Υπολογιστικές Μέθοδοι

http://eclass.uoa.gr/courses/PHYS186/

Διδάσκοντες:
Φ. Διάκονος
Δ. Φασουλιώτης

- Επίλυση μερικών διαφορικών εξισώσεων

Τα δύο είδη ταξινόμησης των ΜΔΕ:

Σύμφωνα με τα μορφολογικά χαρακτηριστικά τους, οι ΜΔΕ μπορούν να χωριστούν σε τρεις κατηγορίες:

$$a\frac{\partial^2 A}{\partial x^2}+b\frac{\partial^2 A}{\partial x\partial y}+c\frac{\partial^2 A}{\partial y^2}+d\frac{\partial A}{\partial x}+e\frac{\partial A}{\partial y}+fA+g=0$$

b^2-4ac>0 υπερβολική
b^2-4ac=0 παραβολική
b^2-4ac<0 ελλειπτική

Σύμφωνα με την υπολογιστική αντιμετώπισή τους

Προβλήματα αρχικών τιμών

Θερμική διάχυση

$$\frac{\partial}{\partial t}T(x,t)=\kappa\frac{\partial^2}{\partial x^2}\;T(x,t)$$

Κυματική εξίσωση

$$\frac{\partial^2}{\partial t^2}A(x,t)=v^2\frac{\partial^2}{\partial x^2}A(x,t)$$

Προβλήματα συνοριακών τιμών

$$\frac{\partial^2}{\partial x^2}\Phi(x,y)+\frac{\partial^2}{\partial y^2}\Phi(x,y)=-\frac{1}{\varepsilon}\rho(x,y)$$

Εξίσωση Poisson

Παραβολικές ΜΔΕ

$$\frac{\partial}{\partial t}T(x,t)=\kappa\frac{\partial^2}{\partial x^2}\ T(x,t)$$

1. Conduction of heat in bars and solids
2. Diffusion of concentration of liquid or gaseous substance in physical chemistry
3. Diffusion of neutrons in atomic piles
4. Diffusion of vorticity in viscous fluid flow
5. Telegraphic transmission in cables of low inductance or capacitance
6. Equilization of charge in electromagnetic theory.
7. Long wavelength electromagnetic waves in a highly conducting medium
8. Slow motion in hydrodynamics
9. Evolution of probability distributions in random processes.

$$\frac{\partial^2}{\partial t^2} A(x,t) = v^2 \frac{\partial^2}{\partial x^2} A(x,t)$$

1. Linearized supersonic airflow
2. Sound waves in a tube or a pipe
3. Longitudinal vibrations of a bar
4. Torsional oscillations of a rod
5. Vibration of a flexible string
6. Transmission of electricity along an insulated low-resistance cable
7. Long water waves in a straight canal.

Ελλειπτικές ΜΔΕ

$$\frac{\partial^2}{\partial x^2}\Phi(x,y)+\frac{\partial^2}{\partial y^2}\Phi(x,y)=-\frac{1}{\varepsilon}\rho(x,y)$$

1. Steady state temperature
2. Steady state electric field (voltage)
3. Inviscid fluid flow
4. Gravitational field.

Φιλοσοφία της μεθόδου

1) Δημιουργούμε χρονικό και χωρικό πλέγμα με βήμα **τ** και **h**
2) Σε κάθε χρονικό βήμα υπολογίζουμε όλες τις χωρικές ποσότητες **ικανοποιώντας τις** όποιες **συνοριακές συνθήκες**
3) Προχωράμε σε επόμενο χρονικό βήμα

Για να το πετύχουμε αυτό μετατρέπουμε τις παραγώγους σε διαφορές:

$$\frac{\partial A(\vec{r},t)}{\partial t} = \frac{A(\vec{r},t_{n+1}) - A(\vec{r},t_n)}{\tau} \qquad \frac{\partial A(\vec{r},t)}{\partial t} = \frac{A(\vec{r},t_{n+1}) - A(\vec{r},t_{n-1})}{2\tau}$$

$$\frac{\partial^2 A(\vec{r},t)}{\partial t^2} = \frac{A(\vec{r},t_{n+1}) - 2A(\vec{r},t_n) + A(\vec{r},t_{n-1})}{\tau^2}$$

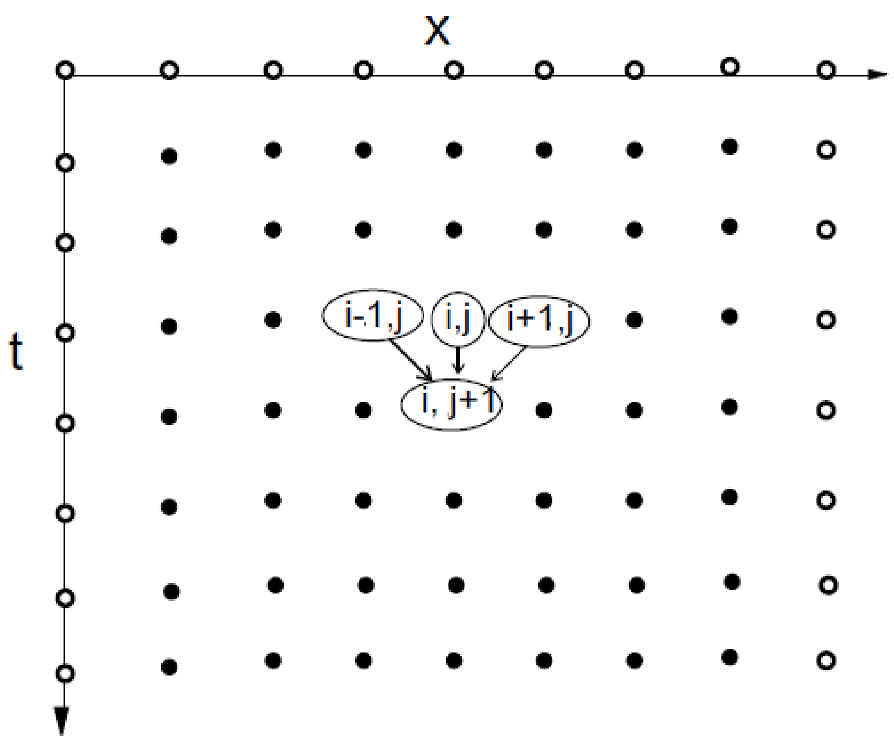
Οπότε η διαφορική εξίσωση μετατρέπεται σε εξίσωση διαφορών

Φιλοσοφία της μεθόδου

1) Δημιουργούμε χρονικό και χωρικό πλέγμα με βήμα **τ** και **h**

2) Σε κάθε χρονικό βήμα υπολογίζουμε όλες τις χωρικές ποσότητες **ικανοποιώντας τις** όποιες **συνοριακές συνθήκες**

3) Προχωράμε σε επόμενο χρονικό βήμα



Θερμική διάχυση

$$\mathbf{H} = -K\,\nabla T(\mathbf{x},t)$$

$$Q(t) = \int d\mathbf{x}\, C\,\rho(\mathbf{x})\,T(\mathbf{x},t)$$

$$\frac{\partial T(\mathbf{x},t)}{\partial t} = \frac{K}{C\rho}\nabla^2 T(\mathbf{x},t)$$

Μία διάσταση

$$\frac{\partial}{\partial t}T(x,t) = \kappa\,\frac{\partial^2}{\partial x^2}T(x,t)$$

Για να ορίσουμε το πρόβλημα χρειαζόμαστε

Αρχικές συνθήκες

Συνοριακές συνθήκες

π.χ.

$$T(x,t=0) = 100^\circ\mathrm{C} \qquad T(x=0,t) = T(x=L,t) = 0^\circ\mathrm{C}$$

$$\frac{\partial}{\partial t}T(x,t) = \kappa \frac{\partial^2}{\partial x^2}T(x,t)$$

$$\frac{\partial}{\partial t}T(x,t) \rightarrow \frac{T_i^{n+1} - T_i^n}{\tau}$$

$$\frac{\partial^2}{\partial x^2}T(x,t) \rightarrow \frac{T_{i+1}^n + T_{i-1}^n - 2T_i^n}{h^2}$$

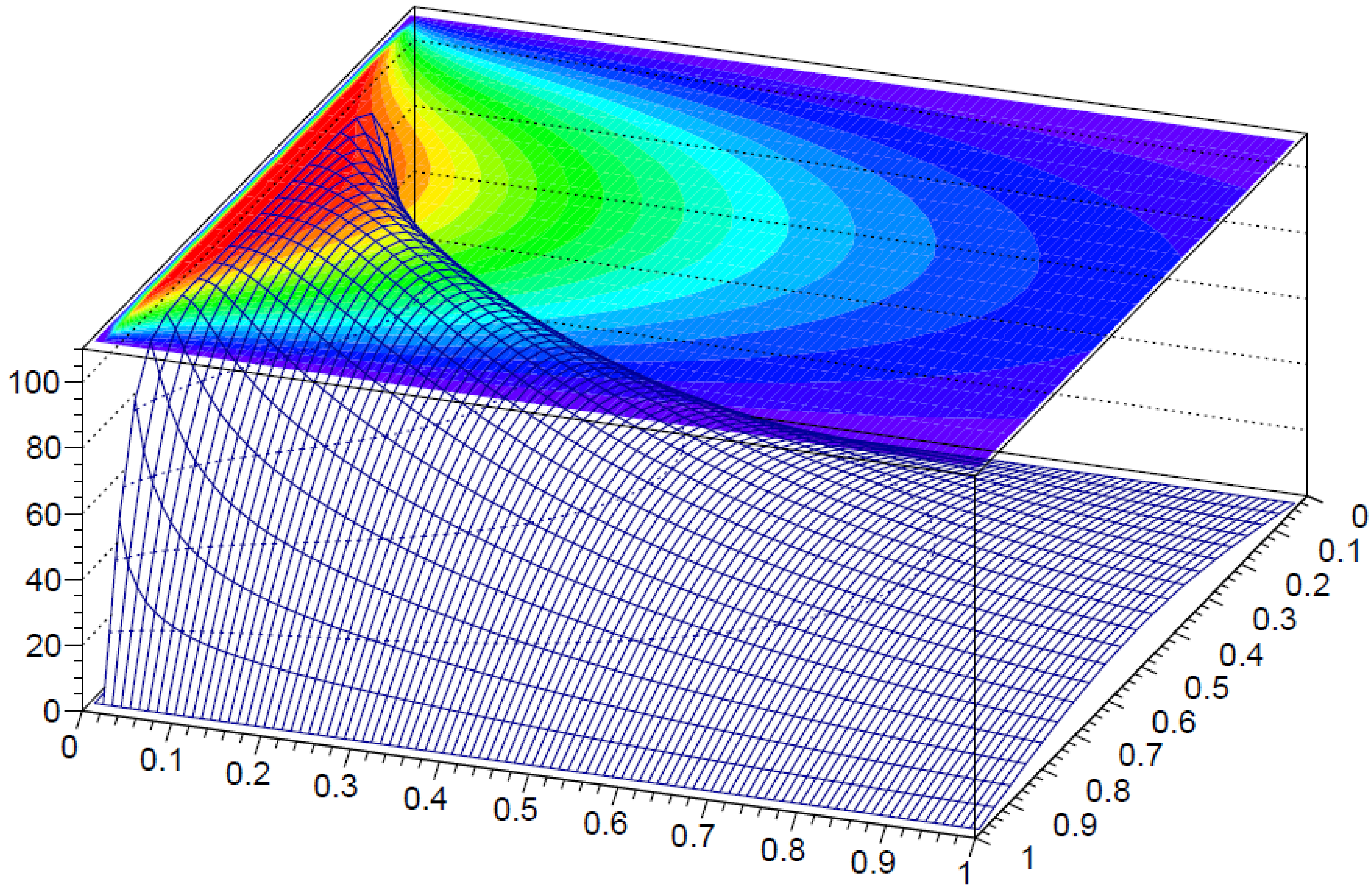
$$\boxed{T_i^{n+1} = T_i^n + \frac{\kappa\tau}{h^2}(T_{i+1}^n + T_{i-1}^n - 2T_i^n)}$$

Forward Time Centered Space

Θερμική διάχυση: Μέθοδος FTCS

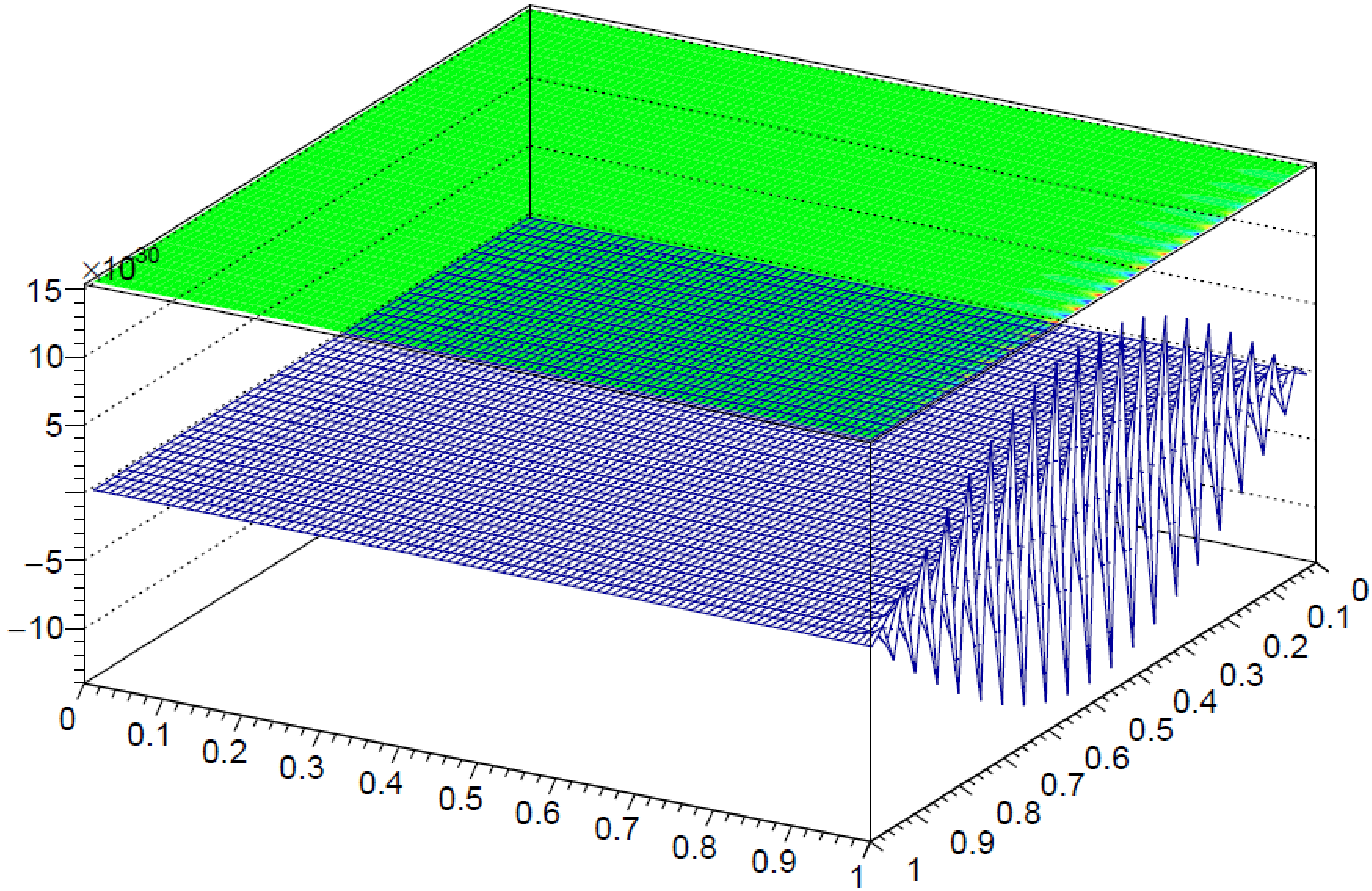
```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <TH1F.h>

void ftcs(double h, double dt)
{
          const int nx=1./h+1;
          TH2F *his=new TH2F("his"," ",nx, 0., 1.,100,0.,1.);
          double T[nx], T_new[nx];
          double time=0; int nt=10./dt;
          double k=0.5;
          double ksi=k*dt/(h*h);
          cout <<ksi<<endl;
          for(int i=0; i<nx; i++)T[i] = 100.;     // initial condition
          T[0]=0.; T[nx-1]=0.;                    // boundary conditions
          for(int it=1;it<nt;it++)
          {
            for(int i=1; i<nx;i++) T_new[i]=T[i]+ksi*(T[i+1]+T[i-1]-2*T[i]);
            for(int i=1; i<nx;i++) T[i]=T_new[i];
            if(it%10==0){
            for(int i=0;i<nx;i++)his->SetBinContent(i+1,it/10+1,T[i]); }
          }
          his->Draw("surf3");
}
```

