

## Содержание

1. Область применения . . . . .	5
2. Основные компоненты и описание операций . . . . .	6
3. Диаграмма последовательности . . . . .	10
4. Протокол работы . . . . .	10
5. Аварийные ситуации . . . . .	11
6. Верификация модели . . . . .	11
7. Выходные данные . . . . .	12

## **1. Область применения**

Данный программный продукт - является системой для имитационного моделирования диспетчеризации процессов, целью которой является подготовка данных для дальнейшей сравнительной оценки нескольких систем. Фактором систем является алгоритм планирования.

## 2. Основные компоненты и описание операций

При запуске программы открывается главное меню Рисунок 1.

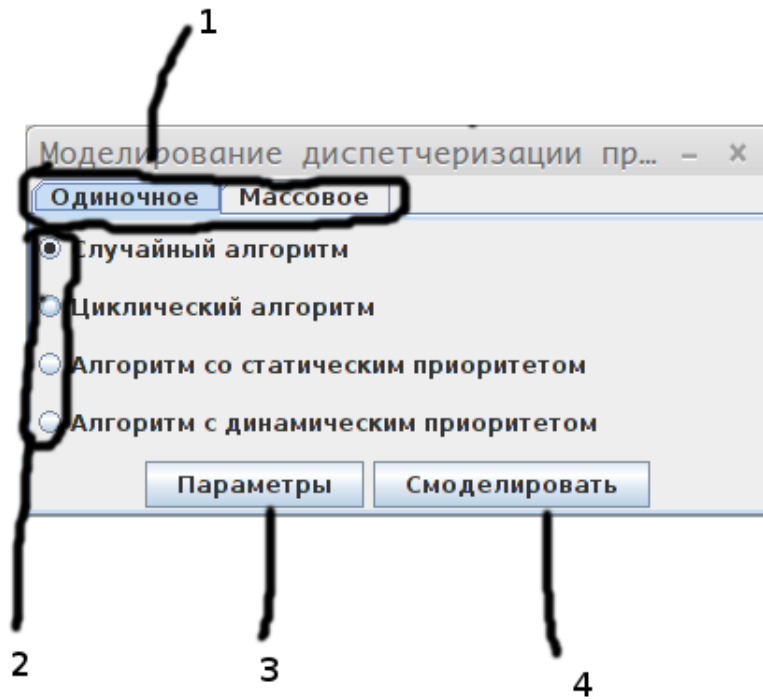


Рисунок 1. Главное меню

1. Режимы моделирования:
  - Одиночное - моделирование один раз, только для одного выбранного алгоритма
  - Массовое - моделирование указанное количество раз, для выбранных алгоритмов
2. Панель для выбора алгоритма(-ов, в случае массового) планирования, для которого будет произведено моделирование.
3. Кнопка вызова окна для задания параметров моделирования Рисунок 2.
4. Кнопка для запуска моделирования

