Evolución orbital de exoplanetas por fuerzas de marea

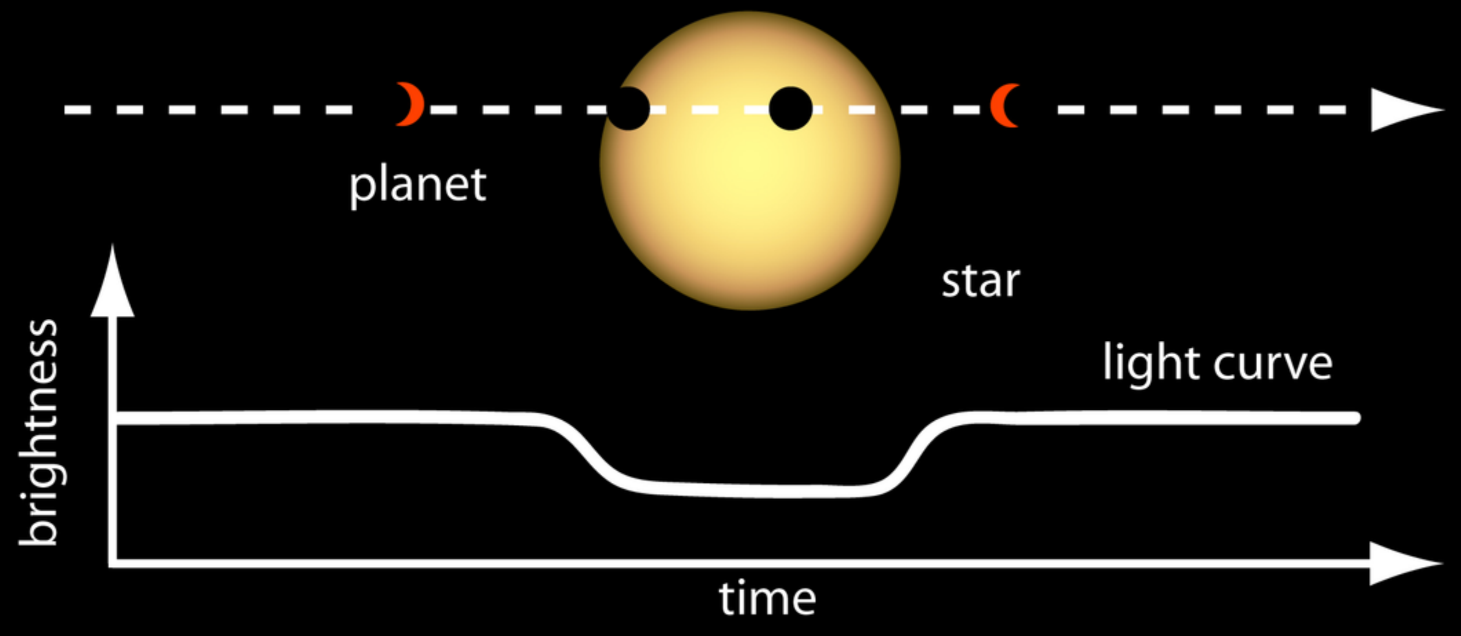
Valeria Abraham

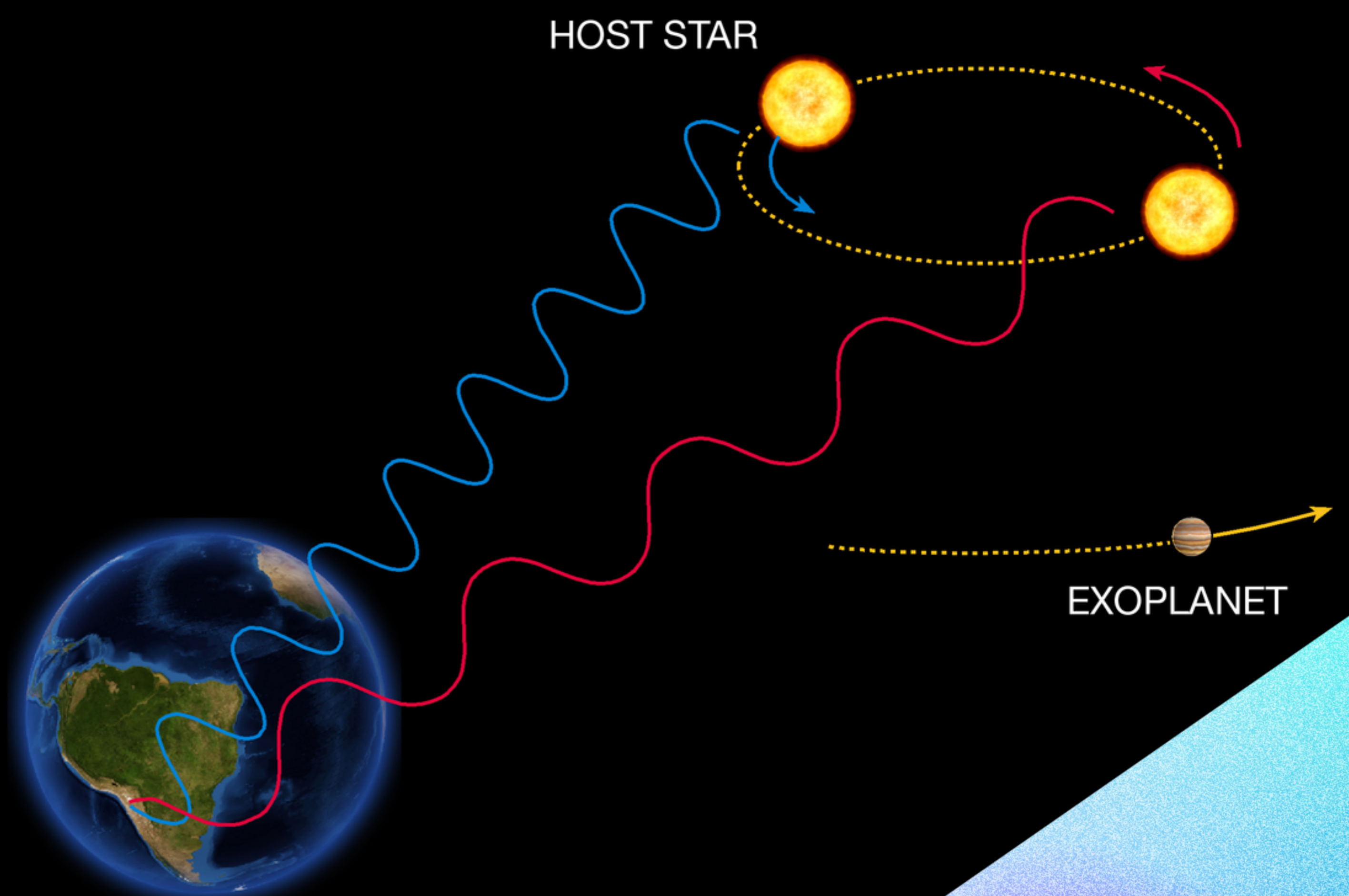
INTRODUCCIÓN

¿Qué es un exoplaneta?

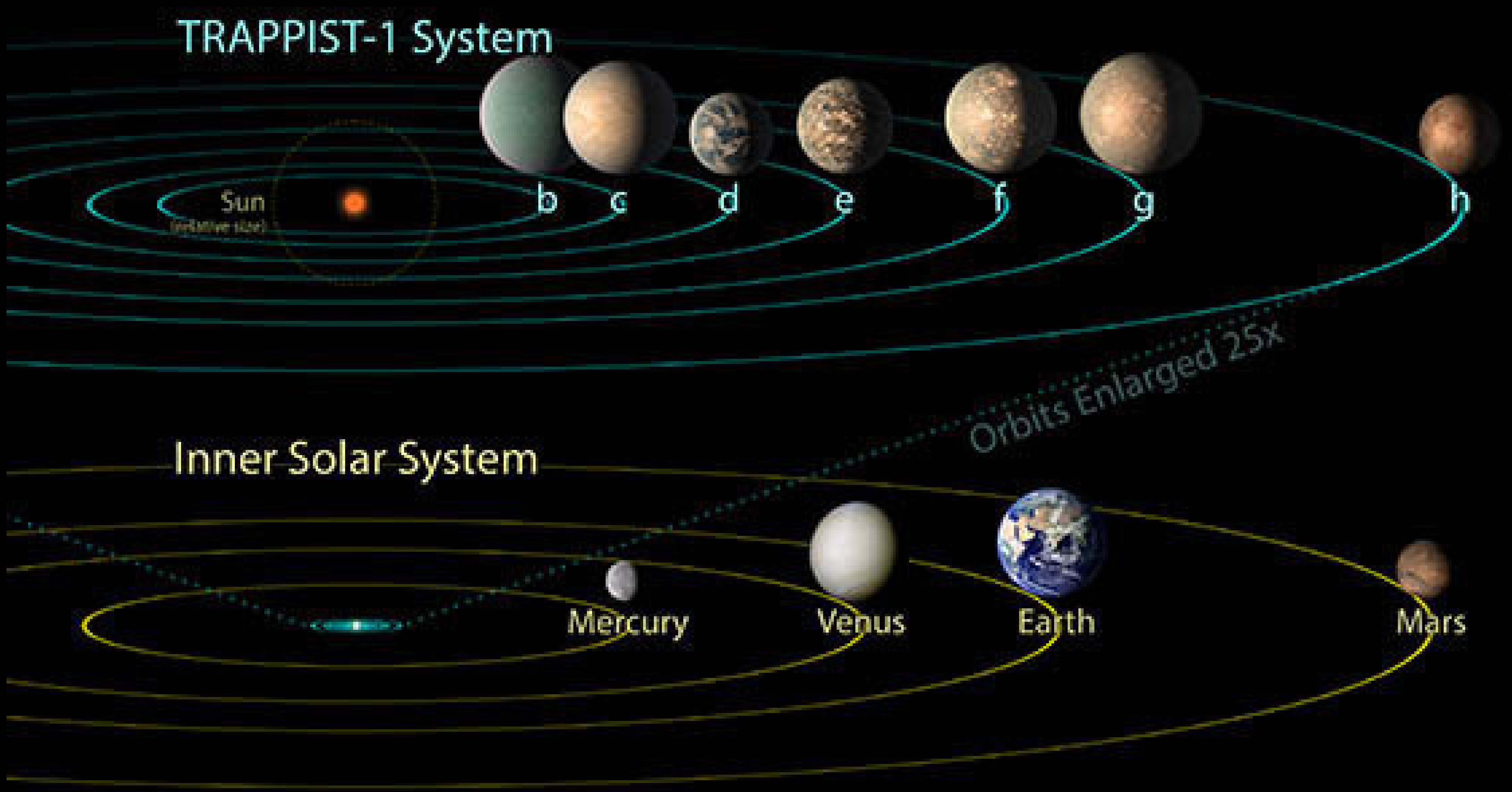
Planeta que orbita una o varias estrellas que no son el Sol.

