ПРАКТИЧЕСКАЯ ЭФФЕКТИВНОСТЬ МЕТОДОВ ПОРОЖДЕНИЯ СЛУЧАЙНЫХ КОНЕЧНЫХ КВАЗИГРУПП

Р. А. Жигляев (Москва)

Квазигруппы имеют применение в построении различных шифров и хэш-функций [1]. Для большей стойкости шифров рекомендуется использовать достаточно большие по размеру квазигруппы. На практике, большой размер квазигрупп может вызвать ряд проблем, таких как долгая работа алгоритма генерации или невозможность хранения большого латинского квадрата в памяти компьютера. Выходом из ситуации может быть функциональное задание квазигрупп. При этом, получаемое таким образом множество квазигрупп необходимо сделать как можно больше.

Была проведена работа по программной реализации ряда алгоритмов генерации конечных квазигрупп. Проводилось сравнение алгоритмов по паре критериев — реальное время генерации квазигрупп и мощность получаемых множеств.

Определение. Конечное множество Q, на котором задана бинарная операция умножения f_Q: $Q \times Q \to Q$, такая, что для любых элементов $a, b \in Q$ уравнения $f_Q(a, x) = b$ и $f_Q(y, a) = b$ однозначно разрешимы в Q, называется *конечной квазигруппой*. Операцию f_Q будем называть квазигрупповой.

В дальнейшем слово «конечная» будем опускать и будем говорить только о конечных квазигруппах порядка $N = k^n$, где k и n — некоторые натуральные числа и $k \geq 2$. Элементы квазигруппы $(q_0, ..., q_{N-1})$ будем отождествлять с числами $(0, ..., N-1)$ и с их записями в k-ичной системе счисления.

В работе [2] был предложен алгоритм генерации равномерного распределения на множестве всех квазигрупп заданного порядка. Далее этот алгоритм будем называть JM. В [3] предложен метод генерации равномерного распределения на множестве всех правильных семейств заданного порядка n и значности k, а также формула, по которой из правильного семейства можно получить «случайную» квазигруппу. Метод, при котором будет генерироваться случайное правильное семейство, а из него случайная квазигруппа, обозначим PF. В работе [4] рассматриваются два класса конструкций — регистры сдвига с обратной связью и обобщенные сети Фейстеля, а также приводятся условия, при которых эти конструкции задают перестановки с полным дифференциалом, а следовательно и квазигрупповые операции. Алгоритмы порождения этих двух классов и преобразования их в квазигруппу обозначим SR и FN соответственно. Все алго-

ритмы были реализованы в программе, исходный код которой можно найти по ссылке https://github.com/Gerror/Quasigroup. Приведем утверждение, показывающее мощности порождаемых множеств.

Утверждение. *а) Алгоритм JM задает равномерное порождение на множестве всех квазигрупп заданного порядка, мощность которого* $\left((1+\alpha_N)\frac{k^n}{e^2}\right)^{k^{2n}}$, *где* $\alpha_N \to 0$ *при* $N \to \infty$. [2, 5]

б) Для фиксированного правильного семейства порядка n *в* k*-значной логике число попарно различных квазигрупп равно* M^{k^2}, *где* M *— число принимаемых семейством значений. Достижимая оценка сверху на всё выражение —* $(k^{k^2})^{n-1}$. *Найдется константа* A, *такая, что число правильных семейств удовлетворяет неравенству* $n^{A2^n} \leq P(n)$. [3]

в) Обобщенные сети Фейстеля с использованием алгоритма FN позволяют задать равномерное распределение на множестве из $(k!)^2$ *попарно различных квазигрупп.* [4]

г) Регистры сдвига с обратной связью с использованием алгоритма SR позволяют задать равномерное распределение на множестве из $(k!)^{k^{n-2}}$ *попарно различных квазигрупп.* [4]

В экспериментах генерировались от 100 до 1000 квазигрупп заданного порядка и замерялось среднее время порождения одного объекта. В случае, когда квазигруппы задавались функционально, перебирались различные варианты значений k и n, при которых можно получить нужный порядок. Результаты экспериментов представлены в таблице.

Порядок	**JM**	**PF**	**FN**	**SR**
16	0.00037	k = 2, n = 4: 0.00501 k = 4, n = 2: 0.0012	0.00004	k = 2, n = 4, 0.0001 k = 4, n = 2 0.00004
32	0.00301	0.06708	-	0.00038
64	0.02394	k = 2, n = 6: 0.85705 k = 4, n = 3: 0.19669 k = 8, n = 2: 0.0823	0.00023	k = 2, n = 6: 0.00165 k = 4, n = 3: 0.00065 k = 8, n = 2: 0.00037
81	0.04790	k = 3, n = 4: 0.76527 k = 9, n = 2: 0.17597	0.00034	k = 3, n = 4: 0.00151 k = 9, n = 2 0.00052

128	0.19671	10.5716	-	0.00764
256	1.52354	k = 2, n = 8: 134.987 k = 4, n = 4: 31.9878 k = 16, n = 2: 8.53718	0.00327	k = 2, n = 8: 0.0342 k = 4, n = 4: 0.01439 k = 16, n = 2: 0.005

В первом столбце указан порядок порождаемых квазигрупп, в остальных столбцах — время в секундах для соответствующих методов. Для функциональных квазигрупп время в таблице указано с учетом вычисления всего латинского квадрата.

Можно заметить, что наиболее эффективными по времени генерации оказываются квазигруппы на основе сетей Фейстеля. Время работы этого алгоритма многократно меньше, чем у остальных. В текущей реализации хуже всего себя показывает алгоритм генерации правильного семейства с последующей генерации квазигруппы. С учетом того, что алгоритм Джейкобсона-Мэтьюза дает все квазигруппы, а также работает быстрее, он является более предпочтительным вариантом. Стоить отметить, что для одного и того же порядка все рассматриваемые функциональные семейства работают ощутимо быстрее, если выбрать k максимально возможным, а n наименьшим. Однако, этот же подход может уменьшить мощность множества порождаемых объектов.

Список литературы

1. Глухов М. М., О применениях квазигрупп в криптографии // Прикладная дискретная математика. — 2008. — № 2. — С. 28–32.

2. Jacobson M. T.,Matthews P., Generating uniformly distributed random Latin squares // Journal of Combinatorial Designs. — 1996. — Vol. 4, no. 6. — P. 405–437

3. Galatenko A. V., Pankratiev A. E., Staroverov V. M. Generation of proper families of functions // Lobachevskii Journal of Mathematics. — 2022. — Vol. 43, no. 3. — P. 571–581

4. Галатенко А. В., Носов В. А., Панкратьев А. Е. Построение подстановок с полным дифференциалом // Мат-лы XIII Международного семинара «Дискретная математика и ее приложения» имени академика О. Б. Лупанова. — М.: Изд-во механико-математического факультета МГУ, 2019. — С. 317–319.

5. Ryser H. J., Permanents and systems of distinct representatives // Proc. of the Conference on Combinatorial mathematics and its applications. — University of North Carolina, 1967. — P. 55–70

DOI: 10.20948/dms-2022-83