

**Universidad
Rey Juan Carlos**

Escuela Superior de Ciencias Experimentales y Tecnología

GRADO EN INGENIERÍA MECÁNICA

Trabajo Fin de Grado

**Desarrollo de un software para el diseño de
protecciones balísticas basado en el modelo
analítico de Tate-Alekseevskii**

Pablo Villabrille Patallo

Director: Jesús Rodríguez Pérez

Curso Académico 2024/25

Contenidos

1. Resumen	4
2. Introducción.....	5
3. Objetivos	10
4. Desarrollo teórico	10
4.1 Introducción a la analogía hidrodinámica	10
4.2 Desarrollo de la ecuación de Bernoulli	12
4.3 Derivación de la ecuación constituyente del modelo de Tate - Alekseevskii	14
4.4 Balance de momento a lo largo de la línea central de acción.....	16
4.5 Deceleración de un proyectil impactando sobre una pared rígida.....	19
4.6 El modelo de Tate - Alekseevskii	20
5. Resultados y discusión	22
5.1 Algoritmo de cálculo	22
5.2 Funcionamiento del programa.....	23
5.3 Validación del modelo	31
5.4 Aplicación del modelo y el límite balístico	39
6. Conclusiones	41
Referencias	42
Anexos.....	43
Anexo A. Expresiones matemáticas y simbología	43
Notación vectorial	43
Delta de Kronecker.....	43
Símbolo de Levi-Civita o de permutación.....	43
El operador Nabla.....	44
Anexo B. Leyes físicas y teoremas	45
La derivada material.....	45
Teorema de la divergencia o de Gauss-Ostrogadsky.....	46
Teorema del transporte de Reynolds	46
Ley de conservación de la masa	46
Ley de conservación de momento	47
Ley de conservación de la energía.....	48
Tensor de deformación	49
Anexo C. Programación del modelo simplificado de Tate – Alekseevskii.....	50
Script base	50
Interfaz para selección de caso	52
Interfaz de introducción de datos para “Caso único”	54

Módulo de cálculo para “Caso único”	63
Generador de gráficas para “Caso único”	71
Interfaz de introducción de datos para “Rango de velocidades”	74
Interfaz de introducción de rango de velocidades para “Rango de velocidades”	81
Módulo de cálculo para “Rango de velocidades”	84
Generador de gráficas para “Rango de velocidades”	89

1. Resumen

El tema de este trabajo fin de grado es la mecánica de la penetración y los desarrollos realizados se basan en el modelo formulado por Tate [1] y Alekseevskii [2]

El objetivo principal del proyecto es desarrollar un programa de cálculo que permita diseñar protecciones balísticas frente a determinadas amenazas. Asimismo, se pretenden comprender los mecanismos que actúan en el impacto de un proyectil de energía cinética, desde que el proyectil hace contacto hasta que finaliza la interacción proyectil-blanco. Para ello se utilizará el modelo simplificado de Tate – Alekseevskii como base para el desarrollo del software.

El estudio de la interacción proyectil-blanco es un proceso complejo, donde intervienen ondas, conceptos de mecánica de medios continuos y resistencia de materiales, expansión de cráteres, erosión, penetración rígida, etc, todos ellos con fuertes componentes matemáticas. Es, por tanto, un tema que abarca muchos conceptos estudiados en la carrera de Ingeniería Mecánica.

El estudio de la influencia de los múltiples parámetros del proceso, tanto del proyectil como del blanco, se llevará a cabo a partir del desarrollo teórico del modelo y la implementación práctica se llevará a cabo utilizando Python, un lenguaje de programación en auge debido a su gran flexibilidad.

Se ha seleccionado este modelo ya que, a pesar de que existen herramientas empíricas que son capaces de predecir razonablemente la penetración de un proyectil en un blanco, se basan en curvas de ajuste sobre datos experimentales previamente realizados, y no proporcionan información sobre la física que actúa en el proceso de penetración. Por lo tanto, se opta por un modelo analítico simple que permita comenzar a comprender los mecanismos influyentes en este proceso.

Si bien es cierto que hoy en día existen programas muy refinados de simulación mediante elementos finitos o diferencias finitas, estos pueden tener un coste computacional demasiado elevado, pues dichos programas tienen en cuenta una cantidad muy numerosa de parámetros para ajustarse lo máximo posible a comportamientos reales. Un claro ejemplo de estos programas puede ser el CTH “*Eulerian Wave Propagation Code*” o *AUTODYN*. Es por ello por lo que modelos analíticos como el de Tate – Alekseevskii pueden ser de gran utilidad, al enfocarse más en los propios mecanismos del proceso para comprenderlo y poder simularlo de manera simplificada, reduciendo enormemente el coste computacional sin perder demasiada información sobre la esencia del mecanismo de penetración.

Tras desarrollar el modelo, se compara con un caso experimental para proceder a su validación y, se podrá observar que, aunque el modelo tiene sus limitaciones para realizar un diseño de protección balística, este es capaz de predecir de forma razonable la mecánica de la penetración, sobre todo para un estado estacionario, que es el más longevo del proceso.

Se concluye, por tanto, que el modelo es correcto para obtener una primera toma de contacto sobre el funcionamiento del proceso de penetración, pero que existen alternativas analíticas más completas para acercarse más a un comportamiento real.

2. Introducción

La mecánica de la penetración, a pesar de su poca visibilidad, es un campo que abarca múltiples ámbitos. A pesar de que en este trabajo se profundizará más en vehículos de combate y blindaje, la mecánica de la penetración es también empleada en otras áreas como por ejemplo la espacial.

Durante la Primera Guerra Mundial la efectividad de los carros de combate, o los más comúnmente conocidos como tanques, dio lugar a un rápido desarrollo de estos durante la Segunda Guerra Mundial.

Inicialmente el objetivo de los primeros carros de combate era el de poder atravesar con facilidad trincheras enemigas, lo cual era una tarea extremadamente complicada para la infantería. El elemento que protegía a la tripulación de ser abatida por las distintas amenazas de la época era el blindaje, que en ese momento consistía en la disposición de planchas metálicas de cierto espesor alrededor del vehículo para impedir la penetración de los proyectiles en la zona donde se situaba la tripulación.



Figura 1. El Mark V, uno de los primeros carros de combate puestos en servicio de la historia.

Durante esta guerra las únicas amenazas efectivas que se emplearon frente a estos vehículos fueron los rifles antitanque, que, con su gran potencia de fuego, permitían atravesar el blindaje con gran eficacia.

Con la llegada de la Segunda Guerra Mundial la tecnología de los carros de combate comenzó a modernizarse de forma muy acelerada, pues comenzaban a surgir amenazas cada vez más eficaces frente a estos vehículos. Dentro de esta modernización de los carros de combate, la mejora de los elementos

de protección tomó un papel muy importante ya que, si se aumentaba la protección del carro, este podría prolongar su vida útil y reducir el número de bajas de los carristas.

Está claro que una solución sencilla sería utilizar un blindaje de mayor espesor, pero deben tenerse en cuenta múltiples consecuencias que vienen con esta solución, como podría ser el coste, peso, capacidad de ocultación, entre otros. Es por ello por lo que se empezaron a desarrollar nuevas estrategias para defender el vehículo minimizando la pérdida de eficiencia en otros ámbitos. Un claro ejemplo de esto último es el blindaje inclinado, el cual mostró su gran capacidad de protección con la llegada del carro de combate T34.



Figura 2. Columna de T34 soviéticos.

En estos momentos la mecánica de la penetración comenzó a tomar especial importancia, ya que la comprensión del fenómeno de la penetración permitiría aumentar la efectividad del blindaje de los carros.

Está claro que una buena comprensión de cómo se comporta el proyectil cuando incide sobre un elemento protector es vital para poder crear blindajes y/o proyectiles más eficaces.

Los avances más modernos incluyen la denominada ERA (*Explosive Reactive Armor*) que se trata de un elemento activo de protección contra proyectiles explosivos montado en vehículos susceptibles a estas amenazas.



Figura 3. El carro de combate Abrams M1.

La protección activa no será abordada en este estudio, ya que está pensada para proyectiles con carácter explosivo y se pretende estudiar proyectiles que simplemente emplean energía cinética para lograr la penetración.

Aunque no se vaya a trabajar con elementos de protección activa, es interesante comprender su modo de operación. Este tipo de protección funciona mediante la detonación de módulos con explosivo plástico almacenado dentro de ellos cuando se detecta un proyectil inminente, alterando la trayectoria o inhabilitando a este mediante la detonación preventiva de la carga explosiva que almacenan.

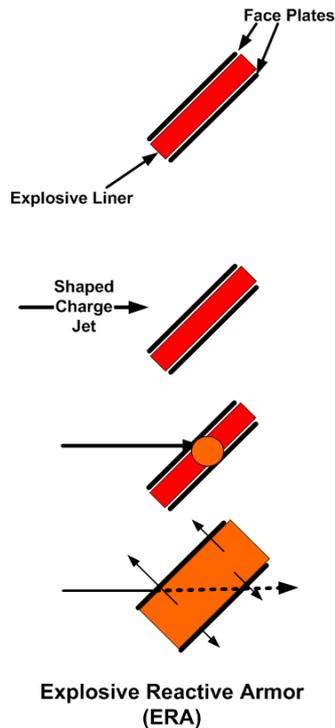


Figura 4. Esquema de funcionamiento de la protección activa ERA.

Volviendo al caso a estudiar, los blindajes (pasivos) más modernos están caracterizados por el uso de distintos materiales, los llamados *composites*. Estos blindajes aprovechan las propiedades de diferentes materiales como combinaciones de capas de metales, cerámicas, incluso materiales poliméricos, para aumentar la protección del vehículo y su eficiencia.

Aunque claramente este trabajo está más enfocado a carros de combate y protección balística en general, es necesario aclarar que la protección frente a proyectiles es utilizada también en el sector espacial. Los satélites en órbita se mueven a velocidades extremadamente altas, por lo que cualquier objeto en su trayectoria, por pequeño que sea, puede causar grandes daños si dicho satélite no está bien protegido. Por ello, aunque el estudio de los mecanismos de penetración no sea demasiado conocido, tiene aplicaciones muy importantes en ámbitos conocidos.

El estudio a realizar será en una dimensión espacial sobre la interacción de un proyectil cinético que impacta de forma perpendicular sobre una plancha de material semi-infinita.

Los proyectiles de energía cinética son aquellos que, como su nombre bien indica, solamente emplean la energía cinética para lograr la penetración. Un claro ejemplo de estos proyectiles sería el APFSDS (Armour Piercing Fin-Stabilized Discarding Sabot).

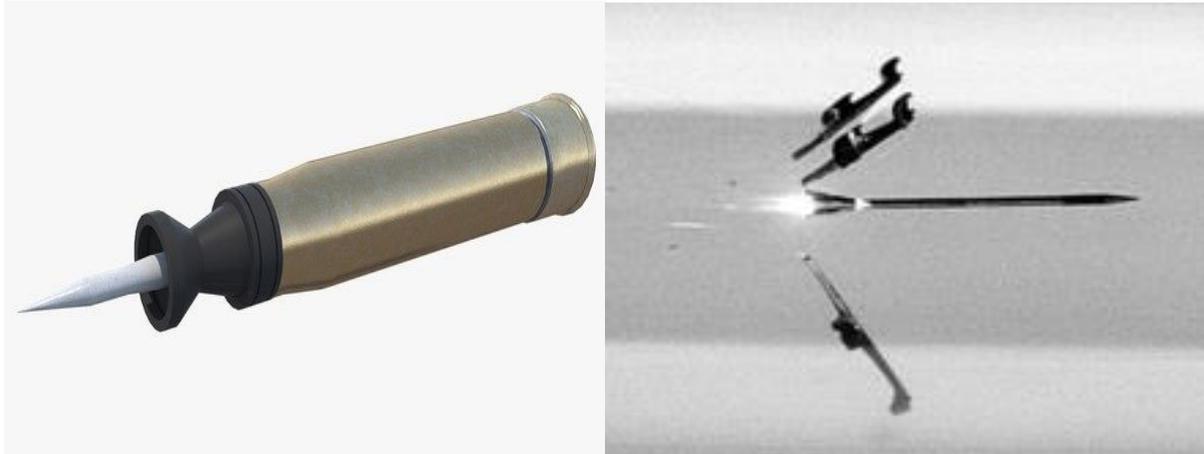


Figura 5. Munición APFSDS. proyectil armado (izquierda) y proyectil en vuelo (derecha).

Otro tipo de proyectil que puede ser simulado mediante este modelo es el proyectil de carga hueca, que, mediante explosivo almacenado de forma cónica, dirige un jet de metal fundido (generalmente cobre) a muy alta velocidad contra el blanco

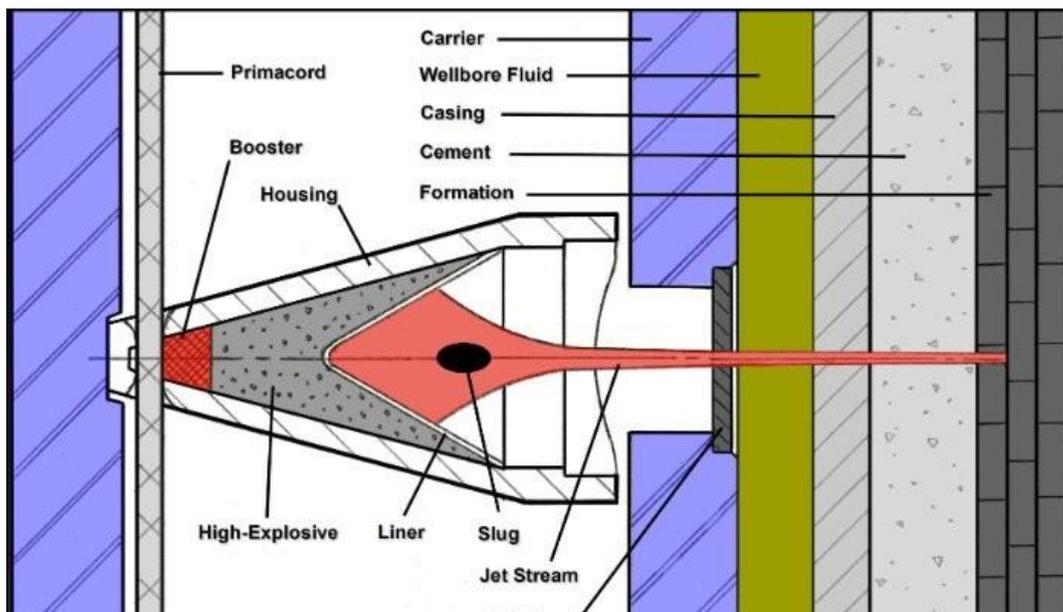


Figura 6. Representación gráfica de un proyectil de carga hueca.

Si bien puede parecer que se está reduciendo demasiado la complejidad al realizar un análisis unidimensional, cabe destacar que en los modelos analíticos se realizan muchas simplificaciones, pues si no se hicieran, serían extremadamente difíciles de leer y aplicar. Aun así, estas simplificaciones se hacen en base a ciertos criterios, viendo que no resultan en desviaciones grandes de datos experimentales y que permiten obtener una buena aproximación a los mecanismos que intervienen en la realidad. Permitiendo, por ende, comprender los procesos que influyen en este campo sin perder demasiada información.

3. Objetivos

La finalidad última de este Trabajo Fin de Grado es la de obtener conocimiento sobre el comportamiento del proyectil y del blanco en un caso de penetración balística. Para ello se formulan los siguientes objetivos:

- Conocer y comprender el modelo analítico de Tate-Alevseeskii
- Desarrollar un programa que implemente el modelo de Tate-Alevseeskii y proceder a su validación.
- Aplicar el modelo a la resolución de casos particulares de impacto balístico

También se comprobará cuanto se acerca el modelo analítico a la realidad, comparando los resultados obtenidos con simulaciones numéricas, además de datos experimentales.

Antes de comenzar a programar, es necesario conocer los fundamentos teóricos del modelo. Para ello, en primer lugar, se expondrá el desarrollo teórico que permita obtener las expresiones deseadas para realizar la simulación y posteriormente, se analizarán los resultados obtenidos para poder comprobar la validez del modelo.

4. Desarrollo teórico

4.1 Introducción a la analogía hidrodinámica

Antes de comenzar con el desarrollo, es recomendable haber leído anteriormente el Anexo A, para poder tener una mayor comprensión de los distintos elementos matemáticos que se emplearán en este apartado y poder seguir con más facilidad el desarrollo del modelo.

El modelo de Tate- Alekseevskii se basa en realizar una analogía hidrodinámica para poder predecir la penetración en un objetivo, es decir, tratar la interfase proyectil-blanco como un fluido.

Cuando se produce el contacto entre el proyectil y el blanco, se genera una onda de choque planar en ambos, que poco después toma la forma que se ve en la Figura 6 debido a la reflexión en las superficies tanto de proyectil como blanco, que generan ondas que debilitarán progresivamente a la onda de choque inicialmente generada.

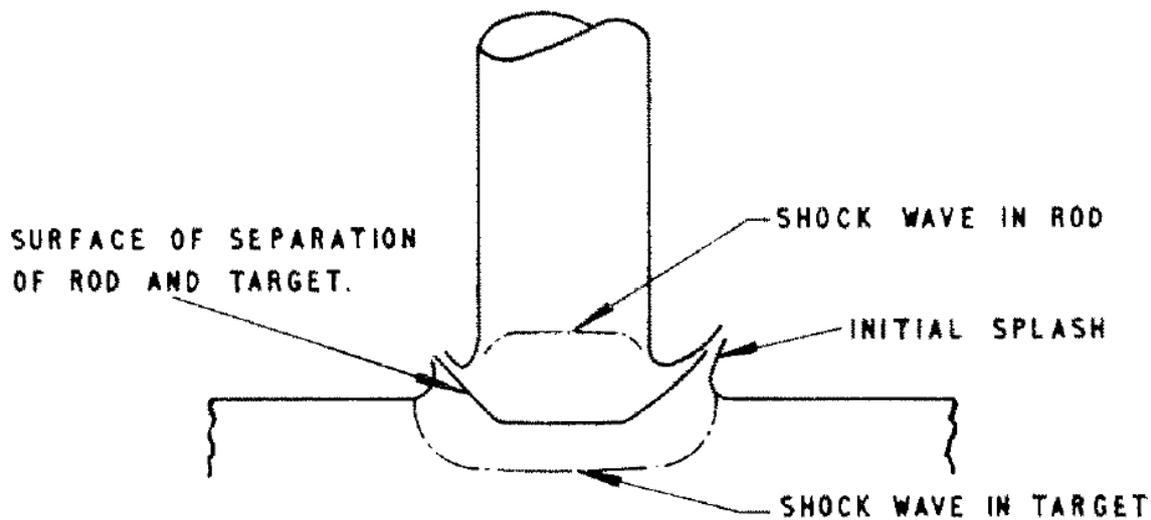


Figura 7. Esquema de ondas instantes después del impacto. Obtenido de A. Tate, "A theory for the deceleration of long rods after impact".

