

Попков В. Е., магистрант, Московский государственный технический университет им. Н.Э. Баумана

Чекулина М. Ю., магистрант, Московский государственный технический университет им. Н.Э. Баумана

АНАЛИТИЧЕСКОЕ И ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ КОЛИЧЕСТВЕННОГО СОСТАВА ОБСЛУЖИВАЮЩИХ СЛУЖБ

Аннотация: В данной статье представлены различные методы расчета количественного состава персонала различных служб. На практике проведен аналитический и имитационный анализ нагрузки и необходимого состава ремонтных служб, сравнение результатов двух методов для подтверждения результатов. Приведены формулы расчета аналитической модели и код для программной реализации имитационной части. Данное решение актуально почти для всей сферы обслуживания и не привязано к предметной области, что повышает актуальность статьи.

Ключевые слова: аналитическая модель, имитационная модель, ремонтные службы, gpss, golang.

Annotation: This article presents various methods for calculating the quantitative composition of personnel in various services. In practice, an analytical and simulation analysis of the load and the required composition of repair services, comparing the results of two methods to confirm the results. Formulas for calculating the analytical model and code for software implementation of the simulation part are given. This solution is relevant for almost the entire service sector and is not linked to the subject area, which increases the relevance of the article.

Key words: analytical model, simulation model, repair services, gpss, golang.

Введение

В наше время компьютеризация занимает значительную роль в развитии организаций любого рода. Без персональных компьютеров(ПК) сложно представить работу бухгалтеров и экономистов, контроль над производством со стороны главных инженеров. Однако компьютеризация предусматривает не только ПК. Это новые оргтехника, специализированные комплексы средств, а также сервера, консоли, усложнение СКС и огромный перечень необходимых расходных материалов (кабели, перемычки, розетки). Таким образом число применяемых вычислительных машин постоянно растет, а в месте с этим растет и зависимость от их работоспособности. Организация будет нести большие потери при длительном простое неисправного оборудования. В связи с этим наличие специальных ремонтных служб является необходимостью для успешной работы любого предприятия.

Однако если необходимость наличия ремонтных служб это реальный факт, то её состав является серьезным вопросом. Сколько требуется специалистов для обслуживания офиса с парком в 50 вычислительных машин? Как минимизировать потери от простоя техники? Поскольку сотрудники ремонтных служб так же требуют зарплаты, потери от содержания большого штата могут даже превысить потери от некоторого простоя. Поэтому данный вопрос является актуальным и важным для любой организации.

Аналитическое моделирование

Аналитическое моделирование - это мощный инструмент, позволяющий здраво оценить спроектированную или уже работающую систему на предмет оптимальности распределения ресурсов, в том числе и людских.

Моделирование подразумевает под собой анализ и расчет характеристик систем массового обслуживания (СМО) на основе неких вполне определенных для каждой системы базовых характеристик [2]. Целью написания данной статьи является обзор и применение аналитического моделирования, а также практическое его применение.

Таким образом можно сформулировать следующую задачу: "Необходимо рассчитать рациональный состав ремонтных служб, на основании размеров парка вычислительных машин и нагрузки". Решаться данная задача может посредством аналитического моделирования и перебором нескольких вариантов составов ремонтных служб.

Возьмем в качестве предметной области службы ремонта - необходимый аспект любой современной организации, поскольку офисная техника, от телефонов и факсов до персональных компьютеров и серверов имеет свойство выходить из строя. Задачами данной службы является поддержка аппаратуры в рабочем состоянии и ремонт в случае поломки. Задачей моделирования же будет расчет оптимального количества ремонтников, при котором будут достигнуты удовлетворяющие характеристики простоя техники в неисправном состоянии и нагрузки на персонал вкупе с минимизацией затрат и потерь. На практике это сведется к задаче расчета характеристик модели при $X1/X2/X3$ числе работников и выбор наилучшего варианта.

Для оценки характеристик функционирования рассматриваемой замкнутой СМО $M/M/c/N/PPPO/N$ следует использовать аналитические выражения, которые известны в ТМО как аналитическая модель ремонтника.

