Моделирование и анализ данных
2021. Том 11. № 4. С. 49–58
DOI: https://doi.org/10.17759/mda.2021110404
ISSN: 2219-3758 (печатный)
ISSN: 2311-9454 (online)

Modelling and Data Analysis
2021. Vol. 11, no. 4, pp. 49–58
DOI: https://doi.org/10.17759/mda.2021110404
ISSN: 2219-3758 (print)
ISSN: 2311-9454 (online)

УДК 519.854.3

Применение задачи Джонсона для решения прикладных задач

Волкова Т.Б.*
Московский авиационный институт
(национальный исследовательский университет)
г. Москва, Российская Федерация
ORCID: https://orcid.org/0000-0002-0777-1111
e-mail: tbvolkova@mail.ru

Осокина А.Д.**
Московский авиационный институт
(национальный исследовательский университет)
г. Москва, Российская Федерация
ORCID: https://orcid.org/0000-0002-0777-1122
e-mail: nastaosokina2@gmail.com

В данной статье рассматривается применение алгоритма Джонсона для анализа работы волонтерской справочной службы и работы интернет-магазина в случае, если исходные данные для алгоритма – случайные числа из заданного интервала. Для этого разработана программа решения задачи Джонсона для моделирования работы, т.е. имитации обработки заявок (заказов, звонков), написанная на языке программирования Python с использованием IDE – PyCharm 2021.1, Qt Designer и PyQt5, в которой время обработки заявок является случайной величиной из заданного интервала. Анализ полученных результатов позволяет давать рекомендации о количестве заявок, которые может обработать обслуживающая система, если время обработки случайно, но принадлежит заданному интервалу, а также прогнозировать время ожидания поступления заявок.

Ключевые слова: задача Джонсона, целочисленное программирование

Волкова Татьяна Борисовна, кандидат физико-математических наук, доцент, доцент кафедры математической кибернетики института «Информационные технологии и прикладная математика» Московского авиационного института (национального исследовательского университета), г. Москва, Российская Федерация, ORCID: https://orcid.org/0000-0002-0777-1111, e-mail: tbvolkova@mail.ru

****Осокина Анастасия Дмитриевна,*** студентка магистратуры института «Информационные технологии и прикладная математика» Московского авиационного института (национального исследовательского университета), г. Москва, Российская Федерация, ORCID: https://orcid.org/0000-0002-0777-1122, e-mail: nastaosokina2@gmail.com

Для цитаты:
Волкова Т.Б., Осокина А.Д. Применение задачи Джонсона для решения прикладных задач // Моделирование и анализ данных. 2021. Том 11. № 4. С. 49–58. DOI: https://doi.org/10.17759/mda.2021110404

1. ВВЕДЕНИЕ

Изучение и решение многих проблем, возникающих из-за их сложности в области экономики, планирования, инженерии и других областях, осуществляется на основе математических моделей, в том числе аппарата целочисленного линейного программирования (ЦЛП). Целочисленное условие позволяет учитывать такие факторы, как неделимость объектов, дискретность процессов, наличие альтернатив, фиксированные надбавки и ряд других факторов.

В статье на примере дискретной задачи планирования производства с интервальными данными использован алгоритм Джонсона – один из алгоритмов целочисленного программирования, позволяющий решать задачи оперативно-календарного планирования работы машин для предприятий мелкосерийного и единичного производства.

Идеи этого метода могут быть использованы для решения таких актуальных задач как поиск наилучшего распределения времени и материальных ресурсов при выполнении заданного множества технологических операций; нахождение кратчайшего пути между всеми парами вершин взвешенного ориентированного графа; поиска наилучшего распределения мест в очереди работ при известной и фиксированной их продолжительности и др.

2. МАТЕМАТИЧЕСКАЯ ПОСТАНОВКА ЗАДАЧИ ДЖОНСОНА И АЛГОРИТМЫ ЕЕ РЕШЕНИЯ

Пусть $N = \{1,2,\ldots,n\}$ – множество работ, которые выполняются на $m(m = 2,3)$ машинах, каждая из которых имеет свой номер; пусть t_{ij} - время выполнения *i*-ой работы на *j*-ой машине. Время простоя вычисляется по формуле:

$$\max_{i,j}\left(t_{11}, t_{11}+t_{21}-t_{12}, t_{11}+t_{21}+t_{31}-t_{12}-t_{22}, \ldots, \sum_{i=1}^{n} t_{i1} - \sum_{j=1}^{n-1} t_{j2}\right), \tag{1}$$

где t_{i1}, t_{j2} – время выполнения работы на первой и второй машине соответственно;
n – количество выполняемых работ.

