УДК 519.21

Н.Ю. КУЗНЕЦОВ

УСКОРЕННОЕ МОДЕЛИРОВАНИЕ МЕТОДОМ МОНТЕ-КАРЛО КОЛИЧЕСТВА «ХОРОШИХ» ПЕРЕСТАНОВОК НА МНОГОПРОЦЕССОРНОМ КОМПЛЕКСЕ СКИТ-4

Аннотация. Перестановка $(s_0, s_1, \dots, s_{N-1})$ символов $0, 1, \dots, N-1$ называется «хорошей», если набор $(t_0, t_1, \dots, t_{N-1})$, построенный согласно правилу $t_i = i + s_i \pmod N$, $i = 0, 1, \dots, N-1$, также является перестановкой. Предложен метод ускоренного моделирования, реализация которого на многопроцессорном комплексе СКИТ-4 позволяет оценить количество «хороших» перестановок для $N \le 305$ с относительной погрешностью, не превышающей 1%. Приведены оценки количества «хороших» перестановок для $N = 25, 35, \dots, 305$.

Ключевые слова: «хорошая» перестановка, модифицированный метод ускоренного моделирования, несмещенная оценка, выборочная дисперсия, относительная погрешность.

Одной из наиболее сложных классических задач дискретной математики, не потерявших актуальности, является задача перечисления всех полных отображений на алгебраической структуре $(G, +)$ [1, 2]. Отображение $f: G \to G$ называется полным, если $f(\cdot)$ — биекция и отображение $h(x) = x + f(x)$ также биекция. В работе [3] исследовалась сложность проблемы нахождения количества всех полных отображений для различных структур $(G, +)$. Доказано, что для замкнутых структур эта проблема является NP-полной.

Пусть $\bar{s} = (s_0, s_1, \dots, s_{N-1})$ — произвольная перестановка символов $0, 1, \dots, N-1$. Построим новый набор $\bar{t} = (t_0, t_1, \dots, t_{N-1})$ согласно правилу $t_i = i + s_i \pmod N$, $i = 0, 1, \dots, N-1$. Если набор $\bar{t}$ является перестановкой, то исходная перестановка $\bar{s}$ называется «хорошей» (данный термин введен в [4]). В работе [5] такое отображение названо сильным полным отображением.

Исчерпывающий анализ основ применения «хороших» перестановок в криптографии приведен в [4]; принципы использования таких перестановок в роторных шифровальных системах описаны в [6].

Несмотря на многочисленные попытки [4, 7–11], до сих пор не удалось найти аналитический подход к решению задачи нахождения количества M_N «хороших» перестановок. Известно только, что $M_N = 0$ для четных значений N. Использование быстрых методов направленного перебора и современных многопроцессорных вычислительных комплексов позволило вычислить M_N для всех нечетных значений $N \le 25$. Так, значения M_N для $N \le 19$ приведены в [4], для $N \le 23$ — в [11], для $N \le 25$ — в [12]. При дальнейшем увеличении N на 2 время вычислений возрастает на два порядка, что делает малоперспективным нахождение точных значений M_N при нечетных $N > 25$ даже при быстром развитии вычислительных мощностей. Поэтому основной акцент в исследованиях M_N делается на приближенных методах расчета, в частности асимптотических и статистических. Среди близких по тематике публикаций отметим работы [13–16].

Поскольку общее количество перестановок равно $N!$, сформулированная выше детерминированная задача нахождения M_N эквивалентна вероятностной задаче оценки вероятности P_N того, что выбранная случайным образом перестановка является «хорошей», в этом случае $P_N = M_N / N!$.

В работах [17, 18] для оценки P_N предложен принципиально новый подход, основанный на использовании алгоритма ускоренного моделирования вероят-

ности P_N. Этот алгоритм позволил строить несмещенные оценки для P_N при $N \le 205$ с относительной погрешностью, не превышающей 5%. Дальнейшее совершенствование метода (см. следующий раздел статьи) позволяет ещё от 10 до 100 раз сократить затраты времени на оценку вероятности P_N при заданной относительной погрешности. Совместное использование ускоренного моделирования и многопроцессорного вычислительного комплекса СКИТ-4 при Институте кибернетики им. В.М. Глушкова НАН Украины позволяет вычислять P_N при $N = 305$ с относительной погрешностью 1%. Вероятность P_N можно представить в виде

$$P_N = e^{\alpha_N - \beta_N N}. \tag{1}$$

Была выдвинута гипотеза (см. [4]), что

$$\beta_N \to 1,\ \alpha_N \to \text{const при } N \to \infty. \tag{2}$$

Результаты вычислений, проведенных в настоящей работе, показывают, что полученные статистические данные не противоречат гипотезе (1), (2).

