# ЭТАП 1. Попросить Николая Игоревича создать аккаунт на сайте и на сервере СДЕЛАНО!

# ЭТАП 2. Консультируясь с Алиной и Полиной, установить версию rect на сервере и научиться ею пользоваться удаленно. Алина уже научилась!

**После установки пакета, пожалуйста, из папки rect в терминале выполните следующие команды**

**git pull d0c0fccf3cc230107d1f7341a434fd484bd5a20a**

**./build.sh**

**Так как последняя версия rect сломана.**

**После этого, пожалуйста, ручками замените файл в папке rect/elastic/trainfunction.h на одноименный файл, присланный мною.**

**А файл rect/elastic/src/trainfunction.cpp тоже замените на одноименный файл, присланный мною.**

**Затем снова из папки rect в терминале команда**

**./build.sh**

**Так как мои ползуны и моё гранусловие ещё не выложены в репозиторий.**

При малейших затруднениях, пожалуйста, спрашивайте Алину. По возможности запишите, пожалуйста, набор детальных инструкций. Пригодится. Сделаем методичку-мануал на их основе.

Была проблема – такое-то решение.

Но если не хотите, можно этого не делать.

# ЭТАП 3. Сделать серию запусков с приложенными конфигурационными файлами.

\* В локальной папке, из которой идет запуск, помимо стандартной папки VTK должны быть следующие папки Sxx, Sxy, Syy, V, Vx, Vy, Fr. В каждой из этих папок будет сохранена соответствующая компонента рассчитанного поля. А в папке Fr – разрушения рельса. А в папке VTK несколько начальных срезов модуля скорости

А вот папки H0, H1… которые нужны для генератора, которым пользуется Алина, здесь наоборот не нужны.

Также в расчетной папке должен лежать помимо конфигурационного файла файл impulse.txt.

## Подробнее про сохранение.

Всего делается от 0 до Nt шагов по времени. Сохранить все Nt шагов – не хватит места на диске. Поэтому сохранение устроено следующим образом.

В шаги от 0 до Nt\_soft\_half поезд никуда не едет, происходит плавное возрастание давления, про которое можно почитать в приложенной статье.

В папке VTK сохраняется 0 шаг, save-ый, 2\*save-ый и так несколько. Для тестовых запусков.

В папках Sxx, Sxy, Syy, V, Vx, Vy сохраняются соответствующие величины начиная с шага по времени Nt\_soft\_half по шаг по времени Nt, каждый save-ый шаг по времени.

В папке Fr сохраняются разрушения рельса начиная с шага по времени 0 по шаг по времени Nt, каждый save-ый шаг по времени.

Варьируя параметр save можно уменьшить количество необходимого места на жестком диске. Однако, при этом не получится детально исследовать волновые процессы. Нужно всякий раз искать золотую середину. Если исследуются только разрушения, параметр save может быть очень большим.

## Комментарии для параметров генератора конфигурационных файлов

ВНИМАНИЕ!

Если в файле RZD\_10.py меняется n\_x\_impulse, то надо сделать следующее.

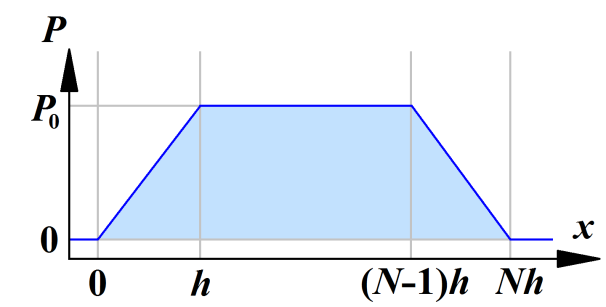
\* Удалить файл impulse.txt

\* заменить n\_x\_impulse на новый в файле Impulse\_Generator\_8.py

\* выполнить в терминале команду

*python Impulse\_Generator\_8.py >> impulse.txt*

Должен появиться новый файл impulse.txt. Он содержит форму импульса, про которую раздел в статье.



На рисунке N – это то самое n\_x\_impulse. Можно прямо открыть файл impulse.txt екселем и построить график, чтобы его увидеть.

Остальные комментарии параметров даны прямо в файле генераторе конфигурационных файлов RZD\_10.py.

## План исследований

предлагаю использовать гибкий (корректируется по мере выполнения) по двум основным направлениям.

Направление исследований Насти №1.

просто давление разных поездов на рельс (Настя, это Вам задание. У Вас давление одного вагона. **А Антон будет делать давление всего состава. Антон, сейчас колеса четыре, а Вам надо сделать, чтобы их было N\*4. Формулы как в приложенной работе, и они «сидят» в файле rect/elastic/src/trainfunction.cpp Не разберетесь за 5 минут - спросите**).

