

Алгоритм FDTD

СМВТФ

Егоров Н.Н.

Введение в метод FDTD

- FDTD - "finite-difference time-domain",
- в русскоязычной литературе КРВО - "конечные разности во временной области"
- метод FDTD - один из многочисленных методов решения дифференциальных уравнений, но среди тех, кто занимается решением задач электродинамики, FDTD является синонимом решения вихревых дифференциальных уравнений Максвелла.

- В 1966 г. Йе (Yee) разработал технику, реализующую явную конечно - разностную схему второго порядка для решения вихревых уравнений Максвелла в пространстве и времени.

K. S. Yee, "Numerical Solution of Initial Boundary Value Problems Involving Maxwell's Equations in Isotropic Media," IEEE Transactions on Antennas and Propagation, vol. 14, pp. 302-307, May 1966.

Исходными являются уравнения
Максвелла в дифференциальной
форме

$$\text{rot}(\mathbf{H}) = \partial \mathbf{D} / \partial t + \mathbf{J}; \quad \text{div}(\mathbf{B}) = 0; \quad (1)$$

$$\text{rot}(\mathbf{E}) = - \partial \mathbf{B} / \partial t; \quad \text{div}(\mathbf{D}) = \rho;$$

$$\mathbf{D} = \varepsilon \varepsilon_0 \mathbf{E};$$

$$\mathbf{J} = \sigma \mathbf{E}; \quad (2)$$

$$\mathbf{B} = \mu \mu_0 \mathbf{H}$$

Два уравнения (1) содержат
пространственные и временные
производные.

Для решения уравнения (1) следует выразить в декартовых координатах векторы **E** и **H**:

$$\mathbf{E} = E_x(t,x,y,z)\mathbf{X} + E_y(t,x,y,z)\mathbf{Y} + E_z(t,x,y,z)\mathbf{Z}; \quad (3)$$

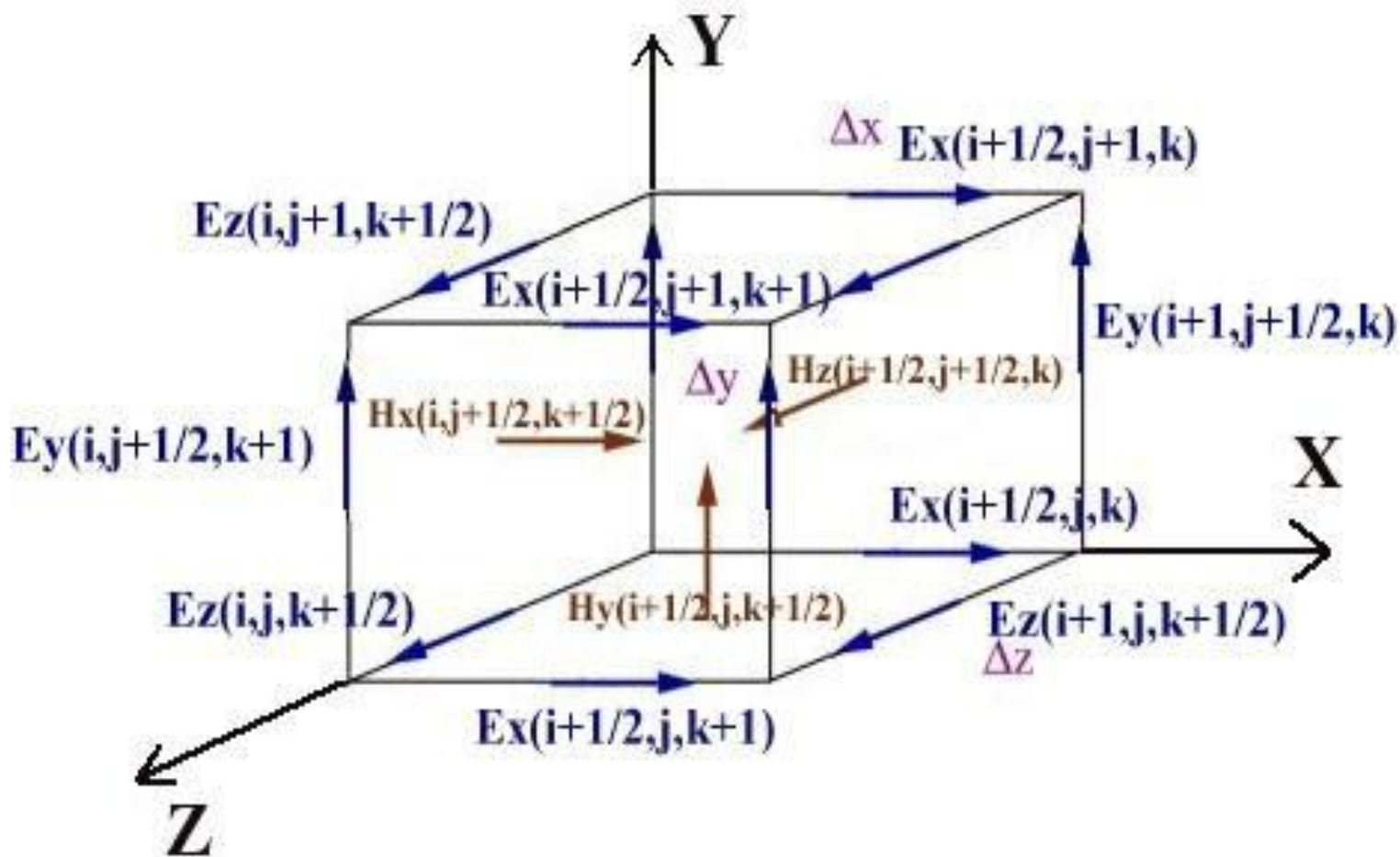
$$\mathbf{H} = H_x(t,x,y,z)\mathbf{X} + H_y(t,x,y,z)\mathbf{Y} + H_z(t,x,y,z)\mathbf{Z};$$

где $E_x, E_y, E_z, H_x, H_y, H_z$ - проекции векторов на координатные оси, а **X**, **Y**, **Z** - единичные векторы.

Остальные величины в (1) - **D**, **B**, **J** - выразим через **E** и **H**. Величины **E** и **H** для нас будут основными.

Примечание: существуют и другие подходы, когда в уравнениях (1) вначале оставляют **D** и/или **B**, но в конце концов всё равно выражаются вектора **E** и **H**. Также следует указать, что уравнения (1) записаны не полностью. Например, в них не учитываются сторонние токи.

Yee (1966) предложил пространственную сетку для конечно-разностной аппроксимации, в которую поместил вектора $E_x, E_y, E_z, H_x, H_y, H_z$. Фрагмент сетки Yee показан на (рис.1).



Все компоненты ($E_x, E_y, E_z, H_x, H_y, H_z$) находятся в разных местах, т.е. разнесены в пространстве.

- E-компоненты находятся посередине ребер,
- H - компоненты - по центру граней.

Все компоненты независимы друг от друга, т.е. каждой из них можно присвоить свои уникальные электрические (для E) и магнитные (для H) параметры.

Пространственные координаты каждого вектора x , y и z выражаются в номерах ячеек i , j и k соответственно, время t выражается в шагах n по времени:

$$x = i\Delta x; \quad y = j\Delta y; \quad z = k\Delta z; \quad t = n\Delta t; \quad (4)$$

где $\Delta x, \Delta y, \Delta z$ - размеры пространственной ячейки, Δt - шаг по времени.

Поля E и H вычисляются со сдвигом на полшага по времени. Обозначения, введенные Yee, следующие:

- E^n - значение поля E на только что вычисленном шаге;
- E^{n+1} - значение поля E на вычисляемом сейчас шаге по времени.
- $H^{n-1/2}$ - значение поля H на только что вычисленном шаге;
- $H^{n+1/2}$ - значение поля на вычисляемом сейчас полушаге по времени.

Из этих обозначений следует, что процедура вычислений начинается с поля $H^{n+1/2}$, потому что в момент $t=0$ ($n=0$) установлены начальные условия по всему счетному объему: **все значения полей E и H равны нулю**. Хотя в принципе это лишь наиболее распространенная условность. Можно считать, что пространственная сетка проходит через вектора H , что процедура счета начинается с поля E .

Теперь, когда введены основные обозначения, покажем вывод выражений, пригодных для расчетов с помощью компьютера и которым уже 50 лет.

Поставим (3) и (2) в (1). Получим:

$$\text{rot}(\mathbf{H}) \mathbf{X} = \varepsilon \varepsilon_0 \partial E_x / \partial t + \sigma E_x; \quad (5)$$

$$\text{rot}(\mathbf{E}) \mathbf{Y} = -\mu \mu_0 \partial H_y / \partial t;$$

Применяя конечно-разностную аппроксимацию, преобразуем (5) в выражения для шагов n и n+1, учитывая (4), получим:

$$\sigma E_x^{n+1/2} \approx \sigma_{(i+1/2,j,k)} (E_x^n_{(i+1/2,j,k)} + E_x^{n+1}_{(i+1/2,j,k)})/2;$$

$$\varepsilon \varepsilon_0 \partial E_x^{n+1/2} / \partial t \approx \varepsilon_{(i+1/2,j,k)} \varepsilon_0 (E_x^{n+1}_{(i+1/2,j,k)} - E_x^n_{(i+1/2,j,k)})/\Delta t;$$

$$\mu \mu_0 \partial H_y^n / \partial t \approx \mu_{(i+1/2,j,k+1/2)} \mu_0 (H_y^{n+1/2}_{(i+1/2,j,k+1/2)} - H_y^{n-1/2}_{(i+1/2,j,k+1/2)})/\Delta t; \quad (6)$$

$$\text{rot}(\mathbf{H}^{n+1/2}) \mathbf{X} \approx (H_z^{n+1/2}_{(i+1/2,j+1/2,k)} - H_z^{n+1/2}_{(i+1/2,j-1/2,k)})/\Delta y - (H_y^{n+1/2}_{(i+1/2,j,k+1/2)} -$$

$$H_y^{n+1/2}_{(i+1/2,j,k-1/2)})/\Delta z;$$

$$\text{rot}(\mathbf{E}^n) \mathbf{Y} \approx (E_x^n_{(i+1/2,j,k+1)} - E_x^n_{(i+1/2,j,k)})/\Delta z - (E_z^n_{(i+1,j,k+1/2)} - E_z^n_{(i,j,k+1/2)})/\Delta x;$$