МОДИФИЦИРОВАННЫЙ МЕТОД УСКОРЕННОГО МОДЕЛИРОВАНИЯ

В работе [17] предложен алгоритм ускоренного моделирования, позволяющий направленным образом размещать символы $0, 1, \ldots, N-1$ на одной из N позиций (усовершенствованный вариант алгоритма см. [18]). Основная цель — добиться существенного возрастания вероятности построения «хорошей» перестановки. Несмещенность оценки достигается за счет соответствующего нормирующего множителя. Предложенный в этом разделе подход позволяет еще более повысить данную вероятность. В то же время достижение значений вероятности выше 0,8 нецелесообразно, поскольку это ведет к значительному росту вычислительных затрат. Предлагаемый ниже алгоритм состоит из трех этапов.

Введем целочисленные параметры r и m, $1 \le r \le m \le N-1$, которые в дальнейшем считаем фиксированными (об их рациональном выборе далее в этом разделе). На первом этапе располагаем символы на позициях с номерами $m, \ldots, N-1$. Поскольку вероятность получения «плохой» перестановки весьма мала, используем самый простой (наиболее быстрый) алгоритм. По мере заполнения позиций существенно возрастает вероятность получения «плохой» перестановки. Поэтому на втором этапе проводится более тщательный анализ в целях определения символов, которые могут быть размещены на тех или иных позициях. При этом выполняется анализ, не является ли частично заполненная перестановка тупиковой, из которой нельзя получить ни одной «хорошей» перестановки. На данном этапе заполняются $m - r$ позиций. На третьем этапе (детерминированный алгоритм) последние r позиций заполняются таким образом, чтобы получить все возможные «хорошие» перестановки (определяется количество «хороших» перестановок, которые можно получить из частично заполненной перестановки, построенной на втором этапе). Введение соответствующих нормирующих множителей позволяет достичь несмещенности оценки.

Текущим состоянием перестановки назовем N-мерный вектор $\bar{s} = (s_0, s_1, \ldots, s_{N-1})$, где $s_i \in \{-1, 0, 1, \ldots, N-1\}$. Запись $s_i = -1$ означает, что позиция i еще не занята ни одним из символов; если $s_i = k \ge 0$, то на позиции i размещен символ k. Состояние $\bar{s}$ назовем «тупиковым», если ни при каком размещении символов из этого состояния нельзя получить «хорошую» перестановку.

Обозначим:

$$\nu_i(\bar{s}) = \begin{cases} 1, & \text{если } i = s_k \neq -1 \text{ для некоторого } k, \\ 0 & \text{в противном случае, } i = 0, 1, \ldots, N-1; \end{cases}$$

$$\mu_j(\bar{s}) = \begin{cases} 1, & \text{если } j = k + s_k \pmod N,\ s_k \neq -1, \text{ для некоторого } k, \\ 0 & \text{в противном случае, } \quad j = 0, 1, \ldots, N-1. \end{cases}$$

Величина $\nu_i(\bar{s})$ является индикатором того, что при состоянии $\bar{s}$ i-й символ уже расположен на одной из позиций вектора $\bar{s}$, $\mu_j(\bar{s})$ — индикатор того, что при состоянии $\bar{s}$ j-й символ уже расположен на одной из позиций вектора $\bar{t}$. В приведенном ниже алгоритме строится оценка $\hat{P}_{N1}$ в одной реализации для вероятности P_N. Заметим, что генерирование независимых равномерно распределенных на $[0,1]$ случайных величин осуществляется с помощью стандартной техники построения последовательности псевдослучайных чисел.

Этап 1 (алгоритм телефонного диска).

