

# DINÁMICA MOLECULAR

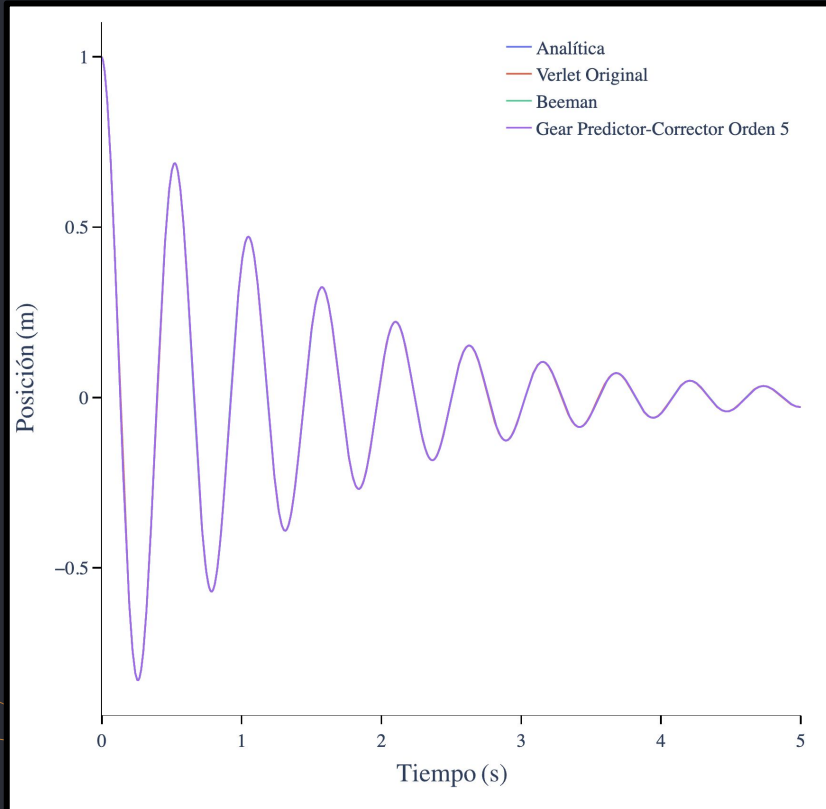
"Regida por el paso temporal"

SS - G9 - TP4 - 2022

# Oscilador Amortiguado

# OSCILADOR AMORTIGUADO

## COMPARACIÓN CON SOLUCIÓN ANALÍTICA

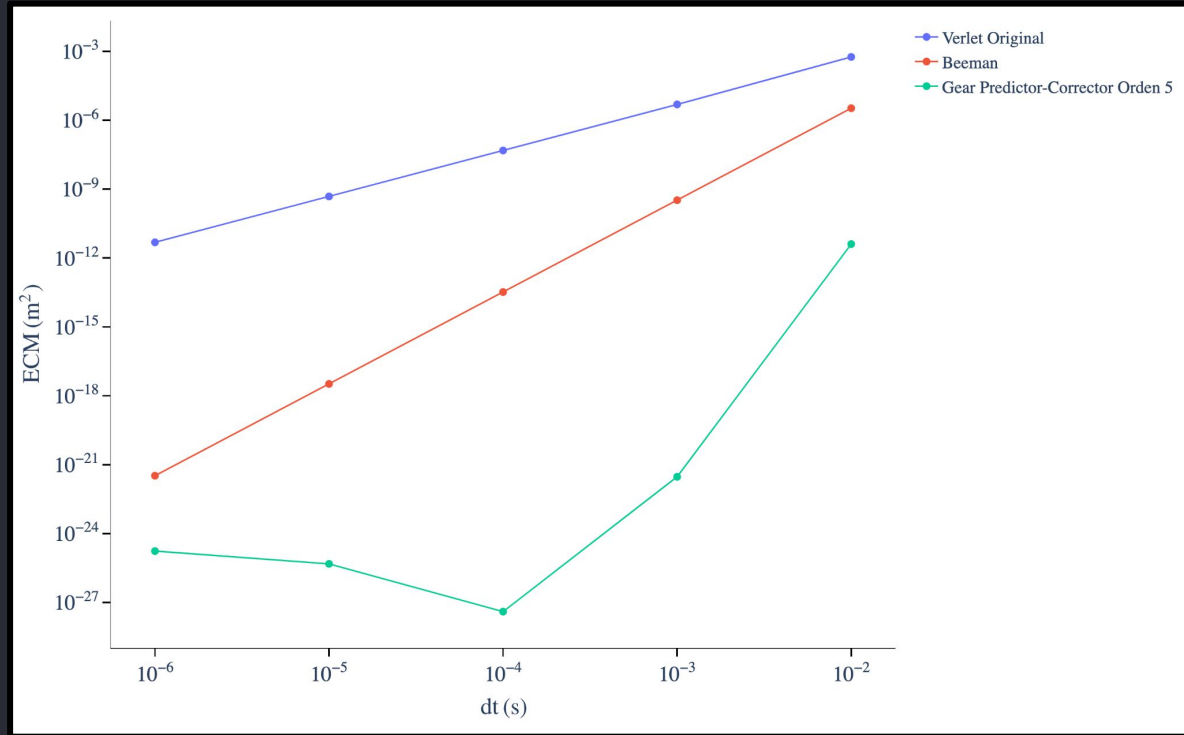


Algoritmo	ECM (m <sup>2</sup> )
Verlet Original	$4.89 * 10^{-6}$
Beeman	$3.30 * 10^{-10}$
Gear Predictor-Corrector Orden 5	$2.96 * 10^{-22}$