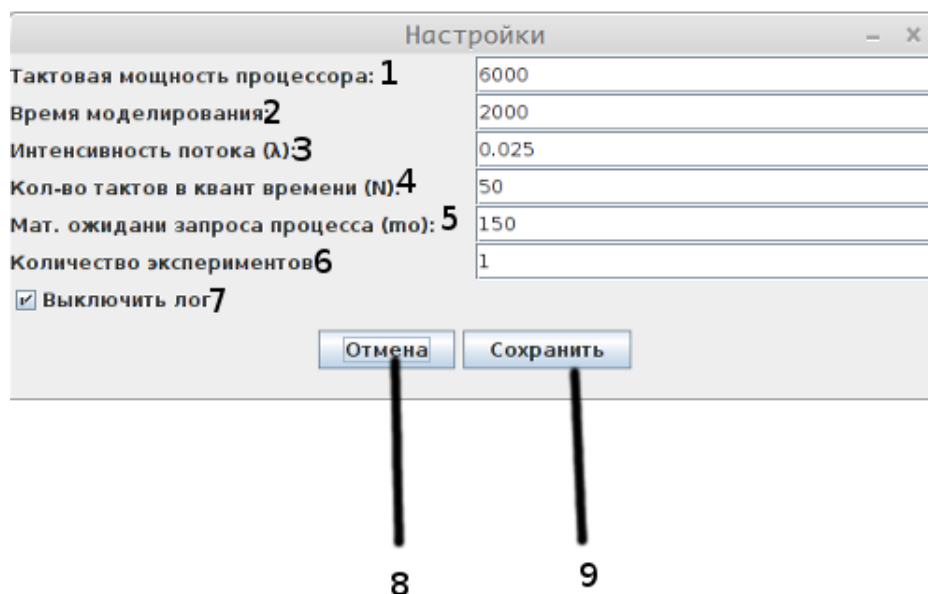


Рисунок 2. Параметры

1. Тактовая мощность процессора ИМ (такты)
2. Время моделирования (секунды)
3. Интенсивность простого потока событий,  $\lambda^{-1}$  - примерное количество одновременно запущенных процессов.
4. Количество тактов в квант времени, через которое происходит прерывание и опрос алгоритма диспетчеризации
5. Математическое ожидание запроса процесса(по нормальному распределению)
6. Количество экспериментов, которое следует провести для выбранных алгоритма.
7. Флажок для включения/выключения лога работы программы
8. Закреть окно без сохранения
9. Сохранить параметры моделирования

По нажатию на кнопку «Смоделировать» появляется индикатор выполнения моделирования Рисунок 3.

