

# Уравнения продуктивности системы взаимовлияющих скважин в резервуаре-секторе на установившемся режиме

Представленный модуль по расчету продуктивности системы взаимовлияющих скважин в резервуаре-секторе на установившемся режиме был разработан для использования уравнений и зависимостей полученных в ходе исследования “Jing Lu, Shawket Ghedan, and Tao Zhu, The Petroleum Institute; and Anh Dinh, Schlumberger; and Djebbar Tiab, University of Oklahoma” описанных в соответствующей SPE статье на данную тему.

Для конкретной информации касающейся методики расчета, уравнений или их применимости, просьба обращаться к первоисточнику (статье).

## ВНИМАНИЕ

Для корректной работы приложений следует использовать **Python 3.9**.

## Обозначения величин

- $P_e$  ( $P_e$ ) – давление на границах пласта,  $MPa$ ;
- $\Phi$  ( $\Phi$ ) – угол раскрытия сектора, *границы* :  $[0, 360]$ ,  $^{\circ}(deg)$ ;
- $Q$  ( $Q$ ) – дебит скважины,  $m^3/d$ ;
- $r_e$  ( $r_e$ ) – радиус резервуара-сектора,  $m$ ;
- $H$  – эффективная толщина пласта,  $m$ ;
- $K_r$  ( $K_r$ ) – радиальная проницаемость,  $\mu m^2$ ;
- $K_z$  ( $K_z$ ) – вертикальная проницаемость,  $\mu m^2$ ;
- $\mu$  ( $\mu$ ) – вязкость нефти,  $mPa \cdot s$ ;
- $B$  – объемный коэффициент нефти,  $m^3/m^3$ ;
- $p_w f$  ( $P_{wf}$ ) – забойное давление скважины,  $MPa$ ;
- $r_i$  ( $r_i$ ) – радиус вектор расположения скважины от центра сектора,  $m$ ;
- $\varphi$  ( $\phi_i$ ) – угол расположения скважины,  $^{\circ}(deg)$ ;
- $r_w$  ( $r_w$ ) – радиус скважины,  $m$ ;
- $S$  ( $skin$ ) – скин-фактор скважины.

## Использованные в методике допущения

1. В начальный момент времени  $t = 0$ , пластовое давление равномерно распределено по всему резервуару и равно начальному  $P_{ini}$ ;
2. Все скважины параллельны вертикальной оси  $z$  и их эффективная мощность равна эффективной толщине пласта, также нижние границы пласта полностью не проницаемые;
3. Подразумевается движение однофазной среды с небольшой и постоянной сжимаемостью  $C_f$  и постоянной вязкостью  $\mu$ , а также постоянным объемным коэффициентом  $B$ . Свойства жидкости не зависят от давления. Силы тяжести пренебрежимо малы;
4. Система скважин находится и резервуаре формы сектора с постоянной горизонтальной проницаемостью  $K_h$ , вертикальной проницаемостью  $K_z$  и эффективной толщиной  $H$ . Горизонтальная проницаемость  $K_h$  равна радиальной проницаемости  $K_r$ . На границах резервуара-сектора принято постоянное давление,  $P_e = P_{ini}$ , и границы пласта не проницаемые.
5. В каждый отдельный промежуток времени принято постоянные : кол-во скважин, их радиус и механический скин-фактор.

## Описание файлов и их применения

### Библиотека функций

- **ssp\_eq\_mws\_sfr.py** :  
Питон файл в котором применены все описанные в статье уравнения необходимые для расчета.  
Их правильность была проверена на данных из статьи и представленных в ней аналитических и численных решений.  
Все вычисления проводятся в **единицах СИ**.

### .exe файлы

В папке **dist** находятся файлы с названиями, соответствующим python-файлам, с идентичным функционалом, но с возможностью запуска на любой Windows системе (для этого требуется только один .exe файл).

**Замечание** : довольно долгий запуск приложений (5-10 сек). В приложении с графическим интерфейсом отсутствуют картинки.

