

УДК 519.23; 303.717

DOI: [10.26102/2310-6018/2026.53.2.007](https://doi.org/10.26102/2310-6018/2026.53.2.007)

Компьютерная реализация точного распределения ранговых статистических критериев методами динамического программирования

Л.В. Агамиров^{1,2,3}, В.Л. Агамиров^{1,2}, В.А. Вестяк¹, Н.В. Тутова²✉

¹Московский авиационный институт, Москва, Российская Федерация

²Московский технический университет связи и информатики, Москва, Российская Федерация

³Национальный исследовательский университет «МЭИ», Москва, Российская Федерация

Резюме. В рамках настоящего исследования анализируется актуальная проблема, связанная с определением точных распределений для ранговых непараметрических критериев при отсутствии замкнутых аналитических решений. Классический подход, основанный на полном переборе всех возможных перестановок рангов, хотя и является теоретически точным, оказывается практически неприменимым уже для небольших объемов выборок вследствие комбинаторного взрыва числа вариантов. Прямой перебор всех возможных перестановок рангов, являющийся методом точного расчета, оказывается вычислительно неразрешимой задачей даже для выборок небольшого объема из-за комбинаторного взрыва. Рассмотрены наиболее известные ранговые непараметрические критерии, не имеющие аналитического решения для получения полной функции распределения, такие как критерии Лемана-Розенבלата, Краскела-Уоллиса и Муда. Существующие аппроксимации (нормальная, хи-квадрат) часто оказываются неудовлетворительными для малых выборок. В работе предлагается эффективное решение на основе метода динамического программирования, позволяющее в сотни раз сократить вычислительные затраты по сравнению с наивной генерацией перестановок. Реализована методика, включающая генерацию последовательностей рангов, расчет статистик для каждой из них и последующую агрегацию результатов для построения функции распределения. Проведенные вычислительные эксперименты однозначно демонстрируют, что метод динамического программирования является наиболее эффективным для генерации точных распределений. Разработаны и представлены в открытом доступе программные реализации на C++ и Python, проведено их сравнительное тестирование, которое подтвердило ожидаемое лидерство C++ с точки зрения быстродействия.

Ключевые слова: непараметрическая статистика, ранговые критерии, точное распределение, p -value, динамическое программирование, вычислительная эффективность, открытый код.

Для цитирования: Агамиров Л.В., Агамиров В.Л., Вестяк В.А., Тутова Н.В. Компьютерная реализация точного распределения ранговых статистических критериев методами динамического программирования. *Моделирование, оптимизация и информационные технологии*. 2026;14(2). URL: <https://moitvvt.ru/ru/journal/article?id=2175> DOI: 10.26102/2310-6018/2026.53.2.007

Computer implementation of exact distribution of rank statistical criteria using dynamic programming methods

L.V. Agamirov^{1,2,3}, V.L. Agamirov^{1,2}, V.A. Vestyak¹, N.V. Toutova²✉

¹Moscow Aviation Institute, Moscow, the Russian Federation

²Moscow Technical University of Communications and Informatics, Moscow, the Russian Federation

³National Research University «MPEI», Moscow, the Russian Federation

Abstract. This paper considers the problem of calculating exact distributions for nonparametric rank tests in the absence of analytical solutions. The classical approach based on a complete enumeration of all possible permutations of ranks, although theoretically accurate, turns out to be practically inapplicable even for small sample sizes due to the combinatorial explosion of the number of variants. A straightforward enumeration of all possible rank permutations, which is an exact calculation method, proves computationally intractable even for small samples due to combinatorial explosion. The most well-known nonparametric rank tests lacking an analytical solution for obtaining the full distribution function are considered, including the Lehmann-Rosenblatt, Kruskal-Wallis, and Mood tests. Existing approximations (normal, chi-square) often prove unsatisfactory for small samples. This paper proposes an efficient solution based on dynamic programming, which reduces computational costs by hundreds of times compared to naive permutation generation. The methodology implemented includes generating rank sequences, calculating statistics for each sequence, and then aggregating the results to construct the distribution function. Computational experiments conducted clearly demonstrate that dynamic programming is the most effective method for generating accurate distributions. Software implementations in *C++* and *Python* have been developed and made publicly available, and comparative testing has confirmed the expected performance advantage of *C++*.

Keywords: nonparametric statistics, rank tests, exact distribution, *p*-value, dynamic programming, computational efficiency, open source.

For citation: Agamirov L.V., Agamirov V.L., Vestyak V.A., Toutova N.V. Computer implementation of exact distribution of rank statistical criteria using dynamic programming methods. *Modeling, Optimization and Information Technology*. 2026;14(2). (In Russ.). URL: <https://moitvvt.ru/ru/journal/article?id=2175> DOI: 10.26102/2310-6018/2026.53.2.007

Введение

Задача проверки статистических гипотез с использованием непараметрических методов часто упирается в вычислительную сложность получения точных распределений соответствующих статистик. Когда аналитическое решение для полной функции распределения отсутствует, единственным точным методом остается подсчет вероятностей путем перебора всех возможных перестановок рангов¹. Однако, как отмечается в литературе, даже при сравнительно малых объемах выборок количество таких перестановок становится астрономически большим, что делает прямой перебор практически нереализуемым.

