**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**Факультет прикладной математики-процессов управления**

**Программа бакалавриата**

**“Большие данные и распределенная цифровая платформа”**

**ОТЧЕТ**

**по семестровой работе**

**по дисциплине «Алгоритмы и структуры данных»**

**Студенты гр. 23Б15-пу**

**Кубякин Н. А.**

**Черевко М. Е.**

**Бек В. А.**

**Преподаватели**

**Дик А.Г.**

**Щеголева Н. Л.**

**Санкт-Петербург**

**2025 г.**

Оглавление

1. [Формулировка задачи 3](#_Toc198476650)
2. [Цель работы 4](#_Toc198476651)
3. [Теоретическая часть 5](#_Toc198476652)
4. [Блок схема программы 10](#_Toc198476653)
5. [Описание программы 15](#_Toc198476654)
6. [Рекомендации пользователя 21](#_Toc198476655)
7. [Рекомендации программиста 23](#_Toc198476656)
8. [Контрольный пример 24](#_Toc198476657)
9. [Анализ работы 32](#_Toc198476658)
10. [Вывод 35](#_Toc198476659)
11. [Источники 36](#_Toc198476660)

# Формулировка задачи

Современные гиперзвуковые системы имеют от 6 до 12 разделяющихся боеголовок. Оценить, как влияет увеличение их количества на расширение области поражения и на глубину проникновения в поражаемый объект.

# Цель работы

**Цель работы** —разобраться в принципах функционирования современных гиперзвуковых систем с разделяющимися боевыми блоками, определить влияние количества боеголовок на область поражения — в частности, на расширение области поражения и глубину проникновения в цель, найти оптимальное количество боеголовок для данных целей. Для этого необходимо:

1. Изучить физические основы, связанные с гиперзвуковыми ракетами и их боеголовками
2. Подобрать необходимые формулы для расчёта диаметра кратера и его глубины в зависимости от характеристик боеголовок
3. Провести аналитическое сравнение при различных параметрах: количестве боеголовок, плотности материала боеголовок, плотности материала мишени, скорости боеголовок и т.д.
4. Сформулировать выводы о зависимости эффективности поражения от числа боеголовок и других характеристик

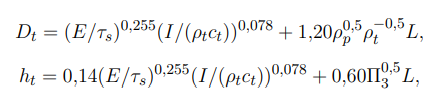
# Теоретическая часть

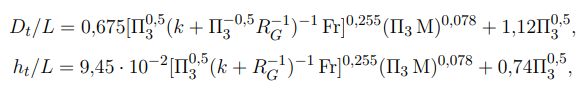
Для данной работы было решено рассматривать две различные формы тела: сферическую, как наиболее часто используемую для теоретических расчетов подобного типа и цилиндрическую, наиболее напоминающую настоящую форму боеголовки.

Важно понимать, что процесс гиперзвукового удара разделён на две фазы: **гидродинамическую** (когда и фазу **твёрдого тела** (когда ) где . При переходе из гидродинамической в твёрдую фазу снаряд **теряет массу**. Рассмотрим формулы для каждой фазы, а также дополнительные выражения.

**Формулы для сферического тела, гидродинамическая фаза:**

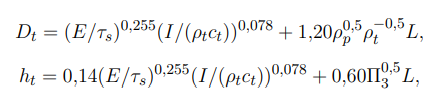
Для данной формы тела лучше всего подходила статья, описывающая достаточно схожую с нашей задачей работу, а именно: [**ВЛИЯНИЕ ИМПУЛЬСА МЕТЕОРИТА НА РАЗМЕРЫ УДАРНОГО КРАТЕРА Н. И. Шишкин**](https://www.sibran.ru/upload/iblock/8a5/8a534ba0ff27ddedeb8494579c60e436.pdf). Из данной статьи и ещё нескольких источников были подобраны формулы для расчёта глубины проникновения боеголовки в мишень, диаметра и объёма получившегося от такого столкновения кратера:



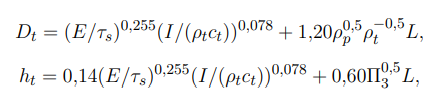


Из представленного обилия формул, на основе нескольких десятков экспериментов, было решено выбрать две представленные ниже, так как они давали наиболее приемлемый и точный результат.

**Итоговая формула расчета диаметра кратера:**



**Итоговая формула расчета глубины кратера:**



Где E = mv2/2 — кинетическая энергия ударника; m = 0,52ρpL3 — масса ударника, I = mv — импульс ударника; ct — скорость распространения объемных волн в мишени, τs — прочность материала мишени (коры планеты) на глубине проникания h, П3 — отношение плотности ударника к плотности преграды.

На (Рис. 1) формула, выбранная в итоге, имеет номер 2.2.

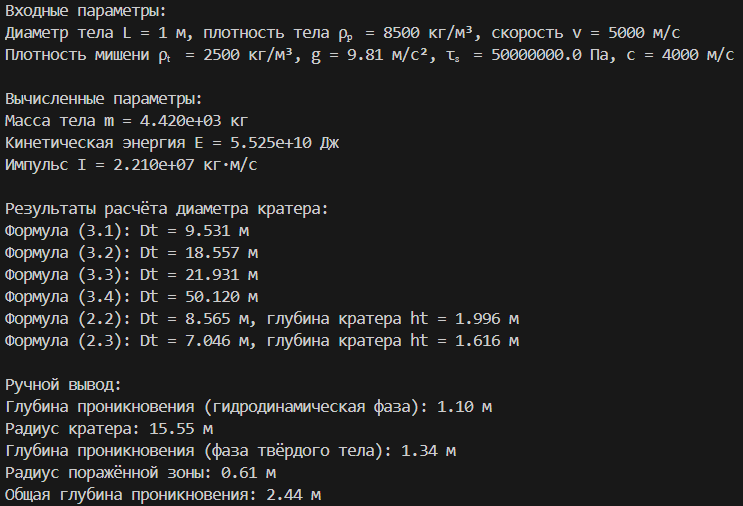


Рис. 1 Сравнение формул

Объёма получившегося кратера можно оценивать по формуле:



**Формулы для цилиндрического тела, гидродинамическая фаза:**

Для цилиндрического тела так же были использованы формулы из статьи для сферической формы тела, но с учётом специфики формы снаряда.

**Итоговая формула расчета глубины кратера:**

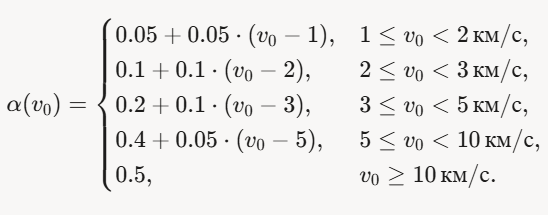
**Итоговая формула расчета диаметра кратера:**

* — кинетическая энергия (Дж),
* — импульс (кг·м/с),
* — масса снаряда (кг)
* — объём снаряда ()

* — площадь сечения снаряда ()
* — скорость
* — плотность снаряда (кг/м³)
* — плотность снаряда и мишени (кг/м³),
* — длина снаряда (м),
* — диаметр снаряда (м),
* — скорость звука в мишени (м/с),
* — эффективная прочность мишени (Па),
* — статическая прочность мишени (Па).