1. Положим $\hat{P}_{N1} = 1$ — начальное значение оценки. Без ограничения общности в качестве начального состояния $\bar{s}$ можно выбрать: $\bar{s} = (0, -1, \ldots, -1)$, $\nu_i(\bar{s}) = 0$, $\mu_j(\bar{s}) = 0$, $i, j = 1, \ldots, N-1$, $\nu_0(\bar{s}) = 1$, $\mu_0(\bar{s}) = 1$.

2. Предположим, что l (номер позиции) последовательно принимает значения $m+1,\ m+2, \ldots, N-1$. Определим множество символов, которые можно расположить на позиции l:

$$A_l(\bar{s}) = \{i:\ \nu_i(\bar{s}) = 0,\ \mu_k(\bar{s}) = 0,\ k = l + i \pmod N\}. \tag{3}$$

Обозначим $|A_l(\bar{s})|$ количество символов во множестве $A_l(\bar{s})$. Если $|A_l(\bar{s})| = 0$, то этап 1 (вместе с ним и алгоритм) окончен и в качестве оценки имеем $\hat{P}_{N1} = 0$ (в этой реализации не удалось построить «хорошей» перестановки). Если $|A_l(\bar{s})| > 0$, то с вероятностью $\dfrac{1}{|A_l(\bar{s})|}$ выбираем один из символов множества $A_l(\bar{s})$. Если это символ i, то полагаем

$$\hat{P}_{N1} := \hat{P}_{N1}\,\frac{|A_l(\bar{s})|}{N+m-l},\ s_l = i,\ \nu_i = 1,\ \mu_k = 1,$$

где $k = l + i \pmod N$ (символ := означает, что новое значение $\hat{P}_{N1}$ вычисляется как произведение старого значения $\hat{P}_{N1}$ на соответствующий множитель). Заметим, что $N + m - l$ — это количество символов, которые еще не были размещены при состоянии $\bar{s}$. Увеличивая l на единицу, повторяем шаг 2 алгоритма до тех пор, пока не будет заполнена $(N-1)$-я позиция.

Этап 2 (алгоритм направленного перебора).

Пусть $\bar{s}$ — некоторое состояние, в котором незаполненными являются L позиций. Данному состоянию однозначно соответствует вектор $\overline{W}$ следующей структуры:

$$\overline{W} = (\theta_1, \kappa_1, \gamma_1^{(1)}, \ldots, \gamma_{\kappa_1}^{(1)}; \ldots;\ \theta_L, \kappa_L, \gamma_1^{(L)}, \ldots, \gamma_{\kappa_L}^{(L)}), \tag{4}$$

где $\theta_1, \ldots, \theta_L$ — номера незаполненных позиций, κ_l — количество символов, которые могут быть размещены на позиции θ_l, а $\gamma_1^{(l)}, \ldots, \gamma_{\kappa_l}^{(l)}$ — сами символы. В дальнейшем именно вектор $\overline{W}$ будет характеризовать текущее состояние перестановки. Алгоритм на этапе 2 формулируем следующим образом.

1. Если $\bar{s}$ — состояние, построенное в результате работы алгоритма на этапе 1, то формируем вектор $\overline{W}$ вида (4), где $L = m$, $(\theta_1, \ldots, \theta_L) = (1, \ldots, m)$.

2. Пусть $\overline{W}$ — текущее состояние. Вычисляем $l^* = \arg\min\{\kappa_1, \ldots, \kappa_L\}$ (если минимум достигается на нескольких значениях $\{\kappa_j\}$, то выбираем наименьшее значение j). Если $\kappa_{l*} = 0$, то ни один из символов не может быть расположен на позиции l^*. В этом случае этап 2 (а также алгоритм) окончен и в качестве оценки имеем $\hat{P}_{N1} = 0$ (в этой реализации не удалось построить «хорошей» перестановки). Если $\kappa_{l*} > 0$, то с вероятностью $\dfrac{1}{\kappa_{l*}}$ выбираем один из символов $\gamma_1^{(l*)}, \ldots, \gamma_{\kappa_{l*}}^{(l*)}$. Если им является символ i, то располагаем его на позиции θ_{l*} и строим новое состояние $\overline{W}$ с учетом того, что позиция θ_{l*} уже занята. Кроме того, исключаются из рассмотрения все $\gamma_j^{(l)}$ такие, что $\gamma_j^{(l)} = i$ либо $\gamma_j^{(l)} + \theta_l = i + \theta_{l*} \pmod N$. При этом пересчитываем нормирующий множитель:

$$\hat{P}_{N1} := \hat{P}_{N1} \frac{\kappa_{l*}}{L}.$$

Алгоритм шага 2 этапа 2 повторяем $m - r$ раз. В результате получим состояние $\overline{W}$ вида (4), в котором будет ровно $L = r$ незаполненных позиций.

Этап 3 (детерминированный алгоритм перечисления всех «хороших» перестановок, которые можно получить из состояния $\overline{W}$). Каждое состояние вида (4) может порождать не более l^* новых состояний (некоторые из этих состояний могут оказаться тупиковыми). Вновь полученные состояния порождают ряд новых состояний и т.д. Данный рекуррентный алгоритм повторяется $r-1$ раз. Если на одном из промежуточных шагов алгоритма все состояния окажутся тупиковыми, то $\hat{P}_{N1} = 0$. Если удалось пройти все $r-1$ шагов, то получаем $R > 0$ состояний (в данной реализации построено R «хороших» перестановок). В этом случае алгоритм окончен и в качестве несмещенной оценки, построенной в одной реализации для P_N, выбираем

$$\hat{P}_{N1} := \hat{P}_{N1} \cdot \frac{R}{r!}$$

(во всех случаях при построении оценки $\hat{P}_{N1}$ использовалось классическое определение вероятности, поэтому оценка является несмещенной).

Описанный алгоритм зависит от двух параметров: r и m, $1 \le r \le m \le N-1$. Фактически каждая пара (r, m) определяет свой алгоритм построения несмещенной оценки. Поэтому построение оценок для P_N при различных (r, m) служит дополнительной проверкой достоверности результата. Известно, что время, необходимое на построение несмещенной оценки с заданной точностью, пропорционально произведению среднего времени одной реализации на дисперсию оценки. Задача оптимального выбора r и m состоит в минимизации данного произведения. Поскольку с возрастанием как r, так и m увеличивается вероятность построения хотя бы одной «хорошей» перестановки, логично предположить, что дисперсия оценки монотонно не возрастает по r и m. В то же время с увеличением как r, так и m возрастает объем обрабатываемой информации, а следовательно, и среднее время одной реализации. Еще одна принципиальная сложность определения оптимальных значений r и m состоит в том, что точное значение дисперсии оценки неизвестно, поэтому приходится использовать выборочную дисперсию, которая не отличается особой устойчивостью. Практической рекомендацией является проведение предварительных вычислений (при относительно небольшом количестве реализаций), позволяющих найти значения r и m, близкие к оптимальным. Затем для нескольких пар (r, m) нужно построить оценки с заданной точностью и выбрать ту пару, для которой затраченное время является минимальным. При этом можно установить определенную закономерность выбора r и m в зависимости от N. Данный подход использован при получении численных данных, приведенных в следующем разделе.

ЧИСЛЕННЫЕ РЕЗУЛЬТАТЫ

Представим оценки вероятности P_N, $N = 25, 35, \ldots, 305$, построенные описанным выше методом на многопроцессорном вычислительном комплексе СКИТ-4 (табл. 1). Все оценки построены с использованием только 20 процессоров. Обозначим:

ε — относительная погрешность оценки (в процентах);

r, m — параметры метода, при которых построены оценки;

$\hat{P}_N$ и $\hat{M}_N$ — оценки, построенные для P_N и M_N с достоверностью 0,99 и относительной погрешностью ε;

$\hat{C}_N$ — оценка относительной среднеквадратической погрешности (отношение корня выборочной дисперсии к оценке $\hat{P}_N$);

$\hat{K}_N$ — оценка частоты «успешных реализаций», т.е. оценка пропорции количества реализаций, в которых удалось построить «хорошие» перестановки (в процентах);

$\hat{T}_N$ — время, затраченное суперкомпьютером СКИТ-4 на построение оценки $\hat{P}_N$; например, запись *ww*:*xx*:*yy*:*zz* означает, что было использовано *ww* суток, *xx* часов, *yy* минут и *zz* секунд.