### Расчетный модуль с графическим интерфейсом

- **calculaion\_app\_gui.py** :  
Главный файл модуля в котором реализован весь функционал приложения. Разрабатывался максимально понятным с целью упрощения работы с ним.  
По умолчанию все ячейки заполнены значениями для **“Example One, Case 1”** из статьи SPE.

### Описание функционала кнопок

- **find Q** - устанавливает режим расчета в том случае когда известен *скин-фактор* каждой скважины и высчитывает расход ( $Q, m^3/d$ ) и суммарный дебит по всем скважинам;
- **find S** - устанавливает режим расчета в том случае когда известны дебиты скважин ( $Q, m^3/d$ ) и высчитывает их *скин-фактор* ;
- **+**, **-** - кнопка добавления/удаления скважины;
- **Calculate** - производит сам расчет;
- **Save results** - создает Excel-файл с именем 'manual\_save\_test.xlsx' в который записывается все входные и выходные данные **последнего** проведенного расчета в папку в которой была запущена программа;
- **Calculate from excel** - реализует функционал файла 'cmd\_only\_excel\_mode.py' (описание см. ниже) и сохраняет результат в Excel-файл с именем 'output\_solution\_example.xlsx' в той папке в которой была запущена программа.

## Ошибки

Все **распространенные** ошибки (неправильный разделитель десятичных знаков (*правильный* - точка : .), деление на ноль, неправильный путь к файлу и т.д.) были учтены и вынесены в '*исключения*', чтобы уберечь приложения от неожиданных сбоев.

В случае возникновения одной из ошибок соответствующее сообщение будет выведено в командную строку.

## Расчет Excel-файла из командной строки

- **cmd\_only\_excel\_mode.py** :

Для ускорения расчетов можно избежать запуска полноценного приложения и обойтись только взаимодействием с командной строкой. На вход подается только путь соответствующего Excel-файла, который требуется рассчитать.

Результат будет сохранен в Excel-файл с именем 'cmd\_solution.xlsx' в той же папке, откуда был запущен скрипт.

## Шаблоны Excel-файлов

### Шаблон для входного расчетного файла

- **input\_data\_template.xlsx** :

Данный шаблон позволяет рассчитать разом необходимое кол-во '*кейсов*'.

Входной файл необходимо формировать **точно как в шаблоне** для того, чтобы расчет прошел корректно.

- Для каждого '*кейса*' должны быть определены его границы путем написания метки 'START' и 'END' в соответствующих местах (см. input\_data\_template.xlsx);
- Также должны присутствовать метки данных для резервуара 'RESV' и для скважин 'WELL';
- Заголовки свойств менять не следует, также как и их порядок. Кроме колонки 'SKIN', ее можно заменить на 'Q', соответствующая метка будет определять тип данных и тип расчета;
- Не допускается наличие пустых строк между метками 'START' и 'END', в то время как между '*кейсами*' может быть все что угодно (лучше чтобы не было);
- Кол-во скважин может быть сколь угодно большим.

### Пути к файлам

Если запускаемый скрипт и файл находятся в одной папке, то можно указывать только его имя с расширением. В противном случае следует указывать абсолютный путь (рекомендуется делать так всегда).

### Выходные файлы

Файлы созданные в процессе сохранения результатов расчета формируются автоматически и сохраняются в той же папке откуда был запущен соответствующий скрипт. Если они получились пустыми, то значит в процессе расчета пошло что-то не так и надо проверить или шаблон входных данных или см. пук **Ошибки**.

- **output\_solution\_example.xlsx**
- **manual\_save\_test.xlsx**
- **cmd\_solution.xlsx**

**Замечание** : результаты всегда сохраняются с одним и тем же именем, а также в них записывается информация только о последнем проведенном расчете, это значит, что они могут перезаписываться.

## Дополнительно

- папка **images** : папка с картинками используемыми в приложении;
- **calculation\_app\_gui.ui** : файл интерфейса, созданного в PyQt5 Designer;
- **requirements.txt** : список требований к используемым библиотекам.

## Опечатки найденные в статье

- в **формуле (22)** степень <sup>1/2</sup> ошибочная, правильная степень : <sup>1</sup>.