В работах [1, 2] предпринимались попытки решить эту проблему за счет рекурсивных алгоритмов и производящих функций, позволяющих вычислить точные значения *p-value* без полного перебора. В частности, техника Ван де Вилля [2, 3] с разделением объединенного рейтинга и рекурсивным вычислением совместного распределения сумм рангов показала свою эффективность для ряда критериев. Тем не менее, для таких широко используемых критериев, как Краскела-Уоллиса, проблема вычислительной сложности оставалась острой, особенно для больших выборок, где время расчетов растет экспоненциально [4, 5].

Современные исследования в этой области сфокусированы на разработке оптимизированных алгоритмов, использующих методы динамического программирования [6], символьные вычисления [7], а также распараллеливание и GPU-ускорение [8]. Эти подходы позволяют значительно расширить границы применимости точных методов. В данной работе мы представляем реализацию алгоритма динамического программирования для расчета точных распределений нескольких ключевых ранговых критериев, проводим анализ его вычислительной эффективности и предлагаем открытые программные реализации, устраняющие проблему недоступности алгоритмов в существующих коммерческих и открытых пакетах.

¹ Mehta C.R., Patel N.R. *IBM SPSS Exact Tests*. IBM Corp.; 2013. 226 p.

Для анализа современных возможностей компьютерного расчета точных распределений целесообразно рассмотреть следующие категории программных решений:

1. Специализированные статистические пакеты. Классическим примером является пакет *exactRankTests* для языка *R*. Его архитектура основана на алгоритме сдвига (*Shift Algorithm*), предложенном Штрайтбергом и Ремелем [9], который оптимизирует расчет точного распределения для умеренных объемов данных. Несмотря на открытую лицензию, пакет использует ряд устаревших алгоритмов, что может приводить к снижению производительности и точности при обработке данных с большим количеством связей (*ties*).

2. Коммерческие платформы. Модуль *Exact Tests* в составе *IBM SPSS* представляет собой мощное коммерческое решение с широким набором точных тестов [10]. Он обеспечивает надежный расчет *p-value* для малых выборок и данных со связями, поддерживая такие критерии, как Манна-Уитни, Уилкоксона и другие. Однако это решение обладает рядом существенных недостатков: закрытый исходный код, высокая стоимость, ограниченные возможности интеграции с открытыми экосистемами (такими как *Python* или *R*), а также санкционные ограничения, затрудняющие его использование в РФ.

3. Библиотеки для языков общего назначения. Встроенная библиотека *SciPy* для *Python* [11] включает реализацию ряда непараметрических тестов (Манна-Уитни, Краскела-Уоллиса и др.), и, начиная с версии 1.7, для некоторых из них поддерживается расчет точного *p-value*. Процедура различия 2-х связанных выборок с использованием непараметрического критерия знаков в MS Excel реализована в [12]. Для сценариев, требующих максимальной точности на малых выборках, могут использоваться специализированные пакеты, например, *exact-pvalues*, или интерфейсы для интеграции с *R*, такие как *rpy2*. Анализ документации и исходного кода этих пакетов показывает, что многие из них восходят к алгоритмам, разработанным в рамках проекта *Royal Statistical Society «Applied Statistics algorithms»* [13, 14], что подчеркивает преемственность в развитии вычислительных методов статистики.

Проведенный анализ выявил, что, несмотря на обилие реализаций, многие алгоритмы остаются «черными ящиками», а их код – труднодоступным для изучения и модификации. Это затрудняет их верификацию, адаптацию под специфические исследовательские задачи и интеграцию в новые вычислительные среды. Для преодоления этих ограничений в данной работе разработаны и представлены собственные программные модули с открытым исходным кодом.

Материалы и методы

В настоящей работе рассматриваются наиболее известные ранговые непараметрические критерии, не имеющие аналитического решения для получения полной функции распределения. Подробное описание критериев с примерами содержится в справочнике [15], а также в других цитируемых литературных источниках. Здесь рассмотрим лишь краткое описание критериев.

Критерий Лемана-Розенבלата. В рамках данной работы рассматривается ранговый непараметрический критерий, предложенный Леманом и Розенблаттом [13], который применяется для проверки гипотезы об однородности (идентичности) двух независимых выборок. Содержательно нулевая гипотеза H_0 формулируется как предположение о том, что обе выборки извлечены из одной и той же генеральной совокупности, то есть $F(x) = G(x)$ для любого x .

