workgroup moeglich ist darf nur eine Workgroup existieren. Das bedeutet, dass der lokaler Problembereich und der globale Problembereich gleichgrosz sein muessen. Die mir zu Verfuegung stehende GPU hatte eine maximale Workgroupsize von 256. Demzufolge koennen nur 256 Kernel verwendet werden. Ein dreidimensionaler Problembereich kann daher nur  $256^{(1/3)} = 6.349 \approx 6 \Rightarrow 6^3$  grosz sein. Soll nun ein physikalischer Problembereich von mehr als  $6^3$  Zellen berechnet werden muss ein Kernel die Berechnung fuer mehrere Zellen durchfuehren. Das fuehrt zu folgendem Code, der auch hier nur fuer Flieszkommazahlen einfacher Genauigkeit angegeben ist.

```
kernel void fdtd2d_noiter(global float* ez_old ,
                          global float* hx_old ,
                          global float* hy_old ,
                          global float* ez_current ,
                          global float* hx_current ,
                          global float* hy_current ,
                          global float* ez_integral ,
                          global float* hx_integral ,
                          global float* hy_integral ,
                          constant float* ez_factor1 ,
                          constant float* ez_factor2 ,
                          constant float* ez_factor3 ,
                          constant float* hx_factor1 ,
                          constant float* hx_factor2 ,
                          constant float* hx_factor3 ,
                          constant float* hy_factor1 ,
                          constant float* hy_factor2 ,
                          constant float* hy_factor3 ,
                          float grid_width_x ,
                          float grid_width_y ,
                          int uwidth)
{
    // Get the actual id
    const int row = uwidth;
```

```

const int id = get_global_id(0) +
    get_global_id(1) * row;

/*

// Calculate the curl terms to simplify the
    actual calculation
const float hx_curl_term = ((ez_old[id + row]
    - ez_old[id]) / grid_width_y);
const float hy_curl_term = ((ez_old[id + 1] -
    ez_old[id]) / grid_width_x);

// Euler integration of the curl terms
hx_integral[id] = hx_curl_term;
hy_integral[id] = hy_curl_term;

// Update the magnetic field
hx_current[id] = hx_factor1[id] * hx_old[id]
    - hx_factor2[id] *
        hx_curl_term
    - hx_factor3[id] *
        hx_integral[id];
hy_current[id] = hy_factor1[id] * hy_old[id];
    + hy_factor2[id] *
        hy_curl_term
    + hy_factor3[id] *
        hy_integral[id];

// Calculate the curl term to simplify the
    actual calculation
const float ez_curl_term = ((hy_current[id] -
    hy_current[id-1]) / grid_width_x)
    - ((hx_current[id] - hx_current[id -
    row]) / grid_width_y);
// Euler integration of the curl term
ez_integral[id] += ez_old[id];
// Calculate the electric field
ez_current[id] = ez_factor1[id] * ez_old[id]
    - ez_factor2[id] *
        ez_integral[id]

```

```

+ ez_factor3[id] *
  ez_curl_term;

*/
ez_current[id] = 4;
}

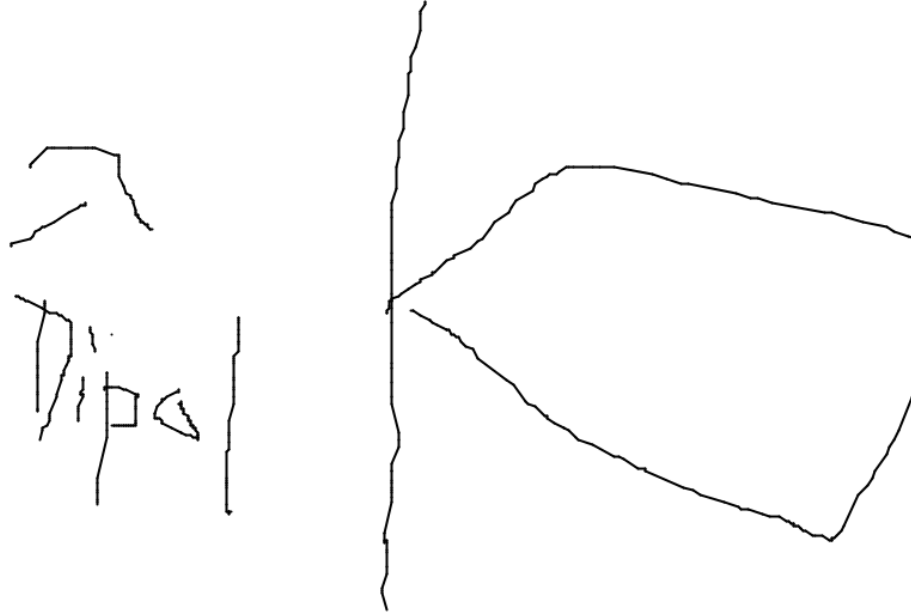
```

