|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ:            «СПЕЦИАЛЬНОЕ МАШИНОСТРОЕНИЕ»

КАФЕДРА:        «РАКЕТНЫЕ И ИМПУЛЬСНЫЕ СИСТЕМЫ» (СМ-6)

**РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**К КУРСОВОЙ РАБОТЕ**

**НА ТЕМУ:**

**Баллистическое проектирование  
артиллерийских орудий**

ВАРИАНТ № 33

Выполнил: студент группы СМ6-71 **\_\_\_\_\_\_\_\_\_\_\_\_** И. М. Берулава

(Подпись, дата) (И. О. Фамилия)

Руководитель курсовой работы: **\_\_\_\_\_\_\_\_\_\_\_\_** М. С.Товарнов

(Подпись, дата) (И. О. Фамилия)

**Министерство науки и высшего образования Российской Федерации**

**Федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**«Московский государственный технический университет имени Н.Э. Баумана**

**(национальный исследовательский университет)»**

**(МГТУ им. Н.Э. Баумана)**

УТВЕРЖДАЮ

Заведующий кафедрой СМ6

(Индекс)

              В.М. Кашин

(И.О.Фамилия)

«       »                          202   г.

# ЗАДАНИЕ

**на выполнение курсовой работы**

по дисциплине:                       баллистика ракетного и ствольного оружия

Студент группы:   СМ6-71

Берулава Илья Мамукович

(Фамилия, имя, отчество)

Тема курсовой работы:      баллистическое проектирование артиллерийских орудий

Направленность КР:           учебная

Источник тематики (кафедра, предприятие, НИР):        кафедра

График выполнения работы: 25% к   4   нед., 50% к  6   нед., 75% к   9   нед., 100% к   11   нед.

***Вариант задания:***       **№ 33**

***Задание:***

Определить такую марку пороха и условия заряжания, при которых достигается минимальное значение объема канала ствола и обеспечивается дульная скорость снаряда = 1000 м/с, если снаряд имеет массу кг, а максимально допустимое среднебаллистическое давление в канале ствола не должно превышать МПа. Ствол имеет калибр м и длину м. Помимо этого, максимальное допустимое ускорение снаряда при температуре окружающей среды : , а давление на дульном срезе при температуре не должно превышать МПа. Задачу требуется решить, используя квазиодномерную модель внутрибаллистических процессов.

***Оформление курсовой работы***

Курсовая работа содержит:

расчетно-пояснительную записку на 42 листах формата А4;

2 программы в среде Python.

Дата выдачи задания « 16 »   сентября   2022 г.

**Руководитель курсовой работы**                                              М. С. Товарнов

(Подпись, дата) (И. О. Фамилия)

**Студент**                                             И. М. Берулава

(Подпись, дата) (И. О. Фамилия)

**ОГЛАВЛЕНИЕ**

[ЗАДАНИЕ 2](#_Toc119330914)

[Введение 4](#_Toc119330915)

[1. Прямая задача внутренней баллистики 5](#_Toc119330916)

[1.1. Математическая модель прямой задачи в квазиодномерной постановке 5](#_Toc119330917)

[1.2. Численное интегрирование системы уравнений 10](#_Toc119330918)

[1.3. Обоснование шага интегрирования 14](#_Toc119330919)

[Отсюда следует, что необходимо принять количество ячеек больше, чем 200, чтобы получить решение без потери заданной точности. 15](#_Toc119330920)

[1.4. Проверка работы программы по решению прямой задачи 15](#_Toc119330921)

[2. Задача баллистического проектирования 18](#_Toc119330922)

[2.1. Постановка задачи баллистического проектирования 18](#_Toc119330923)

[2.2. Алгоритмическая и программная реализация решения задачи баллистического проектирования 20](#_Toc119330924)

[2.3. Решение задачи баллистического проектирования 21](#_Toc119330925)

[2.4. Решение прямой задачи внутренней баллистики для найденного оптимального решения задачи баллистического проектирования 25](#_Toc119330926)

[Заключение 26](#_Toc119330927)

[Список использованной литературы 27](#_Toc119330928)

[Приложение 1 28](#_Toc119330929)

[Приложение 2 36](#_Toc119330930)

# Введение

Цель курсовой работы состоит в формировании у студентов умений и навыков в решении задач баллистического проектирования, которые, в общем случае, относится к классу задач многокритериальной параметрической оптимизации [1, с. 9], и в получении необходимых знаний для составления методики баллистического проектировании, состоящей в разработке стратегии, позволяющей максимально эффективно решать задачи баллистического проектирования и подобные ей задачи.

В курсовой работе основной задачей является решение задачи баллистического проектирования ствола артиллерийского орудия, состоящая в выборе наилучшего решения из множества решений обратной задачи, суть которой состоит в выборе наилучших параметров канала ствола, условий заряжания и основных характеристик заряда, обеспечивающих достижение необходимого результата при следующих входных данных: калибр ствола, тип ствола, масса снаряда, скорость снаряда на дульном срезе, а также дополнительных ограничений на решение.

В первой части рассматривается математическая модель основной задачи внутренней баллистики в «квазиодномерной» постановке.

Во второй части приведено решение обратной задачи по заданному критерию при ограничениях варианта курсовой работы и точности ответа. Критерием определения наилучшего решения является объем канала ствола .

В результате решения задачи баллистического проектирования ожидается получить такие условия заряжания, при которых обеспечиваются заданные параметры выстрела, в соответствии со всеми ограничениями, и обеспечивается минимальный объем каморы.

# Прямая задача внутренней баллистики

Прямая задача внутренней баллистики состоит в том, чтобы при известных условиях заряжания, массе снаряда и геометрии ствола определить основные баллистические характеристики выстрела, такие как дульная скорость снаряда и максимальное среднебаллистическое давление в канале ствола , а также зависимости , , , и. Решение задачи баллистического проектирования основывается на многократном решении прямой задачи и выбора оптимального решения из всех рациональных.

Так как получение точных данных по распределению массы газа и несгоревшего пороха затруднено, то для учета затраты энергии их перемещения вводится следующее допущение: пороховые газы и пороховые элементы представляют собой гомогенную смесь, которая движется с общей для газа и частиц скоростью. Такое упрощение называется гипотезой односкоростной газопороховой смеси (ОГПС).

## Математическая модель прямой задачи в квазиодномерной постановке

Выстрел представляет собой очень сложный процесс, при описании которого даже сегодня требуется пользоваться разными приближенными теориями для его описания. В данном варианте курсовой работы предлагается решить задачу баллистического проектирования пользуясь квазиодномерным методом моделирования внутрибаллистических процессов. Более корректный учет волновых движений газа позволяют осуществить газодинамические модели внутренней баллистики. В большинстве они все квазиодномерные. Основное различие моделей может состоять в отдельном учете движения конденсированной фазы. При этом модель горения пороха, как правило остается в рамках геометрического закона.

Газодинамическая задача рассматривается в приближении ОГПС. ОГПС в данном случае представляет собой «псевдогаз», ее движение в заснарядном объеме описывается стандартными уравнениями сохранения массы, импульса и энергии, которые можно записать в двух формах – эйлеровой и лагранжевой. В последнем случае применяются лагражевы координаты, поскольку суммарная масса ОГПС в заснарядном объеме не изменяется. [1, c. 18]

Рассмотрим допущения, которые используются при использовании «квазиодномерной» модели:

* Гипотеза ОГПС: предполагается, что пороховые газы и несгоревший порох перемешаны равномерно и представляют собой смесь, которая движется с общей для газовой и конденсированной фаз скоростью;
* Геометрический закон горения пороха:
  + Однородность пороховых элементов по свойствам и размерам;
  + Одновременное мгновенное воспламенение всех пороховых элементов;
  + Горение пороховых элементов осуществляется параллельными слоями;
* Термодинамические характеристики пороховых газов принимаются постоянными (
* Реальные свойства пороховых газов учитываются введением в уравнение состояния коволюма ;
* Выполняется степенной закон скорости горения (в частном случае : – линейный закон скорости горения);
* Теплоотдача от пороховых газов к стенкам канала ствола учитывается в соответствии с уравнением Ньютона-Рихмана:
* где температура пороховых газов, температура стенки канала ствола, коэффициент теплоотдачи от пороховых газов к стенке канала ствола;
* Процесс врезания ведущего пояска снаряда в нарезы канала ствола учитывается введением давления форсирования
* Утечка газа через зазоры между ведущими устройствами и стенками канала ствола не учитываются;
* Не учитывается влияние деформации стенок канала ствола на заснарядный объём.