Θερμική διάχυση: Μέθοδος FTCS



Ksi<0.5

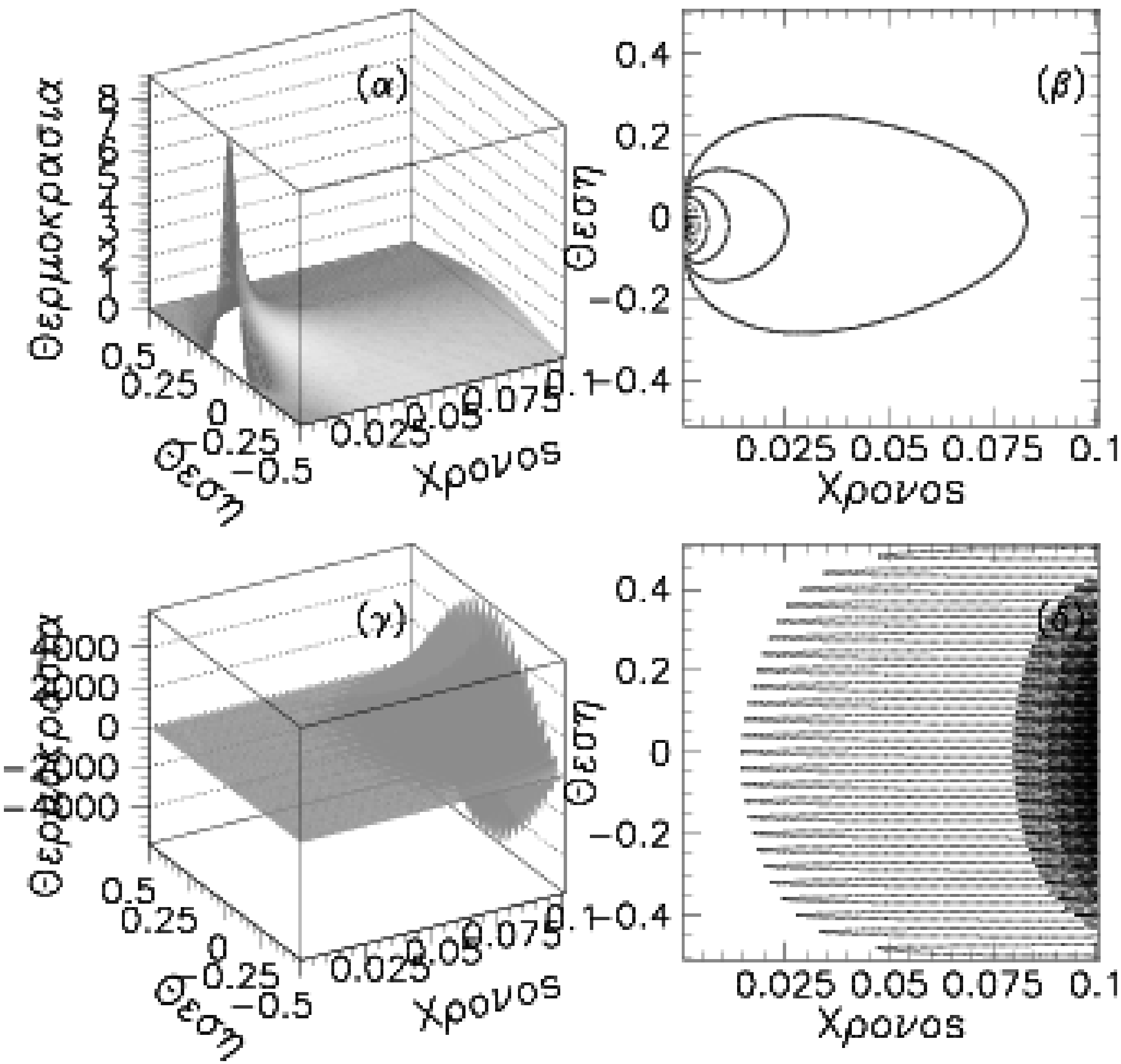
Θερμική διάχυση: Μέθοδος FTCS



Ksi=0.52

Θερμική διάχυση: Μέθοδος FTCS

Θερμική διάχυση με αρχική συνθήκη συνάρτηση δ και άκρα σε μηδενική θερμοκρασία



$\tau > h^2/2\kappa$

Επιλογή βήματος
(κριτήριο Von Neumann)

Θεωρούμε λύσεις της διαφορικής εξίσωσης της μορφής:

$$u_j^n = \xi^n e^{ikjh}$$

Όπου ξ=ξ(k) είναι ένας μιγαδικός αριθμός που εξαρτάται από το k.

Έτσι η χρονική εξάρτηση είναι διαδοχικές ακέραιες δυνάμεις του μιγαδικού αριθμού ξ.

$\Rightarrow$

Ευστάθεια :

$|\xi(k)| \leq 1$

για κάθε k

Αντικαθιστώντας στην εξίσωση διαφορών FTCS, έχουμε:

$$\xi(k) = 1 + \frac{\kappa\tau^2}{h}(\cos kh - 2)$$

$\Rightarrow$

$\tau \leq h^2/2\kappa$

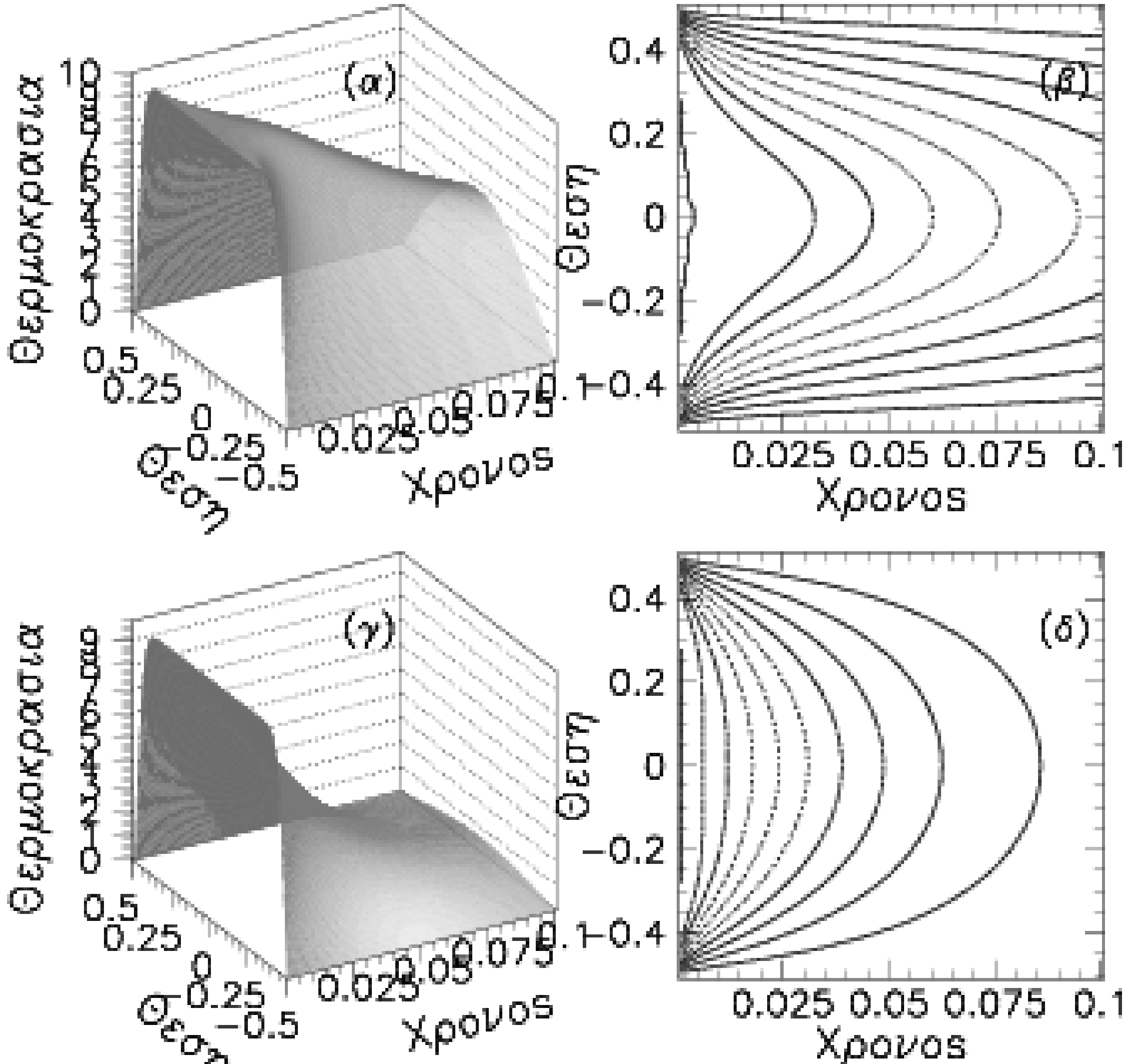
Θερμική διάχυση: Μέθοδος FTCS

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <TH1F.h>

void ftcs(double h, double dt)
{
          const int nx=1./h+1;
          TH2F *his=new TH2F("his"," ",nx, 0., 1.,100,0.,1.);
          double T[nx], T_new[nx];
          double time=0; int nt=10./dt;
          double k=1.;
          double ksi=k*dt/(h*h);
          cout <<ksi<<endl;
          for(int i=0; i<nx; i++)T[i] = 100.;      // initial condition
          T[0]=0.; T[nx-1]=0.;                     // boundary conditions
          for(int it=1;it<nt;it++)
          {
// Radiation on top of heat flow
            for(int i=1; i<nx;i++) T_new[i]=T[i]+ksi*(T[i+1]+T[i-1]-2*T[i]) -a*pow(T[i],4);
            for(int i=1; i<nx;i++)T[i]=T_new[i];
            if(it%10==0){
            for(int i=0;i<nx;i++)his->SetBinContent(i+1,it/10+1,T[i]); }
          }
          his->Draw("surf3");
}
```

Θερμική διάχυση: Μέθοδος FTCS

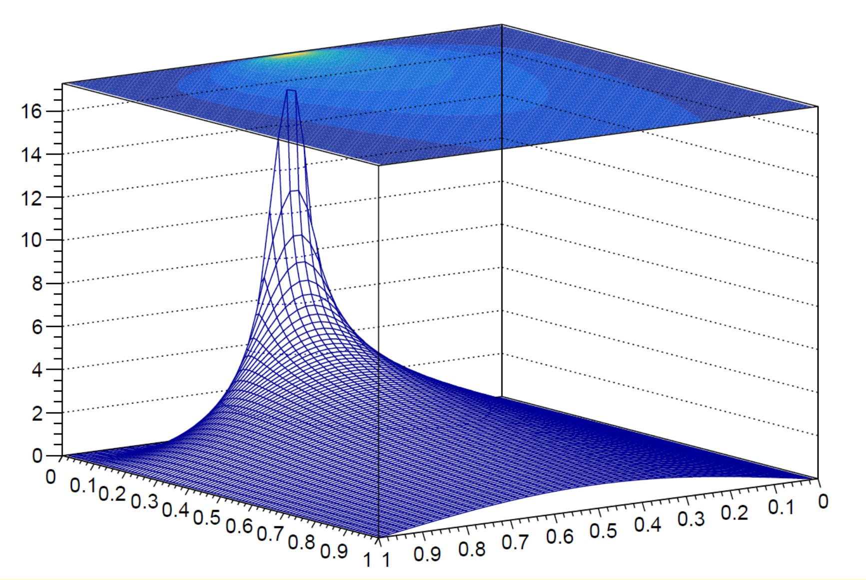
Θερμική διάχυση
ομοιόμορφη αρχική θερμοκρασία
άκρα σε μηδενική θερμοκρασία

Θερμική διάχυση και **ακτινοβολία** ομοιόμορφη αρχική θερμοκρασία
άκρα σε μηδενική θερμοκρασία



Θερμική διάχυση: Τροποποιημένη Μέθοδος FTCS

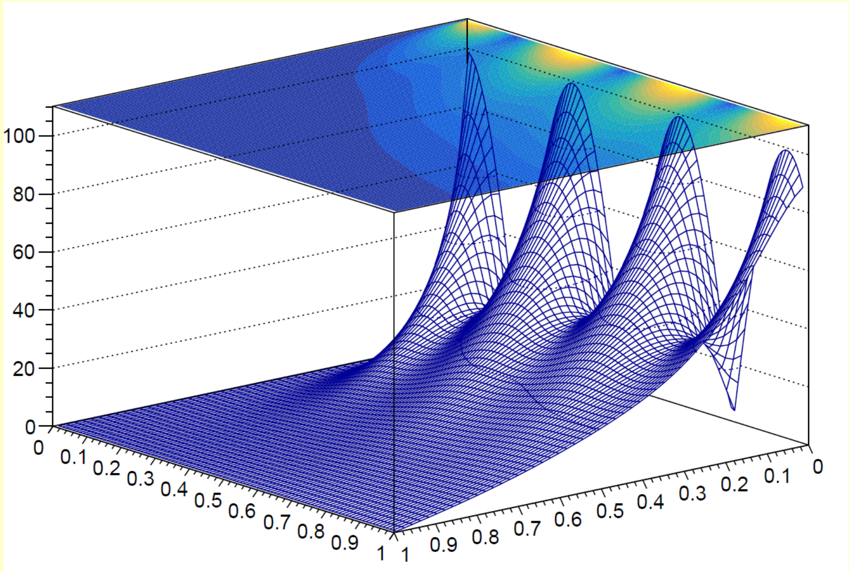
```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <TH1F.h>

