

# **TRABAJO PRÁCTICO: “MODELO PVI DEL DESCENSO DEL ROVER PERSEVERANCE EN MARTE”**

## **GRUPO N°1**

| <b>Apellido, Nombres</b>       | <b>Email</b>        | <b>Nº Padrón</b> |
|--------------------------------|---------------------|------------------|
| Chen, Helen Elizabeth          | hchen@fi.uba.ar     | 110195           |
| Godoy, Nahuel                  | ngodoy@fi.uba.ar    | 100225           |
| Guardia, Franco                | faguardia@fi.uba.ar | 109374           |
| Liceri Martínez, Lucía María   | lliceri@fi.uba.ar   | 105964           |
| Luere Cahuatico, Walter Anibal | wluere@fi.uba.ar    | 103721           |

|                       |            |
|-----------------------|------------|
| <b>Fecha entrega:</b> | 28/06/2024 |
|-----------------------|------------|

## **Introducción**

El presente trabajo trata de modelar el descenso de la nave Rover Perseverance a la superficie de Marte la cual se divide en 4 etapas (en el trabajo solo se desarrollarán las primeras 3).

Se piensa al objeto como si estuviera en caída libre y se quiere emular de manera realista las condiciones diversas y complejas a las que está sometido en su paso por cada una de esas etapas. Por lo tanto, en cada una de ellas el movimiento de la nave estará descrito por una ecuación diferencial no lineal de segundo orden, que se resolverá en este caso aplicando el método de Runge-Kutta de orden 2. De esta forma conociendo las alturas de comienzo y fin de cada etapa será posible obtener información sobre los tiempos que le toma a la nave pasar por cada etapa y también conocer la velocidad con la que lo hace, para entonces con esa información realizar gráficos y así interpretar mejor lo que ocurre con la nave en el descenso.

---

# DESARROLLO

Todos los cálculos fueron realizados utilizando el lenguaje de programación Python y las librerías Numpy y Matplotlib para los gráficos. El código utilizado para implementar el método de Runge-Kutta es:

```
#función que aplica el método de Runge-Kutta de orden n
def runge_kutta(f, y0, t0, tf, paso, orden):
    pasos = int((tf - t0) / paso)
    t = np.linspace(t0, tf, pasos + 1)
    u = np.zeros((pasos + 1, len(y0)))
    k = np.zeros((orden + 1, len(y0)))
    u[0] = y0

    for i in range(pasos):
        sum = 0
        for j in range(orden):
            if(j == 0):
                k[j] = f(t[i], u[i])
            elif (j == orden - 1):
                k[j] = f(t[i] + paso, u[i] + k[j - 1])
            else:
                k[j] = 2 * f(t[i] + paso/2, u[i] + k[j - 1]/2)
            sum += k[j]
        u[i+1] = u[i] + paso * (sum/((orden - 1)*2))

    return t, u
```

*Fig.1: Runge-Kutta en Python*

Donde es posible utilizar el método para cualquier orden.

Luego, la discretización de la ecuación diferencial dada en cada etapa es:

## ETAPA 1

```
#función que aplica la ecuación diferencial de la etapa 1
def etapa1(t, yv):
    y, v = yv
    dy_dt = v
    dv_dt = -g + (BETA / M) * np.exp(-y / ALFA) * v**2
    return np.array([dy_dt, dv_dt])
```

Fig.2: Código para calcular la etapa 1 (entrada a la atmósfera)

## ETAPA 2

```
#función que aplica la ecuación diferencial de la etapa 2
def etapa2(t, yv):
    y, v = yv
    dy_dt = v
    dv_dt = -g + (ETA/M) * v**2
    return np.array([dy_dt, dv_dt])
```

Fig.3: Código para calcular la etapa 2 (apertura del paracaídas supersónico)