Die Quelle wurde durch einen Quellenfaktorarray implementiert das jeder Zelle einen Quellenfaktor implementiert. Dadurch ist es moeglich ohne die Verwendung einer if-Abfrage die elektrischen Feldstaerkewerte von Zellen mit einer Quelle auf null zu setzen um dann den aktuellen Wert des Quellsignals aufzuaddieren. Dadurch wird die Quelle als Hard Source implementiert. Kontrollstrukturen wie if-Abfragen werden in optimiertem Code vermieden, da sie im Vergleich zu mathematischen Operationen langsam sind.

## 6 Benchmarking

## 7 Simulation einer $\lambda$ -Dipolantenne

Eine  $\lambda$ -Dipolantenne ist ein Dipolantenne mit einer Drahtlaenge gleich der Wellenlaenge der Sende- und Empfangsfrequenz. Bei einer symmetrischen Einspeisung weist sie eine keulenfoermige Richtcharakteristik auf.



Source: [Krischke(2013)]

Figure 1: Richtcharakteristik eine  $\lambda$ -Dipolantenne bei symmetrischer Einspeisung

## 8 Zusammenfassung und Ausblicke

### References

- [Globus et al.()] Globus, Linden, and John] Al Globus,  
Derek S. Linden, and Jason D. John. Auto-  
mated antenna design with evolutionary algorithms.  
"http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.102.9841&rep=rep1&type=pdf".
- [R. Kourant(1928)] H. Lewy R. Kourant, K. Friedrichs. Über die partiellen  
differenzengleichungen der mathematischen physik, 1928.
- [Hagness(2005)] Hagness. *Computational Electrodynamics: The Finite-  
Difference Time-Domain Method*, 3rd ed. Artech House, 2005.

- [Steunfeld()] Thomas Steunfeld. Liste numerischer verfahren.  
["https://mathepedia.de/Numerischeverfahren.html"](https://mathepedia.de/Numerischeverfahren.html).
- [Luebbers and H.S()] R.J Luebbers and Langdon H.S. A simple feed model that reduces time steps needed for fdtd antenna and microstrip calculations.  
["http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.461.3208&rep=rep1&type=pdf"](http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.461.3208&rep=rep1&type=pdf).
- [Rumpf(a)] Prof. Raymond Rumpf. Lecture 4 – electromagnetics and fdtd. ["http://emlab.utep.edu/ee5390fdtd/Lecture 4 – Electromagnetics and FDTD.pdf"](http://emlab.utep.edu/ee5390fdtd/Lecture%204%E2%80%93Electromagnetics%20and%20FDTD.pdf), a. Seite 3.
- [Rumpf(b)] Prof. Raymond Rumpf. Lecture 4 – electromagnetics and fdtd. ["http://emlab.utep.edu/ee5390fdtd/Lecture 4 – Electromagnetics and FDTD.pdf"](http://emlab.utep.edu/ee5390fdtd/Lecture%204%E2%80%93Electromagnetics%20and%20FDTD.pdf), b. Seite 4.
- [Rumpf(c)] Prof. Raymond Rumpf. Lecture 4 – electromagnetics and fdtd. ["http://emlab.utep.edu/ee5390fdtd/Lecture 4 – Electromagnetics and FDTD.pdf"](http://emlab.utep.edu/ee5390fdtd/Lecture%204%E2%80%93Electromagnetics%20and%20FDTD.pdf), c. Seite 4.
- [Rumpf(d)] Prof. Raymond Rumpf. Lecture 4 – electromagnetics and fdtd. ["http://emlab.utep.edu/ee5390fdtd/Lecture 4 – Electromagnetics and FDTD.pdf"](http://emlab.utep.edu/ee5390fdtd/Lecture%204%E2%80%93Electromagnetics%20and%20FDTD.pdf), d. Seite 5.
- [Rumpf(e)] Prof. Raymond Rumpf. Lecture 4 – electromagnetics and fdtd. ["http://emlab.utep.edu/ee5390fdtd/Lecture 4 – Electromagnetics and FDTD.pdf"](http://emlab.utep.edu/ee5390fdtd/Lecture%204%E2%80%93Electromagnetics%20and%20FDTD.pdf), e. Seite 9.
- [Rumpf(f)] Prof. Raymond Rumpf. Lecture 5 – formulation of 1d fdtd. ["http://emlab.utep.edu/ee5390fdtd/Lecture 5 – Formulation of 1D FDTD.pdf"](http://emlab.utep.edu/ee5390fdtd/Lecture%205%E2%80%93Formulation%20of%201D%20FDTD.pdf), f. Seite 11.
- [Rumpf(g)] Prof. Raymond Rumpf. Lecture 4 – electromagnetics and fdtd. ["http://emlab.utep.edu/ee5390fdtd/Lecture 4 – Electromagnetics and FDTD.pdf"](http://emlab.utep.edu/ee5390fdtd/Lecture%204%E2%80%93Electromagnetics%20and%20FDTD.pdf), g. Seite 24.
- [Rumpf(h)] Prof. Raymond Rumpf. Lecture 4 – electromagnetics and fdtd. ["http://emlab.utep.edu/ee5390fdtd/Lecture 4 – Electromagnetics and FDTD.pdf"](http://emlab.utep.edu/ee5390fdtd/Lecture%204%E2%80%93Electromagnetics%20and%20FDTD.pdf), h. Seite 26.
- [Rumpf(i)] Prof. Raymond Rumpf. Lecture 5 – formulation of 1d fdtd. ["http://emlab.utep.edu/ee5390fdtd/Lecture 5 – Formulation of 1D FDTD.pdf"](http://emlab.utep.edu/ee5390fdtd/Lecture%205%E2%80%93Formulation%20of%201D%20FDTD.pdf), i. Seite 13.