Требуется так организовать выполнение работ, чтобы суммарное время простоя машин оказалось минимальным.

2.1 Алгоритм решения задачи Джонсона для двух машин

1. Составляется матрица времени выполнения работ для первой и второй машины.
2. В полученной матрице фиксируется минимальный по времени в любом столбце элемент.
3. Если минимальное время обработки этого элемента на первой машине, то этот элемент перемещается в начало очереди, а если на второй машине, то в конец очереди.

4. Обработанный элемент исключается из матрицы.
5. Если еще остались не обработанные элементы, то переходим к п.2. Если элементов не осталось, то переходим к п.6.
6. Вычисляется время ожидания по формуле (1).

2.2 Алгоритм решения задачи Джонсона для трех машин

1. Записывается матрица времени выполнения работ на 3-х машинах.
2. Осуществляется проверка выполнения одного из следующих условий:
   - $\min_i t_{i1} \ge \max_i t_{i2}$, (2)
   - $\min_i t_{i3} \ge \max_i t_{i2}$, где t_{ij} – элемент матрицы времени. (3)
3. Если одно из условий выполняется, строится новая матрица, элементами которой является сумма элементов времени обработки на 1-й и 2-й, 2-й и 3-й машинах.
4. Далее применяется алгоритм Джонсона для двух машин, описанный выше, начиная с п.2.

3. ПРИМЕНЕНИЕ АЛГОРИТМА ДЖОНСОНА ДЛЯ МОДЕЛИРОВАНИЯ РАБОТЫ ВОЛОНТЕРСКОГО СПРАВОЧНОГО ЦЕНТРА

3.1 Постановка задачи

В волонтерский справочный центр поступает конечное число заявок n = [n1, n2, n3, n4] на обслуживание по каждому из 4-х типов: «Помощь по дому», «Доставка», «Медицинская помощь», «Другое» (рис.1). Клиенты оставляют свои заявки на официальном сайте волонтерского справочного центра.

Поступившая заявка обрабатывается на трех этапах.

1. Рассмотрение поступившей заявки.
2. Подготовка к обслуживанию заявок.
3. Оказание услуги.

Время обработки – случайная величина, подчиняющаяся нормальному закону распределения, которая находится в определенном промежутке времени [t1, t2]; [t3, t4];[t5, t6]. Для каждого этапа обработки имеется свой промежуток времени выполнения в соответствии с типом заявки.

Требуется оценить эффективность работы волонтерской службы, проанализировав:

– при каких предельных значениях n все заявки будут обработаны за смену;
– при каких значениях n время ожидания заявки оператором будет минимальным;
– как влияет интервал времени обработки на количество обработанных заявок.

Дать рекомендации по улучшению работы волонтерской справочной службы, выяснив:

– справляется ли система с обработкой входящего потока заявок и требуется ли увеличение числа операторов;
– как влияет время обработки каждой заявки на число обслуженных заявок.

Рис. 1. Схема работы волонтерского справочного центра

Для моделирования работы разработан алгоритм, основные этапы которого приводятся ниже.

3.2. Применение алгоритма Джонсона для анализа работы волонтерской справочной службы.

1. Вводится количество заявок каждого типа.
2. Выбирается время обработки заявок на первом, втором и третьем этапе случайным образом из заданных интервалов времени для каждого типа заявок.
3. Составляется матрица времен обработки поступивших заявок на каждом из трех этапов;
4. Вычисляется время обработки заявки без учета ожидания по формуле: $\sum_{i=1}^{3}\sum_{j=1}^{n} t_{ij}$.
5. Осуществляется проверка выполнения одного из следующих условий:
   - $\min_i t_{i1} \geq \max_i t_{i2}$,
   - $\min_i t_{i3} \geq \max_i t_{i2}$, где t_{ij} – время обработки заявки на одном из этапов.
6. Если одно из условий выполняется, строится новая матрица, элементами которой является сумма времен обработки на 1-ом и 2-ом, 2-ом и 3-ем этапах обработки заявки.
7. Вычисляется фактическое время ожидания по формуле (1).
8. В полученной матрице фиксируется элемент с минимальным временем обработки заявки.
9. Если минимальное время обработки этого элемента находится в первом столбце матрицы, то элемент перемещается в начало очереди, а если во втором, то в конец очереди.
10. Обработанный элемент исключается из рассмотрения.

11. Если еще остались необработанные элементы, то переходим к п.6. Если элементов не осталось, переходим к п.10.
12. Вычисляется время ожидания обработки заявки по формуле и общее время с учетом времени ожидания. Алгоритм завершен.