Статистика критерия рассчитывается по формуле:

$$\omega^2 = \frac{1}{n \cdot m} \left[\frac{1}{6} + \frac{\sum_{i=1}^n (R_i - i)^2}{m} + \frac{\sum_{j=1}^m (S_j - j)^2}{n} \right] + \frac{2}{3}, \quad (1)$$

где R_i, S_j – ранги первой (объемом n) и второй (объемом m) выборок соответственно, расположенные в общем вариационном ряду. Критическая область для проверки нулевой гипотезы (H_0) является правосторонней: гипотеза отвергается, если рассчитанное значение статистики ω^2 превышает критическое (ω_α^2) для выбранного уровня значимости (α). При справедливости (H_0) распределение статистики (ω^2) асимптотически сходится к предельному нормальному закону, что позволяет использовать следующую нормальную аппроксимацию для принятия решения о гипотезе однородности:

$$u_{1-\alpha/2} \leq u \leq u_{\alpha/2}, \quad (2)$$

$$u = \frac{\left(\frac{nm}{n+m} \omega^2 - a \right)}{(45d)^{0.5}} + 1/6,$$

$$\text{где } a = \frac{1}{6} \cdot \left(1 + \frac{1}{n+m} \right), d = \frac{1}{45} \left(1 + \frac{1}{n+m} \right) \left[1 + \frac{1}{n+m} - \frac{3}{4} \left(\frac{1}{n} + \frac{1}{m} \right) \right],$$

где u_α – представляет собой соответствующую квантиль стандартного нормального распределения. Представленная аппроксимация демонстрирует удовлетворительную точность для практического применения уже при объемах выборок ($n, m > 7$). Важно отметить, что в доступных литературных источниках, включая [15], аналитическое решение для точного (а не асимптотического) распределения статистики данного критерия отсутствует.

Критерий Краскела-Уоллиса [16] обобщает задачу о двух выборках на случай σ_y^2 выборок: $\sigma_y^2 \approx \left(\frac{\partial y(\lg N)}{\partial \lg N} \right)^2 \cdot \sigma_{\lg N}^2 = (1 - \chi)^2 \cdot (\lg N)^{-2\chi} \cdot \sigma_{\lg N}^2$ с функциями распределения $y = (\lg N)^{1-\chi}$, где $\sigma_y^2 \approx (1 - \chi)^2 \cdot h^2$ – число наблюдений в j -й выборке. Нулевая гипотеза утверждает, что h, χ выборки из произвольных совокупностей можно рассматривать как одну (объединенную) выборку из общей совокупности, то есть утверждается равенство параметров сдвига $y = (\lg N)^{1-\chi}$, когда не задано значение общего параметра масштаба $\sigma_a = \sigma_{-1} + q \cdot (\lg N)^{1-\chi}$ против альтернативы $\chi = \sigma_a$, $y = (\lg N)^{1-\chi}, b_1 = b_2 \cdot (\bar{x} - \sigma_{-1}), b_2 = 1/q$ не все равны. Для проверки нулевой гипотезы строят общий вариационный ряд из $N = \sum_{i=1}^k n_i$ наблюдений и рассчитывают статистику:

$$H = \frac{12}{N \cdot (N+1)} \cdot \sum_{i=1}^k \frac{R_i^2}{n_i} - 3 \cdot (N+1), \quad (3)$$

где $y = (\lg N)^{1-\chi}$ – сумма рангов i -ой выборки в общем вариационном ряду. Далее рассчитывают величину $\hat{b}(\hat{\sigma}_{-1}) = [\tilde{X}(\hat{\sigma}_{-1})^T \cdot V^{-1} \cdot \tilde{X}(\hat{\sigma}_{-1})]^{-1} \cdot \tilde{X}(\hat{\sigma}_{-1})^T \cdot V^{-1} \cdot \hat{y}$:

$$H_1 = \frac{H}{2} \cdot \left(1 + \frac{N-k}{N-1-H} \right), \quad (4)$$

которую сравнивают с критическим значением $p = 0,5$:

$$H_\alpha = 0,5 \cdot [(k-1) \cdot F_{1-\alpha} + \chi_{1-\alpha}^2], \quad (5)$$

где $F_{1-\alpha}$ – квантиль уровня $1 - \alpha$ – F -распределения с числами степеней свободы $f_1 = k-1, f_2 = N-k$; $\chi_{1-\alpha}^2$ – квантиль уровня $1 - \alpha$ распределения χ^2 с числом степеней свободы $f = k-1$.

Нулевую гипотезу принимают, если F с уровнем значимости $F = s_2^2/s_1^2$, в противном случае принимают альтернативную гипотезу.

Критерий Муда. Критерий Муда относится к непараметрическим ранговым критериям сравнения параметров масштаба. Данные критерии строятся на базе соответствующих критериев сдвига изменением либо статистики критерия, либо правил присвоения рангов наблюдениям. Критерии масштаба преследуют цель выявить возможные различия в мерах разброса (изменчивости) наблюдений в двух или более выборках.