Направление исследований Насти №2.

исследования при наличии ползунов. Почитать, какие сейчас рекомендации по транспортировке вагонов, проверить, как они выполняются с нашими расчетами при типичных условиях. Проверить, какие условия влияют.

# Вообще, план добавки ползуна такой

1. Антон дописывает N вагонов БЕЗ ПОЛЗУНОВ и тестит запуск на кластере, масштабируемость и т.д.

2. Настя тестит один единственный ползун и один единственный вагон/локомотив.

3. Я (консультируясь с Колей) добавляю ползун в N вагонов Антона с заданием произвольного количества ползунов и произвольного номера колес.

4. Окончательное тестирование.

5. Вывесить в репозиторий.

Сейчас расчет проведен

\* для ползуна 2 мм и скорости 15 км/ч,

\* для ползуна 2 мм и скорости 120 км/ч

\* без ползуна и скорости 120 км/ч

Остальные параметры заданы в высланном Вами файле RZD\_10.py

**Начать можно с выполнения расчета для ползуна 1 мм и скорости 120 км/ч. Посмотреть, в чём будут отличия в разрушениях.**

**Вообще, Вы план исследований составьте и со мною согласуйте. Если хотите – я сама Вам его составлю. Просто мне кажется, самой интересней.**

# Разработка механико-математических моделей (выполняется независимо от запусков)

Оба типа исследований делаются с разными… ползунами, скоростью, массой вагона/локомотива, шпалами, расстоянием между шпалами, в дождь и в сухую погоду (в дождь меняются акустические свойства гравия под насыпью) и т.д.

Ещё сейчас удар ползуном приходится на пространство между шпалами. Интересно посмотреть, будет ли разница, если удар придется на рельс, который непосредственно над шпалой.

*ВАЖНО! Я бы сама, если честно, несколько расчетов провела. Больно любопытно, что же получится. Поэтому нам с Вами надо как-то разделить план исследований, чтобы считать разное. Я о запущенных мною постановках буду информировать Вас на почту. И Вы, пожалуйста, делайте также. Для этого надо скинуть файл RZD\_10.py, который Вами запущен, или просто перечислить параметры из него, как я сделала выше. Можно менять этим файлам имена. Главное в командочке*

*python БЫЛО\_RZD\_10\_А\_ТАК\_ЛЮБОЕ\_ИМЯ\_ФАЙЛА.py >> ТОЖЕ\_ЛЮБОЕ\_ИМЯ\_ФАЙЛА.conf*

*./run.sh ТОЖЕ\_ЛЮБОЕ\_ИМЯ\_ФАЙЛА.conf*

*писать правильные имена, и одинаковое имя конфигурационного файла использовать. И вот этот самый*

*БЫЛО\_RZD\_10\_А\_ТАК\_ЛЮБОЕ\_ИМЯ\_ФАЙЛА.py содержит все параметры расчёта.*

*Да, под Windows файлы, созданные под Linux, надо открывать не блокнотом, а Notepad++*

[*https://notepad-plus-plus.org/*](https://notepad-plus-plus.org/)

Им же открываются файлы .py, .conf, .h, .cpp под Windows.

В общем, вначале разработать ряд типовых постановок, диапазонов варьируемых величин и их физического смысла. Можно консультироваться с Вашим отцом.

Предлагаю параллельно описывать каждую характерную механико-математическую постановку, как сделано в статье, которую Вы переводили. Формат произвольный. Пойдет в Вашу диссертацию и в отчёт по гранту за следующий год.

Затем для этих постановок начинаются вычисления.

Каждый расчёт может длиться около 2-3 суток, так что параллельно можно делать другую работу. Отрисовка каждого расчета тоже может занять 2-3 суток. Делается в программе paraview. Объем расчетных данных достаточно велик. Уменьшить его количество можно увеличивая save. Можно ради скорости сохранять и рисовать только картинку с разрушениями. Это займет пару минут. Отрисовывать тоже можно только разрушения. Однако, надо следить, чтобы за время расчёта удар ползуном успел бы произойти. Для разной скорости движения поезда нужно разное Nt. Выполняется на основе формул из статьи. Реальное время найдется как (Nt-1)\*dt. Кстати, величина Nt-1-Nt\_soft\_half должна быть кратна параметру save, иначе смысла не имеет считать «лишние» шаги по времени, всё равно результаты не сохранятся.

Поэтому прежде чем запускать расчёты, согласуйте план исследований со мною. Я проверю, все ли параметры выбраны корректно.

Если Вам пока что сложно составить план исследования самостоятельно - помогу.