Таблица 1

N	ε, %	r	m	$\hat{P}_N$	$\hat{M}_N$	$\hat{C}_N$	$\hat{K}_N$, %	$\hat{T}_N$
25	0,01	9	12	$2{,}682 \cdot 10^{-9}$	$4{,}161 \cdot 10^{16}$	1,39	62,3	19,51
35	0,01	11	15	$2{,}008 \cdot 10^{-12}$	$2{,}075 \cdot 10^{27}$	1,89	50,9	01:08:48
45	0,01	12	20	$1{,}326 \cdot 10^{-17}$	$1{,}586 \cdot 10^{39}$	2,55	40,5	02:57:48
55	0,01	14	22	$8{,}117 \cdot 10^{-22}$	$1{,}031 \cdot 10^{52}$	2,97	39,1	06:54:23
65	0,01	15	26	$4{,}727 \cdot 10^{-26}$	$3{,}899 \cdot 10^{65}$	3,70	33,1	13:42:18
75	0,03	16	28	$2{,}660 \cdot 10^{-30}$	$6{,}600 \cdot 10^{79}$	4,81	25,1	02:59:05
85	0,03	17	33	$1{,}453 \cdot 10^{-34}$	$4{,}094 \cdot 10^{94}$	5,49	24,6	05:20:13
95	0,03	18	35	$7{,}800 \cdot 10^{-39}$	$8{,}057 \cdot 10^{109}$	6,63	20,4	09:12:43
105	0,03	19	39	$4{,}117 \cdot 10^{-43}$	$4{,}452 \cdot 10^{125}$	7,48	19,9	15:14:42
115	0,03	20	45	$2{,}137 \cdot 10^{-47}$	$6{,}250 \cdot 10^{141}$	7,54	22,0	21:51:40
125	0,03	21	50	$1{,}104 \cdot 10^{-51}$	$2{,}078 \cdot 10^{158}$	8,65	23,2	01:15:08:35
135	0,1	22	53	$5{,}604 \cdot 10^{-56}$	$1{,}508 \cdot 10^{175}$	9,66	22,9	05:41:57
145	0,1	23	53	$2{,}831 \cdot 10^{-60}$	$2{,}279 \cdot 10^{192}$	10,61	20,1	08:11:03
155	0,1	23	65	$1{,}414 \cdot 10^{-64}$	$6{,}773 \cdot 10^{209}$	13,06	18,0	12:32:45
165	0,1	24	69	$7{,}060 \cdot 10^{-69}$	$3{,}829 \cdot 10^{227}$	13,00	19,4	16:23:12
175	0,1	25	71	$3{,}519 \cdot 10^{-73}$	$3{,}957 \cdot 10^{245}$	13,74	19,9	23:43:02
185	0,1	26	72	$1{,}736 \cdot 10^{-77}$	$7{,}159 \cdot 10^{263}$	14,91	19,9	01:12:57:21
195	0,3	26	91	$8{,}484 \cdot 10^{-82}$	$2{,}199 \cdot 10^{282}$	16,22	19,8	05:17:21
205	0,3	27	88	$4{,}165 \cdot 10^{-86}$	$1{,}132 \cdot 10^{301}$	18,55	19,1	08:06:52
215	0,3	28	89	$2{,}030 \cdot 10^{-90}$	$9{,}416 \cdot 10^{319}$	18,89	20,2	11:35:26
225	0,3	28	111	$9{,}843 \cdot 10^{-95}$	$1{,}240 \cdot 10^{339}$	20,91	19,8	15:13:13
235	0,3	29	108	$4{,}796 \cdot 10^{-99}$	$2{,}555 \cdot 10^{358}$	22,72	20,4	22:25:01
245	0,3	30	103	$2{,}305 \cdot 10^{-103}$	$7{,}942 \cdot 10^{377}$	22,16	19,8	01:03:42:17
255	1	30	129	$1{,}115 \cdot 10^{-107}$	$3{,}736 \cdot 10^{397}$	26,73	20,2	03:52:18
265	1	31	122	$5{,}337 \cdot 10^{-112}$	$2{,}572 \cdot 10^{417}$	24,45	20,4	04:06:24
275	1	31	151	$2{,}573 \cdot 10^{-116}$	$2{,}600 \cdot 10^{437}$	31,16	19,7	07:17:39
285	1	32	143	$1{,}223 \cdot 10^{-120}$	$3{,}724 \cdot 10^{457}$	28,47	20,8	07:33:13
295	1	33	126	$5{,}868 \cdot 10^{-125}$	$7{,}642 \cdot 10^{477}$	31,33	18,3	11:04:14
305	1	34	128	$2{,}810 \cdot 10^{-129}$	$2{,}196 \cdot 10^{498}$	33,70	21,2	22:22:45