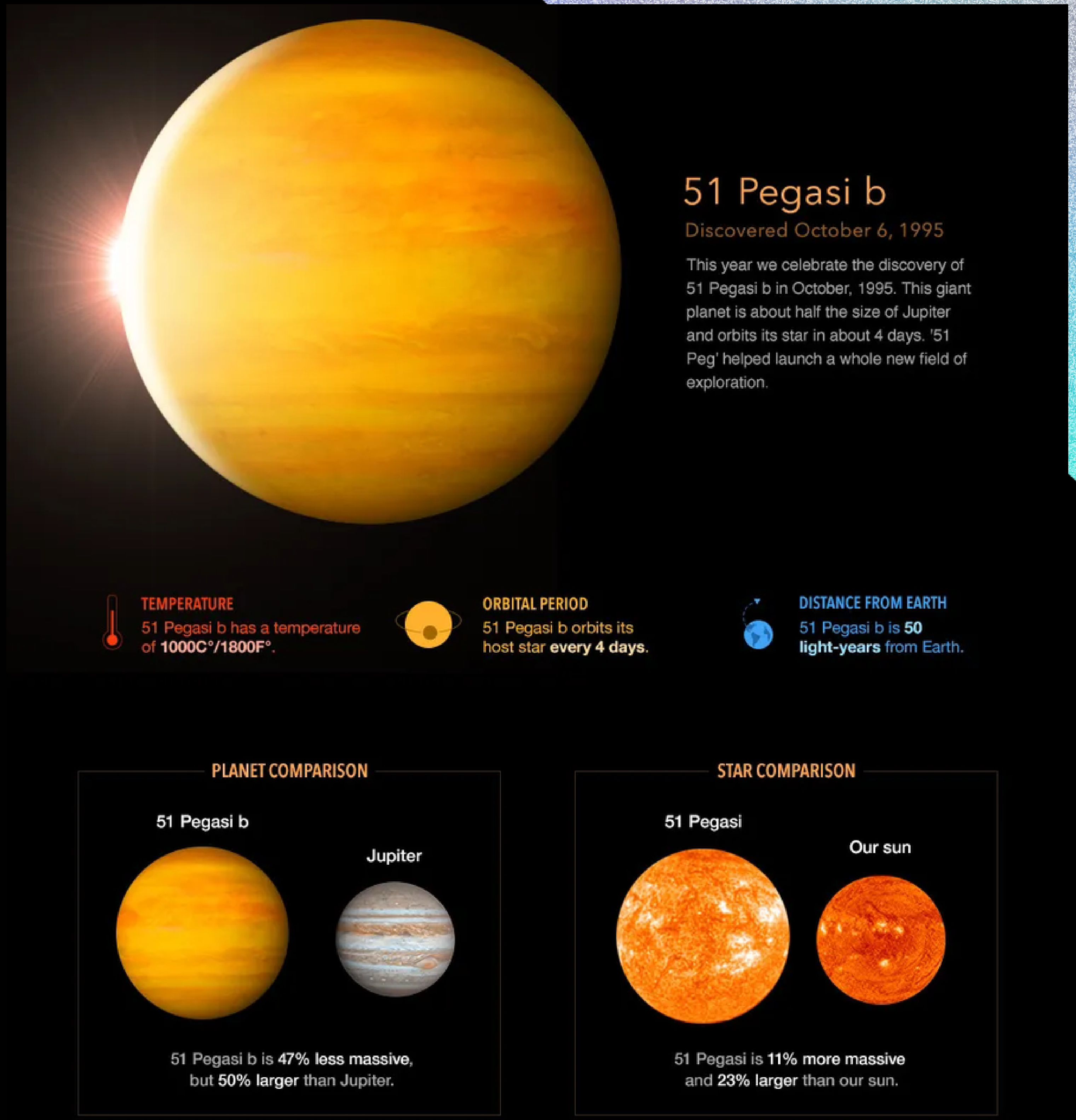
¿Cómo se descubren?



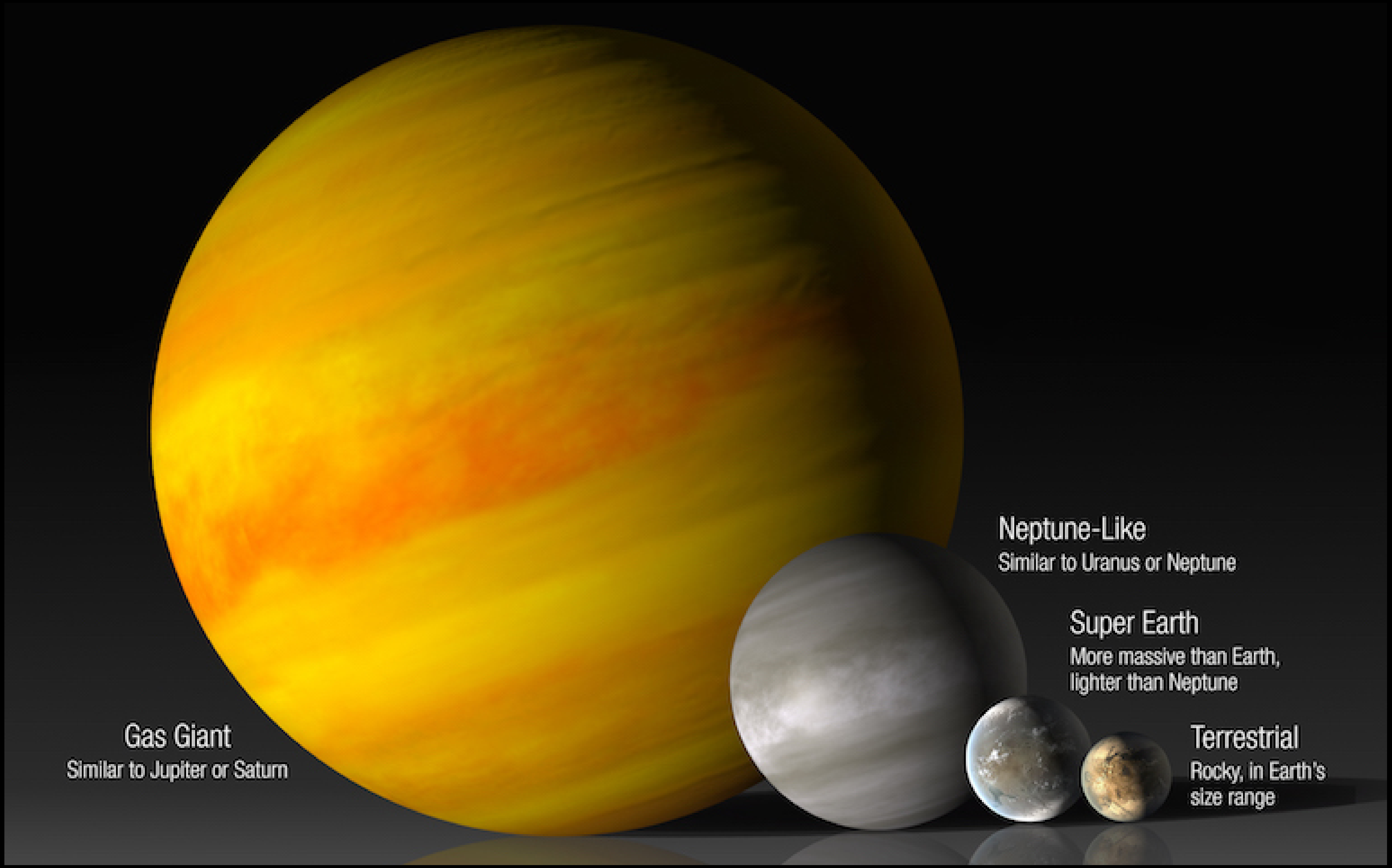


¿Por qué nos interesan?





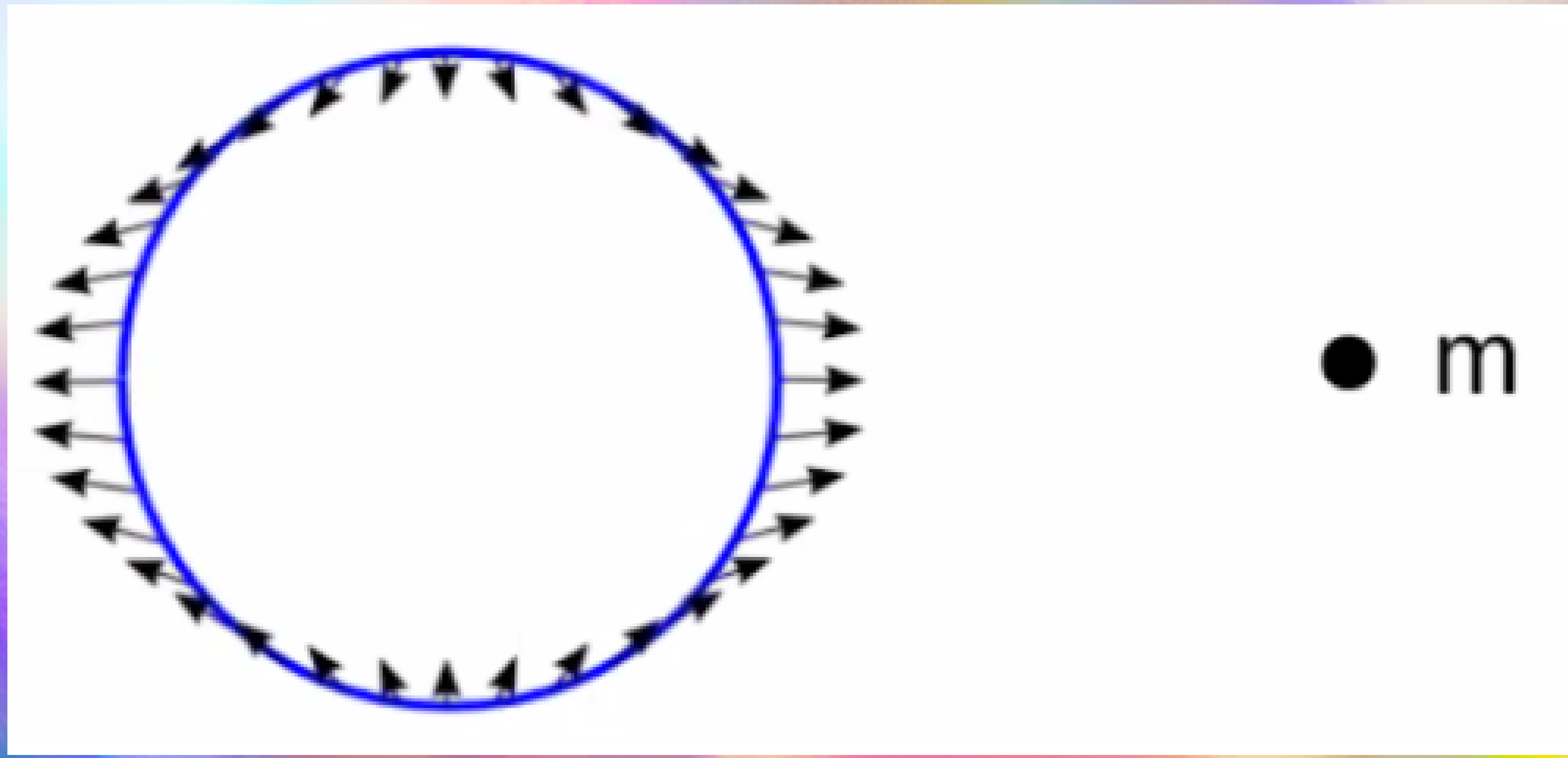
Clasificación:



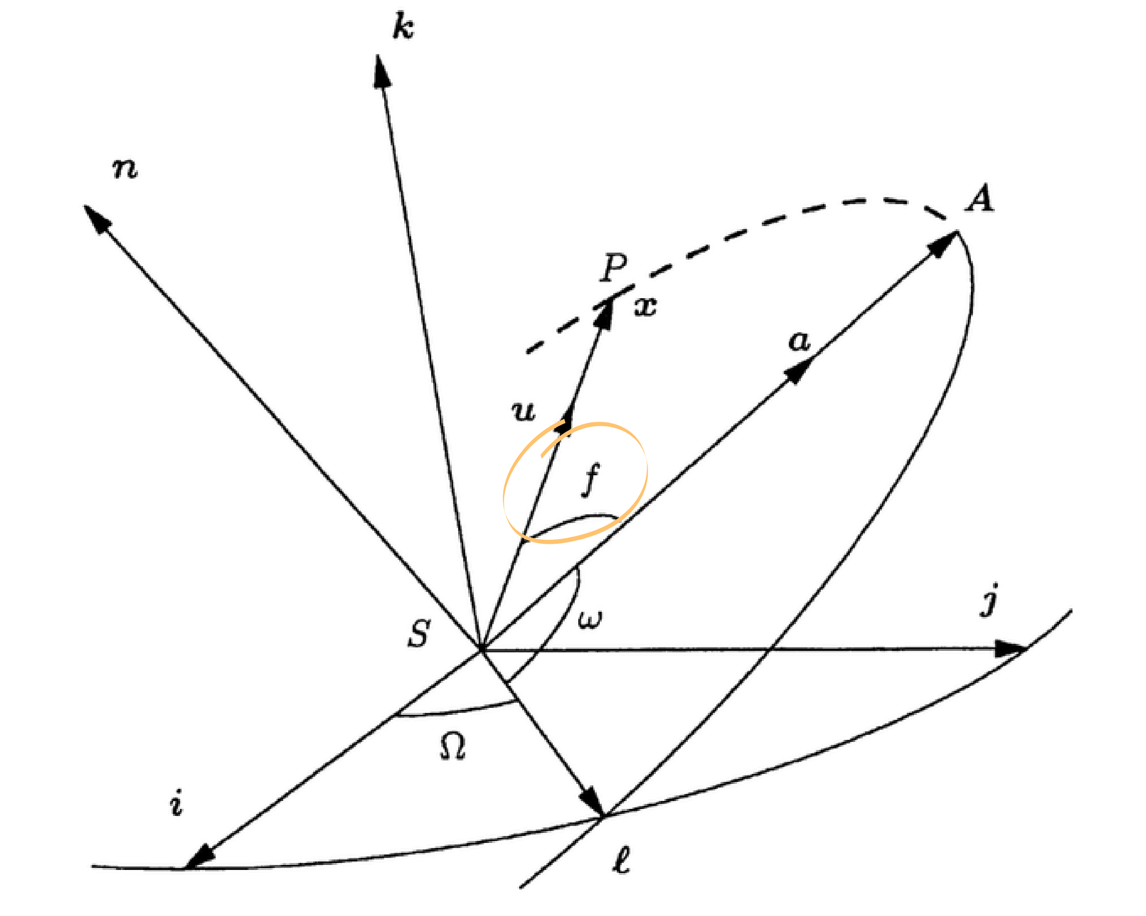
Fuerza de marea:

Efecto secundario de la fuerza gravitacional entre dos cuerpos masivos no puntuales.

Expansión de
Ferraz - Mello et al. (2008)



¿Cómo estudiamos la **evolución orbital?**



Variaciones medias de *a* y *e*.

Ecuaciones de Gauss

$$\langle \dot{a} \rangle = \frac{1}{\sqrt{\mu a}(1-e^2)\pi}\int_0^{2\pi}[eR\sin f + T(1+e\cos f)]r^2 df$$

$$\langle \dot{e} \rangle = \frac{1}{\sqrt{\mu a^3}2\pi}\int_0^{2\pi}\left[R\sin f + T\left(\cos f + \frac{1}{e}\left(1+\frac{r}{a}\right)\right)\right]r^2 df$$

Ahora sí, las ecuaciones a integrar:

A partir de Ferraz - Mello et al. (2008) y Rodríguez y Ferraz - Mello (2010).

$$\boxed{\begin{aligned}\langle \dot{a} \rangle &= -\frac{2}{3} n a^{-4} [(2 + 46e^2)\hat{s} + 7e^2 \hat{p}] \\ \langle \dot{e} \rangle &= -\frac{1}{3} n e a^{-5} [18\hat{s} + 7\hat{p}]\end{aligned}}$$

$$\hat{s} = \frac{9}{4} \frac{k_{d*}}{Q_*} \frac{m_p}{m_*} R_*^5$$

$$\hat{p} = \frac{9}{2} \frac{k_{dp}}{Q_p} \frac{m_*}{m_p} R_p^5$$

Introducimos al segundo planeta...

A partir de Mardling (2007) e introduciendo la disipación de energía por el efecto de marea.

$$\dot{e}_p = -W_o e_c \sin\eta - W_T e_p$$
$$\dot{e}_c = W_c e_p \sin\eta$$
$$\dot{\eta} = \overline{W_q} - W_o \frac{e_c}{e_p} \cos\eta$$

$$W_o = \frac{15}{16} n_p \left(\frac{a_p}{a_c}\right)^4 \frac{m_c}{m_*} \varepsilon_c^{-5}$$

$$W_c = \frac{15}{16} n_c \left(\frac{a_p}{a_c}\right)^3 \frac{m_p}{m_*} \varepsilon_c^{-4}$$

$$W_T = \frac{21}{2} n_p \frac{k_p}{Q_p} \frac{m_*}{m_p} \left(\frac{R_p}{a_p}\right)^5$$

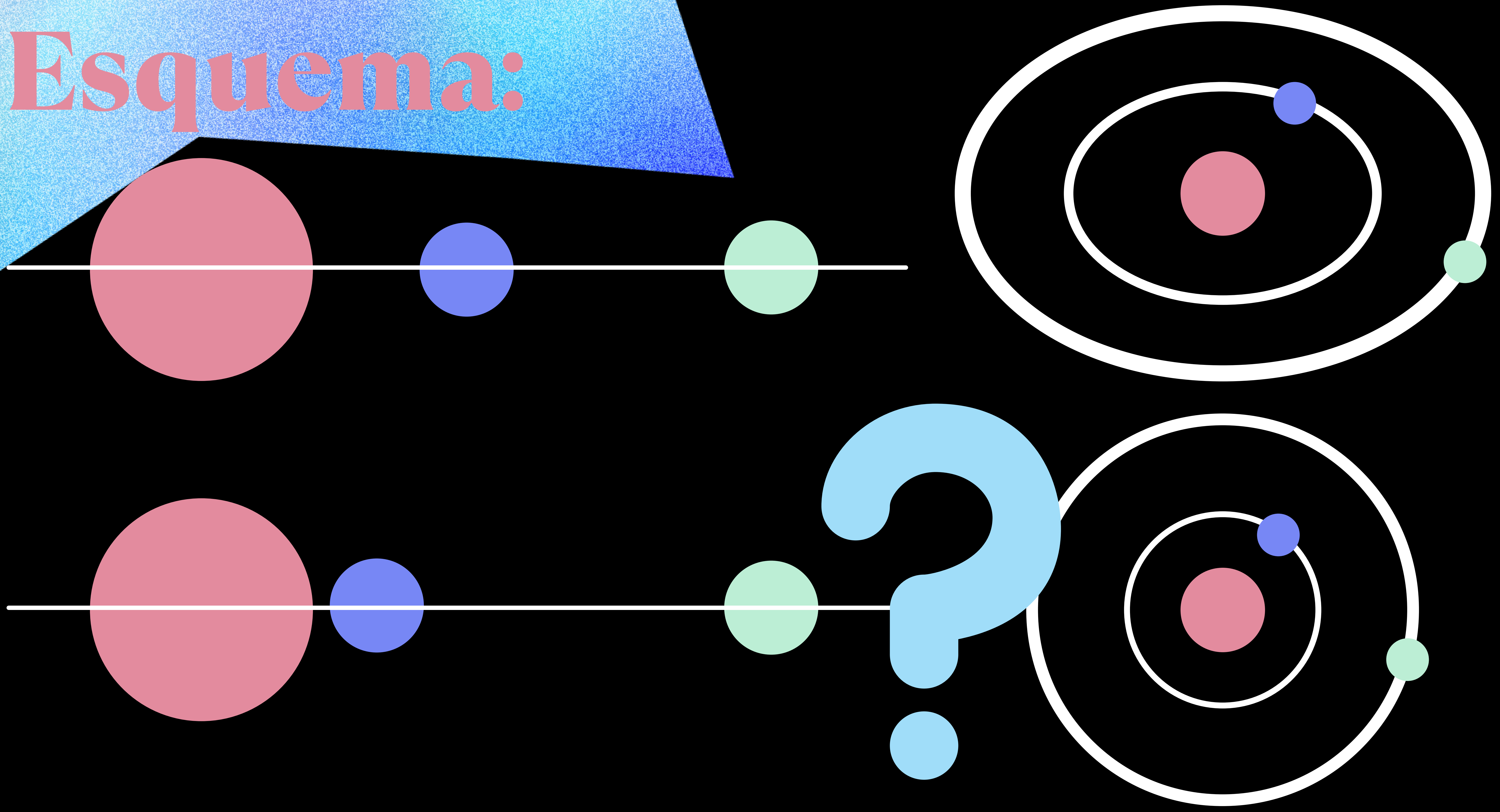
$$\overline{W_q} = \frac{3}{4} \left(\frac{a_p}{a_c}\right)^3 \left(\frac{m_c}{m_*}\right) \varepsilon_c^{-3} \left[1 - \sqrt{\frac{a_p}{a_c} \frac{m_p}{m_c}} \varepsilon_c^{-1} + \gamma \varepsilon_c^3\right]$$

$$\varepsilon_c = \sqrt{1 - e_c^2}$$
$$\eta \equiv \varpi_p - \varpi_c$$

Objetivos:

- Crear varios sistemas con una estrella central como el Sol y dos planetas.
- Integrar numéricamente las ecuaciones medias para las variaciones de semieje y excentricidad.
- Variar las condiciones iniciales y masas de los planetas.