В массовых лагранжевых координатах система уравнений движения ОГПС имеет следующий вид:

где m – массовая лагранжева координата, – плотность газопороховой смеси, – внутренняя энергия единицы массы газа.

Для учета горения эту систему уравнений нужно дополнить [1, с. 18]:

* Уравнениями горения пороха:

где z – безразмерная толщина сгоревшего свода порохового элемента, – полный импульс давления пороховых газов, ,, и – коэффициенты формы порохового зерна, функция Хевисайда.

* Уравнением состояния ОГПС:

где – относительная масса сгоревшей части заряда, – плотность ОГПС, – плотность пороха, – коволюм порохового газа (эффективный собственный объем молекул).

* Уравнением движения снаряда:

где – давление на дно снаряда со стороны пороховых газов, – суммарная сила сопротивления движению снаряда.

С целью упрощения расчетов можно применить для газодинамической модели подход для учета потерь на теплоотдачу как для «нульмерной» модели. При этом, принятые при расчете теплоотдачи для «нульмерной» модели допущения нужно применить в масштабе одной ячейки. Температуру поверхности ствола при этом в каждой ячейке считают постоянной.

Число Рейнольдса определяют по формуле:

где коэффициент динамической вязкости; средняя скорость потока. В качестве скорости потока принимают значение скорости потока в ячейке.

где теплоемкость, плотность и теплопроводность материала ствола соответственно; начальная температура.

Коэффициент динамической вязкости пороховых газов можно аппроксимировать формулой Сазерленда:

где .

Динамика нагрева стенки описывается формулой (2) для каждой ячейки, а учет изменения температуры при передвижении площади контакта осуществляется посредством пересчета средних температур в ячейках на каждом временном шаге для новых границ ячеек. [1, c. 26]

В традиционном описании баллистических процессов большую часть форм сопротивления движению снаряда учитывают, умножая его массу на коэффициент фиктивности.

Однако в газодинамической постановке коэффициент фиктивности массы снаряда не используется – разгон газа учитывается в самих уравнениях сохранения. [1, с. 19]

Две наиболее существенные составляющие сопротивления движению снаряда – сила, возникающая при сжатии воздушного столба перед снарядом, и сила сопротивления движению по нарезам (либо сила трения ведущих устройств о стенки канала ствола, если нарезы отсутствуют):

где – давление воздушного столба перед снарядом; – сила взаимодействия ведущих устройств со стволом. Для нарезов постоянной крутизны эта сила пропорциональна давлению на дно снаряда и может быть учтена в форме:

где (для данного варианта курсовой работы).

Противодавление воздуха перед снарядом существенно влияет на движение снаряда со сверхзвуковыми относительно невозмущенного воздуха скоростями и может быть приближенно учтено с использованием известной формулы для точного решения задачи о поршне, сжимающем газ:

где , и – показатель адиабаты, давление и скорость звука в невозмущенном воздухе. [1, c. 23]

В ряде случаев силой трения в нарезах можно пренебречь. Учет сопротивления движению снаряда осуществляется домножением левой части уравнения движения снаряда на коэффициент учета сил сопротивления . Вследствие чего эквивалентный вид уравнения движения снаряда будет выглядеть следующим образом:

* 1. **Численное интегрирование системы уравнений**

Полная система квазиодномерных уравнений внутренней баллистики и массовых лагранжевых координат имеет вид [1, с. 31]:

где – плотность теплового потока от газа в стенки ствола, – число Нуссельта, – число Рейнольдса, – число Прандтля (для порохового газа = 0,74), – теплопроводность пороховых газов, – давление воздушного столба перед снарядом, – массовая доля пороховых газов, доля сгоревшего пороха, плотность пороха, коволюм порохового газа, сила пороха, показатель адиабаты, газовая постоянная пороховых газов и масса – й навески смеси соответственно.

Введем равномерную разностную сетку по переменным и , и будем обозначать индексом временные узлы, а индексом – пространственные (массовые). При этом полуцелые координатные индексы используются для обозначения параметров внутри ячейки (таких как , 𝑝 и ), целые – на границах (координаты границ ячеек и скорости границ ).

Конечно – разностная явная аппроксимация имеет следующий вид:

Здесь как описывалось ранее и – индексы узлов разностной сетки по времени и массовой координате соответственно; – номер порохового состава.

Порядок формул приведен в соответствии с последовательностью вычислений. Схема имеет первый порядок аппроксимации по времени и координате.

Шаг по времени вычисляется исходя из условий устойчивости Куранта– Фридрихcа – Леви:

где CFL – число Куранта (0 < CFL < 1), – скорость звука в ячейке,

Удобство приведенной численной схемы заключается в ее простой физической интерпретации: закрытые ячейки газа (между ячейками отсутствует обмен массой), в каждую из которых заключен газ c массой и давлением , приводят в движение другие ячейки, а также перегородки между ними, массой . Одной из таких перегородок является снаряд, что позволяет отказаться от отдельного уравнения, описывающего его движение, если ввести массу перегородки , как это сделано в первом уравнении системы (1). Условие на неподвижной левой границе может быть приближенно заменено введением фиктивной очень большой массы . Это позволяет использовать данную схему без явного выделения границ. [1, с. 32]

Следует описать начальные и граничные условия. Однако перед этим следует дать определением давлению форсирования. В расчетах применяют гипотезу мгновенного врезания: движение снаряда начинается в момент, когда разность давлений, действующих на него, превышает некоторый условный предел, называемый давлением форсирования . В рамках данной курсовой работы для нарезных орудий эта величина равна 30 МПа. Постановка начальных условий зависит от момента времени, который принимают за начало отсчета. [1. с. 28]

В случае отсчета от момента вспышки начальные условия имеют вид:

* для ОГПС
* для снаряда

В этом случае до достижения момента форсирования, определяемого условием снаряд должен оставаться неподвижным.

При отсчете времени с момента форсирования начальные условия имеют вид:

* для ОГПС
* для снаряда

Удобство отсчета времени от момента форсирования заключается в отсутствии необходимости учета того, что снаряд должен оставаться неподвижным до достижения в камере давления . [1, с. 29]

При решении задачи в газодинамической постановке помимо начальных условий требуется задать также и граничные условия. В качестве таковых выбирают условия непротекания (равенство скорости газа скорости поверхности) как на левой неподвижной границе (дно канала) расчетной области, так и на правой подвижной границе (дно снаряда):

При этом координата – переменная и отвечает текущему положению снаряда.

Существует готовая библиотека pyballistics для решения прямой задачи внутренней баллистики в нульмерной и квазиодномерной постановках на языке программирования Python версии 3.6 и выше. Используемый метод для квазиодномерной модели называется ozvb\_largange. Исходный код библиотеки находится в открытом доступе. [2]

Расчет будет остановлен, если выполнено одно из условий:

* снаряд набрал необходимую дульную скорость;
* превышено допустимое среднебаллистическое давление;
* превышена допустимая длина ствола (координата снаряда);
* количество шагов интегрирования превысило 100 000 точек.
  1. **Обоснование шага интегрирования**

Правильный подбор количества ячеек при решении прямой задачи позволяет получить максимально производительные вычисления с необходимой точностью. Для наглядности построим график зависимости критерия оптимальности (в нашем случае ) от количества ячеек для исходных данных:

* Калибр мм;
* Масса снаряда ;
* Дульная скорость ;
* Максимальное среднебаллистическое давление не более ;
* Максимальное дульное давление не более при температуре ;
* Длина ствола не более 65 калибров;
* Оптимальная плотность заряжания должна быть найдена с точностью , а оптимальная относительная масса порохового заряда – с точностью .

Параметры БР: марка пороха «6/7 П-5 БП фл».

На рисунке 1 видно, что при увеличении количества ячеек уменьшается погрешность. При значении количества ячеек равном области БР уже не пересекаются. Это означает, что решение удовлетворяет заданной точности.