$$dt = 1 * 10^{-3} s$$

# OSCILADOR AMORTIGUADO

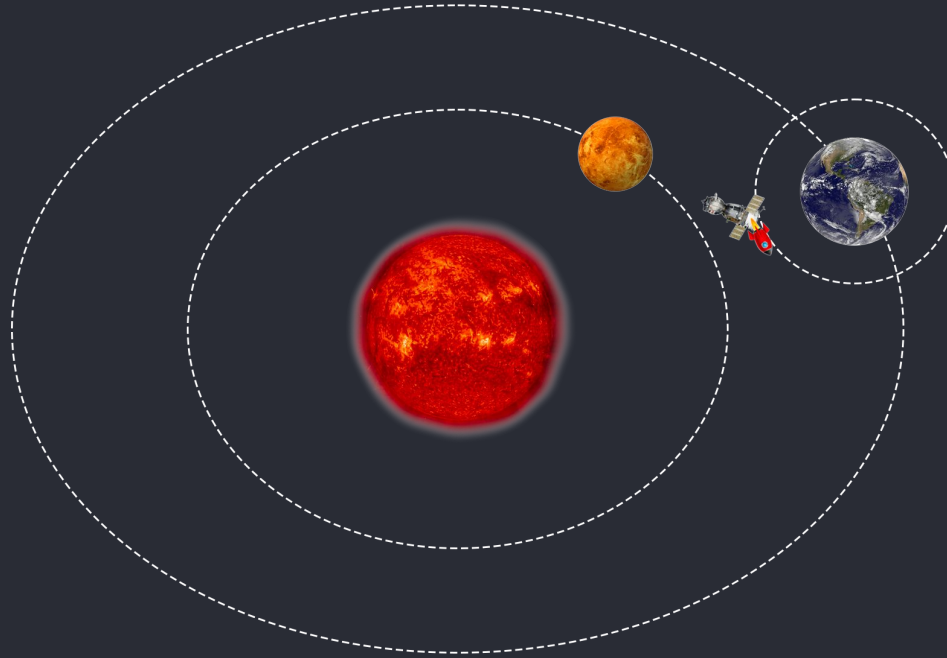
## ERROR CUADRÁTICO MEDIO



# Introducción

# INTRODUCCIÓN SISTEMA

- Sistema de cuerpos atraídos entre sí por fuerzas gravitacionales.



# INTRODUCCIÓN MODELO

- N partículas.
- Choques no instantáneos.
- Simulación regida por paso temporal.
- Interacción mediante fuerzas dependiente de distancia, según:

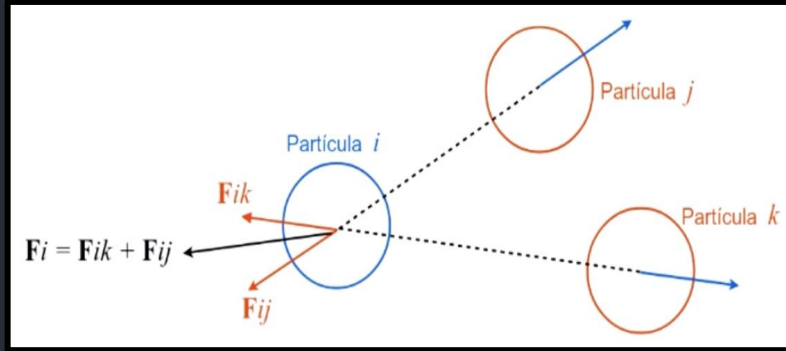
$$F_{ij} = G \frac{m_i m_j}{r_{ij}^2} e_{ij}$$

Donde:

- $G = 6.693 * 10^{-11} \text{ m}^3 / \text{kg} * \text{s}^2$ .

# INTRODUCCIÓN MODELO

## Fuerza de interacción



$$F_i^{Tot} = \sum_j F_{N_{ij}}$$

$$F_{iy}^{Tot} = \sum_j F_{N_{ij}} e_{y_{ij}}^n$$

$$F_{ix}^{Tot} = \sum_j F_{N_{ij}} e_{x_{ij}}^n$$

Donde:

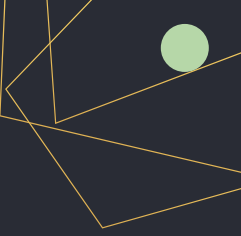
$$e_x^n = \frac{(x_j - x_i)}{|r_j - r_i|}$$

$$e_y^n = \frac{(y_j - y_i)}{|r_j - r_i|}$$



# INTRODUCCIÓN

## MODELO



Energía potencial gravitatoria

$$E_{ij}^{pot} = -G \frac{m_i m_j}{r_{ij}}$$

Energía cinética

$$E_{c_i} = \frac{1}{2} m_i v_i^2$$

- Sistema conservativo.



# Implementación

# IMPLEMENTACIÓN LENGUAJE

- Motor de simulación: Java.
- Post procesamiento: Python.



# IMPLEMENTACIÓN

## CLASES DESTACADAS

1

### Particle

**p** **id** **int**  
**p** **mass** **double**  
**p** **radius** **double**

3

### R

**m** **get(int)** **Pair**  
**m** **set(int, double, double)** **void**  
**m** **add(double, double)** **void**

2

### State

**p** **position** **Position**  
**p** **velocityX** **double**  
**p** **velocityY** **double**

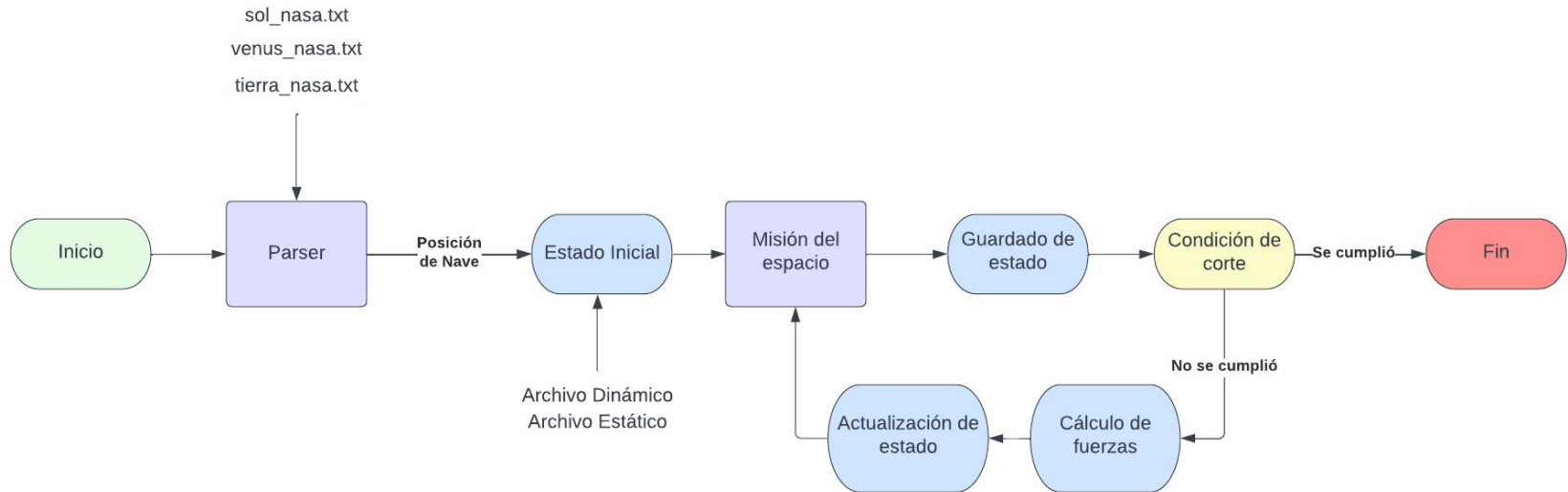
4

### SpaceMission

**m** **calculateInitialRs**(Map<Particle, State>) **Map<Particle, R>**  
**m** **getDeltaR2**(Map<Particle, R>, **double**) **Map<Particle, Pair>**  
**m** **execute**(Map<Particle, State>, **double, double**) **AlgorithmResults**  
**m** **calculateAcceleration**(Particle, Pair, Map<Particle, R>) **Pair**  
**m** **storeStates**(Map<Double, Map<Particle, State>>, Map<Particle, R>, **double**)  
**m** **correct**(Map<Particle, R>, Map<Particle, Pair>, **double**) **Map<Particle, R>**  
**m** **predict**(Map<Particle, R>, **double**) **Map<Particle, R>**