void ftcs_m(double h, double dt)
{
  const int nx=1./h+1;
  TH2F *his=new TH2F("his"," ",nx, 0., 1.,100,0.,1.);
  double T[nx];
  double time=0.; int nt=10./dt;
  double k=1.;
  double ksi=k*dt/(h*h);
  cout <<ksi<<endl;
  // initial condition
  for(int i=0; i<nx; i++)T[i] = 0.;
  T[(nx-1)/2]=100.;
  // boundary conditions
  T[0]=0.; T[nx-1]=0.;
  for(int it=1;it<nt;it++)
    { time+=dt;
      for(int i=1; i<nx-1;i++) T[i]=T[i]+ksi*(T[i+1]+T[i-1]-2*T[i]);
      if(it%10==0){
      for(int i=0;i<nx;i++)his->SetBinContent(i+1,it/10+1,T[i]); }
    }
  his->Draw("surf3");
}
```



Αλλάζει η απαίτηση;

$\tau \leq h^2/2\kappa$

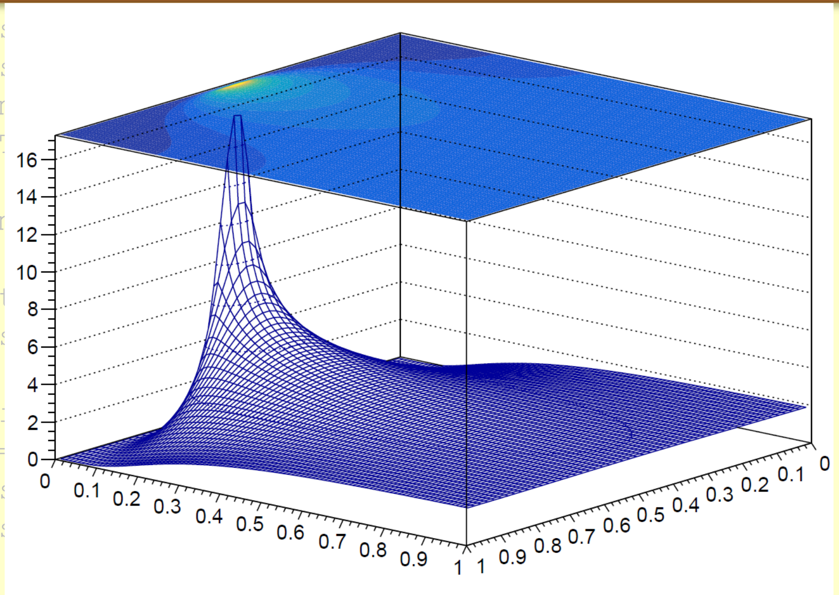
Θερμική διάχυση: Χρονο-εξαρτώμενες συνοριακές συνθήκες

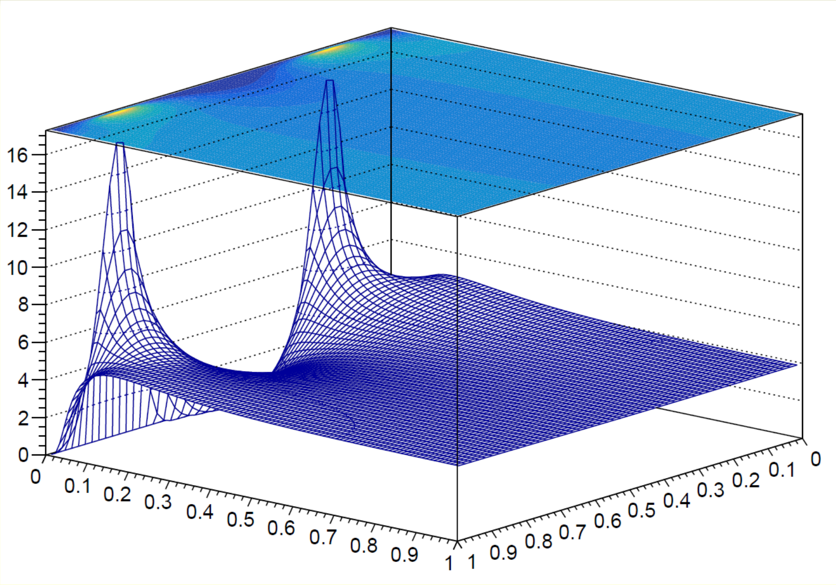



```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <TH1F.h>

void ftcs_m(double h, double dt)
{
  const int nx=1./h+1;
  TH2F *his=new TH2F("his"," ",nx, 0., 1.,100,0.,1.);
  double T[nx];
  double time=0.; int nt=10./dt;
  double k=1.;
  double ksi=k*dt/(h*h);
  cout <<ksi<<endl;
  // initial condition
  for(int i=0; i<nx; i++)T[i] = 0.;
  // boundary conditions
  T[0]=100.*abs(cos(100*time)); T[nx-1]=0.;
  for(int it=1;it<nt;it++)
    {
      time+=dt;
      T[0]=100.*abs(cos(100*time));
      for(int i=1; i<nx-1;i++) T[i]=T[i]+ksi*(T[i+1]+T[i-1]-2*T[i]);

     if(it%10==0){
     for(int i=0;i<nx;i++)his->SetBinContent(i+1,it/10+1,T[i]); }
    }
  his->Draw("surf3");
}
```

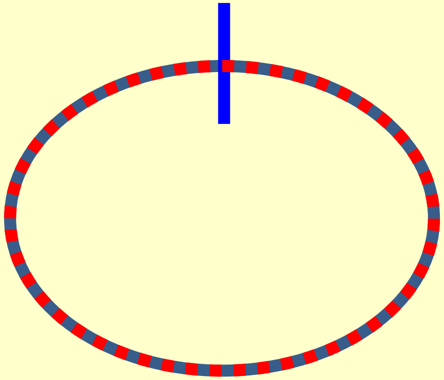
Θερμική διάχυση: Περιοδικές συνοριακές συνθήκες

```
#include <s
#include <s
#include <r
#include <T

void ftcs_r
{
  const int
  TH2F *his
  double T
  double ti
  double k=
  double ks
  cout <<ks
```





```
  // initial condition
  for(int i=0; i<nx; i++)T[i] = 0.;
  T[10]=100.; T[nx-10]=100.;
  for(int it=1;it<nt;it++)
    {
      time+=dt;
      // periodic boundary conditions
      T[0]=T[0]+ksi*(T[1]+T[nx-1]-2*T[0]);
      T[nx-1]=T[nx-1]+ksi*(T[0]+T[nx-2]-2*T[nx-1]);
      for(int i=1; i<nx-1;i++) T[i]=T[i]+ksi*(T[i+1]+T[i-1]-2*T[i]);

     if(it%10==0){
     for(int i=0;i<nx;i++)his->SetBinContent(i+1,it/10+1,T[i]); }
    }
  his->Draw("surf3");
}
```



Έστω για x=0

$$\frac{\partial u}{\partial x} = \alpha(t)u + g(t), \quad \alpha(t) \geq 0.$$

$$\frac{U_1^n - U_0^n}{\Delta x} = \alpha^n U_0^n + g^n$$

$$U_0^n = \beta^n U_1^n - \beta^n g^n \Delta x,$$

$$\beta^n = \frac{1}{1 + \alpha^n \Delta x}.$$

Κυματική εξίσωση

$$\frac{\partial^2}{\partial t^2}\mathrm{A}(x,t) = \mathrm{v}^2 \frac{\partial^2}{\partial x^2} A(x,t)$$

Με τους μετασχηματισμούς

$$r \equiv v\frac{\partial A}{\partial x} \qquad s \equiv \frac{\partial A}{\partial t}$$

$$\frac{\partial r}{\partial t} = v\frac{\partial s}{\partial x}$$

$$\frac{\partial s}{\partial t} = v\frac{\partial r}{\partial x}$$

Που σε διανυσματική γραφή

$$\frac{\partial \vec{A}}{\partial t} = -\frac{\partial \vec{F}(\vec{A})}{\partial x}$$

με

$$\vec{F}(\vec{A}) = \begin{pmatrix} 0 & -v \\ -v & 0 \end{pmatrix} \cdot \vec{A}$$

$$\frac{\partial a}{\partial t} = -v\frac{\partial a}{\partial x}$$

Εξίσωση συνέχειας

$$\frac{\partial a}{\partial t} = -v\frac{\partial a}{\partial x}$$

Μέθοδος FTCS:

$$a_i^{n+1} = a_i^n - \frac{v\tau}{2h}(a_{i+1}^n - a_{i-1}^n)$$

Αλλά στην περίπτωση αυτή:

$$\xi(k) = 1 - i\frac{v\tau}{h}\sin kh$$

Επομένως η μέθοδος FTCS είναι **ασταθής** ως προς την επίλυση της κυματικής εξίσωσης **ανεξαρτήτως συνθηκών**

$$a_i^{n+1} = a_i^n - \frac{v\tau}{2h}(a_{i+1}^n - a_{i-1}^n)$$

$$a_i^n \to \frac{1}{2}(a_{i+1}^n - a_{i-1}^n)$$

$$\boxed{a_i^{n+1} = \frac{1}{2}(a_{i+1}^n + a_{i-1}^n) - \frac{v\tau}{2h}(a_{i+1}^n - a_{i-1}^n)}$$

Μέθοδος Lax

Στη μέθοδο Lax:

$$\xi(k) = \cos kh - i\frac{v\tau}{h}\sin kh$$

Συνθήκη Ευστάθειας:

$$\boxed{\frac{|v|\tau}{h} \le 1}$$

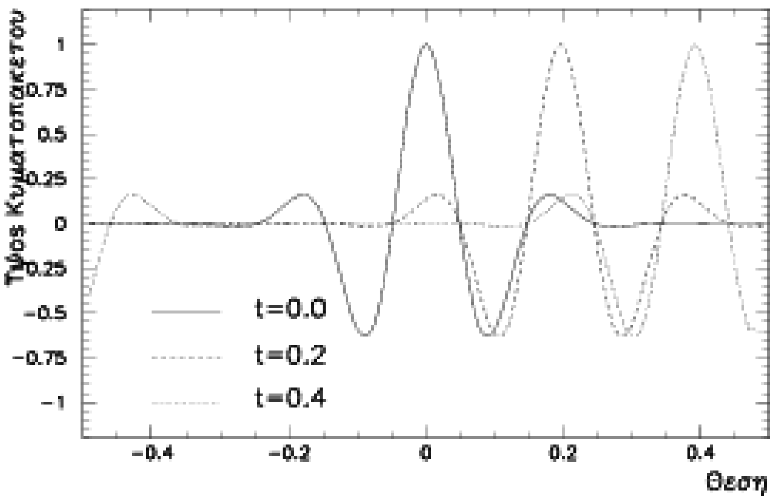
Η κυματική εξίσωση: Μέθοδος Lax

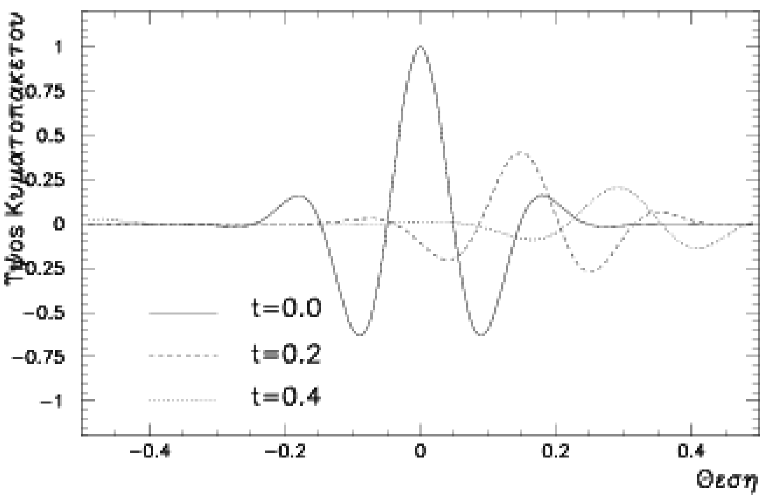
Όμως για τ μικρότερα από την ισότητα στη συνθήκη ευστάθειας έχουμε μείωση του κύματος, χωρίς φυσική αιτία !

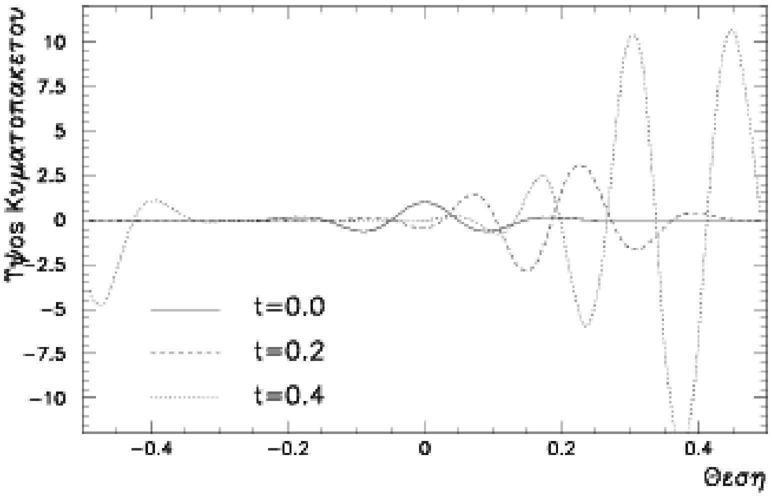
Ποιο είναι το κέρδος λοιπόν αν δεν χρησιμοποιήσουμε τ=h/v ;

Έχοντας λύση ευσταθή, αν και αφύσικα φθίνουσα,
μπορούμε να μελετήσουμε όλα τα λοιπά ποιοτικά
χαρακτηριστικά του προβλήματος
με εξαίρεση το μέγεθος του πλάτους της ταλάντωσης

Η κυματική εξίσωση: Μέθοδος Lax







Θεραπεία: Μέθοδος ανώτερης τάξης

$$\frac{\partial a}{\partial t} = -v\frac{\partial a}{\partial x} \Rightarrow a_j^{n+1} - a_j^{n-1} = -\frac{v\tau}{h}(a_{j+1}^n - a_{j-1}^n)$$

$$\Downarrow$$

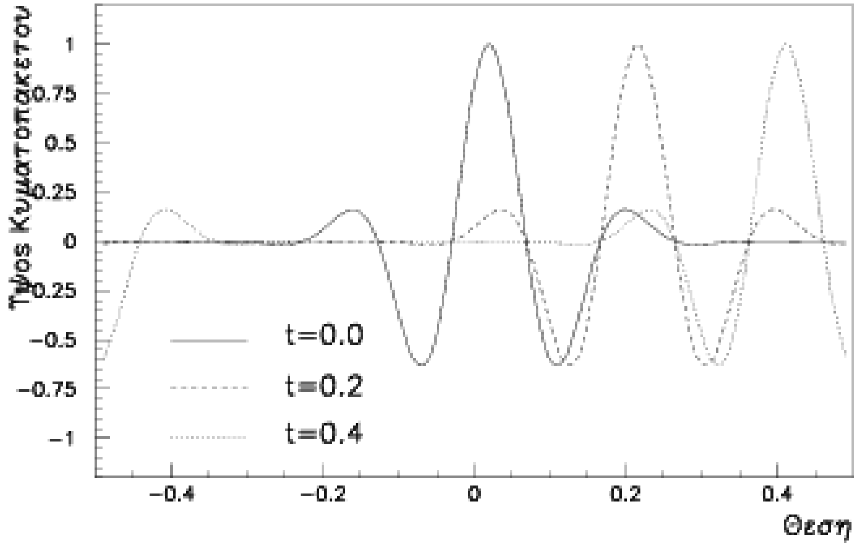
$$\xi^2 - 1 = -2i\xi\frac{v\tau}{h}\sin kh$$

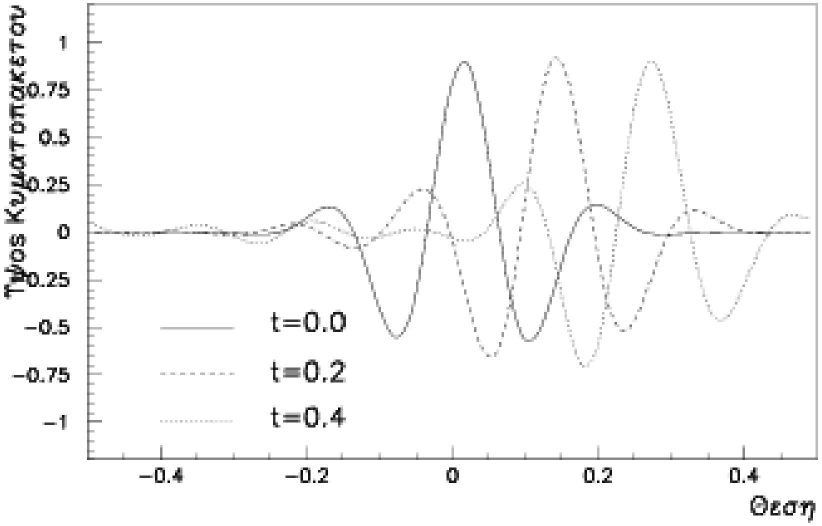
Και τελικά η **κυματική εξίσωση** γράφεται:

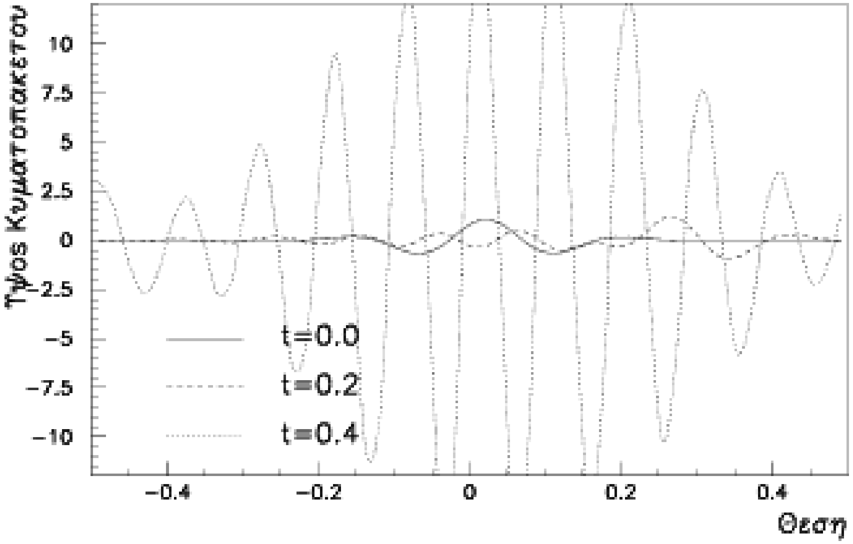
$$\frac{A_j^{n+1} - 2A_j^n + A_j^{n-1}}{\tau^2} = v^2\frac{A_{j+1}^n - 2A_j^n + A_{j-1}^n}{h^2}$$

Πολύ-βηματική μέθοδος

Η κυματική εξίσωση: Μέθοδος Staggered Leapfrog