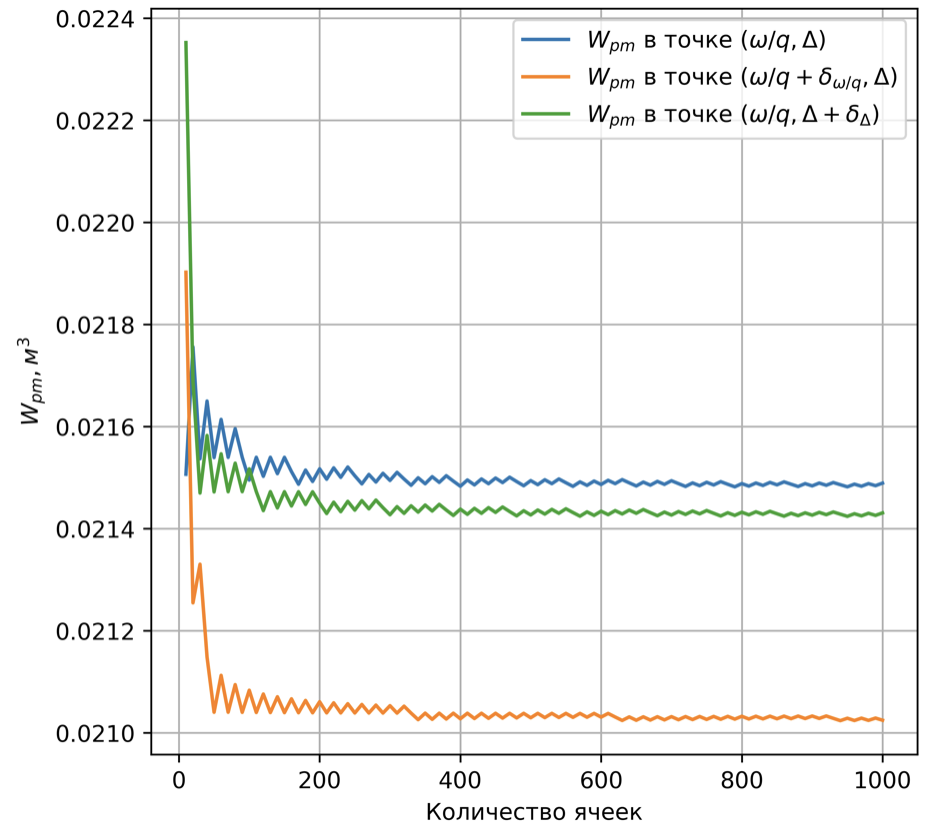


Рис. 1. Зависимость критерия от количества ячеек.

Отсюда следует, что необходимо принять количество ячеек больше, чем 300, чтобы получить решение без потери заданной точности.

## Проверка работы программы по решению прямой задачи

Прежде чем, перейти к решению прямой задачи баллистического проектирования, необходимо проверить ее программную реализацию решением тестовой задачи. Тестовая задача представляет собой прямую задачу внутренней баллистики, результаты решения которой известны. Исходные данные такой задачи: [1, с. 63]

* калибр орудия мм;
* масса снаряда кг;
* тип орудия: нарезное (Мпа);
* длина ведущей части ствола м;
* объем каморы м3;
* масса порохового заряда кг;
* марка пороха порохового заряда: 17/7;
* начальная температура заряда K;
* давление вспышки МПа;
* теплообмен с поверхностью ствола и сопротивление воздуха перед снарядом отсутствуют.

Характеристики марки порохового заряда можно найти в таблице П1 [1]. В таблице 1 сравниваются полученные в результате решения прямой задачи с помощью написанной программы значения с заранее известными значениями, обозначенные в таблице эталонными. На рисунках 2 и 3 соответственно приведены графики скорости снаряда, среднебаллистического давления и координаты снаряда в зависимости от времени для полученного решения и эталонного. Программный код решения задачи представлен в приложении 1.

Таблица 1. Сравнение полученных результатов с эталонным решением.

|  |  |  |  |
| --- | --- | --- | --- |
| Характеристика | Результат | Эталон | |
| Дульная скорость, м/с | 787.11 | 787.15 | |
| Максимальное среднебаллистическое давление, МПа | 271.34 | 271.23 | |
| Дульное среднебаллистическое давление, МПа | 64.46 | 64.47 | |
| Время начала движения снаряда, мс | 5.7896 | | 5.786 |
| Время достижения максимального среднебаллистического давления, мс | 11.776 | | 11.779 |
| Максимальное давление на дно снаряда, МПа | 273.56 | | 273.43 |
| Максимальное давление на дно канала ствола, МПа | 277.12 | | 277.37 |
| Время вылета снаряда, мс | 18.455 | | 18.459 |
| Максимальное ускорение снаряда, ед. g | 12170 | | 12165 |
| Время распада пороха (у дна снаряда), мс | 12.799 | | 12.802 |
| Время распада пороха (у дна канала ствола), мс | 12.367 | | 12.369 |
| Время полного сгорания пороха (у дна снаряда), мс | 17.075 | | 17.188 |
| Время полного сгорания пороха (у дна канала ствола), мс | 15.325 | | 15.406 |

Окончание таблицы 1.

|  |  |  |
| --- | --- | --- |
| Положение снаряда в момент сгорания пороха, м | 3.940 | 4.022 |

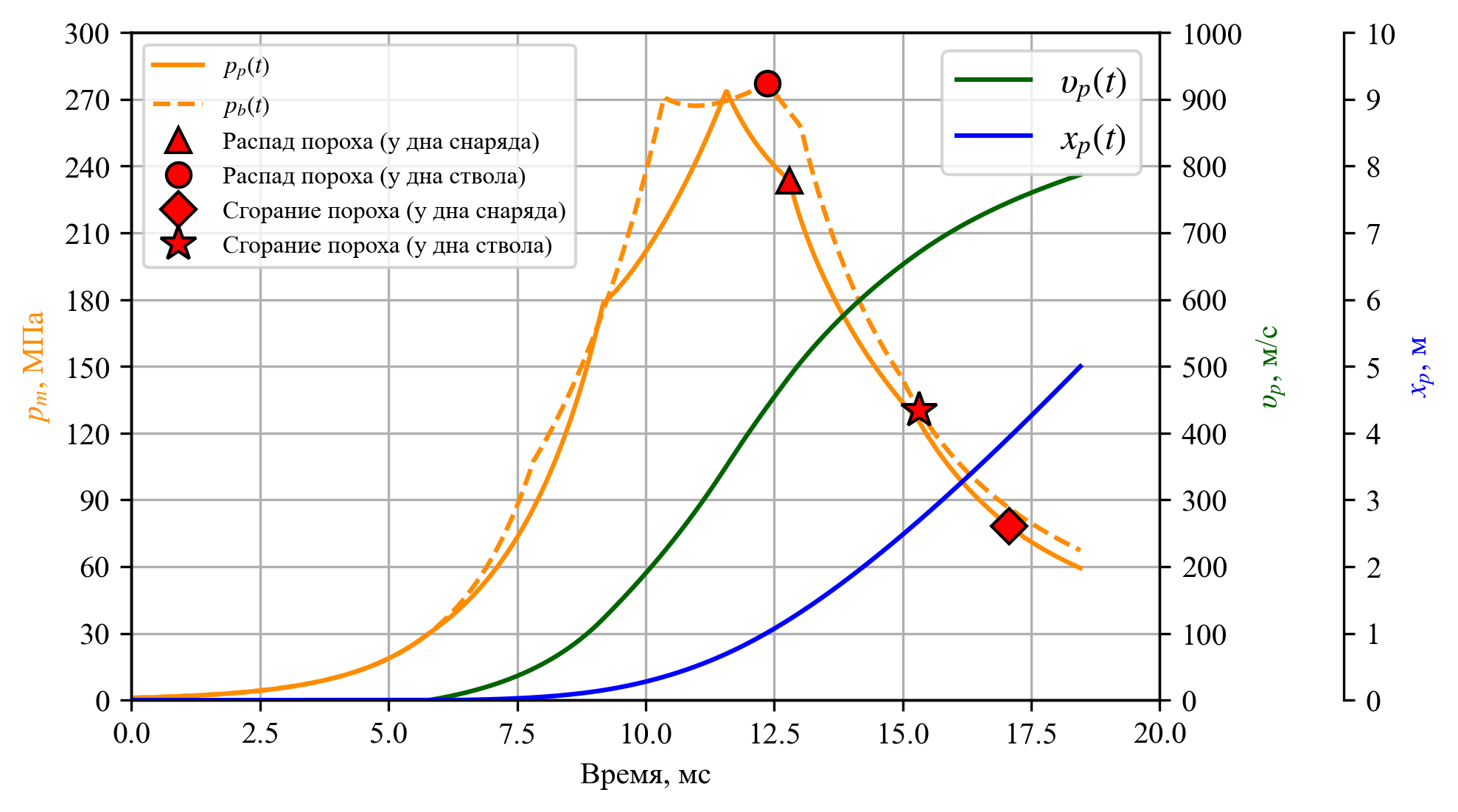


Рис. 2. Рассчитанные скорость, давление и координата снаряда.