# IMPLEMENTACIÓN

## FLUJO DE EJECUCIÓN



# Simulaciones

# SIMULACIÓN

## PARÁMETROS E INPUTS

### Parámetros

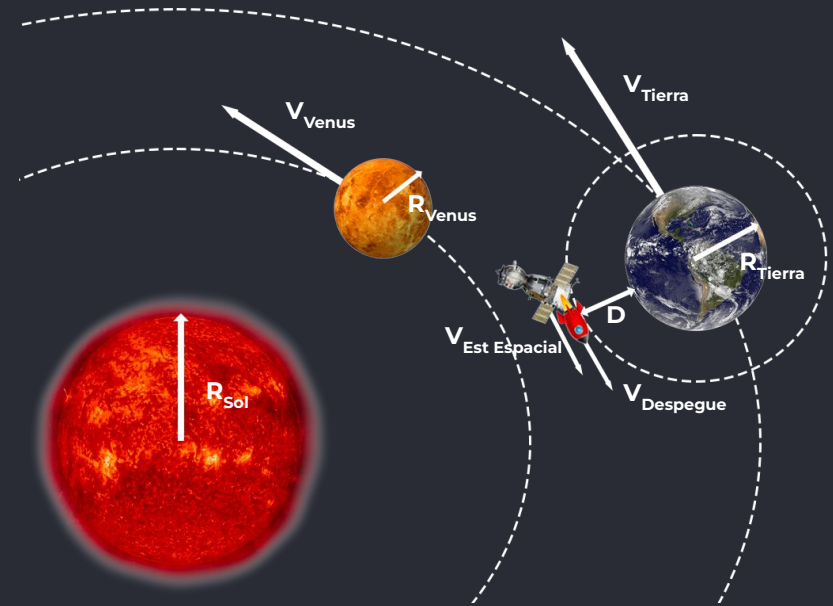
- $m_{\text{Sol}} = 1.989 \times 10^{30} \text{ kg.}$
- $R_{\text{Sol}} = 6.96 \times 10^5 \text{ km.}$
- $m_{\text{Tierra}} = 5.97 \times 10^{24} \text{ kg.}$
- $R_{\text{Tierra}} = 6371.01 \text{ km.}$
- $m_{\text{Venus}} = 48.69 \times 10^{23} \text{ kg.}$
- $R_{\text{Venus}} = 6051.84 \text{ km.}$
- $m_{\text{Nave}} = 2 \times 10^5 \text{ kg.}$
- $V_{\text{Despegue}} = 8 \text{ km/s.}$
- $V_{\text{Est. Espacial}} = 7.12 \text{ km/s.}$
- $D = 1500 \text{ km.}$

### Inputs

- $(X_0, Y_0)$  planetas.
- $(V_x, V_y)$  planetas.

### Outputs

- $(X, Y)$
- $(V_x, V_y)$



# SIMULACIÓN OBSERVABLES

- Mínima distancia al planeta destino.
- Tiempo de viaje.
- Energía de la nave.
- Velocidad relativa de la nave respecto al planeta destino al arribar.
- Evolución temporal del módulo de la velocidad de la nave.



# SIMULACIÓN

## ANÁLISIS: FECHA DE DESPEGUE

- Fecha de despegue que asegure arribo a Venus.
  - **Misión exitosa:** alcanzar órbita (distancia menor a 1500 km) en menos de un año.
- **Análisis 1:** despegue cada 1 día en intervalo [23/09/2022 - 23/09/2023] (1 año).
- **Análisis 2:** despegue cada 30 minutos centrado en fecha óptima del análisis 1.
- **Análisis 3:** despegues cada 5 minutos en fecha óptima del análisis 2.
- **Paso temporal:**  $dt = 300 \text{ s}$  (5 minutos).
- **Fecha óptima:** arribar Venus en menor cantidad de días.

SIMULACIÓN

# ANÁLISIS: VARIACIÓN DE LA VELOCIDAD DE DESPEGUE

- Variación velocidad de despegue en intervalo  $[4, 12]$  km/s.
- **Objetivo:** optimizar tiempo de viaje.

## SIMULACIÓN

# ANÁLISIS: FECHA DE REGRESO A LA TIERRA

- Fecha de despegue que asegure regreso a la Tierra.
  - **Misión exitosa:** alcanzar órbita (distancia menor a 1500 km) en menos de un año.
- **Análisis 1:** despegue cada 1 día en intervalo [18/07/2023 - 16/02/2025] (2 años).
- **Análisis 2:** despegue cada 30 minutos centrado en fecha óptima del análisis 1.
- **Análisis 3:** despegues cada 5 minutos en fecha óptima del análisis 2.
- **Paso temporal:**  $dt = 300 \text{ s}$  (5 minutos).
- Estación espacial a 1500 km de Venus (dirección contraria al sol).
- $V_{\text{Despegue}} = 4.4 \text{ km/s}$ .
- $V_{\text{Est. Espacial}} = 5.8 \text{ km/s}$ .

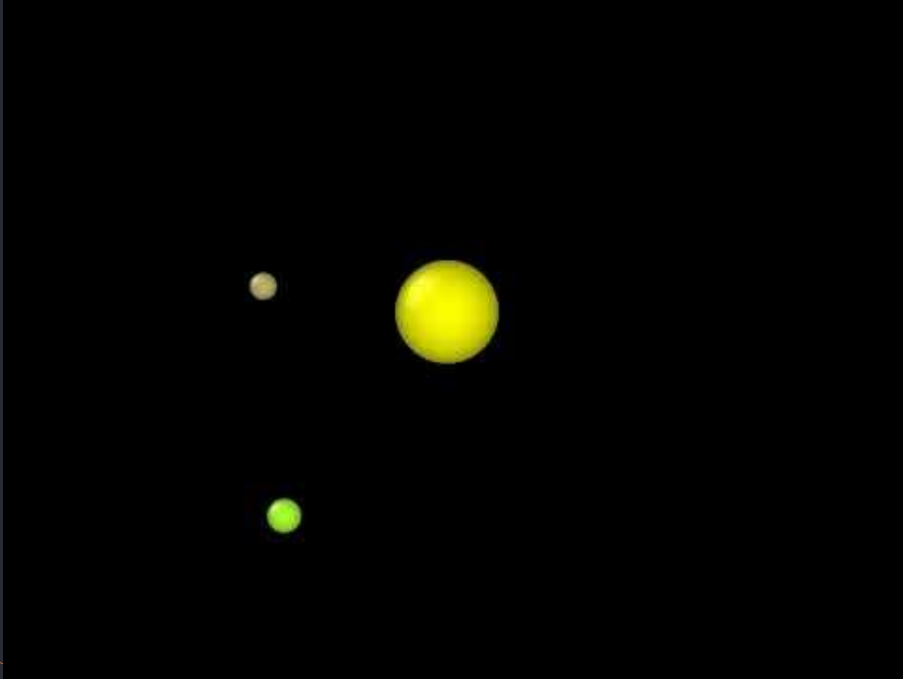
# SIMULACIÓN

## CRITERIOS DE CORTE

- Tiempo arbitrario de simulación (365 días de viaje).
- Choque con planeta destino.