Esta forma de seta que se produce es debido a las ondas reflejadas en la superficie de ambos materiales, pues se expanden de forma radial, generando ese desplazamiento de material.

Desde la perspectiva del proyectil, la onda de choque se debilita en la dirección longitudinal progresivamente hasta llegar a dos o tres diámetros del proyectil, donde se vuelve tan débil que se puede despreciar su contribución al proceso.

Con respecto al blanco, la onda de choque generada se propaga de forma semiesférica (en el caso de 2D sería de forma radial) hasta un punto en el que se debilita tanto que se vuelve inestable y se divide en dos ondas, una onda elástica precursora, seguida por una onda de deformación plástica. Finalmente, la onda de deformación plástica deja de propagarse y la elástica continúa su expansión [3].

En el momento en el que las ondas o bien se han debilitado lo suficiente o se encuentran muy alejadas de la cavidad generada, es en el que la analogía semi-hidrodinámica puede ser de utilidad, pues las presiones que están soportando tanto penetrador como blanco exceden en magnitud las propiedades mecánicas de ambos materiales y se establece un estado casi estacionario donde la cavidad se agranda a una velocidad aproximadamente constante [2].

Aun así, esto no implica que las propiedades de ambos materiales no sean influyentes en el proceso, al contrario, son una parte muy importante de este y serán utilizadas para definir el modelo analítico. A medida que el proyectil se frena, las propiedades de los materiales van ganando importancia, pues en la fase final de penetración los mecanismos que influyen predominantemente en el proceso provendrán de fenómenos elástico-plásticos.

Habiendo comprendido que la fase mayoritaria de la penetración puede aproximarse a un comportamiento semi-líquido de los materiales, se procederá a desarrollar la ecuación de Bernoulli para flujo estacionario.

4.2 Desarrollo de la ecuación de Bernoulli

Se comienza definiendo la aceleración de una partícula en un fluido

$$\frac{Dv_i}{Dt} = \frac{\partial v_i}{\partial t} + v_j \frac{\partial v_i}{\partial x_j} \quad (4.1)$$

Por la regla de la cadena

$$v_j \frac{\partial v_j}{\partial x_i} = \frac{1}{2} \frac{\partial (v_j v_j)}{\partial x_i} \quad (4.2)$$

Y con un poco de álgebra, la expresión de la aceleración puede escribirse como

$$\frac{Dv_i}{Dt} = \frac{\partial v_i}{\partial t} + v_j \frac{\partial v_i}{\partial x_j} = \frac{\partial v_i}{\partial t} + \frac{1}{2} \frac{\partial (v_j v_j)}{\partial x_i} + v_j \left(\frac{\partial v_i}{\partial x_j} - \frac{\partial v_j}{\partial x_i} \right) \quad (4.3)$$

El término en paréntesis de la derecha contiene las componentes del rotacional de la velocidad.

Si se utiliza el tensor de rotación W que almacena dicho rotacional, y sus componentes

$$W_{ij} = \frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} - \frac{\partial v_j}{\partial x_i} \right) \quad (4.4)$$

$$\vec{\omega} = \frac{1}{2} \text{rot}(\vec{v}) \quad (4.5)$$

Y desarrollando

$$2W\vec{v} = \begin{pmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} = \begin{vmatrix} \hat{e}_1 & \hat{e}_2 & \hat{e}_3 \\ \omega_1 & \omega_2 & \omega_3 \\ v_1 & v_2 & v_3 \end{vmatrix} = \vec{\omega} \times \vec{v} = -\vec{v} \times \vec{\omega} \quad (4.6)$$

Si se introducen estas expresiones en la ecuación (4.3) se obtiene

$$\frac{D\vec{v}}{Dt} = \frac{\partial \vec{v}}{\partial t} + \frac{1}{2} \text{grad}(|\vec{v}|^2) - \vec{v} \times \text{rot}(\vec{v}) \quad (4.7)$$

Que es la expresión en base a la cual se formulará la ecuación de Bernoulli junto con la ley de conservación del momento, aplicando ciertas restricciones al flujo, que son las siguientes:

1. El fluido no es viscoso, lo que quiere decir que las tensiones que se producen provienen únicamente de la parte hidrostática del tensor de tensiones, es decir

$$\sigma_{ij} = -p\delta_{ij} \quad (4.8)$$

Por lo cual no existen tensiones cortantes en el fluido, ya que, al no tener viscosidad, las corrientes adyacentes no generan ninguna fuerza.

2. Las fuerzas de volumen son conservativas, por lo tanto, pueden expresarse en términos de una función potencial φ

$$b_i = -\frac{\partial\varphi}{\partial x_i} \quad (4.9)$$

3. El flujo es estacionario, es decir, no existe variación de sus propiedades con respecto al tiempo, por tanto

$$\frac{\partial\vec{v}}{\partial t} = 0 \quad (4.10)$$

4. El flujo es incompresible, por lo tanto $div(\vec{v}) = 0$, lo que implica que la densidad del fluido permanece constante ya que no existen sumideros ni fuentes (ver apéndice A). Expresado matemáticamente sería $\rho = \rho_0$.

Si se divide la conservación de momento (obtenida en el Apéndice B) por la densidad

$$\frac{1}{\rho} \frac{\partial\sigma_{ji}}{\partial x_j} + b_i = \frac{Dv_i}{Dt} \quad (4.11)$$

E insertando las 4 condiciones mencionadas anteriormente junto con la expresión de la aceleración

$$-\frac{1}{\rho} \frac{\partial p}{\partial x_i} - grad(\varphi) = \frac{1}{2} grad(|\vec{v}|^2) - \vec{v} \times rot(\vec{v}) \quad (4.12)$$

Reorganizando

$$-grad\left(\frac{p}{\rho_0}\right) - grad(\varphi) = \frac{1}{2} grad(|\vec{v}|^2) - \vec{v} \times rot(\vec{v}) \quad (4.13)$$

$$grad\left(\frac{p}{\rho_0} + \varphi + \frac{1}{2} |\vec{v}|^2\right) = \vec{v} \times rot(\vec{v}) \quad (4.14)$$

Si uno se sitúa a lo largo de una línea de corriente podrá ver localmente como una partícula del fluido se desplaza con velocidad \vec{v} , por lo tanto, si se multiplica escalarmente ambos lados de la ecuación (4.14) se obtiene que el lado derecho es igual a cero. Esto se debe a que, el producto cruz devuelve un vector perpendicular a sus dos parámetros de entrada, en este caso, \vec{v} y $rot(\vec{v})$, por lo tanto, si se realiza el producto punto entre \vec{v} y el resultado del rotacional, al ser ambos vectores perpendiculares entre sí, el producto punto entre ellos será cero.

$$\vec{v} \cdot \text{grad} \left(\frac{p}{\rho_0} + \varphi + \frac{1}{2} |\vec{v}|^2 \right) = 0 \quad (4.15)$$

Por las razones explicadas anteriormente, las componentes del gradiente perpendiculares a la velocidad no entran en el producto punto, por lo tanto, debe satisfacerse que las componentes tangenciales de dicho gradiente a la velocidad sean cero. Esta condición se cumple si la suma de los términos en la función gradiente es constante sobre la línea de corriente.

$$\frac{p}{\rho_0} + \varphi + \frac{1}{2} |\vec{v}|^2 = cte \quad (4.16)$$

Si se multiplica por la densidad, recordando que permanece constante por la condición de flujo incompresible

$$p + \rho_0 \varphi + \frac{1}{2} \rho_0 |\vec{v}|^2 = cte \quad (4.17)$$

Proporciona finalmente la ecuación de Bernoulli.

4.3 Derivación de la ecuación constituyente del modelo de Tate - Alekseevskii

Tomando inicialmente el proyectil como un jet de fluido, si se observa una partícula en la línea central de dicho jet alejada del blanco o “pared”, podrá observarse como se mueve con velocidad v . A medida que esta partícula se acerca al blanco, se irá desacelerando progresivamente hasta detenerse justo cuando alcanza al blanco, a lo que se le llama punto de estancamiento.

Si se definen dos zonas a lo largo de las cuales ocurre este suceso, tomando la parte izquierda como una zona alejada del blanco y la parte derecha como una zona próxima al blanco, y teniendo en cuenta que en todo momento se está siguiendo la línea central del jet, se puede utilizar la ecuación (4.17) para determinar el estado en el que se encuentran dichas zonas.

$$p_{iz} + \rho \varphi_{iz} + \frac{1}{2} \rho |\vec{v}_{iz}|^2 = p_d + \rho \varphi_d + \frac{1}{2} \rho |\vec{v}_d|^2 \quad (4.18)$$

Correspondiéndose los subíndices iz y d con izquierda y derecha respectivamente. Asumiendo que no existen fuerzas de volumen y que la parte derecha de dicha interfaz se encuentra en reposo

$$p_{iz} + \frac{1}{2} \rho |\vec{v}_{iz}|^2 = p_d \quad (4.19)$$

Adicionalmente se asume que en una zona a la izquierda alejada de la interfaz la presión es cero, por lo tanto

$$p_d = \frac{1}{2} \rho |\vec{v}_{iz}|^2 \quad (4.20)$$

A este resultado se le conoce como presión dinámica, pues es un término de presión que viene dado por el movimiento del fluido.

Pasando al caso de interés, donde un material impacta sobre un blanco, se tomará como marco de referencia la interfaz proyectil – blanco, es decir, justo sobre este marco se está produciendo la penetración. Teniendo a la derecha de dicho marco material del blanco y a la izquierda el proyectil.

Asumiendo que la interfaz proyectil – blanco se mueve con velocidad u , que es la velocidad de penetración, y que el proyectil se mueve con velocidad v

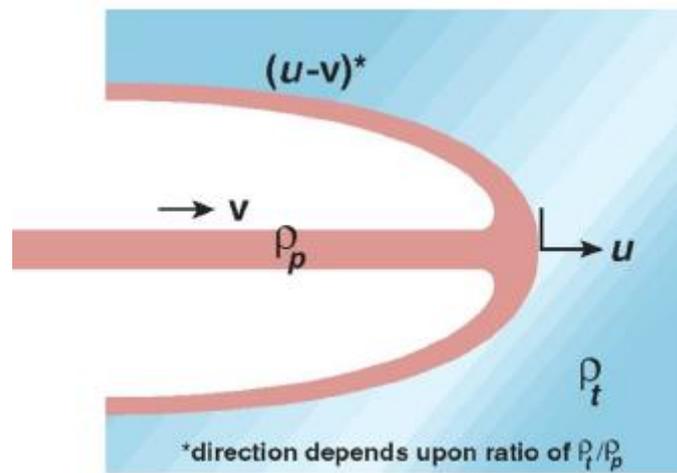


Figura 8. Esquema de las velocidades en el proceso de penetración. Obtenido de Anderson Jr. Charles E. "Analytical Models for Penetration Mechanics: A review".

Siguiendo estas definiciones, con respecto a la interfaz, el material del blanco se mueve hacia esta con velocidad $-u$ y un punto cercano a dicha interfaz del proyectil se mueve con velocidad $(v - u)$.

En el punto de estancamiento las presiones a ambos lados son iguales, por lo tanto

$$p_{iz} = p_d \quad (4.21)$$

$$v_{iz} = (v - u) \quad (4.22)$$

$$v_d = -u \quad (4.23)$$

Sustituyendo en la ecuación (4.18) recordando que no se contemplan fuerzas de volumen y que se asume que la presión en la zona izquierda alejada de la interfaz es cero

$$\frac{1}{2} \rho_p (v - u)^2 = \frac{1}{2} \rho_b (-u)^2 \quad (4.24)$$

Donde los subíndices p y b indican proyectil y blanco respectivamente.

Con esta ecuación se comenzaron a realizar cálculos para hallar velocidades de penetración para proyectiles de carga hueca en la Segunda Guerra Mundial [4]. Este modelo es el de *aproximación hidrodinámica*.

Si se analiza detenidamente la ecuación, podrá observarse que no existen términos que se opongan al avance del proyectil dentro del blanco, cuando es bien sabido que este último presentará una oposición a ser penetrado. Por ello Tate y Alekseevskii introdujeron dos términos para representar fuerzas en el blanco y proyectil, que son respectivamente R_t e Y_p . Donde los subíndices t y p se corresponden con “target” y “projectile” que traducido al castellano serían blanco y proyectil.

A partir de este momento se utilizarán estos últimos subíndices para representar los términos del blanco y proyectil, quedando la ecuación del modelo de Tate – Alekseevskii de la siguiente manera

$$\frac{1}{2} \rho_p (v - u)^2 + Y_p = \frac{1}{2} \rho_t u^2 + R_t \quad (4.25)$$

Estos términos añadidos encajan perfectamente en la ecuación de Bernoulli, pues se les trata como una presión a cierta distancia, o una fuerza de volumen, que representarán en cierto modo la resistencia de ambos materiales, dando un efecto de “solidez”, pues estos términos son los encargados de mantener cierta cohesión en los materiales.

Posteriormente se demostrará que el término Y_p puede ser tratado como una constante, mientras que para R_t se ha demostrado que tiene una gran importancia en el modelo y no debe ser tratado como constante si se requiere de precisión. Pero para el caso más simplificado que se aborda en este proyecto se tratará como una constante. Aun así, es importante no confundir este término como una propiedad del material simplemente, ya que se trata de un parámetro encargado de recoger los distintos fenómenos que afectan al proceso de penetración, además de caracterizar la propia resistencia del material.

4.4 Balance de momento a lo largo de la línea central de acción

Para conocer el estado tensional de la interfaz de penetración se introduce el balance de momento a lo largo de su línea de acción. Recordando la ecuación de conservación de momento (obtenida en el Apéndice B)

$$\frac{\partial \sigma_{ji}}{\partial x_j} + \rho b_i = \rho \frac{Dv_i}{Dt} \quad (4.26)$$

Asumiendo que no existen fuerzas de volumen durante el evento (lo cual no es cierto, pero toman muy poco protagonismo en comparación con los demás términos), se elimina este término para facilitar los cálculos, además de que se asumirá que la densidad se mantiene prácticamente constante.

Se busca integrar a lo largo de todo el evento de penetración sobre la línea de simetría, que será el eje x , con la velocidad del proyectil dada por $v_x(x, t)$. Haciendo un cambio de variable para incluir la posición de la interfaz, $v_x(x, t) = v_x(x - x_i(t), \tau)|_{\tau=t}$, donde $x_i(t)$ es precisamente, la posición de la interfaz proyectil-blanco, dependiente del tiempo. A demás se introduce una variable τ encargada de incluir pequeños cambios temporales. Esta variable puede sustituirse en este caso simplemente por t , ya que el modelo de aproximación hidrodinámica asume que no existen cambios temporales. Aun así, se desarrollará sin sustituir para mostrar el caso general.

Sabiendo que la derivada material está definida como

$$\frac{Dv_x}{Dt} = \frac{\partial v_x}{\partial t} + v_x \frac{\partial v_x}{\partial x} \quad (4.27)$$

Si se introducen los cambios de variable anteriormente realizados se obtiene

$$\frac{\partial v_x}{\partial t} + v_x \frac{\partial v_x}{\partial x} = \left[-\frac{dx_i}{dt} \frac{\partial v_x}{\partial x} + \frac{\partial v_x}{\partial \tau} \Big|_{\tau=t} \right] + \frac{1}{2} \frac{\partial v_x^2}{\partial x} \quad (4.28)$$

Reorganizando y dándose cuenta de que el cociente $\frac{dx_i}{dt}$ es la velocidad de penetración u

$$\left[-\frac{dx_i}{dt} \frac{\partial v_x}{\partial x} + \frac{\partial v_x}{\partial \tau} \Big|_{\tau=t} \right] + \frac{1}{2} \frac{\partial v_x^2}{\partial x} = \frac{\partial}{\partial x} \left[v_x \left(\frac{v_x}{2} - u \right) \right] + \frac{\partial v_x}{\partial \tau} \Big|_{\tau=t} \quad (4.29)$$

Si uno integra multiplicando por la densidad para empezar a introducir la conservación de momento

$$\begin{aligned} \int_{x_p}^{x_i} \rho \frac{Dv_x}{Dt} dx &= \int_{x_p}^{x_i} \rho \frac{1}{dx} \left[v_x \left(\frac{v_x}{2} - u \right) \right] dx + \int_{x_p}^{x_i} \rho \frac{\partial v_x}{\partial \tau} \Big|_{\tau=t} dx \\ &= \rho \left[v_x \left(\frac{v_x}{2} - u \right) \right] \Big|_{x_p}^{x_i} + \int_{x_p}^{x_i} \rho \frac{\partial v_x}{\partial \tau} \Big|_{\tau=t} dx \end{aligned} \quad (4.30)$$

Si se reconoce que se está trabajando sobre el proyectil, entonces $\rho = \rho_p$. El límite inferior de integración x_p se corresponde con la parte trasera del proyectil, con velocidad v , mientras que el límite superior x_i indica la nariz del proyectil, con velocidad u . Sustituyendo sobre el primer término

$$\rho_p \left[u \left(\frac{u}{2} - u \right) - v \left(\frac{v}{2} - u \right) \right] = \rho_p \left(-\frac{u^2}{2} - \frac{v^2}{2} + vu \right) = -\frac{1}{2} \rho_p (v - u)^2 \quad (4.31)$$

El lado del proyectil quedaría finalmente como

$$\int_{x_p}^{x_i} \rho \frac{Dv_x}{Dt} dx = -\frac{1}{2} \rho_p (v - u)^2 + \rho_p \int_{x_p}^{x_i} \frac{\partial v_x}{\partial \tau} \Big|_{\tau=t} dx \quad (4.32)$$

Para el blanco los límites de integración van desde la interfaz x_i hasta el infinito $(+\infty)$, con densidad ρ_t y con velocidad en una zona muy profunda del blanco $v_x(+\infty) = 0$

$$\int_{x_i}^{+\infty} \rho \frac{Dv_x}{Dt} dx = -\frac{1}{2}\rho_t (-u)^2 + \rho_t \int_{x_i}^{+\infty} \frac{\partial v_x}{\partial \tau} \Big|_{\tau=t} dx \quad (4.33)$$

Rescatando el término tensional de la ecuación (4.26), la sumatoria de las parciales puede dividirse en componentes cortantes y axiales, y recordando que el tensor de tensiones es simétrico

$$\int_{x_p}^{x_i} \frac{\partial \sigma_{ji}}{\partial x_j} dx = 2 \int_{x_p}^{x_i} \frac{\partial \sigma_{yx}}{\partial y} dx + \int_{x_p}^{x_i} \frac{\partial \sigma_{xx}}{\partial x} dx = 2 \int_{x_p}^{x_i} \frac{\partial \sigma_{yx}}{\partial y} dx + \sigma_{xx} \Big|_{x_p}^{x_i} \quad (4.34)$$

La parte trasera del proyectil y una zona alejada del blanco no presentan tensiones por lo que $\sigma_{xx}(x_p) = \sigma_{xx}(+\infty) = 0$

Como se tiene

$$\frac{\partial \sigma_{ji}}{\partial x_j} = \rho \frac{Dv_i}{Dt} \quad (4.35)$$

La parte del proyectil queda

$$\begin{aligned} 2 \int_{x_p}^{x_i} \frac{\partial \sigma_{yx}}{\partial y} dx + \sigma_{xx} \Big|_{x_p}^{x_i} &= -\frac{1}{2}\rho_p (v-u)^2 + \rho_p \int_{x_p}^{x_i} \frac{\partial v_x}{\partial \tau} \Big|_{\tau=t} dx ; \\ \tilde{\sigma}_{xx}(x_i) &= \frac{1}{2}\rho_p (v-u)^2 - \rho_p \int_{x_p}^{x_i} \frac{\partial v_x}{\partial \tau} \Big|_{\tau=t} dx + 2 \int_{x_p}^{x_i} \frac{\partial \sigma_{yx}}{\partial y} dx \end{aligned} \quad (4.36)$$

Mientras que la del blanco

$$\begin{aligned} 2 \int_{x_i}^{+\infty} \frac{\partial \sigma_{yx}}{\partial y} dx + \sigma_{xx} \Big|_{x_i}^{+\infty} &= -\frac{1}{2}\rho_t (-u)^2 + \rho_t \int_{x_i}^{+\infty} \frac{\partial v_x}{\partial \tau} \Big|_{\tau=t} dx ; \\ \tilde{\sigma}_{xx}(x_i) &= \frac{1}{2}\rho_t u^2 + \rho_t \int_{x_i}^{+\infty} \frac{\partial v_x}{\partial \tau} \Big|_{\tau=t} dx - 2 \int_{x_i}^{+\infty} \frac{\partial \sigma_{yx}}{\partial y} dx \end{aligned} \quad (4.37)$$

Donde la notación $\tilde{\sigma}$ indica que los esfuerzos de compresión serán tratados como positivos, contrario a la norma general, donde esfuerzos axiales de tracción son tratados como positivos mientras que esfuerzos de compresión se tratan como negativos. Como en este tipo de situaciones predominan los esfuerzos de compresión se emplea esta notación por comodidad.