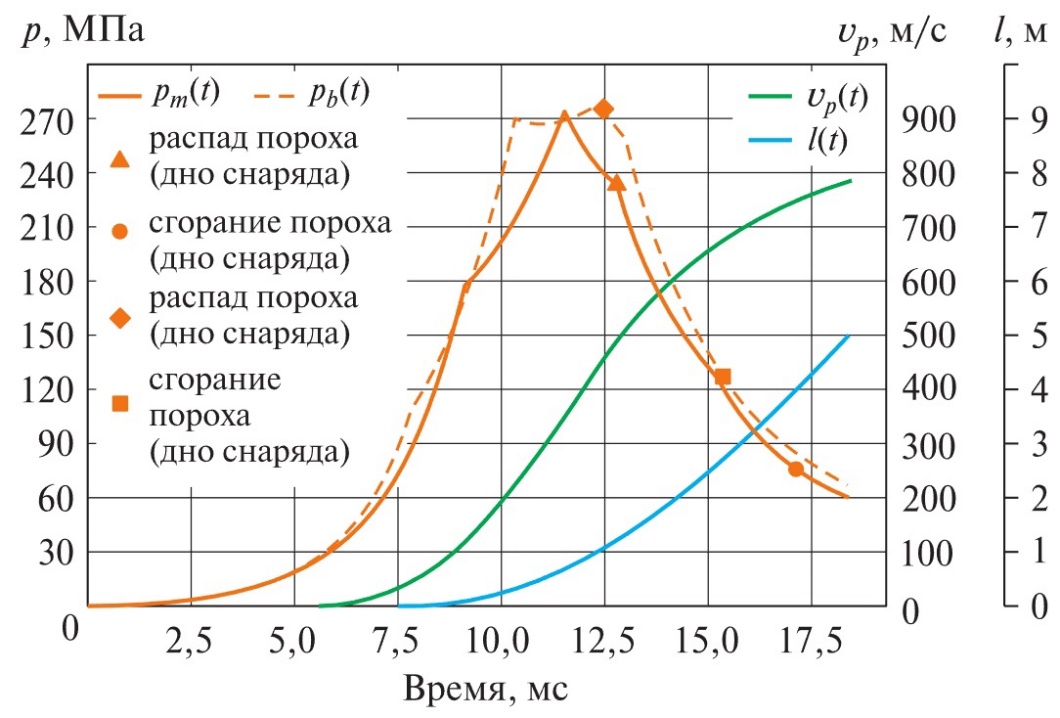


Рис. 3. Эталонные скорость, давление и координата снаряда.

Результаты решения совпадают с эталонными с достаточной точностью, это означает, что программа написана верно.

1. **Задача баллистического проектирования**

Задача баллистического проектирования состоит в нахождении оптимального решения среди множества решений обратной задачи внутренней баллистики. Обратная задача внутренней баллистики заключается в нахождении всех возможных значений конструктивных параметров и условий заряжания, обеспечивающих достижение заданных баллистических характеристик орудия.

Индивидуальные исходные данные варианта № 33:

* Калибр мм.
* Масса снаряда кг.
* Дульная скорость снаряда м/с.
* Тип орудия – нарезной.
* Тип математической модели: квазиодномерная.
* Критерий оптимальности: минимум объёма каморы

Общие ограничения к задаче:

* Максимально допустимое среднебаллистическое давление в канале ствола при нормальных условиях выстрела МПа.
* Максимальная длина ведущей части ствола м.

В соответствии с вариантом задания имеется два дополнительных ограничения:

* Максимально допустимое дульное среднебаллистическое давление при начальной температуре :
* Максимально допустимое ускорение снаряда при начальной температуре :м/

Согласно условию варианта курсовой работы, критерием оптимальности служит критерий минимума объёма каморы .

Также для решения существует требование по точности:

* Плотность заряжания должна быть найдена с точностью не менее 1 кг/;
* Относительная масса заряда должна быть найдена с точностью не менее 0.002.

По исходным данным получается, что критерии решения задачи выглядят как:

Прежде всего необходимо выбрать диапазоны варьирования входных параметров, к которым относятся:

* Марка пороха;
* Плотность заряжания ;
* Относительная масса заряда .

Поскольку тип орудия в данной курсовой работе не зависит от калибра (т.к. границы по этому параметру условны), для плотности заряжания, исходя из разумных предположений был взят следующий диапазон:

.

Количество ячеек в заснарядном объеме в рамках данной курсовой работы было выбрано равным 400. Данное количество ячеек обеспечивает достаточно быстрый расчет и требуемую точность. Поскольку математическая модель начинает обеспечивать требуемую точность при количестве ячеек равным 300 и выше.

Решение обратной задачи можно разбить на 2 этапа.

На первом этапе сетка разбивается грубо, с крупным шагом, способным дать общее представление о распределении множества допустимых баллистических решений для конкретного пороха и информацию о его наличии, если множество допустимых решений не является пустым, то производится поиск лучшего решения.

На втором этапе, когда часть неподходящих порохов убраны из рассмотрения и получены лучшие решения для всех остальных порохов, полученные решения уточняется путем уменьшения шага сетки, в один или два подхода, до необходимых значений для обеспечения заданной точности. Из зависимости (2) хорошо видно, что значения критерия оптимальности распределены равномерно, без областей с непредсказуемым его значением, что позволяет уточнять сетку в окрестности найденного решения без опасения потерять результат в другой части этой сетки.