Esquema:



MÉTODO DE INTEGRACIÓN

Runge - Kutta

Partimos de una ecuación diferencial de la forma:

$$\dot{x} = \frac{dx(t)}{dt} = f(x,t)$$

x(t): solución

Desarrollo de Taylor

$$x(t+h) = x(t) + h\frac{dx}{dt} + \frac{1}{2}h^2\frac{d^2x}{dt^2} + \ \ldots$$

$$x(t) + hf(x,t) + O(h^2)$$

h pequeño -> error pequeño

Orden 1 o método de Euler

Tomamos la expansión en t + 1/2 h para obtener x(t+h)

$$x(t+h) = x\left(t+\frac{1}{2}h\right) + \frac{h}{2}\left(\frac{dx}{dt}\right)_{t+\frac{1}{2}h} + \frac{1}{8}h^2\left(\frac{d^2x}{dt^2}\right)_{t+\frac{1}{2}h} + O(h^3)$$

También podemos derivar una expresión para x(t):

$$x(t) = x\left(t+\frac{1}{2}h\right) - \frac{h}{2}\left(\frac{dx}{dt}\right)_{t+\frac{1}{2}h} + \frac{1}{8}h^2\left(\frac{d^2x}{dt^2}\right)_{t+\frac{1}{2}h} + O(h^3)$$

Restando y reordenando los términos tenemos:

$$x(t+h) = x(t) + hf\left(x\left(t+\frac{1}{2}h\right), t+\frac{1}{2}\right) + O(h^3)$$

No conocemos x(t+1/2h)

Para resolver este problema usamos el método de Euler

$$x\left(t+\frac{1}{2}h\right)=x(t)+\frac{1}{2}hf(x,t)$$

Entonces el cálculo completo para un solo paso resulta:

$$k_1=hf(x,t)$$
$$k_2=hf\left(x+\frac{1}{2}k_1,t+\frac{1}{2}h\right)$$
$$x(t+h)=x(t)+k_2$$

Runge - Kutta de orden 2

Error como $O(h^3)$

Llegamos al orden 4

Realizando desarrollos de Taylor alrededor de varios puntos y tomando las combinaciones lineales correctas podemos eliminar distintos órdenes de errores.

$$k_1 = hf(x,t)$$
$$k_2 = hf\left(x + \frac{1}{2}k_1, t + \frac{1}{2}h\right)$$
$$k_3 = hf\left(x + \frac{1}{2}k_2, t + \frac{1}{2}h\right)$$
$$k_4 = hf(x + k_3, t + h)$$
$$x(t+h) = x(t) + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

Error de h^5

RESULTADOS

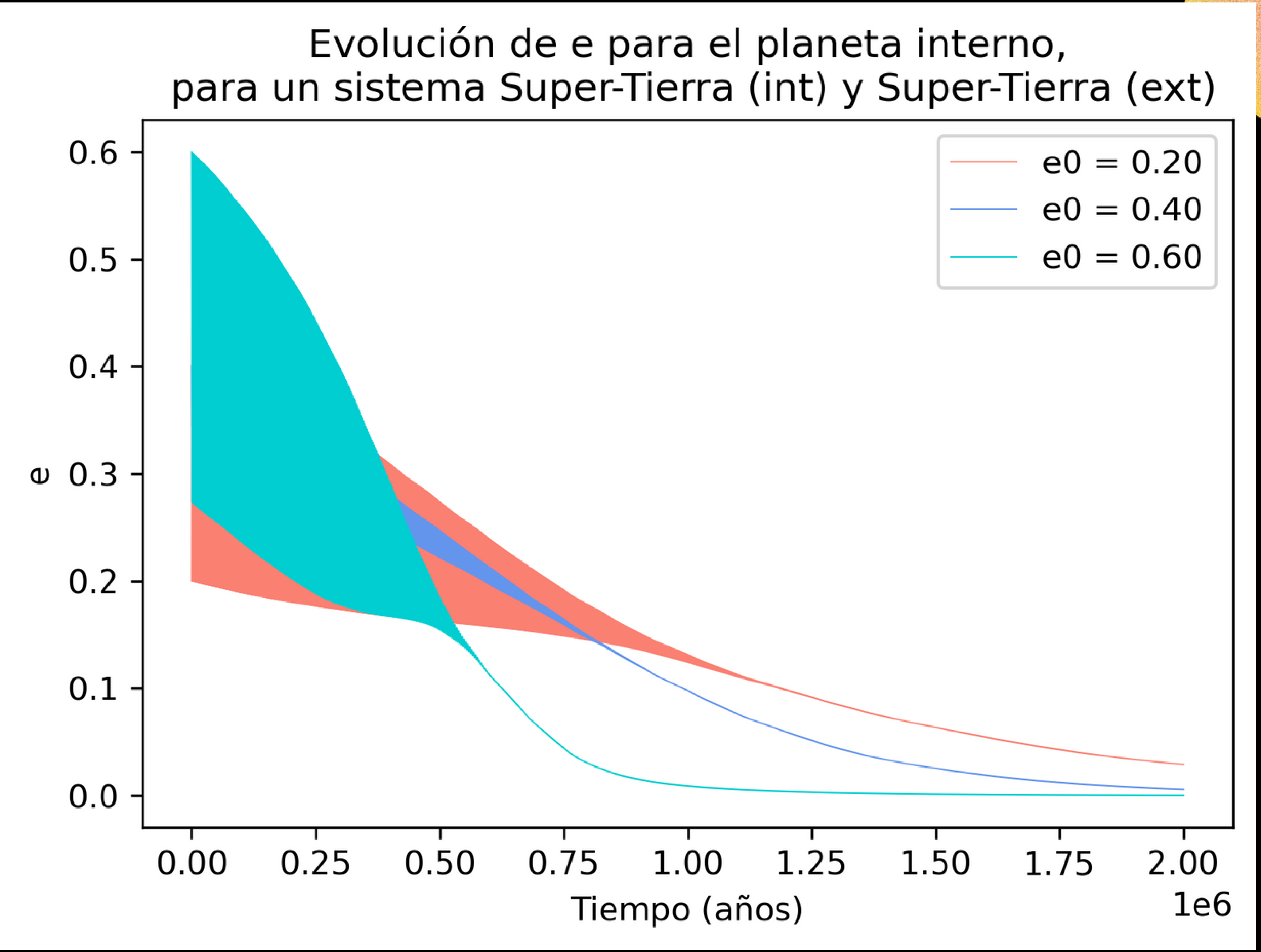
Dos Super - Tierras

Sistema similar al CoRoT - 7

Estrella igual al sol.