**Сферическая и цилиндрическая формы тела, твёрдая фаза:**

**Потеря массы снаряда:**

 При гиперзвуковом ударе часть массы снаряда теряется (т.е. при переходе тела из гидродинамического состояния в твёрдое). Коэффициент потери массы зависит от скорости .

Эффективная масса после потери: где  – начальная масса.

**Глубина проникновения (**ф. Забудского из [Фортификационные расчёты](https://ru.wikipedia.org/wiki/Фортификационные_расчёты)**):**

* + — критическая скорость (2000 м/с)
  + — диаметр снаряда (м),
  + Для железобетона марки свыше 250

**Диаметр поражённой зоны:**

* + — площадь сечения снаряда ()
  + — критическая скорость (2000 м/с)
  + — скорость звука в мишени (м/с),
  + — диаметр снаряда (м),

**Общая глубина проникновения и радиусы поражения:**

Общая глубина складывается из двух фаз:

Диаметры не суммируются, так как описывают разные процессы

* Для радиус кратера — (гидродинамическая фаза).
* Для : зона повреждений —​ (твёрдая фаза).

# Блок схема программы

**Программа запуска:**

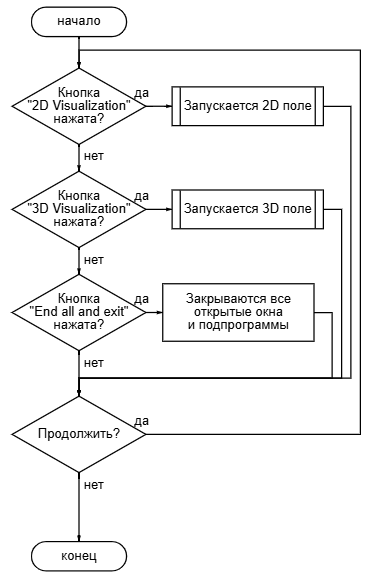


Рис. 3 Блок-схема программы запуска

**3D часть:**

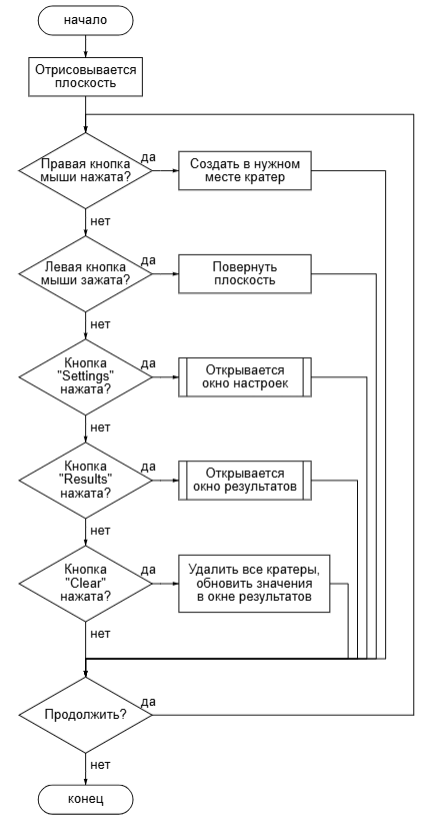


Рис. 3 Блок-схема основной 3D программы

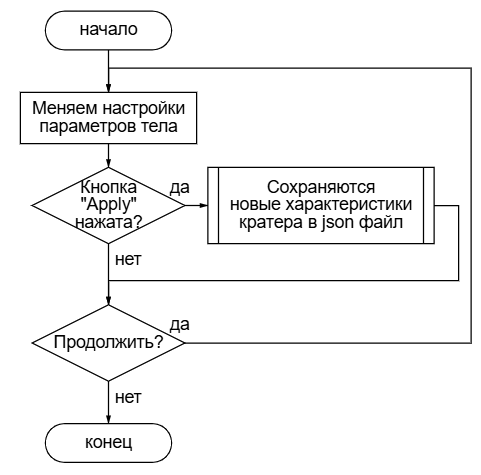


Рис. 4 Блок-схема подпрограммы окна настроек 3D

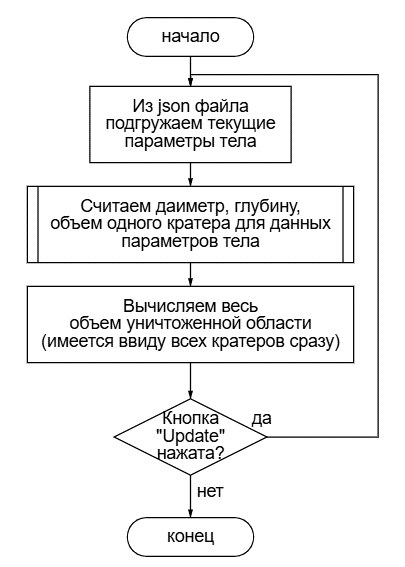


Рис. 5 Блок-схема подпрограммы окна результатов 3D

**2D часть:**

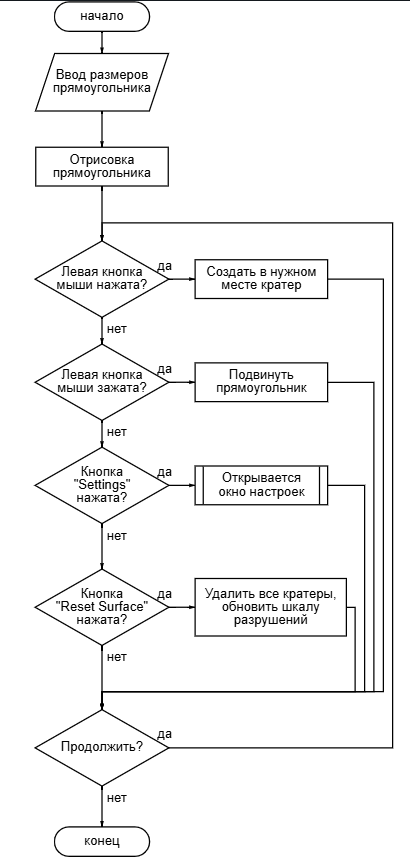
****

Рис. 6 Блок-схема основной 2D программы

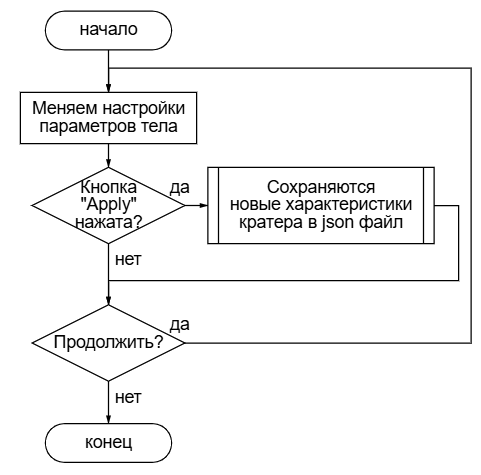


Рис. 7 Блок-схема подпрограммы окна настроек 2D

# Описание программы

Программная реализация создания кратеров от столкновения боеголовки с мишенью написана на языке **Python 3.12.7** с использованием библиотек **tkinter, pygame, numpy, OpenGL**. Программа представляет собой графический интерфейс, позволяющий пользователю на специальных полях (2D и 3D) создавать кратеры исходя из изначально заданных параметров боеголовки. Так же предусмотрена возможность изменения параметров боеголовки прямо в интерфейсе программы.