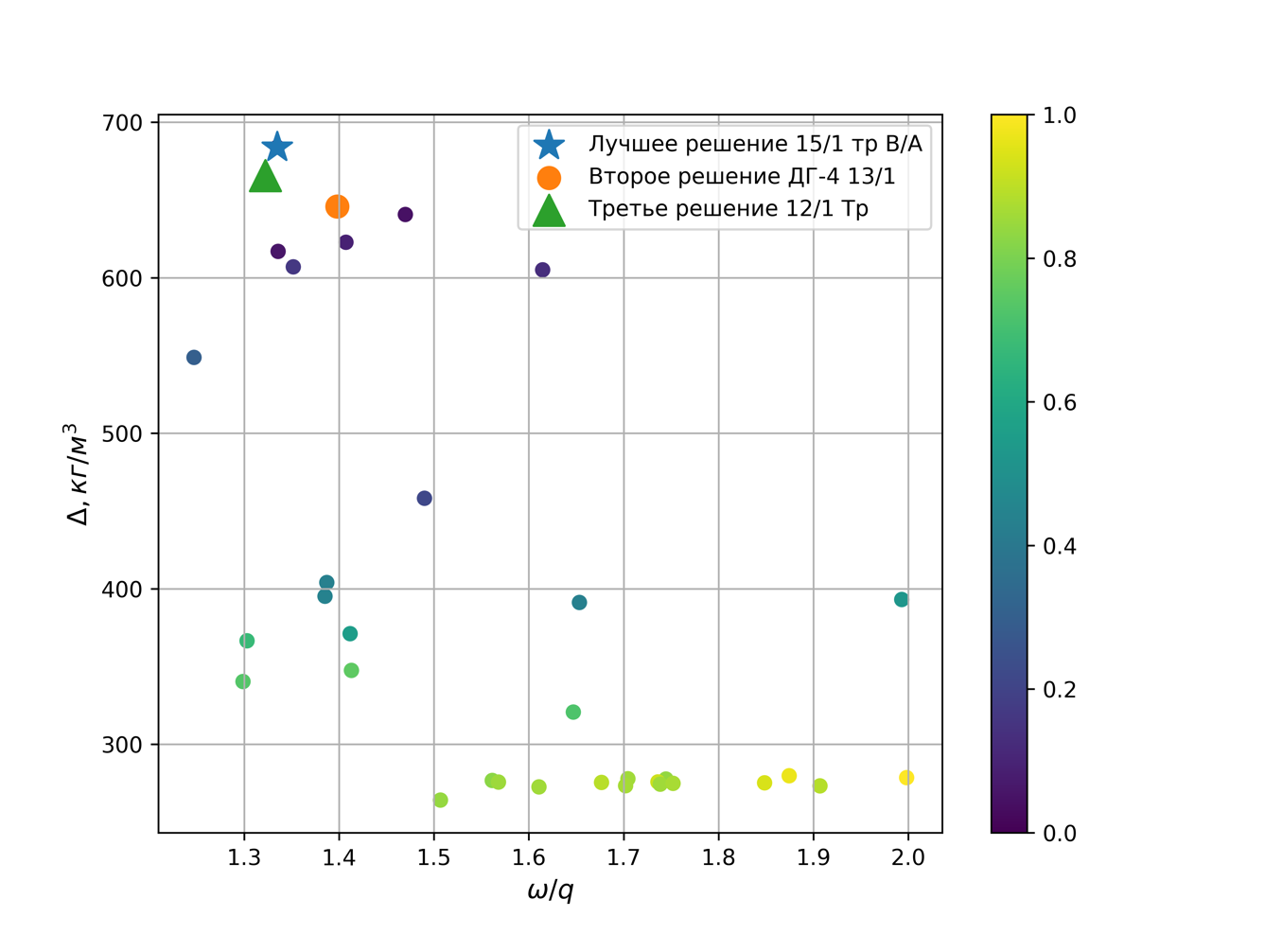


Рис. 2.1. Результаты первого этапа

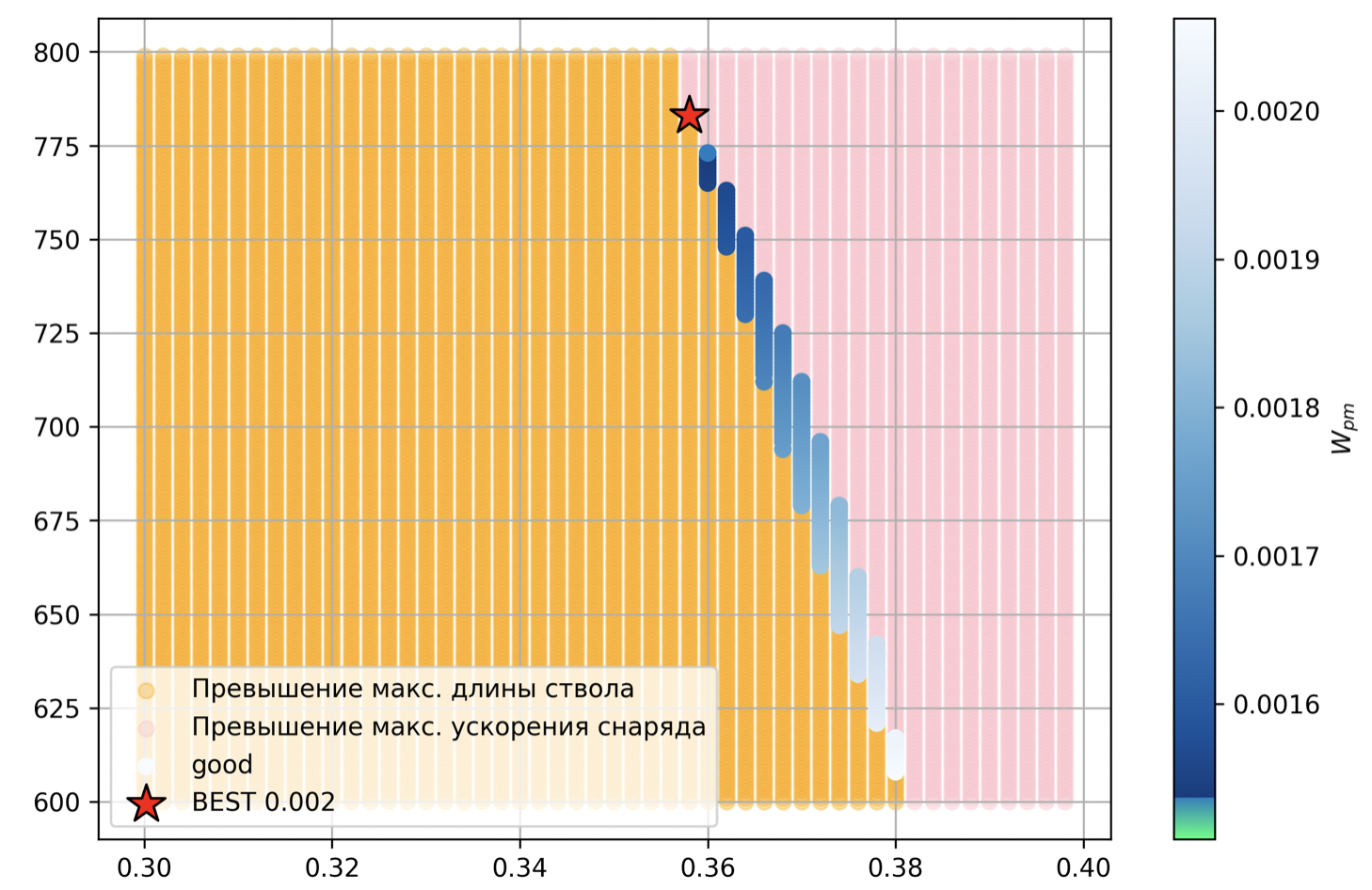
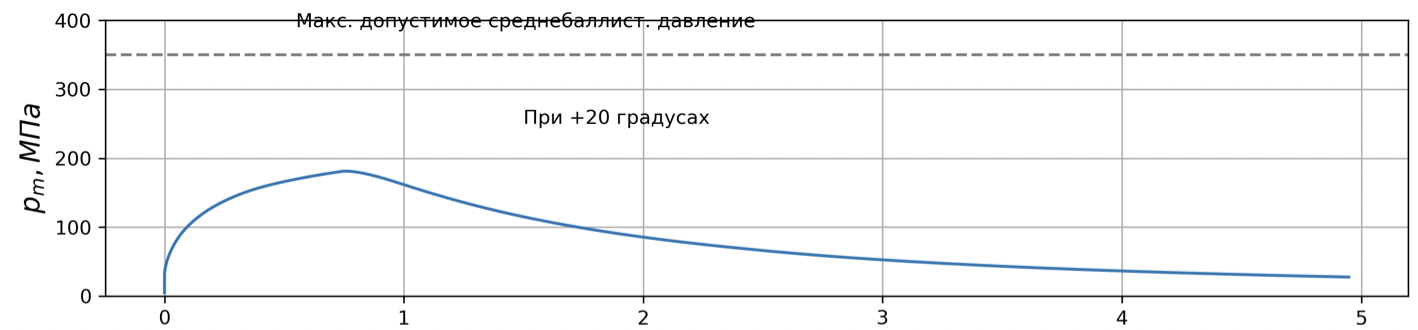
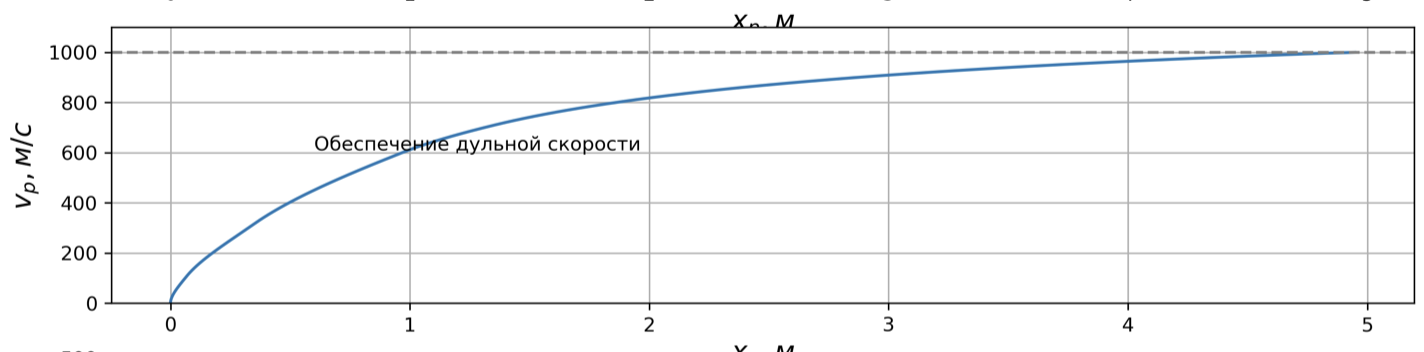


