ТРУДЫ МАТЕМАТИЧЕСКОГО ИНСТИТУТА ИМ. В. А. СТЕКЛОВА, 1997, т. 218, с. 122–133

УДК 510.52

О сложности интегрирования рациональных дробей[1]

©1997 г. С. Б. Гашков

Поступило в феврале 1997 г.

1. Введение. Одним из важнейших и наиболее часто применяемых алгоритмов анализа является алгоритм интегрирования рациональных функций. Как известно, интеграл от рациональной функции представляется в виде суммы рациональной и логарифмической части, причем рациональную часть можно выделить только с помощью арифметических операций. Алгоритм для этого был найден Остроградским в 1844 г. (см. [1]). Другой алгоритм для той же цели был предложен спустя 20 лет Эрмитом (см. [3]). В отечественной литературе алгоритм выделения рациональной части интеграла от рациональной функции называется алгоритмом Эрмита–Остроградского, но излагается фактически обычно алгоритм Эрмита. В зарубежной литературе обычно имя Остроградского не упоминают, хотя его статья [1] была опубликована в Петербурге на французском языке.

После открытия А. А. Карацубой в 1962 г. быстрого алгоритма умножения чисел [4] (о том, как это произошло, он сам интересно написал в недавней статье [5]) возник интерес к оценке сложности алгоритмов анализа и алгебры и построению быстрых модификаций этих алгоритмов. Были найдены различные быстрые алгоритмы для основных арифметических операций с числами и полиномами в работах [6–10] и др., быстрые алгоритмы для вычисления значений полиномов и их интерполяции (например, [11]), быстрые алгоритмы для вычисления определителей и решения систем линейных уравнений (после [12] об этом написаны десятки работ), и много других результатов, среди которых даже те, которые можно рассматривать как полиномиальные алгоритмы, невозможно перечислить в короткой статье (см., например, [13–15] и указанную там литературу).

В [16] было показано, как модернизировать алгоритм Эрмита, чтобы он работал со сложностью $O(n \log^2 n)$, где n — степень знаменателя интегрируемой дроби (которую можно предполагать правильной). Этот алгоритм состоит из четырех подалгоритмов: бесквадратной факторизации знаменателя дроби, разложения ее на простейшие дроби, интегрирования полученных простейших дробей методом по частям и сложения рациональных частей полученных интегралов и подынтегральных выражений, которые тоже являются рациональными функциями. Каждый из этих алгоритмов имеет самостоятельный интерес и многочисленные применения, особенно первые два и последний. Все эти алгоритмы имеют по порядку одну и ту же оценку сложности.

В [16] приведены подробно только оценки сложности первого и третьего из них, так как оценки для остальных двух алгоритмов получены в [17, 18]. В указанных трех работах изложена довольно обширная история вопроса, которую мы опускаем.

Но там не написано, что на самом деле эта история начинается с М. В. Остроградского, так как алгоритм бесквадратной факторизации [16] содержится в [1] (что, вероятно, неизвестно ав-

[1]Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (проект 96-01-01068).

тору [16]). Этот алгоритм несколько сложнее для понимания, чем стандартный алгоритм (восходящий к голландскому математику XVIII в. Гудде), но все же достаточно прост для его изложения первокурсникам и заслуживает более широкого распространения и применения.

2. Оценка сложности алгоритма интегрирования рациональных функций и его подалгоритмов. Оценку сложности интегрирования правильных рациональных дробей степени n можно теперь предложить в виде

$$\lceil \log_2 k \rceil \bigl(21\,\mathrm{M}(n) + 11n\bigr) + 24\,\mathrm{M}(n) + 26n + 4\,\mathrm{GD}(n),$$

где k — число бесквадратных множителей в факторизации Остроградского, $\mathrm{M}(n)$ — оценка сложности умножения двух многочленов степени n, удовлетворяющая условию супераддитивности: $\mathrm{M}(x) + \mathrm{M}(y) \le \mathrm{M}(x+y)$, и $\mathrm{GD}(n)$ — такая же оценка сложности вычисления НОД двух многочленов. Как известно (см., например, [13]), в качестве $\mathrm{M}(n)$ можно взять $Cn\log n$, а в качестве $\mathrm{GD}(n)$ можно взять $C\,\mathrm{M}(n)\log n$ при большом C (это было доказано в [19, 20]).