Таблица 1. main\_window.py

|  |  |  |
| --- | --- | --- |
| Функция | Описание | Возвращаемое значение |
| create\_widgets(self) | Создаёт и размещает виджеты (метки, кнопки) в главном окне приложения. | Нет (None) |
| start\_program(self, program\_type) | Запускает указанную программу (2D или 3D визуализацию) как отдельный процесс. | Нет (None) |
| validate\_path(self, path) | Проверяет существование файла по указанному пути. | Булево значение (True/False) |
| stop\_all(self) | Завершает все запущенные процессы и закрывает приложение. | Нет (None) |
| on\_closing(self) | Обрабатывает событие закрытия окна, завершая все процессы. | Нет (None) |
| check\_processes(self) | Периодически проверяет статус запущенных процессов и обновляет состояние кнопок. | Нет (None) |
| process\_running(process) | Статический метод, проверяет, выполняется ли процесс. | Булево значение (True/False) |
| terminate\_process(process) | Статический метод, завершает указанный процесс, если он активен. | Нет (None) |

Таблица 2. pygame\_main.py

|  |  |  |
| --- | --- | --- |
| Функция | Описание | Возвращаемое значение |
| init\_text() | Инициализирует GLUT для рендеринга текста. | Нет (None) |
| draw\_text(x, y, text) | Отрисовывает текст на экране в указанных координатах (x, y). | Нет (None) |
| draw\_button(x, y, w, h, label, offset\_x=0) | Отрисовывает кнопку с заданными координатами, размерами и меткой. | Нет (None) |
| draw\_plane() | Отрисовывает 3D-плоскость с цветовой кодировкой глубины и сеткой. | Нет (None) |
| get\_mouse\_intersection(mx, my) | Вычисляет точку пересечения луча мыши с плоскостью z=0. | Кортеж (x, y, z) — координаты точки |
| world\_to\_grid(ix, iy) | Преобразует мировые координаты (x, y) в индексы сетки. | Кортеж (gx, gy) — индексы сетки |
| create\_crater(x, y, radius, depth) | Создаёт кратер на плоскости с заданными параметрами, изменяя высоту. | Нет (None) |
| reset\_plane() | Сбрасывает высоты плоскости и счётчик кратеров, активирует режим сброса. | Нет (None) |
| draw\_mouse\_indicator(mouse\_pos) | Отрисовывает индикатор мыши (вертикальную линию) на поверхности. | Нет (None) |
| main() | Основной цикл программы: инициализация, обработка событий, рендеринг. | Нет (None) |

Таблица 3. results\_window.py

|  |  |  |
| --- | --- | --- |
| Функция | Описание | Возвращаемое значение |
| calculate\_crater\_volume(diameter, depth) | Вычисляет объём кратера по формуле π \* diameter² \* depth / 8. | Число (float) — объём в м³ |
| reset\_crater\_count() | Сбрасывает глобальный счётчик кратеров до 0. | Нет (None) |
| refresh\_labels(frame) | Обновляет метки в окне результатов, отображая характеристики кратеров. | Нет (None) |
| create\_results\_window(root) | Создаёт или возвращает окно результатов с характеристиками кратеров. | Объект `tk.Toplevel` — окно результатов |
| show\_results\_window(root) | Отображает окно результатов, создавая новое, если оно не существует. | Нет (None) |
| increment\_crater\_count() | Увеличивает счётчик кратеров на 1. | Нет (None) |

Таблица 4. scene\_state.py

|  |  |  |
| --- | --- | --- |
| Функция | Описание | Возвращаемое значение |
| set\_plane\_data(vertices, spc) | Устанавливает глобальные переменные для вершин плоскости и шага сетки. | Нет (None) |
| calculate\_deformed\_volume() | Вычисляет суммарный объём деформированной области (отрицательные высоты). | Число (float) — объём в м³ |

Таблица 5. settings\_window.py

|  |  |  |
| --- | --- | --- |
| Функция | Описание | Возвращаемое значение |
| create\_settings\_window(root) | Создаёт окно настроек для ввода параметров модели. | Объект `tk.Toplevel` — окно настроек |
| update\_settings\_window(win) | Обновляет существующее окно настроек, если оно существует. | Объект `tk.Toplevel` или `None` |
| show\_settings\_window() | Создаёт новое окно настроек (дублирует `create\_settings\_window`). | Объект `tk.Toplevel` — окно настроек |

Таблица 6. shared\_params.py

|  |  |  |
| --- | --- | --- |
| Функция | Описание | Возвращаемое значение |
| load\_settings() | Загружает настройки из файла `settings.json`, дополняя их значениями по умолчанию. | Нет (None) |
| save\_settings() | Сохраняет текущие настройки в файл `settings.json`. | Нет (None) |

Таблица 7. calculate.py

|  |  |  |
| --- | --- | --- |
| Функция | Описание | Возвращаемое значение |
| crater\_diameter\_and\_depth\_2\_2(E, tau\_s, I, rho\_t, c, rho\_p, L) | Вычисляет диаметр и глубину кратера по заданным параметрам. | Кортеж (Dt, ht) — диаметр и глубина |
| calculate\_penetration(N, L, rho\_p, v0, rho\_t, tau\_0, c, alpha, k\_pr, v\_crit, shape="Цилиндр") | Вычисляет параметры проникновения и кратера для сферы или цилиндра. | Словарь с параметрами (`m`, `E`, `I`, `h\_hydro\_2`, `d\_crater\_2`, `h\_solid`, `r\_solid`, `h\_total`, `A0`) |

Таблица 8. 2d\_module.py

|  |  |  |
| --- | --- | --- |
| Функция | Описание | Возвращаемое значение |
| get\_rectangle\_dimensions() | Отображает диалоговое окно для ввода ширины и высоты прямоугольника, обновляет глобальные переменные. | Нет (None) |
| calculate\_destroyed\_percentage() | Вычисляет процент разрушенной площади прямоугольника на основе текущей и исходной поверхности. | Число (float) — процент разрушения |
| update\_destroyed\_bar() | Обновляет гистограмму, отображающую процент разрушенной площади. | Нет (None) |
| calculate\_crater\_depth(center\_x) | Вычисляет профиль глубины кратера в заданной точке по оси X. | Кортеж (x, depth) — массивы координат и глубин |
| create\_crater\_at(x\_coord) | Создаёт кратер в указанной координате, обновляет поверхность и отображение. | Нет (None) |
| onrelease(event) | Обрабатывает событие отпускания кнопки мыши, создаёт кратер при быстром клике. | Нет (None) |
| onclick(event) | Обрабатывает событие нажатия мыши, фиксирует время клика и обновляет параметры кратера. | Нет (None) |
| calculate\_values() | Получает параметры из shared\_settings и вызывает calculate\_penetration для расчёта характеристик кратера. | Словарь с параметрами кратера (см. calculate\_penetration) |
| update\_globals(values) | Обновляет глобальные переменные (h\_total, h\_hydro, r\_crater, r\_solid) на основе результатов расчёта. | Нет (None) |
| reset\_rectangle() | Сбрасывает поверхность, перерисовывает прямоугольник и обновляет отображение. | Нет (None) |
| zoom(event) | Обрабатывает событие прокрутки колеса мыши для масштабирования области отображения. | Нет (None) |
| pan(event) | Инициирует панорамирование области отображения при нажатии кнопки мыши. | Нет (None) |
| do\_pan(event) | Выполняет панорамирование области отображения при движении мыши. | Нет (None) |
| release\_pan(event) | Завершает панорамирование при отпускании кнопки мыши. | Нет (None) |
| open\_settings() | Открывает окно настроек, предотвращая множественное открытие. | Нет (None) |
| on\_closing() | Закрывает окно приложения, завершает работу Matplotlib и программы. | Нет (None) |