Статистика критерия Муда [17] имеет вид:

$$M = \sum_{i=1}^m \left(R_i - \frac{m+n+1}{2} \right)^2, \quad (6)$$

где R_i – ранги элементов меньшей выборки в общем вариационном ряду. Нулевая гипотеза о равенстве параметров масштаба принимается, если

$$u_{1-\alpha/2} \leq u \leq u_{\alpha/2}, \quad (7)$$

$$u = \frac{|M - E_M + 0,5|}{(D_M)^{0,5}}, \quad (8)$$

$$E_M = \frac{m(m+n+1)(m+n-1)}{12}, \quad D_M = \frac{mn(m+n+1)(m+n+2)(m+n-2)}{180}, \quad (9)$$

где $u_{\alpha/2}$ – квантиль нормированного нормального распределения для уровня значимости α . Эффективность критерия по отношению к F -критерию Фишера в случае исходного нормального распределения равна 0,76. Необходимо отметить, что критерий Муда (как и все ранее рассмотренные критерии) предполагает равенство параметров положения. Точного аналитического решения для распределения статистики критерия не обнаружено.

Критерий Ансари-Брэдли. Статистика масштабного критерия Ансари-Брэдли [18] для проверки гипотез об однородности параметров масштаба вычисляется следующим образом:

$$S = \sum_{i=1}^{n_1} \left\{ \frac{n_1+n_2+1}{2} - \left| R_i - \frac{n_1+n_2+1}{2} \right| \right\}, \quad (10)$$

где R_i – ранги элементов первой выборки в общем вариационном ряду.

При объемах выборок $n_1, n_2 > 10$ дискретное распределение нормированной статистики достаточно хорошо приближается стандартным нормальным законом. Математическое ожидание и дисперсия статистики (10) имеют вид:

$$E_S = \frac{n_1(n_1+n_2+2)}{4}, \quad D_S = \frac{n_1n_2(n_1+n_2-2)(n_1+n_2+2)}{48(n_1+n_2-1)} \text{ при четном } n_1 + n_2, \quad E_S = \frac{n_1(n_1+n_2+2)^2}{4(n_1+n_2)},$$

$$D_S = \frac{n_1n_2(n_1+n_2+1)[(n_1+n_2)^2+3]}{48(n_1+n_2)^2} \text{ при нечетном } n_1 + n_2.$$

Нулевая гипотеза о равенстве параметров масштаба принимается, если

$$u_{1-\alpha/2} \leq u \leq u_{\alpha/2}, \quad (11)$$

$$u = \frac{|S - E_S|}{(D_S)^{0,5}}. \quad (12)$$

Компьютерная реализация алгоритмов, основанных на методах динамического программирования. Основной вклад настоящей работы заключается в применении методов динамического программирования (ДП) для эффективного построения полной функции распределения статистик ранговых критериев. Предлагаемый подход позволяет кардинально снизить вычислительные затраты по сравнению с наивными методами перебора, которые сталкиваются с проблемой комбинаторного взрыва.

Использование ДП позволяет на несколько порядков сократить вычислительные затраты по сравнению с наивным полным перебором перестановок, который приводит к

комбинаторному взрыву даже при небольшом общем количестве наблюдений. Ключевая идея, заложенная в разработанный алгоритм, заключается в отказе от генерации всех перестановок в пользу инкрементального (пошагового) вычисления распределения статистики.

Метод ДП обходит эту проблему, разбивая задачу на более мелкие перекрывающиеся подзадачи и комбинируя их решения. Ключевая идея заключается в вычислении распределения статистики «инкрементально» (постепенно), по мере того как мы последовательно «добавляем» по одному наблюдению (рангу) в ту или иную группу.

Пусть имеется k групп с размерами n_1, n_2, \dots, n_k . Общее количество наблюдений $N = \sum n_i$. Всем наблюдениям присваиваются ранги от 1 до N .

Алгоритм ДП строится на следующем наблюдении. Пусть $T[S_1, S_2, \dots, S_k]$ – количество способов распределить первые m рангов так, чтобы сумма рангов в группе i была равна S_i . При добавлении следующего ранга $r = m + 1$ можно поместить его в любую группу j при условии, что ее емкость (количество уже помещенных в нее наблюдений) еще не исчерпана. Это приводит к рекуррентному соотношению: $T_{m+1}(S_1, \dots, S_j + r, \dots, S_k) = \sum T_m(S_1, \dots, S_j, \dots, S_k)$ для всех групп j и всех достижимых состояний (S_1, S_2, \dots, S_k) .