```
       COEF = -V*DT/DX ! The coefficient used in leapfrog scheme
* Set Initial and Boundary conditions
      SIGMA = 0.1         ! Width of gaussian pulse
      KWAVE = PI/SIGMA    ! Wave number
      DO I = 1, NX
       X(I) = (-LX/2.)+(I-1)*DX  ! x-coordinates
* Initial state of a gaussian cosine pulse
       A_OLD(I) = DCOS(KWAVE*X(I))*DEXP(-0.5*X(I)**2/SIGMA**2)
       TIME = 0.
      ENDDO
* First time step (Two time steps needed in leapfrog method)
      DO I =2, NX-1
       A(I) = 0.5*(A_OLD(I+1)+A_OLD(I-1))
     &      + 0.5*COEF*(A_OLD(I+1)-A_OLD(I-1))
      ENDDO
      A(1) = 0.5*(A_OLD(2)+A_OLD(NX))
     &     + 0.5*COEF*(A_OLD(2)-A_OLD(NX))
      A(NX)= 0.5*(A_OLD(1)+A_OLD(NX-1))
     &     + 0.5*COEF*(A_OLD(1)-A_OLD(NX-1))
 DO 100 IT = 1, NSTEP
* Periodic boundary conditions
         A_NEW(1)=A_OLD(1)+COEF*(A(2)-A(NX))
* Use staggered Leapfrog relation
         DO 10 I = 2, NX-1
            A_NEW(I)=A_OLD(I)+COEF*(A(I+1)-A(I-1))
 10      CONTINUE
         A_NEW(NX)=A_OLD(NX)+COEF*(A(1)-A(NX-1))
```

Έμμεσες μέθοδοι (Implicit methods)

Πρόβλημα

- **Συνήθως ενδιαφερόμαστε να μελετήσουμε με ακρίβεια χαρακτηριστικά, με χωρικές κλίμακες $\lambda >> h$.**
- **Σ’ αυτήν την περίπτωση και με δεδομένη τη συνθήκη ευστάθειας, θα πρέπει να περιμένουμε χρονικά βήματα της τάξης του λ^2/h^2 πριν μπούμε στην περιοχή ενδιαφέροντος.**
- **Συνήθως τέτοιος αριθμός βημάτων είναι στην πράξη απαγορευτικός.**

Λύση

Χρήση ευσταθών μεθόδων: Έμμεσες μέθοδοι

$$\frac{\partial}{\partial t} T(x,t) = \kappa \frac{\partial^2}{\partial x^2} \; T(x,t)$$

Χωρικές παράγωγοι
Υπολογίζονται στο
Βήμα n+1

$$\frac{T_j^{n+1} - T_j^{n}}{\tau} = \kappa \frac{T_{j+1}^{n+1} + T_{j-1}^{n+1} - 2T_j^{n+1}}{h^2}$$

Η αναδρομική σχέση δεν μπορεί να επιλυθεί άμεσα για να υπολογιστούν οι τιμές πλέγματος του επόμενου βήματος.
Πρέπει να επιλυθεί ένα σύστημα γραμμικών εξισώσεων σε κάθε χρονικό βήμα.

Οπότε καταλήγουμε σε ένα τριδιαγώνιο γραμμικό σύστημα

$$\begin{vmatrix} 1+2\kappa\tau/h^2 & -\kappa\tau/h^2 & 0 & \dots & \dots & 0 \\ -\kappa\tau/h^2 & 1+2\kappa\tau/h^2 & -\kappa\tau/h^2 & 0 & \dots & 0 \\ 0 & -\kappa\tau/h^2 & 1+2\kappa\tau/h^2 & -\kappa\tau/h^2 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & \dots & -\kappa\tau/h^2 & 1+2\kappa\tau/h^2 \end{vmatrix} \mathbf{T^{n+1}} = \mathbf{T^{n}} + \begin{vmatrix} \text{bound} \\ \\ \\ \\ \text{bound} \end{vmatrix}$$

Αν το σύστημα είναι σχετικά μικρό ≤ 100 η επίλυση θα μπορούσε να γίνει με άμεση μέθοδο
Στην περίπτωση που το πλέγμα της ΜΔΕ είναι μεγαλύτερο προτείνεται η λύση με επαναληπτική μέθοδο

Θερμική διάχυση: Έμμεση μέθοδος

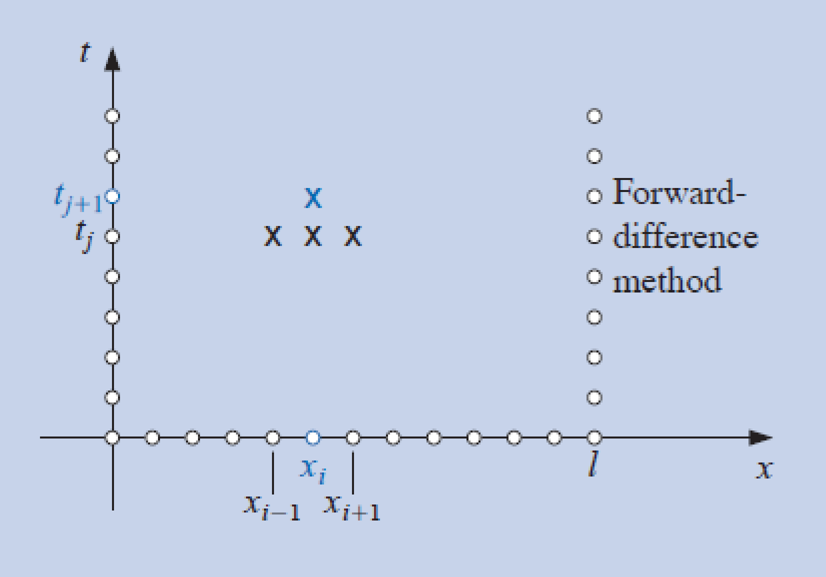
```
     DO 100 IT = 1, NSTEP
          DO 10 I = 2, NX
* form the linear tridiagonal system according to the equation
*   T_NEW(I)+2*KSI*T_NEW(I)-KSI*T_NEW(I+1)-KSI*T_NEW(I-1)=T(I)
           J=I-1
           A(J,J)=(1.+2.*KSI)
           A(J,NX)=T(I) ! value from previous step
           IF(J.NE.1)THEN
             A(J,J-1)=-1.*KSI
           ELSE     ! boundary 1
             A(J,NX) = A(J,NX)+KSI*0.
           ENDIF
           IF(J.NE.NX-1)THEN
             A(J,J+1)=-1.*KSI
           ELSE     ! boundary 2
             A(J,NX) = A(J,NX)+KSI*0.
           ENDIF
 10       CONTINUE
* Solve the system
          CALL SYSTEM(NX-1,A,T_NEW)
* Set the new values of temperature
          DO 20 I = 2, NX
             T(I)=T_NEW(I-1)
 20       CONTINUE
             TIME=IT*DT
 100      CONTINUE
```

Μέθοδος Crank-Nicolson

Άμεση μέθοδος

$$\frac{w_{i,j+1} - w_{i,j}}{k} - \alpha^2 \frac{w_{i+1,j} - 2w_{i,j} + w_{i-1,j}}{h^2} = 0,$$

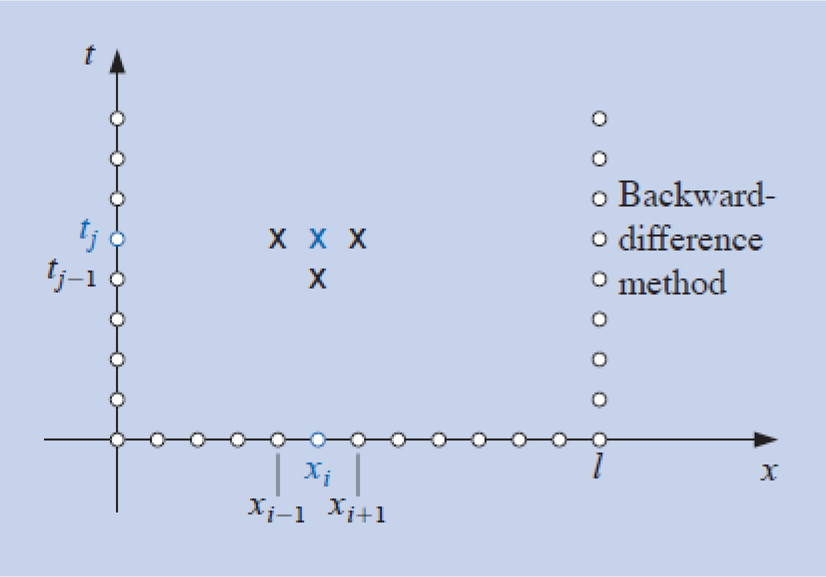
$$\tau_F = \frac{k}{2}\frac{\partial^2 u}{\partial t^2}(x_i, \mu_j) + O(h^2),$$



Έμμεση μέθοδος

$$\frac{w_{i,j+1} - w_{i,j}}{k} - \alpha^2 \frac{w_{i+1,j+1} - 2w_{i,j+1} + w_{i-1,j+1}}{h^2} = 0,$$

$$\tau_B = -\frac{k}{2}\frac{\partial^2 u}{\partial t^2}(x_i, \hat{\mu}_j) + O(h^2).$$



Και οι δύο έχουν ακρίβεια 1ης τάξης ως προς το χρόνο

Μέθοδος Crank-Nicolson

$$\frac{w_{i,j+1} - w_{i,j}}{k} - \alpha^2 \frac{w_{i+1,j} - 2w_{i,j} + w_{i-1,j}}{h^2} = 0,$$

