Aluno: João Victor Alcantara Pimenta 11820812

Professor: Francisco Castilho Alcaraz

1 Prolegômenos

Ondas unidimensionais são descritas pela equação bem conhecida

$$\frac{\partial^2 y}{\partial t^2} = c^2 \frac{\partial^2 y}{\partial x^2}$$

Onde c é a velocidade da onda no meio de propagação. Se desejarmos discretizar esta onda podemos reescrever as derivadas de forma a ter a equação

$$\frac{y(i,n+i) + y(i,n-1) - 2y(i,n)}{(\Delta t)^2} \approx c^2 \frac{y(i+1,n) + y(i-1,n) - 2y(i,n)}{(\Delta x)^2} + O((\Delta t)^2) + O\left(\frac{10^{-8}}{(\Delta t)^2}\right)$$

Temos $x = i\Delta x$ e $t = n\Delta t$. Note que ambas ordens de erro das derivadas devem ser equivalentes, de forma que manteremos a representação somente da espacial. A outra fonte de erro é devida à aproximação feita na precisão números usados. para efeitos deste relatório, usaremos reais de precisão 8.

Explicitando o termo de interesse:

$$y(i,n+1) \approx 2(1-r^2)y(i,n) + r^2[y(i+1,n) + y(i-1,n)] - y(i,n-1) + O\left(\frac{10^{-8}}{(\Delta x)^2} + (\Delta x)^2\right)$$

Onde definimos $r = \frac{c\Delta t}{\Delta x}$ uma grandeza adimensional.

1.1 Minimização da ordem do erro

Para nossas aplicações queremos definir o Δx que minimize a ordem do erro. Isso vai ser feito extremizando a função que dá a ordem do erro. Ou seja, tomando a derivada e procurando pelo valor que a resulte nula. Claro, é necessário checar o ponto para saber se é mínimo, na noss afunção isso não será problema. Para a função obtida teremos minimização para $\Delta x \approx 0.01$ como podemos visualizar no gráfico abaixo.

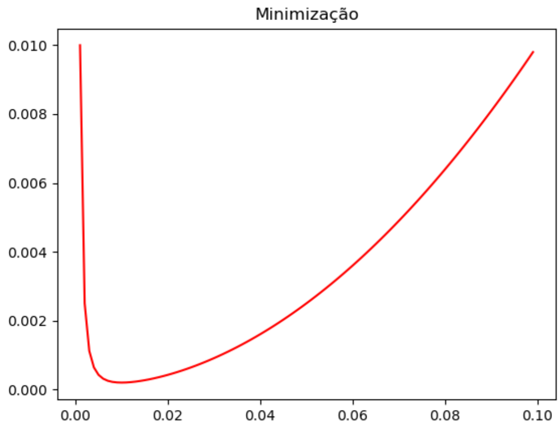


Figure 1: Função $\Delta t^2 + \frac{10^{-8}}{\Delta t^2}$

1.2 Condições de contorno e borda

Claro, para resolvermos a equação será necessário definir as condições iniciais do sistema e as condições de borda. Especificamente, todo passo temporal depende dos dois passos anteriores já que queremos considerar acelerações.

Para a solução, precisaremos então de uma função arbitrária $Y(x, 0) = Y_0(x)$, da velocidade $\dot{Y}(x, 0)$ e das condições de bordo, como por exemplo, bordas fixas. Inicializando os primeiros dois *steps* temporais baseados nas condições iniciais e atualizaremos nosso sistema de acordo com a equação da onda com a condição de borda em consideração.

Em nota, por tratar de um meio finito faremos $i = 0, 1, ..., \frac{L}{\Delta x}$, onde L é comprimento da corda que trataremos.