Planeta interno: R = 1.58 Rt
M = 8 Mt
a0 = 0.017 ua

Planeta externo: M = 13.6 Mt
a0 = 0.046 ua
e0 = 0.4



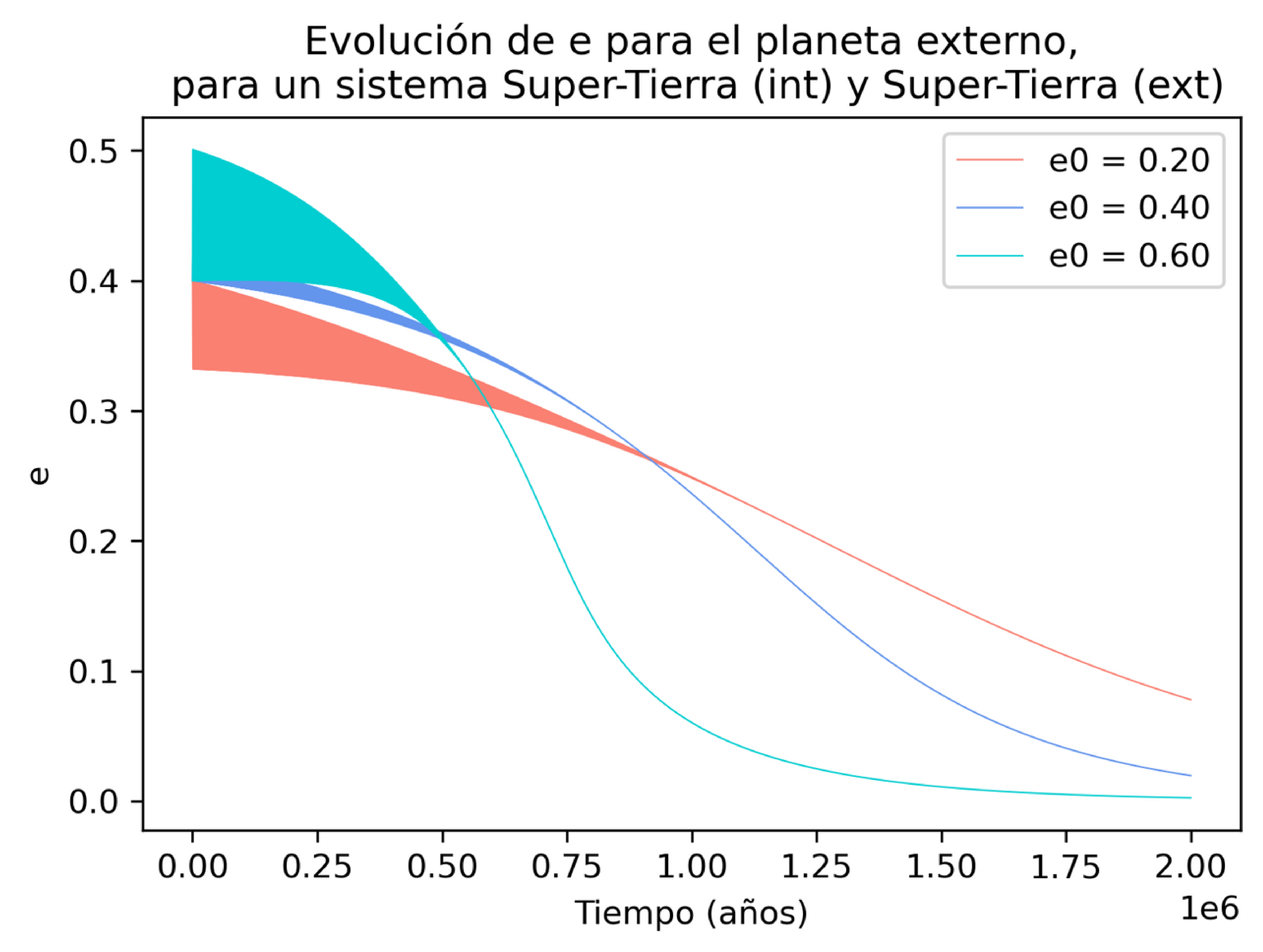
Dos Super - Tierras

Sistema similar al CoRoT - 7

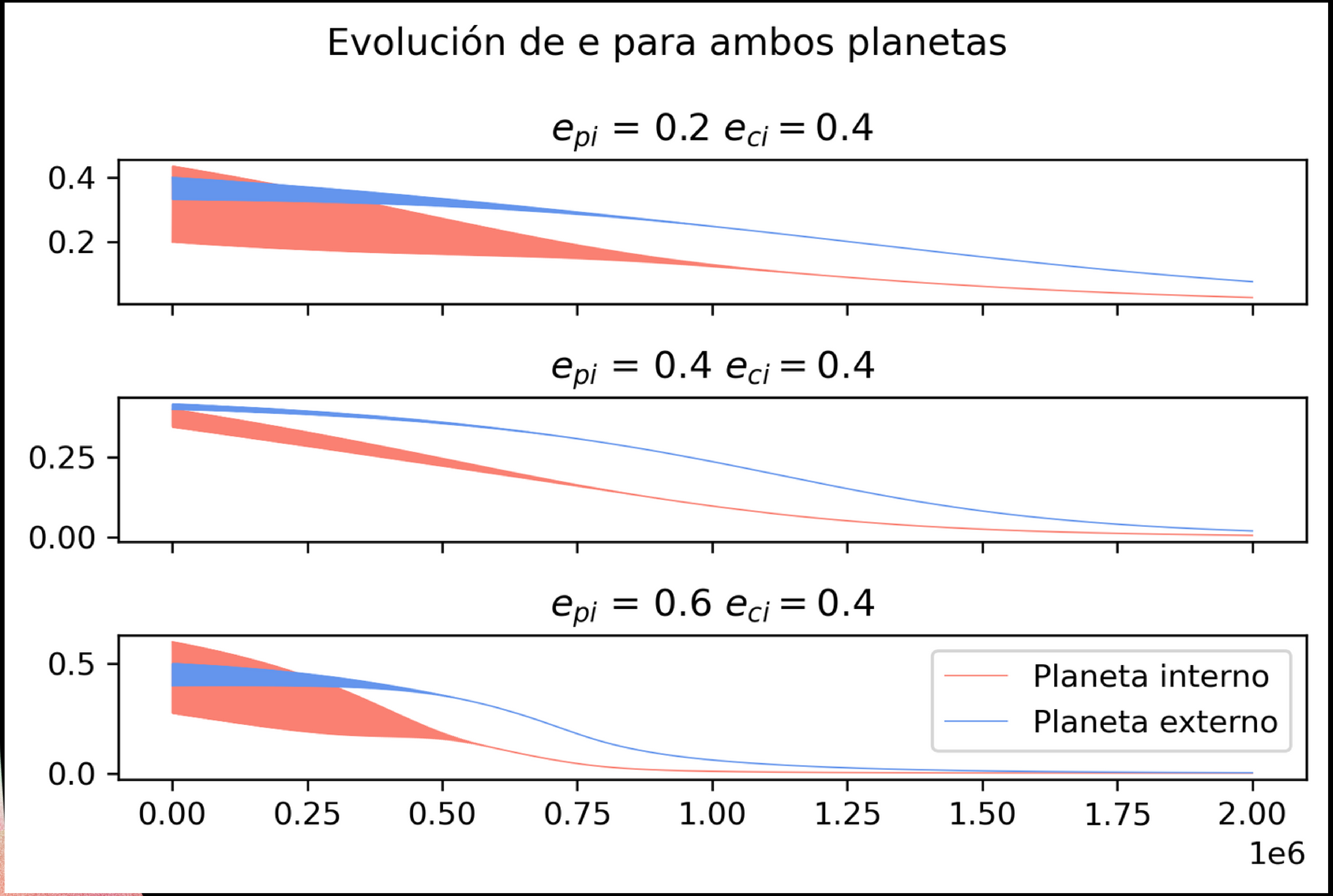
Estrella igual al sol.

Planeta interno: R = 1.58 Rt
M = 8 Mt
a0 = 0.017 ua

Planeta externo: M = 13.6 Mt
a0 = 0.046 ua
e0 = 0.4



Dos Super - Tierras: comparación



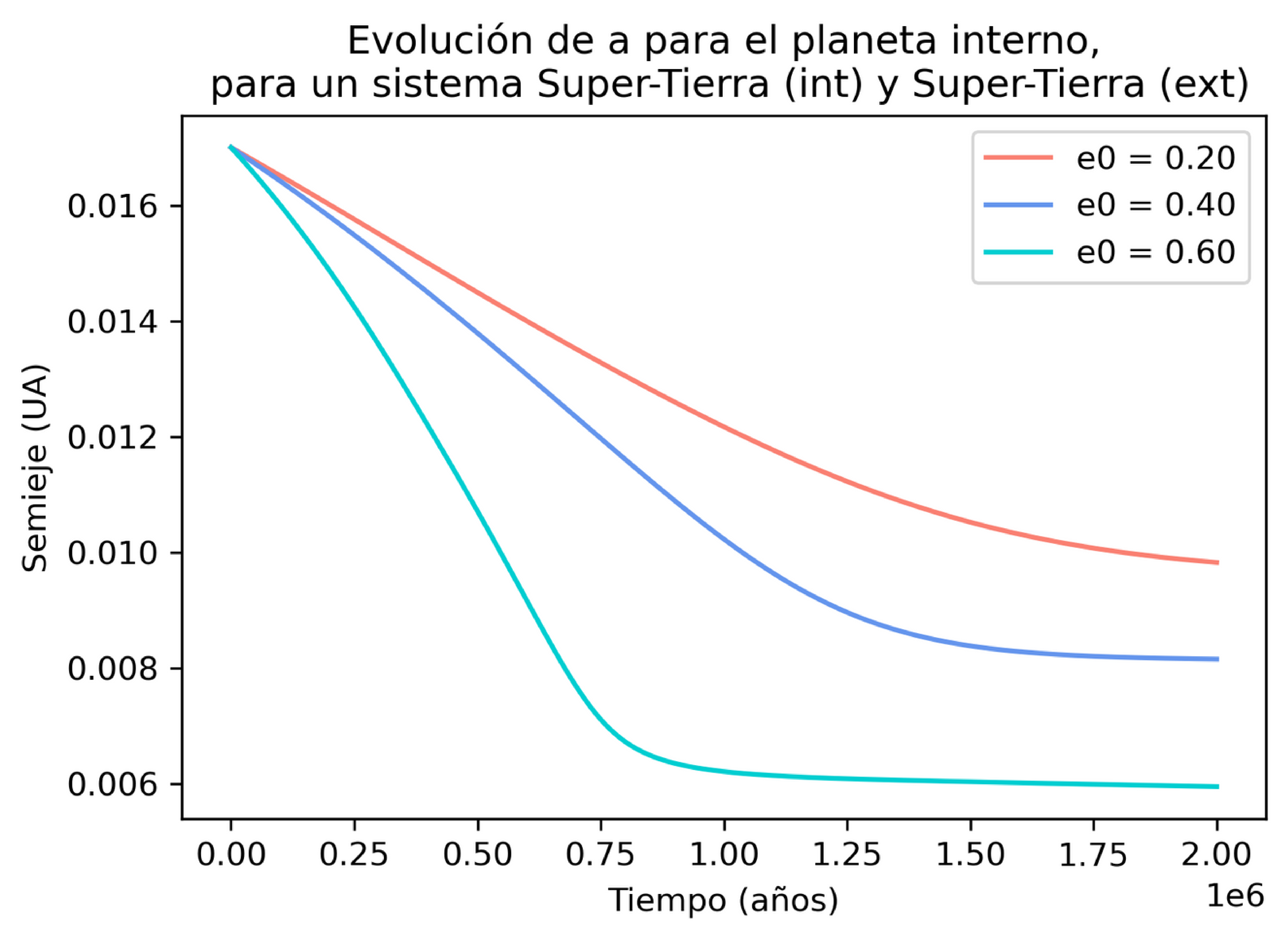
Dos Super - Tierras

Sistema similar al CoRoT - 7

Estrella igual al sol.

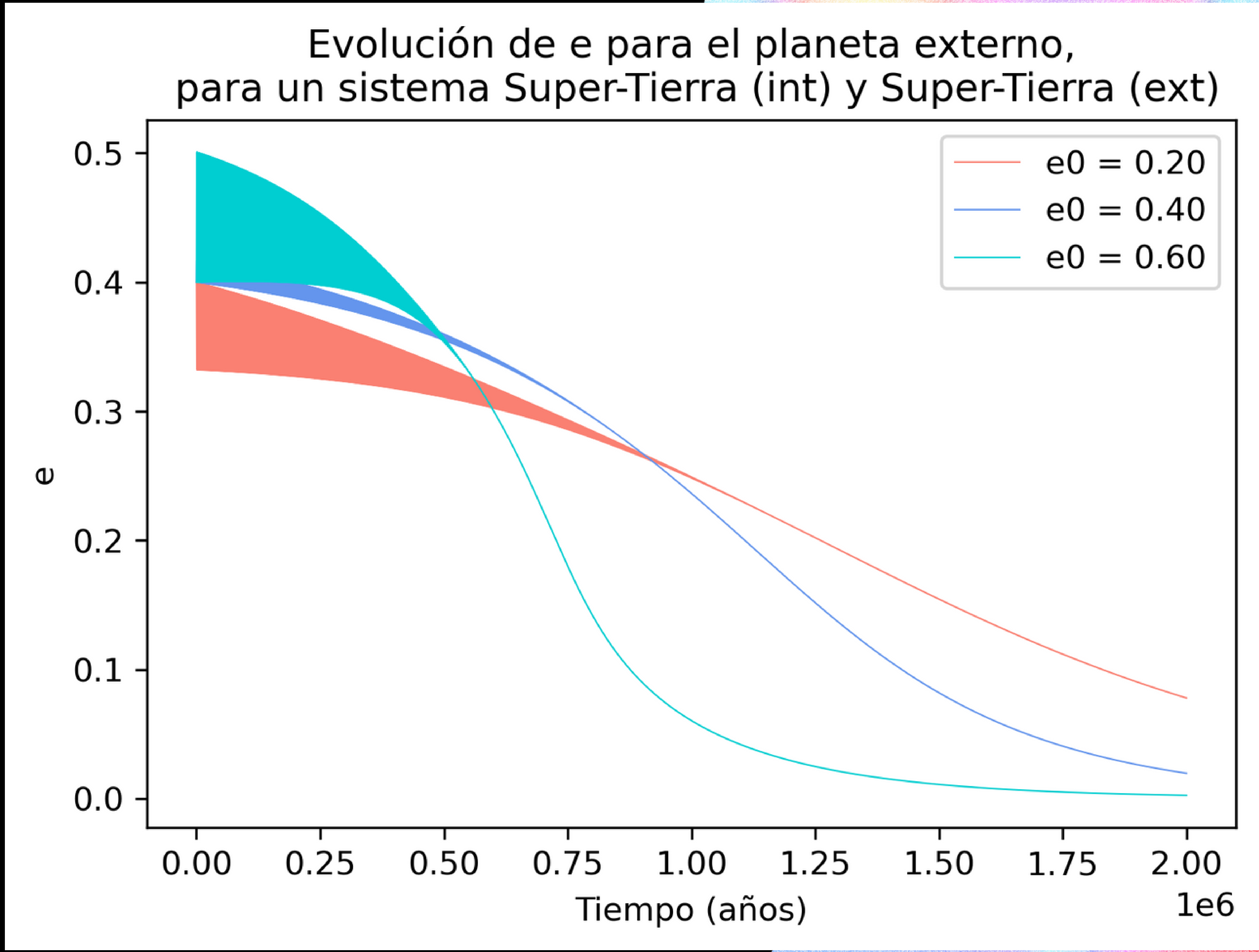
Planeta interno: R = 1.58 Rt
M = 8 Mt
a0 = 0.017 ua