$$\frac{w_{i,j+1} - w_{i,j}}{k} - \alpha^2 \frac{w_{i+1,j+1} - 2w_{i,j+1} + w_{i-1,j+1}}{h^2} = 0,$$

$$\tau_F = \frac{k}{2}\frac{\partial^2 u}{\partial t^2}(x_i, \mu_j) + O(h^2), \qquad \tau_B = -\frac{k}{2}\frac{\partial^2 u}{\partial t^2}(x_i, \hat{u}_j) + O(h^2).$$

Μέθοδος Crank-Nicolson που έχει ακρίβεια 2ης τάξης ως προς το χρόνο

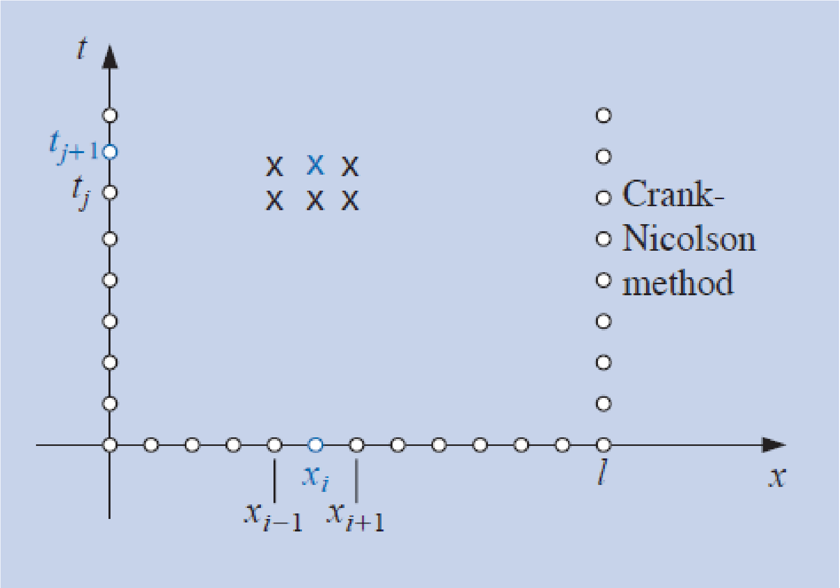
$$\boxed{\frac{w_{i,j+1} - w_{ij}}{k} - \frac{\alpha^2}{2}\left[\frac{w_{i+1,j} - 2w_{i,j} + w_{i-1,j}}{h^2} + \frac{w_{i+1,j+1} - 2w_{i,j+1} + w_{i-1,j+1}}{h^2}\right] = 0,}$$

$$T_j^{n+1} - \frac{\kappa\tau}{2h^2}\left(T_{j+1}^{n+1} + T_{j-1}^{n+1} - 2T_j^{n+1}\right) = T_j^n + \frac{\kappa\tau}{2h^2}\left(T_{j+1}^n + T_{j-1}^n - 2T_j^n\right)$$

$$\mathbf{AT}^{n+1} = \mathbf{BT}^n \qquad \lambda = \frac{\kappa\tau}{h^2}$$

$$A = \begin{bmatrix} (1+\lambda) & -\frac{\lambda}{2} & 0 & \cdots & 0 \\ -\frac{\lambda}{2} & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & -\frac{\lambda}{2} \\ 0 & \cdots & 0 & -\frac{\lambda}{2} & (1+\lambda) \end{bmatrix}$$

$$B = \begin{bmatrix} (1-\lambda) & \frac{\lambda}{2} & 0 & \cdots & 0 \\ \frac{\lambda}{2} & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \frac{\lambda}{2} \\ 0 & \cdots & 0 & \frac{\lambda}{2} & (1-\lambda) \end{bmatrix}.$$



$$\xi = \frac{1 - \frac{\kappa\tau}{h^2}\sin^2(kh/2)}{1 + \frac{\kappa\tau}{h^2}\sin^2(kh/2)}$$

Ευσταθής χωρίς όρους

$$i\frac{\partial\psi}{\partial t} = -\frac{\partial^2\psi}{\partial x^2} + V(x)\psi$$

Αρχική συνθήκη ψ(x,t=0) = f(x)
Συνοριακή συνθήκη ψ(±∞,t)=0

Ακρίβεια 1ης τάξης ως προς το χρόνο
Έμμεσο σχήμα για ευστάθεια

$$i\left(\frac{\psi_j^{n+1} - \psi_j^n}{\Delta t}\right) = -\left(\frac{\psi_{j+1}^{n+1} + \psi_{j-1}^{n+1} - 2\psi_j^{n+1}}{(\Delta x)^2}\right) + V_j\psi_j^{n+1}$$

Χρονοεξαρτώμενη εξίσωση Schröedinger

$$i\left(\frac{\psi_j^{n+1}-\psi_j^n}{\Delta t}\right) = -\left(\frac{\psi_{j+1}^{n+1}+\psi_{j-1}^{n+1}-2\psi_j^{n+1}}{(\Delta x)^2}\right)+V_j\psi_j^{n+1}$$

Πρόβλημα
Η λύση πρέπει να είναι μοναδιαία:

$$\int_{-\infty}^{\infty}|\psi|^2 dx = 1$$

Η αριθμητική λύση όμως δεν είναι!

Ας εκφράσουμε την εξίσωση με τη μορφή

$$i\frac{\partial\psi}{\partial t} = H\psi \quad H = -\frac{\partial^2}{\partial x^2} + V(x) \quad \Rightarrow \quad \psi(x,t) = \exp(-iHt)\psi(x,0)$$

FTCS: $$\psi_j^{n+1} = (1 - iH\Delta t)\psi_j^n$$

$$\psi_j^{n+1} = \psi_j^n + i\Delta t\left(\frac{\psi_{j+1}^n + \psi_{j-1}^n - 2\psi_j^n}{(\Delta x)^2}\right) - i\Delta t V_j \psi_j^n$$

Έμμεσο: $$\psi_j^{n+1} = (1 + iH\Delta t)^{-1}\psi_j^n$$

$$\psi_j^n = \psi_j^{n+1} - i\Delta t\left(\frac{\psi_{j+1}^{n+1} + \psi_{j-1}^{n+1} - 2\psi_j^{n+1}}{(\Delta x)^2}\right) + i\Delta t V_j \psi_j^{n+1}$$

Χρονοεξαρτώμενη εξίσωση Schröedinger – Crank-Nicolson

$$\psi(x,t) = \exp(-iHt)\psi(x,0)$$

Πιο σωστή προσέγγιση για το εκθετικό είναι:

$$\exp(-iHt) \approx \frac{1-\frac{1}{2}iH\Delta t}{1+\frac{1}{2}iH\Delta t} \quad \Rightarrow \quad \left(1+\frac{1}{2}iH\Delta t\right)\psi_j^{n+1} = \left(1-\frac{1}{2}iH\Delta t\right)\psi_j^n$$

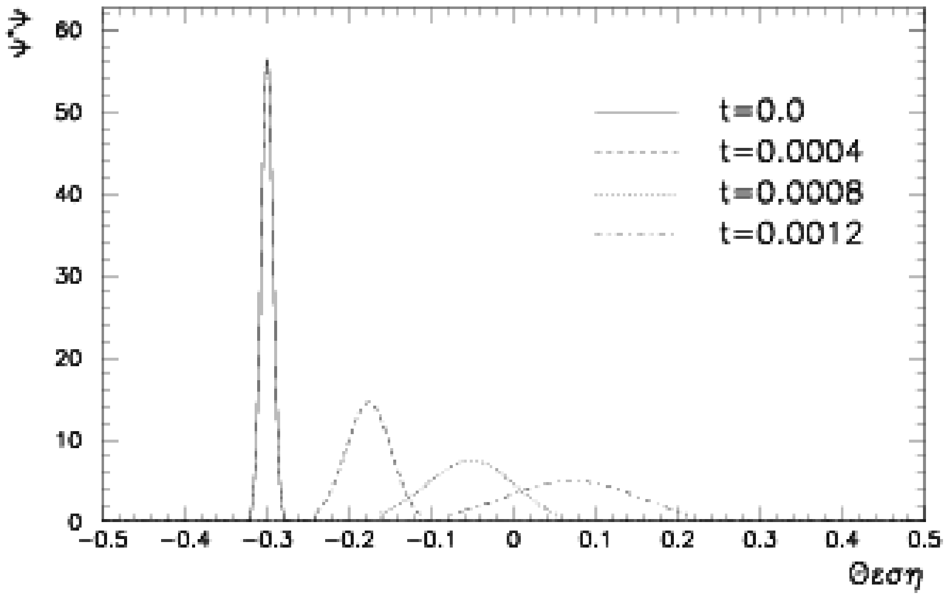
και είναι η μέθοδος Crank-Nicolson

Που είναι ακριβής σε 2η τάξη ως προς το χρόνο και μοναδιαία

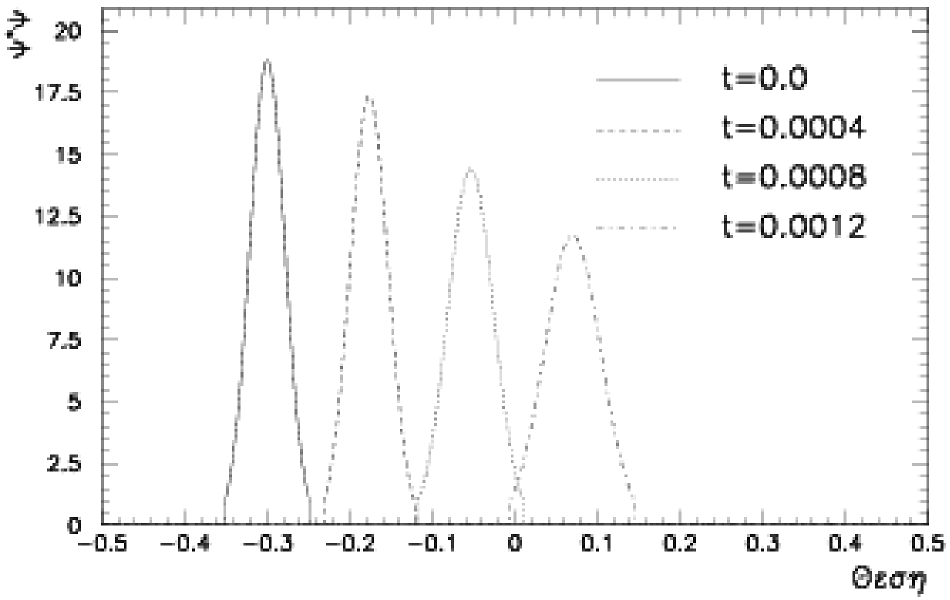
και οδηγεί στην εξίσωση διαφορών

$$\psi_{j+1}^{n+1} + \psi_{j-1}^{n+1} + (2i\lambda - 2h^2V_j - 2)\psi_j^{n+1} =$$
$$-\psi_{j+1}^{n} + \psi_{j-1}^{n} + (2i\lambda - 2h^2V_j - 2)\psi_j^{n}$$

όπου $\lambda = 2h^2/\tau$

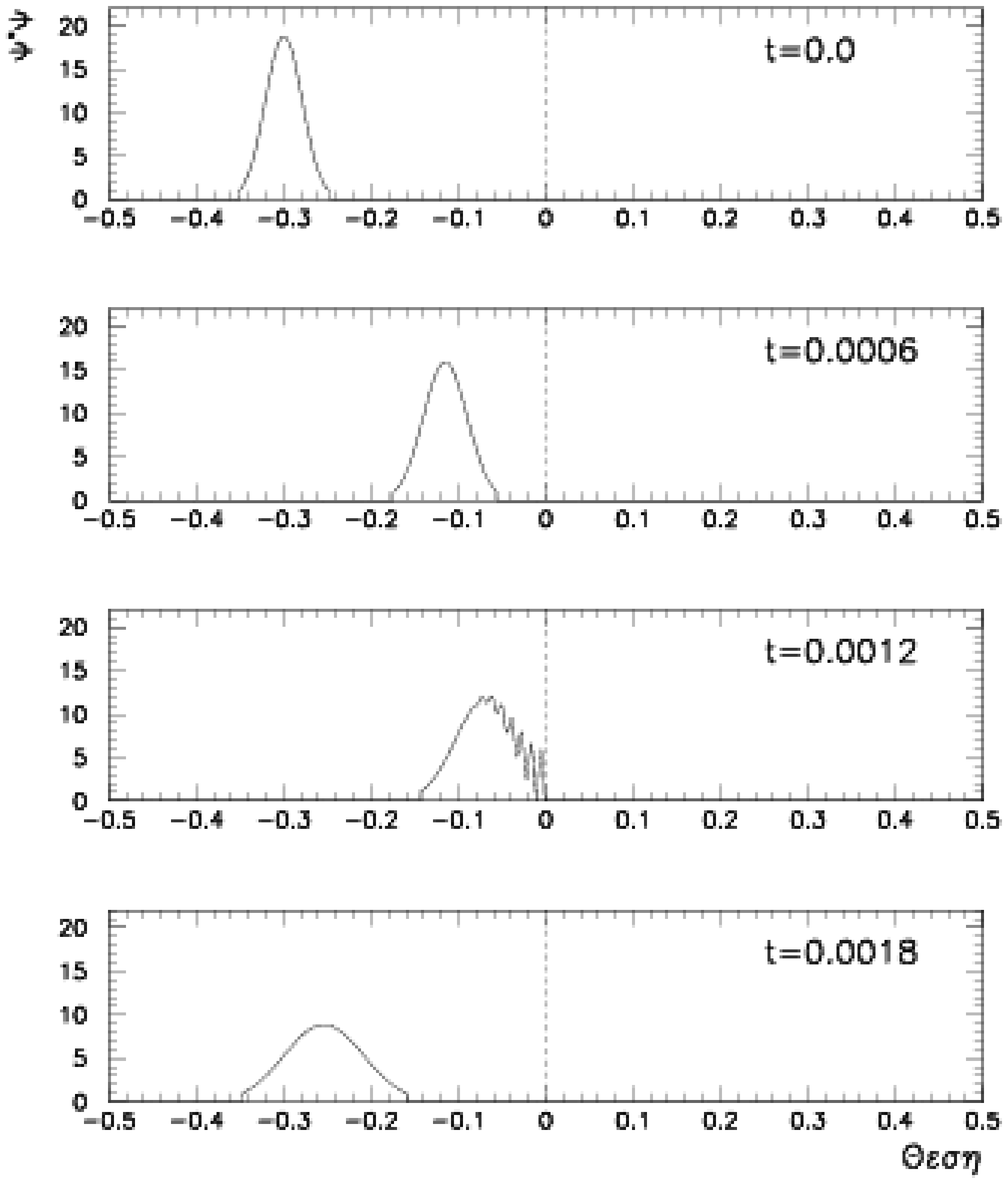


Πολύ εστιασμένο κυματοπακέτο

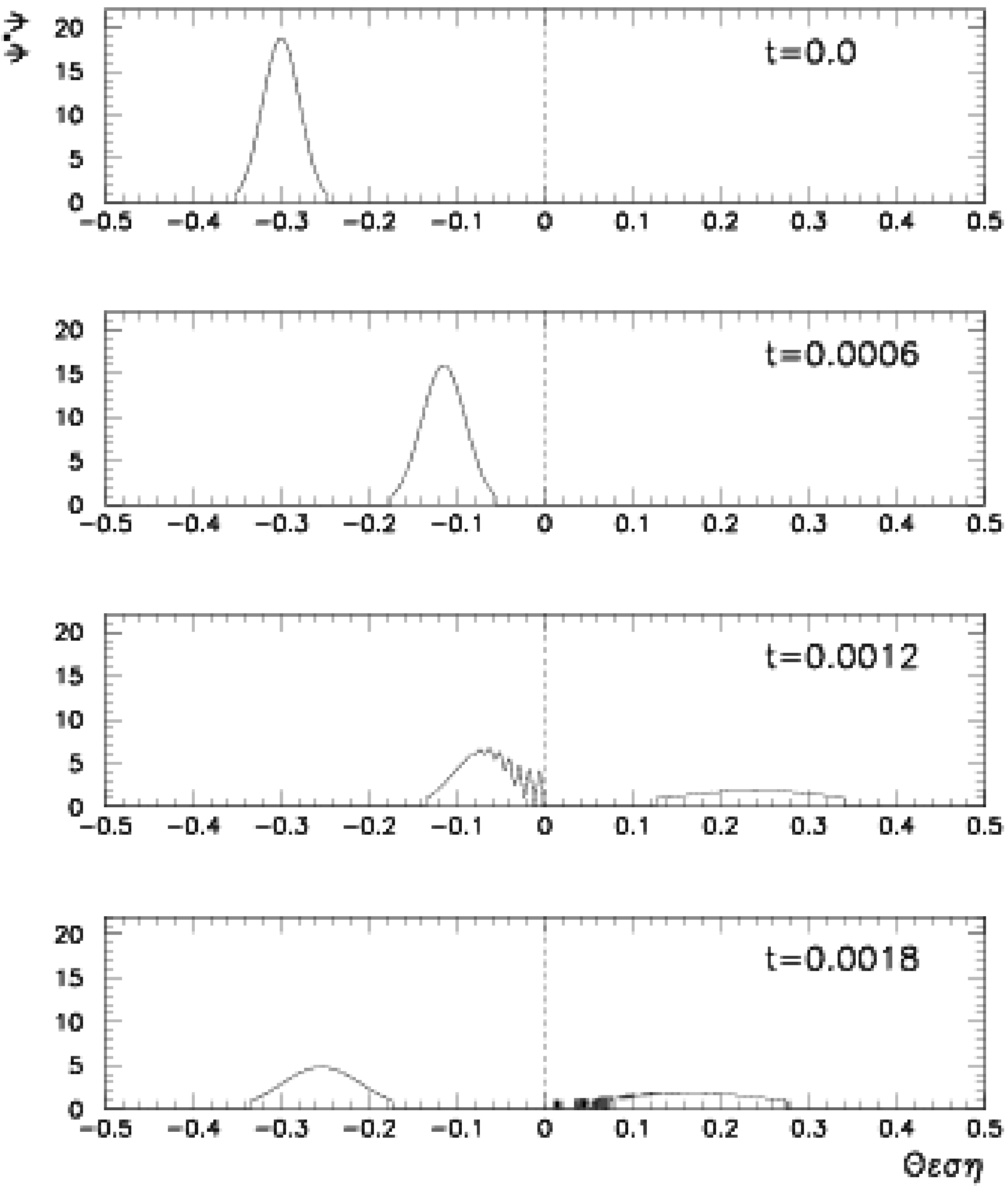


Λίγο εστιασμένο κυματοπακέτο

Παράδειγμα σκέδασης σε πολύ υψηλό θετικό δυναμικό



Παράδειγμα σκέδασης σε χαμηλό αρνητικό δυναμικό



Τα δύο είδη ταξινόμησης των ΜΔΕ:

Σύμφωνα με την υπολογιστική αντιμετώπισή τους

Προβλήματα αρχικών τιμών

Θερμική διάχυση

Κυματική εξίσωση

Προβλήματα συνοριακών τιμών

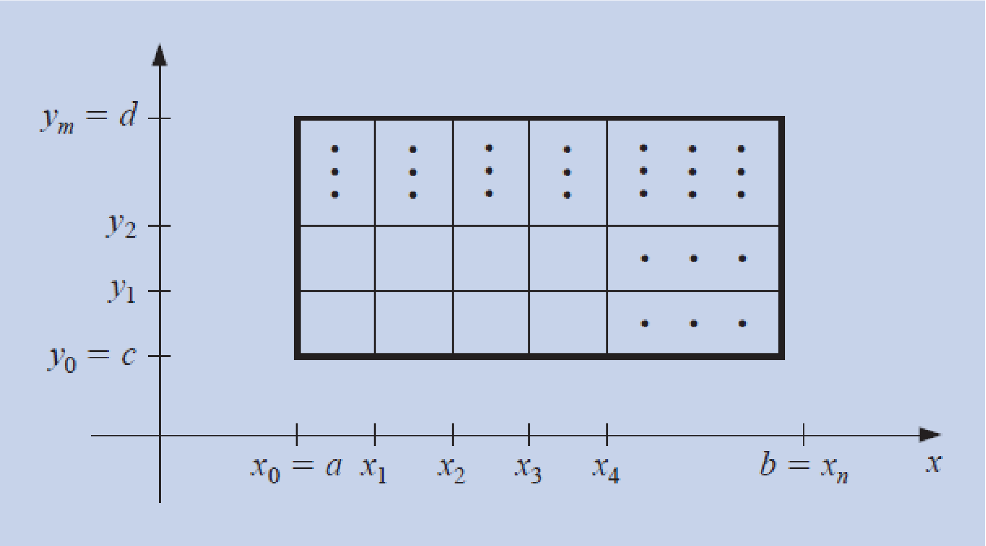
$$\frac{\partial^2}{\partial x^2}\Phi(x,y)+\frac{\partial^2}{\partial y^2}\Phi(x,y)=-\frac{1}{\varepsilon}\rho(x,y)$$

Εξίσωση Poisson

Εξίσωση Poisson