2 Cordas

Vamos simular um sistema de corda. Tomaremos um meio não dissipativo de comprimento $L = 1m$ onde a onda se mova com velocidade $c = 300m/s$. Tomaremos bordas fixas e condições iniciais:

$$Y(x, 0) = Y_0(x) = e^{-\frac{(x-x_0)^2}{\sigma^2}}$$

Com $x_0 = \frac{L}{3}$ e $\sigma = \frac{L}{30}$. Também temos

$$\dot{Y}(x, 0) = 0$$

2.1 Ia

Tomaremos uma grade equivalente à uma matriz M_{100x3} onde o comprimento é discretizado em 100 passos e o tempo em 500. Note que nossa matriz só mantém 3 passos temporais pois é tudo necessário para o algoritmo. Pela escolha da quantidade de passo dimensionais, teremos $dx = 0.01$. Escrevemos o programa:

```fortran
      PROGRAM GENERATE
            IMPLICIT REAL (KIND=8) (a-h,l,o-z)
            DIMENSION grade(100, 3)
            COMMON /ONDA/ grade

c           grade
            ixs = 100
            its = 500

c           parametros
            c = 300.d0
            L = 1.d0
            dx = L / ixs
            r = 1.d0

            open(1, file='onda.dat', status='unknown')

c           condicoes iniciais (bordas devem ser zero)
            DO 10 n = 1,2
                  DO 20 i=1,ixs
                        grade(i,n) = CONDINICIAIS(i*dx, L)
  20              END DO
                  write(1,fmt='(100(F12.8))') (grade(i,n), i=1,ixs)
  10        END DO

c           definir a evolucao temporal
            DO 40 n=3,its
                  CALL UPDATE(ixs, r)
                  write(1,fmt='(100(F12.8))') (grade(i,3), i=1,ixs)
  40        END DO

            close(1)

      END PROGRAM GENERATE

      SUBROUTINE UPDATE(ixs, r)
            IMPLICIT REAL (KIND=8) (a-h,l,o-z)
            DIMENSION grade(100, 3)
            COMMON /ONDA/ grade

c           paredes fixas
            grade(1, 3) = grade(1, 2)
            grade(ixs, 3) = grade(ixs, 2)

c           nova linha (sem paredes)
            DO 10 i=2,ixs-1
                  grade(i, 3) = 2.d0*(1.d0-r*r)*grade(i,2) +
     +                  r*r*(grade(i+1,2)+grade(i-1,2))-grade(i,1)
 10         END DO

            grade(:,1) = grade(:,2)
            grade(:,2) = grade(:,3)

      END SUBROUTINE UPDATE
```

```
      FUNCTION CONDINICIAIS(x, L)
            IMPLICIT REAL (KIND=8) (a-h,l,o-z)
c           F0(x) =  exp[(x  x0)^2 / ^2]
c            = L/30
c           x0 = L/3
            CONDINICIAIS = EXP(-((x-L/3)**2)/(L/30)**2)
      END FUNCTION CONDINICIAIS
```

Podemos plotar a evolução obtida com o programa. Faremos um gráfico tridimensional onde a evolução temporal é representada pelo eixo y, o espaço pelo eixo x e a amplitude pelo eixo z. Vemos:

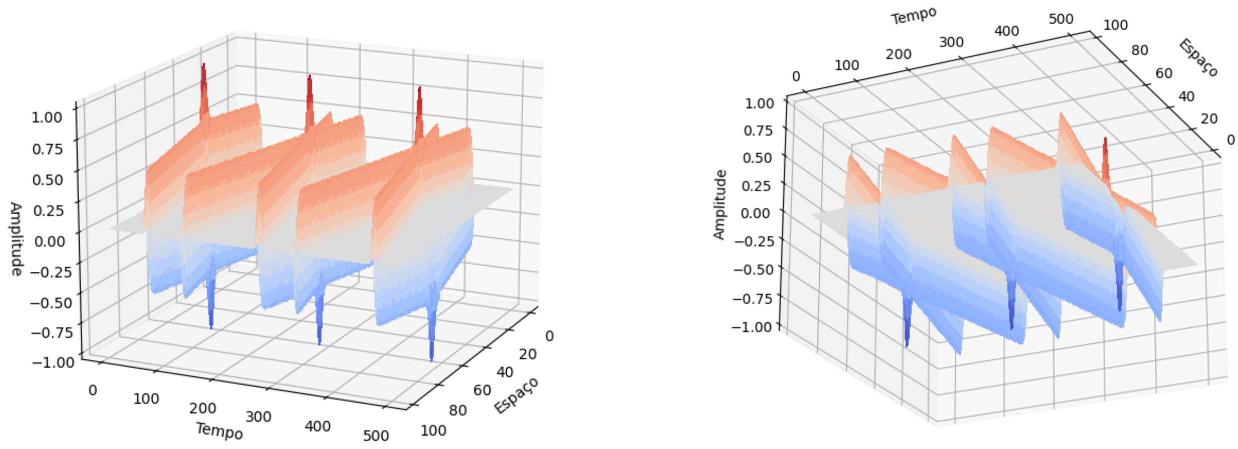
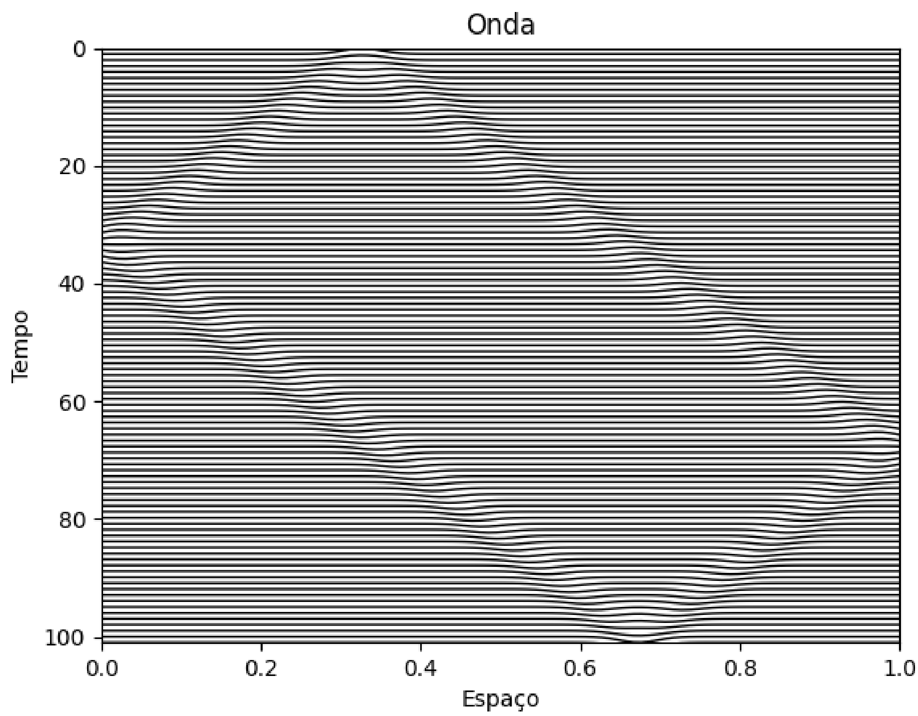


Figure 2: Evolução temporal sistema

Para ter certeza que o processo está claro, faremos um recorte alterando o eixo y. Notavelmente alguns processos como a reflexão e a sobreposição de ondas podem ser observados na figura abaixo.



1. (Ia1) Foi usado $dx = 0.01$, como explicitado na introdução sobre como minimizar a ordem do erro.

2. (Ia2) O pacote não se deforma, ele se propaga integralmente pelo meio neste contexto onde $r = 1$.

3. (Ia3) As reflexões estão de acordo com a teoria. Com a borda fixa, a onda deve espelhar a amplitude, o que pode ser observado nas figuras geradas.

4. (Ia4) As interferências são da forma que se esperaria, os "dois" pacotes que oscilam na corda passam pelo outro e se somam sem modificar a forma do outro após o encontro.