Planeta externo: M = 13.6 Mt
a0 = 0.046 ua
e0 = 0.4



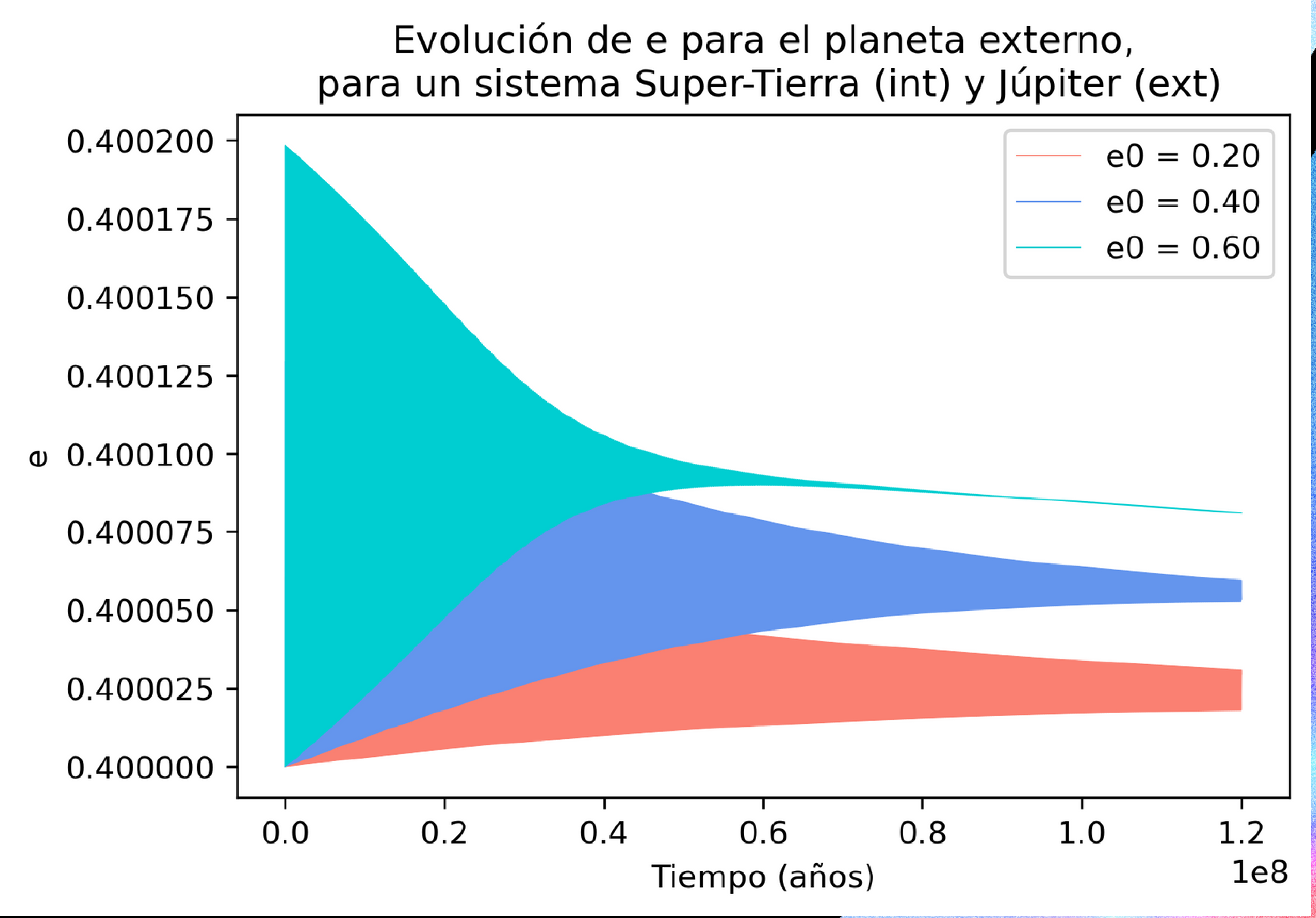
Super - Tierra y Júpiter

Estrella igual al sol.
Planeta interno: $R = 5^{1/3} Rt$
$M = 5\ Mt$
$a0 = 0.04$ ua
Planeta externo: $M = 1\ Mj$
$a0 = 0.1$ ua
$e0 = 0.4$

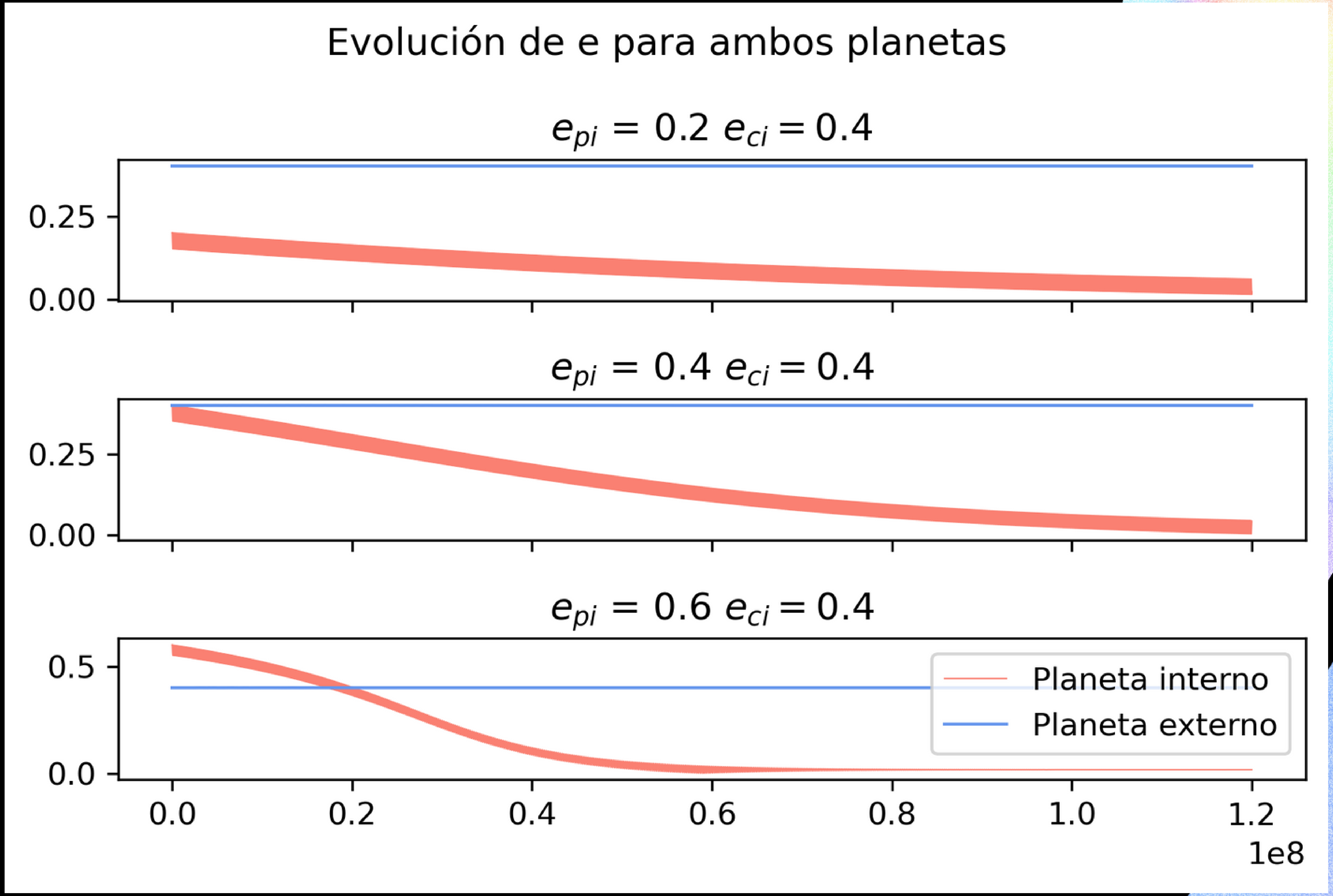


Super - Tierra y Júpiter

Estrella igual al sol.
Planeta interno: $R = 5^{1/3}Rt$
$M = 5\ Mt$
$a0 = 0.04$ ua
Planeta externo: $M = 1\ Mj$
$a0 = 0.1$ ua
$e0 = 0.4$



Super - Tierra y Júpiter: comparación



Super - Tierra y Júpiter

Estrella igual al sol.