Таблица 2

Параметры	Оценка параметров при s						
	25	75	115	155	195	235	275
$\alpha(s)$	5,94215	6,53568	6,80857	7,00970	7,17311	7,31864	7,41937
$\beta(s)$	0,98868	0,99137	0,99252	0,99333	0,99395	0,99448	0,99482

Автор не утверждает, что приведенные значения r и m оптимальны. Однако даже при этих значениях достигается высокая точность оценок при вполне приемлемых затратах времени (наибольшие временны́е расходы — 39 ч для оценки P_{125} с относительной погрешностью 0,03%). Относительная среднеквадратическая погрешность $\hat{C}_N$ с ростом N в целом несколько возрастает (имеются исключения, вызванные либо статистической погрешностью вычислений, либо не самым оптимальным выбором r и m). Легко заметить, что при выборе r и m применялось правило: частота «успешных реализаций» составляет приблизительно 20%.

Как отмечено в начале статьи, была выдвинута гипотеза: если вероятность P_N представить в виде (1), то справедливо соотношение (2). Проверим, не противоречат ли полученные статистические данные этой гипотезе.

Имеются наблюдения $\hat{P}_{25}, \hat{P}_{35}, \hat{P}_{45}, \ldots, \hat{P}_{305}$ (всего 29 оценок). Для оценки асимптотических значений параметров α_N и β_N воспользуемся линейной регрессией, причем оценки данных параметров построим по остатку ряда $\hat{P}_s, \hat{P}_{s+10}, \hat{P}_{s+20}, \ldots, \hat{P}_{305}$ при различных значениях s. Соответствующие оценки обозначим $\hat{\alpha}(s)$ и $\hat{\beta}(s)$. Воспользовавшись методом наименьших квадратов, получим оценки, представленные в табл. 2.

С увеличением s наблюдаем монотонное возрастание $\alpha(s)$ и $\beta(s)$, при этом в обоих случаях это рост с замедлением, что позволяет предположить существование пределов для $\alpha(s)$ и $\beta(s)$, причем предел для $\alpha(s)$ является конечным, а для $\beta(s)$ — весьма близок к единице. Таким образом, результаты вычислений показывают, что полученные статистические данные не противоречат гипотезе (1), (2).

Приведенный в предыдущем разделе алгоритм включает три этапа. Первые два основаны на случайном выборе символа и позиции, а третий — детерминированный. Детерминированную часть алгоритма можно использовать, пропуская первые два этапа. В этом случае получим точное значение количества «хороших» перестановок. Точные значения M_N, вычисленные на СКИТ-4 при $N = 9, 11, \ldots, 23$, приведены в табл. 3. Для сравнения указаны также оценки, полученные на персональном компьютере (фактически один процессор).

Отметим близость статистических оценок к точным значениям M_N. Если при значениях $N \le 15$ точное значение вычисляется практически мгновенно, то с увеличением N время вычислений резко возрастает (более 10 суток на 100 процессорах при $N = 23$; для построения приближенной оценки с $\varepsilon = 0{,}1\%$ потребо-