5. (Ia5) A configuração inicial se repetirá quando os pacotes tiverem o tempo de retornar a mesma posição inicial ao mesmo tempo. Isso ocorrerá quando ambos tiverem percorrido $2L$ cada. Ou seja, demorará $t = 2L/c$.

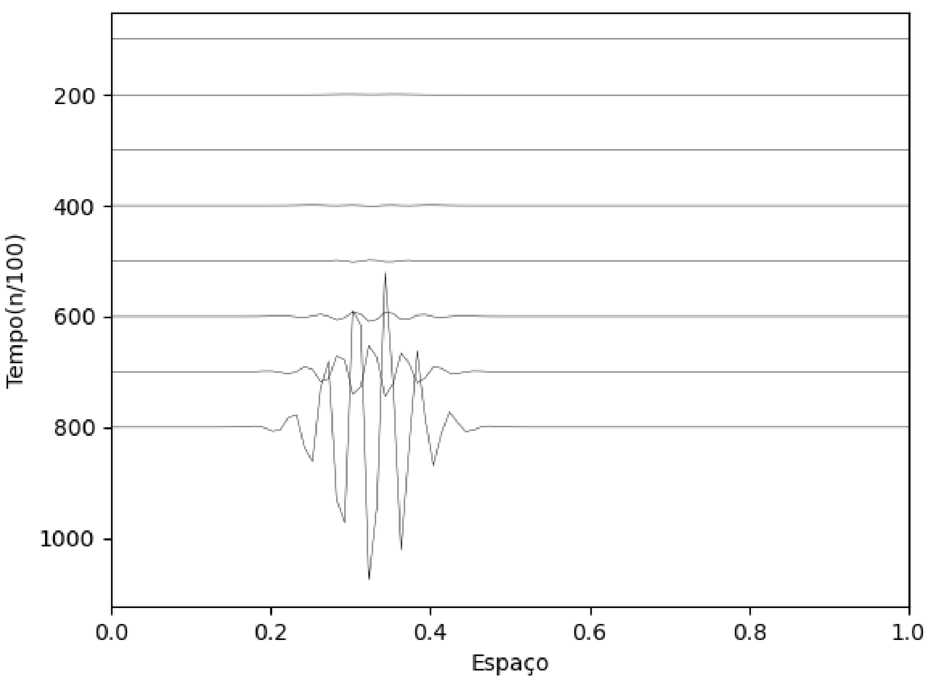
2.2 Valores de r variados

No nosso problema estamos tratando de dois *steps* distintos, um para t e um para x. Até agora definimos $r = 1$. Esta escolha tem uma implicação bem direta na anulação de termos que foram cortados na aproximação. mostraremos os efeitos de se mudar esta razão.

2.2.1 Ib - $r > 1$

Repetiremos a simulação anterior mas tomaremos $r = 2$. Neste caso note que a "velocidade da malha" ($\frac{\Delta x}{\Delta t}$) será o dobro da velocidade da onda no meio.