Si uno considera los términos de las integrales como constantes

$$Y_p = -\rho_p \int_{x_p}^{x_i} \frac{\partial v_x}{\partial \tau} \Big|_{\tau=t} dx + 2 \int_{x_p}^{x_i} \frac{\partial \sigma_{yx}}{\partial y} dx \quad (4.38)$$

$$R_t = \rho_t \int_{x_i}^{+\infty} \frac{\partial v_x}{\partial \tau} \Big|_{\tau=t} dx - 2 \int_{x_i}^{+\infty} \frac{\partial \sigma_{yx}}{\partial y} dx \quad (4.39)$$

Y se igualan las tensiones en la interfaz tanto para proyectil como blanco, se obtiene la ecuación constituyente del modelo

$$\tilde{\sigma} = \frac{1}{2} \rho_p (v - u)^2 + Y_p = \frac{1}{2} \rho_t u^2 + R_t \quad (4.40)$$

Si uno aplica las simplificaciones realizadas para la obtención de la ecuación (4.24), es decir, se asume que los materiales no presentan resistencia y el proceso se produce de forma estacionaria, los términos de las integrales quedan reducidos a cero y se llega de igual manera a dicha ecuación.

El hecho de que los materiales no presenten resistencia implica que la integral de los términos de tensión cortante sea cero y, por otro lado, el estado estacionario implica que no existe una variación de propiedades con el tiempo. Por lo tanto, se estaría hablando de la aproximación hidrodinámica.

Reconociendo que la tasa de cambio de longitud del proyectil es $dL/dt = -(v - u)$ y sustituyendo en la ecuación (4.24), asumiendo que todo el proyectil es consumido por erosión y que

$$\int u dt = P \quad (4.41)$$

Donde P es la profundidad penetrada. Se obtiene el límite hidrodinámico de penetración

$$\left(\frac{P}{L}\right)_{hyd} = \sqrt{\frac{\rho_p}{\rho_t}} \quad (4.42)$$

4.5 Deceleración de un proyectil impactando sobre una pared rígida

En el apartado (4.1) se ha hablado de ondas de choque generadas por el contacto entre proyectil y blanco. Estas ondas generan una desaceleración en el proyectil, por lo tanto, si se obtiene la ecuación que describe como varía la velocidad del proyectil fruto de ese contacto, se tendría posesión de todas las ecuaciones que permitirán describir el comportamiento del proyectil, pudiendo por tanto realizar cálculos para analizar dicho comportamiento.

El desarrollo para obtener la ecuación mencionada anteriormente es extenso y tedioso por lo que se realizará una breve explicación de la metodología para llegar a dicha ecuación sin incluir los cálculos, que pueden ser comprobados por el lector según vienen en el libro de James Walker [5].

Al hablar de ondas de choque es necesario incluir las condiciones de “salto” de Hugoniot, o las condiciones de Rankine-Hugoniot. Desarrolladas por Rankine [6] y Hugoniot [7]. Estas condiciones son

fruto de la derivación de las leyes de conservación de masa, momento y energía, quedando de la siguiente manera

$$\frac{\rho_r}{\rho_l} = 1 - \frac{u_l - u_r}{U - u_r} \quad (4.43)$$

$$\tilde{\sigma}_l - \tilde{\sigma}_r = \rho_r(U - u_r)(u_l - u_r) \quad (4.44)$$

$$E_l - E_r = \frac{1}{2}(\tilde{\sigma}_l + \tilde{\sigma}_r) \left(\frac{1}{\rho_r} - \frac{1}{\rho_l} \right) \quad (4.45)$$

Donde los subíndices l y r significan (traducidos del inglés) izquierda y derecha respectivamente, referido a las condiciones a izquierda o derecha del frente de la onda de choque. Siendo U la velocidad de dicho frente, u la velocidad de una partícula y E la energía en una sección.

A partir de estas condiciones y realizando un estudio elástico-plástico de un proyectil impactando sobre una pared rígida, asumiendo que no existe zona plástica dentro de este y que la velocidad del sonido en el material es infinita, se llega a la expresión que determina el cambio temporal de la velocidad del proyectil

$$\frac{dv}{dt} = -\frac{Y_p}{\rho_p L} \quad (4.46)$$

4.6 El modelo de Tate - Alekseevskii

Juntando la variación de longitud

$$\frac{dL}{dt} = -(v - u) \quad (4.47)$$

Con las ecuaciones (4.25) y (4.46), se obtiene el modelo completo, que permitirá describir de forma aproximada el comportamiento del proceso de penetración.

Es necesario destacar que se está introduciendo un proyectil de longitud finita, contradictorio a las simplificaciones realizadas para el desarrollo de la ecuación de Bernoulli, pues no se estaría cumpliendo la condición de estado estacionario. Aun así, se ha observado que el modelo es aproximadamente correcto y la información obtenida de este se acerca razonablemente a valores experimentales.

Si uno despeja la velocidad de penetración u de la ecuación (4.25) y reorganiza términos para evitar que el denominador pueda ser igual a cero para casos de misma densidad de proyectil y blanco

$$u = \frac{\rho_p v^2 - 2(R_t - Y_p)}{\rho_p v + \sqrt{\rho_p \rho_t v^2 + 2(\rho_p - \rho_t)(R_t - Y_p)}} \quad (4.48)$$

La velocidad de penetración u siempre será menor o igual que la velocidad de la cola del proyectil v .

A continuación, se presentan dos casos fruto del análisis de la ecuación (4.25).

Si se toma el caso donde la velocidad de penetración u es igual a la del proyectil v

$$Y_p = \frac{1}{2}\rho_t v^2 + R_t = \tilde{\sigma} \quad (4.49)$$

Y se despeja la velocidad del proyectil para este caso

$$v_r = \sqrt{\frac{2(Y_p - R_t)}{\rho_t}} \quad (4.50)$$

Se observa que $Y_p > R_t$ y la tensión en la interfaz es igual a Y_p . Por lo tanto, si la velocidad v del proyectil es menor que v_r

$$\tilde{\sigma} = \left(\frac{1}{2}\rho_t v^2 + R_t \right) < Y_p \quad (4.51)$$

La tensión en la interfaz es menor que la resistencia del proyectil, por lo tanto, este no se deforma. Además, la tensión que soporta el blanco es mayor que R_t y, por ende, ocurre la penetración. Se está hablando de penetración rígida, ya que el proyectil no sufre erosión.

Por otro lado, si la velocidad del proyectil es mayor que v_r , entonces la tensión que soporta este es superior a su resistencia y sufrirá erosión hasta que su velocidad sea igual a v_r . A partir de este momento seguirá el comportamiento indicado en el párrafo anterior hasta detenerse completamente.

Otro caso de interés es cuando la velocidad de penetración u es igual a cero

$$R_t = \frac{1}{2}\rho_p v^2 + Y_p = \tilde{\sigma} \quad (4.52)$$

Procediendo de igual manera que para el caso anterior

$$v_p = \sqrt{\frac{2(R_t - Y_p)}{\rho_p}} \quad (4.52)$$

Aquí $R_t > Y_p$ y la tensión en la interfaz es igual a R_t . Con lo cual, si $v < v_p$ la tensión quedaría como

$$\tilde{\sigma} = \left(\frac{1}{2} \rho_p v^2 + Y_p \right) < R_t \quad (4.53)$$

Si se da esta condición, la tensión en la interfaz sería menor que la resistencia del blanco, con lo que no existiría penetración. Aun así, la tensión a la que está sometido el proyectil es superior a su resistencia Y_p , con lo que sufrirá erosión. Si la velocidad de impacto es superior a v_p ocurre penetración con erosión hasta que el proyectil alcanza la condición $v = v_p$.

5. Resultados y discusión

5.1 Algoritmo de cálculo

El programa se ejecuta según el diagrama

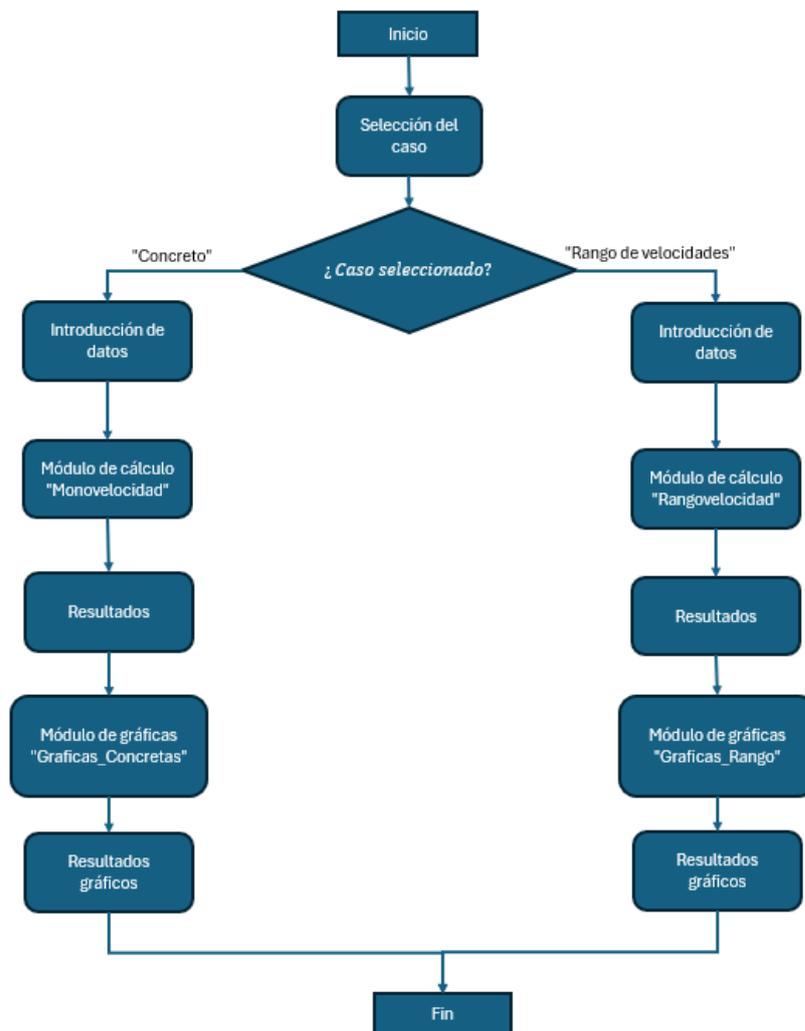


Figura 9. Diagrama de flujo del programa base.

El módulo de cálculo se ejecuta de acuerdo con el siguiente diagrama de flujo

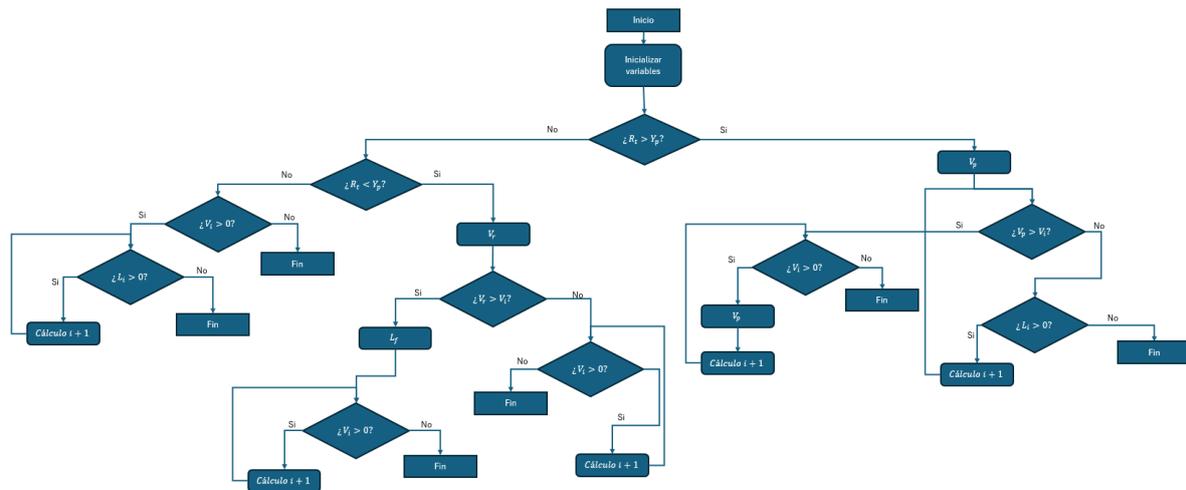


Figura 10. Diagrama de flujo para módulo de cálculo (caso único de penetración).

En cada iteración, si procede, se calculan variables que se utilizarán en el siguiente paso y las propias del paso. Por ejemplo, las tensiones σ se calculan en su respectivo paso i , mientras que la velocidad se calcula para el paso $i + 1$, por simplicidad en la programación.

El funcionamiento para el caso de múltiples velocidades es exactamente el mismo que para el caso de una velocidad, con la diferencia de que se hacen todos los cálculos para cada velocidad, pero dando como salida únicamente la penetración total.

5.2 Funcionamiento del programa

Una vez se dispone de las herramientas para poder iniciar los cálculos, se procede a analizar el ejemplo de un caso real para comprobar su validez.

Se ha decidido implementar el modelo mediante el lenguaje de programación *Python*. El programa está pensado para un sistema proyectil – blanco donde se introducen los datos correspondientes a ambos, se realizan los cálculos pertinentes y ofrece como salida múltiples parámetros.

Se comienza creando una interfaz gráfica que ofrecerá dos opciones: Realizar una simulación para un caso concreto (“Caso único”), o evaluar un caso para un rango de velocidades determinado (“Rango de velocidades”).

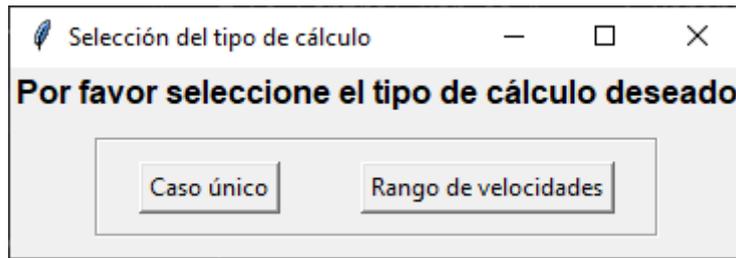


Figura 11. Interfaz de selección de caso de penetración.

En función de la opción seleccionada se ejecutará el módulo de cálculo correspondiente para la situación escogida.

Ambos módulos siguen la misma metodología con la diferencia de que para el caso único, los resultados de cada paso (o incremento de tiempo) se almacenan para posteriormente poder mostrar de forma gráfica el proceso completo. Mientras que para el caso del rango de velocidades solo se almacenarán los resultados finales de cada proceso con su correspondiente velocidad de impacto. Es decir, si se pide analizar 3 casos con velocidades de impacto 200, 300 y 400 [m/s] por ejemplo, se almacenarán tres penetraciones totales, una por cada velocidad de impacto.

Si se pulsa sobre el botón “Caso único” aparece la siguiente ventana

Figura 12. Interfaz de introducción de datos para “Caso único” (vacía).

Dividida en tres secciones principales: “Datos del proyectil”, “Datos del blanco” y “Parámetros de simulación”. En esta ventana se deben introducir correctamente todos los parámetros, pudiendo seleccionar las unidades de cada uno mediante las listas desplegadas situadas a la derecha.

Figura 13. Interfaz de introducción de datos para “Caso único” (rellena).

Si faltase algún dato o se detectara que se ha introducido un parámetro incorrecto (como por ejemplo una letra) aparecerá un mensaje de error que alertará del fallo para poder ser corregido.

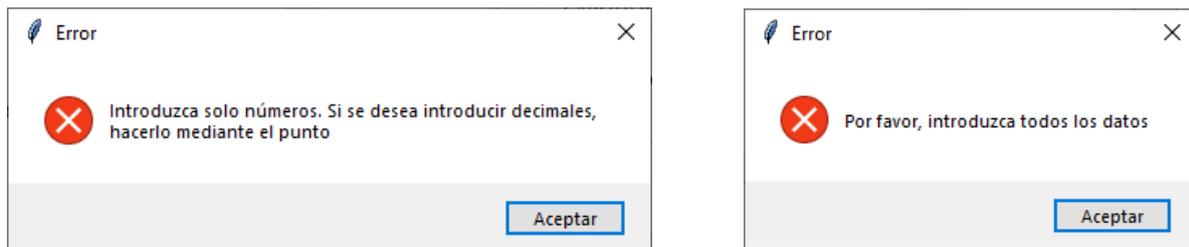


Figura 14. Posibles mensajes de error.

Una vez se presiona el botón de “Iniciar Cálculo” se ejecutan todos los cálculos con los parámetros de entrada registrados anteriormente.

Cuando termina el cálculo aparece un cuadro de diálogo indicando cual ha sido la causa del cese de la penetración, si la ha habido.