$$\frac{\partial^2}{\partial x^2}\Phi(x,y)+\frac{\partial^2}{\partial y^2}\Phi(x,y)=-\frac{1}{\varepsilon}\rho(x,y)$$

Εκφράζουμε τη συνάρτηση Φ(x,y) από τις τιμές της σε ένα διακριτό σύνολο σημείων πλέγματος

$$x_j = x_0 + jh, \quad j = 0,1,\ldots,J$$
$$y_k = y_0 + kh, \quad k = 0,1,\ldots,K$$



όπου h είναι η απόσταση των σημείων του πλέγματος.

Αν εισάγουμε την αναπαράσταση των πεπερασμένων διαφορών για τις δεύτερες παραγώγους έχουμε:

$$\frac{\Phi_{j+1,k}-2\Phi_{j,k}+\Phi_{j-1,k}}{h^2}+\frac{\Phi_{j,k+1}-2\Phi_{j,k}+\Phi_{j,k-1}}{h^2}=\rho_{j,k}$$

$$\Phi_{j+1,k} + \Phi_{j-1,k} + \Phi_{j,k+1} + \Phi_{j,k-1} - 4\Phi_{j,k} = h^2 \rho_{j,k}$$

Μετατρέποντας τις δύο διαστάσεις των σημείων του πλέγματος σε μια μονοδιάστατη σειρά

$$i \equiv j(K+1) + k \quad \gamma\iota\alpha \quad j = 0,1,\ldots,J \quad k = 0,1,\ldots,K$$

$$\Phi_{i+K+1} + \Phi_{i-K-1} + \Phi_{i+1} + \Phi_{i-1} - 4\Phi_{i} = h^2 \rho_{i}$$

Η εξίσωση αυτή ισχύει μόνο για τα σημεία που βρίσκονται στο εσωτερικό του πλέγματος.
Για όλα τα υπόλοιπα σημεία που βρίσκονται στο σύνορο, είτε η Φ είτε η παράγωγός της καθορίζεται από τις συνοριακές συνθήκες

A·Φ = b

Σύστημα NxN

Απευθείας Μέθοδοι

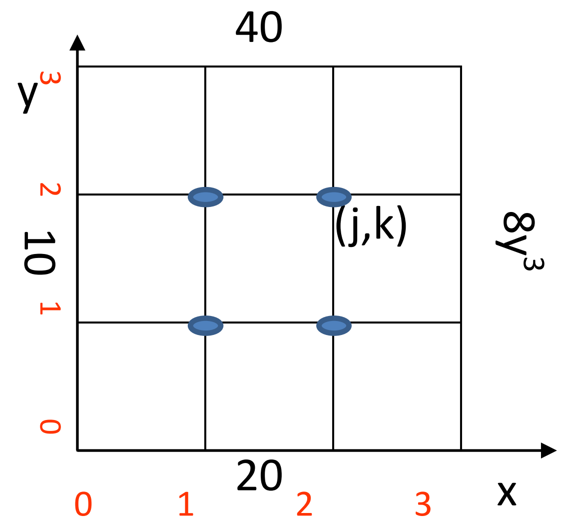
Μέθοδοι Εκτόνωσης

Μέθοδοι Φάσματος

$$\frac{\partial^2}{\partial x^2}\Phi(x,y)+\frac{\partial^2}{\partial y^2}\Phi(x,y)=10xy$$

$$\frac{\Phi_{j+1,k}-2\Phi_{j,k}+\Phi_{j-1,k}}{h^2}+\frac{\Phi_{j,k+1}-2\Phi_{j,k}+\Phi_{j,k-1}}{h^2}=10x_j y_k$$

$$\frac{1}{h^2}\left(\Phi_{j+1,k}+\Phi_{j-1,k}+\Phi_{j,k+1}+\Phi_{j,k-1}-4\Phi_{j,k}\right)=10x_j y_k$$



P_{11}: $\Phi_{01}+\Phi_{10}+\Phi_{21}+\Phi_{12}-4\Phi_{11}=10$
P_{21}: $\Phi_{11}+\Phi_{20}+\Phi_{31}+\Phi_{22}-4\Phi_{21}=20$
P_{12}: $\Phi_{02}+\Phi_{11}+\Phi_{22}+\Phi_{13}-4\Phi_{12}=20$
P_{22}: $\Phi_{12}+\Phi_{21}+\Phi_{32}+\Phi_{23}-4\Phi_{22}=40$

Με

$\Phi_{01}=\Phi_{02}=10$, $\Phi_{10}=\Phi_{20}=20$,
$\Phi_{13}=\Phi_{23}=40$, $\Phi_{31}=8$, $\Phi_{32}=64$

$\Phi_{21}+\Phi_{12}-4\Phi_{11} = -20$

$\Phi_{11}+\Phi_{22}-4\Phi_{21} = -8$

$\Phi_{11}+\Phi_{22}-4\Phi_{12} = -30$

$\Phi_{12}+\Phi_{21}-4\Phi_{22} = -64$

Ας υποθέσουμε ότι θέλουμε να λύσουμε την ελλειπτική εξίσωση
***L*u = ρ**
όπου *L* είναι ένας ελλειπτικός τελεστής και ρ οι πηγές.

Ξαναγράφουμε την εξίσωση ως:

$$\frac{\partial u}{\partial t} = Lu - \rho$$

π.χ. Laplace

$$\frac{\partial^2}{\partial x^2}\Phi(x,y) + \frac{\partial^2}{\partial y^2}\Phi(x,y) = 0$$

Διάχυση

$$\frac{\partial}{\partial t}T(x,y,t) = \kappa[\frac{\partial^2}{\partial x^2}T(x,y,t) + \frac{\partial^2}{\partial y^2}T(x,y,t)]$$

Για t→∞ Laplace

$$\frac{\partial^2}{\partial x^2}T_\infty(x,y) + \frac{\partial^2}{\partial y^2}T_\infty(x,y) = 0$$

Μέθοδος FTCS

$$\Phi_{i,j}^{n+1} = \Phi_{i,j}^{n} + \frac{k\tau}{h_x^2}(\Phi_{i+1,j}^{n} + \Phi_{i-1,j}^{n} - 2\Phi_{i,j}^{n}) + \frac{k\tau}{h_y^2}(\Phi_{i,j+1}^{n} + \Phi_{i,j-1}^{n} - 2\Phi_{i,j}^{n})$$

Η ευστάθεια απαιτεί

$$\frac{k\tau}{h_x^2} + \frac{k\tau}{h_y^2} \le \frac{1}{2} \qquad k\tau/h^2 \le 1/4$$

Θέλουμε το μεγαλύτερο χρονικό βήμα γιατί μας ενδιαφέρει η ασυμπτωτική λύση στο άπειρο

$$\Phi_{i,j}^{n+1} = \frac{1}{4}(\Phi_{i+1,j}^{n} + \Phi_{i-1,j}^{n} + \Phi_{i,j+1}^{n} + \Phi_{i,j-1}^{n})$$

Ο δείκτης n δε δηλώνει πλέον, πορεία του συστήματός μας στο φυσικό χρόνο, αλλά την πορεία της σύγκλισης της λύσης μας από μια «τυχαία» αρχική τιμή προς την πραγματική

Εξίσωση Poisson: Μέθοδοι εκτόνωσης

Μέθοδος Jacobi

$$\Phi_{i,j}^{n+1} = \frac{1}{4}(\Phi_{i+1,j}^{n} + \Phi_{i-1,j}^{n} + \Phi_{i,j+1}^{n} + \Phi_{i,j-1}^{n})$$

Μέθοδος Gauss-Seidel

$$\Phi_{i,j}^{n+1} = \frac{1}{4}\underbrace{(\Phi_{i+1,j}^{n} + \Phi_{i-1,j}^{n+1} + \Phi_{i,j+1}^{n} + \Phi_{i,j-1}^{n+1})}$$

Over-Relaxation
με 1<w<2

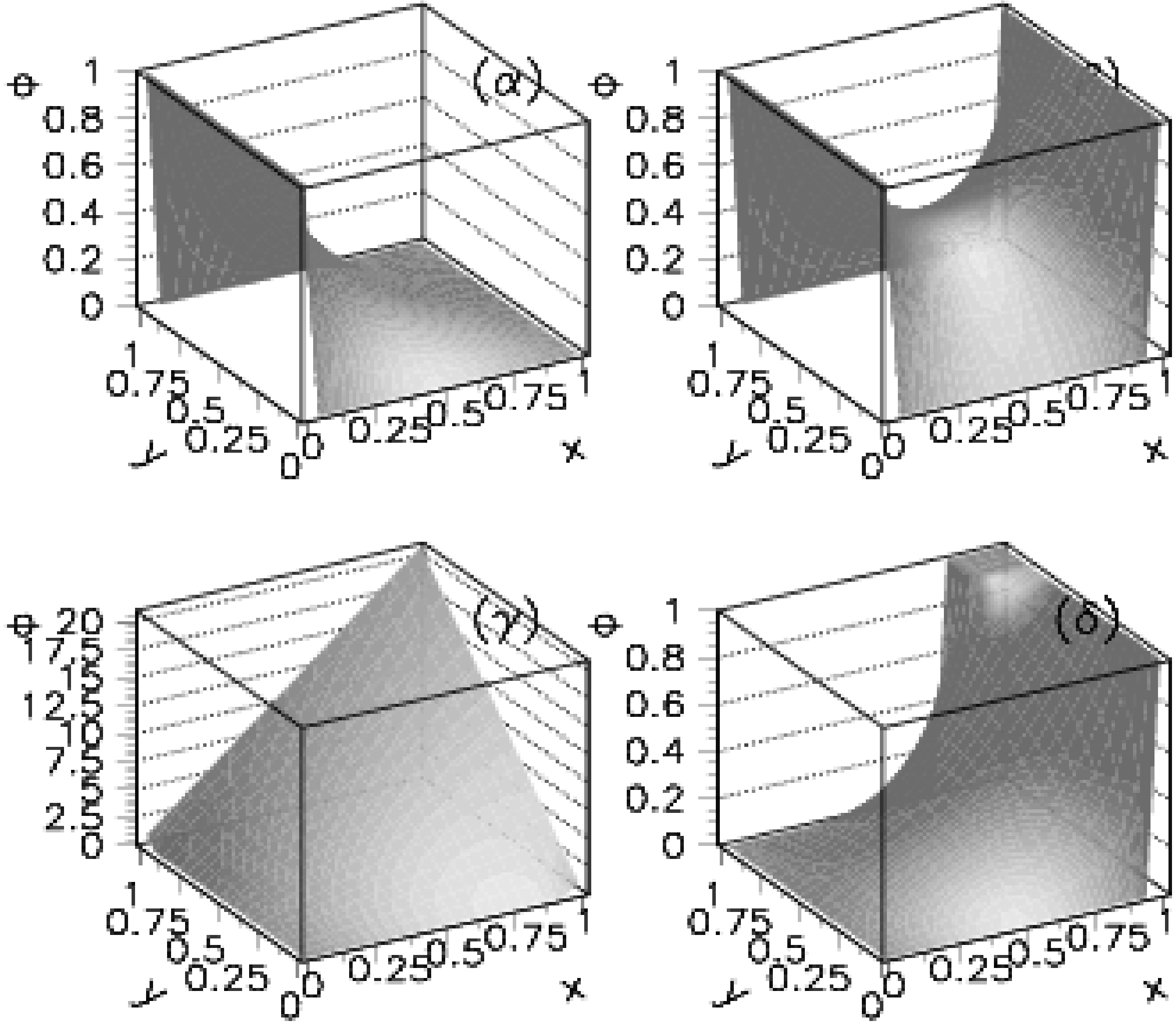
$$\Phi_{i,j}^{n+1} = w\Phi_{ave} + (w-1)\Phi_{i,j}^{n}$$

Αλγόριθμος Μεθόδου

- Ορίζουμε το πλέγμα
- Εφαρμόζουμε τις συνοριακές συνθήκες για τα εξωτερικά σημεία
- Δίνουμε την αρχική τιμή του δυναμικού στα εσωτερικά σημεία του πλέγματος
- Εκτελούμε επαναλήψεις προσδιορίζοντας την τιμή του δυναμικού στα εσωτερικά σημεία μέχρι την ικανοποίηση κάποιου κριτηρίου σύγκλισης