Определяем вероятности состояний рассматриваемой замкнутой СМО:

$$P_0 = \left[\sum_{k=0}^c \frac{N! \cdot \Psi^k}{k! \cdot (N-k)!} + \sum_{k=c+1}^N \frac{N! \cdot \Psi^k}{c^{k-c} \cdot c! \cdot (N-k)!} \right]^{-1} \quad (1)$$

$$P_k = \frac{N! \cdot \Psi^k}{k! \cdot (N-k)!} * P_0, \quad 1 \leq k \leq c \quad (2)$$

$$P_k = \frac{N! \cdot \Psi^k}{c^{k-c} \cdot c! \cdot (N-k)!} * P_0, \quad c < k \leq N \quad (3)$$

Определяем Q - среднее количество компьютеров, находящихся в очереди на ремонт:

$$Q = \sum_{k=c}^N (k - c) * P_k \quad (4)$$

Определяем L - среднее количество компьютеров, находящихся в неисправном состоянии, т.е в очереди на ремонт и на ремонте:

(5)

$$L = \sum_{k=1}^N k * P_k$$

Определяем U – среднее количество компьютеров, которое непосредственно - ремонтируется специалистами:

$$U = L - Q \quad (6)$$

Определяем ρ_0 – коэффициент загрузки одного специалиста, занятого ремонтом компьютеров:

$$\rho_0 = U/C \quad (7)$$

Определяем T_p – среднее время пребывания компьютера в неисправном состоянии (в очереди на ремонт и ремонте):

$$T_p = \frac{L * t_{но}}{N-L} \quad (8)$$

Определяем W – среднее время нахождения компьютера в очереди на ремонт:

$$W = T_p - t_o \quad (9)$$

Определяем $T_{ц}$ – среднее время цикла для компьютера:

$$T_{ц} = T_p + t_{но} \quad (10)$$

Определяем ρ_e – коэффициент загрузки компьютера, т.е. долю времени, в течение которого он находится в исправном состоянии:

$$\rho_e = \frac{t_{но}}{T_{ц}} \quad (11)$$

Определяем n – среднее количество исправных компьютеров:

$$n = N - L \quad (12)$$

Определяем режим работы службы ремонта и обслуживания компьютеров: $\frac{\rho_e}{\rho_0}$

Убытки организации при i -м варианте организации работы службы ремонта компьютеров определяются по формуле 13:

$$Y_i = m_i * S_1 + L_i * S \quad (13)$$

где m_i – количество специалистов, занятых ремонтом компьютеров при этом варианте организации работы службы ремонта компьютеров.

При этом наилучший вариант (d) организации работы службы ремонта компьютеров определяется по формуле 14:

$$Y_d = \min_i Y_i. \quad (14)$$

Разберем предложенный метод моделирования на примере.

Логика работы ремонтной службы представлена на рисунке 1. ОА - обслуживаемая аппаратура, работающая параллельно и работающая в течении некоего среднего времени $t_{\text{наработки_на_отказ}}$ до поломки. Неисправная аппаратура попадает в очередь на ремонт. По прохождению очереди она чинится за время t_0 . После ремонта аппаратура снова включается в работу.