Таблица 3

N	M_N	Количество процессоров	Время (СКИТ-4)	Оценка на ПК (относительная погрешность 0,1%)	Время (ПК)
9	2025	2	0,00	$2{,}0242 \cdot 10^3$	9
11	37 851	2	0,00	$3{,}7878 \cdot 10^4$	15
13	1 030 367	2	0,00	$1{,}0309 \cdot 10^6$	36
15	36 362 925	2	0,00	$3{,}6371 \cdot 10^7$	01:11
17	1 606 008 513	2	02:08	$1{,}6069 \cdot 10^9$	02:37
19	87 656 896 891	10	25:36	$8{,}7677 \cdot 10^{10}$	03:06
21	5 778 121 715 415	50	4:39:28	$5{,}7808 \cdot 10^{12}$	06:35
23	452 794 813 251 584	100	10:08:10:40	$4{,}5328 \cdot 10^{14}$	15:47

валось менее 16 мин на одном процессоре). При дальнейшем увеличении N вычислительные расходы возрастают на два порядка. Этим можно объяснить, что до сих пор не удалось вычислить точное значение количества «хороших» перестановок для $N = 27$.

Таким образом, предложенный метод является реальным инструментом, позволяющим при относительно небольших затратах времени с высокой точностью оценивать количество «хороших» перестановок для $N > 300$.

Автор глубоко благодарен академику НАН Украины И.Н. Коваленко за привлечение внимания к указанной проблеме и критические замечания, способствовавшие улучшению статьи.

СПИСОК ЛИТЕРАТУРЫ

1. Сачков В.Н. Введение в комбинаторные методы дискретной математики. — М.: Наука, 1982. — 384 с.
2. Сачков В.Н. Цепи Маркова итерационных систем преобразований // Тр. по дискретной математике. — 2002. — **6.** — С. 165–183.
3. Hsiang J., Hsu D.F., Shieh Y.P. On the hardness of counting problems of complete mappings // Discrete Mathematics. — 2004. — **277.** — P. 87–100.
4. Cooper C., Gilchrist R., Kovalenko I.N., Novakovic D. Deriving the number of “good” permutations, with application to cryptography // Кибернетика и системный анализ. — 1999. — № 5. — С. 10–16.
5. Hsu D.F., Keedwell A.D. Generalized complete mappings, neofields, sequenceable groups and block designs. II // Pacific J. Math. — 1985. — **117.** — P. 291–312.
6. Konheim R. Cryptography: a primer. — Chichester: Wiley, 1991. — 432 p.
7. Cooper C., Kovalenko I.N. An upper bound for the number of complete mappings // Теория вероятностей и мат. статистика. — 1995. — **53.** — С. 69–75.
8. Kovalenko I.N. On an upper bound for the number of complete mappings // Кибернетика и системный анализ. — 1996. — № 1. — С. 81–85.
9. Левитская А.А. Одна комбинаторная задача в классе перестановок над кольцом Z_n вычетов по нечетному модулю n // Проблемы управления и информатики. — 1996. — № 5. — С. 99–108.
10. Новакович Д. Подсчет числа полных отображений для перестановок // Кибернетика и системный анализ. — 2000. — № 2. — С. 106–109.
11. Shieh Y.P. Partition strategies for #*P*-complete problem with applications to enumerative combinatorics: Ph.D. Thesis. — National Taiwan Univ., 2001.
12. http://oeis.org/A003111.
13. Shieh Y.P., Hsiang J., Hsu D.F. On the enumeration of Abelian *k*-complete mappings // Congressus Numerantium. — 2000. — **144.** — P. 67–88.
14. McKay B.D., McLeod J.C., Wanless I.M. The number of transversals in a Latin square // Des. Codes Cryptogr. — 2006. — **40.** — P. 269–284.
15. Cavenagh N.J., Wanless I.M. On the number of transversals in Cayley tables of cyclic groups // Disc. Appl. Math. — 2010. — **158.** — P. 136–146.
16. Stones D.S., Wanless I.M. Compound orthomorphisms of the cyclic group // Finite Fields Appl. — 2010. — **16.** — P. 277–289.
17. Кузнецов Н.Ю. Применение ускоренного моделирования к нахождению количества «хороших» перестановок // Кибернетика и системный анализ. — 2007. — № 6. — С. 80–89.
18. Кузнецов Н.Ю. Оценка количества «хороших» перестановок модифицированным методом ускоренного моделирования // Кибернетика и системный анализ. — 2008. — № 4. — С. 101–109.

Поступила 01.07.2015