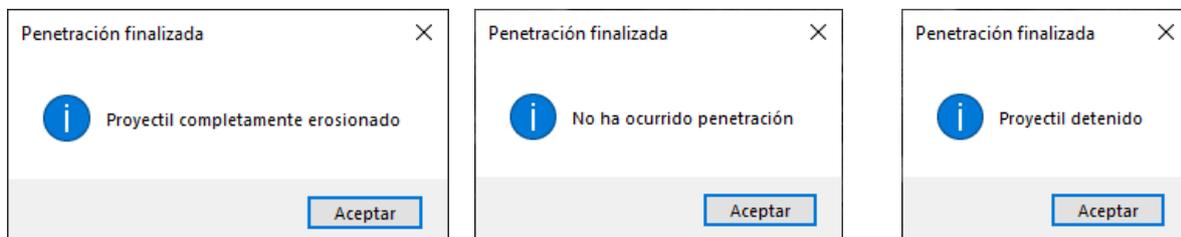


Figura 15. Mensajes indicadores de posibles condiciones de parada alcanzadas.

Para el caso de múltiples velocidades el método es muy similar, aparecerá una ventana igual al anterior caso, pero esta vez sin la entrada de “Velocidad de impacto”

The screenshot shows a software window titled "Entrada de parámetros". It is divided into three main sections:

- Datos del proyectil:** Contains three rows of input fields. Each row has a text label (Densidad, Esfuerzo de fluencia, Longitud) on the left, a numerical input box in the middle, and a dropdown menu on the right.
- Datos del blanco:** Contains two rows of input fields. Each row has a text label (Densidad, Esfuerzo de fluencia) on the left, a numerical input box in the middle, and a dropdown menu on the right.
- Parámetros de simulación:** Contains one row with the label "Delta de t" on the left, a numerical input box in the middle, and a dropdown menu on the right.

At the bottom center of the window is a button labeled "Siguiete".

Figura 16. Interfaz de introducción de datos para el caso “Rango de velocidades”.

Una vez se introducen correctamente todos los parámetros y se presiona en el botón “Siguiete” aparece una nueva ventana donde se deberá especificar la velocidad de impacto mínima, la máxima y los incrementos de velocidad con los que se quiere trabajar

The screenshot shows a software window titled "Rango de velocidades". It features a main heading "Introduzca el rango de velocidades" and three input fields:

- Velocidad inferior [m/s]:** A numerical input box on the left.
- Velocidad superior [m/s]:** A numerical input box on the right.
- Incrementos de velocidad [m/s]:** A numerical input box centered below the other two.

At the bottom center is a button labeled "Iniciar Cálculo".

Figura 17. Interfaz de introducción de velocidades para el caso “Rango de velocidades”.

Una vez se introducen los datos correctamente y se presiona el botón de “Iniciar Cálculo”, se ejecuta el módulo de cálculo.

El código empleado para el programa completo se ha incluido en el Anexo C, donde se mostrará completo.

A continuación se describe el funcionamiento de los módulos de cálculo, comenzando por el correspondiente a la selección de un caso concreto de penetración.

Una vez se introducen los parámetros para comenzar la simulación, se realiza una distinción entre los casos discutidos en la sección anterior, en base a si el término R_t es mayor que Y_p o si son iguales. Una

vez se determina el caso, se realiza un bucle que irá realizando los cálculos para cada paso de la simulación. El paso está definido por Δt , por lo tanto, si se le otorga un valor muy pequeño se obtendrá mayor precisión, al poder disponer de un mayor número de datos. Una mayor precisión se consigue a costa de aumentar el coste computacional. Debe tenerse en cuenta que el evento de penetración presenta una duración del orden de los microsegundos.

Es importante recordar que para el cálculo numérico los términos diferenciales se aproximan a incrementos tal que

$$\frac{dv}{dt} \approx \frac{\Delta v}{\Delta t} \quad (5.1)$$

Con lo que para tratar ecuaciones como, por ejemplo, la ecuación (4.46) se realiza la siguiente transformación

$$\frac{dv}{dt} = -\frac{Y_p}{\rho_p L} \rightarrow \Delta v \approx -\frac{Y_p}{\rho_p L} \Delta t \quad (5.2)$$

Y la nueva velocidad para el siguiente paso sería la suma de la anterior más el incremento calculado.

Tras cumplir una condición de parada, el programa devuelve la recopilación de los valores calculados para poder ser mostrados de forma gráfica. En este caso (caso único de penetración) se muestra un cuadro de diálogo indicando el motivo de la parada (Figura 13), otro cuadro indicando la penetración en milímetros y múltiples gráficas:

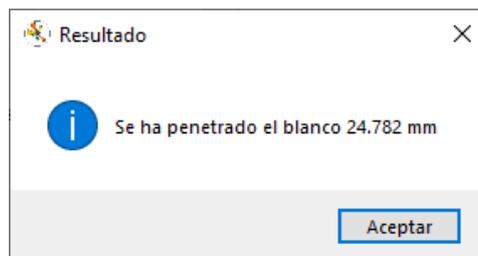
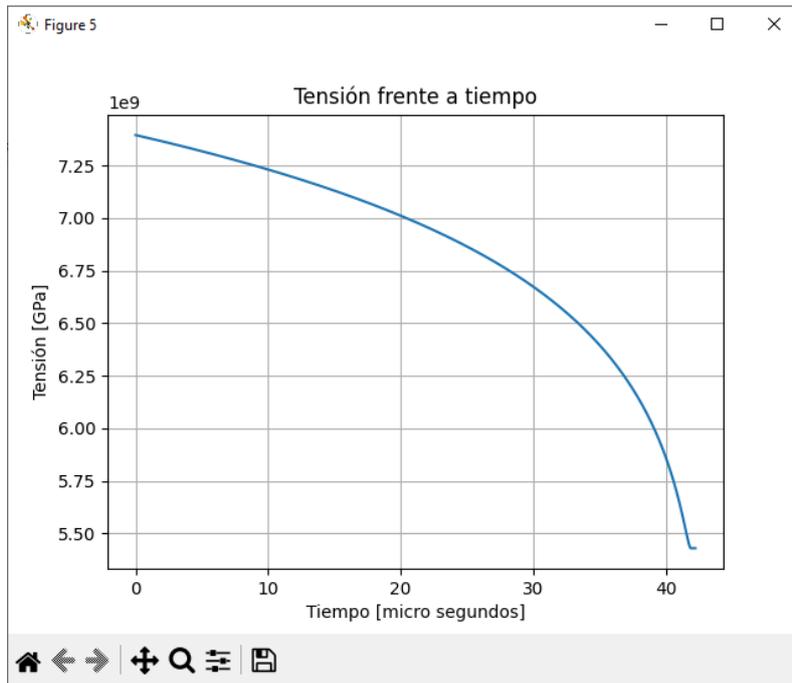
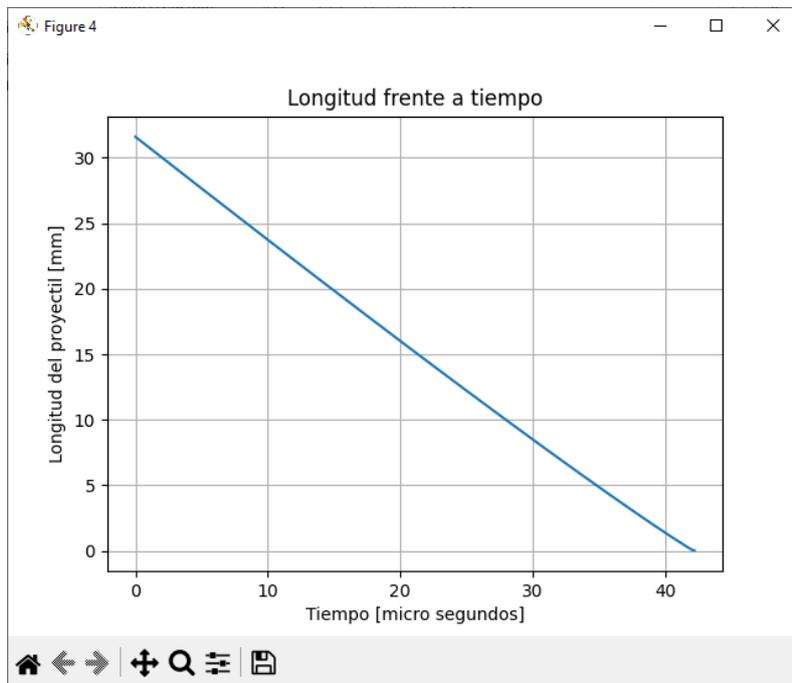


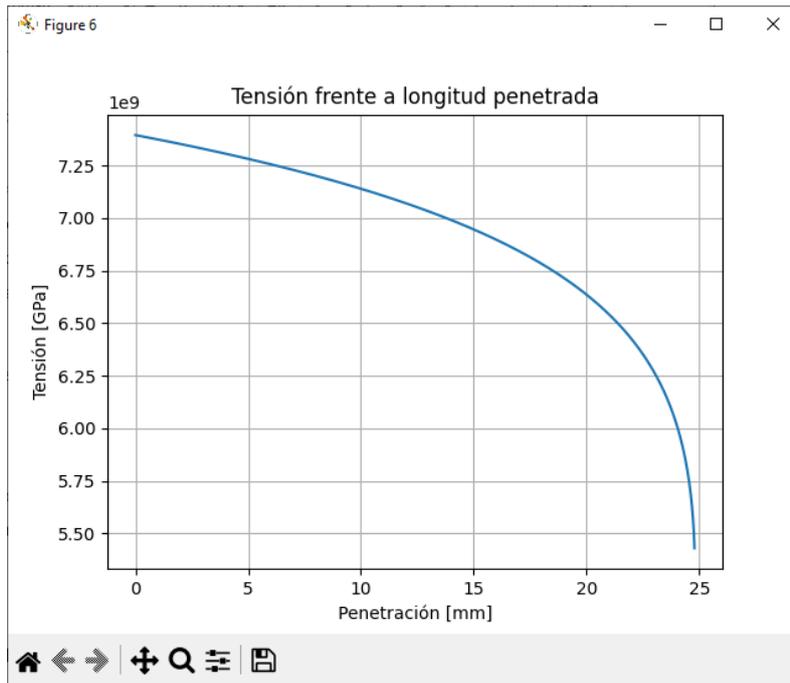
Figura 18. Mensaje indicador de penetración final.



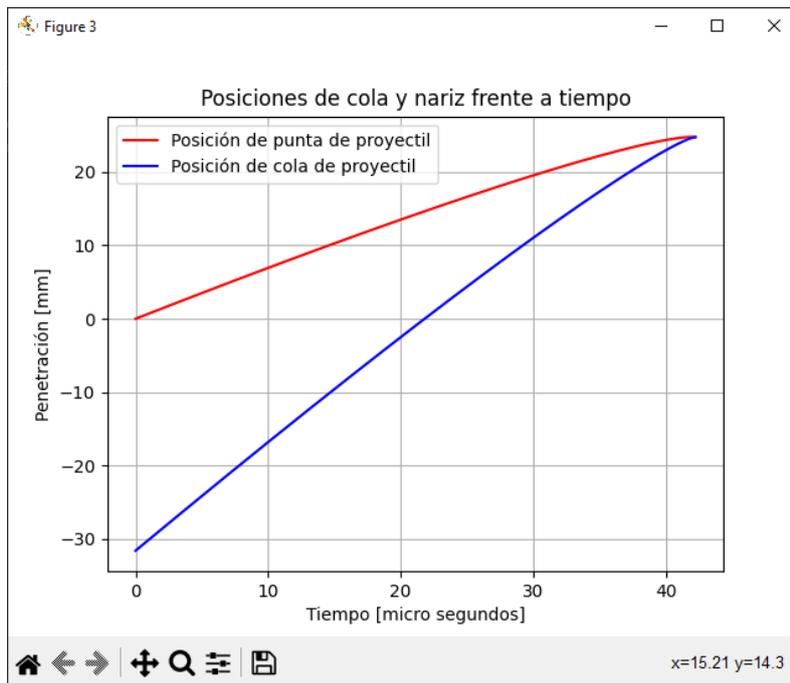
Gráfica 1. Tensión en la interfaz frente a tiempo.



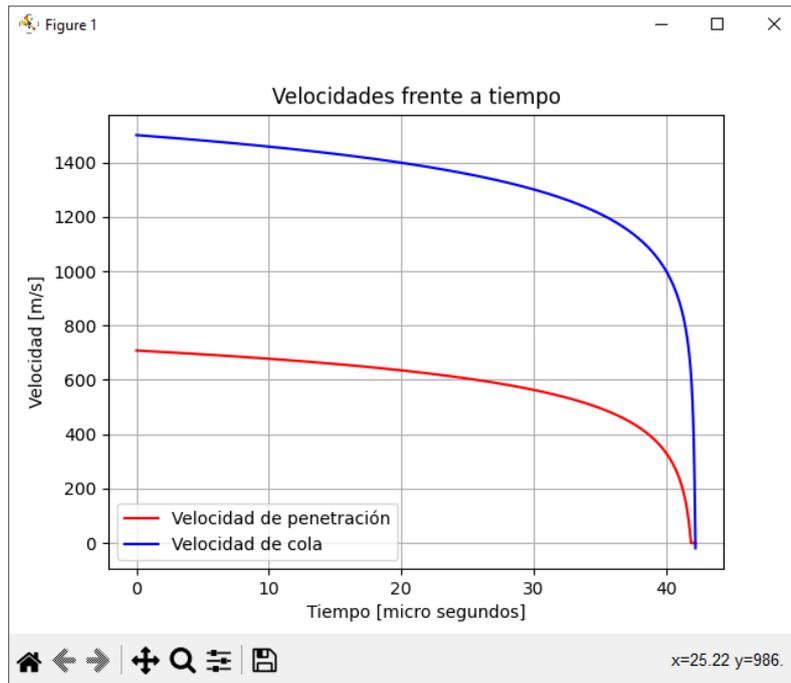
Gráfica 2. Longitud del proyectil frente a tiempo.



Gráfica 3. Tensión en la interfaz frente a distancia penetrada.

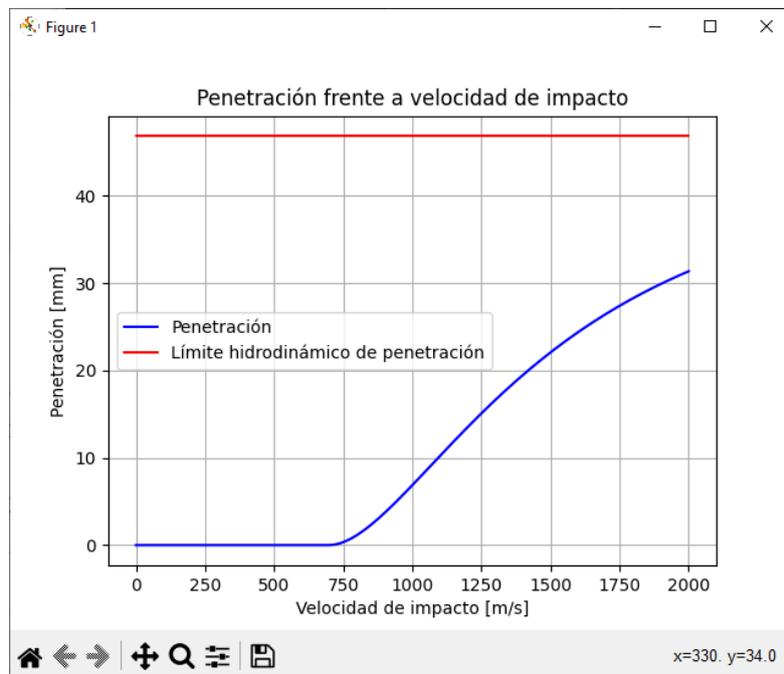


Gráfica 4. Posición de punta y cola del proyectil frente a tiempo.



Gráfica 5. Perfil de velocidades frente a tiempo.

Si se realiza una simulación sobre un rango de velocidades solo se mostrará una gráfica



Gráfica 6. Penetración frente a velocidad de impacto.

La metodología para el cálculo del caso de un rango de velocidades es muy similar al anterior, con la diferencia de que solo se almacena el resultado de la penetración final para cada velocidad de impacto.

Cuando se le pide al usuario que introduzca los datos mostrados en la Figura 15, se genera una lista que contiene todas las velocidades de impacto, para posteriormente realizar los cálculos de igual manera que en el módulo de “Caso único” con cada velocidad inicial almacenada en esta lista. De esta manera, tras finalizar los cálculos para cada velocidad, se obtiene una lista que almacena cada penetración final, para poder ser mostrada gráficamente. Además, se representa el límite hidrodinámico de penetración, que proviene de la ecuación (4.42).

5.3 Validación del modelo

Para determinar si el modelo puede considerarse como “correcto” o preciso, es necesario realizar una comparación con resultados reales de experimentación. Para ello se tomará un experimento de penetración llevado a cabo por Anderson, Walker y Hauver [8]. Posteriormente se compararán resultados experimentales de penetración [9],[10],[11] con múltiples configuraciones proyectil – blanco para reforzar la validez del modelo en distintos casos de penetración.

Este experimento se compone de un evento real de penetración y una simulación mediante hidrocódigo “*Eulerian wave propagation code CTH*”

La configuración es de un proyectil de aleación de Wolframio, concretamente Kennametal W-10 con una composición 90W-7Ni-3Fe de densidad 17.2 g/cm^3 . El blanco es un acero S-7 con densidad 7.84 g/cm^3 . El blanco se dispara contra el proyectil a una velocidad de 1.5 km/s y se obtienen los datos de penetración mediante rayos X.

El proyectil tiene una longitud de 31.6 mm y se ha tomado un valor para Y_p de 2 GPa , que representa de forma razonable la resistencia dinámica del material.