Проиллюстрируем этот процесс на простом примере. Предположим, имеется две группы ($k = 2$) и обрабатывается ранг ($r = 2$). Если на предыдущем шаге было состояние $(S_1 = 1, S_2 = 0)$, означающее, что ранг 1 был назначен первой группе, то назначение ранга 2 первой группе приведет к новому состоянию $(S_1 = 3, S_2 = 0)$. Если же ранг 2 будет назначен второй группе, состояние сменится на $(S_1 = 1, S_2 = 2)$. Таким образом, система инкрементально перебирает не перестановки, а пространство возможных сумм рангов для групп.

Далее алгоритм итерируется по всем рангам от 1 до N и для каждого возможного сочетания текущих сумм (состояние ДП) обновляет счетчики. Размерность таблицы состояний равна k (количество групп), а ее размер определяется максимально возможной суммой рангов в группе, что на практике много меньше, чем $N!$.

После обработки всех N рангов, таблица $T_N(S_1, S_2, \dots, S_k)$ содержит частоту (количество перестановок) для каждого возможного вектора сумм. Вероятность вычисляется как сумма частот всех конфигураций, для которых значение статистики критерия (вычисленное по S_i) больше или равно наблюдаемому значению, отнесенная к общему числу перестановок.

Более подробно слово «инкрементально» (от англ. *Increment* – приращение, увеличение) означает постепенно, шаг за шагом, путем добавления небольших порций. В контексте алгоритма динамического программирования это означает, что вместо того, чтобы пытаться сразу рассмотреть всю огромную задачу (распределение всех N рангов по всем группам), алгоритм решает ее по частям. Он начинает с самой маленькой и простой подзадачи и на каждом шаге немного ее усложняет, используя результат предыдущего шага.

На первом этапе имеется 0 рангов и все суммы групп равны 0. Затем добавляется ранг 1 ($r = 1$). Можно поместить этот первый ранг в любую группу, которая еще не заполнена. Получим несколько возможных состояний: $(S_1 = 1, S_2 = 0, \dots)$, $(S_1 = 0, S_2 = 1, \dots)$ и т. д. Запишем, что каждое из этих состояний достижимо 1 способом. Затем добавим ранг 2 ($r = 2$). Теперь рассматриваются все состояния, которые получились после шага 1. К каждому из этих состояний добавляется ранг 2 в каждую из групп. Если у нас было состояние $(S_1 = 1, S_2 = 0)$, то добавление ранга 2 в первую группу даст новое состояние $(S_1 = 3, S_2 = 0)$. Добавление во вторую группу даст $(S_1 = 1, S_2 = 2)$. То есть инкрементально обновляется таблица состояний, основываясь на предыдущих результатах. Процесс продолжается, пока не «израсходуются» все ранги от 1 до N . На каждом шаге известно полное распределение вероятностей для всех возможных сумм рангов, полученных из первых рангов.

В итоге алгоритм не делает полный перебор, а строит решение от простого к сложному, вычисляя и запоминая результаты маленьких подзадач, чтобы с их помощью решить подзадачи. Это ядро динамического программирования.

Метод даёт точное распределение с вычислительной сложностью $O(N^{2k-1})$ существенно ниже, чем $O(N!)$ у прямого перебора.

При этом один раз построив таблицу распределения сумм рангов, можно быстро вычислить статистику для любого критерия, основанного на суммах рангов.

Данный метод является развитием идей, заложенных в алгоритме «*Network Algorithm*» для таблиц сопряженности, примененных к проблеме рангов. В последнее время он активно используется в пакетах точных тестов [1]. Работы 2020-х годов фокусируются на дальнейшей оптимизации: использовании симметрии, эффективных структур данных для разреженных матриц состояний, а также на распараллеливании и GPU-ускорении для обработки выборок большого размера.

Алгоритм точного распределения ранговых непараметрических критериев методом динамического программирования на примере критерия Краскела-Коллиса представлен на блок-схеме на Рисунке 1. Состояние системы кодируется в компактном векторе, что позволяет использовать его как ключ в хеш-таблице. Динамическое программирование осуществляется «снизу-вверх»: строится решение для N рангов, итеративно обновляя множество достижимых состояний. Алгоритм автоматически учитывает, что перестановки внутри групп не влияют на сумму рангов, что заложено в формуле для `total_combinations`.

Входные данные: n_groups : вектор целых чисел, представляющих размеры k групп. Например, $n_groups = [3,4,5,6]$ – вектор для четырех групп с 3, 4, 5 и 6 наблюдениями.

Выходные данные: вектор кортежей $(H, P(H), cumulative_P)$, где H – значение статистики Краскела-Уоллиса, $P(H)$ – точная вероятность данного значения (функция частот), $cumulative_P$ – кумулятивная (накопленная) вероятность.

Полный код программ точного распределения ранговых критериев методом динамического программирования на C++ и Python приведен в Github-репозитории.^{2,3}

Результаты

Как уже отмечалось для k -выборочных критериев затраты машинного времени и памяти оказываются значительными. В Таблице 1 представлены несколько вариантов расчетов времени решения перестановкой с повторениями и рассмотренным методом динамического программирования.