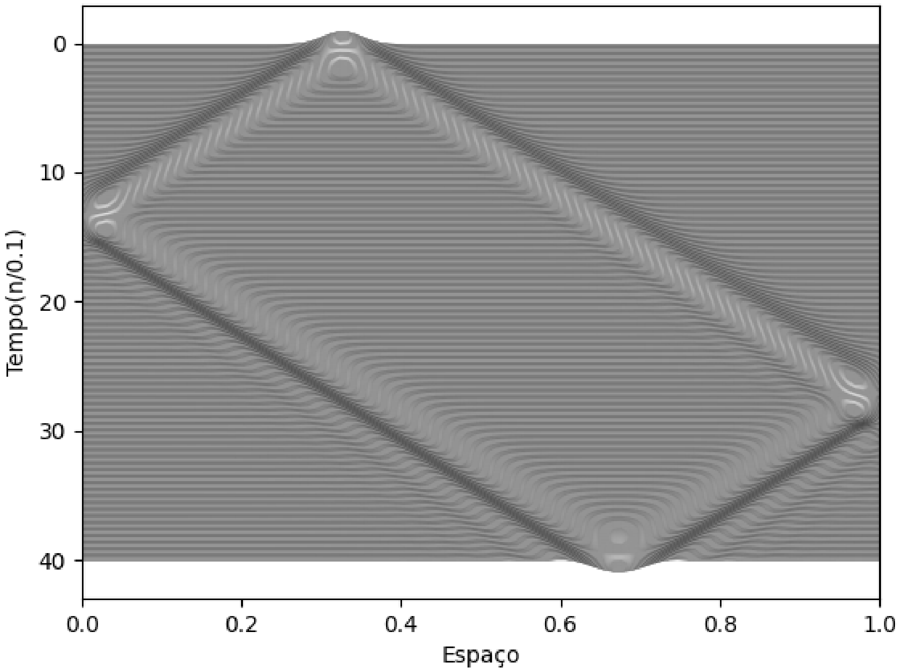
Neste caso a amplitude estoura rapidamente em algumas iterações. A simulação se torna instável. a representação gráfica do fenômeno é até difícil de ser realizada visto que o número cresce extremamente rápido.



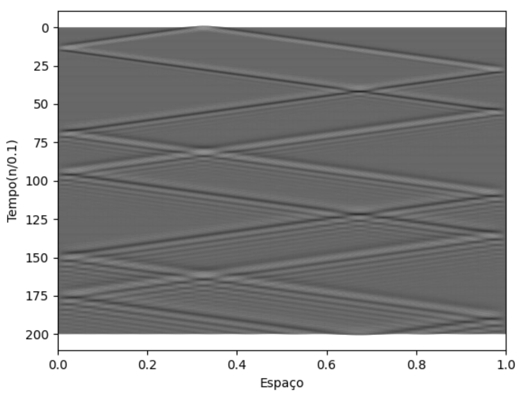
Isto está relacionado com o fato do algoritmo não poder resultar corretamente para *steps* tais que a velocidade da malha é menor que a da onda no meio, $\frac{\Delta x}{\Delta t} < c$.

2.2.2 Ic - $r < 1$

Repetiremos a simulação anterior mas tomaremos $r = 0.25$. Neste caso teremos que $\frac{\Delta x}{\Delta t} < c$ e podemos na verdade rodar nosso algoritmo com alguma estabilidade. De fato podemos gerar simulações como as anteriores.



Com um porém, temos uma situação ainda não ideal de cancelamento de termos da expansão que discretizou nosso sistema. Com isso, se rodarmos por tempo suficiente, estes termos se fazem aparente.



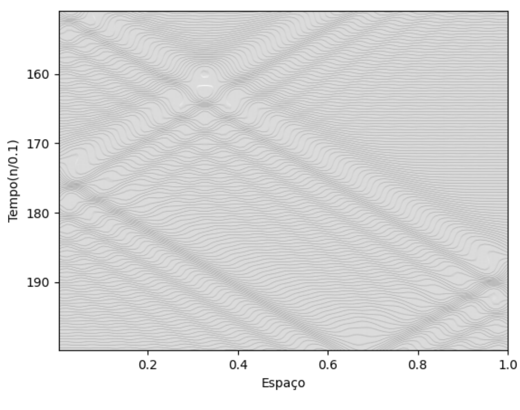


Figure 3: Evolução temporal com propagação do erro

Pode-se observar nas figuras como depois de algum tempo surgem outros picos de onda da imperfeição de reflexões e interferências para este cenário.