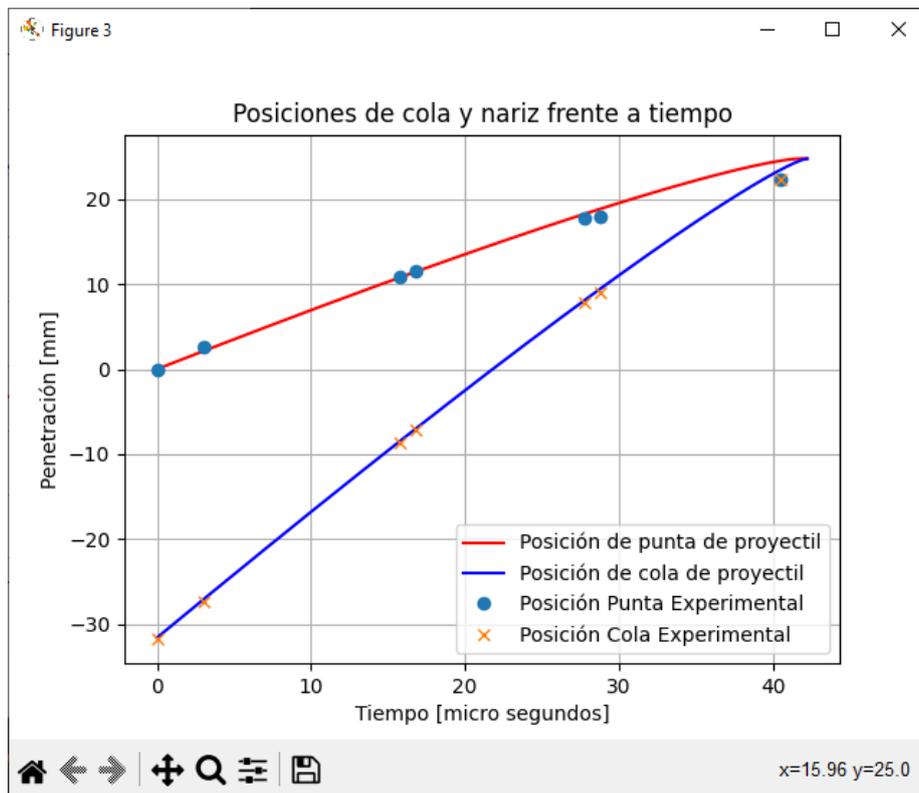
Para el caso de R_t se ha ajustado a un valor de 5.43 GPa , para que la penetración final obtenida de la simulación con el hidrocódigo sea igual que la del modelo. A continuación se presentan los resultados experimentales obtenidos

Tabla 1. Resultados experimentales. Obtenida de J.Walker, "Modern Impact and Penetration Mechanics"

t (μs)	z_i (mm)	z_p (mm)
0.0	0.0	-31.8
3.0	2.6	-27.3
15.8	10.8	-8.7
16.8	11.5	-7.2
27.8	17.7	7.9
28.8	18.0	9.1
40.5 ^a	22.3	final depth

^aEstimated

Si se grafica de forma conjunta tanto los resultados experimentales como los propios del modelo

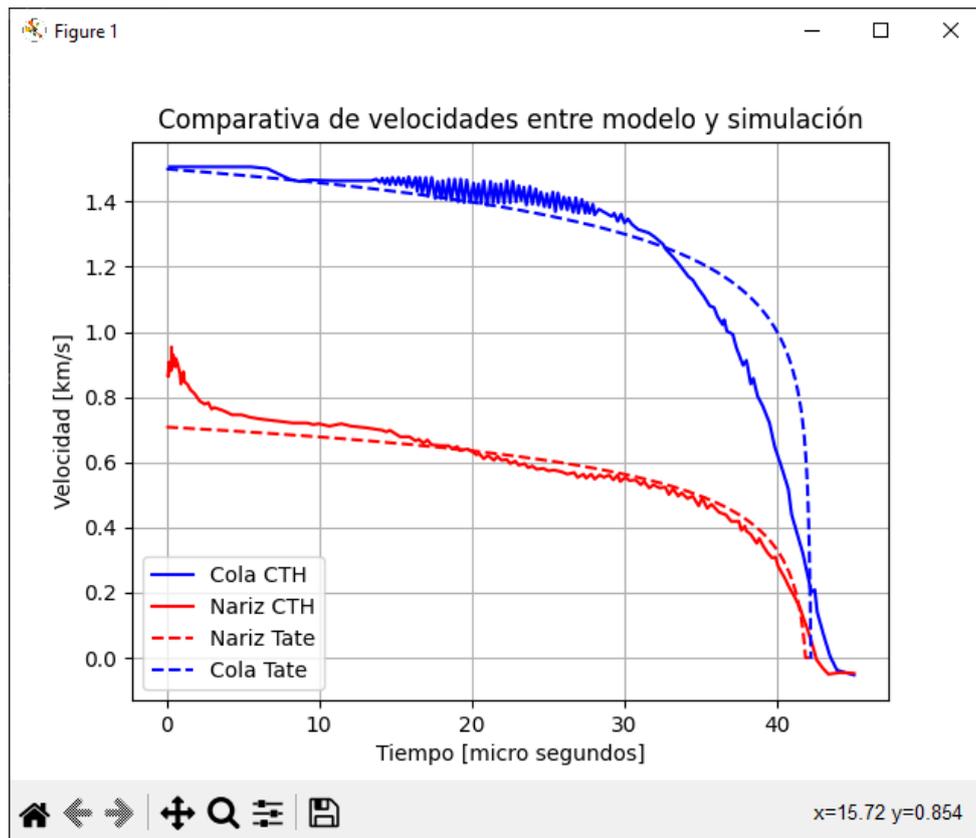


Gráfica 7. Comparativa del modelo con resultados experimentales. Con $R_t = 5.43 \text{ GPa}$.

Se puede observar que la tendencia del modelo se aproxima bastante a la realidad. Aun así, hay que destacar que la penetración real obtenida es de 22.3 mm, mientras que el modelo predice una penetración de 24.78 mm, lo que incurre en un error del 11%, que no puede ser menospreciado.

Aunque a simple vista las curvas obtenidas con el modelo se acercan bastante a los datos experimentales, pueden observarse dos zonas donde existe más discrepancia.

Al comienzo de la penetración se había hablado en el apartado 4.1 que se generaba una onda de choque en el momento en el que el proyectil hace contacto con el blanco. Esta onda de choque genera un estado no estacionario que no es capturado por el modelo de Tate, y, claramente, es influyente en el comportamiento del proceso. Aun así, este efecto perdura por muy poco tiempo y, aunque no es capturado por el modelo, no genera una gran desviación con respecto a los datos reales. De todas formas, esta desviación no debe pasarse por alto, ya que se está ignorando su efecto en los primeros instantes de la penetración.

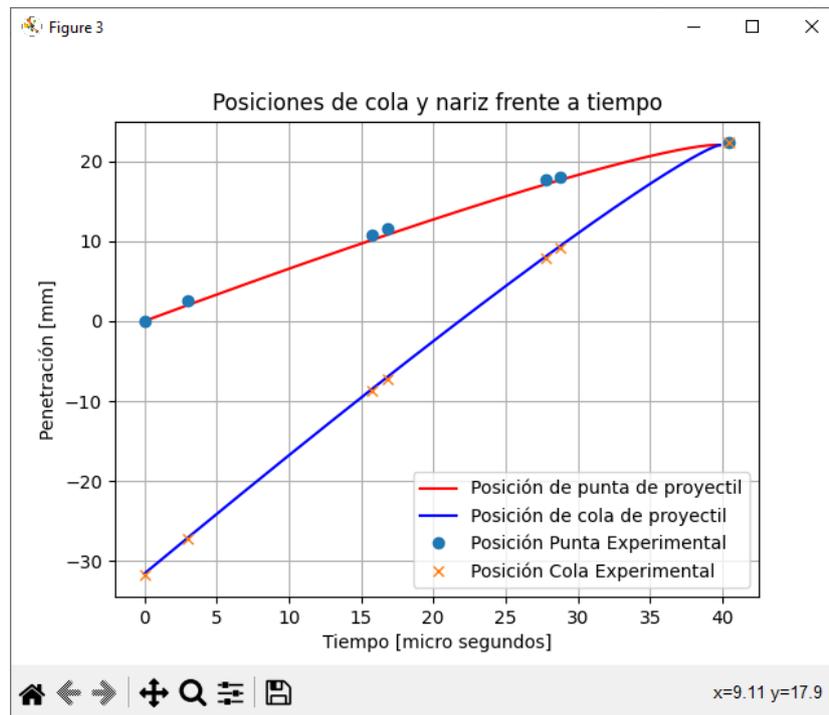


Gráfica 8. Comparativa de simulación numérica con CTH y el modelo de Tate.

Como puede observarse la simulación numérica muestra este estado producido por la onda de choque al comienzo de la penetración, con una duración aproximada de $3\mu s$. Tras estabilizarse, el modelo de Tate captura correctamente el comportamiento de la interfaz, pues se llega a un estado casi estacionario, hasta que se llega a la siguiente zona donde se producen discrepancias entre el modelo y los datos experimentales, que es la zona final.

Por lo tanto, puede decirse que el modelo de Tate captura de forma correcta el comportamiento para el estado estacionario (que es el predominante) del evento de penetración.

Para igualar la profundidad de penetración con respecto a los datos experimentales, se selecciona un valor de R_t de 6.18 GPa



Gráfica 9. Comparativa del modelo con resultados experimentales. Con $R_t = 6.18$ GPa.

Puede observarse que las posiciones de la cola del proyectil se ven bien representadas por el modelo. Como esta posición se ve muy influenciada por el parámetro Y_p , debido a la ecuación (4.46), uno podría afirmar que este parámetro no varía durante el evento de penetración, al coincidir razonablemente los datos experimentales y del modelo. Por lo tanto, la deceleración del proyectil debido a las ondas reflejadas en este describe correctamente su comportamiento en este caso.

Si se desea obtener una mayor precisión, queda demostrado que el término R_t no puede ser constante y debe variar para tener en cuenta efectos transitorios varios que se suceden durante el evento.

Por ello, posteriormente, Tate dividió el evento de penetración en varias partes para tener en cuenta estos efectos [10], [12].

$$P = P_{tr} + P_{Tate\ model} + P_{af} \quad (5.3)$$

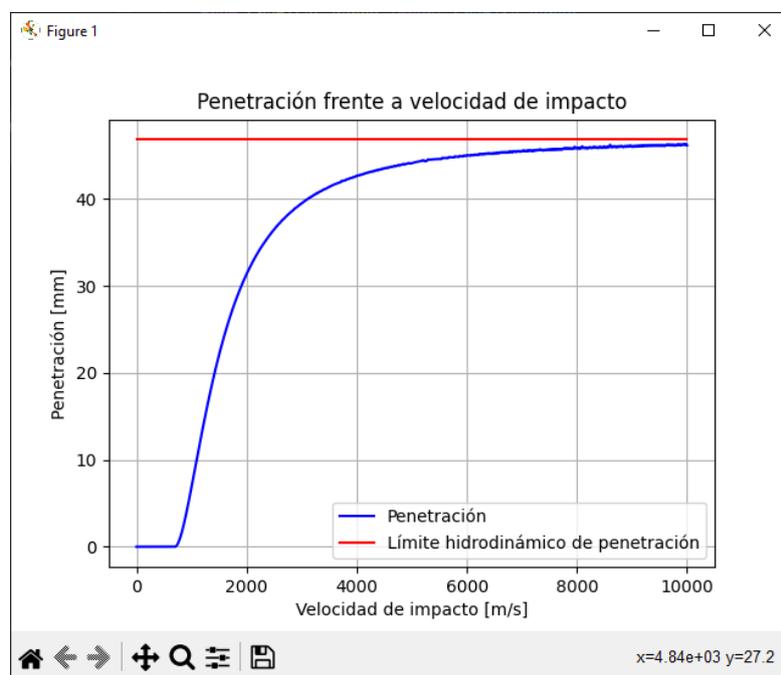
En donde cada término por orden sería, la penetración en la fase inicial transitoria, penetración durante el estado estacionario y penetración de la fase final.

En ningún momento se ha hablado del impacto que tiene en la penetración el diámetro del proyectil, pues en este proyecto se está evaluando el modelo simplificado de Tate – Alekseevskii. Aun así, se ha demostrado experimentalmente [13] que el diámetro, concretamente el cociente L/D , toma un papel

importante en el evento y debe ser incluido para poder predecir con precisión el comportamiento de la interfaz de penetración.

Esto se consigue mediante la introducción del comportamiento del cráter generado en el blanco, pues este sufre unas deformaciones que dan lugar a zonas elásticas y plásticas en su interior, que a su vez generan tensiones en la interfaz, alterando su comportamiento. El efecto de la cavidad generada toma mayor importancia al final de la penetración, pues como bien se explicó en el apartado 4.1 de introducción, a medida que se desacelera la interfaz, las propiedades mecánicas de los materiales toman un mayor protagonismo, pues con velocidades más reducidas el proceso se aleja de tener un comportamiento similar al de un fluido.

Si para este mismo caso se grafican múltiples velocidades variando de 0 a 10 km/s se obtiene



Gráfica 10. Penetración frente a rango de velocidades.

Con lo que se observa un comportamiento asintótico hacia el límite hidrodinámico de penetración. Lo que implica que a medida que aumenta la velocidad, tanto el comportamiento del blanco como del proyectil se asemeja más al de un fluido.

Puede parecer que estas velocidades son excesivas, pero si se tiene en cuenta que la velocidad de un satélite en órbita puede llegar a los 10 km/s entonces se estarían representando casos realistas de penetración. Por ejemplo, si un satélite en órbita impacta contra un material metálico de basura espacial, podría utilizarse este modelo para estimar los daños causados en dicho satélite.

Para los demás datos experimentales se han escogido como materiales una aleación de Wolframio (D17), un acero ST-52 y un acero HzB20 W (acero alemán para blindaje).

Las diferentes propiedades de estos materiales quedan resumidas en la siguiente tabla

Tabla 2. Propiedades de materiales de experimentación

Material	ρ [g/cm^3]	Y_p [GPa]	R_t [GPa]
D17	17	1,55	5,72
HzB20	7,85		5,18
Acero ST-52	7,85		4,4
Aleación de Al 1	2,85	1,38	
Plomo	11,2		0,33

Y con los resultados de la experimentación

Tabla 3. Resultados de aleación de wolframio (D17) impactando sobre aleación de wolframio (D17).

V (m/s)	P/L	V (m/s)	P/L
544	0,018	1.514	0,636
715	0,050	1.815	0,779
848	0,100	2.382	0,982
1.033	0,243	2.985	1,071
1.195	0,400	3.093	1,125
1.260	0,436	3.733	1,286

Tabla 4. Resultados de aleación de wolframio (D17) impactando sobre HzB20 (acero alemán para blindaje).

V (m/s)	P/L	V (m/s)	P/L
407	0,020	1.497	0,768
646	0,072	1.835	1,133
829	0,148	2.117	1,264
835	0,157	2.549	1,400
885	0,178	2.814	1,457
975	0,258	3.397	1,536
1.273	0,582	3.760	1,607
1.348	0,653		

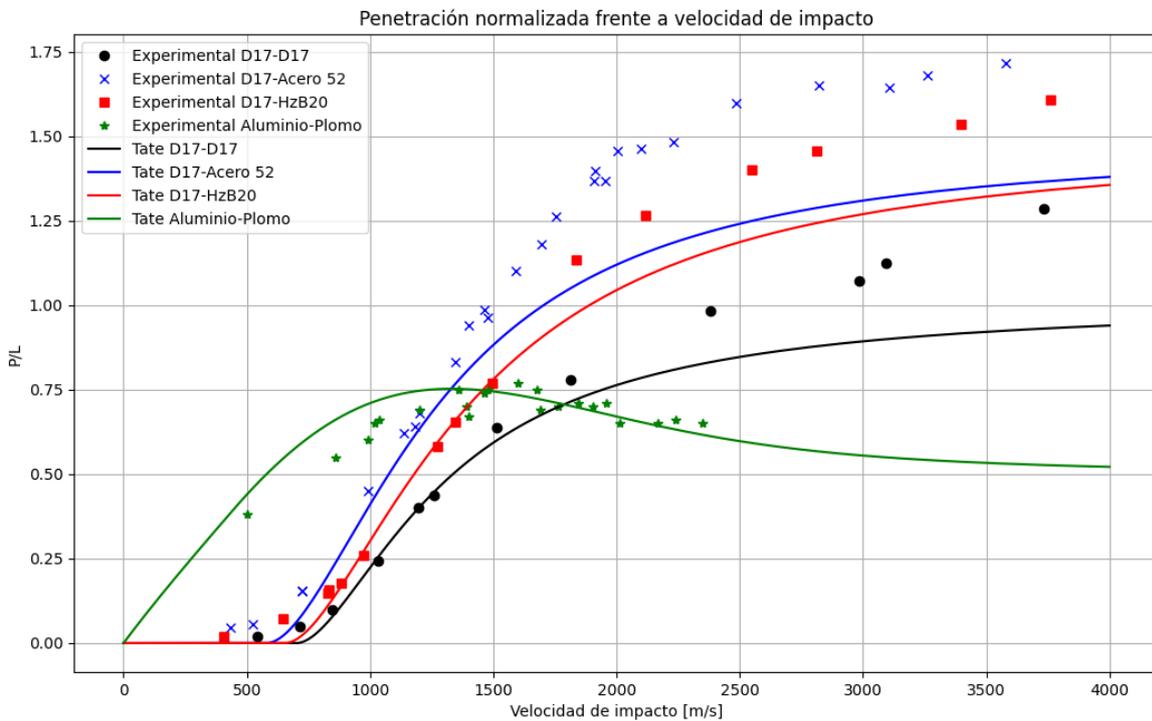
Tabla 5. Resultados de aleación de wolframio (D17) impactando sobre acero ST-52.

V (m/s)	P/L	V (m/s)	P/L
434	0,045	1.696	1,180
523	0,055	1.756	1,262
722	0,153	1.910	1,368
723	0,155	1.912	1,397
993	0,450	1.952	1,367
1.137	0,621	2.004	1,457
1.183	0,640	2.100	1,464
1.201	0,680	2.232	1,483
1.346	0,833	2.484	1,596
1.400	0,939	2.820	1,650
1.464	0,987	3.107	1,643
1.478	0,962	3.263	1,679
1.591	1,100	3.578	1,714

Tabla 6. Resultados de aleación de aluminio impactando sobre plomo

V (m/s)	P/L	V (m/s)	P/L
500	0,38	1.598	0,77
860	0,55	1.677	0,75
994	0,60	1.689	0,69
1.018	0,65	1.762	0,70
1.036	0,66	1.844	0,71
1.201	0,69	1.902	0,70
1.360	0,75	1.957	0,71
1.393	0,70	2.012	0,65
1.402	0,67	2.168	0,65
1.463	0,74	2.240	0,66
1.476	0,75	2.347	0,65

Se procede a representar los resultados experimentales frente al modelo programado, obteniendo la siguiente gráfica



Gráfica 11. Comparativa de resultados experimentales para diferentes configuraciones proyectil-blanco frente a modelo de Tate simplificado.

Hay que destacar que el caso del aluminio impactando contra el plomo es el único caso donde el término R_t es superior al Y_p .

Como puede observarse el modelo predice con precisión las penetraciones en un rango entre 885 y 1273 m/s para las configuraciones que toman como blanco la aleación de wolframio D17 y el acero Hzb20. Adicionalmente, parece que se estiman correctamente penetraciones para los casos de un proyectil y blanco de aleación de wolframio D17 en el rango de 848 a 1195 m/s y para un proyectil D17 frente a un blanco de acero ST-52 entre 723 y 1137 m/s.

Como bien se explicó en la comparativa con los datos del experimento llevado a cabo por Anderson, Walker y Hauver, existen dos zonas donde el modelo se aleja de la realidad, concretamente la fase inicial y final del evento de penetración. Esto podría explicar el porqué de las discrepancias con los datos experimentales para velocidades elevadas de impacto, pues la acumulación de error resultante de no tener en cuenta efectos transitorios parece que tiene un impacto negativo en la validez del modelo para el límite superior de velocidades.

Por otro lado, se puede observar que a medida que aumenta la velocidad para el caso del conjunto aluminio-plomo, se reduce progresivamente la distancia penetrada. Este comportamiento fue predicho por Tate para los casos donde se $Y_p > R_t$. Por lo tanto, contrario al sentido común, diseñar un proyectil demasiado resistente podría llegar a ser contraproducente en términos de eficacia según este modelo.

5.4 Aplicación del modelo y el límite balístico

El límite balístico de un blanco es una velocidad V_{bl} por debajo de la cual este no es completamente penetrado por un proyectil. Por lo tanto, es la velocidad mínima con la que debe impactar un proyectil para lograr una penetración completa.