# Resultados

# RESULTADOS | FECHA ÓPTIMA DESPEGUE A VENUS ANIMACIÓN

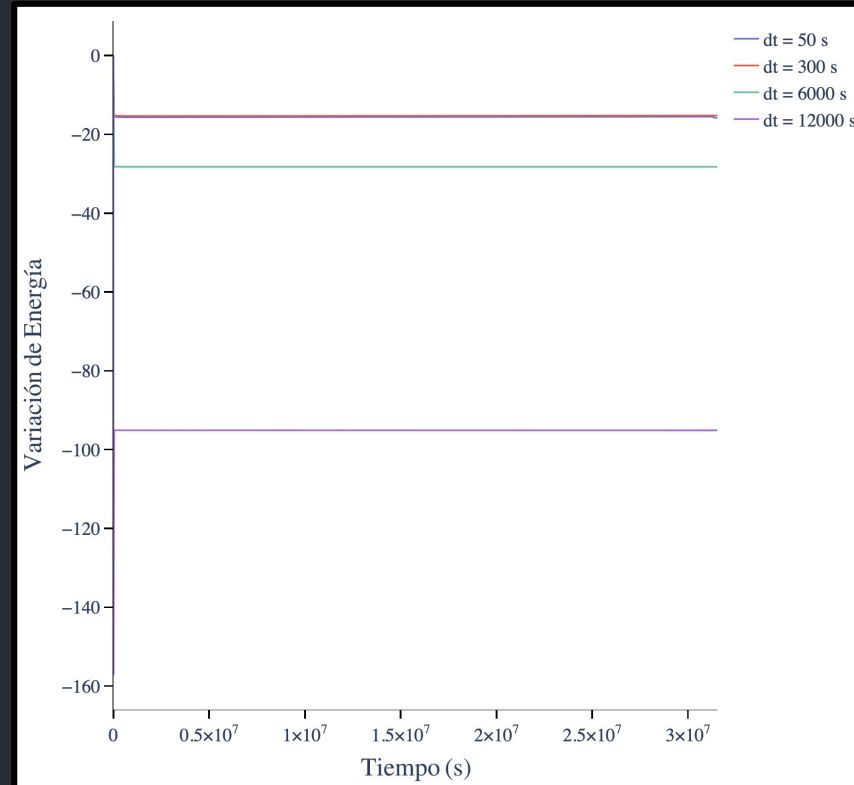


- Fecha de salida: 12-05-2023 11:15 hs.
- $dt = 300$  s.
- Tiempo de viaje: 66 días.