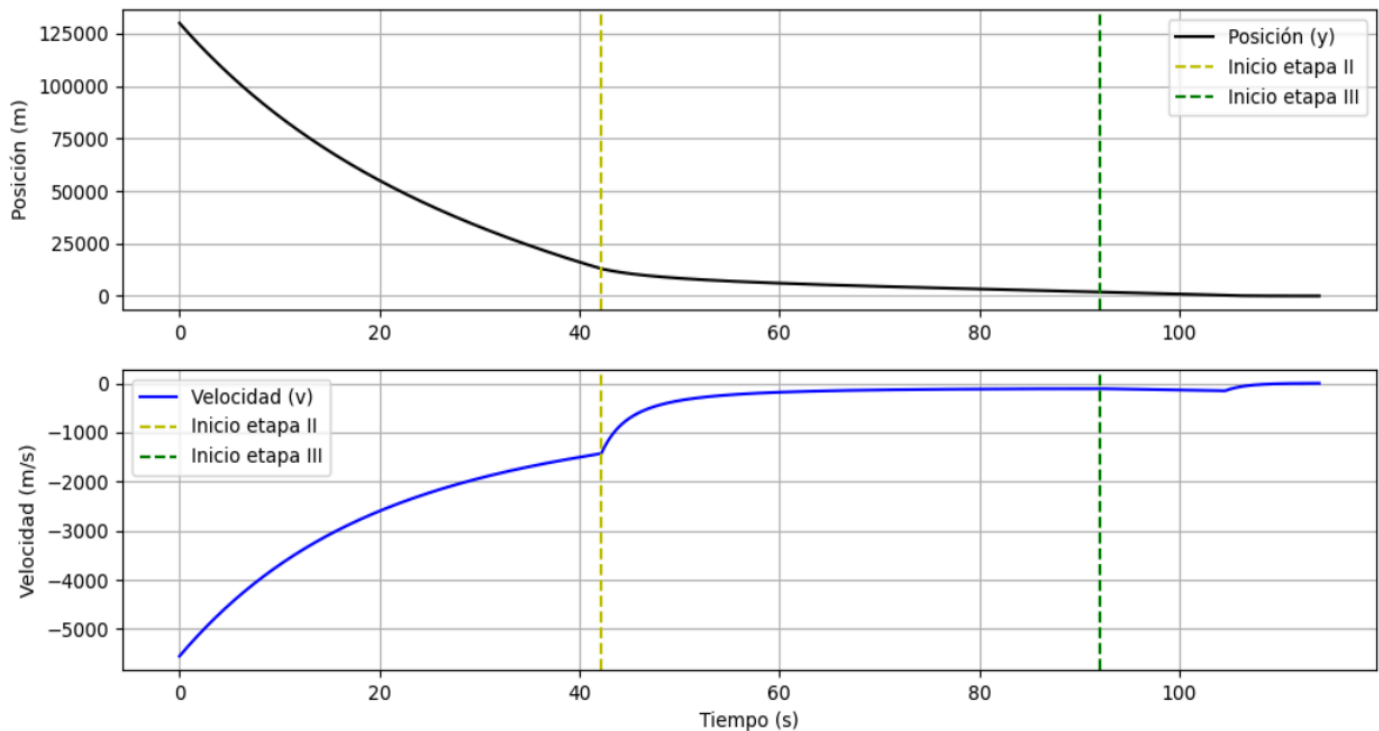
## ETAPA 3

```
#función que aplica la ecuación diferencial de la etapa 3
def etapa3(t, yvm):
    y, v, m = yvm
    dy_dt = v
    dm_dt = mc0 - k1 * (y - Hgrua) - k2 * v
    if dm_dt < 0:
        dm_dt = 0
    dv_dt = -g + (1 / M) * dm_dt * (ve - v)
    return np.array([dy_dt, dv_dt, dm_dt])
```

Fig.4: Código para calcular la etapa 3 (liberación del paracaídas y descenso con propulsores)

Haciendo las cuentas con estas funciones y los valores iniciales dados para la primera etapa (considerando que los valores finales de una etapa son los iniciales de la siguiente), se obtuvieron los siguientes gráficos para tanto la posición como la velocidad de la nave en las primeras 3 etapas:

### Modelo de descenso del Rover Perseverance en Marte

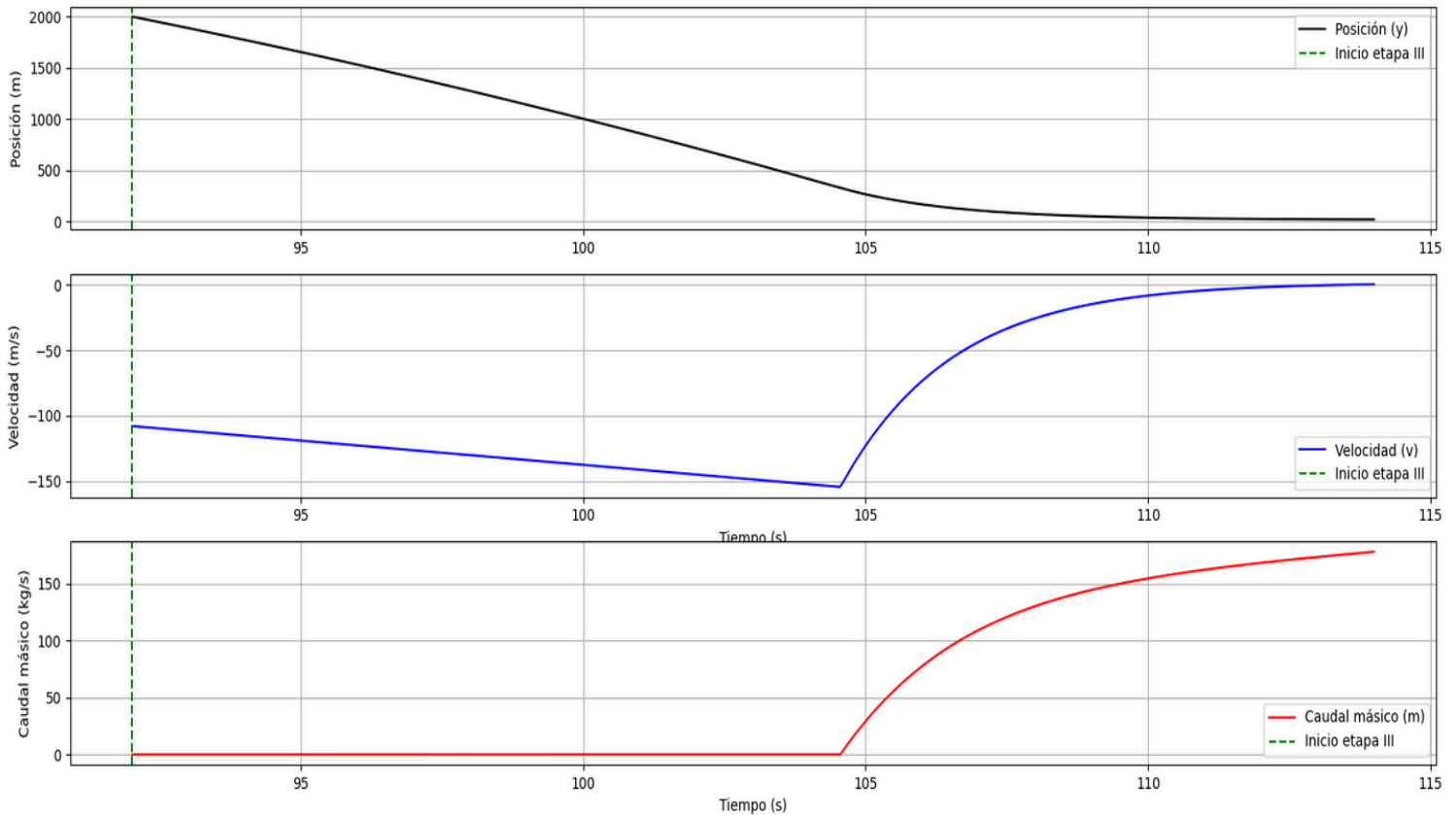


*Fig.5: Gráficos de la posición y la velocidad en función del tiempo de las 3 etapas*