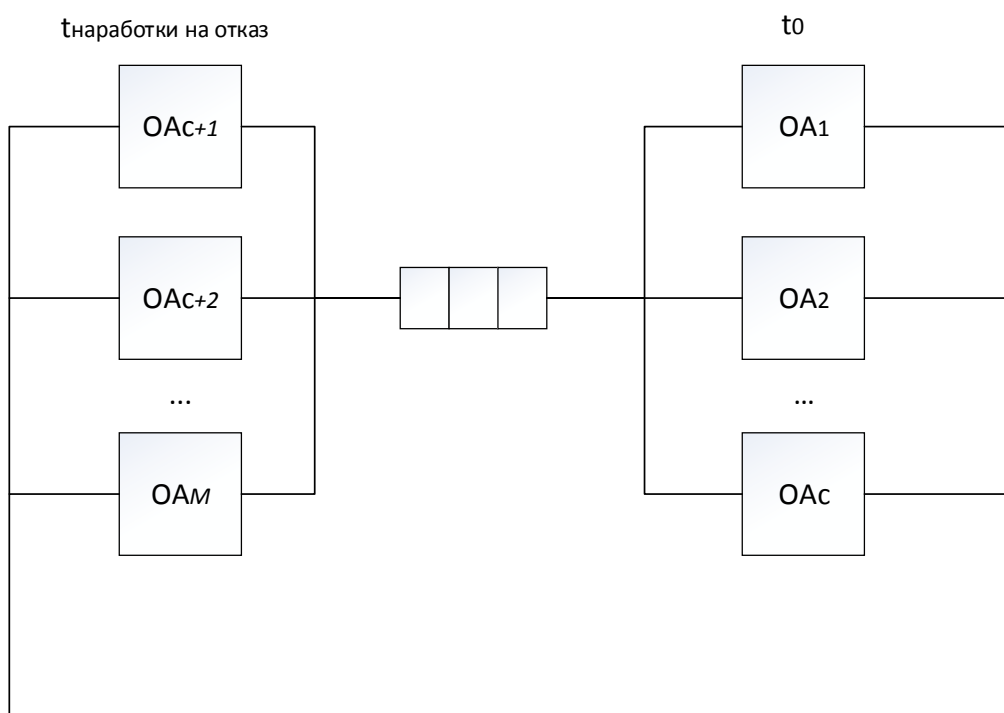


Рисунок 1. Модель ремонтной службы

В качестве результатов расчетов нас интересуют следующие параметры:

Q- среднее количество компьютеров, находящихся в очереди на ремонт

L - среднее количество компьютеров, находящихся в неисправном состоянии, т.е в очереди на ремонт

U – среднее количество компьютеров, которое непосредственно ремонтируются;

ρ_0 – коэффициент загрузки одного специалиста;

n – среднее количество исправных компьютеров;

ρ_e – коэффициент загрузки компьютера;

W – среднее время нахождения компьютера в очереди на ремонт;

T_p – среднее время пребывания компьютера в неисправном состоянии (в очереди на ремонт и ремонте);

$T_{\text{ц}}$ – среднее время цикла для компьютера;

$\frac{\rho_e}{\rho_o}$ - режим работы службы ремонта и обслуживания компьютеров;

Y - Убытки организации.

В качестве параметров рабочей организации выберем следующие параметры:

$t_{\text{но}} = 500$ час - среднее время наработки на отказ одного компьютера;

$t_o = 10$ час – среднее время ремонта одного компьютера;

$\mu_{\text{но}} = \frac{1}{t_{\text{но}}} = 0,002 \frac{1}{\text{час}}$ – интенсивность отказов одного компьютера;

$\mu_o = \frac{1}{t_o} = 0,1 \frac{1}{\text{час}}$ – интенсивность ремонта компьютера;

$N = 40$ – количество компьютеров;

C – количество специалистов, занятых ремонтом компьютеров (2/3/4);

P_k - вероятность, что k компьютеров находятся в состоянии отказа;

$\Psi = \frac{\mu_{\text{но}}}{\mu_o} = 0,02$ - коэффициент отношения интенсивности наработки на

отказ к интенсивности восстановления работоспособности компьютера.

Заработная плата специалиста за один час $S_1 = 150$ руб./час

Финансовые потери организации от неисправного компьютера за один час составляют $S = 3500$ руб./час

Применив последовательно формулы составим таблицу 1. Варианты 1-3 соответствуют моделям с 2, 3 и 4 сотрудниками в ремонтной службе.

Таблица 1 Результаты расчетов для каждого варианта организации работы

Параметр	Вариант 1	Вариант 2	Вариант3
c	2	3	4
Q	0,127	0,015	0,0017
L	0,908	0,799	0,786
U	0,782	0,784	0,784
ρ_0	0,391	0,261	0,196
n	39,091	39,201	39,214
ρ_e	0,977	0,980	0,980
W	1,619	0,193	0,022
T_p	11,619	10,193	10,022
$T_{ц}$	511,619	510,193	510,022
ρ_e/ρ_0	2,5	3,75	5
Y	3025,329	2847,370	2958,071

Из того, что соотношение $\frac{\rho_e}{\rho_0} > 1$ видим, что компьютеры загружены намного больше, чем специалисты, занятые их ремонтом, и, следовательно, в системе мало неисправных компьютеров.

Из отношения денежных затрат Y заметно, что они ниже при 3х работниках и возрастают при уменьшении и увеличении числа работников. Данный вывод визуализирован на рисунке 2.