Animación en: <https://www.youtube.com/watch?v=La5lUngjygl&t=1s>

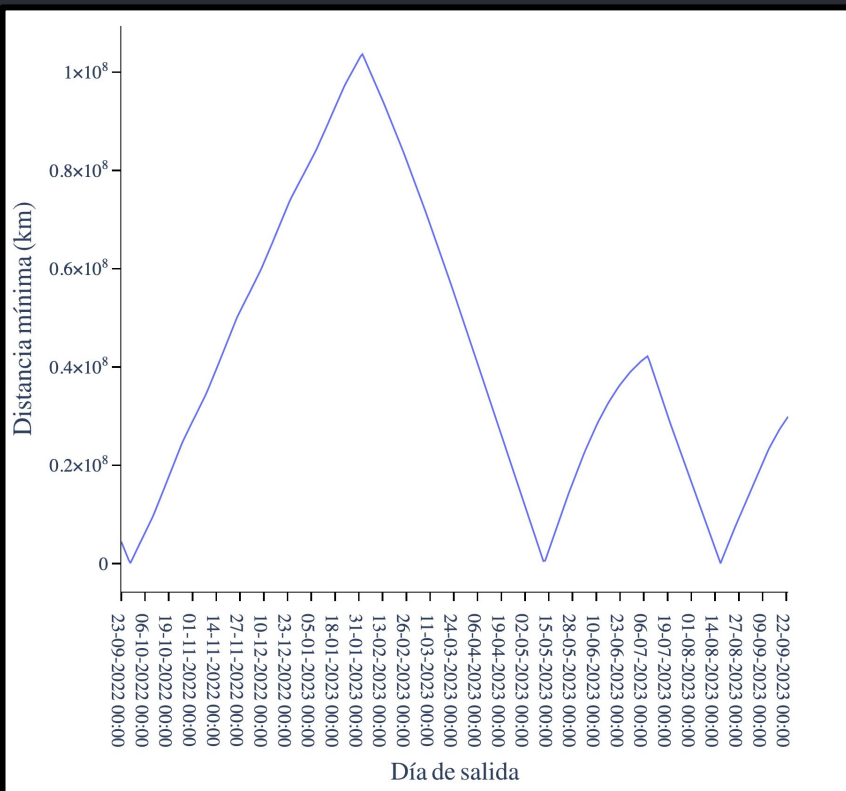
# RESULTADOS | ENERGÍA DEL SISTEMA

## VARIACIÓN ENERGÍA NAVE



# RESULTADOS | DESPEGUE A VENUS

## ANÁLISIS FECHA ÓPTIMA

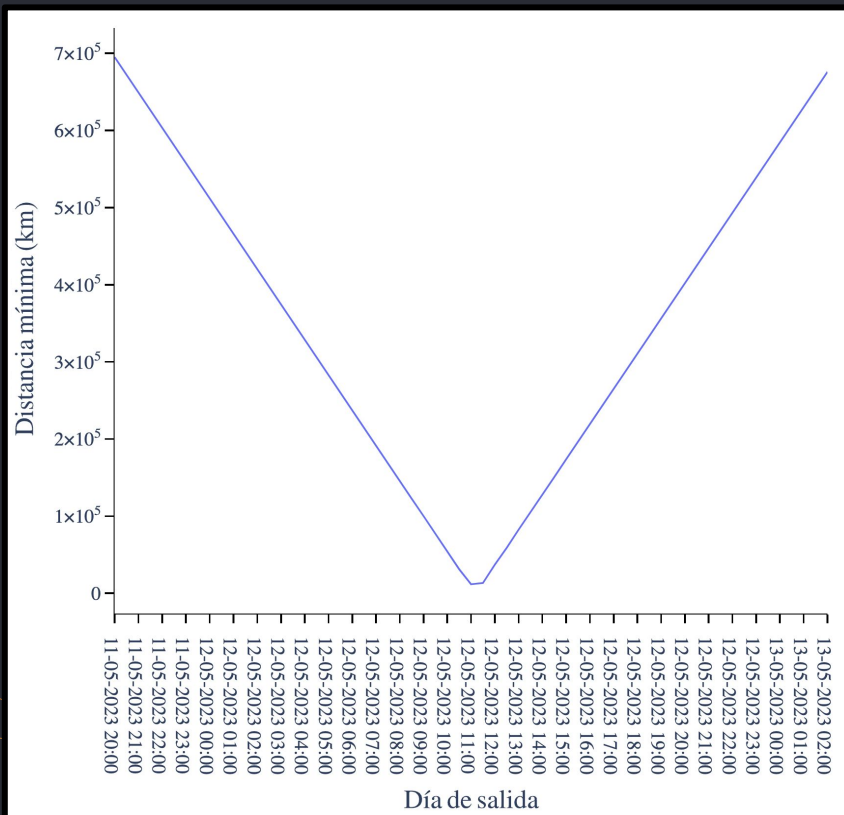


- Despegues cada 1 día.
- Mínimos:
  - 12-05-2023 (66 días).
  - 16-08-2023 (251 días).
- $dt = 300$  s.



# RESULTADOS | DESPEGUE A VENUS

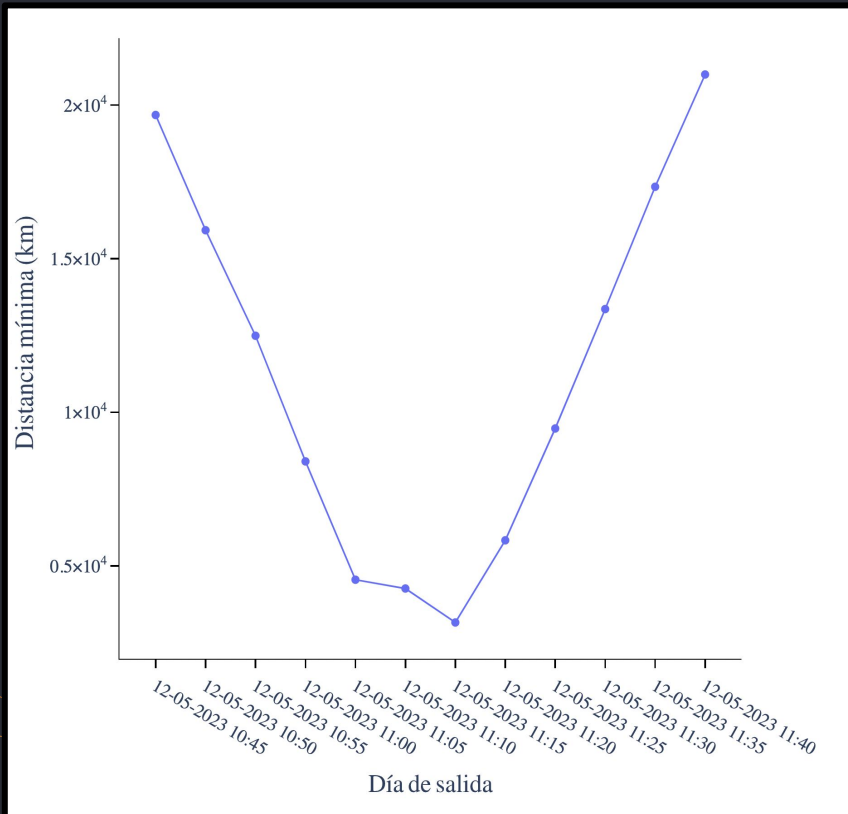
## ANÁLISIS FECHA ÓPTIMA



- Despegues cada 30 minutos.
- Mínimo: [10:00, 12:00] hs.
- $dt = 300$  s.

# RESULTADOS | DESPEGUE A VENUS

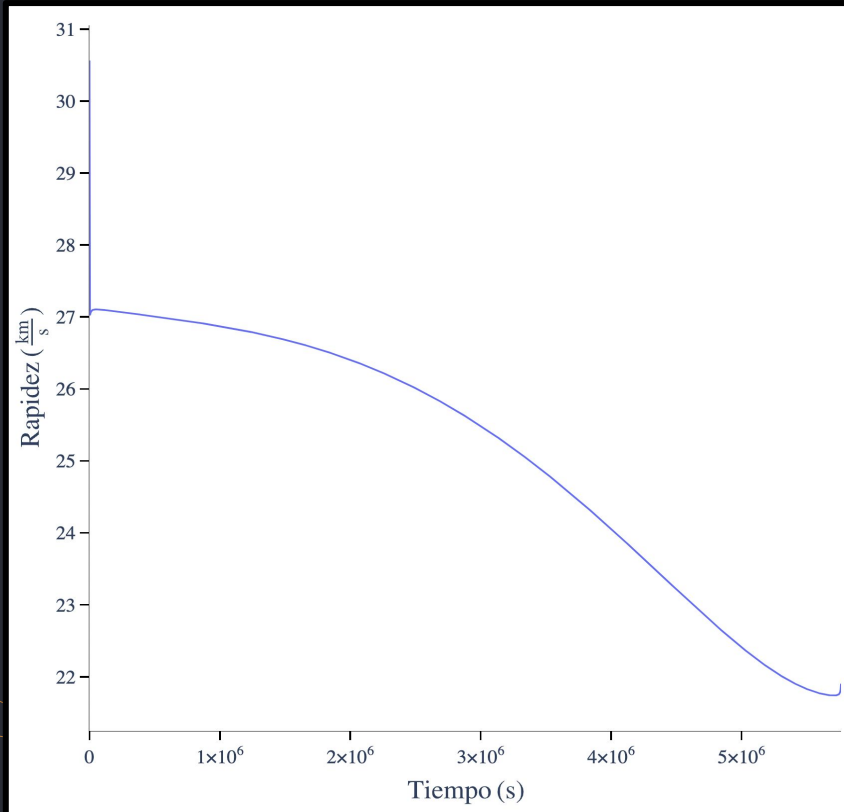
## ANÁLISIS FECHA ÓPTIMA



- Despegues cada 5 minutos.
- Fecha de salida: 12-05-2023 11:15 hs.
- $dt = 300$  s.
- Tiempo de viaje: 66 días