Сложность бесквадратной факторизации многочлена P степени n при этом можно оценить как $2\,\mathrm{GD}(n)+7\,\mathrm{M}(n)+6n$, сложность разложения правильной дроби со знаменателем P на простейшие дроби — как

$$\mathrm{GD}(n) + \lceil \log_2 k \rceil \Bigl(17\,\mathrm{M}(n) + \frac{17n}{2}\Bigr) + 8\,\mathrm{M}(n) + 5n,$$

сложность интегрирования получившихся при этом простейших дробей — как

$$3\,\mathrm{M}(n) + 6\,\mathrm{M}(p) + \mathrm{GD}(n-p) + 15p,$$

где $p = \deg(P, dP/dx)$, сложность суммирования полученных при этом интегрировании простейших дробей — как

$$\lceil \log_k t \rceil \Bigl(3\,\mathrm{M}(n) + \mathrm{M}(p) + \frac{5p}{2}\Bigr) + \mathrm{M}(n-p) + 3(n-p).$$

3. Алгоритм Остроградского бесквадратной факторизации многочленов. Изложим алгоритм бесквадратной факторизации в обозначениях самого Остроградского и оценим его сложность (т.е. число выполняемых им операций). Пусть $P = P_1$ — факторизуемый многочлен, положим

$$P_2 = \Bigl(P_1, \frac{dP_1}{dx}\Bigr), \quad Q_1 = \frac{P_1}{P_2}, \quad R_1 = \frac{dP_1/dx}{P_2}, \quad q_1 = \Bigl(Q_1, R_1 - \frac{dQ_1}{dx}\Bigr),$$
$$Q_2 = \frac{Q_1}{q_1}, \quad R_2 = \frac{R_1 - dQ_1/dx}{q_1}, \quad q_2 = \Bigl(Q_2, R_2 - \frac{dQ_2}{dx}\Bigr)$$

и т.д., пока не получим $R_{k+1} = dQ_{k+1}/dx$ (здесь и далее через (f,g) обозначаем НОД многочленов f и g). Тогда $P = q_1 q_2^2 \cdots q_k^k$ — искомое разложение, потому что q_i не имеют кратных корней и взаимно просты. Отметим, что попутно найдено разложение $P = Q_1 Q_2 \cdots Q_k$, где $Q_i = q_i \cdots q_k$, $i = 1, \ldots, k$, — тоже бесквадратные множители, но уже не взаимно простые, а последовательно делящие друг друга.

Сравним этот алгоритм со стандартным алгоритмом. В последнем рекуррентно вычисляется последовательность многочленов по формулам $P_{i+1} = (P_i, dP_i/dx)$ и с использованием теоремы о понижении кратности корня при дифференцировании проверяется, что

$$P_i = q_i q_{i+1}^2 \cdots q_k^{k-i+1} = Q_i Q_{i+1} \cdots Q_k,$$

после чего находится делением, что

$$Q_i = \frac{P_i}{P_{i+1}}, \qquad q_i = \frac{Q_i}{Q_{i+1}}.$$

Обосновывается алгоритм Остроградского с помощью его замечательных тождеств

$$R_{i+1} = \frac{dP_{i+1}/dx}{P_{i+2}} = \frac{R_i - dQ_i/dx}{q_i}, \qquad q_i = \left(Q_i, R_i - \frac{dQ_i}{dx}\right).$$

Для удобства читателя приведем их доказательство, благо оно короткое. Применяя индукцию, получаем, что

$$R_i - \frac{dQ_i}{dx} = \frac{dP_i/dx}{P_{i+1}} - \frac{d(P_i/P_{i+1})}{dx} = \frac{P_i \cdot (dP_{i+1}/dx)}{P_{i+1}^2} = \frac{Q_i \cdot (dP_{i+1}/dx)}{P_{i+1}} = \frac{Q_{i+1}q_i \cdot (dP_{i+1}/dx)}{P_{i+1}},$$