Рис. 2.2. Геометрическое место решений для пороха 6/7 П-5 БП фл

 Рис. 2.3. ограничение по среднебаллистическому давлению.

 Рис. 2.4. ограничение по дульной скорости.

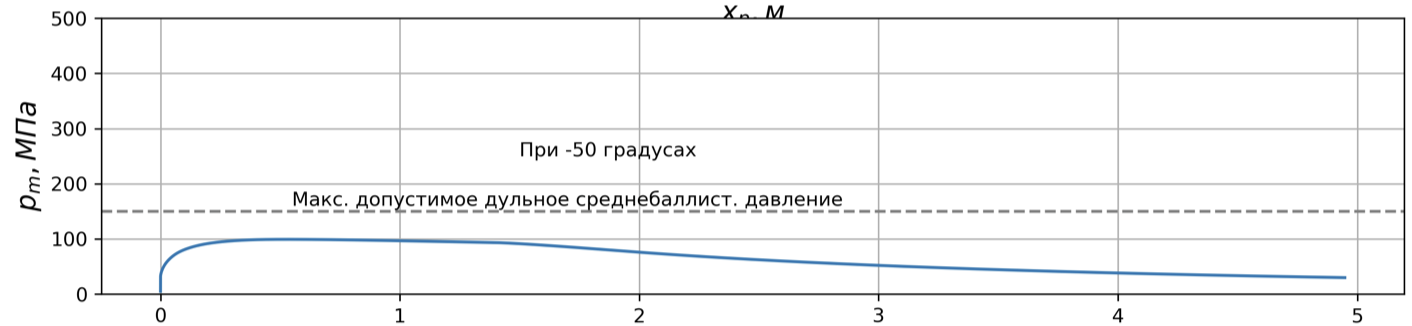


Рис. 2.5. ограничения по дульному давлению.

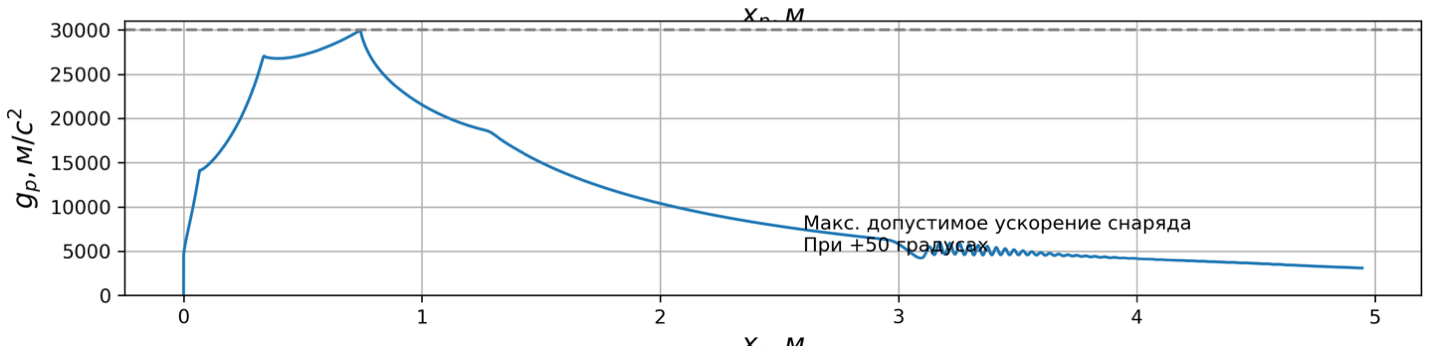


Рис. 2.6. ограничения по ускорению снаряда.

Табл. 2. Сравнение решения с ограничениями

|  |  |  |
| --- | --- | --- |
| Ограничение | Результаты решения | Ограничение |
|  | 181.1920 | 380 |
|  | 64.943 | 65 |
|  | 29931.838 | 30000 |
|  | 29.770 | 150 |

# Вывод

После перебора множества решений для всех порохов, получилось успешно найти решение с заданной точностью, с минимальным объёмом каморы, удовлетворяющее заданным ограничениям.

Снаряд калибра 76,2 мм и массой 3,3 кг развивает дульную скорость в 1000 м/с; Максимальное давление не превышает 380 МПа;  
Ведущая часть имеет длину в 4,949 м;  
При температуре -50 0С среднебаллистическое давление на дульном срезе не

превышает 150 МПа;  
При температуре +50 0С ускорение снаряда составляет 29931.838 м/. Объём каморы равен 0,00151 м3.

# Список использованной литературы

1. Баллистическое проектирование артиллерийских орудий:  
   учебно-методическое пособие / Н. В. Быков, М. С. Товарнов,  
   О. С. Серпинский: под общей редакцией Н. В. Быкова. – Москва: издательство МГТУ им. Н. Э. Баумана, 2021. – 93 c.: ил;
2. pyballistics PyPl [Электронный ресурс] – режим доступа: https://pypi.org/project/pyballistics/. Дата обращения: 18.10.2022

# Приложение 1

from scipy.optimize import minimize, differential\_evolution

import numpy as np

import pandas as pd

import sys

import os

from matplotlib import cm

from matplotlib.colors import ListedColormap, LinearSegmentedColormap

from matplotlib.patches import Patch

from matplotlib.lines import Line2D

from tqdm.notebook import tqdm

import matplotlib as mpl

import matplotlib.pyplot as plt

mpl.rcParams['figure.figsize'] = (6,4)

import json

from pyballistics import ozvb\_lagrange

from pyballistics import get\_powder\_names

# %config InlineBackend.figure\_format = 'svg'

%matplotlib inline

viridis = cm.get\_cmap('Blues\_r', 1024)

newcolors = viridis(np.linspace(0, 1, 1024))

newcolors[:52, :] = cm.get\_cmap('winter\_r', 52)(np.linspace(0.0, 0.5, 52))

newcmp = ListedColormap(newcolors)

def getPsr(res):

PS = list()

for layer in res['layers']:

xs = layer['x']

x\_c = (xs[:-1]+xs[1:])/2

ps = layer['p']

S = np.sum(0.5\*(x\_c[1:]-x\_c[:-1])\*(ps[1:]+ps[:-1]))

p = S / (x\_c[-1] - x\_c[0])