# RESULTADOS | FECHA ÓPTIMA DESPEGUE A VENUS

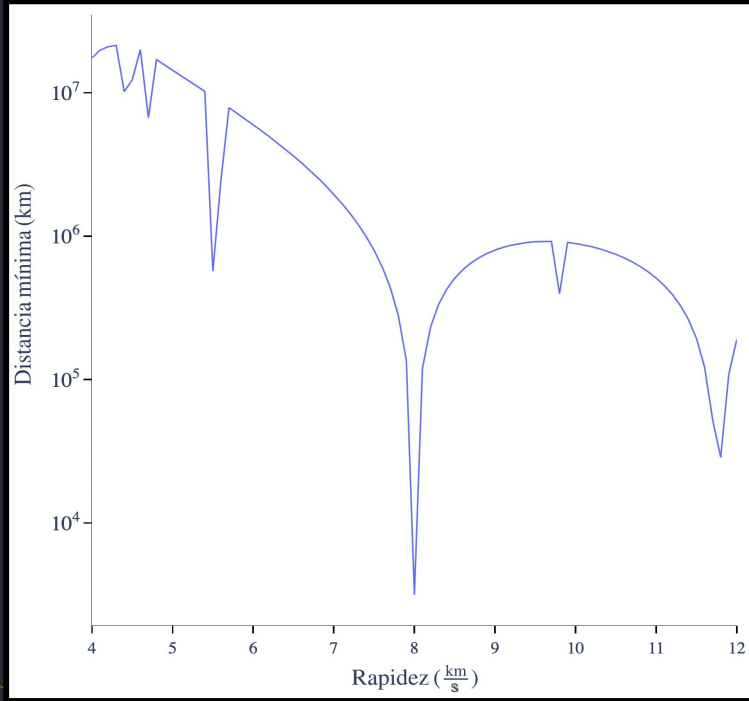
## EVOLUCIÓN DEL MÓDULO DE VELOCIDAD



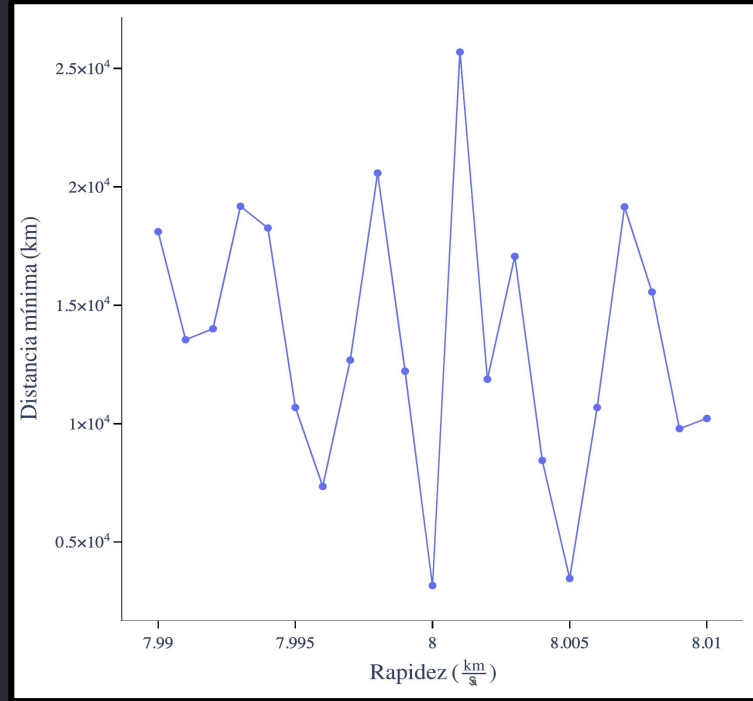
- Fecha de salida: 12-05-2023 11:15 hs.
- $dt = 300$  s.
- Tiempo de viaje: 66 días.
- Velocidad relativa a Venus 26.34 km/s.
- Impacta contra la superficie.