Подставляя (6) в (5) и решая получившиеся выражения относительно $H_y^{n+1/2}(i+1/2,j,k+1/2)$ и $E_x^{n+1}(i+1/2,j,k)$ получим:

$$H_y^{n+1/2}(i+1/2,j,k+1/2) = H_y^{n-1/2}(i+1/2,j,k+1/2) + C_{Hy(i+1/2,j,k+1/2)} ((E_z^n(i+1,j,k+1/2) - E_z^n(i,j,k+1/2))/\Delta x - (E_x^n(i+1/2,j,k+1) - E_x^n(i+1/2,j,k))/\Delta z);$$

$$C_{Hy(i+1/2,j,k+1/2)} = \Delta t / (\mu_{(i+1/2,j,k+1/2)} \mu_o); \quad (7)$$

$$E_x^{n+1}(i+1/2,j,k) = C_{1Ex(i+1/2,j,k)} E_x^n(i+1/2,j,k) + C_{2Ex(i+1/2,j,k)} (H_z^{n+1/2}(i+1/2,j+1/2,k) - H_z^{n+1/2}(i+1/2,j-1/2,k))/\Delta y - (H_y^{n+1/2}(i+1/2,j,k+1/2) - H_y^{n+1/2}(i+1/2,j,k-1/2))/\Delta z); \quad (8)$$

$$C_{1Ex(i+1/2,j,k)} = (\epsilon_{(i+1/2,j,k)} \epsilon_o - 0,5\sigma_{(i+1/2,j,k)} \Delta t) / (\epsilon_{(i+1/2,j,k)} \epsilon_o + 0,5\sigma_{(i+1/2,j,k)} \Delta t);$$

$$C_{2Ex(i+1/2,j,k)} = \Delta t / (\epsilon_{(i+1/2,j,k)} \epsilon_o + 0,5\sigma_{(i+1/2,j,k)} \Delta t).$$

Аналогичные выражения можно получить для остальных четырех компонент ячейки Yee.

Из выражений (7) и (8) видно, что значения μ , ϵ и σ задаются для каждого их векторов ячейки и могут быть различными в разных направлениях. Т.е. при необходимости можно задать анизотропию материалов для E и/или H полей.

Выражения (7) и (8) являются достаточными для многих решаемых задач, но для расчетов сосредоточенных элементов (источников напряжения, индуктивностей, транзисторов и т.п.), а также для расчетов материалов с нелинейными свойствами требуется их модификация.

Следует упомянуть, что явные конечно-разностные схемы требуют специальных условий для устойчивой работы. Для метода FDTD это условие имеет вид:

$$\Delta t \leq \frac{1}{v \sqrt{\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} + \frac{1}{\Delta z^2}}},$$

где v - максимальная скорость электромагнитных волн в счетном объеме.

Обычно $v = c$ (скорости света в вакууме).

Базовый алгоритм FDTD

Выше кратко описан вывод конечно-разностных уравнений в декартовых координатах для двух компонент сетки Yee из шести.

На рис. 2. приводится полная система для всех векторов сетки Yee

$$H_{x(i,j,k)}^{n+\frac{1}{2}} = H_{x(i,j,k)}^{n-\frac{1}{2}} - \frac{\Delta t}{\mu} \left[\frac{E_z^n(i,j+1,k) - E_z^n(i,j,k)}{\Delta y} - \frac{E_y^n(i,j,k+1) - E_y^n(i,j,k)}{\Delta z} \right]$$

$$H_{y(i,j,k)}^{n+\frac{1}{2}} = H_{y(i,j,k)}^{n-\frac{1}{2}} - \frac{\Delta t}{\mu} \left[\frac{E_x^n(i,j,k+1) - E_x^n(i,j,k)}{\Delta z} - \frac{E_z^n(i+1,j,k) - E_z^n(i,j,k)}{\Delta x} \right]$$

$$H_{z(i,j,k)}^{n+\frac{1}{2}} = H_{z(i,j,k)}^{n-\frac{1}{2}} - \frac{\Delta t}{\mu} \left[\frac{E_y^n(i+1,j,k) - E_y^n(i,j,k)}{\Delta x} - \frac{E_x^n(i,j+1,k) - E_x^n(i,j,k)}{\Delta y} \right]$$

$$E_{x(i,j,k)}^{n+1} = \left(\frac{1 - \frac{\sigma \Delta t}{2\varepsilon}}{1 + \frac{\sigma \Delta t}{2\varepsilon}} \right) E_{x(i,j,k)}^n + \frac{\Delta t}{\varepsilon} \left[\frac{H_{z(i,j,k)}^{n+\frac{1}{2}} - H_{z(i,j-1,k)}^{n+\frac{1}{2}}}{\Delta y} - \frac{H_{y(i,j,k)}^{n+\frac{1}{2}} - H_{y(i,j,k-1)}^{n+\frac{1}{2}}}{\Delta z} \right]$$

$$E_{y(i,j,k)}^{n+1} = \left(\frac{1 - \frac{\sigma \Delta t}{2\varepsilon}}{1 + \frac{\sigma \Delta t}{2\varepsilon}} \right) E_{y(i,j,k)}^n + \frac{\Delta t}{\varepsilon} \left[\frac{H_{x(i,j,k)}^{n+\frac{1}{2}} - H_{x(i,j,k-1)}^{n+\frac{1}{2}}}{\Delta z} - \frac{H_{z(i,j,k)}^{n+\frac{1}{2}} - H_{z(i-1,j,k)}^{n+\frac{1}{2}}}{\Delta x} \right]$$

$$E_{z(i,j,k)}^{n+1} = \left(\frac{1 - \frac{\sigma \Delta t}{2\varepsilon}}{1 + \frac{\sigma \Delta t}{2\varepsilon}} \right) E_{z(i,j,k)}^n + \frac{\Delta t}{\varepsilon} \left[\frac{H_{y(i,j,k)}^{n+\frac{1}{2}} - H_{y(i-1,j,k)}^{n+\frac{1}{2}}}{\Delta x} - \frac{H_{x(i,j,k)}^{n+\frac{1}{2}} - H_{x(i,j-1,k)}^{n+\frac{1}{2}}}{\Delta y} \right]$$

Примечания.

1. Полуцелые индексы, которые применены в нотации Y_{ee} , часто заменяются на целые путем уменьшения на пол-индекса, например, $(j+1/2)$ заменено на (j) , а $(j-1/2)$ заменено на $(j-1)$. Это сделано для удобства программирования, т.к. не бывает полуцелых индексов массивов.
2. Переменные ε , σ и μ повсюду должны быть представлены как $\varepsilon(i,j,k)$ $\sigma(i,j,k)$ $\mu(i,j,k)$, но на рисунке индексы (i,j,k) опущены для экономии места.
3. ε и μ здесь - это абсолютные проницаемости, т.е. сразу готовое произведение $\varepsilon\varepsilon_o$ и $\mu\mu_o$, как это представлено ранее (см. введение). Путаница в этих обозначениях - широко распространенное явление в литературе, поэтому надо держать ухо востро.
4. Коэффициенты C_1 и C_2 из (8), на первый взгляд, не такие, как на этом рисунке. На самом деле это то же самое. Попробуйте разделить числитель и знаменатель коэффициентов C_1 и C_2 (8) на $(\varepsilon_{(i+1/2,j,k)}\varepsilon_o)$.

Шесть уравнений базового алгоритма очень просты в реализации. Для этого необходимо создать шесть массивов E_x , E_y , E_z , H_x , H_y и H_z . А также массивы электрофизических характеристик ϵ , σ , μ , которые назовем Eps , Sig и Mu . Этих массивов три штуки, если электрофизические характеристики задаются для ячейки Yee в целом. Здесь возможны разные варианты. Если ϵ , σ и μ планируется задавать для каждого вектора свои, то потребуются массивы Eps_x , Eps_y , Eps_z , Sig_x и т.д. Если не планируется применение магнитных материалов, то массив Mu можно вообще не создавать. Если планируется проводить расчеты только для идеальных проводников (ПЕС) в свободном пространстве, то массивы электрофизических констант не нужны вовсе. В этом случае векторы E , которые принадлежат ПЕС-объекту, попросту обнуляются, а все остальные считаются принадлежащими свободному пространству.

Чем больше возможностей мы хотим реализовать в базовом алгоритме, тем больший объем памяти требуется для переменных и, соответственно, больше затраты времени при расчетах.

Раньше значительное ускорение вычислений можно было получить создав еще ряд массивов и поместив в них предварительно вычисленные коэффициенты C , C_1 и C_2 из уравнений (7) и (8). Но сейчас в гигагерцовых процессорах скорость вычислений арифметики с плавающей точкой выросла больше, чем скорость выборки дополнительных элементов массива из памяти, поэтому смысл в предварительном вычислении этих коэффициентов снизился.

В программе ZFDTD пожертвовали скоростью во имя экономии памяти, хотя в ранних версиях программы было наоборот.

Вычисление полей E и H ведется рекурсивно: для вычисления значений на текущем шаге по времени используются значения на предыдущем. При этом результат сразу помещается в прежний массив "затирая" старые значения. Но в некоторых случаях все-таки требуется иметь значения полей на предыдущем шаге по времени одновременно с текущим шагом. Например, для того, чтобы вычислить значение поля E на полушаге по времени как среднее арифметическое значений на двух соседних шагах. В этом случае волей-неволей приходится использовать еще несколько массивов. При вычислении энергии и других величин, производных от полей E и H , также потребуются дополнительные массивы для хранения результатов. В общем, по отношению к оперативной памяти алгоритм FDTD довольно прожорлив, но для больших задач этот алгоритм, по литературным данным, требует меньше памяти, чем интегральные методы.