Рисунок 2. График затрат (АМ)

Имитационное моделирование

Другой способ решить данную задачу – имитационное моделирование.

Существует достаточно обширный инструментарий для создания имитационных моделей любой сложности – наиболее известные из них GPSS, Simio и AnyLogic.

GPSS - *General Purpose Simulation System* – язык моделирования различных систем, в основном – массового обслуживания. Модель на данном языке представляет собой код и тем самым не имеет графической интерпретации в принципе, что уменьшает наглядность. Статические элементы (источники, очереди, устройства и т.п.) в коде располагаются блоками. Динамическим объектом является абстрактный транзакт – элемент который перемещается между блоками [1].

Simio – достаточно старый имитационный язык, наследующий некоторые особенности Arena. Главная особенность – воспринимание каждого элемента моделирования как *интеллектуальный объект*. Таким образом система позволяет оперировать объектами и их параметрами *без* программирования, предоставляет 2D и 3D визуализацию моделей.

AnyLogic – российское решение в области имитационного моделирования. Представляет собой огромную платформу, поддерживающую

различные методы моделирования (системную динамику, дискретно-событийное и агентное моделирование), а также моделирование бизнес-процессов. Продукт написан на Eclipse Java и включает в себя как графическое моделирование, так и возможность расширения моделей Java-кодом.

Для решения поставленной задачи воспользуемся реализацией языка GPSS на GoLang. Go – современный компилируемый многопоточный язык программирования. Из его особенностей – простота и наличие средств параллельного программирования. По большому счету Go – это существенно доработанный Си, с параллелизмом и без наследования реализаций.

Для данного языка существует пакет с реализацией стандартных статических блоков из оригинального GPSS. Исходники находятся в открытом доступе на Github [3]. Реализация изначальной задачи поставленной в данной статье представлена в листинге 1.

Листинг 1 – Реализация задачи на языке Golang

```
1. package main
2. import (
3.     "fmt"
4.     "os"
5.     "os/signal"
6.     "syscall"
7.     "gpss"
8. )
9.
10. func main() {
11.     p := gpss.NewPipeline("Repair Team Pipeline")
12.     g := gpss.NewGenerator("Comps", 0, 0, 0, 40, nil)
13.     a1 := gpss.NewAdvance("Need to repair", 500*60, 60)
14.     q := gpss.NewQueue("Queue to repair")
15.     f1 := gpss.NewFacility("Master1", 600, 30)
16.     f2 := gpss.NewFacility("Master2", 600, 30)
17.     p.Append(g, a1)
18.     p.Append(a1, q)
19.     p.Append(q, f1, f2, f3, f4)
20.     p.Append(f1, a1)
```

```

21. p.Append(f2, a1)
22. p.Start(540*20*12)
23. exit := make(chan struct{})
24. closeSignal := make(chan os.Signal)
25. signal.Notify(closeSignal, os.Interrupt, syscall.SIGTERM)
26. go func() {
27.     <-closeSignal
28.     close(exit)}()
29. go func() {
30.     <-p.Done
31.     close(exit)}()
32. <-exit
33. p.Report()
34. fmt.Println("Exit program")
35. }

```

Логика алгоритма проста – создаются статические объекты: конвейер (pipeline p), генератор (generator g) и т.д. Затем согласно логике задачи (рисунок 1) соединяем объекты. Главное отличие данного решения от аналитического – мы явно задаем срок моделирования. Параметр (540*20*12) – 540 минут рабочего времени эксплуатации (9-часовой рабочий день), 20 – 4 рабочих недели по 5 дней и 12 месяцев. Таким образом можно промоделировать любой срок эксплуатации объекта.