```
*  Set Initial and Boundary conditions
      DO I = 1, NX+1
       X(I)  = (I-1)*DX
       DO J = 1, NY+1
        Y(J)  = (J-1)*DY
        RO(I,J)=0.       ! Charge density
        IF(I.EQ.1)THEN
          FFI(I,J)  = 1. ! Boundary condition at X=0.
        ELSE
          FFI(I,J)  = 0. ! Initial value of F=0 everywhere
        ENDIF
       ENDDO
      ENDDO
      DO 100 IT = 1, MAXSTEP ! Loop over the desired steps
        DO 20 I = 2,NX
             DO 10 J = 2,NY
               FNEW(I,J)=
     &  0.25*(FFI(I+1,J)+FFI(I-1,J)+FFI(I,J+1)+FFI(I,J-1))
     &  +DX*DY*RO(I,J) ! charge density
  10      CONTINUE
  20    CONTINUE
  100 CONTINUE
```

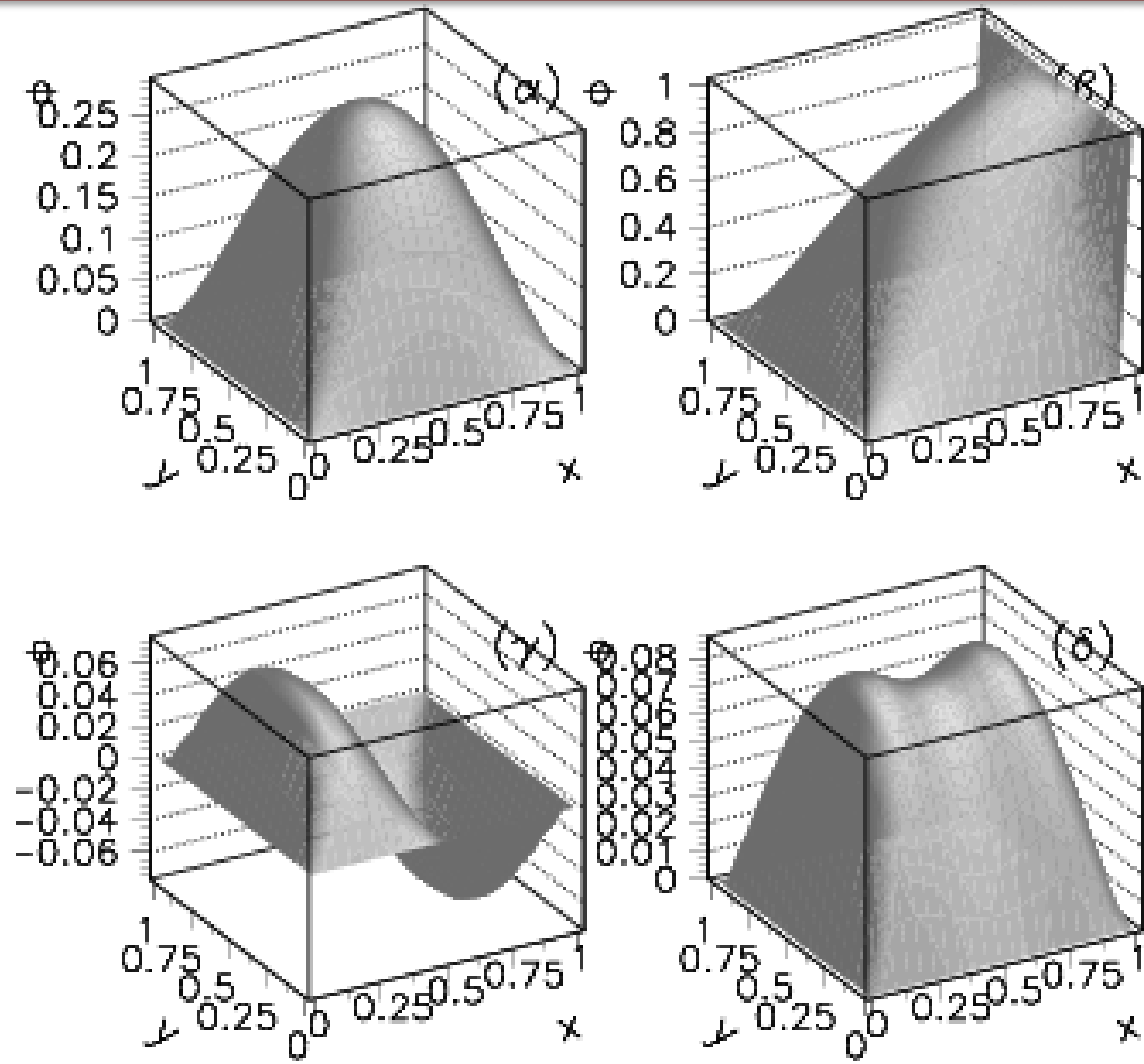
Εξίσωση Poisson: Μέθοδοι εκτόνωσης



Εξίσωση Laplace, σε δύο διαστάσεις, με τη χρήση της μεθόδου εκτόνωσης

α) Δυναμικό Φ=1 για x=0 και Φ=0 αλλού. β) Δυναμικό Φ=1 για x=0 και x=L και Φ=0 αλλού. γ) Γραμμικό δυναμικό για x=L και y=L. δ) Δυναμικό Φ=1 για x=L και επιπλέον σε μία γωνία του χώρου.

Εξίσωση Poisson: Μέθοδοι εκτόνωσης

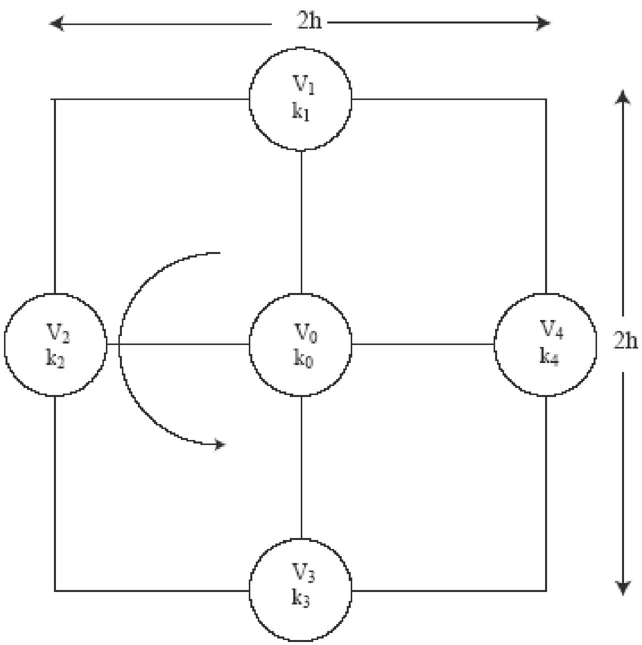


Εξίσωση Poisson, σε δύο διαστάσεις, με τη χρήση της μεθόδου εκτόνωσης

α) Δυναμικό Φ=0 στο σύνορο και ομοιόμορφη κατανομή φορτίου

β) Δυναμικό Φ=1 για x=Λ και x=L και Φ=0 αλλού και ομοιόμορφη κατανομή φορτίου.

γ) δ) Δυναμικό Φ=0 στο σύνορο και ημιτονοειδής πυκνότητα φορτίου



Τα διηλεκτρικά μπορούν να συμπεριληφθούν στην αριθμητική λύση χρησιμοποιώντας την τιμή τους σε κάθε σημείο του πλέγματος και απαιτώντας:

$D_{1n}=D_{2n}$ ή

$$0 = \oint_l k\nabla V \cdot d\mathbf{l} = \oint_l k\frac{\partial V}{\partial n}dl,$$

$$0 = k_1\frac{V_1 - V_0}{h}2h + k_2\frac{V_2 - V_0}{h}2h + k_3\frac{V_3 - V_0}{h}2h + k_4\frac{V_4 - V_0}{h}2h$$

$$V_0 = \frac{1}{4(k_1 + k_2 + k_3 + k_4)}[k_1V_1 + k_2V_2 + k_3V_3 + k_4V_4]$$

Εξίσωση Laplace: Μέθοδοι εκτόνωσης
Εφαρμογή σε κυματοδηγούς

Επίλυση της κυματικής εξίσωσης:

Όπου $\Phi=E_Z$ για ΕΗ κύματα

$\Phi=B_Z$ για ΕΜ κύματα

$$\nabla^2\Phi + k^2\Phi = 0$$

$$k^2 = \omega^2\mu\varepsilon - \beta^2$$

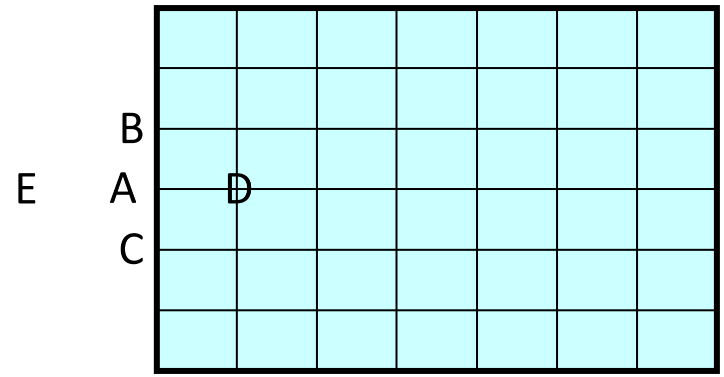
Για να εφαρμόσουμε τη μέθοδο πεπερασμένων διαφορών διακριτοποιούμε την τομή του κυματοδηγού και οδηγούμαστε στην:

$\Phi(i+1,j)+ \Phi(i-1,j)+ \Phi(i,j+1)+ \Phi(i,j-1)=(4-h^2k^2)\Phi(i,j)$, $h=\Delta x=\Delta y$

Συνοριακές Συνθήκες

Για ΕΜ κύματα

Συνθήκη Dirichlet: **$\Phi_A=0$**



Για ΕΗ κύματα

Συνθήκη Neumann: $\frac{\partial\Phi}{\partial n}|_A = 0, \quad \Phi_E = \Phi_D$

$\Phi_B+ \Phi_C+ 2\Phi_D-(4-h^2k^2)\Phi_A=0$

Επίλυση προβλήματος ιδιοτιμών

$A\Phi=\lambda\Phi$

Έμπνευση από αναλυτικές τεχνικές

Λύση ως άπειρο άθροισμα συναρτήσεων βάσης

Προσεγγιστική λύση της μορφής

$$\begin{aligned}\Phi(\mathrm{x},\mathrm{y}) &= \Phi_\alpha(\mathrm{x},\mathrm{y}) + \mathrm{E}(\mathrm{x},\mathrm{y}) \\ &= \sum_{k=1}^{K} \alpha_\mathrm{k} \mathrm{f}_\mathrm{k}(\mathrm{x},\mathrm{y}) + \mathrm{E}(\mathrm{x},\mathrm{y})\end{aligned}$$

π.χ. Εισάγοντας την παραπάνω λύση στην εξίσωση Poisson

$$\nabla^2\left[\sum_k \alpha_k f_k(x,y)\right] + \frac{1}{\varepsilon_0}\rho(x,y) = R(x,y)$$

Σκοπός μας είναι
η ελαχιστοποίηση του υπολοίπου

$$R(x,y) = -\nabla^2 E(x,y)$$

Η μέθοδος **Galerkin**, απαιτεί η συνάρτηση του υπολοίπου να είναι ορθογώνια με όλες τις δοκιμαστικές συναρτήσεις.

$$\int_0^L dx \int_0^L dy f_k(x,y) R(x,y) = 0$$

Έστω καρτεσιανό πρόβλημα με συνοριακές συνθήκες τύπου Neumann

$$f_{m,n}(x,y) = \cos\left(m\pi x / L\right)\cos\left(n\pi y / L\right)$$

με m,n=0,1,…,M.

Η λύση μας θα δίνεται από μια έκφραση της μορφής:

$$\Phi_\alpha(x,y) = \sum_{m=0}^{M}\sum_{n=0}^{M} \alpha_{m,n} \cos(m\pi x / L) \cos(n\pi y / L)$$

Όπου πρέπει να υπολογίσουμε τους συντελεστές $\boldsymbol{\alpha}_{m,n}$

Εισάγωντας τη λύση στην εξίσωση Poisson θα έχουμε:

$$-\sum_{m=0}^{M}\sum_{n=0}^{M}\alpha_{m,n}(m^2+n^2)\frac{\pi^2}{L^2}f_{m.n}(x,y)+\frac{1}{\varepsilon_0}\rho(x,y)=R(x,y)$$

Εφαρμόζοντας στη συνέχεια και στα δύο μέρη

$$\int_0^L dx\int_0^L dy f_{m',n'}(x,y)$$

και χρησιμοποιώντας τις σχέσεις ορθογωνιότητας

$$\alpha_{m,n}=\frac{4}{\pi^2\varepsilon_0(m^2+n^2)(1+\delta_{m,0})(1+\delta_{n,0})}\int_0^L dx\int_0^L dy\rho(x,y)\cos(m\pi x/L)\cos(n\pi y/L)$$

- έκφραση της λύσης σαν ένα πεπερασμένο άθροισμα ορθογωνίων συναρτήσεων
- σχεδιάζονται με σκοπό την επίλυση με πολύπλοκες συνοριακές συνθήκες
- οι συναρτήσεις αυτές είναι τοπικές

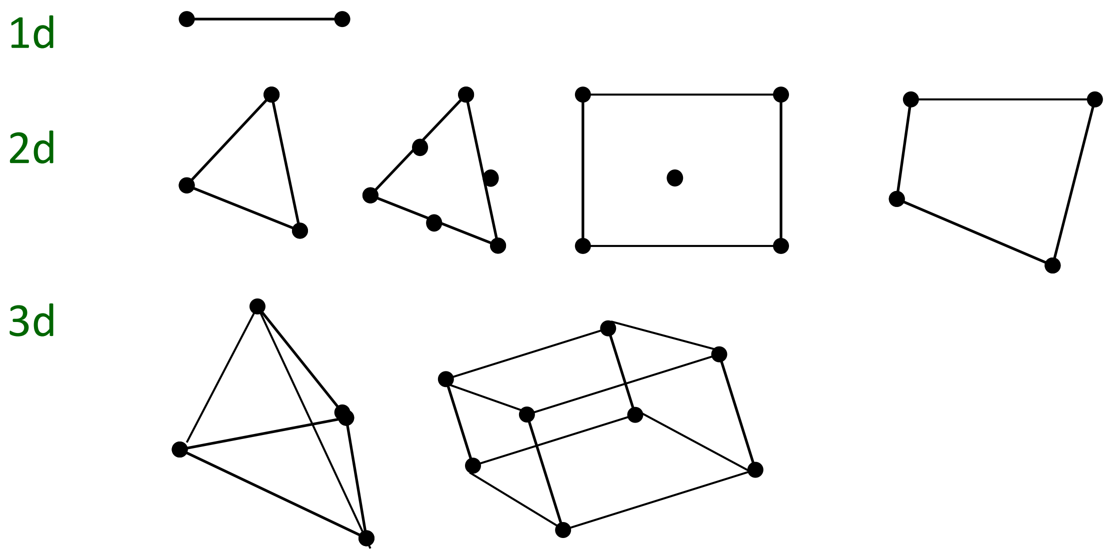
ορίζονται δηλαδή μόνο
στην περιοχή γύρω από
κάποιο σημείο του πλέγματος

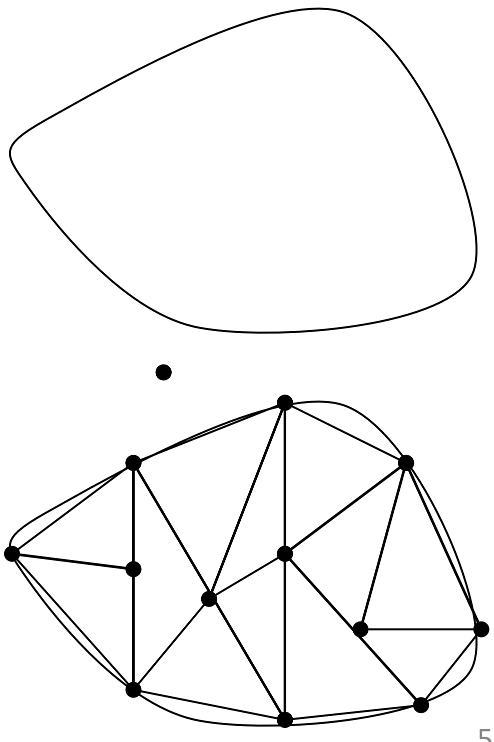
•

Προβλήματα συνοριακών τιμών
Μέθοδοι Πεπερασμένων στοιχείων

Ο συνεχής χώρος στον οποίο αναζητούμε τη λύση χωρίζεται σε υποπεριοχές που ονομάζονται πεπερασμένα στοιχεία.

Παραδείγματα πεπερασμένων στοιχείων σε 1,2 και 3 διαστάσεις





Διαχωρισμός περιοχής σε Πεπερασμένα στοιχεία

π.χ. στη μονοδιάστατη εξίσωση Poisson
Αν εισάγουμε λύση της μορφής

$$\Phi_n(x) = \sum_{i=1}^{n} \alpha_i \mathrm{f}_i(\mathrm{x})$$

Θα έχουμε ένα υπόλοιπο από την πραγματική λύση

$$r_n(x) = \frac{d^2\Phi_n(x)}{dx^2} + \frac{\rho(x)}{\varepsilon_0}$$

Ο σκοπός στη συνέχεια είναι να επιλέξουμε ένα σχήμα που κατά την επιλογή των α_i, θα ελαχιστοποιούσε τα υπόλοιπα r_n, σε όλο το χώρο

Συνήθως επιλέγονται πολυωνυμικές συναρτήσεις για τα f_i, π.χ.:
1d $f_i(x)=a+bx$
2d τριγωνικό στοιχείο: $f_i(x,y)=a+bx+cy$
2d τετράπλευρο στοιχείο: $f_i(x,y)=a+bx+cy+dxy$ κ.ο.κ

Η περιοχή ενδιαφέροντος διαιρείται σε πολλά μικρά μέρη, συνήθως της ίδιας τοπολογίας (για να διευκολύνονται οι υπολογισμοί), αλλά όχι απαραίτητα του ίδιου μεγέθους

Επιλέγονται οι τοπικές συναρτήσεις f_i και οι συναρτήσεις βάρους w_i με σκοπό το μηδενισμό του ολοκληρώματος:

$$g_i = \int_0^1 r_n(x) w_i(x) dx$$

παντού στο χώρο

Αν $w_i = f_i$, τότε έχουμε το λεγόμενο σχήμα Galerkin

Στη μονοδιάστατη
Εξίσωση Poisson

$$g_i = \int_0^1 r_n(x) w_i(x) dx$$

$$\Rightarrow \quad g_i = \int_0^1 \left(\sum_{j=1}^{n} a_j f_j^{//} + \rho(x)/\varepsilon_0 \right) w_i(x) dx = 0$$

$\Rightarrow$ **Aα=b**

$$A_{i,j} = -\int_0^1 f_i^{//}(x) w_j(x) dx$$

$$b_i = \frac{1}{\varepsilon_0} \int_0^1 \rho(x) w_i(x) dx$$

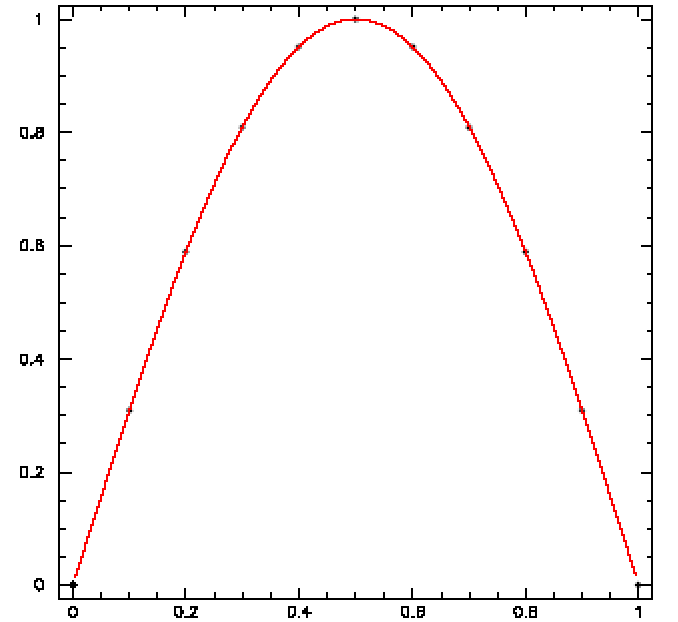
Έστω ότι χρησιμοποιούμε τοπικές συναρτήσεις και αντίστοιχα βάρη της μορφής

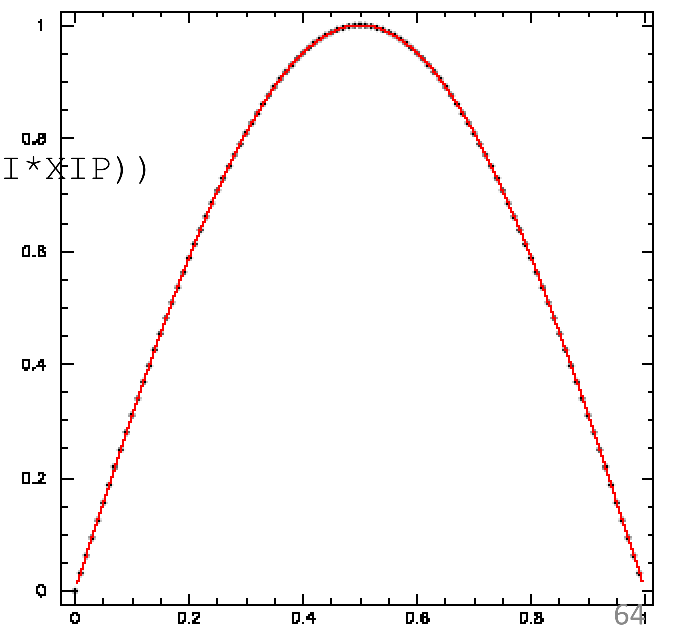
$$f_i(x) = \begin{cases} (x - x_{i-1})/h & x_{i-1} < x < x_i \\ (x_{i+1} - x)/h & x_i < x < x_{i+1} \\ 0 & \alpha\lambda\lambda o\upsilon \end{cases} \Rightarrow A_{ij} = \int_0^1 f_i'(x) f_j'(x) = \begin{cases} 2/h & i = j \\ -1/h & i = j \pm 1 \\ 0 & \alpha\lambda\lambda o\upsilon \end{cases}$$

Έστω ότι η πυκνότητα φορτίου στο πρόβλημά μας είναι αντίστοιχα της μορφής

$$\rho(x) = \frac{\pi^2}{\varepsilon} \sin \pi x \Rightarrow \begin{aligned} b_i &= \frac{1}{\varepsilon}\int_0^1 \rho(x) f_i(x) dx_i \\ &= \frac{\pi}{h}\left(x_{i-1} + x_{i+1} - 2x_i\right)\cos \pi x_i \\ &+ \frac{1}{h}\left(2\sin \pi x_i - \sin \pi x_{i-1} - \sin \pi x_{i+1}\right) \end{aligned}$$