PS.append(p)

return PS

def full\_calc(otn\_m\_z, delta, porox, n):

q = 3.3

omega = q \* otn\_m\_z

W\_0 = omega/delta

d = 0.0762

n\_S = 1.04

opts = {

'powders': [{'omega': omega, 'dbname': porox}],

'init\_conditions': {

'q': q,

'd': d,

'W\_0': W\_0,

'phi\_1': 1.04,

'p\_0': 30e6,

'n\_S': n\_S

},

'igniter': {'p\_ign\_0': 5e6},

'heat':{'enabled':True, 'heat\_barrel':True},

'windage':{'shock\_wave':True},

'meta\_lagrange': {

'n\_cells': n,

'CFL': 0.7},

'stop\_conditions': {

'steps\_max': 100000,

'x\_p': 65\*d,

'p\_max':380e6,

'v\_p': 1000

}

}

res20 = ozvb\_lagrange(opts)

results = {

'n': n

}

if res20['stop\_reason'] != 'v\_p':

results['criteria'] = 99

return results

opts['init\_conditions']['T\_0'] = 273.15 - 50

opts['stop\_conditions'] = {

'x\_p': res20['layers'][-1]['x'][-1],

'steps\_max': 100000

}

res50 = ozvb\_lagrange(opts)

if res50['stop\_reason'] != 'x\_p':

results['criteria'] = 99

return results

PSR50 = getPsr(res50)

if PSR50[-1] > 150e6:

results['criteria'] = 99

return results

results['criteria'] = W\_0 + res20['layers'][-1]['x'][-1] \* n\_S \* np.pi \* d \*\* 2 /4

return results

def plot\_results(results):

plt.figure(figsize=(6,6), dpi = 600)

x1 = []

y1 = []

x2 = []

y2 = []

x3 = []

y3 = []

for res in results['res1']:

x1.append(res['n'])

y1.append(res['criteria'])

for res in results['res2']:

x2.append(res['n'])

y2.append(res['criteria'])

for res in results['res3']:

x3.append(res['n'])

y3.append(res['criteria'])

plt.plot(x1, y1, label = '$W\_{pm}$ в точке $(\\omega/q, \\Delta)$', zorder=3)

plt.plot(x2, y2, label = '$W\_{pm}$ в точке $(\\omega/q+\\delta\_{\\omega/q}, \\Delta)$', zorder=3)

plt.plot(x3, y3, label = '$W\_{pm}$ в точке $(\\omega/q, \\Delta+\\delta\_{\\Delta})$', zorder=3)

plt.xlabel('Количество ячеек')

plt.ylabel('$W\_{pm}, м^{3}$')

plt.legend()

plt.grid()

plt.show()

porox = '9/7'

results = {}

results1 = []

results2 = []

results3 = []

otn\_m\_z = 0.358801

delta = 779.979866

sh\_po\_wq = 0.002

sh\_po\_delta = 1

n = 10

while n <= 1000:

res1 = full\_calc(otn\_m\_z, delta, porox, n)

results1.append(res1)

n += 10

results['res1'] = results1

otn\_m\_z = otn\_m\_z + sh\_po\_wq

n = 10

while n <= 1000:

res2 = full\_calc(otn\_m\_z, delta, porox, n)

results2.append(res2)

n += 10

results['res2'] = results2

otn\_m\_z = 0.358801

delta = delta + sh\_po\_delta

n = 10

while n <= 1000:

res3 = full\_calc(otn\_m\_z, delta, porox, n)

results3.append(res3)

n += 10

results['res3'] = results3

plot\_results(results)

results

# Приложение 2

from scipy.optimize import minimize, differential\_evolution

import numpy as np

import pandas as pd

import sys

import os

from matplotlib import cm

from matplotlib.colors import ListedColormap, LinearSegmentedColormap

from matplotlib.patches import Patch

from matplotlib.lines import Line2D

from tqdm.notebook import tqdm

import matplotlib as mpl

import matplotlib.pyplot as plt

mpl.rcParams['figure.figsize'] = (6,4)

import json

from pyballistics import ozvb\_lagrange

from pyballistics import get\_powder\_names

# %config InlineBackend.figure\_format = 'svg'

%matplotlib inline

viridis = cm.get\_cmap('Blues\_r', 1024)

newcolors = viridis(np.linspace(0, 1, 1024))

newcolors[:52, :] = cm.get\_cmap('winter\_r', 52)(np.linspace(0.0, 0.5, 52))

newcmp = ListedColormap(newcolors)

styles = {

'p\_max': {'color': 'darkred', 'alpha': 0.3, 'label': 'Превышение макс. давления'},

'x\_p': {'color': 'orange', 'alpha': 0.3, 'label': 'Превышение макс. длины ствола'},

'p\_mz': {'color': 'purple', 'alpha': 0.3, 'label': 'Превышение макс. дульного среднебаллист. давления'},

'Acc': {'color': 'pink', 'alpha': 0.3, 'label': 'Превышение макс. ускорения снаряда'},

'steps\_max': {'color': 'black', 'alpha': 0.3, 'label': 'Максимальное количество шагов'}

}

def getPsr(res):

PS = list()

for layer in res['layers']:

xs = layer['x']

x\_c = (xs[:-1]+xs[1:])/2

ps = layer['p']

S = np.sum(0.5\*(x\_c[1:]-x\_c[:-1])\*(ps[1:]+ps[:-1]))

p = S / (x\_c[-1] - x\_c[0])

PS.append(p)

return PS

def getGs(res):

v\_ps = np.array([lr['u'][-1] for lr in res['layers']])

ts = np.array([lr['t'] for lr in res['layers']])

gs = (v\_ps[1:] - v\_ps[:-1]) / (ts[1:] - ts[:-1])

return np.max(gs) / 9.81

dbname = get\_powder\_names()

#dbname = [ '15/1 тр В/А', '4/7 ПТ-3 фл', '4/1', '4/7 Ц гр', '4/7', '5/7 н/а', '4/7 св']

def get\_rnd(otn\_m\_z\_min, otn\_m\_z\_max, delta\_min, delta\_max, porox, \*\*kwargs):

otn\_m\_z = np.random.uniform(otn\_m\_z\_min, otn\_m\_z\_max)

delta = np.random.uniform(delta\_min, delta\_max)

return full\_calc(otn\_m\_z, delta, porox, \*\*kwargs)

def full\_calc(otn\_m\_z, delta, porox, ret\_all=True):

q = 3.3

omega = q \* otn\_m\_z

W\_0 = omega/delta

d = 0.0762

n\_S = 1.04

opts = {

'powders': [{'omega': omega, 'dbname': porox}],

'init\_conditions': {

'q': q,

'd': d,

'W\_0': W\_0,

'phi\_1': 1.04,

'p\_0': 30e6,

'n\_S': n\_S

},

'igniter': {'p\_ign\_0': 5e6},

'heat':{'enabled':True, 'heat\_barrel':True},

'windage':{'shock\_wave':True},

'meta\_lagrange': {

'n\_cells': 400,

'CFL': 0.7},

'stop\_conditions': {

'steps\_max': 100000,

'x\_p': 65\*d,

'p\_max':380e6,

'v\_p': 1000

}

}

print(opts)

res20 = ozvb\_lagrange(opts)

results = {

'args': {

'otn\_m\_z': otn\_m\_z,

'delta': delta,

'pname': porox,

'ret\_all': ret\_all }

}

if ret\_all:

results['res20'] = res20

if res20['stop\_reason'] != 'v\_p':

results['isgood'] = False

results['label'] = res20['stop\_reason']

results['criteria'] = 99

return results

#Ограничение 1

opts['init\_conditions']['T\_0'] = 273.15 - 50

opts['stop\_conditions'] = {

'x\_p': res20['layers'][-1]['x'][-1],

'steps\_max': 100000

}

res\_50 = ozvb\_lagrange(opts)

if ret\_all:

results['res\_50'] = res\_50

if res\_50['stop\_reason'] != 'x\_p':

results['isgood'] = False

results['label'] = res\_50['stop\_reason']

results['criteria'] = 99

return results

PSR50 = getPsr(res\_50)

if PSR50[-1] > 150e6:

results['isgood'] = False

results['label'] = 'p\_mz'

results['criteria'] = 99

return results

# Ограничениe 2

opts['init\_conditions']['T\_0'] = 273.15 + 50

opts['stop\_conditions'] = {

'x\_p': res20['layers'][-1]['x'][-1],

'steps\_max': 100000

}

res50 = ozvb\_lagrange(opts)

if ret\_all:

results['res50'] = res50

if res50['stop\_reason'] != 'x\_p':

results['isgood'] = False

results['label'] = res50['stop\_reason']

results['criteria'] = 99

return results

Acc = getGs(res50)

if Acc > 30000:

results['isgood'] = False

results['label'] = 'Acc'

results['criteria'] = 99

return results

results['isgood'] = True

results['label'] = 'good'

results['criteria'] = W\_0

return results

def plot\_results(results):