Таблица 2. Результаты имитационного моделирования

Number of masters	2	3	4
Queue to repair			
Max content	38	37	36
Total entries	146	159	160
Zero entries	45	50	46
% of zero entries	30.82%	31.45%	28.75%
Current contents	0	0	0
Average content	1.80	1.18	0.86
Average time/trans	1601.84	961.47	695.57
Average time/trans without zero entries	2315.52	1402.50	976.24

Master1			
Average advance	600.21	606.31	599.50
Average utilization	33.81%	25.26%	18.50%
Number entries	73.00	54.00	40.00
Master2			
Average advance	598.99	605.63	600.25
Average utilization	33.74%	24.30%	18.53%
Number entries	73.00	52.00	40.00
Master3			
Average advance	-	602.55	599.12
Average utilization	-	24.64%	18.49%
Number entries	-	53.00	40.00
Master4			
Average advance	-	-	601.05
Average utilization	-	-	18.55%
Number entries	-	-	40.00

В таблице 2 представлены результаты имитационного моделирования проведенного согласно листинга 1. Первый блок таблицы, Queue to repair, это статический объект очереди на ремонт. Сюда попадают все сломанные объекты. Здесь интерес представляют параметры “% of zero entries”, “average content”, “average time/trans” и “Average time/trans without zero entries”.

“% of zero entries” или «процент нулевых входов» - параметр показывает сколько раз попавший в очередь на ремонт объект был единственным в ней.

“Average content” – среднее количество элементов в очереди, т.е. объект сломан, помещен в очередь, но свободного мастера нет.

“Average time/trans” и “Average time/trans without zero entries” – среднее время ожидания в очереди с и без учета «нулевых проходов». Вторая переменная показывает среднее время элементов, которые ждали своей очереди – без учета объектов, которые немедленно попадали к мастеру.

Графики зависимостей данных переменных приведены на рисунках 3 и 4.