# RESULTADOS

## VARIACIÓN DE LA VELOCIDAD DE DESPEGUE



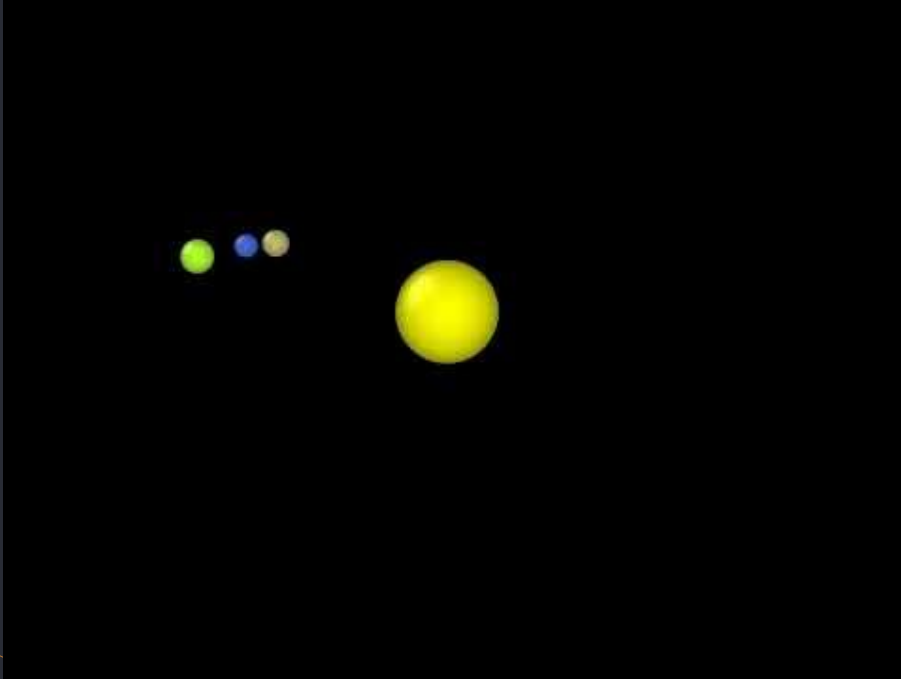
$V_t = 0.1 \text{ km/s}$



$V_t = 0.001 \text{ km/s}$

# RESULTADOS | FECHA REGRESO A LA TIERRA

## ANIMACIÓN

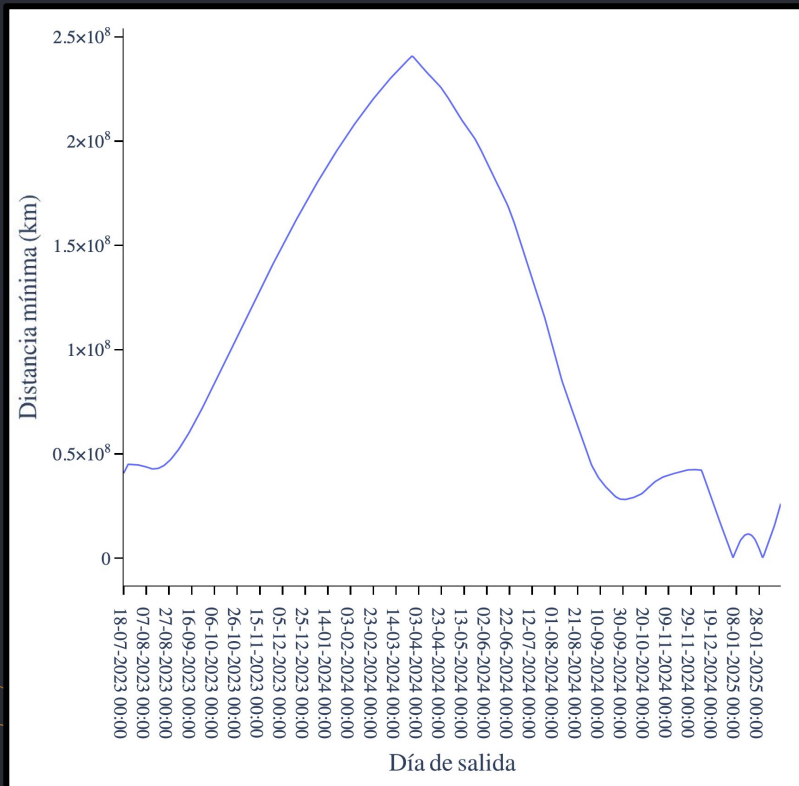


- Fecha de salida: 04-01-2025 20:10
- $dt = 300$  s.
- Tiempo de viaje: 114 días.

Animación en: <https://www.youtube.com/watch?v=8PDC2m9ggdU>

# RESULTADOS | DESPEGUE REGRESO A TIERRA

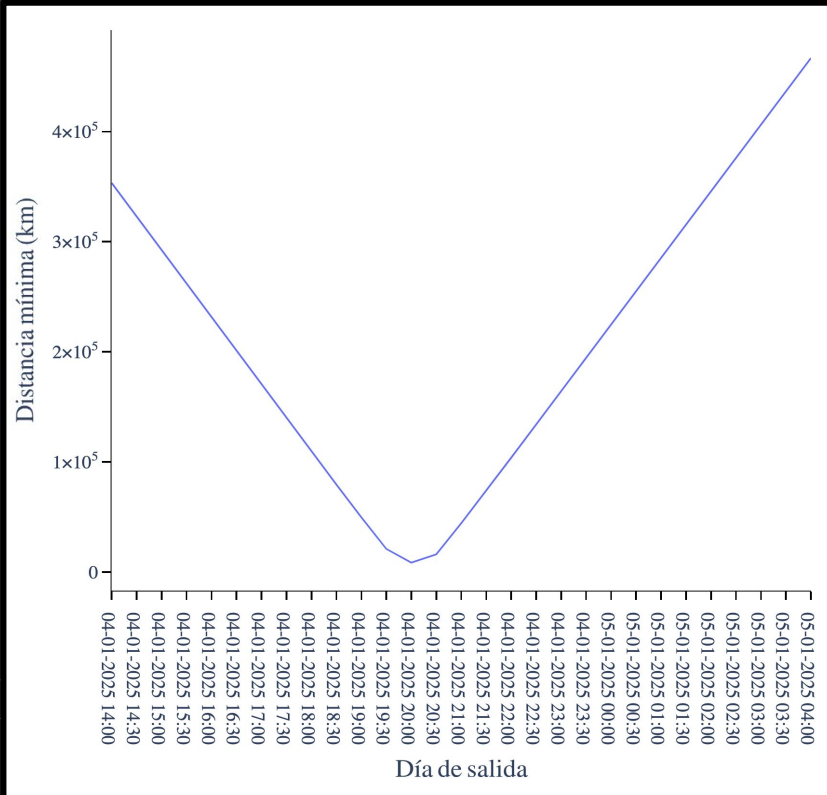
## ANÁLISIS FECHA



- Despegues cada 1 día.
- Mínimos:
  - 05-01-2025 (114 días).
  - 31-01-2025 (227 días).
- $dt = 300$  s.

# RESULTADOS | DESPEGUE REGRESO A TIERRA

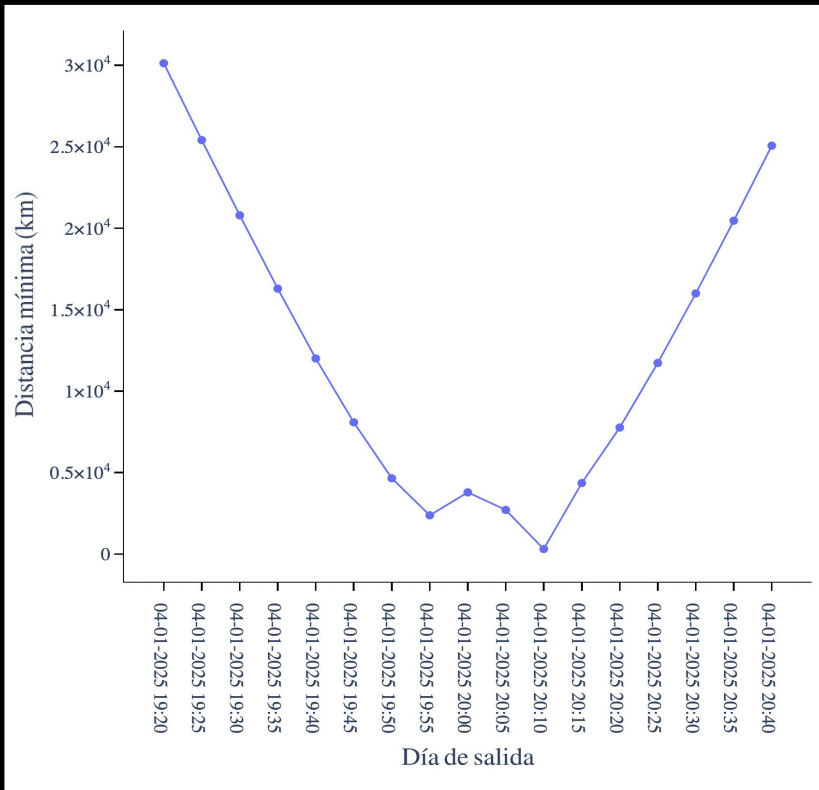
## ANÁLISIS FECHA DESPEGUE



- Despegues cada 30 minutos.
- Mínimo: 20:00 hs.
- $dt = 300$  s.