global z

plt.figure(figsize=(9,6), dpi = 300)

groups = {}

for res in results:

label = res['label']

if label not in groups:

groups[label] = []

groups[label].append(res)

for label, group in groups.items():

xs, ys, crits = [], [], []

for res in group:

xs.append(res['args']['otn\_m\_z'])

ys.append(res['args']['delta'])

crits.append(res['criteria'])

if label == 'good':

plt.scatter(xs, ys, c=crits, label=label, cmap=newcmp, zorder=3)

cb = plt.colorbar()

cb.set\_label('$W\_{pm}$')

i\_best = np.argmin(crits)

file = {'Порох': porox, 'Объем': crits[i\_best],'Отн.масса заряда': xs[i\_best],'Плотность заряжания': ys[i\_best]}

z.append(file)

print('Относительная масса заряда: ', xs[i\_best],'Плотность заряжания: ', ys[i\_best], 'Наименьший объем камеры: ', crits[i\_best])

plt.scatter([xs[i\_best]], [ys[i\_best]], marker='\*', s=250, color='red',

edgecolor='black', label=f'BEST {crits[i\_best]:.3f}', zorder=4)

else:

plt.scatter(xs, ys, \*\*styles.get(label, {}))

plt.grid()

plt.legend()

plt.show()

def plot\_result(result):

fig = plt.figure(figsize = (12, 12), dpi = 300)

ax1 = fig.add\_subplot(4,1,1)

p20 = []

x\_p = []

vs = []

for layer in result['res20']['layers']:

x\_p.append(layer['x'][-1])

vs.append(layer['u'][-1])

xs = layer['x']

x\_c = (xs[:-1]+xs[1:])/2

ps = layer['p']

S = np.sum(0.5\*(x\_c[1:]-x\_c[:-1])\*(ps[1:]+ps[:-1]))

p = S / (x\_c[-1] - x\_c[0])/1e6

p20.append(p)

ax1.plot(x\_p, p20)

plt.text(0.55, 390, 'Макс. допустимое среднебаллист. давление')

ax1.axhline(350, color='gray', linestyle='--')

#ax1.axhline(150, color='gray', linestyle='--')

plt.text(1.5, 250, 'При +20 градусах')

plt.xlabel('$x\_{p}, м$', fontsize=14)

plt.ylabel('$p\_{m}, МПа$', fontsize=14)

plt.ylim(0, 400)

ax1.grid()

ax2 = fig.add\_subplot(4,1,2)

ax2.plot(x\_p, vs)

plt.xlabel('$x\_{p}, м$', fontsize=14)

plt.ylabel('$v\_{p}, м/с$', fontsize=14)

plt.text(0.6, 610, 'Обеспечение дульной скорости')

plt.ylim(0, 1100)

ax2.axhline(1000 ,color='gray',linestyle='--')

ax2.grid()

ax3 = fig.add\_subplot(4,1,3)

p50 = []

x\_p1 = []

x\_p2 = []

ts = []

vs = []

for layer in result['res\_50']['layers']:

x\_p1.append(layer['x'][-1])

xs = layer['x']

x\_c = (xs[:-1]+xs[1:])/2

ps = layer['p']

S = np.sum(0.5\*(x\_c[1:]-x\_c[:-1])\*(ps[1:]+ps[:-1]))

p = S / (x\_c[-1] - x\_c[0])/1e6

p50.append(p)

ax3.plot(x\_p1, p50)

plt.text(0.55, 160, 'Макс. допустимое дульное среднебаллист. давление')

ax3.axhline(150, color='gray', linestyle='--')

plt.text(1.5, 250, 'При -50 градусах')

plt.xlabel('$x\_{p}, м$', fontsize=14)

plt.ylabel('$p\_{m}, МПа$', fontsize=14)

plt.ylim(0, 500)

ax3.grid()

ax4 = fig.add\_subplot(4,1,4)

for layer in result['res50']['layers']:

vs.append(layer['u'][-1])

x\_p2.append(layer['x'][-1])

ts.append(layer['t'])

g = np.array(vs[1:]) - np.array(vs[:-1])

s = np.array(ts[1:]) - np.array(ts[:-1])

gs = (g) / (s)

ax4.plot(x\_p2[:-1], gs/9.81)

ax4.axhline(30000, color='gray', linestyle='--')

plt.text(2.6, 5000, 'При +50 градусах')

plt.text(2.6, 7500, 'Макс. допустимое ускорение снаряда')

plt.xlabel('$x\_{p}, м$', fontsize=14)

plt.ylabel('$g\_{p}, м/с^{2}$', fontsize=14)

plt.ylim(0, 31000)

ax4.grid()

plt.savefig('показательные графики.png', bbox\_inches='tight')

plt.show()

z = list()

for porox in tqdm(dbname):

results = []

print(porox)

for i in tqdm(range(2000)):

res = get\_rnd(0.01, 1, 50, 1200, porox, ret\_all=False)

results.append(res)

plot\_results(results)

data\_pow = pd.DataFrame(z)

data\_pow.to\_csv('porox',index = False)

df = pd.read\_csv('porox', delimiter = ',')

df

kolvo = len(df['Объем'])

volume = df['Объем'][0]

for i in range(kolvo):

if df['Объем'][i] < volume:

name = df['Порох'][i]

volume = df['Объем'][i]

print(name)

x1\_min = float(df[df['Объем'] == df['Объем'].min()]['Отн.масса заряда'])

y1\_min = float(df[df['Объем'] == df['Объем'].min()]['Плотность заряжания'])

volume = df['Объем'][0]

for i in range(kolvo):

if df['Порох'][i] == '9/7':

print('Лучшее решение')

elif df['Объем'][i] < volume:

name2 = df['Порох'][i]

volume = df['Объем'][i]

otn\_m\_z2 = df['Отн.масса заряда'][i]

delta2 = df['Плотность заряжания'][i]

print(name2)

volume = df['Объем'][0]

for i in range(kolvo):

if df['Порох'][i] == '6/7 П-5 БП фл' or df['Порох'][i] == '9/7':

print('Лучшее решение')

elif df['Объем'][i] < volume:

name3 = df['Порох'][i]

volume = df['Объем'][i]

otn\_m\_z3 = df['Отн.масса заряда'][i]

delta3 = df['Плотность заряжания'][i]

print(name3)

plt.figure(dpi = 600, figsize = (8, 6))

plt.scatter(df['Отн.масса заряда'], df['Плотность заряжания'], c = df['Объем'])

plt.scatter(x1\_min, y1\_min, marker = '\*', s = 200, label = 'Лучшее решение 9/7')

plt.scatter(otn\_m\_z2, delta2, marker = 'o', s = 100, label = 'Второе решение 6/7 П-5 БП фл')

plt.scatter(otn\_m\_z3, delta3, marker = '^', s = 200, label = 'Третье решение 12/7 В/А')

plt.xlabel('$\\omega/q$', fontsize=12)

plt.ylabel('$\\Delta, кг/м^{3}$ ', fontsize=12, rotation = 0)

plt.legend()

plt.colorbar()

plt.grid()

plt.savefig('МДБР.png')

plt.show()

poroxa = []

kolvo = len(df['Объем'])

for i in range(kolvo):

if df['Объем'][i] < 0.002:

poroxa.append(df['Порох'][i])

print(poroxa)

z = list()

for porox in tqdm(poroxa):

results = []

print(porox)

for i in tqdm(range(5000)):

res = get\_rnd(0.2, 0.6, 500, 1000, porox, ret\_all=False)

results.append(res)

plot\_results(results)

data\_pow = pd.DataFrame(z)

data\_pow.to\_csv('better\_porox',index = False)

df = pd.read\_csv('better\_porox', delimiter = ',')

df

z = list()

porox = '6/7 П-5 БП фл'

results = []

for delta in tqdm(range(600, 800, 1)):

otn\_m\_z = 0.3

while otn\_m\_z <= 0.4:

res = full\_calc(otn\_m\_z, delta, porox, ret\_all=False)

otn\_m\_z += 0.002

results.append(res)

plot\_results(results)

result = full\_calc(0.358, 783, '6/7 П-5 БП фл', ret\_all=True)

plot\_result(result)