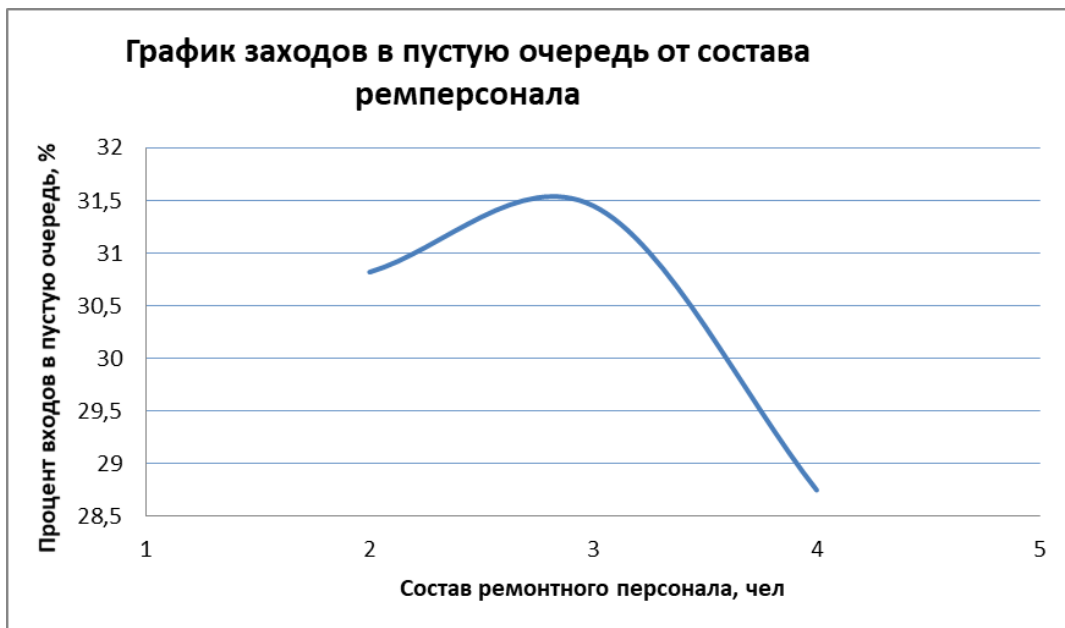


Рисунок 3. График заходов в пустую очередь

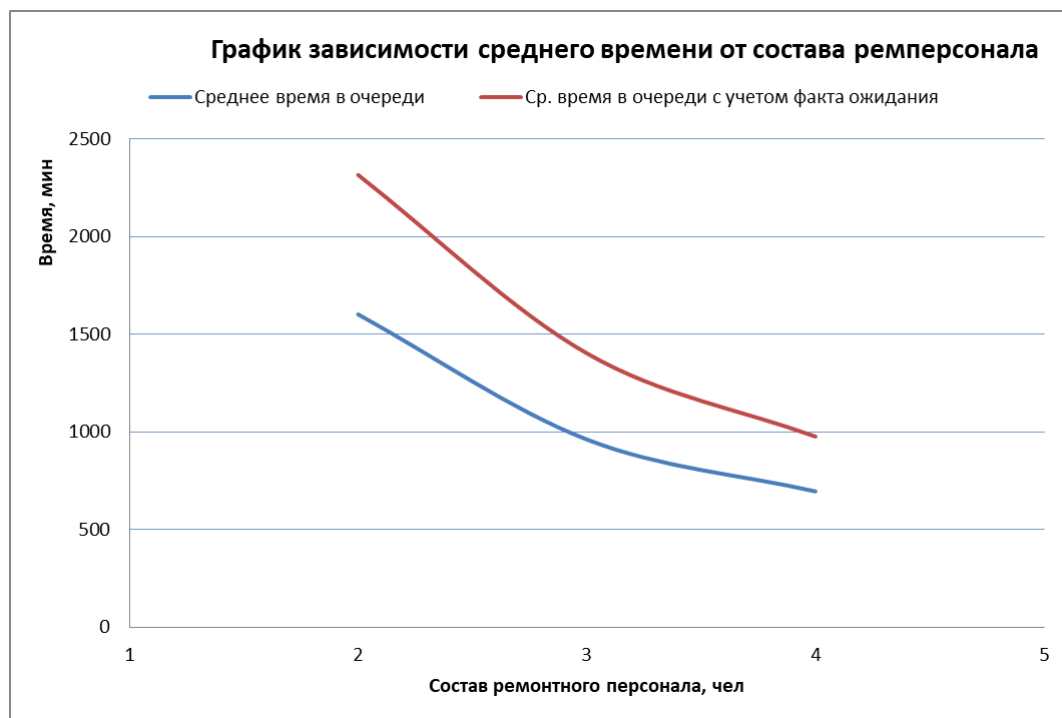


Рисунок 4. График зависимости средних времён

Статический элемент «Мастер» представляет собой блок в котором зашедший объект «чинится» в течении 10 часов и ± 30 минут. Мастера работают параллельно, если первый мастер занят, то объект из очереди на ремонт передается второму мастеру и т.д. Поиск их оптимального количества является целью данного анализа. Здесь интересны первые два параметра.

Average advance – среднее время ремонта одного объекта. Поскольку на ремонт задан разброс по времени (до получаса) то число близко, но не равно 10 часам.

Average utilization – средняя нагрузка мастера, т.е. процент от всего времени, когда он занят. Данный параметр позволяет оценить адекватность рабочей нагрузки и её распределения.

Number entries – количество починенной аппаратуры.

Диаграмма, результирующая полученные значения параметров представлена на рисунке 5.