Se observa que hay un descenso abrupto de la altura y la velocidad de la nave ni bien entra en la atmósfera de Marte hasta que alcanza la segunda etapa, donde sigue habiendo un descenso pero menos pronunciado (ya que corresponde cuando se abre el paracaídas, es decir el descenso es más lento que en la etapa anterior). Este fenómeno se observa mucho más desde que comienza la tercera etapa ya que si bien deja de estar el paracaídas está actuando la fuerza de propulsión en sentido contrario al descenso, que atenúa más la velocidad de la nave. En ese caso se ve que la altura sigue disminuyendo todo el tiempo hasta quedarse fija a 20 metros de altura que corresponde al final de esa etapa y a la altura de la grúa que a partir de ella descenderá hasta la superficie (es la etapa 4 que no se está modelando). Por el lado de la velocidad se observa un cierto comportamiento oscilatorio, primero descendiendo lentamente y de repente de forma más pronunciada durante un corto tiempo para luego de nuevo descender lentamente. Se puede pensar que esto ocurre por efecto de la fuerza de propulsión ya que depende de un caudal de combustible que es variable con la altura y la velocidad (pudiendo llegar a ser nulo según el caso, por la imposición que se hace de que no puede ser un valor negativo) y ambas cosas varían en la trayectoria de la nave en esa etapa. Finalmente, el valor de la velocidad queda fijo y con un valor distinto de cero, se puede pensar que esto pasa de nuevo por la fuerza de propulsión que es variable y puede provocar un aumento de la velocidad de tal forma de que la nave quede ubicada exactamente a 20 metros de altura (momento en el que se volverá cero porque solo descenderá en la grúa pero esa parte no se modela)

Luego solo para la etapa 3 los gráficos de posición, velocidad y caudal másico son:

### Modelo de descenso del Rover Perseverance en Marte (Etapa III)



Se observa con más detalle lo que ocurre en la etapa 3, con el descenso de la altura en todo momento hasta llegar a la altura final a 20 metros. En cuanto a la velocidad durante un tiempo de la etapa, esta aumenta debido a que la nave se desprende del paracaídas al finalizar la etapa 2, y eso le puede dar un leve impulso que es lo que provoca el aumento, para luego volver a disminuir hasta quedar en un valor positivo cercano a cero. Esto es lo que explica también el gráfico del caudal másico, ya que en todo el trayecto en que la velocidad aumenta este es cero y eso, como se comentó en el análisis anterior, tiene que ver con la imposición que se hace del caudal nulo en caso de que su valor (que sale de la expresión a la que se llega en función de la altura y velocidad) sea negativo por la imposibilidad de que los propulsores apunten hacia arriba como ese signo indicaría que ocurre. Luego, cuando la velocidad empieza a disminuir ahora sí se observa un caudal distinto de cero que aumenta con el tiempo. Esto tiene sentido considerando que la altura y la velocidad es cada vez menor (pensando nuevamente en la expresión del caudal haciendo que su valor aumente). Finalmente, queda un valor que parece estar aumentando, debido a que si bien la altura queda fija, la velocidad tiene un valor distinto de cero que no queda fijo.

Analizando los resultados obtenidos se completa el siguiente cuadro:

Valor elegido para  $k_1$ : 50

Valor elegido para  $k_2$ : 50

Son los valores elegidos para la ecuación diferencial de la etapa 3 ya que se pide valores similares en el intervalo [10, 100]

| Valor puntual   | Unidad | Resultado con 3 dígitos significativos |
|---|--------|--|
| Tiempo en el que comienza la etapa 2 (medido desde el inicio)   | [s]    | 41.2                                   |
| Velocidad al comienzo de la etapa 2   | [m/s]  | -1.43E3                                |
| Tiempo en el que comienza la etapa 3 (medido desde el inicio)   | [s]    | 92.0                                   |
| Velocidad al comienzo de la etapa 3   | [m/s]  | -108                                   |
| Tiempo total del descenso hasta llegar a $H_{grua} = 20 \text{ m.}$<br>Considerar descenso completo cuando el módulo de la velocidad es 0.03m/s (o menor) | [s]    | 113                                    |
| La masa de combustible necesaria para lograr con éxito la etapa 3   | [kg]   | 1.11E3                                 |