Ход вычислений рассмотрим на примере, написанном на языке Pascal

Две процедуры вычисляют поля E и H:

// ОПРЕДЕЛЕНИЕ МАГНИТНОГО ПОЛЯ H

Procedure **TimeStepForH**;

var I,J,K: Integer;

begin

For I:=1 to XSize do

for J:=1 to YSize-1 do

for K:=1 to ZSize-1 do

HX[I,J,K]:=HX[I,J,K]-((EZ[I,J+1,K]-EZ[I,J,K])*TY-(EY[I,J,K+1]-EY[I,J,K])*TZ)*AH[I,J,K];

For I:=1 to XSize-1 do

for J:=1 to YSize do

for K:=1 to ZSize-1 do

HY[I,J,K]:=HY[I,J,K]-((EX[I,J,K+1]-EX[I,J,K])*TZ-(EZ[I+1,J,K]-EZ[I,J,K])*TX)*AH[I,J,K];

For I:=1 to XSize-1 do

for J:=1 to YSize-1 do

for K:=1 to ZSize do

HZ[I,J,K]:=HZ[I,J,K]-((EY[I+1,J,K]-EY[I,J,K])*TX-(EX[I,J+1,K]-EX[I,J,K])*TY)*AH[I,J,K];

end;

// КОНЕЦ ОПРЕДЕЛЕНИЯ МАГНИТНОГО ПОЛЯ H

// ОПРЕДЕЛЕНИЕ ЭЛЕКТРИЧЕСКОГО ПОЛЯ E

Procedure **TimeStepForE**;

var I,J,K: Integer;

begin

for I:=1 to XSize-1 do for J:=2 to YSize-1 do for K:=2 to ZSize-1 do

$EX[I,J,K] := EX[I,J,K] * AE[I,J,K] + ((HZ[I,J,K] - HZ[I,J-1,K]) * TY - (HY[I,J,K] - HY[I,J,K-1]) * TZ) * BE[I,J,K];$

for I:=2 to XSize-1 do for J:=1 to YSize-1 do for K:=2 to ZSize-1 do
 $EY[I,J,K] := EY[I,J,K] * AE[I,J,K] + ((HX[I,J,K] - HX[I,J,K-1]) * TZ - (HZ[I,J,K] - HZ[I-1,J,K]) * TX) * BE[I,J,K]$

for I:=2 to XSize-1 do for J:=2 to YSize-1 do for K:=1 to ZSize-1 do
 $EZ[I,J,K] := EZ[I,J,K] * AE[I,J,K] + ((HY[I,J,K] - HY[I-1,J,K]) * TX - (HX[I,J,K] - HX[I,J-1,K]) * TY) * BE[I,J,K];$

end;

// КОНЕЦ ОПРЕДЕЛЕНИЯ ЭЛЕКТРИЧЕСКОГО ПОЛЯ E

Массивы $AE[I,J,K]$, $BE[I,J,K]$ и $AN[I,J,K]$ - это предварительно вычисленные коэффициенты C_1 , C_2 и C соответственно из [\(7\)](#) и [\(8\)](#). $XSize$, $YSize$ и $ZSize$ - размеры счетного пространства. А коэффициенты TX , TY и TZ равны $1/\Delta X$, $1/\Delta Y$ и $1/\Delta Z$ соответственно. Как видно, здесь применены предварительно вычисленные массивы коэффициентов $C1$, $C2$ и C . Приведенный код вычисления полей рассчитан на такое представление объектов, когда считается, что электрофизические характеристики пространства в ячейке (i,j,k) задаются одинаковыми для всех трех $H_{x,y,z}(i,j,k)$ и трех $E_{x,y,z}(i,j,k)$ векторов. При таком представлении точность вычислений на границах раздела сред снижается до первого порядка. Обычно это проявляется в снижении на 25..40% вычисленной резонансной частоты, такого же увеличения вычисленных токов в длинных структурах, снижении в разы скорости затухания свободных колебаний и в других ошибках.

Приведенные процедуры вызываются в цикле по

времени:

// ГЛАВНЫЙ ЦИКЛ ПО ВРЕМЕНИ

T:=0; //Исходное значение времени

for l:=0 to <заданное количество шагов по времени> do

begin

TimeStepForH; //Вычисление H по всему объему

BordersSymmetryH; //Выполнение граничных условий для симметрии по H

TimeStepForE; //Вычисление E по всему объему

BordersABC; // Граничные УСЛОВИЯ ПОГЛОЩЕНИЯ

BordersSymmetryE; //Выполнение граничных условий для симметрии по E

BordersPEC; //Выполнение граничных условий PEC

T:=T+dT; // инкремент времени на один шаг

end;

//ГЛАВНЫЙ ЦИКЛ ПО ВРЕМЕНИ

это самое простое место в программе

Граничные условия для FDTD

В счетном объеме каждый вектор E или H вычисляется через 4 соседних вектора. Так происходит по всему объему. Но на границах самые последние векторы E имеют: на гранях параллелепипеда счетного объема только три соседних вектора H из четырех необходимых; на ребрах - два. Поэтому точно вычислить поле E на границах невозможно.

Проблема вычисления граничных полей решается различными способами.

1. Условия PEC – (photo electrochemical cell) *идеальный проводник.*

Условия PEC таковы, что граничные вектора E никогда не вычисляются, а, значит, всегда равны нулю. Как известно, поле E всегда равно нулю в идеальном проводнике, поэтому такие границы ведут себя как идеальный проводник: электромагнитные волны 100 % отражаются обратно в счетный объем.

2. Условия симметрии.

В некоторых случаях поле E или поле H может быть симметричным относительно некоторой плоскости. Тогда в этой плоскости можно задать условие симметрии и тем самым вдвое уменьшить счетный объем. При этом рядом с данной плоскостью симметрии будет проходить граница счетного объема с условиями симметрии.

Симметрия может быть четной или нечетной.

При нечетной симметрии плоскость симметрии проходит внутри счетного объема параллельно грани на расстоянии пол-ячейки от грани. Условия нечетной симметрии для симметрии по E получаются простым переносом значений ближайших к границе векторов E на саму границу, а для симметрии по H получаются так же, но при этом вектор E меняет свой знак. Ниже приведены примеры нечетной симметрии для E и H .

Пример: нечетная симметрия по E:

```
// ПЛОСКОСТЬ  $Y=1/2$  симметрична по E
  For I:=1 to NX-1 do
For K:=1 to NZ do
  EX[I,1,K]:=EX[I,2,K];
  For I:=1 to NX do
    For K:=1 to NZ-1 do
EZ[I,1,K]:=EZ[I,2,K];
```

Пример: нечетная симметрия по H:

```
// ПЛОСКОСТЬ  $Y=1/2$  симметрична по H
  For I:=1 to NX-1 do
    For K:=1 to NZ do
EZ[I,1,K]:=-EZ[I,2,K];
  For I:=1 to NX do
    For K:=1 to NZ-1 do
EZ[I,1,K]:=-EZ[I,2,K];
```

Условия четной симметрии несколько сложнее. Плоскость симметрии проходит на расстоянии целой ячейки от границы, поэтому кроме поля Е на границе необходимо помнить о прилегающих к границе векторах Н. При этом симметрии по Е и Н отличаются друг от друга только знаками переносимых значений.

Пример: четная симметрия по Н:

// ПЛОСКОСТЬ Y=1 симметрична по Н, вычисление поля Е

begin

For K:=1 to NZ do For I:=1 to NX-1 do EX[I,1,K]:=-EX[I,3,K]; For K:=1 to NZ-1 do For I:=1 to NX do EZ[I,1,K]:=-EZ[I,3,K];
end;

// ПЛОСКОСТЬ Y=1 симметрична по Н, вычисление поля Н

begin

For K:=1 to NZ-1 do For I:=1 to NX do HX[I,1,K]:=HX[I,2,K];
For K:=1 to NZ do For I:=1 to NX-1 do HZ[I,1,K]:=HZ[I,2,K];
end;

Пример: четная симметрия по Е:

// ПЛОСКОСТЬ Y=1 симметрична по Е, вычисление поля Е

begin

For K:=1 to NZ do For I:=1 to NX-1 do EX[I,1,K]:=EX[I,3,K];
For K:=1 to NZ-1 do For I:=1 to NX do EZ[I,1,K]:=EZ[I,3,K];
end;

// ПЛОСКОСТЬ Y=1 симметрична по Е, вычисление поля Н

begin

For K:=1 to NZ-1 do For I:=1 to NX do HX[I,1,K]:=-HX[I,2,K];
For K:=1 to NZ do For I:=1 to NX-1 do HZ[I,1,K]:=-HZ[I,2,K];
end;

При четной симметрии граничные поля Е и Н устанавливаются не одновременно, а на соответствующих полушагах по времени.

Простые условия поглощения (АВС)

Для условий поглощения значения векторов электрического поля на границе вычисляются на основании известных полей в приграничных слоях. Причем берутся приграничные поля не только на текущем шаге по времени, но и на предыдущих шагах. Все эти условия пытаются спрогнозировать поле на границе.

В литературе встречается описание целого ряда простых условий поглощения разных авторов. Всего их наберется с десяток. Практически чаще всего применяют условия Мура (Mur) и Лиао (Liao). Остальные не применяют или используют очень редко из-за их более низкой эффективности (Trefethen-Halpern, Higdon) или неудобства в использовании (Retarded Time - RT), или из-за неприменимости в декартовых координатах (Bayliss-Turkel), или из-за тенденции к потере стабильности (относится почти ко всем условиям, но особо - к условиям, основанным на высших порядках точности конечных разностей).

Все условия имеют довольно низкий коэффициент отражения от границы, составляющий порядка 0,1..1 %, но только при падении волны на границу под прямым углом. При падении под острым углом коэффициент отражения растет вплоть до 100 % при падении по касательной. Из-за этого границы необходимо располагать как можно дальше от источника электромагнитных волн, чтобы волны приходили к границе под как можно большими углами, желательно по нормали к границе.

Следует отметить, что оценка эффективности тех или иных простых граничных условий различна у разных авторов. Например, один источник пишет, что условия Лиао на 20 dB эффективнее условий Мура 2-го порядка. Другой пишет, что условия Лиао на 12 dB хуже условий Мура 1-го порядка. Имеется ввиду коэффициент отражения. И оба доказывают свои выводы графиками сравнительных расчетов.

Истина, скорее всего, посередине: для каждой конкретной задачи имеются свои оптимальные граничные условия. Одно плохо - заранее никогда не знаешь, КАКИЕ лучше.

Примечание. Возможно, введение чисел двойной точности радикально повысит стабильность, но пока это непозволительная роскошь, расходующая в два раза больше памяти и времени.

Итак, граничных условий много. Здесь рассмотрим три варианта простейших граничных условий: Мура 1-го порядка, Лиао 3-го порядка и RT. В таблице S (большое) и s (малое) - разные переменные! $S = (dT * c) / D$ - число Куранта, где D - шаг по пространству, dT - шаг по времени, c - скорость света, а s - координата границы. $s-1$ - одна ячейка внутрь счетного объема, $s-2$ - две ячейки от границы и т.д. На поясняющих рисунках от границы ($surface$ - внешней поверхности счетного объема) по горизонтали отложена координата s , а по вертикали - время в шагах счета (n - временной индекс). Формула приведена в двух видах, дополняющих друг друга в пространстве и времени. На рисунках пустой кружок - вычисляемое значение, черные кружки - требуемые значения. Сразу можно определить, какие переменные приграничной области нужно хранить в дополнительных массивах и сколько шагов по времени они должны храниться.