# Рекомендации пользователя

Для корректного выполнения программы следуйте приведённым шагам:

**Запуск основной программы:**

* 1. **Запустите main\_window.py и выберете желаемый вариант поля, на соответствующие кнопки: “** **3D Visualization” или “** **2D Visualization”**
  2. Если хотите завершить процесс, нажмите: “ Complete all and exit”

**3D часть:**

**В появившемся окне представлен функционал:**

* 1. **Нажатием правой кнопки мыши создавать кратеры на представленном поле**
  2. **Нажатием на левую кнопку мыши менять положение поля**
  3. **Прокручиванием колёсика приближать и отдалять поле**
  4. **При нажатии на кнопку “Clear” в левом верхнем углу очистится всё поле и окно результаты**
  5. **При нажатии на кнопку “Settings” в правом верхнем углу появится новое окно, позволяющее пользователю настраивать все параметры боеголовки (скорость, массу, плотность боеголовки, плотность мишени и т.д.), нажатие на кнопку “Apply” изменит настройки на новые**
  6. **При нажатии на кнопку “Results”, расположенную слева от кнопки “Settings” появится новое окно, позволяющее пользователю смотреть текущие характеристики боеголовки, размеры и объем кратера, который получится от одной такой боеголовки и общий объём после поверженной несколькими боеголовками зоны. Рекомендуется иногда нажимать кнопку “Update” что бы не упустить изменения, которые вносили.**

**2D часть:**

Первым делом откроется окно, позволяющее пользователю задать размер желаемого поля (ширину и высоту), после нажатия “OK” пользователя переместит в окно для создания кратеров, в нём представлен функционал:

* 1. **Нажатие на левую кнопку мыши создаёт кратеры на представленном поле**
  2. **При зажатии левой кнопки мыши поле можно перемещать**
  3. **Прокручиванием колёсика приближать и отдалять поле**
  4. Снизу расположена полоса, показывающее в %-ах количество оставшегося поля.
  5. **При нажатии на кнопку “Settings” в левом верхнем углу появится новое окно, позволяющее пользователю настраивать все параметры боеголовки (скорость, массу, плотность боеголовки, плотность мишени и т.д.), нажатие на кнопку “Apply” изменит настройки на новые**
  6. **Под кнопкой “Settings” в левом верхнем углу расположена кнопка “Reset Surface”, позволяющая пользователю вернуть поле к изначальному виду и снова создавать кратеры**

# Рекомендации программиста

Для корректного функционирования программы рекомендуется выполнить следующие действия:

1. **Установите необходимые библиотеки**:
   * Убедитесь, что у вас установлены библиотеки **tkinter, math, ttk** и **tkinter.simpledialog, sklearn, pygame, numpy, OpenGL**. **thinker** идет в стандартной поставке Python, но если она отсутствует, вы можете установить ее через пакетный менеджер вашей операционной системы.
2. **Проверьте версию Python**:
   * Рекомендуется использовать **Python** версии **3.1** или выше, чтобы избежать возможных проблем с совместимостью библиотек и функциональностью **tkinter**.
3. **Проверка функций и интерфейса**:
   * Убедитесь, что все элементы интерфейса (кнопки, текстовые поля и таблицы) работают корректно. Попробуйте вводить различные значения и проверить, что результаты рассчитываются и отображаются правильно.

**Код программы:**

[Hypersonic\_MultiWarhead\_Impact\_Modeling\_And\_Analysis](https://github.com/FraaaM/Hypersonic_MultiWarhead_Impact_Modeling_And_Analysis)

# Контрольный пример

Ввиду упрощенной модели задачи сложно найти похожие работы и эксперименты, особенно с большими размерами снаряда и высокими скоростями. В нескольких статьях описывается похожий процесс проникновения небольшого стального снаряда в алюминиевую цель. Ввиду использования формул для массивных тел и высоких скоростей (что больше соответствует формулировке задачи), при малых снарядах и небольших скоростях используемые формулы могут дать заметную ошибку в оценке глубины проникновения.

Работу программы оценим на основе статьи “[PENETRATION OF 6061-T6511 ALUMINUM TARGETS BY OGIVE-NOSE STEEL PROJECTILES WITH STRIKING VELOCITIES BETWEEN 0.5 AND 3.0 KM/S](http://ciar.org/ttk/mbt/papers/ijie02/armor.x.ijie.vol23.pp723-734.penetration_of_6061_t6511_aluminum_targets_by_ogive_nose_steel_projectiles_with_striking_velocities_between_0_5_and_3_0_km_s.piekutowski_poormon_forrestal_warren.1999.pdf)” авторства ANDREW J. PIEKUTOWSKI, MICHAEL J. FORRESTAL, KEVIN L. POORMON, THOMAS L. WARREN. В статье содержатся эксперименты по проникновению стального цилиндра в алюминиевую поверхность.

Рассмотрим работу программы с момента запуска до получения результатов:

**1.** **Запуск программы**: запустить main\_window.py и нажать “2D Visualization” (Рис. 8).

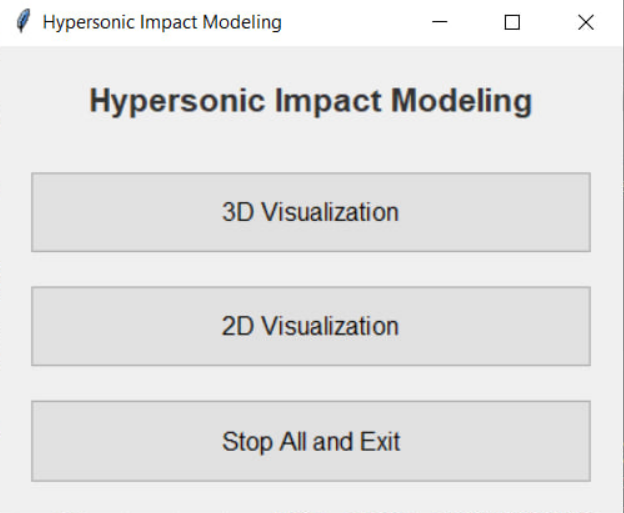


Рис. 8 Главное меню программы

**2.** **Выбор размеров цели**: описанные в статье эксперименты используют цель примерно 60 см в глубину, ширина цели в данном эксперименте роли не играет (Рис. 9 / Рис. 10).