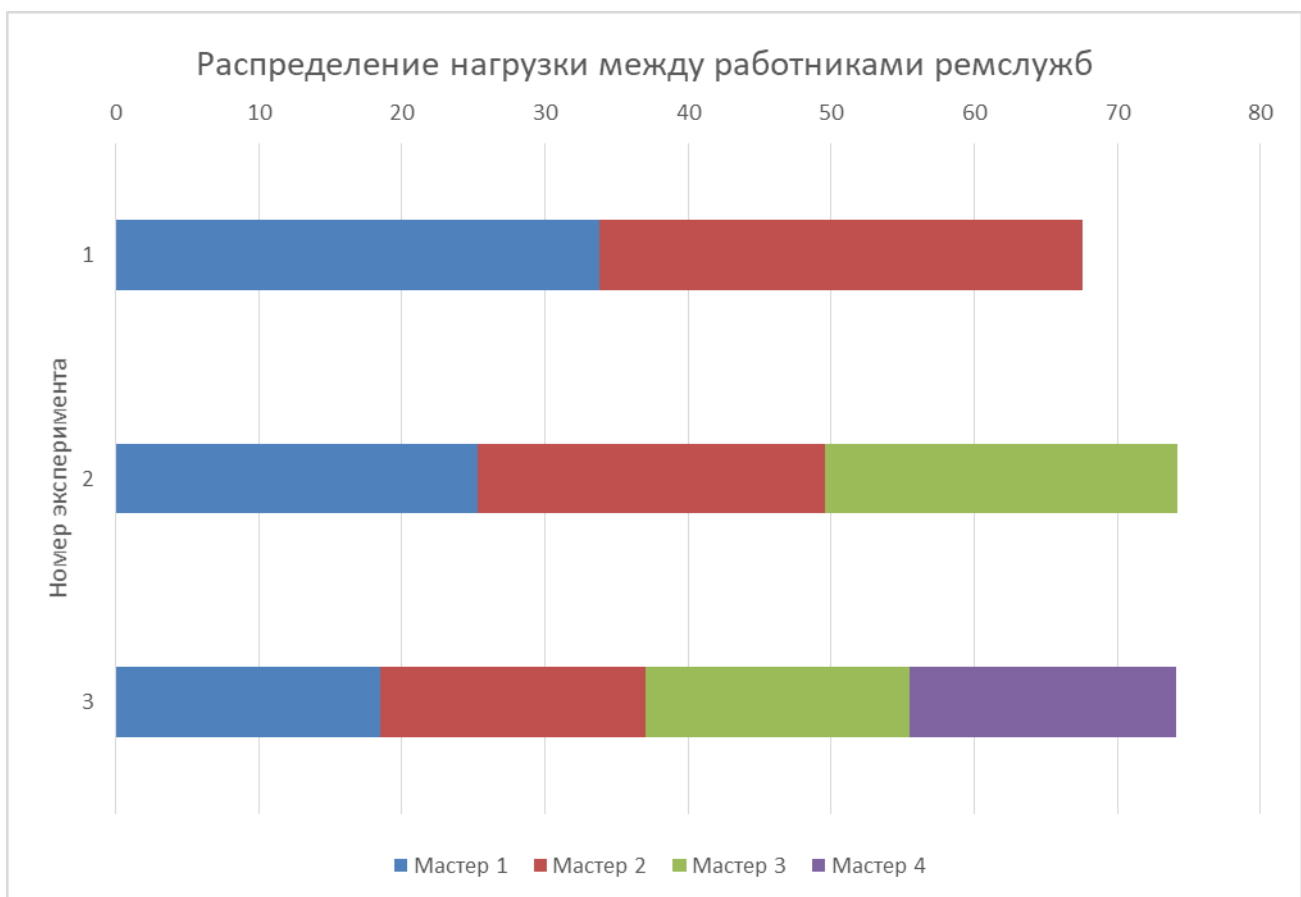


Рисунок 5. Распределение нагрузки

Из данной диаграммы видно, что нагрузка во всех трех экспериментах распределяется равномерно между мастерами. Это обусловлено тем, что время починки достаточно велико, и остальным мастерам достается работа. Иначе, если бы время починки было бы меньше в несколько раз, наблюдалась бы иная картина – нагрузка второго и последующих мастеров уменьшилась,

распределение Number entries имело бы такое же смещение в сторону первого мастера.

Рассчитаем финансовые потери для трех исследуемых вариантов по формуле (11), приняв S_1 и S такими же, как и в аналитическом моделировании (3500 и 150 руб/час соответственно), но переведем их в руб/год, поскольку моделировали работу ремонтной службы за фиксированный период. L – количество аппаратуры в неисправном состоянии в очереди и на ремонте. Логично предположить, что для данного случая:

$$L = \text{Average content} + m$$

Таким образом:

$$S_1 = 324000 \text{ руб/год}$$

$$S = 7560000$$

$$m = 2,3,4$$

$$L = 3,8/3,18/4,86$$

Применив формулу, получим следующий результат, представленный на рисунке 6:



Рисунок 6. График затрат за год (ИМ)

Заключение

По результатам проведенного исследования на практике был рассмотрен пример СМО и проведен расчет состава ремонтного персонала двумя различными способами. Схема расчета применима к любым схожим процессам сферы обслуживания поскольку не привязана к предметной области. Применение данных методов позволит не только снизить затраты, но и обосновать их.

Библиографический список:

1. Кудрявцев Е. М. "GPSS World. Основы имитационного моделирования различных систем"// ДМК-Пресс, 2004.
2. Постников В. М. "Основы эксплуатации автоматизированных систем обработки информации и управления" // Краткий курс: учебное пособие - Москва: МГТУ им. Н.Э. Баумана, 2013.
3. GoLang реализация GPSS. <https://github.com/soldatov-s/go-gpss>.