Aprovechando que existe experimentación con proyectiles de aleación de wolframio D17 y acero Hzb20, y se tiene posesión de sus parámetros, se va a realizar una simulación para comenzar la fase preliminar de un diseño realista de blindaje.

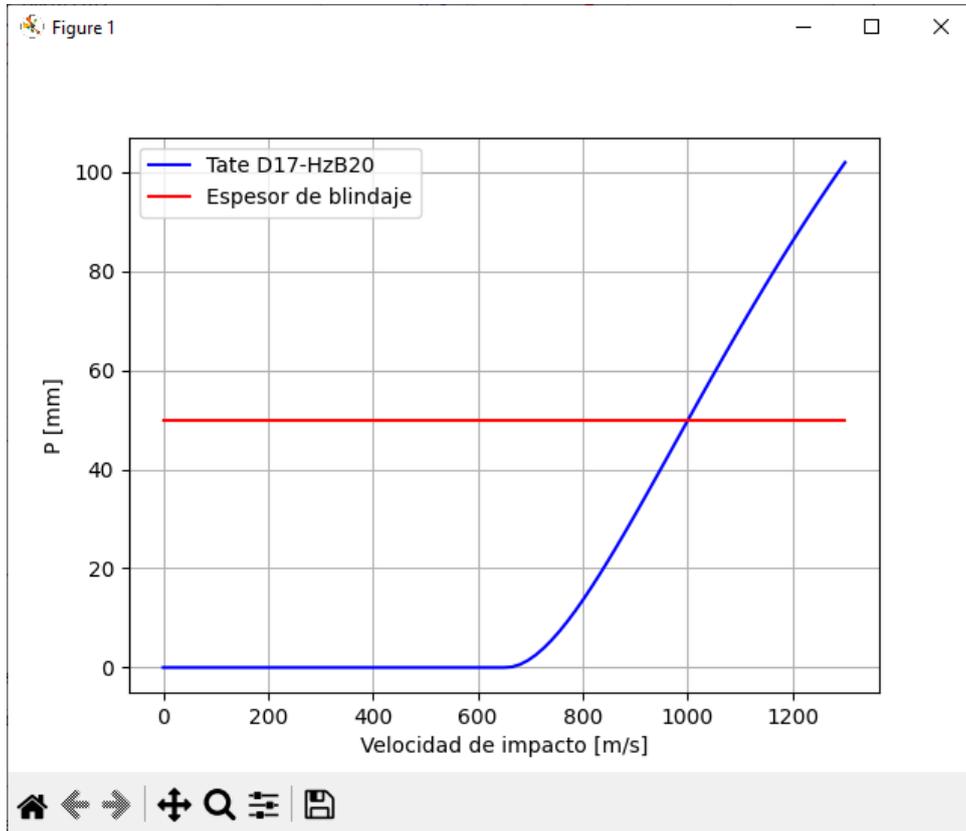
Como se ha visto que el modelo funciona correctamente para un rango de velocidades entre 885 y 1273 m/s, se selecciona como amenaza un proyectil cinético compuesto de wolframio D17 con velocidad de impacto entre los 900 y los 1200 m/s, con una longitud de proyectil de 16,5 cm.

Se desea diseñar un elemento protector para un vehículo de combate de infantería. Los blindajes de estos vehículos se encuentran en torno a los 50 mm, por lo que se escoge este valor de espesor para las planchas de acero Hzb20 montadas en los laterales de este vehículo.

A pesar de que la velocidad de salida del proyectil se encuentra en los rangos mencionados anteriormente, se asumirá que el rango típico de distancia entre la amenaza y el blanco se sitúa sobre los

500 metros. Por lo que, si se tiene en cuenta la desaceleración del proyectil para esta distancia, la velocidad de impacto podría verse reducida por unos 100 m/s.

Si se grafica el caso



Gráfica 12. Representación de caso de penetración realista

De donde se obtiene que para velocidades de impacto superiores a 1027 m/s, el blindaje no será efectivo, ya que el proyectil lo atravesará completamente. El límite balístico en base al modelo de Tate para esta configuración de proyectil-blanco es, por tanto

$$V_{bl} = 1027 \text{ m/s} \quad (5.4)$$

Por lo tanto, si se desea continuar con este diseño, deberá tenerse en cuenta que el vehículo probablemente no será seguro en ambientes urbanos donde pueden darse enfrentamientos frente a esta amenaza a muy poca distancia, recordando que, a menor distancia de enfrentamiento, mayor velocidad de impacto.

6. Conclusiones

En este trabajo se ha implementado en un código de Python el modelo analítico de Tate y Alekseevskii para penetraciones balísticas. Tras los resultados obtenidos se pueden establecer las siguientes conclusiones:

- El modelo simplificado de penetración de Tate – Alekseevskii predice de forma razonable el comportamiento del proyectil para la fase mayoritaria del evento de penetración, que es la de estado estacionario. Por lo tanto, la analogía hidrodinámica puede tomarse como correcta para visualizar el funcionamiento de este proceso para impactos a altas velocidades.
- Es discutible que este modelo simplificado pueda servir como base para el diseño de un blindaje (o proyectil), más bien es un modelo que puede servir como primera toma de contacto para comprender el comportamiento de la interfaz de penetración o para realizar una estimación rápida en una etapa de diseño preliminar.
- Si se desea diseñar un elemento de protección balístico, sería recomendable o bien utilizar otros modelos más completos, como el de Walker – Anderson [14], o introducir las modificaciones propuestas por Tate para tener en cuenta los estados transitorios y así poder realizar diseños más realistas.

Teniendo en cuenta que los elementos de protección balística sirven para evitar bajas humanas principalmente, es necesario que estos sean diseñados con la mayor precisión posible para poder cumplir su objetivo. Si se ignora el error del modelo, se estaría cometiendo una irresponsabilidad como diseñador, pues este error puede acarrear la pérdida de grandes cantidades de dinero si se logra inutilizar un vehículo, o peor, de vidas humanas.

Es necesario tener en cuenta que, si se utilizan modelos más complicados, el coste computacional aumentará. Para el caso de este proyecto un ordenador convencional ha sido capaz de realizar los cálculos referentes a múltiples casos de rango de velocidades con saltos de tiempo del orden de nanosegundos en menos de un minuto, por lo que este modelo puede ser adecuado para tener una primera idea de cómo abordar conceptualmente un elemento de protección balística.

Referencias

1. *A theory for the deceleration of long rods after impact.* **Tate, A.** 1967, J. Mech. Phys. Solids, Vol. 15, págs. 387-399.
2. *Penetration of a rod into a target at high velocity.* **Alekseevskii, V.P.** 2, 1969, Combustion, Explosion, and Shock Waves, Vol. 2, págs. 63-66.
3. *Use of Shock Waves in High Pressure Physics.* **Al'tshuler, L. V.** 1, 1965, Soviet Physics Uspekhi, Vol. 8, págs. 52 - 91.
4. *Explosives with lined cavities.* **Birkhoff, G., y otros.** 1948, Journal of Applied Physics, Vol. 19, págs. 563 - 582.
5. **Walker, James D.** *Modern Impact and Penetration Mechanics.* s.l. : Cambridge University Press, 2021.
6. *On the Thermodynamic Theory of Waves of Finite Longitudinal Disturbance.* **Rankine, W. J. Macquorn.** 1870, Philosophical Transactions of the Royal Society of London, Vol. 160, págs. 277-288.
7. *Mémoire sur la propagation des mouvements dans les corps et spécialement dans les gaz parfaits (première partie).* **Hugoniot, H.**
8. *Target resistance for long-rod penetration into semi-infinite targets.* **Anderson, C. E., Walker, J. D., Hauver, G.E.** 1992, Nuclear Engineering and Design, Vol. 138, págs. 93 - 104.
9. **C. E. Anderson, Jr., B. L. Morris, D. L. Littlefield.** *Penetration data: A Penetration Mechanics Database.* San Antonio, TX : Southwest Research Institute, 1992. SwRI Report 3593/001.
10. *Long rod penetration models-part II. Extensions to the hydrodynamic theory of penetration.* **Tate, A.** 9, 1986, International Journal of Engineering Science, Vol. 28, págs. 599-612.
11. **C. E. Anderson, Jr., B. L. Morris, D. L. Littlefield.** *Penetration data: A Penetration Mechanics Database.* San Antonio, TX : Southwest Research Institute, 1992. SwRI Report 3593/001
12. *Long rod penetration models-part I. A flow field model for high speed long rod penetration.* **Tate, A.** 8, 1986, International Journal of Engineering Science, Vol. 28, págs. 535-548.
13. *On the L/D effect for long-rod penetrators.* **Anderson, C.E., Walker, J. D., Bless, S.J., Partom, Y.** 3, 1996, International Journal of Impact Engineering, Vol. 18, págs. 247-264.
14. *A nonsteady-state model for penetration.* **Walker, J. D., Anderson, C. E., Jr.** s.l. : National Defence Research Establishment, 1992, 13th International Symposium on Ballistics, Vol. 3, págs. 9-16

Anexos

Anexo A. Expresiones matemáticas y simbología

Notación vectorial

Si se tiene un sumatorio de distintas componentes de cierto vector genérico

$$\sum_i^n x_i \quad (\text{A. 1})$$

Por simplicidad y para no ocupar mucho espacio, esta misma operación se escribirá de la siguiente manera

$$x_i \quad (\text{A. 2})$$

Entendiendo que la ecuación anterior implica un sumatorio de las componentes de x , de la componente 1 a la n .

Por ejemplo, el producto vectorial entre dos vectores \vec{a} y \vec{b} se escribiría

$$\vec{a} \cdot \vec{b} = a_1 b_1 + a_2 b_2 + a_3 b_3 = a_i b_i \quad (\text{A. 3})$$

Delta de Kronecker

La delta de Kronecker es una función a trozos que permite escribir de forma más compacta ciertas operaciones. Está definida como

$$\delta_{ij} = \begin{cases} 0 & \text{si } i \neq j \\ 1 & \text{si } i = j \end{cases} \quad (\text{A. 4})$$

Como ejemplo, la matriz identidad I puede definirse mediante la delta de Kronecker tal que

$$I_{ij} = \delta_{ij} \quad (\text{A. 5})$$

Símbolo de Levi-Civita o de permutación

El símbolo de permutación satisface las siguientes propiedades

$$\varepsilon_{ijk} = \begin{cases} +1 & \text{si } (i, j, k) \text{ es } (1, 2, 3), (2, 3, 1), (3, 1, 2) \\ -1 & \text{si } (i, j, k) \text{ es } (3, 2, 1), (1, 3, 2), (2, 1, 3) \\ 0 & \text{si } i = j, j = k, k = i \end{cases} \quad (\text{A. 6})$$

Por ejemplo, si se tiene X como una matriz que almacena componentes de una función genérica A . Siendo los índices i, j la posición dentro de la matriz y el índice k la componente de la función

$$X = \begin{pmatrix} 0 & -A_3 & A_2 \\ A_3 & 0 & -A_1 \\ -A_2 & A_1 & 0 \end{pmatrix} = -\varepsilon_{ijk} A_k \quad (\text{A. 7})$$

El operador Nabla

$$\nabla = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} \right) \quad (\text{A.8})1$$

Este operador puede actuar sobre campos escalares y vectoriales. Es un vector de derivadas parciales, por lo tanto, permite obtener la tasa de variación de estos campos con respecto al espacio.

Si se aplica el operador Nabla a una función escalar cualquiera F , entonces

$$\nabla F = \left(\frac{\partial F}{\partial x}, \frac{\partial F}{\partial y}, \frac{\partial F}{\partial z} \right) \quad (\text{A.9})$$

La operación devuelve un vector que indica cuanto varía la función F con respecto a una dirección del sistema cartesiano. Otra forma de escribir este resultado es mediante el uso de la función *gradiente*

$$\text{grad } F = \frac{\partial F}{\partial x} \mathbf{i} + \frac{\partial F}{\partial y} \mathbf{j} + \frac{\partial F}{\partial z} \mathbf{k} \quad (\text{A.10})$$

Siendo i, j, k los vectores unitarios en las direcciones x, y, z respectivamente.

Otro resultado interesante que deriva de utilizar el operador nabla es la *divergencia*.

Si se define F como una función vectorial cualquiera tal que

$$\mathbf{F}(x, y, z) = F_x(x, y, z)\mathbf{i} + F_y(x, y, z)\mathbf{j} + F_z(x, y, z)\mathbf{k}$$

La divergencia viene dada por

$$\text{div } \mathbf{F} = \nabla \cdot \mathbf{F} \quad (\text{A.11})$$

Donde

$$\nabla \cdot \mathbf{F} = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} \right) \cdot (F_x, F_y, F_z) = \frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y} + \frac{\partial F_z}{\partial z} \quad (\text{A.12})$$

Es importante no confundir las funciones divergencia y gradiente, pues el gradiente de un campo escalar asigna un vector que apunta en la dirección de mayor variación de dicho campo, mientras que la divergencia de un campo vectorial asigna una cantidad (un escalar) indicadora de la tendencia atractiva o repulsiva al flujo de un punto.

Si la divergencia es positiva, al punto se le denomina *fuentes*, pues de ese punto sale más flujo del que entra. Por otro lado, si la divergencia fuese negativa, al punto se le refiere como *sumidero*, pues dicho punto atrae más flujo del que “repele”.

Del producto vectorial, o producto cruz, se obtiene el *rotacional* de un campo vectorial F

$$\text{rot } \mathbf{F} = \nabla \times \mathbf{F} \quad (\text{A.13})$$

Esta función devuelve un vector que indica el eje y sentido de rotación de un flujo además de la magnitud de dicha rotación.

Anexo B. Leyes físicas y teoremas

La derivada material

Sea una función F que depende del espacio y del tiempo tal que $F = F(x,y,z,t)$, de la cual se desea conocer su variación con respecto al tiempo y a su posición. Para obtener esta variación se introduce el concepto de *derivada material* D/Dt .

Si se toma un elemento diferencial de F tal que

$$d\mathbf{F}(x, y, z, t) = \frac{\partial F}{\partial x} dx + \frac{\partial F}{\partial y} dy + \frac{\partial F}{\partial z} dz + \frac{\partial F}{\partial t} dt \quad (\text{B.1})$$

Y se divide por dt

$$\frac{d\mathbf{F}}{dt} = \frac{\partial F}{\partial x} \frac{dx}{dt} + \frac{\partial F}{\partial y} \frac{dy}{dt} + \frac{\partial F}{\partial z} \frac{dz}{dt} + \frac{\partial F}{\partial t} \frac{dt}{dt} \quad (\text{B.2})$$

Se puede observar que los elementos dx_i/dt corresponden con las velocidades en la dirección i , siendo

$i = 1$ La dirección x

$i = 2$ La dirección y

$i = 3$ La dirección z

Y que el cociente dt/dt es igual a uno, obteniendo finalmente la definición de derivada material

$$\frac{d\mathbf{F}}{dt} = \frac{\partial F}{\partial x} v_x + \frac{\partial F}{\partial y} v_y + \frac{\partial F}{\partial z} v_z + \frac{\partial F}{\partial t} = \frac{D\mathbf{F}}{Dt} \quad (\text{B.3})$$

Que escrito de forma más compacta sería

$$\frac{D\mathbf{F}}{Dt} = \frac{\partial F}{\partial t} + \bar{v} \cdot \overline{\text{grad}}(F) \quad (\text{B.4})$$

La derivada material proporciona información sobre el estado de una función dependiente del tiempo y del espacio, lo cual es de especial interés, por ejemplo, en un fluido.

Teorema de la divergencia o de Gauss-Ostrogadsky

El teorema de la divergencia dictamina que para un volumen de material V contenido en una superficie S con un campo vectorial \vec{f} continuo y diferenciable en \mathbb{R}^3

$$\int_V \text{div}(\vec{f}) d\text{vol} = \int_S \vec{f} \cdot \vec{n} ds \quad (\text{B.5})$$

Donde el vector \vec{n} es normal a la superficie S y ds es un diferencial de superficie.

Físicamente se crea una relación que explica lo que sucede en el interior del volumen en base a cuanto flujo de cierto campo entra o sale a través de la superficie que delimita dicho volumen.

Teorema del transporte de Reynolds

Este teorema permite describir como una propiedad varía con el tiempo dentro de un volumen de control que puede estar en movimiento, por ejemplo, si se tiene una propiedad $\delta = \delta(t, x, y, z)$

$$\frac{d}{dt} \int_{V(t)} \delta d\text{vol} = \int_{V(t)} \frac{\partial \delta}{\partial t} d\text{vol} + \int_{S(t)} \delta \vec{v} \cdot \vec{n} dS \quad (\text{B.6})$$

La aplicación del teorema de la divergencia proporciona

$$\frac{d}{dt} \int_{V(t)} \delta d\text{vol} = \int_{V(t)} \left(\frac{\partial \delta}{\partial t} + \text{div}(\vec{v})\delta \right) d\text{vol} \quad (\text{B.7})$$

Ley de conservación de la masa

Partiendo del principio base de que la masa no se crea ni se destruye, se conserva. Si se escribe la variación temporal de dicha masa

$$\frac{d}{dt} \int_V \rho d\text{vol} = 0 \quad (\text{B.8})$$

$$\int_V \frac{\partial \rho}{\partial t} d\text{vol} = - \int_S \rho \vec{v} \cdot \vec{n} dS \quad (\text{B.9})$$

Esta igualdad describe la cantidad de masa que fluye fuera de una superficie. Como se ha dicho antes, este término debe ser nulo. Si se combinan las dos ecuaciones y se hace uso del teorema de la divergencia se obtiene

$$\int_V \left(\frac{\partial \rho}{\partial t} + \text{div}(\rho \vec{v}) \right) d\text{vol} = 0 \quad (\text{B.10})$$

Al tratarse de un volumen arbitrario, finalmente se obtiene la ecuación de conservación de la masa

$$\frac{\partial \rho}{\partial t} + \text{div}(\rho \vec{v}) = 0 \quad (\text{B.11})$$

Ley de conservación de momento

La conservación de momento proviene de la segunda ley de Newton $\vec{F} = m\vec{a}$, de donde se considera que en un cuerpo actúan tanto fuerzas de volumen como de superficie. Las fuerzas de volumen actúan sobre todo el volumen de dicho cuerpo y las de superficie sobre el área que delimita al volumen.

$$\vec{F} = \int_{S(t)} \vec{t} ds + \int_{V(t)} \rho \vec{b} dvol \quad (\text{B.12})$$

Donde \vec{t} es el vector de tensión.