Условия	Формула	Поясняющая
Mur 1-го порядка	$E^{n+1}(s) \cong E^n(s-1) + A_1(E^n(s) - E^{n+1}(s-1))$ $A_1 \equiv (1-S)/(1+S)$ $B(4,4) \approx B(3,3) + A_1 * (B(3,4) - B(4,3))$ <p>типично $S=0,5$ для минимального отражения под прямым углом, но может меняться для получения минимального</p>	
Liao 3-го порядка	$E^{n+1}(s) \cong E^{n-2}(s-3) + 3(E^n(s-1) - E^{n-1}(s-2))$ $B(4,4) \approx B(1,1) + 3 * (B(3,3) - B(2,2))$ <p>Для данного случая $S=0,5$.</p>	
RT-ABC	$E^{n+1}(s) \cong E^{n-1}(s-1) + .5(E^{n-1}(s) - E^{n-1}(s-2)) + (E^n(s-1) - E^{n-2}(s-1))$ $B(4,4) \approx B(2,3) + .5 * (B(2,4) - B(2,2)) + (B(3,3) - B(1,3))$	

Видно, что больше всего массивов переменных нужно хранить для условий RT (5), меньше всего - для условий Мура (1). Для условий Лиао нужно 3 дополнительных массива переменных.

А теперь пример реализации условий Мура 1-го порядка.

Procedure MursFirstABC;

```
var I,J,K: Integer;
begin
// УСЛОВИЯ ПОГЛОЩЕНИЯ. Массивы вида EZX1, EZXN и т.п. - массивы сохраненных на
предыдущем шаге по времени значений поля E в приграничной области. Процедура
сохранения приводится ниже.
// коэффициенты Tx, Ty и Tz равны  $c \cdot dT/dX$ ,  $c \cdot dT/dY$  и  $c \cdot dT/dZ$  соответственно. c-скорость
света, dT- шаг по времени; dX, dY, dZ - шаги по пространству.
//ЛЕВАЯ И ПРАВАЯ ПЛОСКОСТИ
// Ez для двух плоскостей YxZ (левой и правой)
for J:=2 to NY-1 do      for K:=1 to NZ-1 do begin
    EZ[1,J,K]:=EZX1[J,K]+(Tx-1)/(Tx+1)*(EZ[2,J,K]-EZ[1,J,K]);
    EZ[NX,J,K]:=EZXN[J,K]+(Tx-1)/(Tx+1)*(EZ[NX-1,J,K]-EZ[NX,J,K]);
end;
// Ey для двух плоскостей YxZ [левой и правой]
for J:=1 to NY-1 do      for K:=2 to NZ-1 do begin
    EY[1,J,K]:=EYX1[J,K]+(Tx-1)/(Tx+1)* (EY[2,J,K]-EY[1,J,K]);
    EY[NX,J,K]:=EYXN[J,K]+(Tx-1)/(Tx+1)* (EY[NX-1,J,K] -EY[NX,J,K]);
end;
```

//ВЕРХНЯЯ И НИЖНЯЯ ПЛОСКОСТИ

// Ex для двух плоскостей XxZ [верхней и нижней]

```
for K:=2 to NZ-1 do      for l:=1 to NX-1 do begin
    EX[l,1,K]:= EXY1[l,K]+(Ty-1)/(Ty+1)*(EX[l,2,K]-EX[l,1,K]);
    EX[l,NY,K]:= EXYN[l,K]+(Ty-1)/(Ty+1)*(EX[l,NY-1,K]-EX[l,NY,K]);
end;
```

// Ez для двух плоскостей XxZ [верхней и нижней]

```
for K:=1 to NZ-1 do      for l:=2 to NX-1 do begin
    EZ[l,1,K]:= EZY1[l,K]+(Ty-1)/(Ty+1)*(EZ[l,2,K]-EZ[l,1,K]);
    EZ[l,NY,K]:= EZYN[l,K]+(Ty-1)/(Ty+1)*(EZ[l,NY-1,K]-EZ[l,NY,K]);
end;
```

//БЛИЖНЯЯ И ДАЛЬНЯЯ ПЛОСКОСТИ

// Ey для двух плоскостей XxY [ближней и дальней]

```
for l:=2 to NX-1 do      for J:=1 to NY-1 do begin
    EY[l,J,NZ]:=EYZN[l,J]+(Tz-1)/(Tz+1)*(EY[l,J,NZ-1]-EY[l,J,NZ]);
    EY[l,J,1]:=EYZ1[l,J]+(Tz-1)/(Tz+1)*(EY[l,J,2]-EY[l,J,1]);
end
```

```

// Ex для двух плоскостей ХхУ [ближней и дальней]
for I:=1 to NX-1 do      for J:=2 to NY-1 do begin
    EX[I,J,NZ]:= EXZN[I,J]+(Tz-1)/(Tz+1)*(EX[I,J,NZ-1]-EX[I,J,NZ]);
    EX[I,J,1]:= EXZ1[I,J]+(Tz-1)/(Tz+1)*(EX[I,J,2]-EX[I,J,1]);
end;
// РЕБРА, параллельные оси Z
for K:=1 to NZ-1 do begin
    EZ[1,1,K]:=EZX1[1,K]+(Tx-1)/(Tx+1)*(EZ[2,1,K]-EZ[1,1,K]);
    EZ[1,NY,K]:=EZX1[NY,K]+(Tx-1)/(Tx+1)*(EZ[2,NY,K]-EZ[1,NY,K]);
    EZ[NX,1,K]:=EZXN[1,K]+(Tx-1)/(Tx+1)*(EZ[NX-1,1,K]-EZ[NX,1,K]);
    EZ[NX,NY,K]:=EZXN[NY,K]+(Tx-1)/(Tx+1)*(EZ[NX-1,NY,K]-EZ[NX,NY,K]);
end;
// РЕБРА, параллельные оси Y
for J:=1 to NY-1 do begin
    EY[1,J,1]:=EYX1[J,1]+(Tx-1)/(Tx+1)*(EY[2,J,1]-EY[1,J,1]);
    EY[1,J,NZ]:=EYX1[J,NZ]+(Tx-1)/(Tx+1)*(EY[2,J,NZ]-EY[1,J,NZ]);
    EY[NX,J,1]:=EYXN[J,1]+(Tx-1)/(Tx+1)*(EY[NX-1,J,1]-EY[NX,J,1]);
    EY[NX,J,NZ]:=EYXN[J,NZ]+(Tx-1)/(Tx+1)*(EY[NX-1,J,NZ]-EY[NX,J,NZ]);
end;
// РЕБРА, параллельные оси X
for I:=1 to NX-1 do begin
    EX[I,1,1]:=EXY1[I,1]+(Ty-1)/(Ty+1)*(EX[I,2,1]-EX[I,1,1]);
    EX[I,1,NZ]:=EXYN[I,NZ]+(Ty-1)/(Ty+1)*(EX[I,2,NZ]-EX[I,1,NZ]);
    EX[I,NY,1]:=EXY1[I,1]+(Ty-1)/(Ty+1)*(EX[I,NY-1,1]-EX[I,NY,1]);
    EX[I,NY,NZ]:=EXYN[I,NZ]+(Ty-1)/(Ty+1)*(EX[I,NY-1,NZ]-EX[I,NY,NZ]);
end;
end;

```

// сохранение приграничных полей для MursFirstABC

Procedure GetBorderValues;

```
var I,J,K: Integer;
begin
  // слева и справа
  for J:=1 to NY do for k:=1 to NZ-1 do      begin
    EZX1[J,K]:=EZ[2,J,K];
    EZXN[J,K]:=EZ[NX-1,J,K];                end;
  for J:=1 to NY-1 do for k:=1 to NZ do      begin
    EYX1[J,K]:=EY[2,J,K];
    EYXN[J,K]:=EY[NX-1,J,K];                end;
  // верх и низ
  for I:=1 to NX do for k:=1 to NZ-1 do      begin
    EZY1[I,K]:=EZ[I,2,K];
    EZYN[I,K]:=EZ[I,NY-1,K];                end;
  for I:=1 to NX-1 do for k:=1 to NZ do      begin
    EXY1[I,K]:=EX[I,2,K];
    EXYN[I,K]:=EX[I,NY-1,K];                end;
  // ближняя и дальняя
  for I:=1 to NX do for J:=1 to NY-1 do      begin
    EYZ1[I,J]:=EY[I,J,2];
    EYZN[I,J]:=EY[I,J,NZ-1];                end;
  for I:=1 to NX-1 do for J:=1 to NY do      begin
    EXZ1[I,J]:=EX[I,J,2];
    EXZN[I,J]:=EX[I,J,NZ-1];                end;
end;
```

Примечание: в литературе встречается и несколько иная форма записи условий Мура 1-го порядка. К приведенной здесь формуле добавляется еще один член, который сам Мур относит к условиям второго порядка, а другие авторы утверждают, что этот член должен быть и в формуле 1-го порядка.

4. Условия PML - идеально сочетающиеся слои

"Perfectly Matched Layer" выглядит как "идеально согласованный (сочетающийся) слой"

Условия PML обладают низким коэффициентом отражения (по некоторым данным в миллион раз меньше, чем у условий Мура), а также практической независимостью от угла падения волны.

К недостаткам условий PML следует отнести значительно больший объем требуемой памяти, чем для условий ABC и наличие нижней граничной частоты, для снижения которой требуется увеличения количества слоев PML, а, следовательно, требуемой памяти. Как следствие увеличения требуемого объема памяти происходит снижение скорости вычислений.

За пределами главного счетного объема добавляются дополнительные ячейки, окружающие счетный объем по периметру несколькими слоями. Электрическое и магнитное поле в этих ячейках вычисляется почти так же, как и в счетном объеме, но есть отличия.

Во-первых, в уравнения в обязательном порядке вводятся электрические и магнитные потери. В главном алгоритме этих потерь может не быть вообще. В уравнениях главного алгоритма [\(рис. 2\)](#) учтены только электрические потери. Магнитные потери вводятся так же, как и электрические, путем задания "плотности магнитных токов". Тогда уравнения Максвелла [\(1\) и \(2\)](#) выглядят так:

$$\begin{aligned} \text{rot}(\mathbf{H}) &= \partial \mathbf{D} / \partial t + \mathbf{J}; \\ \text{rot}(\mathbf{E}) &= - \partial \mathbf{B} / \partial t + \mathbf{J}^*; \end{aligned} \quad (9)$$

$$\text{где } \mathbf{D} = \varepsilon \varepsilon_0 \mathbf{E};$$

$$\mathbf{J} = \sigma \mathbf{E};$$

$$\mathbf{B} = \mu \mu_0 \mathbf{H}; \quad (10)$$

$$\mathbf{J}^* = \sigma^* \mathbf{H};$$

Отличие от (1) и (2) только в появлении \mathbf{J}^* - плотности "магнитных токов" и σ^* - "магнитной проводимости". С введением магнитных потерь уравнения (9) стали симметричными.