# RESULTADOS | DESPEGUE REGRESO A TIERRA

## ANÁLISIS FECHA DESPEGUE

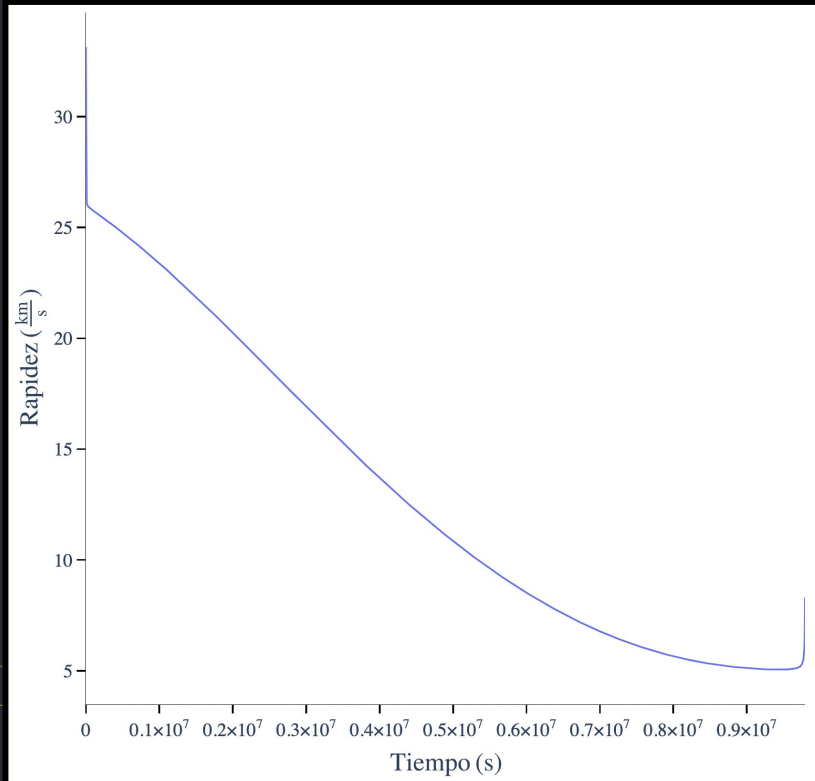


- Despegues cada 5 minutos.
- Fecha de salida: 04-01-2025 20:10 hs.
- $dt = 300$  s.
- Tiempo de viaje: 114 días.



# RESULTADOS | FECHA REGRESO A LA TIERRA

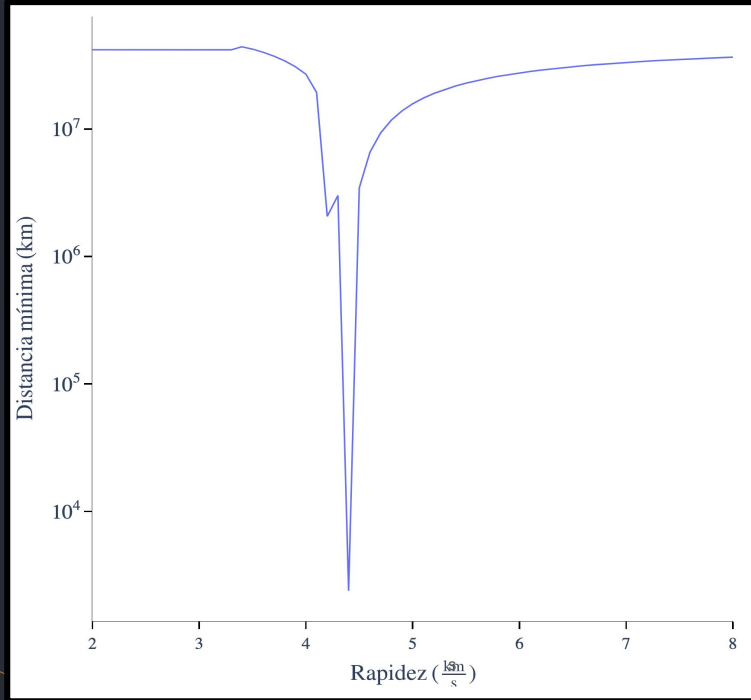
## EVOLUCIÓN DEL MÓDULO DE VELOCIDAD



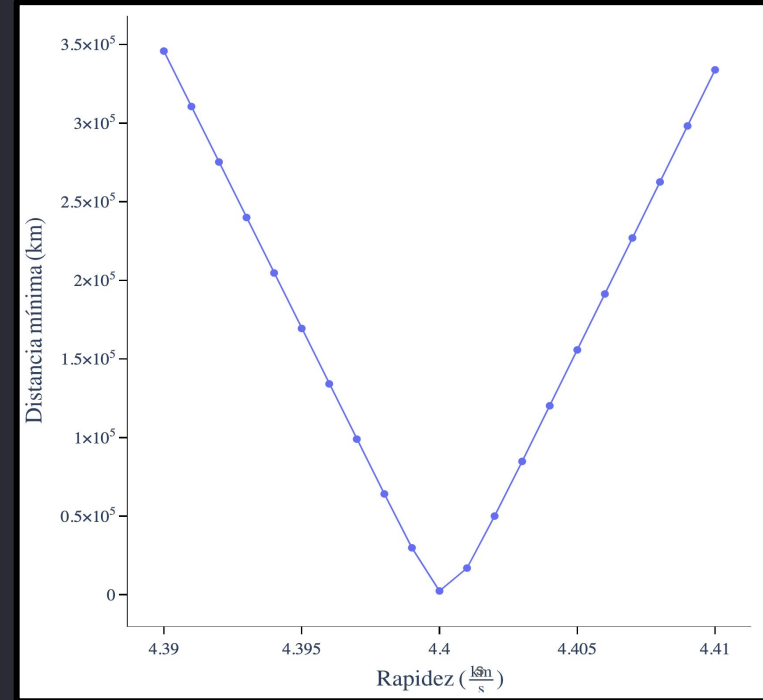
- Fecha de salida: 04-01-2025 20:10 hs.
- $dt = 300$  s.
- Tiempo de viaje: 114 días.
- Velocidad relativa a Tierra 16.73 km/s.
- Impacta contra la superficie.

# RESULTADOS

## VARIACIÓN DE LA VELOCIDAD DE DESPEGUE



$V_t = 0.1$  km/s



$V_t = 0.001$  km/s

# Conclusiones

## CONCLUSIONES

1. Existe instante de despegue de la Tierra que asegura llegada a Venus dadas las condiciones iniciales.
2. Existe instante de despegue de Venus que asegura la vuelta a la Tierra variando velocidad de salida de las condiciones iniciales.
3. Manteniendo condiciones iniciales variando velocidad de despegue, se asegura llegar a destino ante variaciones pequeñas.
4. Para fecha óptima de ida y regreso, módulo de velocidad disminuye al llegar a destino.



FIN  
¡GRACIAS!