откуда имеем

$$R_{i+1} = \frac{R_i - dQ_i/dx}{q_i} = \frac{Q_{i+1} \cdot (dP_{i+1}/dx)}{P_{i+1}} = \frac{dP_{i+1}/dx}{P_{i+2}}.$$

Так как

$$P_i = q_i q_{i+1}^2 \cdots q_k^{k-i+1}, \qquad R_i = \frac{dP_i/dx}{P_{i+1}}, \qquad Q_i = q_i \cdots q_k,$$

то из теоремы о понижении кратности корня при дифференцировании вытекает, что $(R_i, Q_i) = 1$, откуда следует уже без всякой индукции, что

$$\left(Q_i, R_i - \frac{dQ_i}{dx}\right) = (Q_i, R_{i+1}q_i) = (Q_{i+1}q_i, R_{i+1}q_i) = (Q_{i+1}q_i, R_{i+1}q_i) = q_i(Q_{i+1}, R_{i+1}) = q_i.$$

Заметим еще, что из формул

$$R_i = \frac{dP_i/dx}{P_{i+1}}, \qquad Q_i = \frac{P_i}{P_{i+1}}$$

вытекает, что $\deg R_i = \deg P_i - \deg P_{i+1} - 1 = \deg Q_i - 1$.

Преимущество алгоритма Остроградского перед стандартным алгоритмом очевидно ввиду того, что на каждом шаге итерации он требует нахождения НОД многочленов меньшей степени, чем стандартный алгоритм, и вместо двух делений использует только одно, причем для многочленов опять же меньшей степени. Кроме того, он допускает хорошую оценку сложности при использовании в нем быстрых алгоритмов умножения, деления и вычисления НОД.

4. Оценка сложности алгоритма Остроградского. Она получается очень просто. Положим $\deg Q_i = n_i$, $\deg q_i = m_i$, $\deg P_2 = p = n_2 + \cdots + n_k$, тогда $m_i = n_i - n_{i+1}$, $\deg R_i = n_i - 1$. Сложность деления без остатка многочлена степени m на многочлен степени n оценим как $3(\mathrm{M}(m-n) + m - n) + \mathrm{M}(m)$ (см., например, [21, задачи 17.25, 17.32]), значит, сложность деления двух разных многочленов степени не выше m на один и тот же многочлен степени n оценивается как $3(\mathrm{M}(m-n) + m - n) + 2\,\mathrm{M}(m)$ (см., например, решения задач 17.25, 17.26

в [21]), тогда нужная оценка имеет вид

$$n + \mathrm{GD}(n) + 3\,\mathrm{M}(n-p) + 3n - 3p + 2\,\mathrm{M}(n) +$$
$$+ \sum_{i=1}^{k}\bigl(\mathrm{GD}(n_i) + 2\,\mathrm{M}(n_i) + 2n_i\bigr) + \sum_{i=2}^{k}\bigl(3\,\mathrm{M}(n_i) + 3n_i\bigr) \le$$
$$\le n + \mathrm{GD}(n) + 3\,\mathrm{M}(n-p) + 3n - 3p + 2\,\mathrm{M}(n) + \mathrm{GD}(n) + 2\,\mathrm{M}(n) + 2n + 3\,\mathrm{M}(p) + 3p \le$$
$$\le 2\,\mathrm{GD}(n) + 7\,\mathrm{M}(n) + 6n.$$

Полученная оценка показывает, что сложность алгоритма Остроградского бесквадратной факторизации многочлена степени n асимптотически лишь в 2 раза превосходит сложность расширенного алгоритма нахождения НОД двух многочленов степени n.