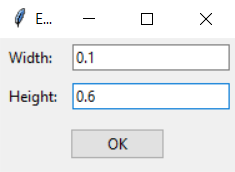


Рис. 9 Выбор размеров цели

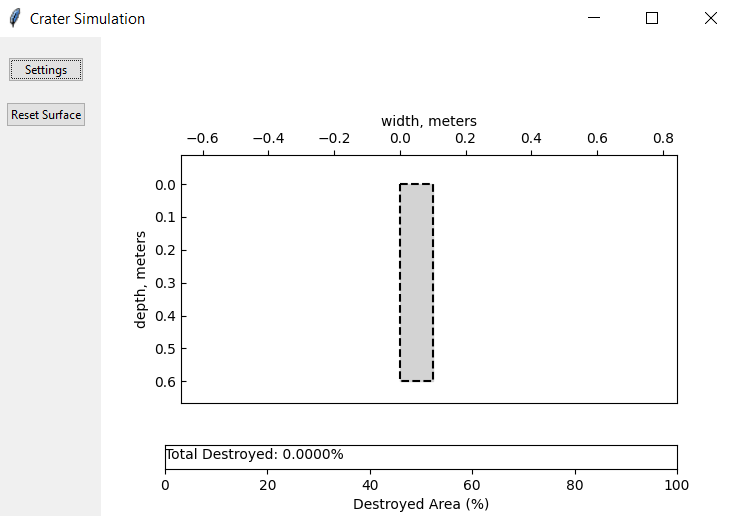


Рис. 10 Интерфейс программы с отображенной целью

**3.** **Подбор параметров**: параметры снаряда подобраны соответственно статье и физическим свойствам стали и алюминия (Рис. 11).

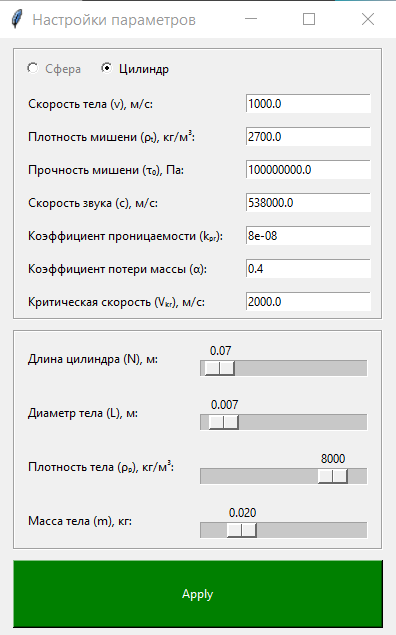


Рис. 11 Окно настроек с заданными параметрами

**4.** **Сравнение с реальными значениями**: выбрано 5 экспериментов из списка в вышеупомянутой статье, таблица сравнения результатов (Талица 9) представлена ниже.

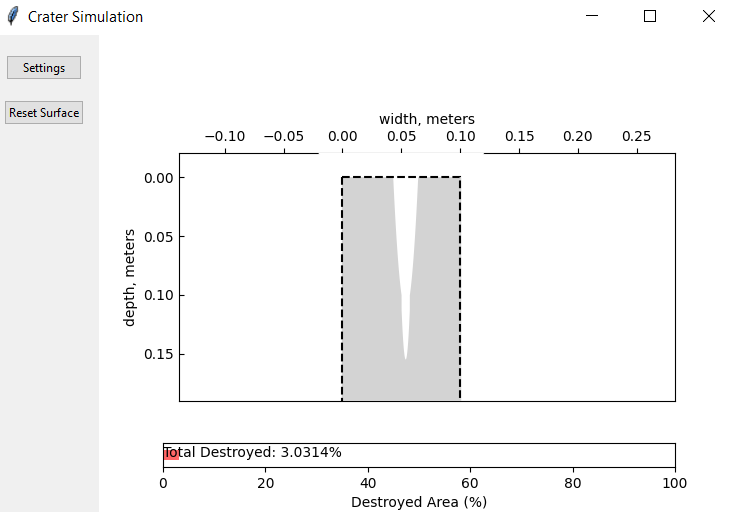


Рис. 12 Кратер после одного удара

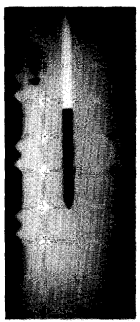


Рис. 13 Реальный вид сбоку (эксперимент в статье)

Таблица 9 Сравнение реальных результатов с полученными

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Номер эксперимента | Скорость при ударе (м/c) | Реальное значение глубины (мм) | Полученное значение глубины (мм) | Ошибка |
| 1 | 1585 | 105 | 154 | 33% |
| 2 | 1770 | 142 | 155 | 8% |
| 3 | 2166 | 124 | 156 | 21% |
| 4 | 2570 | 144 | 157 | 8% |
| 5 | 2988 | 147 | 158 | 7% |

Как сказано ранее, ошибка возникает из-за особенностей используемых физических формул, а также из-за угла удара, сопротивления воздуха, внутреннего трения, разрушения снаряда которые в данной модели задачи не учитываются.