3.3. Анализ полученных результатов

Исходные данные и полученные результаты моделирования приведены в таблице 1.

Таблица 1

Тип заявки	Интервалы времени для обработки заявок, мин.			Число заявок для обслуживания
	Обработка поступившей заявки	Подготовка к обслуживанию заявки	Оказание услуги	
Помощь по дому	t ϵ [1; 7]	t ϵ [10; 20]	t ϵ [30;120]	5
Доставка	t ϵ [1; 10]	t ϵ [20; 30]	t ϵ [30; 60]	7
Мед. помощь	t ϵ [1; 5]	t ϵ [5; 20]	t ϵ [30; 60]	10
Другое	t ϵ [1; 15]	t ϵ [10; 30]	t ϵ [30;120]	8

В таблице 1 представлена информация о временных промежутках обработки заявок в каждой из четырех категорий («Помощь по дому», «Доставка», «Мед. Помощь», «Другое») на каждом из трех этапов обработки («Обработка поступившей заявки», «Подготовка к обслуживанию заявки», «Оказание услуги»). Во временных промежутках предоставлено минимальное и максимальное время обслуживания заявки на выбранном этапе. В столбце «Число заявок для обслуживания» предоставлена информация о среднем количестве заявок по каждому из типов, которое необходимо обрабатывать за рабочее время.

Таблица 2

Время для обслуживания поступившего числа заявок при заданной норме на обслуживание, мин.

	Общее время	Время ожидания обработки заявки
Фактическое время	2346	20
Время, полученное в результате моделирования	2332	6

В таблице 2 отображен результат работы алгоритма при заданных в таблице 1 входных данных. Показано среднее фактическое время обработки заданного количества заявок и отдельно время ожидания их обслуживания. Из таблицы 2 видно, что планируемое время ожидания, полученное с помощью алгоритма Джонсона, значительно меньше. Это позволяет сделать вывод, что, в указанные временные промежутки обработать заданное число заявок в течении одного дня невозможно. Необходимо уменьшить время обработки заявок за счет более четких алгоритмов их обработки, либо увеличить число операторов, чтобы все поступившие заявки были обслужены.

4. ПРИМЕНЕНИЕ АЛГОРИТМА ДЖОНСОНА ДЛЯ МОДЕЛИРОВАНИЯ РАБОТЫ ИНТЕРНЕТ-МАГАЗИНА

4.1 Постановка задачи

В интернет-магазин поступает n заказов на обслуживание. Клиенты оставляют свои заказы на официальном сайте интернет-магазина (рис.2).

Поступившие заказы проходят три этапа.

1. Проверка наличия заказа на складе
2. Сбор заказа на складе.
3. Доставка.

Время обработки – случайная величина, подчиняющаяся нормальному закону распределения, которая находится в определенном промежутке времени [$t1$, $t2$]; [$t3$, $t4$]; [$t5$, $t6$].

Требуется оценить эффективность работы интернет-магазина, проанализировав:

– при каких значениях n все заказы будут обработаны за смену;
– при каких значениях [$t1$, $t2$]; [$t3$, $t4$]; [$t5$, $t6$] время ожидания заказа оператором будет минимальным;
– как влияет интервал времени обработки на количество обработанных заказов.

Дать рекомендации по улучшению работы интернет-магазина, выяснив:

– справляется ли система с обработкой входящего потока заказов;
– надо ли уменьшить интервал времени обработки заказов для улучшения качества обслуживания.

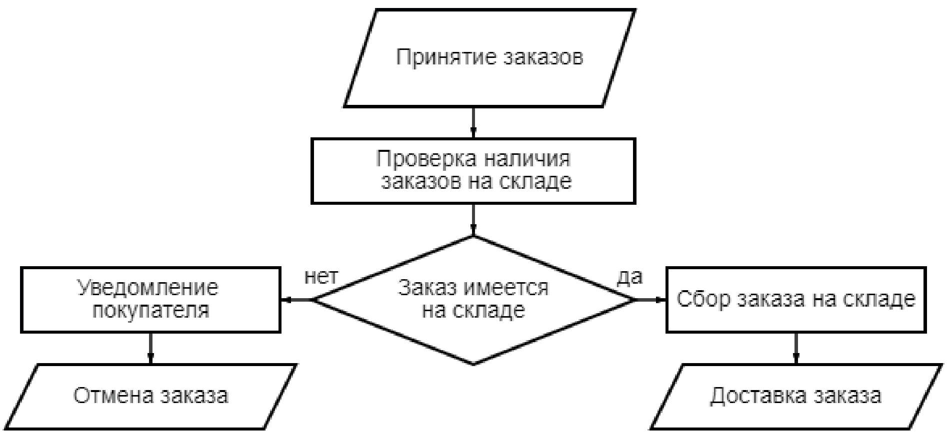

Рис. 2. Схема работы интернет-магазина

4.2. Применение алгоритма Джонсона для анализа работы интернет магазина:

1. Вводится количество заказов, поступивших за смену.
2. Выбирается время обработки заказов на первом, втором и третьем этапа случайным образом из заданных интервалов времени.
3. Составляется матрица времени обработки заказов.

4. Вычисляется время обработки заказов без учета ожидания по формуле: $\sum_{i=1}^{3}\sum_{j=1}^{n} t_{ij}$.
5. Осуществляется проверка выполнения одного из следующих условий:
   - $\min_i t_{i1} \geq \max_i t_{i2}$,
   - $\min_i t_{i3} \geq \max_i t_{i2}$, где t_{ij} - время обработки заявки на одном из этапов.
6. Если одно из условий выполняется, строится новая матрица, элементами которой является сумма элементов времен обработки на 1-ом и 2-ом, 2-ом и 3-ем этапах обработки заказов.
7. Вычисляется фактическое время ожидания по формуле (1).
8. В полученной матрице фиксируется элемент с минимальным временем обработки заявки.
9. Если минимальное время обработки этого элемента находится в первом столбце матрицы, то элемент помещается в начало очереди, а если во втором, то в конец очереди.
10. Обработанный элемент исключается из матрицы.
11. Если еще остались необработанные элементы, то переходим к п.6. Если элементов не осталось, переходим к п.10.
12. Вычисляется время ожидания обработки заявки по формуле и общее время с учетом времени ожидания. Алгоритм завершен.

4.3 Анализ полученных результатов

В таблице 3 представлена информация о временных промежутках обработки на каждом из трех этапов обработки («Проверка наличия заказа на складе», «Сбор заказа на складе», «Доставка заказа»). Во временных промежутках предоставлено минимальное и максимальное время обслуживания заказа на выбранном этапе. В столбце «Число заказов для обслуживания» предоставлена информация о среднем количестве заказов, которое необходимо обрабатывать за рабочее время.

Исходные данные и анализ полученных результатов моделирования.

Таблица 3

<table>
<tr><th rowspan="2"></th><th colspan="3">Интервалы времени для обработки заказов, мин.</th><th rowspan="2">Число заказов для обслуживания</th></tr>
<tr><th>Проверка наличия заказа на складе</th><th>Сбор заказа на складе</th><th>Доставка заказа</th></tr>
<tr><td>Заказ</td><td>t ϵ [1; 7]</td><td>t ϵ [10;20]</td><td>t ϵ [30; 120]</td><td>25</td></tr>
<tr><th colspan="5">Время для обслуживания поступившего числа заказов
при заданной норме на обслуживание, мин.</th></tr>
<tr><th></th><th colspan="3">Общее время</th><th>Время ожидания обработки заказа</th></tr>
<tr><td>Фактическое</td><td colspan="3">2341</td><td>26</td></tr>
<tr><td>Время, полученное в результате моделирования</td><td colspan="3">2327</td><td>12</td></tr>
</table>

В таблице 4 отображен результат работы алгоритма при приведенных в таблице 3 входных данных. Показано среднее фактическое время обработки заданного количества заказов и время ожидания фактическое и рассчитанное с применением алгоритма Джонсона. Из таблицы 4 видно, что время ожидания, рассчитанное с использованием алгоритма Джонсона, значительно меньше. Из таблицы видно, что за заданные временные промежутки обработать заданное число заказов за один день невозможно. Рекомендуется сократить время обработки заказов за счет рационального использования времени или увеличить количество обслуживающего персонала.