Во-вторых, вводится разделение векторов и их раздельное вычисление. Каждый вектор декартовой сетки \mathbf{U}_{ee} в границах PML делится на два параллельных вектора (две компоненты). Сумма этих векторов есть полный вектор: $\mathbf{E} = \mathbf{E}_x + \mathbf{E}_y$; $\mathbf{E} = \mathbf{E}_y + \mathbf{E}_z$; $\mathbf{H} = \mathbf{H}_x + \mathbf{H}_y$ и т.д. Обозначения расшифровываются так: \mathbf{E}_{xy}^x - вектор \mathbf{E} в направлении x , полученный через соседние векторы \mathbf{H}_z , лежащие на прямой, параллельной оси y . Понять это можно, посмотрев на систему уравнений Беренгера:

$$(11)$$

(11)

$$\varepsilon \frac{\partial E_{xy}}{\partial t} + \sigma_y E_{xy} = \frac{\partial(H_{zx} + H_{zy})}{\partial y}$$

$$\varepsilon \frac{\partial E_{xz}}{\partial t} + \sigma_z E_{xz} = -\frac{\partial(H_{yz} + H_{yx})}{\partial z}$$

$$\varepsilon \frac{\partial E_{yz}}{\partial t} + \sigma_z E_{yz} = \frac{\partial(H_{xy} + H_{xz})}{\partial z}$$

$$\varepsilon \frac{\partial E_{yx}}{\partial t} + \sigma_x E_{yx} = -\frac{\partial(H_{zx} + H_{zy})}{\partial x}$$

$$\varepsilon \frac{\partial E_{zx}}{\partial t} + \sigma_x E_{zx} = \frac{\partial(H_{yz} + H_{yx})}{\partial x}$$

$$\varepsilon \frac{\partial E_{zy}}{\partial t} + \sigma_y E_{zy} = -\frac{\partial(H_{xy} + H_{xz})}{\partial y}$$

$$\mu \frac{\partial H_{xy}}{\partial t} + \sigma_y^* H_{xy} = -\frac{\partial(E_{zx} + E_{zy})}{\partial y}$$

$$\mu \frac{\partial H_{xz}}{\partial t} + \sigma_z^* H_{xz} = \frac{\partial(E_{yz} + E_{yx})}{\partial z}$$

$$\mu \frac{\partial H_{yz}}{\partial t} + \sigma_z^* H_{yz} = -\frac{\partial(E_{xy} + E_{xz})}{\partial z}$$

$$\mu \frac{\partial H_{yx}}{\partial t} + \sigma_x^* H_{yx} = \frac{\partial(E_{zx} + E_{zy})}{\partial x}$$

$$\mu \frac{\partial H_{zx}}{\partial t} + \sigma_x^* H_{zx} = -\frac{\partial(E_{yz} + E_{yx})}{\partial x}$$

$$\mu \frac{\partial H_{zy}}{\partial t} + \sigma_y^* H_{zy} = \frac{\partial(E_{xy} + E_{xz})}{\partial y},$$

Примечание

Схему Беренгера называют «раздельной», т.к. вводится разделение векторов. Есть также предложения других авторов применять «нераздельную» схему. Если кто-нибудь проверял работу «нераздельной» схемы, поделитесь впечатлениями. Я не доверяю автору «нераздельной» схемы из-за целого ряда некорректных статей за его авторством, а также авторством его учеников и соратников. Сложилось впечатление, что некий клан печет статьи как пирожки не придавая внимания содержанию – лишь бы побольше опубликовать. Буду рад, если мое впечатление ошибочно.

В третьих, теоретически, если выполняется условие

$$\sigma_i / \varepsilon_0 = \sigma_i^* / \mu_0, \quad (12)$$

то на границе раздела двух сред скорость электромагнитных волн не изменяется и отражение равно нулю. В то же время, поскольку σ_i и σ_i^* не равны нулю, то происходит поглощение электромагнитных волн в недрах PML.

К сожалению, отражение все же есть:

- от первого слоя PML;
- между слоями PML, поскольку для экономии вычислительных ресурсов потери растут от слоя к слою (закон изменения потерь от первого слоя к последнему называется "профилем потерь");
- после последнего слоя PML, поскольку там находится PEC - граница.

Отражение от первого слоя PML и между слоями PML вызвано ошибками конечно-разностной дискретизации, и, в первую очередь, тем, что векторы E и H (а, следовательно, σ_i и σ_i^*) не совпадают в пространстве. Для снижения отражения внутри PML необходимо ограничивать скорость роста потерь некоторым разумным пределом.

Беренгер показал, что при каждом конкретном значении σ_i и σ_i^* , вследствие цифрового отражения, происходит отражение нормальных (перпендикулярных) к границе электромагнитных волн, частота которых ниже f_c (частота отсечки). Чем больше σ_i и σ_i^* , тем выше f_c . Поэтому при проникновении волны в границы PML вначале идет отражение от первого слоя с проводимостью σ_0 тех частот, которые ниже f_{c0} . Затем идет отражение от полуслоя с проводимостью σ_0^* частот ниже f_{c0}^* . Причем $f_{c0} < f_{c0}^*$. И так далее – все более и более высокие частоты отражаются, причем отражение от σ_i и от σ_i^* .

Отражение от PEC границы после последнего слоя PML происходит обычно уже для весьма ослабленной волны. Отраженная волна на обратном пути продолжает ослабляться. Но если слоев мало (обычно < 5), то отраженная волна может быть существенной.

Для уменьшения отражения от первого слоя значения σ_1 специально выбирается маленьким. Для уменьшения отражения между слоями профиль потерь выбирается с ограниченной скоростью роста потерь. Для уменьшения влияния волны, отраженной от PEC границы, увеличивается количество слоев PML.

Как вариант, Беренгер предлагает следующий геометрический профиль по

$$\sigma(\rho) = -\frac{\epsilon_0 c}{2 \Delta x} \frac{\ln g}{g^N - 1} \ln R(0) g^{\rho/\Delta x},$$

где g - коэффициент геометрической прогрессии; Δx - шаг по пространству; c - скорость света; N - номер PML -слоя, считая от интерфейса счетного региона и границы; ρ - расстояние от границы; $R(0)$ - коэффициент отражения от первого слоя. Рекомендуемое значение $R(0) = 0.01$ (1 %) Коэффициент прогрессии g рекомендуется брать 2,15. Это значение получено Беренгером экспериментально, хотя встречаются и другие рекомендации.

$\sigma^*(r)$ получается через $\sigma(r)$ с использованием (12).

На низких частотах наблюдается резкое увеличение коэффициента отражения от границ PML. Нижняя граничная частота отсечки для известного σ электрической проводимости на границе находится из выражения $f_c = \frac{\sigma(0)}{2\pi\epsilon_0}$ (14)

Для расчетов откликов от импульсов - ступенек с постоянной составляющей обратная величина к (14) - $1/f_c$ - это максимальное время счета, при котором коэффициент отражения от первого слоя еще не превышает заданного $R(0)$.

На самом деле, f_c полученное по (14), - сильно заниженное значение. Реально происходят отражения и от других слоев, которые, как указано выше, имеют более высокую частоту отсечки. Практически f_c нужно брать в 5 раз больше, что показывается в разделе [сравнение граничных условий](#).

Существуют и другие профили потерь. В статье Беренгера "PML for the FDTD Solution of Wave-Structure Interaction Problems" (IEEE trans. on antennas and prop., vol. 44, N1, Jan 1996) есть богатый материал для размышлений о выборе профиля потерь и количества слоев в зависимости от разных факторов.

Дискретизация уравнений (11) происходит так же, как и дискретизация уравнений главного алгоритма. Для двух векторов E_x :

$$E_x^{n+1}(i+1/2, j, k) = (1 - 0,5\sigma(r)\Delta t) / (1 + 0,5\sigma(r)\Delta t) E_x^n(i+1/2, j, k) + \Delta t / (1 + 0,5\sigma(r)\Delta t) / \Delta y \cdot (H_{xy}^{n+1/2}(i+1/2, j+1/2, k) + H_{xy}^{n+1/2}(i+1/2, j-1/2, k) - H_{zy}^{n+1/2}(i+1/2, j-1/2, k))$$

$$E_z^{n+1}(i+1/2, j, k) = (1 - 0,5\sigma(r)\Delta t) / (1 + 0,5\sigma(r)\Delta t) E_z^n(i+1/2, j, k) + \Delta t / (1 + 0,5\sigma(r)\Delta t) / \Delta z \cdot (H_{xz}^{n+1/2}(i+1/2, j, k+1/2) + H_{xz}^{n+1/2}(i+1/2, j, k-1/2) - H_{yx}^{n+1/2}(i+1/2, j, k-1/2))$$

Для остальных векторов E все аналогично. Для векторов H уравнения аналогичные, но вместо $\sigma(r)$ берется $\sigma^*(r)$.