**5.** **Демонстрация функционала нескольких ударов с другими значениями:**

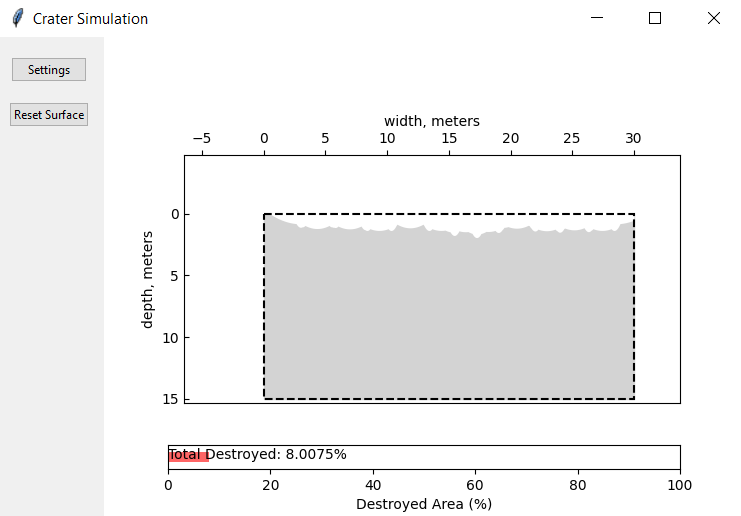


Рис. 14 Несколько ударов вдоль поверхности цели

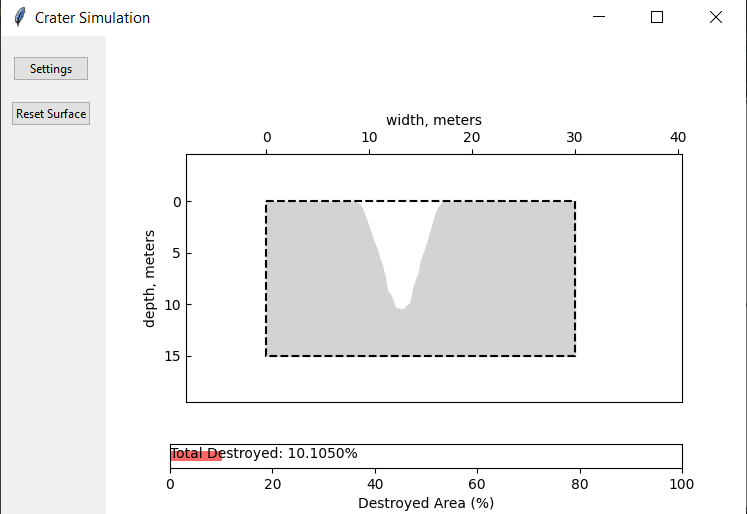


Рис. 15 Несколько ударов в окрестности одной точки

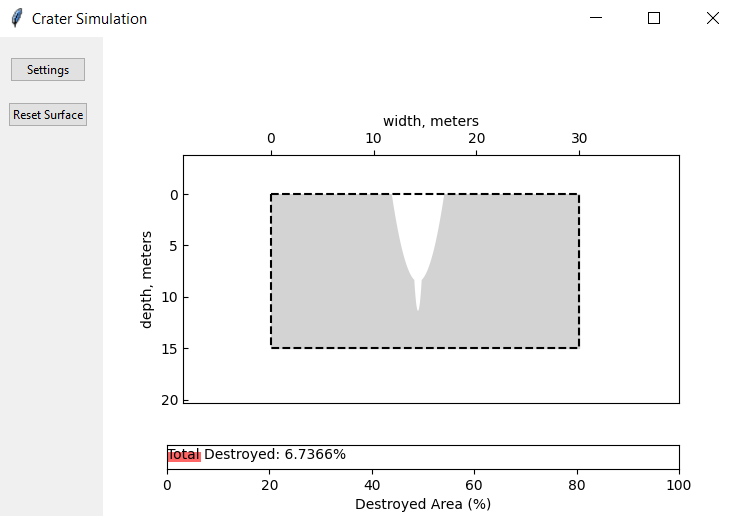


Рис. 16 Несколько ударов точно в одну точку

**6.** **Демонстрация функционала нескольких ударов в 3D реализации:**

Для 2D и 3D случая использовались одинаковые формулы, и естественно получатся идентичные значения глубины и ширины кратера при правильно подобранных параметрах, поэтому повторные сравнения с реальными данными излишни.

