Reunião de acompanhamento

15/12/2021

Conteúdo

- Método integrador (Boris stepper)
- Passos de implementação
- Implementação da rotação
- Código

Método integrador

- Força de Lorentz (Newton-Lorentz)

$$m\frac{d\boldsymbol{v}}{dt} = q(\boldsymbol{E} + \boldsymbol{v} \times \boldsymbol{B}) \quad (1) \qquad\qquad \frac{d\boldsymbol{x}}{dt} = \boldsymbol{v} \quad (2)$$

- Diferença Centrada

$$\frac{\boldsymbol{v}_{t+\Delta t/2} - \boldsymbol{v}_{t-\Delta t/2}}{\Delta t} = \frac{q}{m}\left[\boldsymbol{E} + \frac{\boldsymbol{v}_{t+\Delta t/2} + \boldsymbol{v}_{t-\Delta t/2}}{2} \times \boldsymbol{B}\right] \quad (3)$$

Método integrador

- Separação das forças elétricas e magnéticas

→ Substituição

$$\boldsymbol{v}_{t-\Delta t/2} = \boldsymbol{v}^{-} - \frac{q\boldsymbol{E}}{m}\frac{\Delta t}{2} \qquad (4)$$

$$\boldsymbol{v}_{t+\Delta t/2} = \boldsymbol{v}^{+} + \frac{q\boldsymbol{E}}{m}\frac{\Delta t}{2} \qquad (5)$$

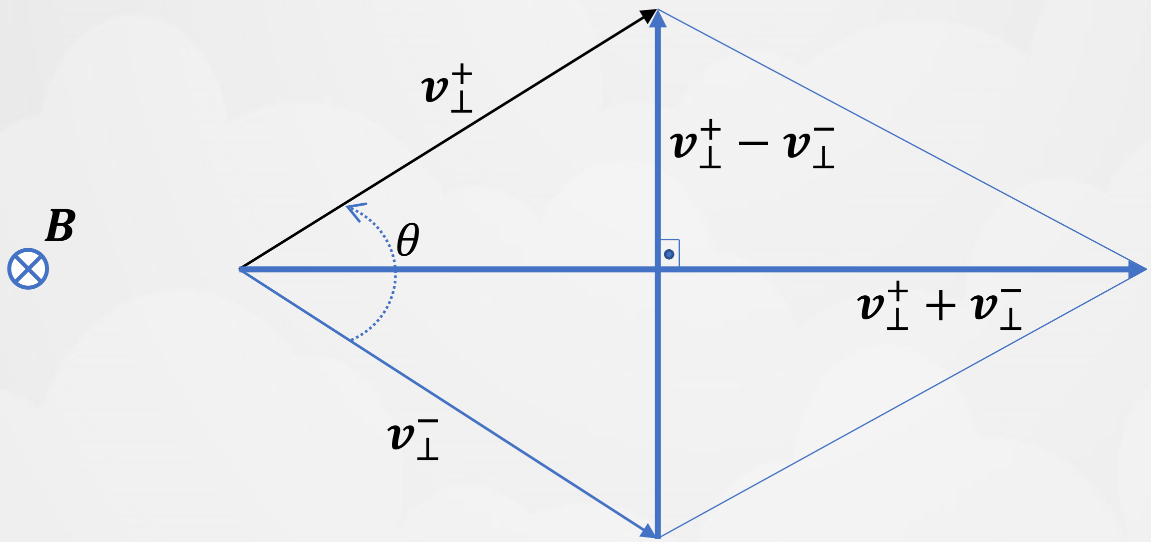
→ Subst. (4) e (5) em (3)

$$\frac{\boldsymbol{v}^{+} - \boldsymbol{v}^{-}}{\Delta t} = \frac{q}{2m}\left(\boldsymbol{v}^{+} + \boldsymbol{v}^{-}\right) \times \boldsymbol{B} \qquad (6)$$

→ O campo E é inteiramente cancelado.