Si se establece la relación entre el vector de tensión y el tensor de tensiones tal que $t_i = n_j \sigma_{ji}$ y se sustituye en la ecuación (B.12)

$$\vec{F} = \int_{S(t)} \vec{n} \sigma ds + \int_{V(t)} \rho \vec{b} dvol \quad (\text{B.13})$$

Recordando el teorema de la divergencia

$$\int_{S(t)} \vec{n} \sigma ds = \int_{V(t)} \text{div}(\sigma) \hat{e} dvol \quad (\text{B.14})$$

Siendo \hat{e} el vector unitario normal a la respectiva componente del tensor de tensiones. Aplicando notación de índices y combinando expresiones se obtiene

$$\vec{F} = \int_{V(t)} \frac{\partial \sigma_{ji}}{\partial x_j} \hat{e}_{x_i} dvol + \int_{V(t)} \rho \vec{b} dvol \quad (\text{B.15})$$

Sabiendo que el momento \vec{p} de un material viene dado por

$$\vec{p} = \int_{V(t)} \rho \vec{v} dvol \quad (\text{B.16})$$

O en su forma ya integrada

$$\vec{p} = m\vec{v} \quad (\text{B.17})$$

Si se aplica el teorema del transporte de Reynolds a la ecuación (B.16) se obtiene

$$\frac{d\vec{p}}{dt} = \int_{V(t)} \rho \frac{D\vec{v}}{Dt} dvol \quad (\text{B.18})$$

Recordando que una variación de momento genera una fuerza que trata de impedir dicha variación y, por tanto, $\vec{F} = \frac{d\vec{p}}{dt}$

$$\int_{V(t)} \frac{\partial \sigma_{ji}}{\partial x_j} \hat{e}_{x_i} dvol + \int_{V(t)} \rho \vec{b} dvol = \int_{V(t)} \rho \frac{D\vec{v}}{Dt} dvol \quad (\text{B.19})$$

Teniendo en cuenta que las integrales son sobre un volumen arbitrario

$$\frac{\partial \sigma_{ji}}{\partial x_j} + \rho b_i = \rho \frac{Dv_i}{Dt} \quad (\text{B.20})$$

Siendo la ecuación (B.20) la forma base de la conservación de momento.

Ley de conservación de la energía

La energía en un sistema, de modo similar a la masa, no se crea ni se destruye, se transfiere, por lo tanto

$$\frac{d}{dt} \int_V \tilde{E} dvol = 0 \quad (\text{B.21})$$

Donde E es la energía específica total, que se descompone en la propia energía interna del material e y la energía cinética de este. Es importante destacar que la energía cinética debida al movimiento de las partículas viene incluida en la energía interna e . También hay que recordar que una propiedad específica no es más que dicha propiedad por unidad de masa.

$$\tilde{E} = \rho E = \rho \left(\frac{1}{2} v_i v_i + e \right) \quad (\text{B.22})$$

Siendo \tilde{E} la energía total por unidad de volumen.

La variación total de energía para cierto volumen viene dada por

$$\frac{d}{dt} \int_V \tilde{E} dvol = \int_V \frac{\partial \tilde{E}}{\partial t} dvol = - \int_S \tilde{E} \vec{v} \cdot \vec{n} dS + \int_S \vec{t} \cdot \vec{v} dS + \int_V \rho \vec{b} \cdot \vec{v} dvol + \int_V \rho r dvol - \int_S \vec{q} \cdot \vec{n} dS \quad (\text{B.23})$$

Donde los términos a la derecha de la igualdad son, en orden, la energía que sale del volumen, el trabajo realizado por tracciones en su superficie; y el que deriva de la acción de fuerzas de volumen, energía depositada en el volumen (como podría ser por radiación, reacciones químicas, etc) y el flujo de calor a través de la superficie.

Si se emplea el teorema de la divergencia, el tensor de tensiones y se reconoce que se trata de un volumen arbitrario

$$\frac{\partial \tilde{E}}{\partial t} + \frac{\partial}{\partial x_i} (\tilde{E} v_i) - \frac{\partial}{\partial x_i} (\sigma_{ij} v_j) - \rho b_i v_i - \rho r + \frac{\partial q_i}{\partial x_i} = 0 \quad (\text{B.24})$$

Obteniendo la ecuación de conservación de energía.

Tensor de deformación

Si se toma un tensor que almacene los gradientes de la velocidad, que se denominará L

$$L_{ij} = \frac{\partial v_i}{\partial x_j} \quad (\text{B.25})$$

Se puede dividir en una parte simétrica y antisimétrica tal que

$$L_{ij} = \frac{1}{2} (L + L^T)_{ij} + \frac{1}{2} (L - L^T)_{ij} = \frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) + \frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} - \frac{\partial v_j}{\partial x_i} \right) \quad (\text{B.26})$$

De donde se obtiene

$$D_{ij} = \frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) \quad (\text{B.27})$$

$$W_{ij} = \frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} - \frac{\partial v_j}{\partial x_i} \right) \quad (\text{B.28})$$

Siendo D_{ij} el *tensor de deformación* y W_{ij} el *tensor rotacional*. El tensor de deformación contiene información sobre cómo se deforma un material para cada punto de este, mientras que el tensor rotacional almacena el rotacional de la velocidad, describiendo, por tanto, la rotación del material en cada punto.

Anexo C. Programación del modelo simplificado de Tate – Alekseevskii

Se adjunta el código escrito para el programa clasificado por el nombre del script al que pertenecen

Script base

```
from Seleccion_Caso import Seleccion

Control_Var = Seleccion()# -----> Obtención del tipo de cálculo deseado

# Si Control_Var es 0 ----> Deberá introducirse una velocidad

# Si Control_Var es 1 ----> Deberá introducirse un rango de velocidades

#INICIALIZACIÓN DE CASO CONCRETO DE CÁLCULO

if Control_Var == 0:

    from Introd_Datos import Almac_Datos

    from Monovelocidad import Caso_Concreto

    from Graficas_Concretas import Graf_Concreto

    Entradas = Almac_Datos()#-----> Entrada y almacenamiento de datos

    Resultados = Caso_Concreto(Entradas)#-----> Se inicia el cálculo con las entradas anteriores

    Graf_Concreto(Resultados)# -----> Una vez obtenidos los resultados se realizan varias gráficas

else: #INICIALIZACIÓN DE CASO CON MÚLTIPLES VELOCIDADES

    from Rangos import Selecc_Rango

    from Rangovelocidad import Casos_Multiples

    from Introd_Datos_Mod import Almac_Datos_Mod

    from Graficas_Rango import Grafica_Rango

    import numpy as np

    Entradas_rango = Almac_Datos_Mod()

    Rangos = Selecc_Rango()#-----> Introducción de rango de velocidades
```

```

h = round((Rangos[1]-Rangos[0])/Rangos[2]) #-----> Divisiones de velocidad
V = np.linspace(Rangos[0],Rangos[1],h) #-----> Lista de velocidades

cont = 0

i = [0]

lim_hyd = np.sqrt(Entradas_rango[0]/Entradas_rango[1])*Entradas_rango[6]*1000 #-----> Límite
hidrodinámico de penetración en mm

Lista_P = []

hyd =[]

for velocidad in V:

    Entradas_rango[2] = velocidad

    Resultados = Casos_Multiples(Entradas_rango)

    Lista_P.append(Resultados[3])

    hyd.append(lim_hyd)

    if velocidad < Rangos[1]:

        i.append(cont)

        cont += 1

    else:

        break

Grafica_Rango(Lista_P,V,hyd)

```

Interfaz para selección de caso

```
def Seleccion():
```

```
    import tkinter
```

```
    def Box_Caso_Concreto():
```

```
        global Control_Var
```

```
        Control_Var = 0
```

```
        window.destroy()
```

```
        return Control_Var
```

```
    def Box_Caso_Rango():
```

```
        global Control_Var
```

```
        Control_Var = 1
```

```
        window.destroy()
```

```
        return Control_Var
```

```
    window = tkinter.Tk()
```

```
    window.title('Selección del tipo de cálculo')
```

```
    frame = tkinter.Frame(window)
```

```
    frame.pack()
```

```
    text_frame = tkinter.Label(frame, text = 'Por favor seleccione el tipo de cálculo deseado', font = ('Helvetica',12,'bold'))
```

```
    text_frame.grid(row = 0, column = 0)
```

```
    Selec_frame = tkinter.LabelFrame(frame)
```

```
    Selec_frame.grid(row = 1, column = 0, padx = 20, pady = 10)
```

```
casoconc = tkinter.Button(Selec_frame, text = 'Caso único', command = Box_Caso_Concreto)
```

```
casoconc.grid(row = 0, column = 0, padx = 20, pady = 10)
```

```
casomult = tkinter.Button(Selec_frame, text = 'Rango de velocidades', command =  
Box_Caso_Rango)
```

```
casomult.grid(row = 0, column = 1, padx = 20, pady = 10)
```

```
window.mainloop()
```

```
return Control_Var
```

Interfaz de introducción de datos para “Caso único”

```
def Almac_Datos():
```

```
    import tkinter
```

```
    from tkinter import ttk
```

```
    from tkinter import messagebox as mb
```

```
def CuadroDiag():
```

```
    def button_click():
```

```
        if (Rop_entry.get() == " or Rop_Uni.get() == " or Rot_entry.get() == " or Rot_Uni.get() == " or  
Vel_entry.get() == " or Vel_Uni.get() == " or YieldStr_entry.get() == "
```

```
            or YieldStr_Uni.get() == " or TargetR_entry.get() == " or TargetR_Uni.get() == " or  
delt_entry.get() == " or delt_Uni.get() == " or Long_entry.get() == "
```

```
            or Long_Uni.get() == "):
```

```
                mb.showerror(title = 'Error', message = 'Por favor, introduzca todos los datos')
```

```
                return
```

```
    global rop,rot,v,Yp,Rt,delt,L,Data
```

```
    try:
```

```
        rop = float(Rop_entry.get())
```

```
        Urop = Rop_Uni.get()
```

```
    except:
```

```
mb.showerror(title = 'Error', message = 'Introduzca solo números. Si se desea introducir  
decimales, hacerlo mediante el punto')
```

```
if Urop == 'g/cm^3':
```

```
    rop = rop*10**3
```

```
try:
```

```
    rot = float(Rot_entry.get())
```

```
    Urot = Rot_Uni.get()
```

```
except:
```

```
    mb.showerror(title = 'Error', message = 'Introduzca solo números. Si se desea introducir  
decimales, hacerlo mediante el punto')
```

```
if Urot == 'g/cm^3':
```

```
    rot = rot*10**3
```

```
try:
```

```
    v = float(Vel_entry.get())
```

```
    Uv = Vel_Uni.get()
```

```
except:
```

```
    mb.showerror(title = 'Error', message = 'Introduzca solo números. Si se desea introducir  
decimales, hacerlo mediante el punto')
```

```
if Uv == 'km/s':
```

```
    v = v*10**3
```

```
try:
```

```

    Yp = float(YieldStr_entry.get())

    UYp = YieldStr_Uni.get()

except:

    mb.showerror(title = 'Error', message = 'Introduzca solo números. Si se desea introducir
    decimales, hacerlo mediante el punto')

if UYp == 'GPa':

    Yp = Yp*10**9

else:

    Yp = Yp*10**6

try:

    Rt = float(TargetR_entry.get())

    URt = TargetR_Uni.get()

except:

    mb.showerror(title = 'Error', message = 'Introduzca solo números. Si se desea introducir
    decimales, hacerlo mediante el punto')

if URt == 'GPa':

    Rt = Rt*10**9

else:

    Rt = Rt*10**6

try:

    delt = float(delt_entry.get())

    Udelt = delt_Uni.get()

except:

```

```
mb.showerror(title = 'Error', message = 'Introduzca solo números. Si se desea introducir  
decimales, hacerlo mediante el punto')
```

```
if Udelt == 'micro s':
```

```
    delt = delt*10**(-6)
```

```
try:
```

```
    L = float(Long_entry.get())
```

```
    UL = Long_Uni.get()
```

```
except:
```

```
    mb.showerror(title = 'Error', message = 'Introduzca solo números. Si se desea introducir  
decimales, hacerlo mediante el punto')
```

```
if UL == 'cm':
```

```
    L = L*10**(-2)
```

```
elif UL == 'mm':
```

```
    L = L*10**(-3)
```

```
Data = (rop,rot,v,Yp,Rt,delt,L)
```

```
window.destroy()
```

```
window = tkinter.Tk()
```

```
window.title('Entrada de parámetros')
```

```

frame = tkinter.Frame(window)

frame.pack()

#-----Datos del proyectil-----#

Project_data_frame = tkinter.LabelFrame(frame,text = 'Datos del proyectil')

Project_data_frame.grid(row = 0, column = 0, padx = 20, pady = 10)

#Densidad

Rop_label = tkinter.Label(Project_data_frame, text = 'Densidad')

Rop_label.grid(row = 0, column = 0)

Rop_entry = tkinter.Entry(Project_data_frame)

Rop_entry.grid(row = 0, column = 1)

Rop_Uni = ttk.Combobox(Project_data_frame, values = ['kg/m^3', 'g/cm^3'])

Rop_Uni.grid(row = 0, column = 2)

#Esfuerzo de fluencia

YieldStr_label = tkinter.Label(Project_data_frame, text = 'Esfuerzo de fluencia')

YieldStr_label.grid(row = 1, column = 0)

YieldStr_entry = tkinter.Entry(Project_data_frame)

```

```
YieldStr_entry.grid(row = 1, column = 1)
```

```
YieldStr_Uni = ttk.Combobox(Project_data_frame, values = ['GPa', 'Mpa'])
```

```
YieldStr_Uni.grid(row = 1, column = 2)
```

```
#Velocidad inicial
```

```
Vel_label = tkinter.Label(Project_data_frame, text = 'Velocidad de impacto')
```

```
Vel_label.grid(row = 2, column = 0)
```

```
Vel_entry = tkinter.Entry(Project_data_frame)
```

```
Vel_entry.grid(row = 2, column = 1)
```

```
Vel_Uni = ttk.Combobox(Project_data_frame, values = ['km/s', 'm/s'])
```

```
Vel_Uni.grid(row = 2, column = 2)
```

```
#Longitud
```

```
Long_label = tkinter.Label(Project_data_frame, text = 'Longitud')
```

```
Long_label.grid(row = 3, column = 0)
```

```
Long_entry = tkinter.Entry(Project_data_frame)
```

```
Long_entry.grid(row = 3, column = 1)
```

```
Long_Uni = ttk.Combobox(Project_data_frame, values = ['m', 'cm', 'mm'])
```

```
Long_Uni.grid(row = 3, column = 2)
```

```
#####ORDEN#####
```

```
for widget in Project_data_frame.winfo_children():
```

```
    widget.grid_configure(padx = 10, pady = 5)
```

```
#-----Datos del blanco-----#
```

```
Target_data_frame = tkinter.LabelFrame(frame, text = 'Datos del blanco')
```

```
Target_data_frame.grid(row = 1, column = 0, padx = 20, pady = 10)
```

```
#Densidad
```

```
Rot_label = tkinter.Label(Target_data_frame, text = 'Densidad')
```

```
Rot_label.grid(row = 0, column = 0)
```

```
Rot_entry = tkinter.Entry(Target_data_frame)
```

```
Rot_entry.grid(row = 0, column = 1)
```

```
Rot_Uni = ttk.Combobox(Target_data_frame, values = ['kg/m^3', 'g/cm^3'])
```

```
Rot_Uni.grid(row = 0, column = 2)
```

```
#Resistencia
```

```
TargetR_label = tkinter.Label(Target_data_frame, text = 'Esfuerzo de fluencia')
```

```
TargetR_label.grid(row = 1, column = 0)
```

```

TargetR_entry = tkinter.Entry(Target_data_frame)

TargetR_entry.grid(row = 1, column = 1)

TargetR_Uni = ttk.Combobox(Target_data_frame, values = ['GPa', 'Mpa'])

TargetR_Uni.grid(row = 1, column = 2)

#####ORDEN#####

for widget in Target_data_frame.winfo_children():

    widget.grid_configure(padx = 10, pady = 5)

#-----Parámetros de simulación-----#

Sim_data_frame = tkinter.LabelFrame(frame, text = 'Parámetros de simulación')

Sim_data_frame.grid(row = 2, column = 0, padx = 20, pady = 10)

#Delta de tiempo

delt_label = tkinter.Label(Sim_data_frame, text = 'Delta de t')

delt_label.grid(row = 0, column = 0)

delt_entry = tkinter.Entry(Sim_data_frame)

delt_entry.grid(row = 0, column = 1)

delt_Uni = ttk.Combobox(Sim_data_frame, values = ['s', 'micro s'])

delt_Uni.grid(row = 0, column = 2)

```

```
#####ORDEN#####  
  
for widget in Sim_data_frame.winfo_children():  
    widget.grid_configure(padx = 10, pady = 5)  
  
#Finalizar  
  
button = tkinter.Button(frame, text = 'Iniciar Cálculo', command = button_click)  
button.grid(row = 4, column = 0, padx = 20, pady = 10)  
  
window.mainloop()  
  
return Data  
CuadroDiag()  
return Data
```

Módulo de cálculo para “Caso único”

```
def Caso_Concreto(Entradas):
```

```
    import numpy as np
```

```
    from tkinter import messagebox as mb
```

```
    #INICIALIZACIÓN DE VARIABLES
```

```
    Error_raiz = 0
```

```
    rop = Entradas[0]
```

```
    rot = Entradas[1]
```

```
    v = [Entradas[2]] #Velocidad inicial
```

```
    Yp = Entradas[3]
```

```
    Rt = Entradas[4]
```

```
    delt = Entradas[5]
```

```
    L = [Entradas[6]] #Longitud inicial
```

```
    tension = []
```

```
    t = [0]
```

```
    P = 0
```

```
    i = 0 #Nº de iteraciones
```

```
    x = [0]
```

```
    PosCola = []
```

```
    nose = []
```

```
    P = 0
```

```
    #FUNCIÓN PARA CÁLCULO DE VELOCIDAD DE PENETRACIÓN (NARIZ)
```

```
    def penvel (rop,rot,v,Yp,Rt,i):
```

```

if (rot*rop*v[i]**2 + 2*(rop-rot)*(Rt-Yp)) < 0:

    Error_raiz = 1

    return Error_raiz

u = (rop*v[i]**2 - 2*(Rt-Yp))/(rop*v[i] + np.sqrt(rot*rop*v[i]**2 + 2*(rop-rot)*(Rt-Yp)))

return u

tensionInic = ((1/2)*rot*(penvel(rop,rot,v,Yp,Rt,i))**2 + Rt)/10**9
tension.append(tensionInic)
noseinic = penvel(rop,rot,v,Yp,Rt,i)
nose.append(noseinic)
PosCola.append(-L[i])

#----- BLOQUE Rt > Yp -----#
if Rt > Yp:

    Vc = np.sqrt((2*(Rt-Yp))/rop) #-----> Cálculo de velocidad crítica

    if Vc > v[0]:

        mb.showinfo(title = 'Penetración finalizada', message = 'No ha ocurrido penetración')

return t,nose,v,P,L,x,tension

```

```
while v[i] > Vc: # Si no se cumpliere, no habría penetración
```

```
    if L[i] < 0:
```

```
        return t,nose,v,P,L,x,tension
```

```
    if Error_raiz == 1:
```

```
        return t,nose,v,P,L,x,tension
```

```
u = penvel(rop,rot,v,Yp,Rt,i)
```

```
tension.append((((1/2)*rot*u**2 + Rt)/10**9)
```

```
nose.append(u)
```

```
deltv = -(Yp/(rop*L[i]))*delt
```

```
v.append(v[i] + delv)
```

```
dell = -(v[i]-u)*delt
```

```
L.append(L[i] + dell)
```

```
x.append(u*delt + x[i])
```

```
PosCola.append(x[i] - L[i])
```

```
t.append(t[i]+delt)
```