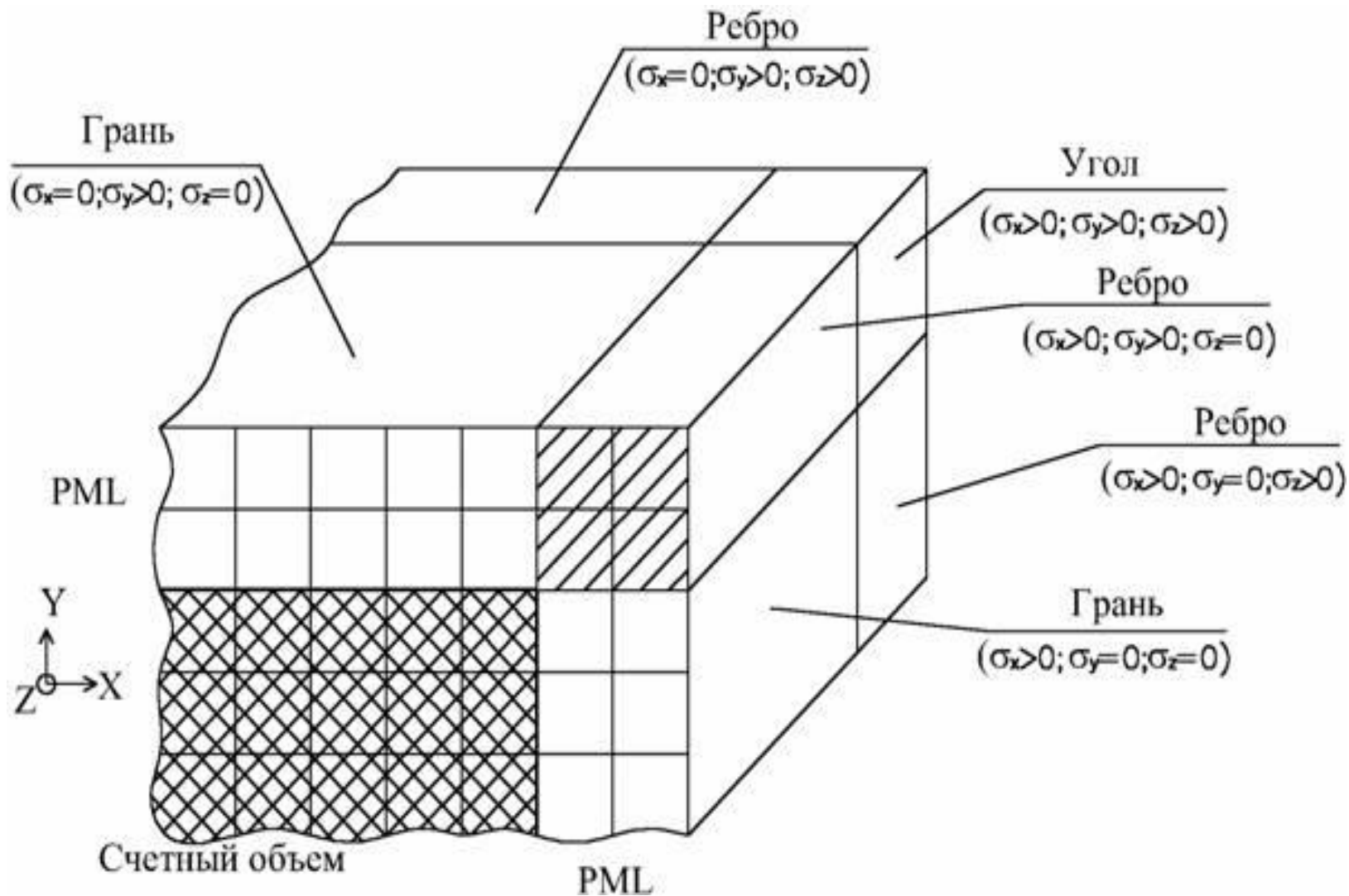
$$H_{xy}^{n+1/2}(i, j+1/2, k+1/2) = (1 - 0,5\sigma^*(r)\Delta t) / (1 + 0,5\sigma^*(r)\Delta t) H_{xy}^{n-1/2}(i, j+1/2, k+1/2) - \Delta t / (1 + 0,5\sigma^*(r)\Delta t) / \Delta y \cdot (E_{zx}^n(i, j+1/2, k+1/2) + E_{zy}^n(i, j+1/2, k+1/2) - E_{zx}^n(i, j, k+1/2) - E_{zy}^n(i, j, k+1/2))$$

Как можно заметить, в уравнениях для векторов E имеются суммы типа $(H_{zx}^{n+1/2}(i+1/2, j+1/2, k) + H_{zy}^{n+1/2}(i+1/2, j+1/2, k))$. Это и есть полный вектор, в данном случае H_z . Когда вычисляется поле на границе счетного объема, то из счетного объема берется полный вектор H_z , а из граничного PML-слоя – сумма $H_{zx} + H_{zy}$. Также и для других векторов.

На внешней границе PML, как уже говорилось, тангенциальные векторы E равны нулю (PES).

Профиль потерь зависит только от координаты, ведущей от интерфейса "счетный объем" - PML вглубь PML. На любой грани прямоугольного счетного объема вглубь PML ведет только одна координата. Допустим, это координата x . Тогда $\sigma_x(r)$ меняется по заданному закону, а $\sigma_y(r) = \sigma_z(r) = 0$. На ребрах вглубь PML ведут уже две координаты (одна из σ равна нулю), а в углах вглубь ведут все три координаты и, следовательно, изменяются все σ (рис. 1).

Рис. 1. Счетный объем в окружении слоев PML



Расширения базового алгоритма

Существует множество дополнений базовому алгоритму, расширяющих его возможности.

Здесь представлены два вида расширений:

- Сосредоточенные элементы
- Полярные диэлектрики

Сосредоточенные элементы

В англоязычной литературе сосредоточенные элементы называются «lumped», «lumped circuit elements».

M. Piket-May, A. Taflov J. Baron «FD-TD Modeling of Digital Signal Propagation in 3-D Circuits With Passive and Active Loads», IEEE Trans. On Microwave Theory and Techiques, 1994, vol.42, №8. В этой статье приведены уравнения резисторов, резистивных источников напряжения, конденсаторов, индуктивностей, диодов и биполярных транзисторов. В более поздних публикациях разные авторы совершенствуют модели, добавляют новые элементы (например, источники тока, полевые транзисторы).

Отсутствие сосредоточенных элементов сужает область применения метода FDTD. При их отсутствии в расчетных задачах можно конструировать резисторы – ячейки с такой проводимостью, чтобы их сопротивление было равно заданному, и конденсаторы – три ячейки вдоль одной линии, две из них являются обкладками, а центральная – диэлектриком с такой диэлектрической проницаемостью, чтобы емкость была равна заданной. Я использовал этот способ задания элементов, пока не потребовалось задать индуктивность. Большую индуктивность задавать из ячеек практически нереально. И здесь на помощь пришла вышеупомянутая статья. Приведу кратко суть статьи с собственными пояснениями.

Добавим к базовой формулировке вихревого уравнения для \mathbf{H} (см. систему (1 из [Введения](#))), которое выглядит как

$$\nabla \times \mathbf{H} = \mathbf{J}_c + \frac{\partial \mathbf{D}}{\partial t}$$

(здесь и далее форма записи взята

Предположим, что элемент находится в свободном пространстве и ориентир плотность тока выразится через ток I_L как

$$\mathbf{J}_L = \frac{I_L}{\Delta x \Delta y}$$

I_L может быть дифференциальной, линейной или нелинейной функцией электрического потенциала

$$E_z|_{i,j,k} \Delta z$$

Проведя вывод дискретного уравнения так, как это было во введении, но с учетом тока через сосредоточенный элемент, получим:

$$E_z|_{i,j,k}^{n+1} = E_z|_{i,j,k}^n + \frac{\Delta t}{\epsilon_0} \nabla \times H|_{i,j,k}^{n+1/2} - \frac{\Delta t}{\epsilon_0 \Delta x \Delta y} I_L^{n+1/2} \quad \bullet (1)$$

Примечание. Среднее слагаемое правой части уравнения – это общепринятый способ краткой записи разностной схемы. В данном случае это

$$(\Delta t / \epsilon_0 (H_x^{n+1/2}_{(i,j+1/2,k+1/2)} - H_x^{n+1/2}_{(i,j-1/2,k+1/2)}) / \Delta y - (H_y^{n+1/2}_{(i+1/2,j,k+1/2)} - H_y^{n+1/2}_{(i-1/2,j,k+1/2)}) / \Delta x),$$

Узнаете, что это за уравнение?

Резистор

Ток резистора находится по известной формуле $I=U/R$. В нашем случае $U=\Delta Z(E^{n+1}+E^n)/2$, т.е. берется среднее арифметическое напряженности E поля на соседних шагах по времени, чтобы получить напряженность на шаге $(n+1/2)$. Это приближение необходимо для придания устойчивости вычислительной схемы.

Имеем:

$$I_z|_{i,j,k}^{n+1/2} = \frac{\Delta z}{2R} (E_z|_{i,j,k}^{n+1} + E_z|_{i,j,k}^n)$$

Подставляя последнее уравнение в (1) и

$$E_z|_{i,j,k}^{n+1} = \left(\frac{1 - \frac{\Delta t \Delta z}{2R\epsilon_0 \Delta x \Delta y}}{1 + \frac{\Delta t \Delta z}{2R\epsilon_0 \Delta x \Delta y}} \right) E_z|_{i,j,k}^n + \left(\frac{\frac{\Delta t}{\epsilon_0}}{1 + \frac{\Delta t \Delta z}{2R\epsilon_0 \Delta x \Delta y}} \right) \nabla \times H|_{i,j,k}^{n+1/2} \quad \text{ИМ:}$$

Резистивный источник напряжения

Это резистор, в котором встроен источник напряжения. Очень удобный и абсолютно физичный элемент. Позволяет моделировать генератор с заданным выходным сопротивлением. Отличается от резистора только наличием дополнительного тока, равного напряжению источника деленному на сопротивление:

$$I_z|_{i,j,k}^{n+1/2} = \frac{\Delta z}{2R_s} \left(E_z|_{i,j,k}^{n+1} + E_z|_{i,j,k}^n \right) + \frac{V_s^{n+1/2}}{R_s}$$

$$E_z|_{i,j,k}^{n+1} = \left(\frac{1 - \frac{\Delta t \Delta z}{2R_s \epsilon_0 \Delta x \Delta y}}{1 + \frac{\Delta t \Delta z}{2R_s \epsilon_0 \Delta x \Delta y}} \right) E_z|_{i,j,k}^n + \left(\frac{\frac{\Delta t}{\epsilon_0}}{1 + \frac{\Delta t \Delta z}{2R_s \epsilon_0 \Delta x \Delta y}} \right) \nabla \times H|_{i,j,k}^{n+1/2} + \left(\frac{\frac{\Delta t}{R_s \epsilon_0 \Delta x \Delta y}}{1 + \frac{\Delta t \Delta z}{2R_s \epsilon_0 \Delta x \Delta y}} \right) V_s^{n+1/2}$$

Конденсатор

Ток через конденсатор $I = C(du/dt)$, где C – емкость. Производную напряжения по времени выразим через разность напряженности поля на соседних шагах по времени: $I_z|_{i,j,k}^{n+1/2} = \frac{C\Delta z}{\Delta t} (E_z|_{i,j,k}^{n+1} - E_z|_{i,j,k}^n)$

Дальнейший вывод несложен, как и предыдущие:

$$E_z|_{i,j,k}^{n+1} = E_z|_{i,j,k}^n + \left(\frac{\frac{\Delta t}{\epsilon_0}}{1 + \frac{C\Delta z}{\epsilon_0\Delta x\Delta y}} \right) \nabla \times H|_{i,j,k}^{n+1/2}$$

Индуктивность

Ток в индуктивности, как известно, равен $I = (1/L) \int U dt$, т.е. величина интегральная. В дискретной форме ток

запишем как $I_z|_{i,j,k}^{n+1/2} = \frac{\Delta z \Delta t}{L} \sum_{m=1}^n E_z|_{i,j,k}^m$

Отсюда:

$$E_z|_{i,j,k}^{n+1} = E_z|_{i,j,k}^n + \frac{\Delta t}{\epsilon_0} \nabla \times H|_{i,j,k}^{n+1/2} - \frac{\Delta z (\Delta t)^2}{\epsilon_0 L \Delta x \Delta y} \sum_{m=1}^n E_z|_{i,j,k}^m$$

Полученная формула работает очень хорошо и устойчиво. Для вычислений требуется хранение суммы поля E_z в данной ячейке на всех предыдущих шагах, что не представляет особой трудности.