Método integrador

(Rotação)



$$\left|\tan\frac{\theta}{2}\right| = \frac{|\boldsymbol{v}_\perp^+ - \boldsymbol{v}_\perp^-|}{|\boldsymbol{v}_\perp^+ + \boldsymbol{v}_\perp^-|} = \frac{qB}{m}\frac{\Delta t}{2} = \frac{\omega_c \Delta t}{2} \quad (7) \qquad \omega_c \equiv \frac{q}{m}B \quad (8)$$

$$\theta = 2\tan^{-1}\left(\frac{qB}{m}\frac{\Delta t}{2}\right) = \omega_c \Delta t\left(1 - \frac{(\omega_c \Delta t)^2}{12} + \cdots\right) \quad (9)$$

Passos de implementação

- Somar um "impulso elétrico" em $\boldsymbol{v}_{t-\Delta t/2}$ na eq. (4)

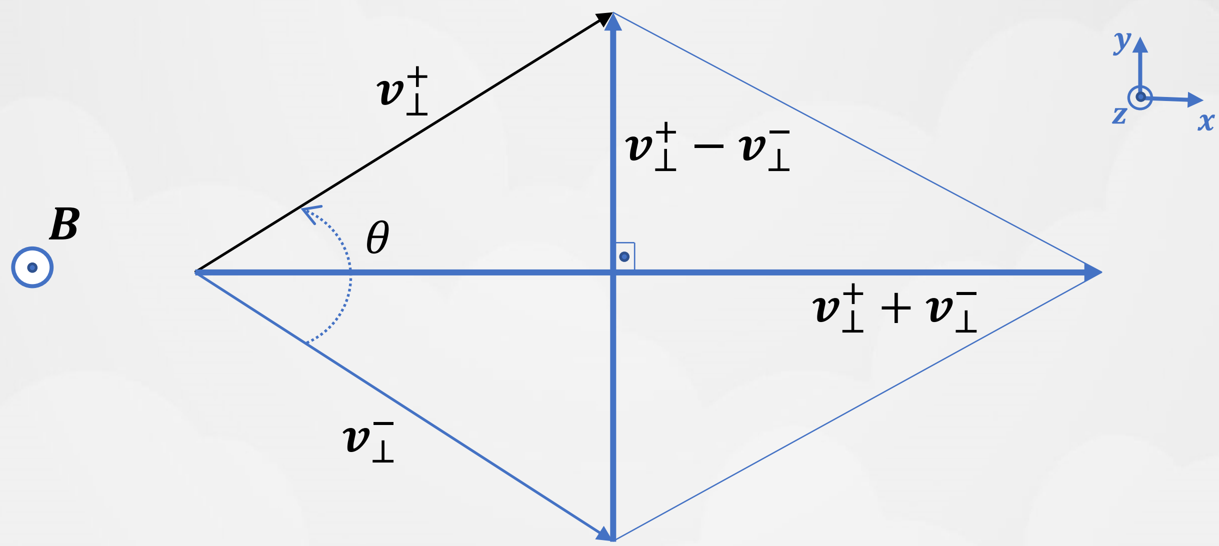
$$\boldsymbol{v}^{-} = \boldsymbol{v}_{t-\Delta t/2} + \frac{q\boldsymbol{E}}{m}\frac{\Delta t}{2} \qquad (10)$$

- Rotacionar $\boldsymbol{v}^{-}$ para obter $\boldsymbol{v}^{+}$ através da eq. (6) (implementação da rotação)

- Somar o "impulso elétrico" a $\boldsymbol{v}^{+}$ na eq. (5) para obter $\boldsymbol{v}_{t-\Delta t/2}$

$$\boldsymbol{v}_{t+\Delta t/2} = \boldsymbol{v}^{+} + \frac{q\boldsymbol{E}}{m}\frac{\Delta t}{2} \qquad (11)$$

Implementação da rotação



$$\tan\frac{\theta}{2} = -\frac{qB}{m}\frac{\Delta t}{2} \qquad (12)$$

$$\frac{\theta}{2} \sim \frac{qB}{m}\frac{\Delta t}{2} \qquad (13)$$

$$\tan\frac{\theta}{2} = -\tan\left(\frac{qB\Delta t}{2m}\right) \qquad (14)$$

Implementação da rotação

$$t \equiv -\tan\left(\frac{qB\Delta t}{2m}\right)$$

$$s \equiv \sin\theta = \frac{2t}{1+t^2} \qquad (16)$$

$$c \equiv \cos\theta = \frac{1-t^2}{1+t^2} \qquad (17)$$

→ Matriz de rotação

$$M(\theta) = \begin{pmatrix} c & s \\ -s & c \end{pmatrix} \qquad (18)$$

→ Rotação de $\boldsymbol{v}^-$

$$\begin{pmatrix} v_x^+ \\ v_y^+ \end{pmatrix} = \begin{pmatrix} c & s \\ -s & c \end{pmatrix} \begin{pmatrix} v_x^- \\ v_y^- \end{pmatrix} \longrightarrow \begin{cases} v_x^+ = cv_x^- + sv_y^- & (19) \\ v_y^+ = -sv_x^- + cv_y^- & (20) \end{cases}$$