Planeta interno: $R = 5^{1/3} Rt$

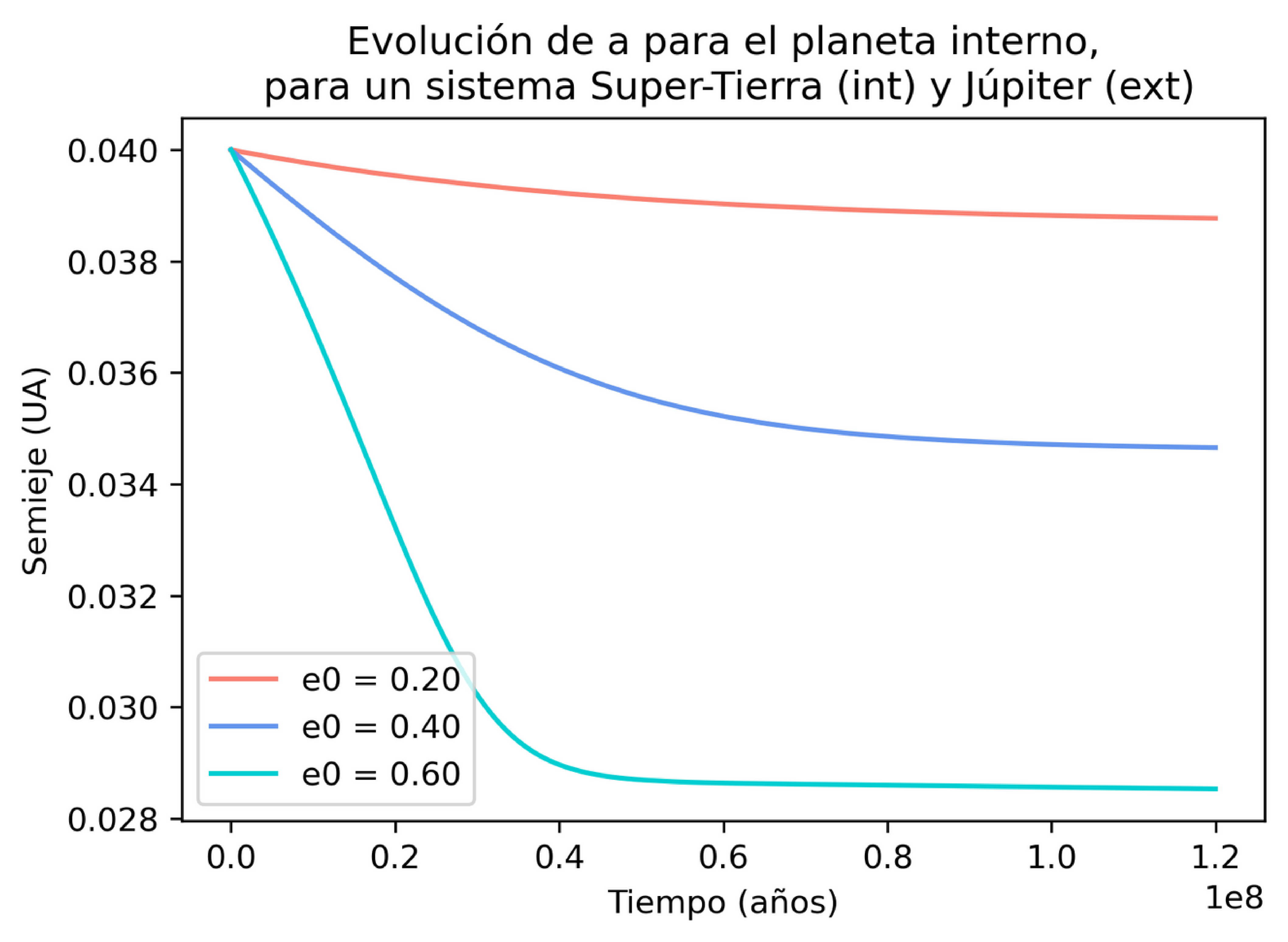
$M = 5$ Mt

$a0 = 0.04$ ua

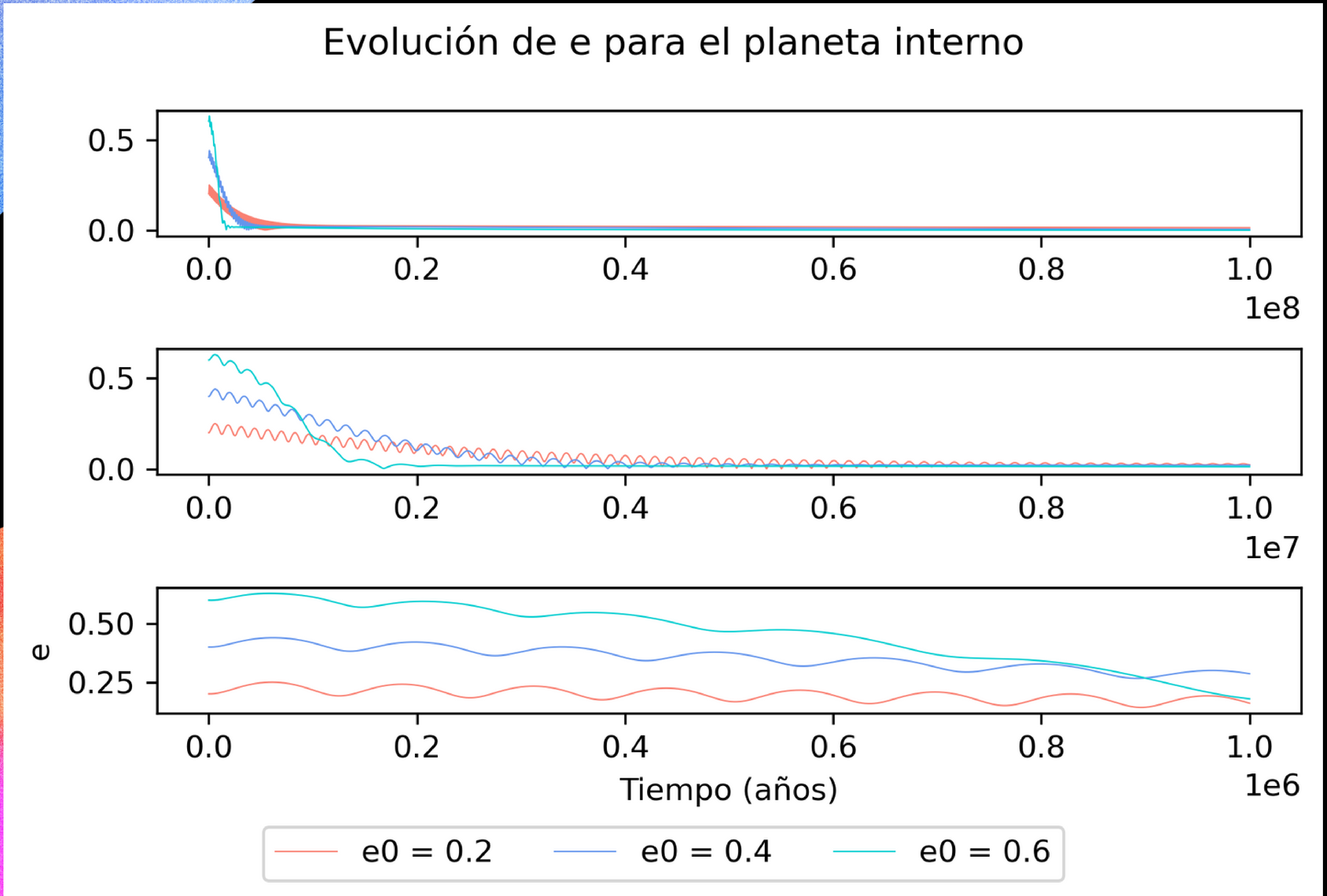
Planeta externo: $M = 1$ Mj

$a0 = 0.1$ ua

$e0 = 0.4$



Júpiter y Neptuno



Estrella igual al sol.

Planeta interno:

R = 2 Rj

M = 0.46 Mj

a0 = 0.05 ua

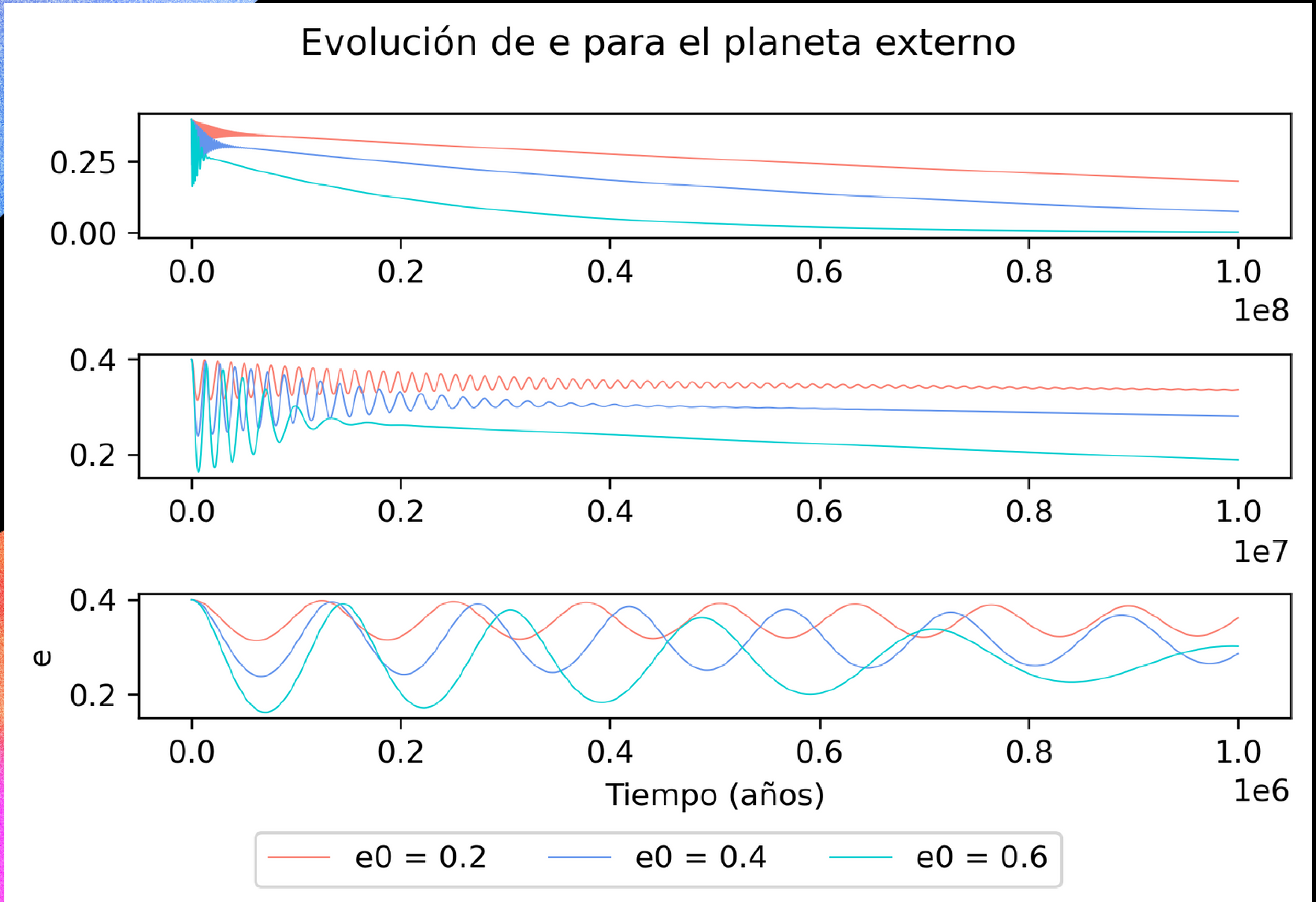
Planeta externo:

M = Mn

a0 = 0.5 ua

e0 = 0.04

Júpiter y Neptuno



Estrella igual al sol.

Planeta interno:

R = 2 Rj

M = 0.46 Mj

a0 = 0.05 ua

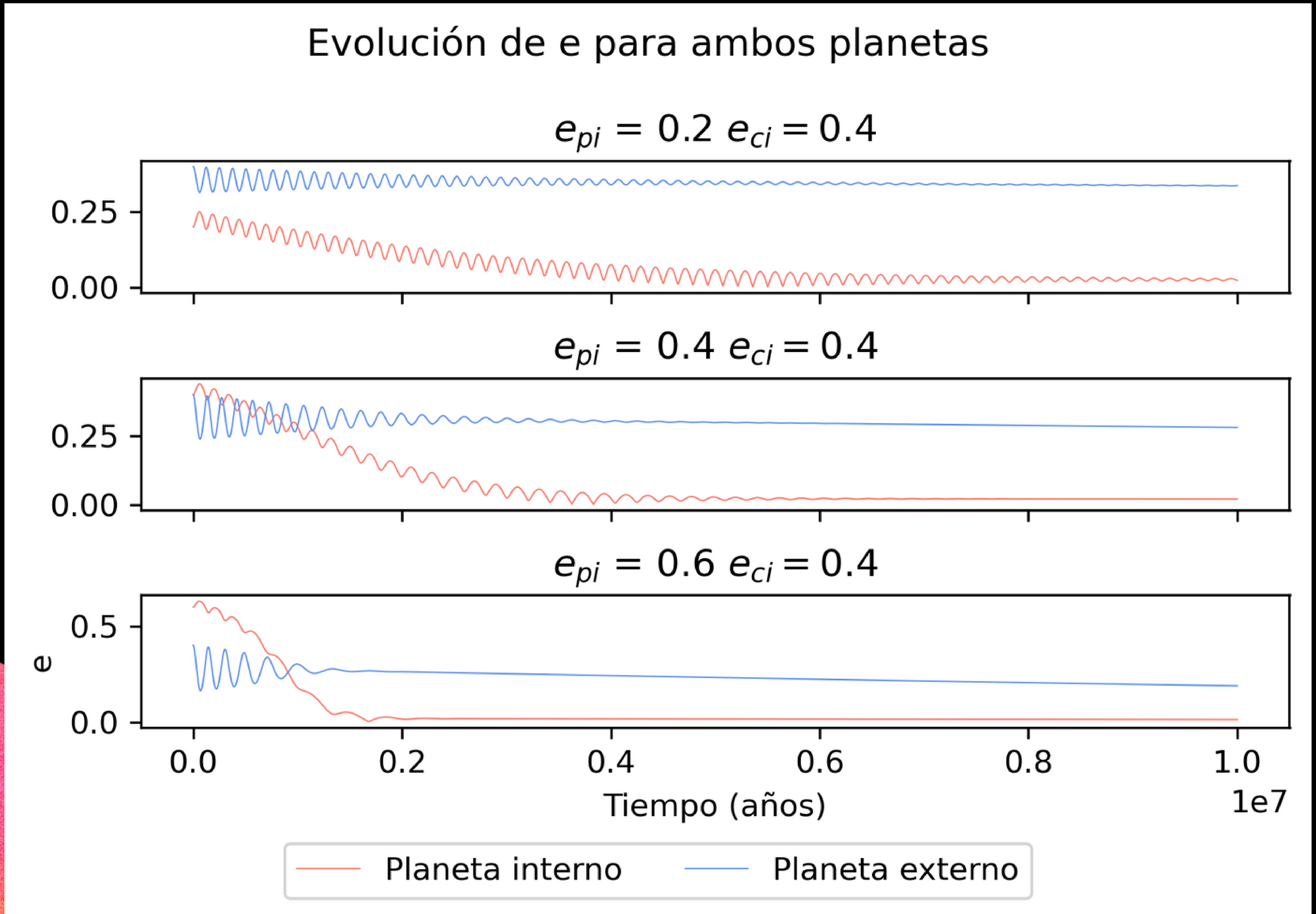
Planeta externo:

M = Mn

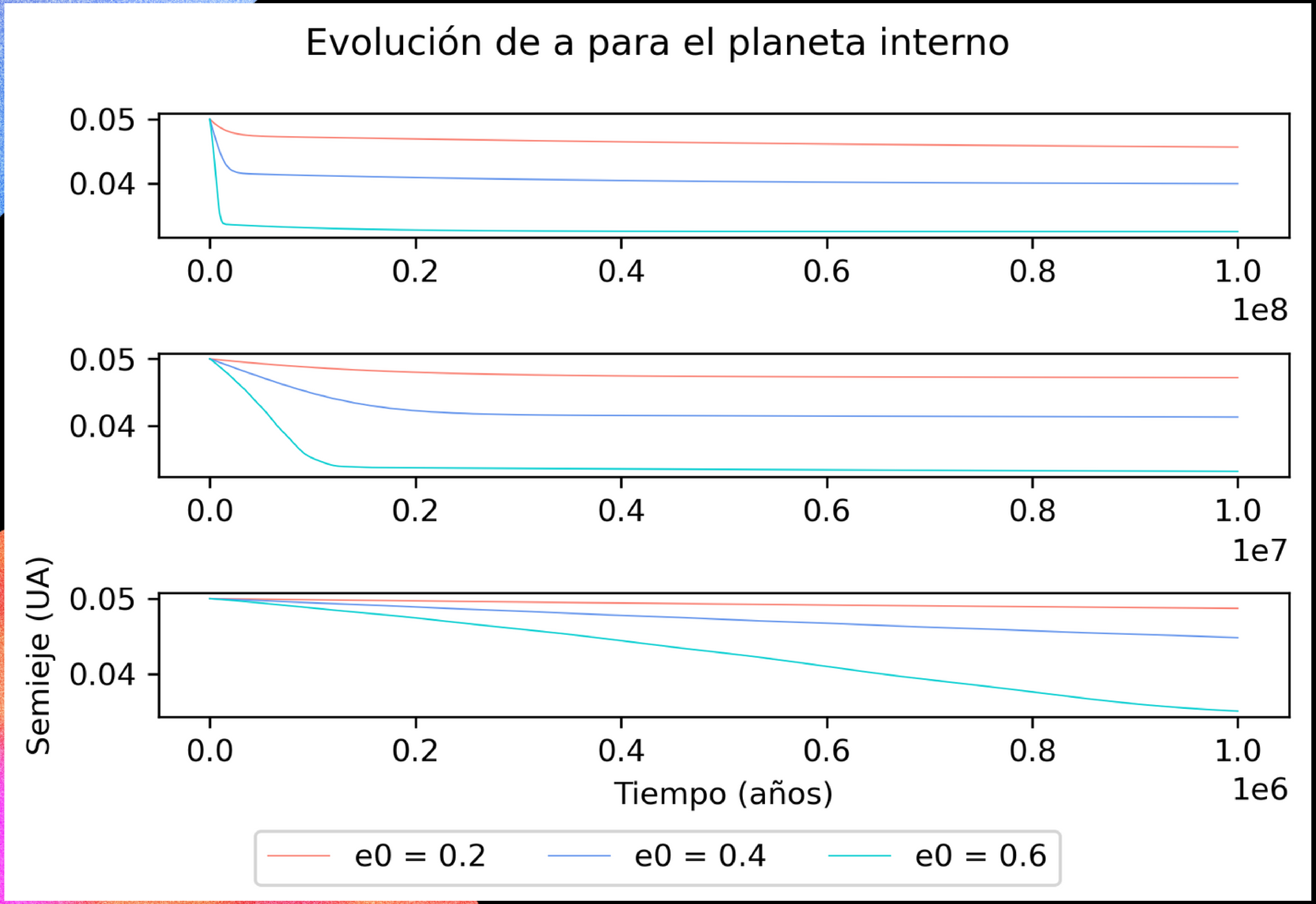
a0 = 0.5 ua

e0 = 0.04

Júpiter y Neptuno: comparación



Júpiter y Neptuno



Estrella igual al sol.

Planeta interno:

R = 2 Rj

M = 0.46 Mj

a0 = 0.05 ua

Planeta externo:

M = Mn

a0 = 0.5 ua

e0 = 0.04

CONCLUSIONES

Dos Super - Tierras

- Resultados similares al trabajo de Rodríguez (2011).
  Circularización de ambas órbitas en millones de años.

Super - Tierra y Júpiter

- Resultados similares al trabajo de Rodríguez (2011).
  Circularización del planeta externo muy lenta.

Júpiter caliente y Neptuno

- Circularización completa en más de 100 millones de años.
  Circularización del planeta interno mucho más rápida que el externo.
  Buen funcionamiento del método.

- Cuánto mayor es la excentricidad inicial del planeta interno más rápida es la evolución orbital del sistema.
- La excentricidad del planeta interno decae más rápido.
- La evolución del semieje mayor depende fuertemente de la excentricidad inicial (cuánto mayor es *e* la caída de *a* es mayor).
- El sistema más rápido en evolucionar es el de dos Super - Tierras.

Todas las órbitas se circularizan y el planeta interno migra hacia la estrella como esperábamos.

Método Runge - Kutta 4

- Puede reproducir resultados correctos (reproducción del trabajo de otros autores).
- Cálculos rápidos que no consumen mucho para la computadora.
- Búsqueda del par ideal de tiempo de integración y número de iteraciones para un correcto funcionamiento.

MUCHAS GRACIAS

Extractos del código

```
def evolucion(a,b,N,x,mp,mc,Rp,ac,Qp):

    # Función que calcula la evolución orbital de un sistema planetario de dos planetas
    # y una estrella, teniendo en cuenta el efecto de marea sobre la estrella y el
    # planeta más cercano

    # a y b: extremos de integración
    # N: cantidad de pasos
    # x: vector con las condiciones iniciales
    # mp y mc: masas del planeta interno y externo respectivamente
    # Rp: radio del planeta interno
    # ac: semieje mayor del planeta externo
    # Qp: constante de disipación del planeta interno (afectado por marea)

    # Masa reducida del planeta principal y del compañero
    mu = G * (ms + mp)
    muc = G * (ms + mc)

    nc = sqrt(muc/ac**3) # Movimiento medio del compañero

    s = 9/4 * kds/Qs * mp/ms * Rs**5 # Efecto de marea sobre la estrella
    p = 9/2 * kdp/Qp * ms/mp * Rp**5 # Efecto de marea sobre el planeta

    def rk4(a,b,N,x): # Rutina de Runge - Kutta 4
        h = (b-a)/N # Definimos el espaciado temporal
        tp = arange(a,b,h) # Definimos el vector con todos los tiempos utilizados para luego graficar

        # Definimos las diferentes listas para guardar los resultados de la integración
        epp = []
        ecp = []
        etap = []
        app = []
```

```
        def f(x): # Función a integrar
            # A partir del parámetro x tomamos las diferentes variables del problema
            ep = x[0] # Excentricidad del planeta interior
            ec = x[1] # Excentricidad del planeta exterior
            eta = x[2] # Diferencia entre las longitudes del periihelio de ambos planetas (w2 - w1)
            ap = x[3] # Semieje del planeta interior

            np = sqrt(mu/ap**3) # Movimiento medio del planeta interior
            epsilonc = sqrt(1 - ec**2) # Lo definimos para simplificar la escritura de las siguientes ecuaciones

            # Constantes que surgen a partir de las aproximaciones hechas
            Wo = 15/16 * np * (ap/ac)**4 * (mc/ms) * epsilonc**-5
            Wc = 15/16 * nc * (ap/ac)**3 * (mp/ms) * epsilonc**-4
            Wt = 21/2 * np * (kp/Qp) * (ms/mp) * (Rp/ap)**5
            Wq = 3/4 * np * (ap/ac)**3 * mc/ms * epsilonc**-3 * (1 - sqrt(ap/ac) * mp/mc * epsilonc**-1 + gamma * epsilonc**3)

            # Las diferentes ecuaciones diferenciales a integrar
            dep = -Wo * ec * sin(eta) - Wt * ep
            dec = Wc * ep * sin(eta)
            deta = Wq - Wo * ec/ep * cos(eta)
            dap = -(2/3) * np * (ap**-4) * ((2 + 46 * ep**2) * s + 7 * ep**2 *p)

            return array([dep,dec,deta,dap],float) # Devolvemos los valores de las ecuaciones diferenciales (derivadas)
        # Comenzamos a integrar utilizando el método propuesto (rk4)
        for t in tp:
            epp.append(x[0])
            ecp.append(x[1])
            etap.append(x[2])
            app.append(x[3])

            k1 = h * f(x)
            k2 = h * f(x + 0.5 * k1)
            k3 = h * f(x + 0.5 * k2)
            k4 = h * f(x + k3)
            x += (k1 + 2 * k2 + 2 * k3 + k4) / 6

        return array([epp,ecp,app,tp],float) # Devolvemos las listas que nos interesan para el posterior análisis y gráficas

    epp, ecp, app, tp = rk4(a,b,N,x) # Llamamos a la función que integra

    return epp, ecp, app, tp # Devolvemos las listas que nos interesan
```

```python
###############################################################################
########################## Definición de constantes ###########################
###############################################################################

# Trabajamos con unidades solares para masa, unidades astronómicas para distancia y años para tiempo

# Masas en masas solares
ms = 1
mj = ms * 9.547919 * 1e-4 # datos de Júpiter
mt = ms * 1/333000 # datos de la Tierra

# Radios en unidades astronómicas
Rs = 0.00465047
Rj = 0.10045 * Rs # datos de Júpiter
Rt = 0.00914927 * Rs # datos de la Tierra

G = 4*pi**2 # Constante de gravitación en ua, años y masas solares

# Constantes referentes a la disipación
kp = 1
kds = 1
kdp = 1

Qs = 1e7 # Constante de disipación para una estrella como el Sol

gamma = 0 # Constante referente a la relatividad, no asumo efectos relativistas

a = 0 # Tiempo incial de integración, es el mismo para todos los casos
```