5. Еще один вариант алгоритма Остроградского. Интересно отметить, что Остроградский предложил в заметке [2] еще один вариант алгоритма факторизации, основанный на тождестве $q_j = \bigl(Q, R - j\cdot(dQ/dx)\bigr)$, где

$$Q = Q_1 = \frac{P_1}{P_2} = \frac{P}{P_2}, \qquad R = R_1 = \frac{dP_1/dx}{P_2}, \qquad P_2 = \left(\frac{dP_1}{dx}, P_1\right).$$

Доказательство вытекает из цепочки равенств (в которой для краткости дифференцирование обозначается штрихом)

$$R = \frac{(P_1)'}{P_2} = \frac{(q_1 q_2^2 \cdots q_k^k)'}{q_2 \cdots q_{k-1}^k} = \sum_{i=1}^{k} i q_1 \cdots q_{i-1}(q_i)' q_{i+1} \cdots q_k,$$

$$Q' = \sum_{i=1}^{k} q_1 \cdots q_{i-1}(q_i)' q_{i+1} \cdots q_k, \qquad R - jQ' = \sum_{i=1}^{k}(i-j) q_1 \cdots q_{i-1}(q_i)' q_{i+1} \cdots q_k$$

(обосновываемой правилом дифференцирования Лейбница) и замечания о том, что в последней сумме все слагаемые, кроме s-го, делятся на q_s, а оно взаимно просто с многочленом q_s (так как все его сомножители с ним взаимно просты), следовательно, $(R - jQ', q_s) = 1$ при $s \ne j$ и, очевидно, многочлен $R - jQ$ делится на многочлен q_j.

Указанные формулы Остроградского более элегантны, чем предыдущие, но сложность последнего алгоритма, очевидно, больше, чем предыдущего, так как в нем степени последовательности многочленов, у которых приходится вычислять НОД, не уменьшаются и он не допускает очевидной хорошей верхней оценки. Однако если нужно вычислить не все множители q_i, а только любой один из них, то этот алгоритм, вероятно, является быстрейшим.

6. Оценка сложности алгоритма разложения рациональной функции на простейшие дроби. Основная идея быстрого алгоритма разложения на простейшие дроби состоит в применении метода Карацубы деления пополам.

Пусть P/Q — произвольная правильная дробь степени n, $Q = q_1 \cdots q_s$, $(q_i, q_j) = 1$, $i \ne j$. Положим $Q_i = Q/q_i$, тогда, как известно, справедливо равенство

$$\frac{P}{Q} = \sum_{i=1}^{s} \frac{p_i}{q_i},$$

где

$$p_i = (P \bmod q_i \cdot Q_i^{-1} \bmod q_i) \bmod q_i.$$

Для обоснования которого нужно проверить, что $(Q_1, \dots, Q_s) = 1$, воспользоваться линейным представлением НОД $1 = \sum_{i=1}^{s} U_i Q_i$ и заметить, что $U_i \bmod q_i = Q_i^{-1} \bmod q_i$ и

$$\frac{P}{Q} = \sum_{i=1}^{s} \frac{P \cdot U_i}{q_i} = \sum_{i=1}^{s} \frac{(P \cdot U_i) \bmod q_i}{q_i}.$$

Допустим, что $P \bmod q_i$ и $Q_i \bmod q_i$ уже вычислены. Тогда $Q_i^{-1} \bmod q_i$ находится со сложностью $\mathrm{GD}(\deg q_i)$ применением к паре $Q_i \bmod q_i, q_i$ расширенного алгоритма Евклида, который вычисляет линейное представление $1 = (Q_i, q_i) = UQ_i + Vq_i$, где $U = Q_i^{-1} \bmod q_i$, а $p_i = (P \bmod q_i) \times$ $\times (Q_i^{-1} \bmod q_i)$ находится со сложностью $7\,\mathrm{M}(\deg q_i) + 5 \deg q_i$ путем умножения и последующего деления результата на q_i (см. [21, задачи 17.32, 17.34]). Учитывая супераддитивность, имеем оценку $\mathrm{GD}(n) + 7\,\mathrm{M}(n) + 5n$.