Implementação da rotação

- Considerando as equações (16), (17), (19) e (20) tem-se:

→ 7 multiplicações

→ 1 divisão

→ 5 adições

- BIRDSALL e LANGDON (1991) sugerem um artifício que reduz a quantidade de operações para 4 multiplicações, 1 divisão e 5 adições.

$$v'_x = v_x^- + v_y^- t \qquad (21)$$

$$v_y^+ = v_y^- + v'_x s \qquad (22)$$

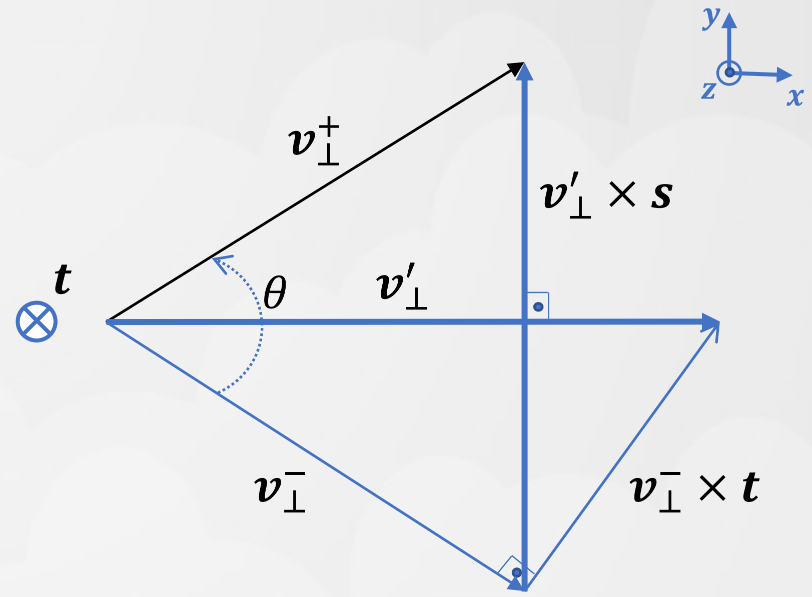
$$v_x^+ = v'_x + v_y^+ t \qquad (23)$$

Artifício

- $\boldsymbol{v}' = \boldsymbol{v}^- + \boldsymbol{v}^- \times \boldsymbol{t} \qquad (24)$

$$\boldsymbol{t} \equiv -\hat{\boldsymbol{z}} \tan\frac{\theta}{2} = \frac{q\boldsymbol{B}}{m}\frac{\Delta t}{2}$$

- $\boldsymbol{v}^+ = \boldsymbol{v}^- + \boldsymbol{v}' \times \boldsymbol{s} \qquad (25)$



⟶ **s** é paralelo a $\boldsymbol{B}$ (e a $\boldsymbol{t}$) e sua magnitude é determinada pela condição

$$|\boldsymbol{v}^-|^2 = |\boldsymbol{v}^+|^2 \qquad (26)$$

$$\boldsymbol{s} = \frac{2\boldsymbol{t}}{1 + t^2} \qquad (27)$$

Caso relativístico

- Retomando a equação da Força de Lorentz (Newton-Lorentz)

$$\frac{d\boldsymbol{p}}{dt} = q(\boldsymbol{E} + \boldsymbol{v} \times \boldsymbol{B}) \tag{28}$$

- Introduzindo o fator de Lorentz

$$\gamma = \frac{1}{\sqrt{1 - \frac{v^2}{c^2}}} \tag{29}$$

- Definindo $\boldsymbol{p} = \gamma m \boldsymbol{v}$ e $\boldsymbol{u} = \gamma \boldsymbol{v}$, tem-se:

$$m\frac{d\boldsymbol{u}}{dt} = q\left(\boldsymbol{E} + \frac{\boldsymbol{u} \times \boldsymbol{B}}{\gamma}\right) \tag{30}$$