Таблица 1 – Сравнение времени вычислений точного критерия Краскела-Уоллиса перестановкой с повторениями и методом ДП

Table 1 – Comparison of the computation time of the exact Kruskal-Wallis test by permutation with repetitions and the DP method

№	Выборка	Полное количество вариантов	t, сек		
			Перестановка с повторениями (C++)	Динамическое программирование	
				Python	C++
1	3,4,4,4	15765750	15,016	3,03	1,179
2	3,4,4,5	50450400	50,06	5,45	2,127
3	4,5,5,5	2933186256	–	29,37	11,83
4	5,5,5,5	11732745024	–	50,81	18,786

² AVL095. crit_rank_7. GitHub. URL: https://github.com/AVL095/crit_rank_7 (дата обращения: 14.12.2025).

³ AVL095. crit_rank_8. GitHub. URL: https://github.com/AVL095/crit_rank_8 (дата обращения: 14.12.2025).

Заключение

Основные результаты проведенного исследования приведены на Рисунке 1.

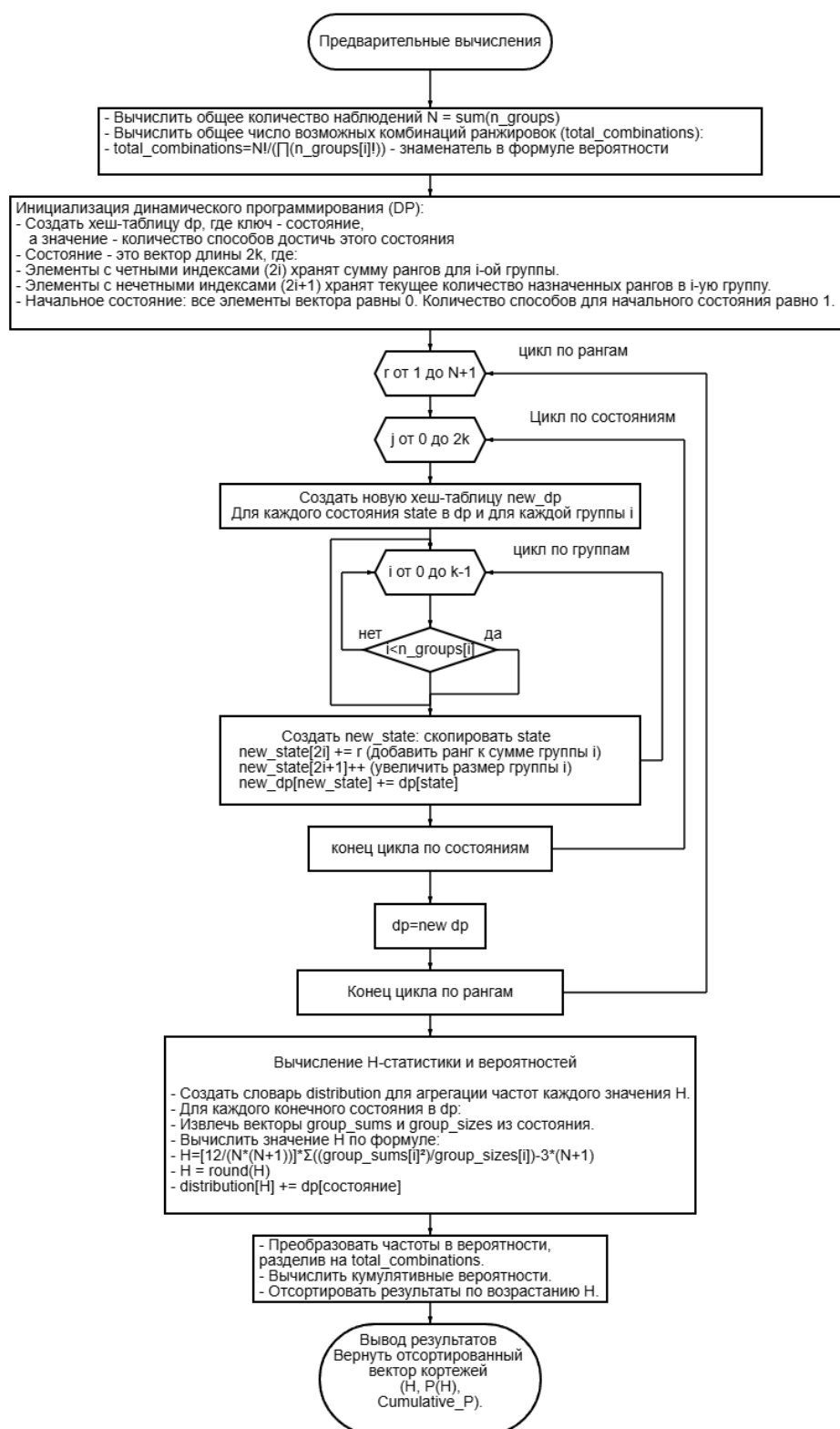


Рисунок 1 – Алгоритм точного вычисления распределения статистики Краскела-Уоллиса с помощью динамического программирования