Итак, приведено описание четырех сосредоточенных элементов.

Расширения базового

Полярные диэлектрики **алгоритма**

Наличие относительной диэлектрической проницаемости большей, чем единица, связано с поляризацией молекул диэлектрика. При приложении внешнего электрического поля к диэлектрику молекулы полярного диэлектрика поворачиваются, а неполярного - деформируются.

Здесь рассматриваются полярные диэлектрики. Их молекулы всегда представляют собой диполи. Например, молекулы воды. Наличие механически поворачивающихся молекул приводит к двум основным эффектам:

1. Всегда найдется такая высокая частота, при которой молекулы не успевают повернуться. Из-за этого диэлектрическая проницаемость падает.
2. При повороте молекул часть их кинетической энергии растрачивается на соударения с соседними молекулами и происходит нагрев диэлектрика. С ростом частоты этот процесс становится все заметнее и, наконец, говорят о том, что в диэлектрике растут потери на поляризацию или просто потери. С точки зрения стороннего наблюдателя, нагрев диэлектрика происходит так же, как и нагрев проводника с током за счет его сопротивления. Поэтому говорят о том, что в диэлектрике имеется некая проводимость потерь.

Непосвященный в эти процессы будет удивлен, но факт остается фактом: дистиллированная вода на частоте в пятьсот мегагерц ведет себя как проводник с проводимостью $\sim 0,07$ См/м, но постоянный ток вообще не проводит.

Белки, жиры, углеводы и вода, насыщенная солями, из которых состоят живые организмы, - все это в основном полярные диэлектрики. Сейчас модно рассчитывать эффекты взаимодействия электромагнитного излучения с человеком. Для синусоидального сигнала в расчетах можно брать электрофизические характеристики для конкретной частоты. Но для широкополосных воздействий учет частотных зависимостей совершенно необходим.

Применение метода FDTD для расчетов распространения нестационарных электромагнитных полей в условиях частотной дисперсии

В классической постановке [1]

1. Yee, K. S.: "Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media." IEEE Transactions on Antennas and Propagation, Vol. 14, No. 3, 1966, pp. 302-307.

метод FDTD требует, чтобы значения проводимости и диэлектрической проницаемости материалов не зависели от частоты. В то же время, для многих материалов эти параметры имеют ярко выраженную частотную зависимость. К частотно-зависимым материалам относятся, в частности биологические ткани и вода. Для биологических тканей в диапазоне частот от 40 МГц до 400 МГц диэлектрическая проницаемость уменьшается в 2-3 раза, проводимость возрастает также в 2-3 раза.

Если вычисления по методу FDTD проводятся для одной частоты, то можно задать значения проводимости и диэлектрической проницаемости материалов для конкретной частоты и этого обычно достаточно. Проблемы возникают тогда, когда необходимо провести численное моделирование взаимодействия широкополосных электромагнитных полей (например, коротких импульсов) с частотно-зависимыми материалами.

В [2]

2. Luebbers R. et al.: "A frequency-depended finite-difference time-domain formulation for dispersive materials." IEEE Transactions on Electromagnetic Compatibility, Vol. 32, No. 3, Aug. 1990, pp. 222-227.

приводится частотно-зависимая формулировка $(FD)^2TD$ (frequency-dependent FDTD). Эта формулировка приведена с упрощениями, она не учитывает проводимости диэлектрика. Между тем, биологические ткани имеют существенную проводимость, которую следует учесть.

В [3]

3. Sullivan D. M.: "A frequency-depended FDTD Method for Biological Applications." IEEE Transactions on Microwave Theory and Techniques, Vol. 40, No. 3, March 1992, pp. 532-539.

приводит новую формулировку $(FD^2)TD$, в которой учитывается проводимость биологических тканей. К несчастью, эта формулировка в статье приведена с ошибками.

В этой статье приводится новая формулировка $(FD^2)TD$ для биологических тканей с учетом проводимости среды.

Учет проводимости биологических тканей в уравнениях (FD²)TD

Предположим, что биологические материалы линейные и изотропные, магнитная проницаемость является константой, не зависящей от частоты. Для этого случая в [2] приводится вывод уравнений без учета проводимости. Добавляя проводимость (это J в (1), см. ниже), повторим вывод уравнений.

Исходными являются уравнения Максвелла:

$$\nabla \times \vec{E} = -\frac{1}{c} \frac{\partial \vec{H}}{\partial t} \quad (1)$$

$$\nabla \times \vec{H} = \frac{1}{c} \frac{\partial \vec{E}}{\partial t} + \vec{J} \quad (2)$$

где \vec{J} — удельная проводимость. Из уравнения (2) следует выражение для H , которое полностью совпадает с классической формулировкой [1], поэтому вывод формулировки для H не приводится.

Вектор смещения \vec{D} из (1) во временной области записывается как [2, 3]:

где ϵ_0 - диэлектрическая постоянная, ϵ_∞ - относительная диэлектрическая проницаемость на "бесконечной" частоте, ϵ - электрическая восприимчивость.

Используя обозначения, принятые в [1], запишем и для каждой компоненты векторов \vec{D} и \vec{E} в (3) можем записать:

$$D_x = \epsilon_0 E_x + \epsilon \frac{\partial E_x}{\partial t} + \epsilon \frac{\partial^2 E_x}{\partial t^2} + \dots$$
(4)

Уравнение (1) с учетом того, что $x=i\Delta x$, $y=j\Delta y$ и $z=k\Delta z$, в конечно-разностной дискретной форме в декартовых координатах для компоненты D_y имеет вид:

$$D_y = \epsilon_0 E_y + \epsilon \frac{\partial E_y}{\partial t} + \epsilon \frac{\partial^2 E_y}{\partial t^2} + \dots$$
(5)

де:

[The page contains several lines of extremely faint, illegible handwritten or typed text.]

51

Для удобства введем обозначения:

$$\begin{aligned} & \frac{1}{\sqrt{1 - \beta^2}} = \gamma \\ & \frac{1}{\sqrt{1 - \beta^2}} = \gamma \\ & \frac{1}{\sqrt{1 - \beta^2}} = \gamma \\ & \frac{1}{\sqrt{1 - \beta^2}} = \gamma \end{aligned}$$

(9)

Подставляя (9) в (8) и решая последнее уравнение относительно β получаем:

$$\begin{aligned} & \frac{1}{\sqrt{1 - \beta^2}} = \gamma \\ & \frac{1}{\sqrt{1 - \beta^2}} = \gamma \\ & \frac{1}{\sqrt{1 - \beta^2}} = \gamma \\ & \frac{1}{\sqrt{1 - \beta^2}} = \gamma \end{aligned}$$

(10)

Выражения для компонент E_x и E_z могут быть получены аналогичным образом.

Если относительная диэлектрическая проницаемость не зависит от частоты, то $\chi_m = 0$ для всех m и $\chi_0 = 0$. В этом случае выражение (10) совпадает с [1].

Для полярных диэлектриков комплексная диэлектрическая проницаемость определяется как (формула Дебая):

$$\varepsilon(\omega) = \varepsilon_\infty + \chi(\omega), \quad (11)$$

где $\chi(\omega)$ – комплексная диэлектрическая проницаемость, зависящая от частоты.

Фурье- преобразование функции $\chi(\omega)$ имеет вид:

где $\chi(t)$ – функция, зависящая от шага по времени.

Так как $t = m\Delta t$ имеем:

$$\chi(t) = \sum_{m=-\infty}^{\infty} \chi_m \delta(t - m\Delta t)$$

$$\chi_m = \frac{1}{\Delta t} \int_{m\Delta t}^{(m+1)\Delta t} \chi(t) dt$$

$$\chi_m = \frac{1}{\Delta t} \int_{m\Delta t}^{(m+1)\Delta t} \chi(t) dt = \frac{1}{\Delta t} \int_{m\Delta t}^{(m+1)\Delta t} \chi(t) dt$$

Выражение (10) содержит член —
который, на первый взгляд, для
вычисления требует хранения большого
количества переменных. Но, поскольку
функция $\chi(t)$ носит экспоненциальный
характер, то, как показано в [2],
определяя, что
можем записать: (12)

Таким образом, для каждой компоненты
и требуется только одна
переменная.

Вычисление электромагнитного поля в дальней зоне с использованием метода FDTD и интеграла Кирхгофа

Решение уравнений Максвелла методом FDTD дает возможность вычисления электромагнитного внутри некоторой ограниченной области пространства. Ограничение на размеры области, в которой проводятся вычисления, накладываются ограниченной вычислительной мощностью применяемых ЭВМ, а именно тремя факторами: ограниченной скоростью выполнения арифметических операций, ограниченным объемом оперативной памяти и ограниченной скоростью обмена данными между процессором с памятью. Однако существует ряд задач, в которых необходим расчет электромагнитных полей на больших расстояниях от некоторого объекта, излучающего или рассеивающего электромагнитное поле. В их числе расчет диаграммы направленности антенн, определение эффективной поверхности отражения радиолокационных целей, решение задач дифракции и. др. В данной статье предложен эффективный способ вычисления полей в дальней зоне с использованием результатов вычислений ближнего поля методом FDTD. Под дальним полем здесь подразумевается поле за пределами вычислительного объема, а ближнее поле здесь означает поле, вычисляемое непосредственно методом FDTD.

Преобразование ближнего поля в дальнее поле с использованием интеграла

Кирхгофа

Существует несколько методов преобразования ближнего поля в дальнее поле. Все они включают интегрирование по замкнутой поверхности, которая охватывает излучающий или рассеивающий объект. А в остальном имеются значительные отличия. Одни методы интегрируют, используя поверхностные эквивалентные токи (электрические и магнитные), другие непосредственно используют поля E и H . Одни методы применяются в частотной области, другие работают во временной области.