Código

```
def push_particles(q,m,dt,c,efield,bfield,u,r,v):
    import numpy as np
    #me = 9.109e-31; qe = 1.602e-19; Re = 6.371e6
    me = 1.0
    qe = 1.0
    ex = efield[0]; ey = efield[1]; ez = efield[2]
    bx = bfield[0]; by = bfield[1]; bz = bfield[2]
    vx = v[0]; vy = v[1]; vz = v[2]
    ux = u[0]; uy = u[1]; uz = u[2]
    rx = r[0]; ry = r[1]; rz = r[2]

    # Lorentz's factor
    gamma = 1.0/np.sqrt(1-(vx**2+vy**2+vz**2)/c**2)

    # uminus
    umx = ux + ((q*qe)*dt/(2*(m*me)))*ex
    umy = uy + ((q*qe)*dt/(2*(m*me)))*ey
    umz = uz + ((q*qe)*dt/(2*(m*me)))*ez

    # uprime = uminus + (qdt/2mgamma)*(uminus x B)
    uprx = umx +((q*qe)*dt/(2*(m*me)*gamma))*(umy*bz - umz*by)
    upry = umy +((q*qe)*dt/(2*(m*me)*gamma))*(umz*bx - umx*bz)
    uprz = umz +((q*qe)*dt/(2*(m*me)*gamma))*(umx*by - umy*bx)

    # S = (qdt/(mgamma))B/(1+(qdtB/2mgamma)^2)
    bm = np.sqrt(bx**2+by**2+bz**2)
    sx = (((q*qe)*dt/((m*me)*gamma))*bx)/(1 + ((q*qe)*dt*bm/(2*(m*me)*gamma))**2)
    sy = (((q*qe)*dt/((m*me)*gamma))*by)/(1 + ((q*qe)*dt*bm/(2*(m*me)*gamma))**2)
    sz = (((q*qe)*dt/((m*me)*gamma))*bz)/(1 + ((q*qe)*dt*bm/(2*(m*me)*gamma))**2)

    # uplus = uminus + (uprime x S)
    upx = umx + (upry*sz - uprz*sy)
    upy = umy + (uprz*sx - uprx*sz)
    upz = umz + (uprx*sy - upry*sx)

    # u^(n+1/2) = uplus + (qdt/2m)*E (electric field)
    ux = upx + ((q*qe)*dt/(2*(m*me)))*ex
    uy = upy + ((q*qe)*dt/(2*(m*me)))*ey
    uz = upz + ((q*qe)*dt/(2*(m*me)))*ez
    um = np.sqrt(ux*ux + uy*uy + uz*uz)
    gamma = np.sqrt(1+(um*um)/(c*c))

    #updates particle's velocity
    u[0] = ux; u[1] = uy; u[2] = uz
    vx = ux/gamma; vy = uy/gamma; vz = uz/gamma
    #ub = np.array([ux      , uy      , uz      ])
    #vb = np.array([ux/gamma, uy/gamma, uz/gamma])
    v[0] = vx; v[1] = vy; v[2] = vz

    #updates particle's position
    #rb = np.array([rx + (dt*vx)/Re, ry + (dt*vy)/Re, rz + (dt*vz)/Re])
    r[0] = rx + (dt*vx)
    r[1] = ry + (dt*vy)
    r[2] = rz + (dt*vz)
    return r,v
```

Referências

- BIRDSALL, C K; LANGDON, A B. **Plasma Physics via Computer Simulation**. Bristol, Philadelphia And New York: Adam Hilger, 1991.

$$\frac{\boldsymbol{v}^+ - \boldsymbol{v}^-}{\Delta t} = \frac{q}{2m}(\boldsymbol{v}^+ + \boldsymbol{v}^-) \times \boldsymbol{B}$$

- $(\boldsymbol{v}^+ - \boldsymbol{v}^-) \cdot (\boldsymbol{v}^+ + \boldsymbol{v}^-) = (v^+)^2 - (v^-)^2$
- $\big((\boldsymbol{v}^+ + \boldsymbol{v}^-) \times \boldsymbol{B}\big) \cdot (\boldsymbol{v}^+ + \boldsymbol{v}^-) = 0$

$$v^+ = v^-$$