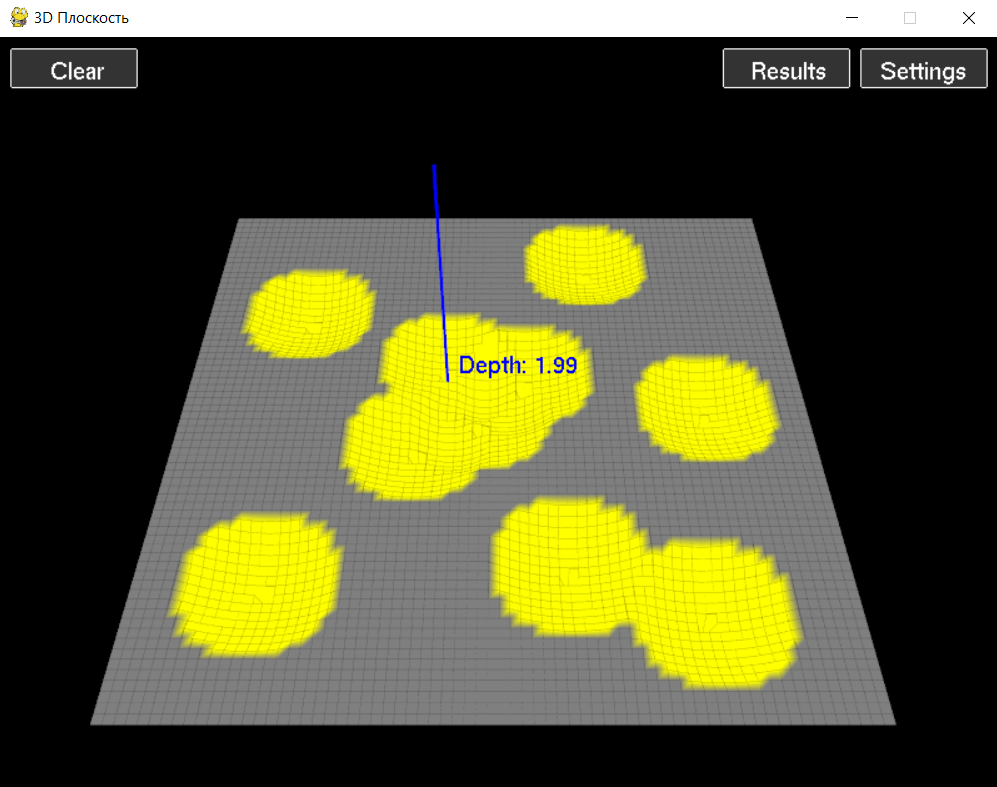


Рис 17. Несколько хаотичных ударов

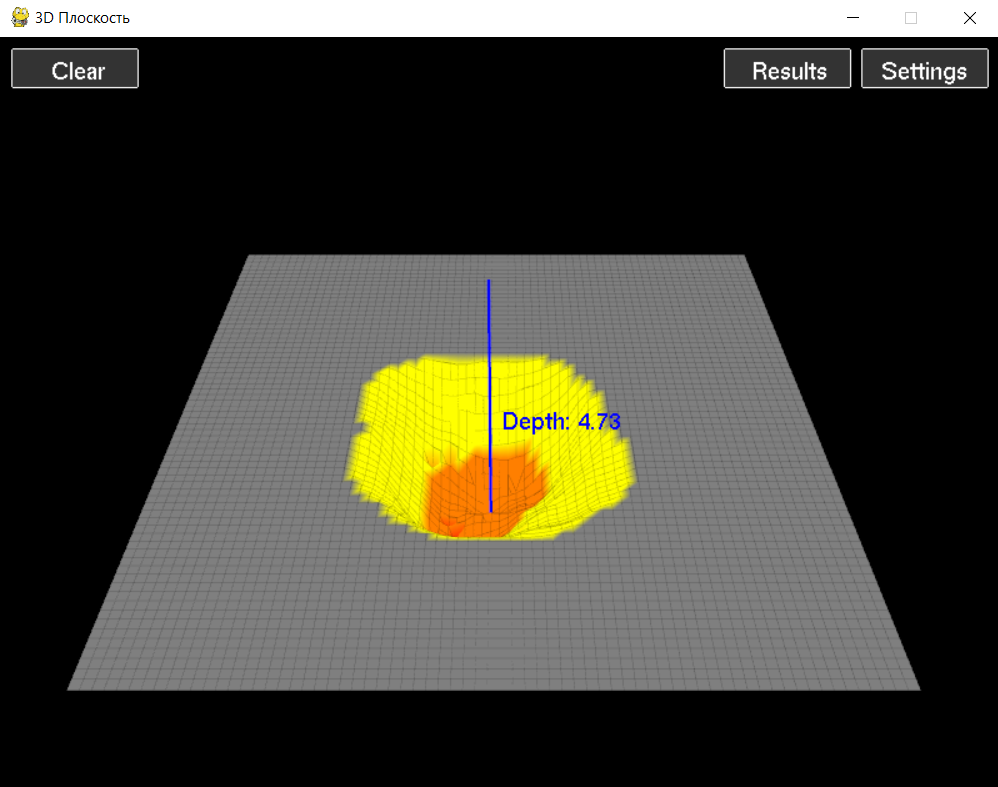


Рис. 18 Несколько ударов в окрестности одной точки, вид сверху



Рис. 19 Несколько ударов в окрестности одной точки, вид сбоку

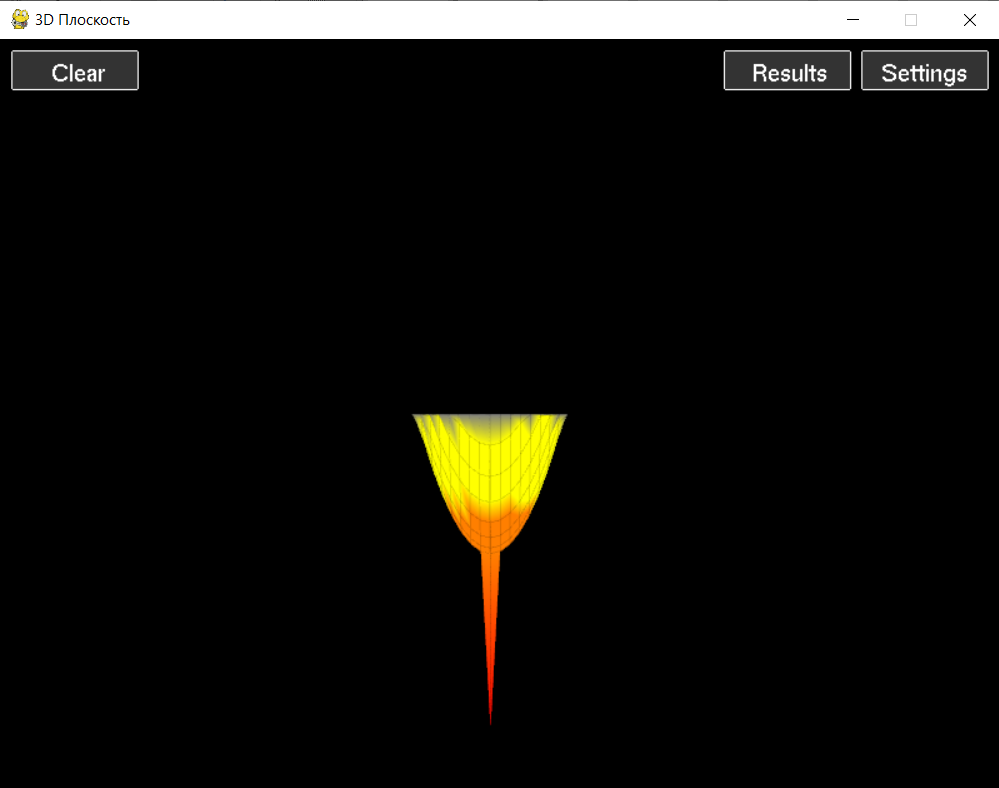


Рис. 20 Несколько ударов точно в одну точку

Как можно заметить, твёрдая фаза у кратера (“Глубокий носик”) получается достаточно острым, это обусловлено тем, что в 3D случае кратер строится по квадратной сетке, которая в свою очередь приводит к более резким переходам в форме кратера, особенно в областях с высокой градиентной изменчивостью. Данная проблема решается уменьшением сетки, но требует больших мощностей от компьютера.

# Анализ работы

Для вычисления оптимального количества боеголовок и оценки, как влияет увеличение их количества на расширение области поражения и на глубину проникновения в поражаемый объект, было проведено множество испытаний с заданными параметрами: Суммарная масса головной части – 3540 кг, масса боеголовок: 6 – 590 кг, 7 – 505 кг, 8 – 442 кг, 9 – 393 кг, 10 – 354 кг, 322 кг, 295 кг. Плотность грунта – 1500 кг/м3, бетона – 2500 кг/м3, коэффициент потери массы – 0.4, критическая скорость – 2000 м/с. Для бетона: прочность мишени – 25000000 Па, скорость звука – 4000 м/с, коэффициент проницаемости – 0.0000007. Для каменистого грунта: прочность мишени – 10000000 Па, скорость звука – 1500 м/с, коэффициент проницаемости – 0.000004.