Например, для вычисления дальнего поля часто применяют интегрирование по элементам, на которые разбивается замкнутая поверхность. Каждый элемент поверхности является элементарным электрическим и магнитным диполем одновременно. Если при этом ближнее поле вычисляется методом FDTD, выполняются следующие операции:

- Все компоненты E и H полей на поверхности, по которой происходит интегрирование, сохраняются на всех шагах по времени. Т.е. к концу вычислений известна временная форма поля в каждой точке поверхности.
- Поля E и H на поверхности преобразуются в эквивалентные электрические и магнитные токи.
- Осуществляется преобразование токов на поверхности в частотную область.
- Вычисляются составляющие E_ϕ , E_θ , E_r и соответствующие составляющие поля H для всех частот и от каждого элемента Гюйгенса на поверхности. Элементы Гюйгенса в данном случае - это обычные ячейки FDTD. С каждой ячейкой сопоставляется своя собственная система полярных координат.
- Значения E_ϕ , E_θ , E_r и соответствующих составляющих поля H преобразуются в прямоугольные координаты (E_x , E_y , E_z) и складываются в точке наблюдения. Только на этом шаге происходит собственно интегрирование по поверхности.
- Производится обратное преобразование во временную область. Теперь искомое поле найдено.

Существует ряд модификаций приведенного метода расчета дальнего поля, но при применении метода FDTD более логичным выглядит вычисление поля в дальней зоне с использованием только временной области и без преобразования полей в эквивалентные токи. Кроме того, желательно обойтись без преобразования в другие системы координат. Такой метод существует – это вычисления с использованием поверхностного интеграла Кирхгофа.

Способ применения поверхностного интеграла Кирхгофа совместно с методом FDTD приводится в [3]. Однако в [3] допущены три грубые ошибки в формулах. Кроме того, в порядок самого интегрирования введена путаница. Пришлось заново вывести формулы, воспользовавшись идеей Рамахи (Ramahi, автор [3]). Результат получился хорошим, поэтому, несмотря на досадные ошибки в статье, хочется выразить автору свою благодарность.

Интеграл Кирхгофа связывает поле внутри ограниченного объема с полем и его производными на поверхности, ограничивающей объем. Эта формула выведена в середине 19 века немецким физиком Г. Кирхгофом, и во временной области имеет вид [3]:

$$\frac{1}{4\pi} \oint_S \left[\frac{\partial \psi}{\partial n} + \frac{1}{c} \frac{\partial^2 \psi}{\partial t^2} R \right] dA' \quad (6)$$

где p, p' – точка наблюдения и точка на поверхности соответственно; \tilde{n} – единичный вектор нормали к поверхности; ψ – скалярная функция, которая может быть любой из шести компонент поля; R – расстояние от точки наблюдения до точки на поверхности; \vec{R} – вектор $p - p'$; c – скорость света; A' – площадь элемента поверхности. В формуле (6) ret означает, что интегрирование осуществляется с учетом запаздывающего времени $t' = t - R/c$. Вектор нормали направлен внутрь замкнутого объема

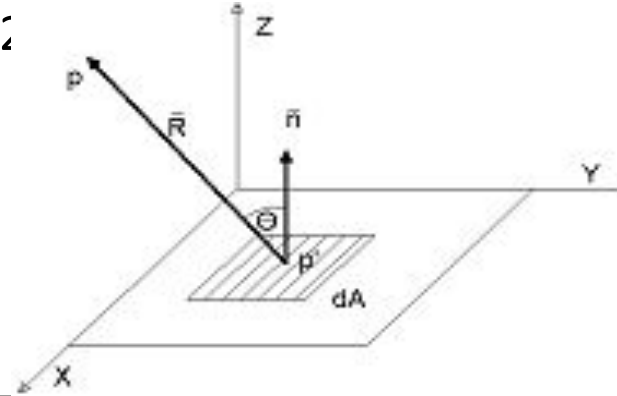
3. O. M. Ramahi, "Near- and far-field calculations in FDTD simulations using Kirchhoff surface integral representation", IEEE Transactions on Antennas and Propagation, vol. 45, no 5, may 1997.

Формула (6) выражает принцип Гюйгенса, согласно которому каждая точка на волновом фронте служит фиктивным источником воображаемой сферической волны. Каждый участок поверхности dA' излучает волну, которая в точку наблюдения r приходит с задержкой R/c . При этом на каждом шаге FDTD по времени на поверхности интегрирования возникает совокупность фиктивных источников, поле от которых придет в точку наблюдения с *разным запаздыванием*, поскольку расстояние R для всех точек различно. Это означает, что на одном временном шаге FDTD из (6) получаются вклады участков dA' в *разные* временные участки выходного сигнала в точке наблюдения.

Шаг по времени при вычислении интеграла (6) тесно связан с шагом по времени FDTD и равен ему. Выходная последовательность в точке наблюдения имеет такой же шаг по времени. Однако задержка R/c может не быть кратной шагу по времени. Поэтому получаемое время задержки округляется до ближайшего времени, кратного шагу FDTD. Возникающая при этом ошибка незначительна, т.к. шаг по времени в классическом FDTD мал по сравнению с периодом колебаний вычисляемого сигнала.

Чтобы привести (6) к виду, удобному для применения совместно с алгоритмом FDTD, необходимо выполнить ряд преобразований. При этом учтем, что все участки поверхности, по которой ведется интегрирование, в алгоритме FDTD всегда перпендикулярны одной из осей координат.

Предположим, что поверхность dA' перпендикулярна оси Z и точка p' имеет координаты (i, j, k_o) (рис. 1)



Применим преобразование подынтегральных слагаемых (6) в конечные разности:

$$\begin{aligned} & \frac{1}{4\pi R^2} \left(\frac{\partial}{\partial x} \left(\frac{1}{R} \frac{\partial \phi}{\partial x} \right) + \frac{\partial}{\partial y} \left(\frac{1}{R} \frac{\partial \phi}{\partial y} \right) + \frac{\partial}{\partial z} \left(\frac{1}{R} \frac{\partial \phi}{\partial z} \right) \right) \\ & \approx \frac{1}{4\pi R^2} \left(\frac{\partial}{\partial x} \left(\frac{1}{R} \frac{\partial \phi}{\partial x} \right) + \frac{\partial}{\partial y} \left(\frac{1}{R} \frac{\partial \phi}{\partial y} \right) + \frac{\partial}{\partial z} \left(\frac{1}{R} \frac{\partial \phi}{\partial z} \right) \right) \end{aligned} \quad (7)$$

Для упрощения дальнейшей записи введем обозначения:

$$\begin{aligned} \Psi(t') &= \Psi(n+1), \quad \Psi(t'-\Delta t) = \Psi(n), \quad \Psi(t'+\Delta t) = \Psi(n+2). \end{aligned} \quad (8)$$

и применим стандартные для FDTD обозначения для дискретных значений времени: $\Psi(t') = \Psi(n+1)$, $\Psi(t'-\Delta t) = \Psi(n)$, $\Psi(t'+\Delta t) = \Psi(n+2)$. Кроме того, вспомним, что время в точке наблюдения $t=t'+R/c$. Округляя t до ближайшего целого шага по времени обозначим его как t_n^* .

Выражение (6) с учетом введенных обозначений (7) и (8) запишется для одной площадки dA в виде:

$$\begin{aligned} & \sum_{i,j} A_{i,j} \left[\Psi(t_n^*) - \Psi(t_n^* - \Delta t) \right] \\ &= \sum_{i,j} A_{i,j} \left[\Psi(n+1) - \Psi(n) \right] \end{aligned} \quad (9)$$

где $A_{i,j} = \Delta x \Delta y$.

В (9) нет смысла записывать сумму по всем участкам поверхности, т.к. t_n^* в общем случае для каждого участка имеет разное значение, поскольку различно расстояние R . Временная последовательность в точке наблюдения p получается суммированием значений $\Psi(p, t_n^*)$ на каждом шаге по времени по всем участкам поверхности с учетом времени запаздывания.

В (9) значение функции $\Psi(p, t_n^*)$ вычисляется с использованием значений Ψ на трех шагах по времени. Но это не означает, что необходимо всегда помнить значения двух предыдущих шагов. Сгруппируем слагаемые (9) по временным шагам и запишем в виде:

$$\Psi(p, t_n^*) = \sum_{k=0}^{n-2} \left(\Psi(p, t_{n-k}^*) - \Psi(p, t_{n-k-1}^*) \right) + \Psi(p, t_0^*) \quad (10)$$

где

$$\Psi(p, t_{n-k}^*) - \Psi(p, t_{n-k-1}^*) = \sum_{j=0}^{k-1} \left(\Psi(p, t_{n-k+j}^*) - \Psi(p, t_{n-k+j-1}^*) \right) + \Psi(p, t_{n-k}^*) - \Psi(p, t_{n-k-1}^*) \quad (11)$$

В приведенных (10) и (11) присутствуют шаги n , $n+1$ и $n+2$. Но можно вычислить все эти значения для одного шага, допустим, текущего шага $n+1$. Тогда, если считать, что вычисляется шаг $n+1$, то $F_1(n)$ добавляется в t_{n+1}^* - шаг вычисляемой последовательности. $F_2(n+1)$ добавляется в предыдущую временную точку t_n^* , а $F_3(n+2)$ добавляется еще на один временной шаг раньше в вычисляемой последовательности (t_{n-1}^*). В этом случае $F_3(n+2) = -F_1(n)$.

Так что не требуется хранить значения полей на поверхности интегрирования. Но для ускорения вычислений можно заранее вычислить значения расстояний и косинусы углов от всех участков поверхности до точки наблюдения.

Вместо функции Ψ можно свободно подставлять E_x, E_y, E_z, H_x, H_y или H_z .

Все компоненты ячейки Y_{ee} находятся в разных местах пространства. Но расстояния и косинусы углов вполне можно вычислить общие для всех компонент. Главное, чтобы они принадлежали одной ячейке. Тогда при вычислении дальнего поля рассчитанные компоненты также окажутся сдвинутыми в пространстве, что удобно использовать при тестировании программы, когда поле одновременно, в одной и той же ячейке, вычисляется как непосредственно алгоритмом FDTD, так и решением поверхностного интеграла Кирхгофа.

Здесь приведен вывод формулы только для одной поверхности. Для остальных пяти поверхностей вывод формул аналогичен. Интегрирование следует вести строго по замкнутой поверхности.

Границы интегрирования можно приблизить вплотную к объекту, заданному в расчетах FDTD.

При вычислении полей на довольно больших расстояниях из уравнения (9) можно исключить слагаемое, убывающее как $1/R^2$, а угол направления на точку наблюдения вычислить один на всю грань поверхности интегрирования. Этот случай соответствует случаю дальней зоны в ее обычном понимании и позволяет ускорить расчеты и уменьшить требуемую для расчетов память.