Se observa que todos los valores obtenidos tienen sentido físico, las velocidades negativas son porque se está tomando en el problema una coordenada vertical con sentido positivo hacia arriba (es decir contrario al descenso de la nave) y se corresponden con lo visto en los gráficos (velocidad disminuyendo etapa a etapa). Se podría decir que la etapa de mayor duración corresponde a la etapa 2 y eso puede deberse a que todavía la nave tiene que recorrer mucha distancia pero la liberación del paracaídas que está presente en toda la etapa hace que disminuya significativamente la velocidad haciendo que la nave tarde más en completarla.

Para modelar de manera más precisa el comportamiento del objeto en caída libre en la atmósfera de Marte, se podrían incorporar las siguientes contemplaciones físicas: en primer lugar, en nuestro caso consideramos la densidad del aire dependiente de la altura únicamente. Sin embargo, para acercarnos más los cálculos a lo que ocurre en la realidad, al obtener la densidad se podría contemplar no sólo la altura, sino que también los cambios de temperatura y presión atmosférica en función del tiempo. En segundo lugar, un factor importante que no se tiene en cuenta en los cálculos es cómo influye el viento en las diferentes capas de la atmósfera, ya que este puede afectar de manera significativa la trayectoria y la velocidad en la que desciende el objeto. También podríamos considerar que en lugar de caída en una única dirección, podríamos descomponer en las tres dimensiones espaciales ya que esto también influiría en el rozamiento del aire respecto del objeto.

## Función a optimizar

Primero tendríamos que proponer cuál es nuestro objetivo a optimizar, en este caso masa total de combustible ajustando los valores de  $k_1$  y  $k_2$ , con el objetivo final de reducir el mismo. Esto es con el objetivo de obtener el valor mínimo de combustible para lograr estabilizar la cápsula, una vez estable, si el consumo fue menor a 200 kg, bastará con mantener la cápsula en estabilidad por más tiempo hasta consumir los 200 kg.

## Opciones de optimización

En primera instancia, probamos con un método de fuerza bruta, es decir, probamos valores de  $k_1$  y  $k_2$ , calculamos la masa de combustible y variamos los valores buscando valores menores en cada caso.

Esto se puede mejorar, creando un método que variemos los valores de  $k_1$  y  $k_2$  con pequeñas perturbaciones y viendo el comportamiento, si la masa de combustible aumenta perturbamos en la dirección contraria. Es decir:

Iteramos hasta criterio de corte:

```
res = calcular_masa_combustible(k1 + delta, k2)
If res <= calcular_masa_combustible(k1,k2):
    k1 = k1 + delta
Else
    k1 = k1 - delta

res = calcular_masa_combustible(k1, k2 + delta)
If res <= calcular_masa_combustible(k1,k2):
    k2 = k2 + delta
Else
    k2 = k2 - delta
```

La idea es un pseudo punto fijo.

## Conclusiones

En primer lugar, es notable la resolución casi directa de un problema que en principio sería muy complejo con un código de no menos de 200 líneas (exagerando mucho), y esto sin conocer la solución analítica del problema. El método de RK2 es uno que se aproxima considerablemente bien a la solución en comparación a otros métodos de ecuaciones diferenciales.

También, cabe destacar el provecho que podemos sacar de la velocidad de resolución de una computadora con estos algoritmos. El mismo problema sería imposible de resolver analógicamente haciendo cuentas ya que al finalizar el ciclo se contaron aproximadamente unas 50 mil iteraciones (100 iteraciones por segundo con sus respectivas iteraciones del orden de RK), cosa que a la computadora no le toma más que un segundo en resolver.

Sobre la estabilidad del algoritmo, el método de RK puede divergir. Sin embargo, en este caso podemos ver que no sucede ya que el sistema se comporta como esperaríamos obteniendo un resultado coherente, dando una idea de que la solución converge al modelo real.