- [Yee(1966)] K. Yee. Numerical solution of initial boundary value problems involving maxwell's equations in isotropic media. *IEEE Transactions on Antennas and Propagation*, 14:302–307, May 1966. doi: 10.1109/TAP.1966.1138693.
- [Rumpf(j)] Prof. Raymond Rumpf. Topic 2b building geometries into data arrays. "http://emlab.utep.edu/ee43865301cmee/Topic2b – BuildingGeometriesintoDataArrays.pdf", j. Seite 7.
- [Rumpf(k)] Prof. Raymond Rumpf. Topic 2b building geometries into data arrays. "http://emlab.utep.edu/ee5390fdtd/Lecture 6 – Implementation0lf 1D FDTD.pdf", k. Seite 13.
- [Rumpf(l)] Prof. Raymond Rumpf. Topic 2b building geometries into data arrays. "http://emlab.utep.edu/ee5390fdtd/Lecture 6 – Implementation0lf 1D FDTD.pdf", l. Seite 18.
- [Rumpf(m)] Prof. Raymond Rumpf. Lecture 6 – implementation of 1d fdtd.pdf. "http://emlab.utep.edu/ee5390fdtd/Lecture 6 – Implementation of 1D FDTD.pdf", m. Seite 25.
- [Rumpf(n)] Prof. Raymond Rumpf. Implementation of 1d fdtd. "http://emlab.utep.edu/ee5390fdtd/Lecture 6 – Implementation of 1D FDTD.pdf", n. Youtube Video.
- [Rumpf(o)] Prof. Raymond Rumpf. Implementation of one dimensional fdtd. "http://emlab.utep.edu/ee5390fdtd/Lecture 6 – Implementation of 1D FDTD.pdf", o. Seite 10.
- [Schneider()] John Schneider. Chapter 11 perfectly matched layer. "https://www.eecs.wsu.edu/~schneidj/ufdtd/chap11.pdf". Seite 307.
- [Rumpf(p)] Prof. Raymond Rumpf. The perfectly matched layer (pml). http://emlab.utep.edu/ee5390fdtd/Lecture 13 – The Perfectly Matched Layer.pdf, p.
- [Rumpf(q)] Prof. Raymond Rumpf. The perfectly matched layer (pml). "http://emlab.utep.edu/ee5390fdtd/Lecture 13 – The Perfectly Matched Layer.pdf", q. Seite 9.
- [Rumpf(r)] Prof. Raymond Rumpf. The perfectly matched layer (pml). "http://emlab.utep.edu/ee5390fdtd/Lecture 13 – The Perfectly Matched Layer.pdf", r. Seite 17.

- [Group()] Khronos Group. Opencl overview.  
["https://www.khronos.org/opencl/"](https://www.khronos.org/opencl/).
- [Munshi et al.(2011)] Munshi, Gaster, Mattson, Fung, and Ginsburg] Aaftab  
Munshi, Benedic R. Gaster, Timothy G. Mattson, James Fung, and Dan  
Ginsburg. *OpenCL Programming Guide*. Pearson Education, 2011.
- [Krischke(2013)] Alois Krischke. *Rothammels Antennenbuch*. DARC Verlag,  
2013.