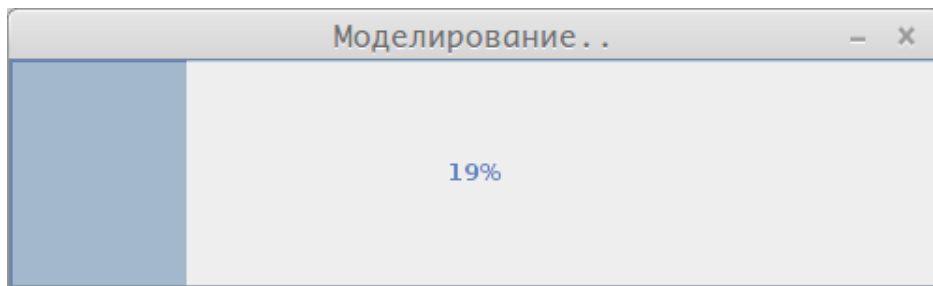


Рисунок 3. Индикатор выполнения

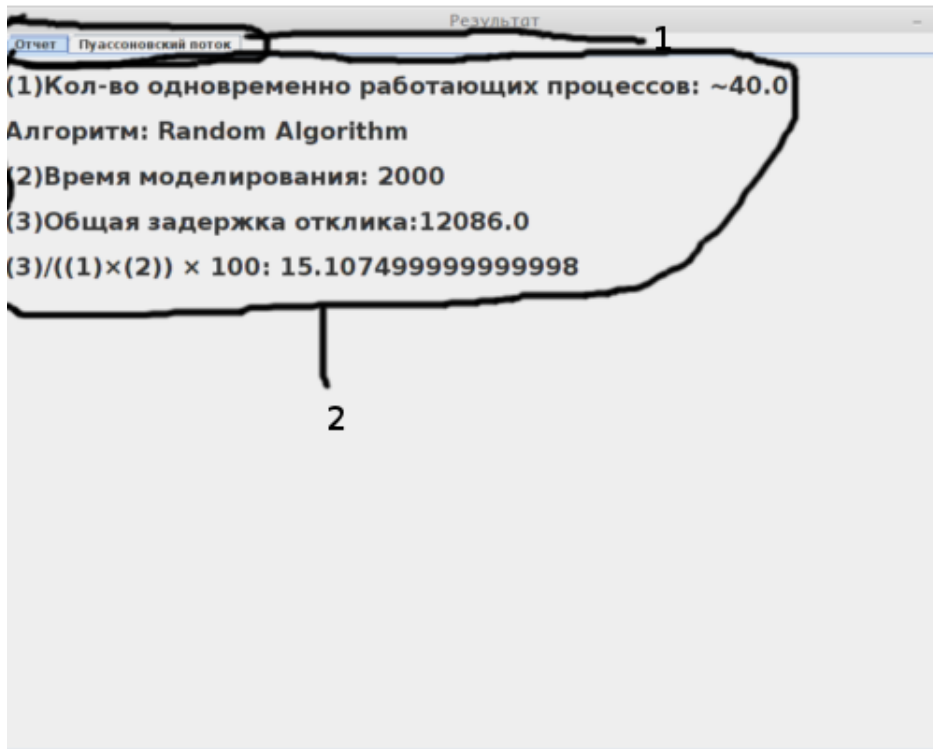


Рисунок 4. Результат одиночного моделирования

1. Панель навигации
2. Результаты моделирования