Для вычисления $P \bmod q_i$ применяем метод деления пополам. Строим последовательность многочленов

$$Q_{\alpha_1,\dots,\alpha_k}, \qquad \alpha_i = 0, 1, \quad k \le l(s) = \lceil \log_2 s \rceil,$$

такую, что

$$Q = Q_0 \cdot Q_1, \quad \dots, \quad Q_{\alpha_1,\dots,\alpha_k} = Q_{\alpha_1,\dots,\alpha_k,0} \cdot Q_{\alpha_1,\dots,\alpha_k,1}, \quad \dots,$$

и количество сомножителей в соседних многочленах $Q_{\tilde{\alpha},0}$, $Q_{\tilde{\alpha},1}$ отличалось бы не более чем на 1. Вычисляем последовательность многочленов $P_{\tilde{\alpha}}$ рекурсивно:

$$P_{\tilde{\alpha},\alpha_{k+1}} = P_{\tilde{\alpha}} \bmod Q_{\tilde{\alpha},\alpha_{k+1}}.$$

Если положить $\deg Q_{\tilde{\alpha}} = n_{\tilde{\alpha}}$, то сложность вычисления обоих многочленов $P_{\tilde{\alpha},0}$, $P_{\tilde{\alpha},1}$ оценивается как

$$\begin{aligned} &3\big(\mathrm{M}(n_{\tilde{\alpha},1}) + n_{\tilde{\alpha},1}\big) + 2\,\mathrm{M}(n_{\tilde{\alpha}}) + n_{\tilde{\alpha},0} + 3\big(\mathrm{M}(n_{\tilde{\alpha},0}) + n_{\tilde{\alpha},0}\big) + 2\,\mathrm{M}(n_{\tilde{\alpha}}) + n_{\tilde{\alpha},1} \le \\ &\quad \le 3\,\mathrm{M}(n_{\tilde{\alpha}}) + 4n_{\tilde{\alpha}} + 4\,\mathrm{M}(n_{\tilde{\alpha}}) = 7\,\mathrm{M}(n_{\tilde{\alpha}}) + 4n_{\tilde{\alpha}}, \end{aligned}$$

а полная сложность — как

$$\begin{aligned} \sum_{k=1}^{l(s)} \sum_{\alpha_1=0,1;\dots;\alpha_k=0,1} 7\,\mathrm{M}(n_{\tilde{\alpha}}) + 4n_{\tilde{\alpha}} &\le l(s)\left(7\,\mathrm{M}\left(\sum_{\alpha_1=0,1;\dots;\alpha_k=0,1} n_{\tilde{\alpha}}\right) + 4 \sum_{\alpha_1=0,1;\dots;\alpha_k=0,1} n_{\tilde{\alpha}}\right) \le \\ &\le l(s)\big(7\,\mathrm{M}(n) + 4n\big). \end{aligned}$$

Сложность вычисления последовательности многочленов $Q_{\tilde{\alpha}}$ также оценивается с помощью метода Карацубы как $(l(s) - 1)\,\mathrm{M}(n)$.

Так как

$$Q_i \bmod q_i = \frac{Q_i \bmod q_i^2}{q_i},$$

то эта последовательность также оценивается с помощью дихотомии той же по порядку оценкой.

7. Уточнение мультипликативных констант в оценке сложности алгоритма разложения на простейшие дроби. Заметим, что возникающая в вычислениях последовательность остатков по $\bmod Q_{\tilde{\alpha}}^2$ представляется в виде $Q_{\tilde{\alpha}} \cdot R_{\tilde{\alpha}}$, где последовательность $R_{\tilde{\alpha}}$ вычисляется рекурсивно:

$$R_{\tilde{\alpha},\alpha_{k+1}} = Q_{\tilde{\alpha},1-\alpha_{k+1}} \cdot R_{\tilde{\alpha}} \bmod Q_{\tilde{\alpha},\alpha_{k+1}},$$