Таблица 4

<table>
<tr><th rowspan="2" colspan="2"></th><th colspan="3">Интервалы времени для обработки заказов, мин.</th><th rowspan="2" colspan="2">Число заказов для обслуживания</th></tr>
<tr><th>Проверка наличия заказа на складе</th><th>Сбор заказа на складе</th><th>Доставка заказа</th></tr>
<tr><td colspan="2">Заказ</td><td>t ϵ [1; 5]</td><td>t ϵ [3; 10]</td><td>t ϵ [15; 25]</td><td colspan="2">25</td></tr>
<tr><td colspan="7">Время для обслуживания поступившего числа заказов при заданной норме на обслуживание, мин.</td></tr>
<tr><td colspan="2"></td><td colspan="4">Общее время</td><td>Время ожидания обработки заказа</td></tr>
<tr><td colspan="2">Фактическое</td><td colspan="4">720</td><td>9</td></tr>
<tr><td colspan="2">Время, полученное в результате моделирования</td><td colspan="4">715</td><td>4</td></tr>
</table>

5. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

Для решения поставленной задачи разработана программа решения задачи Джонсона для моделирования работы, т.е. имитации обработки заявок (заказов, звонков), написанная на языке программирования Python с использованием IDE – PyCharm 2021.1, Qt Designer и PyQt5, в которой время обработки заявок является случайной величиной из заданного интервала.

6. ЗАКЛЮЧЕНИЕ

В представленной статье приведено обоснование применения алгоритма Джонсона для анализа работы волонтерской справочной службы и работы интернет-магазина. Анализ полученных результатов позволяет давать рекомендации о количестве заявок, которые может обработать обслуживающая система, если время обработки случайно, но принадлежит заданному интервалу, а также прогнозировать время ожидания поступления заявок.

Алгоритм Джонсона применим для решения многих прикладных задач. С его помощью можно проанализировать работу любой системы, в которой обработка заявок происходит в несколько этапов.

Литература

1. *Корбут А.А.* Дискретное программирование // А.А. Корбут, Ю.Ю. Финкельштейн – М.: Наука, 1975 г.

2. *Кофман А.* Введение в прикладную комбинаторику // А. Кофман – М.: Наука, 1975 г.
3. *Кормен Т.* Алгоритмы: построение и анализ 2-е изд. // Т. Кормен , Ч. Лейзерсон , Р. Ривест – М.: «Вильямс», 2007 г.
4. *S.M. Johnson.* Optimal two – and three-stage production schedules with setup times included // P-402. Santa Monica, California, the RAND Corporation, 1953. – P. 10.

Application of the Johnson Problem to Solve Applied Problems

Tatiana B. Volkova*
MAI (National Research University), Moscow, Russia
ORCID: https://orcid.org/0000-0002-0777-1111
e-mail: tbvolkova@mail.ru

Anastasia D. Osokina**
MAI (National Research University), Moscow, Russia
ORCID: https://orcid.org/0000-0002-0777-1122
e-mail: nastaosokina2@gmail.com

This article discusses the application of Johnson's algorithm to analyze the work of a volunteer help desk and the work of an online store if the initial data for the algorithm are random numbers from a given interval. For this purpose, a program has been developed for solving the Johnson problem for modeling work, i.e. simulating the processing of applications (orders, calls), written in the Python programming language using IDE – PyCharm 2021.1, Qt Designer and PyQt5, in which the processing time of applications is a random variable from a given interval. The analysis of the results obtained allows us to make recommendations on the number of applications that the service system can process if the processing time is random, but belongs to a given interval, as well as predict the waiting time for applications to arrive.

Keywords: Johnson's algorithm, integer programming.

For citation:
Volkova T.B., Osokina A.D. Application of the Johnson problem to solve applied problems. *Modelirovanie i analiz dannykh = Modelling and Data Analysis,* 2021. Vol. 11, no. 4, pp. 49–58. DOI: https://doi.org/10.17759/mda.2021110404 (In Russ.,abstr. in Engl.).

References

1. A.A. Korbut and Yu. Yu. Finkelstein "Discrete programming" Moscow: Nauka, 1969.
2. A. Kofman "Introduction to applied combinatorics" Moscow: Nauka, 1975.
3. T. Kormen, C. Leiserson, R. Rivest "Algorithms: construction and analysis 2nd ed." Moscow: "Williams", 2007.
4. S.M. Johnson. Optimal two – and three-stage production schedules with setup times included. P-402. Santa Monica, California, the RAND Corporation, 1953. – P. 10.

***Tatiana B. Volkova,** Candidate of Physical and Mathematical Sciences, Associate Professor, Head of the Department of Mathematics and Cybernetics, Institute of Information Technologies and Applied Mathematics, Moscow Aviation Institute (National Research University), Moscow, Russia, ORCID: https://orcid. org/0000-0002-0777-1111, e-mail: tbvolkova@mail.ru

****Anastasia D. Osokina,** Student of the Institute of Information Technologies and Applied Mathematics, Moscow Aviation Institute (National Research University), Moscow, Russia, ORCID: https://orcid.org/0000-0002-0777-1122, e-mail: nastaosokina2@gmail.com