### Пуассоновский поток

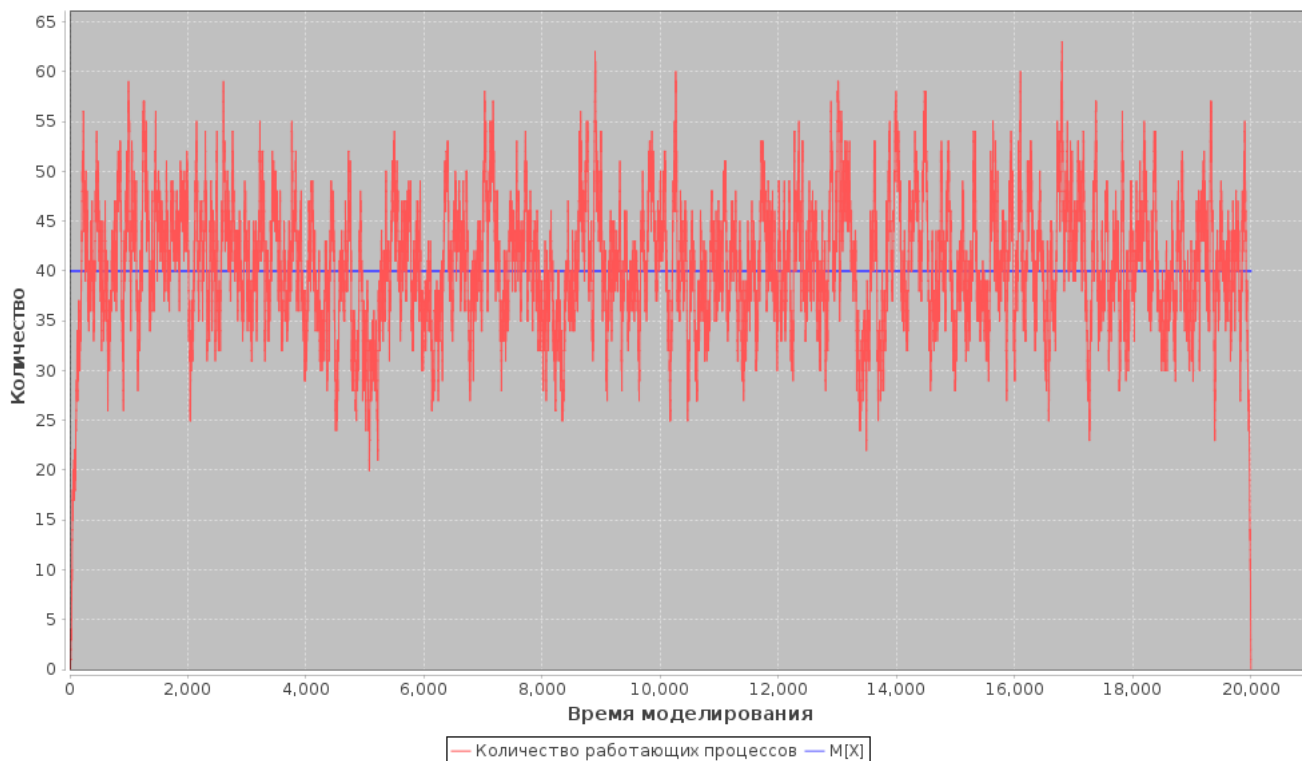


Рисунок 5. График простого потока при одиночном моделировании

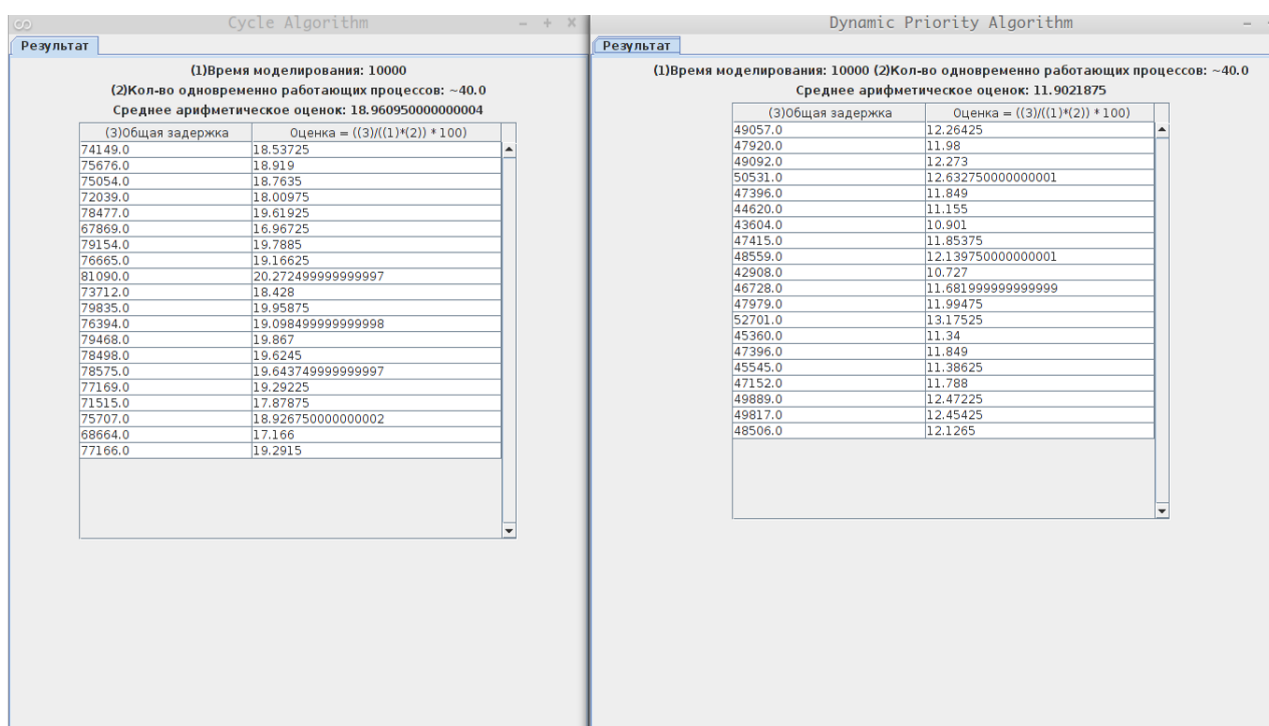


Рисунок 6. Результаты моделирования 20 опытов для 2 алгоритмов, при условии нагруженной системы.