Figure 1 – Algorithm for calculating the distribution of the Kruskal-Wallis statistic using dynamic programming

1. Систематизированы и реализованы на практике различные вычислительные подходы к точному расчету распределений ранговых непараметрических критериев. В их числе методы, основанные на точных аналитических решениях и производящих функциях, рекуррентные формулы, алгоритмы полного перебора, метод Монте-Карло и, что наиболее важно, метод динамического программирования. Разработанный программный комплекс позволяет рассчитывать полную функцию распределения, критические значения и p -value для широкого набора статистик.

2. Наибольший практический интерес представляют алгоритмы для критериев, не имеющих замкнутого аналитического решения. Для таких случаев была детально проработана и реализована методика, включающая генерацию последовательностей рангов, расчет статистик для каждой из них и последующую агрегацию результатов для построения функции распределения.

3. Результаты выполненных вычислительных экспериментов позволяют сделать однозначный вывод о том, что алгоритмы на основе динамического программирования демонстрируют максимальную эффективность при генерации точных распределений. Наблюдаемое преимущество выражается в сокращении времени расчетов на порядки по сравнению с методом простого перебора. Интересным наблюдением является тот факт, что при использовании ДП разрыв в производительности между реализациями на C++ и Python оказывается не столь катастрофическим, как при прямом переборе, что открывает возможности для быстрого прототипирования алгоритмов на Python без критических потерь в скорости.

4. Все разработанные алгоритмы и программы (реализованные на C++, Python, JavaScript и VBA) опубликованы в открытом доступе, что способствует воспроизводимости результатов и дальнейшему развитию методов. Сравнительный анализ подтвердил ожидаемое лидерство C++ с точки зрения чистого быстродействия, что является критически важным фактором при работе с вычислительно сложными задачами расчета точных распределений.

СПИСОК ИСТОЧНИКОВ / REFERENCES

1. Агамиров Л.В., Агамиров В.Л., Вестяк В.А. Численные методы и алгоритмы расчета точных распределений непараметрических критериев проверки статистических гипотез. *Вестник Московского авиационного института*. 2013;20(4):212–218.
Agamirov L.V., Agamirov V.L., Vestyak V.A. Numerical Methods and Algorithms of Calculation of Exact Distributions of Non-Parametrical Criteria Statistical Hypotheses. *Aerospace MAI Journal*. 2013;20(4):212–218. (In Russ.).
2. Van de Wiel M.A. The Probability Generating Function of the Freund-Ansari-Bradley Statistic. In: *Memorandum COSOR: Volume 9711*. Eindhoven: Technische Universiteit Eindhoven; 1997. P. 6–13.
3. Van de Wiel M.A. Exact Distributions of Multiple Comparisons Rank Statistics. *Journal of the American Statistical Association*. 2002;97(460):1081–1089. <https://doi.org/10.1198/016214502388618898>
4. Choi W., Lee J.W., Huh M.-H., Kang S.-H. An Algorithm for Computing the Exact Distribution of the Kruskal-Wallis Test. *Communications in Statistics – Simulation and Computation*. 2003;32(4):1029–1040. <https://doi.org/10.1081/SAC-120023876>
5. Meyer J.P., Seaman M.A. A Comparison of the Exact Kruskal-Wallis Distribution to Asymptotic Approximations for All Sample Sizes up to 105. *The Journal of Experimental Education*. 2013;81(2):139–156. <https://doi.org/10.1080/00220973.2012.699904>
6. Холлендер М., Вулф Д. *Непараметрические методы статистики*. Москва:

- Финансы и статистика; 1983. 518 с.
Hollander M., Wolfe D. *Nonparametric statistical methods*. Moscow: Finansy i statistika; 1983. 518 p. (In Russ.).
7. Odiase J.I., Ogbonmwan S.M. JMASM20: Exact Permutation Critical Values for The Kruskal-Wallis One-Way ANOVA. *Journal of Modern Applied Statistical Methods*. 2005;4(2). <https://doi.org/10.22237/jmasm/1130804820>
 8. Spurrier J.D. On the Null Distribution of the Kruskal-Wallis Statistic. *Journal of Nonparametric Statistics*. 2003;15(6):685–691. <https://doi.org/10.1080/10485250310001634719>
 9. Streitberg B., Rohmel J. Exact Distributions for Permutation and Rank Tests: An Introduction to Some Recently Published Algorithms. *Statistical Software Newsletter*. 1986;12(1):10–17.
 10. Divine G.W., Norton H.J., Barón A.E., Juarez-Colunga E. The Wilcoxon-Mann-Whitney Procedure Fails as a Test of Medians. *The American Statistician*. 2018;72(3):278–286. <https://doi.org/10.1080/00031305.2017.1305291>
 11. Hothorn T., Hornik K., van de Wiel M.A., Zeileis A. Implementing a Class of Permutation Tests: The Coin Package. *Journal of Statistical Software*. 2008;28(8):1–23. <https://doi.org/10.18637/jss.v028.i08>
 12. Антипина Н.М., Захаров В.Н., Протасов Ю.М., Юров В.М. Непараметрический критерий различия для двух связанных выборок в табличном редакторе MS Excel. *Вестник Московского государственного областного университета. Серия: Экономика*. 2021;(2):47–55. <https://doi.org/10.18384/2310-6646-2021-2-47-55>
Antipina N.M., Zakharov V.N., Protasov Yu.M., Yurov V.M. Non-Parametric Criterion of Difference for Two Related Samples in Table Editor MS Excel. *Bulletin of Moscow Region State University. Series: Economics*. 2021;(2):47–55. (In Russ.). <https://doi.org/10.18384/2310-6646-2021-2-47-55>
 13. Lehmann E.L. *Testing Statistical Hypothesis*. New York: Wiley; 1986. 600 p.
 14. Хеттманспергер Т. *Статистические выводы, основанные на рангах*. Москва: Финансы и статистика; 1987. 334 с.
Hettmansperger Th. *Statistical Inference Based on Ranks*. Moscow: Finansy i statistika; 1987. 334 p. (In Russ.).
 15. Кобзарь А.И. *Прикладная математическая статистика. Для инженеров и научных работников*. Москва: ФИЗМАТЛИТ; 2006. 816 с.
 16. Iman R.L., Davenport J.M. New approximations to the exact distribution of the kruskal-wallis test statistic. *Communications in Statistics – Theory and Methods*. 1976;5(14):1335–1348. <https://doi.org/10.1080/03610927608827446>
 17. Mood A.M. On the Asymptotic Efficiency of Certain Nonparametric Two-Sample Tests. *The Annals of Mathematical Statistics*. 1954;25(3):514–522.
 18. Ansari A.R., Bradley R.A. Rank-Sum Tests for Dispersions. *The Annals of Mathematical Statistics*. 1960;31(4):1174–1189.

ИНФОРМАЦИЯ ОБ АВТОРАХ / INFORMATION ABOUT THE AUTHORS

Агамиров Левон Владимирович, доктор технических наук, профессор, Национальный исследовательский университет «МЭИ», Московский технический университет связи и информатики, Московский авиационный институт, Москва, Российская Федерация.
e-mail: itno_agamirov@mail.ru
ORCID: [0009-0009-6909-9399](https://orcid.org/0009-0009-6909-9399)

Levon V. Agamirov, Doctor of Engineering Sciences, Professor, National Research University "MPEI", Moscow Technical University of Communications and Informatics, Moscow Aviation Institute, Moscow, the Russian Federation.

Агамиров Владимир Леонович, кандидат технических наук, доцент, Московский технический университет связи и информатики, Московский авиационный институт, Москва, Российская Федерация.

e-mail: avhere@yandex.ru

ORCID: [0000-0001-9181-7726](https://orcid.org/0000-0001-9181-7726)

Vladimir L. Agamirov, Candidate of Engineering Sciences, Associate Professor, Moscow Technical University of Communications and Informatics, Moscow Aviation Institute, Moscow, the Russian Federation.

Вестяк Владимир Анатольевич, доктор физико-математических наук, доцент, заведующий кафедрой «Прикладные программные средства и математические методы», Московский авиационный институт, Москва, Российская Федерация.

email: kaf311@yandex.ru

ORCID: [0009-0001-8496-9438](https://orcid.org/0009-0001-8496-9438)

Vladimir A. Vestyak, Doctor of Physical and Mathematical Sciences, Docent, Head of the Department of Applied software tools and mathematical methods, Moscow Aviation Institute, Moscow, the Russian Federation.

Тугова Наталья Владимировна, кандидат технических наук, доцент, заведующий кафедрой «Бизнес-информатика» Московского технического университета связи и информатики, Москва, Российская Федерация.

e-mail: e-natasha@mail.ru

ORCID: [0000-0002-2851-8472](https://orcid.org/0000-0002-2851-8472)

Natalia V. Toutova, Candidate of Engineering Sciences, Docent, Head of the Department of Business Informatics, Moscow Technical University of Communications and Informatics, Moscow, the Russian Federation.

Статья поступила в редакцию 06.01.2026; одобрена после рецензирования 10.02.2026; принята к публикации 18.02.2026.

The article was submitted 06.01.2026; approved after reviewing 10.02.2026; accepted for publication 18.02.2026.