$P = P + u \cdot \text{delt}$

$i += 1$

while $v[i] \geq 0$:

$u = 0$

$\text{tension.append}(\frac{((1/2) \cdot \text{rot} \cdot u^2 + R_t)}{10^9})$

$\text{nose.append}(u)$

$\text{delv} = -(\text{Yp}/(\text{rop} \cdot L[i])) \cdot \text{delt}$

$\text{v.append}(v[i] + \text{delv})$

$\text{delL} = -(v[i] - u) \cdot \text{delt}$

$\text{L.append}(L[i] + \text{delL})$

$\text{x.append}(u \cdot \text{delt} + x[i])$

$\text{PosCola.append}(x[i] - L[i])$

```

t.append(t[i]+delt)

P = P + u*delt

i += 1

mb.showinfo(title = 'Penetración finalizada', message = 'Proyectil completamente erosionado')

return t,nose,v,P,L,x,tension,PosCola

#----- BLOQUE Rt < Yp -----#

if Rt < Yp:

    Vr = np.sqrt((2*(Yp-Rt))/rot)

    #Módulo de penetración con erosión

    while v[i] > Vr:

        if v[i] <= 0:

            mb.showinfo(title = 'Penetración finalizada', message = 'Proyectil detenido')

            break

        if Error_raiz == 1:

            break

    u = penvel(rop,rot,v,Yp,Rt,i)

```

```
nose.append(u)
```

```
delv = -(Yp/(rop*L[i]))*delt
```

```
tension.append(((1/2)*rot*v[i]**2 + Rt)/10**9) #Se deja en GPa
```

```
v.append(v[i] + delv)
```

```
delL = -(v[i]-u)*delt
```

```
L.append(L[i] + delL)
```

```
x.append(u*delt + x[i])
```

```
PosCola.append(x[i] - L[i])
```

```
t.append(t[i]+delt)
```

```
P = P + u*delt
```

```
i += int(1)
```

```
#Módulo de penetración rígida
```

```
Lf = L[i]
```

```
while v[i] > 0:
```

```
    delvRig = -(Yp/(rop*L[i]))*delt
```

```
    tension.append(((1/2)*rot*v[i]**2 + Rt)/10**9)
```

```
    nose.append(v[i])
```

```
    v.append(v[i] + delvRig)
```

```
    t.append(t[i]+delt)
```

```
    L.append(Lf)
```

```
    x.append(v[i]*delt + x[i])
```

```
    PosCola.append(x[i] - L[i])
```

```
    P = P + u*delt
```

```
    i += 1
```

```
mb.showinfo(title = 'Penetración finalizada', message = 'Proyectil detenido')
```

```
return t,nose,v,P,L,x,tension,PosCola
```

```
#----- BLOQUE Rt = Yp -----#
```

```
if Rt == Yp:
```

```
    while v[i] > 0 and L[i] > 0:
```

```
        u = penvel(rop,rot,v,Yp,Rt,i)
```

```
        tension.append((((1/2)*rot*u**2 + Rt)/10**9)
```

```
        nose.append(u)
```

```
        delv = -(Yp/(rop*L[i]))*delt
```

```
        v.append(v[i] + delv)
```

```
        delL = -(v[i]-u)*delt
```

```
        L.append(L[i] + delL)
```

```
        x.append(u*delt + x[i])
```

```
        PosCola.append(x[i] - L[i])
```

```
        t.append(t[i]+delt)
```

```
        P = P + u*delt
```

```

    i += 1

    if L[i] < 0:
        mb.showinfo(title = 'Penetración finalizada', message = 'Proyectil completamente erosionado')

    if v[i] < 0:
        mb.showinfo(title = 'Penetración finalizada', message = 'Proyectil detenido')

    return t,nose,v,P,L,x,tension,PosCola

```

Generador de gráficas para “Caso único”

```

def Graf_Concreto(Resultados):

    from tkinter import messagebox as mb

    import matplotlib.pyplot as plt

    Tiempo = [valor*10**6 for valor in Resultados[0]] #En microsegundos

    VelPen = Resultados[1]

    VelPro = Resultados[2]

    Penetracion = Resultados[3]*1000 #En mm

    Long= [valor*1000 for valor in Resultados[4]] #En mm

    Pos = [valor*1000 for valor in Resultados[5]] #En mm

    Tens = [valor*10**9 for valor in Resultados[6]] #En Gpa

    PosCola = [valor*1000 for valor in Resultados[7]] #En mm

    plt.figure(1)

```

```
plt.grid(True)

plt.title('Velocidades frente a tiempo')

plt.plot(Tiempo, VelPen, 'r', label = 'Velocidad de penetración')

plt.plot(Tiempo, VelPro, 'b', label = 'Velocidad de cola')

plt.xlabel('Tiempo [micro segundos]')

plt.ylabel('Velocidad [m/s]')

plt.legend()
```

```
plt.figure(3)

plt.grid(True)

plt.title('Posiciones de cola y nariz frente a tiempo')

plt.plot(Tiempo, Pos, 'r', label = 'Posición de punta de proyectil')

plt.plot(Tiempo, PosCola, 'b', label = 'Posición de cola de proyectil')

plt.xlabel('Tiempo [micro segundos]')

plt.ylabel('Penetración [mm]')

plt.legend()
```

```
plt.figure(4)

plt.grid(True)

plt.title('Longitud frente a tiempo')

plt.plot(Tiempo, Long)

plt.xlabel('Tiempo [micro segundos]')

plt.ylabel('Longitud del proyectil [mm]')
```

```
plt.figure(5)
```

```
plt.grid(True)

plt.title('Tensión frente a tiempo')

plt.plot(Tiempo,Tens)

plt.xlabel('Tiempo [micro segundos]')

plt.ylabel('Tensión [GPa]')
```

```
plt.figure(6)

plt.grid(True)

plt.title('Tensión frente a longitud penetrada')

plt.plot(Pos,Tens)

plt.xlabel('Penetración [mm]')

plt.ylabel('Tensión [GPa]')
```

```
Penetracion = Resultados[3]
```

```
mb.showinfo(title = 'Resultado', message = 'Se ha penetrado el blanco ' +
str(round(Penetracion*1000,3)) + ' mm')
```

```
plt.show()
```

Interfaz de introducción de datos para “Rango de velocidades”

```
def Almac_Datos_Mod():
```

```
    import tkinter
```

```
    from tkinter import ttk
```

```
    from tkinter import messagebox as mb
```

```
def CuadroDiag():
```

```
    def button_click():
```

```
        if (Rop_entry.get() == " or Rop_Uni.get() == " or Rot_entry.get() == " or Rot_Uni.get() == " or  
YieldStr_entry.get() == " or YieldStr_Uni.get() == "
```

```
            or TargetR_entry.get() == " or TargetR_Uni.get() == " or delt_entry.get() == " or delt_Uni.get()  
            == " or Long_entry.get() == " or Long_Uni.get() == "):
```

```
                mb.showerror(title = 'Error', message = 'Por favor, introduzca correctamente todos los datos')
```

```
                return
```

```
global rop,rot,Yp,Rt,delt,L,Data,temporal
```

```
rop = float(Rop_entry.get())
```

```

Urop = Rop_Uni.get()

if Urop == 'g/cm^3':
    rop = rop*10**3

rot = float(Rot_entry.get())
Urot = Rot_Uni.get()

if Urot == 'g/cm^3':
    rot = rot*10**3

Yp = float(YieldStr_entry.get())
UYp = YieldStr_Uni.get()

if UYp == 'GPa':
    Yp = Yp*10**9
else:
    Yp = Yp*10**6

Rt = float(TargetR_entry.get())
URt = TargetR_Uni.get()

if URt == 'GPa':
    Rt = Rt*10**9
else:
    Rt = Rt*10**6

```

```
delt = float(delt_entry.get())

Udelt = delt_Uni.get()

if Udelt == 'micro s':
    delt = delt*10**(-6)

L = float(Long_entry.get())
UL = Long_Uni.get()

if UL == 'cm':
    L = L*10**(-2)
elif UL == 'mm':
    L = L*10**(-3)

temporal = 0

Data = [rop,rot,temporal,Yp,Rt,delt,L]

window.destroy()

window = tkinter.Tk()
window.title('Entrada de parámetros')
```

```

frame = tkinter.Frame(window)

frame.pack()

#-----Datos del proyectil-----#

Project_data_frame = tkinter.LabelFrame(frame, text = 'Datos del proyectil')

Project_data_frame.grid(row = 0, column = 0, padx = 20, pady = 10)

#Densidad

Rop_label = tkinter.Label(Project_data_frame, text = 'Densidad')

Rop_label.grid(row = 0, column = 0)

Rop_entry = tkinter.Entry(Project_data_frame)

Rop_entry.grid(row = 0, column = 1)

Rop_Uni = ttk.Combobox(Project_data_frame, values = ['kg/m^3', 'g/cm^3'])

Rop_Uni.grid(row = 0, column = 2)

#Esfuerzo de fluencia

YieldStr_label = tkinter.Label(Project_data_frame, text = 'Esfuerzo de fluencia')

YieldStr_label.grid(row = 1, column = 0)

YieldStr_entry = tkinter.Entry(Project_data_frame)

```

```
YieldStr_entry.grid(row = 1, column = 1)
```

```
YieldStr_Uni = ttk.Combobox(Project_data_frame, values = ['GPa', 'Mpa'])
```

```
YieldStr_Uni.grid(row = 1, column = 2)
```

```
#Longitud del proyectil
```

```
Long_label = tkinter.Label(Project_data_frame, text = 'Longitud')
```

```
Long_label.grid(row = 2, column = 0)
```

```
Long_entry = tkinter.Entry(Project_data_frame)
```

```
Long_entry.grid(row = 2, column = 1)
```

```
Long_Uni = ttk.Combobox(Project_data_frame, values = ['m', 'cm', 'mm'])
```

```
Long_Uni.grid(row = 2, column = 2)
```

```
#####ORDEN#####
```

```
for widget in Project_data_frame.winfo_children():
```

```
    widget.grid_configure(padx = 10, pady = 5)
```

```
#-----Datos del blanco-----#
```

```
Target_data_frame = tkinter.LabelFrame(frame, text = 'Datos del blanco')
```

```
Target_data_frame.grid(row = 1, column = 0, padx = 20, pady = 10)
```

```
#Densidad
```

```
Rot_label = tkinter.Label(Target_data_frame, text = 'Densidad')
```

```
Rot_label.grid(row = 0, column = 0)
```

```
Rot_entry = tkinter.Entry(Target_data_frame)
```

```
Rot_entry.grid(row = 0, column = 1)
```

```
Rot_Uni = ttk.Combobox(Target_data_frame, values = ['kg/m^3', 'g/cm^3'])
```

```
Rot_Uni.grid(row = 0, column = 2)
```

```
#Resistencia
```

```
TargetR_label = tkinter.Label(Target_data_frame, text = 'Esfuerzo de fluencia')
```

```
TargetR_label.grid(row = 1, column = 0)
```

```
TargetR_entry = tkinter.Entry(Target_data_frame)
```

```
TargetR_entry.grid(row = 1, column = 1)
```

```
TargetR_Uni = ttk.Combobox(Target_data_frame, values = ['GPa', 'Mpa'])
```

```
TargetR_Uni.grid(row = 1, column = 2)
```

```
#####ORDEN#####
```

```
for widget in Target_data_frame.winfo_children():
```

```
    widget.grid_configure(padx = 10, pady = 5)
```

```

#-----Parámetros de simulación-----#

Sim_data_frame = tkinter.LabelFrame(frame,text = 'Parámetros de simulación')

Sim_data_frame.grid(row = 2, column = 0, padx = 20, pady = 10)

#Delta de tiempo

delt_label = tkinter.Label(Sim_data_frame, text = 'Delta de t')

delt_label.grid(row = 0, column = 0)

delt_entry = tkinter.Entry(Sim_data_frame)

delt_entry.grid(row = 0, column = 1)

delt_Uni = ttk.Combobox(Sim_data_frame, values = ['s', 'micro s'])

delt_Uni.grid(row = 0, column = 2)

#####ORDEN#####

for widget in Sim_data_frame.winfo_children():

    widget.grid_configure(padx = 10, pady = 5)

#Finalizar

button = tkinter.Button(frame, text = 'Siguiente', command = button_click)

button.grid(row = 4, column = 0, padx = 20, pady = 10)

```

```
window.mainloop()
```

```
return Data
```

```
CuadroDiag()
```

```
return Data
```

Interfaz de introducción de rango de velocidades para “Rango de velocidades”

```
def Selecc_Rango():
```

```
import tkinter
```

```
from tkinter import messagebox as mb
```

```
def recop():
```

```
global vmin,vmax,incrementos_v
```

```
try:
```

```
vmin = float(velmin_entry.get())
```

```
vmax = float(velmax_entry.get())
```

```
incrementos_v = int(incrementos_v_entry.get())
```

```
except:
```

```
    mb.showerror(title = 'Error', message = 'Por favor, introduzca correctamente todos los datos')
```

```
    return
```

```
    window.destroy()
```

```
window = tkinter.Tk()
```

```
window.title('Rango de velocidades')
```

```
frame = tkinter.Frame(window)
```

```
frame.pack()
```

```
    text_frame = tkinter.Label(frame, text = 'Introduzca el rango de velocidades', font =  
(('Helvetica',12,'bold'))
```

```
    text_frame.grid(row = 0, column = 0)
```

```
range_frame = tkinter.LabelFrame(frame)
```

```
range_frame.grid(row = 1, column = 0, padx = 20, pady = 10)
```

```
velmin_label = tkinter.Label(range_frame, text = 'Velocidad inferior [m/s]')
```

```
velmin_label.grid(row = 0, column = 0, padx = 40, pady = 10)
```

```
velmin_entry = tkinter.Entry(range_frame)
velmin_entry.grid(row = 1, column = 0, padx = 40, pady = 10)

velmax_label = tkinter.Label(range_frame, text = 'Velocidad superior [m/s]')
velmax_label.grid(row = 0, column = 1, padx = 40, pady = 10)

velmax_entry = tkinter.Entry(range_frame)
velmax_entry.grid(row = 1, column = 1, padx = 40, pady = 10)

div_frame = tkinter.LabelFrame(frame)
div_frame.grid(row = 2, column = 0, padx = 20, pady = 10)

incrementos_v_label = tkinter.Label(div_frame, text = 'Incrementos de velocidad [m/s]')
incrementos_v_label.grid(row = 2, column = 0, padx = 20, pady = 10)

incrementos_v_entry = tkinter.Entry(div_frame)
incrementos_v_entry.grid(row = 3, column = 0, padx = 40, pady = 10)

#Finalizar

button = tkinter.Button(frame, text = 'Iniciar Cálculo', command = recop)
button.grid(row = 3, column = 0, padx = 20, pady = 10)
```

```
window.mainloop()
```

```
Rangos = (vmin, vmax, incrementos_v)
```

```
return Rangos
```

Módulo de cálculo para “Rango de velocidades”

```
def Casos_Multiples(Entradas_rango):
```

```
    import numpy as np
```

```
    #INICIALIZACIÓN DE VARIABLES
```

```
    Error_raiz = 0
```

```
    rop = Entradas_rango[0]
```

```
    rot = Entradas_rango[1]
```

```
    v = Entradas_rango[2] #Velocidad inicial
```

```
    Yp = Entradas_rango[3]
```

```
    Rt = Entradas_rango[4]
```

```
    delt = Entradas_rango[5]
```

```
    L = Entradas_rango[6] #Longitud inicial
```

```
    t = 0
```

```
    P = 0
```

```
    nose = v
```

```
    P = 0
```

```

#FUNCIÓN PARA CÁLCULO DE VELOCIDAD DE PENETRACIÓN (NARIZ)
def penvel (rop,rot,v,Yp,Rt):

    if (rot*rop*v**2 + 2*(rop-rot)*(Rt-Yp)) < 0:

        Error_raiz = 1

        return Error_raiz

    u = (rop*v**2 - 2*(Rt-Yp))/(rop*v + np.sqrt(rot*rop*v**2 + 2*(rop-rot)*(Rt-Yp)))

    return u

#----- BLOQUE Rt > Yp -----#
if Rt > Yp:

    Vc = np.sqrt((2*(Rt-Yp))/rop) #-----> Cálculo de velocidad crítica

    if Vc > v:

        return t,nose,v,P,L

    while v > Vc: # Si no se cumpliera, no habría penetración

        if L < 0:

            return t,nose,v,P,L

```

```

if Error_raiz == 1:
    break

u = penvel(rop,rot,v,Yp,Rt)

delv = -(Yp/(rop*L))*delt

v = v + delv

delL = -(v-u)*delt

L = L + delL

t = t+delt

P = P + u*delt

return t,nose,v,P,L

#----- BLOQUE Rt < Yp -----#
if Rt < Yp:

    Vr = np.sqrt((2*(Yp-Rt))/rot)

```

```

#Módulo de penetración con erosión

while v > Vr:

    if L < 0:

        return t,nose,v,P,L

    if Error_raiz == 1:

        break

    u = penvel(rop,rot,v,Yp,Rt)

    delv = -(Yp/(rop*L))*delt

    v = v + delv

    delL = -(v-u)*delt

    L = L + delL

    t = t + delt

    P = P + u*delt

#Módulo de penetración rígida

Lf = L

```

```

while v > 0:

    if Error_raiz == 1:
        break

    u = penvel(rop,rot,v,Yp,Rt)

    delvRig = -(Yp/(rop*Lf))*delt

    v = v + delvRig

    t = t+delt

    P = P + u*delt

    return t,nose,v,P,L

#----- BLOQUE Rt = Yp -----#
if Rt == Yp:

    while v > 0 and L > 0:

        u = penvel(rop,rot,v,Yp,Rt)

        delv = -(Yp/(rop*L))*delt

```

```
v = v + delv
```

```
dell = -(v-u)*delt
```

```
L = L + dell
```

```
t = t + delt
```

```
P = P + u*delt
```

```
return t,nose,v,P,L
```

Generador de gráficas para “Rango de velocidades”

```
def Grafica_Rango(Lista_P,V,hyd):
```

```
    Lista_P_cm = [valor*1000 for valor in Lista_P] #Dejamos la penetración en mm
```

```
    import matplotlib.pyplot as plt
```

```
    plt.figure(1)
```

```
    plt.grid(True)
```

```
    plt.title('Penetración frente a velocidad de impacto')
```

```
    plt.plot(V,Lista_P_cm,'b', label = 'Penetración')
```

```
    plt.plot(V,hyd,'r', label = 'Límite hidrodinámico de penetración')
```

```
    plt.xlabel('Velocidad de impacto [m/s]')
```

```
    plt.ylabel('Penetración [mm]')
```

```
    plt.legend()
```

```
    plt.show()
```