### 3. Диаграмма последовательности

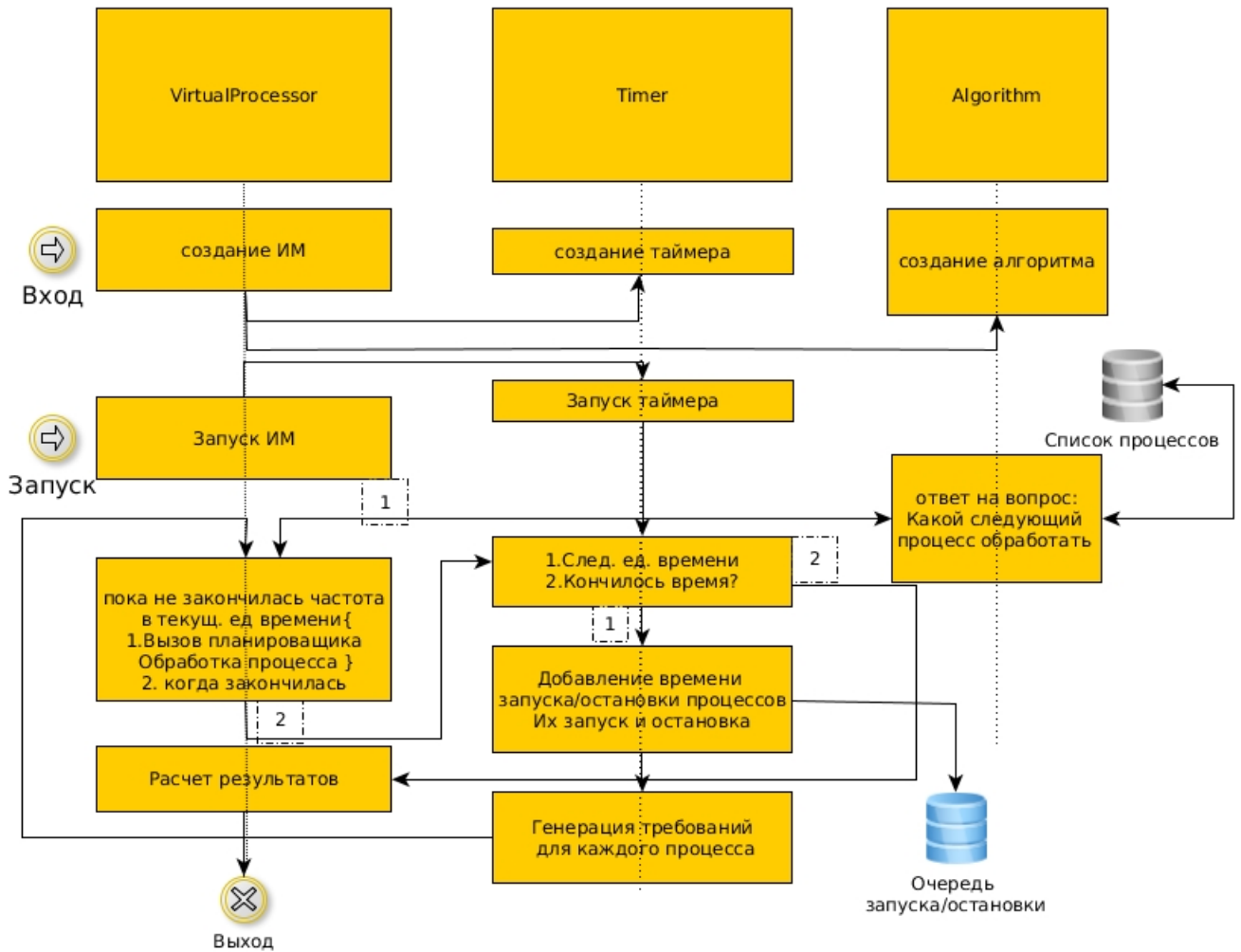


Рисунок 7. Диаграмма последовательности

### 4. Протокол работы

События заносимые в протокол работы:

- «Second  $x$ :», где  $x$  исполняемая секунда модельного времени
- «Добавлен процесс в очередь запуска/остановки:  $start - stop$ », где  $start$  - время запуска,  $stop$  - время остановки
- «Количество запущенных процессов =  $total$ », где  $total$  - количество запущенных процессов в данный момент времени
- «Запущен процесс??», где ?? - уникальный номер процесса
- «Для процесс?? сгенерировано требование в  $s$  тактов»,  $s = Normal(mo, \sigma)$

- «Процесс?? предоставлено  $cur$  тактов |  $s = s - cur$ », где  $cur \leq N$  (квант времени; меньше в том случае, если  $s < N$ , процесс ушел в прерывание или кончились такты процессора на тек. секунду ( $localSpeed$ ))
- «Текущее количество оставшихся тактов процессора на данную секунду  $localSpeed$ , цена за переключение между процессами  $Uniform(1; 10)$ »
- «Процесс?? ушел в блокировку на  $iter$  тактов»
- «Процесс?? вышел из блокировки», если следом процесс уходит в блокировку на  $iter2$  считается что он ушел один раз на  $iter + iter2$ , можно обобщить на  $iter + iter2 + \dots$
- «процессор проработал вхолостую  $n$  тактов» - если остались только необработанными процессы, которые в блокировке, процессор прорабатывает в холостую  $n$  тактов до ближайшего выхода из блокировки с точностью до десятка
- «Процессор отработал одну секунду» - когда закончились такты на эту секунду, либо обработаны все процессы.
- «у процесс?? произошла задержка отклика на  $t$ »,  $t$  = время когда процессор выполнил требование -(минус) время когда требование было сгенерировано

## 5. Аварийные ситуации

Ввод параметров проверяется на корректность. В случаях неверного ввода, значения сбрасываются на стандартные. Запись в лог существенно снижает скорость моделирования. При увеличении времени моделирования рекомендуется не включать лог (из-за детализированности и большого количества операций он будет занимать большие объемы памяти).

## 6. Верификация модели

Верификация проведена с помощью проверки на корректность протокола имитационного моделирования.



## 7. Выходные данные

Поскольку цель программы в относительной оценке, а не в получении абсолютных значений, оценкой производительности установлена

$$\frac{d * \lambda}{t} * 100$$

, где  $d$  - общая(суммарная) задержка всех процессов,  $t$  - время моделирования,  $\lambda$  - интенсивность потока событий.