```
      PI   =  4.0*ATAN(1.0)
      XL   =  1.0
      H    =  XL/(N+1)
      D    =  2.0/H
      E    = -1.0/H
      B0   =  PI/H
      B1   =  1.0/H
C Find the elements in L and U
      W(1) =  D
      U(1) =  E/D
      DO      100 I = 2, N
        W(I) = D-E*U(I-1)
        U(I) = E/W(I)
  100 CONTINUE
C Assign the array B
      DO      200 I = 1, N
        XIM  = H*(I-1)
        XI   = H*I
        XIP  = H*(I+1)
        B(I) = B0*COS(PI*XD)*(XIM+XIP-2.0*XI)
     *         +B1*(2.0*SIN(PI*XI)-SIN(PI*XIM)-SIN(PI*XIP))
  200 CONTINUE
C Find the solution
      Y(1) = B(1)/W(1)
      DO     300 I = 2, N
        Y(I) = (B(I)-E*Y(I-1))/W(I)
  300 CONTINUE
C
      A(N) = Y(N)
      DO     400 I = N-1,1,-1
        A(I) = Y(I)-U(I)*A(I+1)
  400 CONTINUE
```





Εξίσωση Poisson
Μέθοδοι Πεπερασμένων στοιχείων

Αλλαγή συνοριακών συνθηκών, π.χ. $\Phi(0)=\Phi_0$ και $\Phi(1)=\Phi_1$

$$\Phi_n(x) = (1-x)\Phi_0 + x\Phi_1 + \sum_{i=1}^{n} a_i f_i(x)$$

Θέτουμε τους δύο πρώτους όρους για να ικανοποιήσουν τις συνοριακές συνθήκες, δεδομένου ότι το άθροισμα μηδενίζεται στο σύνορο.

… αντίστοιχα για συνοριακές συνθήκες τύπου Neumann

Εξίσωση Poisson: Μέθοδοι πεπερασμένων διαφορών