3 Violão

Vamos simular um sistema de corda de violão. Tomaremos os mesmo parâmetros da corda anteriormente simulada mas com condições inicias diferentes. Tal que:

$$f(x) = \begin{cases} x, & \text{se} x \leq L/4 \\ -\frac{x}{3} + \frac{L}{3}, & \text{doutra forma} \end{cases}$$

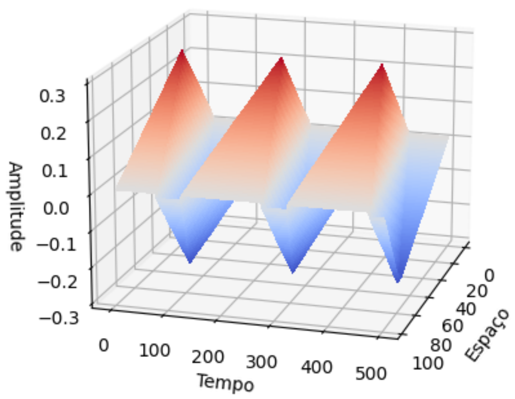
Temos, é claro

$$\dot{Y}(x, 0) = 0$$

Novamente, tomaremos uma grade equivalente à uma matriz M_{100x3} onde o comprimento é discretizado em 100 passos e o tempo em 500. Assim $dx = 0.01$. Adicionamos então ao programa anterior uma nova condição inicial possível em forma de function.

```
      FUNCTION VIOLAO(x, L)
            IMPLICIT REAL (KIND=8) (a-h,l,o-z)
c           FROM 0 TO L/4 -> F(X) = X
c           FROM L/4 TO L -> F(X) = L/3 - x/3
            IF (x .LE. L/4) THEN
                  VIOLAO = x
            ELSE
                  VIOLAO = L/3 - x/3
            END IF
      END FUNCTION VIOLAO
```

Obtemos o perfil de onda do tipo esperado



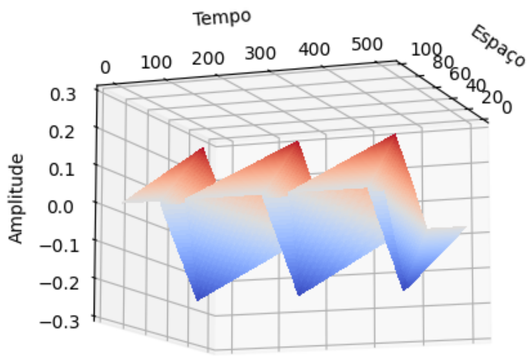
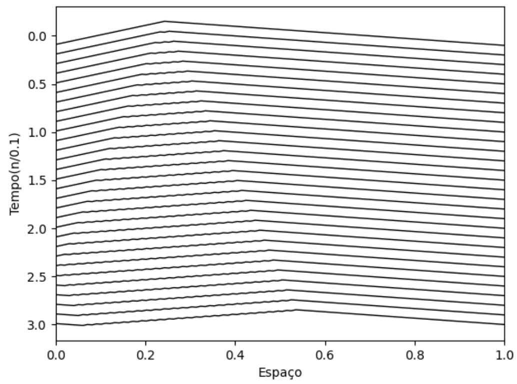


Figure 4: Evolução temporal sistema

Para ter certeza que o processo está claro, faremos alguns recortes novamente. Desta vezes, plotamos menos passos temporais pois nas mesmas condições a onda ficava muito mais discreta.



1. (Ia2) O pacote não se deforma, ele se propaga integralmente pelo meio desde que $r = 1$.

2. (Ia3) As reflexões estão de acordo com a teoria. Com a borda fixa, a onda deve espelhar a amplitude, o que pode ser observado nas figuras geradas.

3. (Ia4) As interferências são da forma que se esperaria, a parte de amplitude negativa interfere com a parte de amplitude positiva e os pacotes que formam se somam quando se encontram.

4. (Ia5) A configuração inicial se repetirá quando as frentes de onda tiverem o tempo de retornar a mesma posição inicial ao mesmo tempo. Isso ocorrerá quando ambos tiverem percorrido $2L$ cada. Ou seja, demorará $t = 2L/c$.