Таблица 10. Анализ работы

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Форма снаряда | Скорость снаряда (м/c) | Материал цели | Количество боеголовок | Глубина поражения (м удары в точку) | Объём поражения (м3, удары вокруг точки) |
| Цилиндр 1:6 | 3400 | Грунт | 6 | 36,6 | 72,87 |
| Цилиндр 1:6 | 3400 | Грунт | 7 | 45,1 | 73,16 |
| Цилиндр 1:6 | 3400 | Грунт | 8 | 46 | 73,54 |
| Цилиндр 1:6 | 3400 | Грунт | 9 | 50,2 | 72,96 |
| Цилиндр 1:6 | 3400 | Грунт | 10 | 53,1 | 73,49 |
| Цилиндр 1:6 | 3400 | Грунт | 11 | 58,5 | 73,88 |
| Цилиндр 1:6 | 3400 | Грунт | 12 | 61,7 | 74 |
| Цилиндр 1:6 | 3400 | Бетон | 6 | 34,7 | 52,41 |
| Цилиндр 1:6 | 3400 | Бетон | 7 | 40,4 | 52,76 |
| Цилиндр 1:6 | 3400 | Бетон | 8 | 41,3 | 53,3 |
| Цилиндр 1:6 | 3400 | Бетон | 9 | 45 | 53,13 |
| Цилиндр 1:6 | 3400 | Бетон | 10 | 47,7 | 52,98 |
| Цилиндр 1:6 | 3400 | Бетон | 11 | 52,5 | 53 |
| Цилиндр 1:6 | 3400 | Бетон | 12 | 55,4 | 53,58 |
| Цилиндр 1:6 | 6800 | Грунт | 6 | 41,2 | 165,61 |
| Цилиндр 1:6 | 6800 | Грунт | 7 | 48 | 165,75 |
| Цилиндр 1:6 | 6800 | Грунт | 8 | 49,1 | 165,98 |
| Цилиндр 1:6 | 6800 | Грунт | 9 | 53,5 | 166,7 |
| Цилиндр 1:6 | 6800 | Грунт | 10 | 56,7 | 166,34 |
| Цилиндр 1:6 | 6800 | Грунт | 11 | 62,3 | 166,2 |
| Цилиндр 1:6 | 6800 | Грунт | 12 | 65,8 | 166,54 |
| Цилиндр 1:6 | 6800 | Бетон | 6 | 37,1 | 123,29 |
| Цилиндр 1:6 | 6800 | Бетон | 7 | 43,2 | 123,14 |
| Цилиндр 1:6 | 6800 | Бетон | 8 | 44,3 | 123,42 |
| Цилиндр 1:6 | 6800 | Бетон | 9 | 48,3 | 122,99 |
| Цилиндр 1:6 | 6800 | Бетон | 10 | 50,6 | 123,1 |
| Цилиндр 1:6 | 6800 | Бетон | 11 | 56,1 | 123,32 |
| Цилиндр 1:6 | 6800 | Бетон | 12 | 59,3 | 123,29 |
| Сфера | 3400 | Грунт | 6 | 14 | 114,85 |
| Сфера | 3400 | Грунт | 7 | 16,3 | 114,7 |
| Сфера | 3400 | Грунт | 8 | 18,6 | 114,64 |
| Сфера | 3400 | Грунт | 9 | 20,8 | 114,93 |
| Сфера | 3400 | Грунт | 10 | 23 | 115 |
| Сфера | 3400 | Грунт | 11 | 25,3 | 114,86 |
| Сфера | 3400 | Грунт | 12 | 27,5 | 114,91 |
| Сфера | 3400 | Бетон | 6 | 12,9 | 83,37 |
| Сфера | 3400 | Бетон | 7 | 15 | 83,31 |
| Сфера | 3400 | Бетон | 8 | 17,1 | 83,29 |
| Сфера | 3400 | Бетон | 9 | 19 | 83,27 |
| Сфера | 3400 | Бетон | 10 | 21,2 | 83,53 |
| Сфера | 3400 | Бетон | 11 | 23 | 83,62 |
| Сфера | 3400 | Бетон | 12 | 25,2 | 83,23 |
| Сфера | 6800 | Грунт | 6 | 16,2 | 269,39 |
| Сфера | 6800 | Грунт | 7 | 18,7 | 268,66 |
| Сфера | 6800 | Грунт | 8 | 21,3 | 269,06 |
| Сфера | 6800 | Грунт | 9 | 23,8 | 268,89 |
| Сфера | 6800 | Грунт | 10 | 26,3 | 269,47 |
| Сфера | 6800 | Грунт | 11 | 28,8 | 269,36 |
| Сфера | 6800 | Грунт | 12 | 31,3 | 269,37 |
| Сфера | 6800 | Бетон | 6 | 15 | 204,7 |
| Сфера | 6800 | Бетон | 7 | 17,4 | 204,13 |
| Сфера | 6800 | Бетон | 8 | 19,6 | 204,52 |
| Сфера | 6800 | Бетон | 9 | 22 | 204,38 |
| Сфера | 6800 | Бетон | 10 | 24,3 | 204,7 |
| Сфера | 6800 | Бетон | 11 | 26,6 | 204,89 |
| Сфера | 6800 | Бетон | 12 | 28,9 | 204,58 |

Из данной таблицы можно сделать вывод, что увеличение количества боеголовок при сохранении суммарной массы головной части позволяет достичь большей глубины проникновения в цель, а иногда и незначительного увеличения объёма поражения. Однако эффективность такого увеличения зависит от материала цели, формы снаряда и начальной скорости.

Более детальное рассмотрение:

* **Для цилиндрических снарядов** при скорости 3400 м/с наблюдается значительный рост глубины поражения с увеличением числа боеголовок как в грунте, так и в бетоне, при этом объём поражения увеличивается лишь незначительно.
* При **увеличении скорости до 6800 м/с** глубина поражения увеличивается заметнее, а объём поражения, особенно в грунте, резко возрастает, оставаясь практически постоянным при дальнейшем росте числа боеголовок — это указывает на насыщение по объёму при заданной энергии удара.
* **Сферическая форма** обеспечивает меньшую глубину проникновения по сравнению с цилиндрической, но при этом даёт существенно больший объём поражения, особенно при высоких скоростях. Это связано с большей фронтальной площадью, создающей более широкую, но менее глубокую зону разрушения.
* Для **бетона** глубина проникновения всегда ниже, чем в грунте, что объясняется его большей плотностью и прочностью, но объём поражения для сферических снарядов остаётся высоким за счёт распределения энергии по большей площади.

Таким образом, оптимальным с точки зрения глубины проникновения и объёма поражения является использование **12 боеголовок**. При этом сохраняется суммарная масса головной части, а увеличение количества боеголовок позволяет достичь максимальных значений глубины при незначительном увеличении объёма поражения.

# Вывод

Для анализа влияния количества боеголовок на область поражения и глубину проникновения были выбраны формулы, описывающие гидродинамическую и твёрдую фазы гиперзвукового удара. Основное внимание уделялось формулам из статьи Н. И. Шишкина для расчёта диаметра и глубины кратера, адаптированным для сферических и цилиндрических тел.

В ходе работы рассматривались следующие ключевые параметры: количество боеголовок (от 6 до 12), суммарная масса головной части, скорость снаряда, плотность материала боеголовок и мишени, прочность мишени, критическая скорость и так далее. Эти параметры позволили оценить зависимость глубины проникновения и объёма поражения от числа боеголовок и условий удара.

Основными критериями успешности работы являлись глубина проникновения (максимальная при ударах в одну точку) и объём поражения (максимальный при распределённых ударах). Оптимальным числом боеголовок признано 12, так как оно обеспечивало максимальную глубину проникновения при незначительном увеличении объёма поражения. Для сферических снарядов объём поражения был выше, но глубина проникновения существенно ниже.

Так же реализована программа на Python, представляющая интерактивный графический интерфейс для 2D и 3D визуализации кратеров. Пользователь может задавать множество параметров боеголовок, создавать кратеры на заданном поле, изменять масштаб и угол обзора, а также анализировать результаты в окне характеристик. Визуализация позволяет наглядно демонстрировать влияние параметров на результаты удара, делая программу удобным инструментом для анализа.

# Источники

* Редактор блок-схем.

[*https://programforyou.ru/block-diagram-redactor*](https://programforyou.ru/block-diagram-redactor)

*дата обращения: (10.05.2025)*

* tkinter — Библиотека для создания графических интерфейсов в Python.

[*https://docs.python.org/3/library/tkinter.html*](https://docs.python.org/3/library/tkinter.html%20)

*дата обращения: (08.04.2025)*

* math — Библиотека для работы с математическими функциями.

[*https://docs.python.org/3/library/math.html*](https://docs.python.org/3/library/math.html)

*дата обращения: (08.04.2025)*

* openpyxl — Библиотека для записи и чтения файлов Excel.

[*https://openpyxl.readthedocs.io/en/stable/*](https://openpyxl.readthedocs.io/en/stable/)

*дата обращения: (08.04.2025)*

* *Со списком литературы, посвященным формулам, гиперзвуковым ракетам, боеголовкам и т.д., плюсами и минусами всех статей и книг можно ознакомится по ссылке:* [*Полный список литературы*](https://docs.google.com/spreadsheets/d/1b0A9oVrynpuSC6iZtCWN483JVTWMmDArV6LuvXwJytQ/edit?gid=1381590120#gid=1381590120)

*дата обращения: (15.05.2025)*