причем окончательные результаты $Q_i \bmod q_i$ имеют вид $R_{\alpha_1,\dots,\alpha_{l(s)}}$. Так как $\deg R_{\tilde{\alpha}} < n_{\tilde{\alpha}}$, то сложность вычисления обоих многочленов $R_{\tilde{\alpha},0}$, $R_{\tilde{\alpha},1}$ оценивается (с помощью задач 17.20, 17.25, 17.32 из [21]) как

$$\begin{aligned}
&\mathrm{M}(n_{\tilde{\alpha}}) + 3\big(\mathrm{M}(2n_{\tilde{\alpha},0}) + 2n_{\tilde{\alpha},0}\big) + \mathrm{M}(n_{\tilde{\alpha}} + n_{\tilde{\alpha},0}) + n_{\tilde{\alpha},1} + \mathrm{M}(2n_{\tilde{\alpha},0}, n_{\tilde{\alpha},1}) + \mathrm{M}(n_{\tilde{\alpha}}) + \\
&\quad + 3\big(\mathrm{M}(2n_{\tilde{\alpha},1}) + 2n_{\tilde{\alpha},1}\big) + \mathrm{M}(n_{\tilde{\alpha}} + n_{\tilde{\alpha},1}) + n_{\tilde{\alpha},0} + \mathrm{M}(2n_{\tilde{\alpha},1}, n_{\tilde{\alpha},0}) \le \\
&\qquad \le 4\,\mathrm{M}(2n_{\tilde{\alpha}}) + \mathrm{M}(3n_{\tilde{\alpha}}) + 3\,\mathrm{M}(n_{\tilde{\alpha}}) + 7n_{\tilde{\alpha}},
\end{aligned}$$

а полная сложность как $l(s)(4\,\mathrm{M}(2n) + \mathrm{M}(3n) + 3\,\mathrm{M}(n) + 7n)$. Если вычислить последовательность $R_{\tilde{\alpha}}$ раньше, чем последовательность $P_{\tilde{\alpha}}$, то при вычислении последней не надо будет "обращать" многочлены $Q_{\tilde{\alpha}}$, так как это уже сделано при вычислении предыдущей последовательности (это ясно из решения задачи 17.26 в [21]), что позволяет уменьшить оценку сложности на $3\,\mathrm{M}(n) + 3n$, и суммарная оценка будет иметь вид $l(s)(4\,\mathrm{M}(2n) + \mathrm{M}(3n) + 8\,\mathrm{M}(n) + 8n) - \mathrm{M}(n)$. Можно еще уменьшить оценку, если перед умножением

$$R_{\tilde{\alpha},\alpha_{k+1}} = Q_{\tilde{\alpha},1-\alpha_{k+1}} \cdot R_{\tilde{\alpha}} \bmod Q_{\tilde{\alpha},\alpha_{k+1}}$$

привести $R_{\tilde{\alpha}}$ по $\bmod Q_{\tilde{\alpha},\alpha_{k+1}}$, тогда "обращать" многочлен $Q_{\tilde{\alpha}}$ придется тоже только один раз, на умножение будет уходить $2\,\mathrm{M}(n)$, а на приведение по модулю $3\,\mathrm{M}(n) + 3n + 4\,\mathrm{M}(n) + 2n$ операций. Суммарная оценка тогда будет иметь вид $l(s)(14\,\mathrm{M}(n) + 6n) - \mathrm{M}(n)$. Можно еще заметить, что при выполнении условия $n_{\tilde{\alpha},0} = n_{\tilde{\alpha},1}$ равенства степеней соседних многочленов эта оценка еще уменьшается на $2l(s)\,\mathrm{M}(n)$.

8. Оценка сложности второго подалгоритма алгоритма разложения рациональной функции на простейшие дроби. Здесь речь идет о разложении правильной дроби вида P/Q^s в сумму еще более простых дробей

$$\sum_{i=1}^{s} \frac{P_i}{Q^i}.$$

Эта задача, очевидно,равносильна разложению многочлена P по степеням Q. Решается она подобно проведенному выше вычислению остатков многочлена по заданным модулям, только все q_i здесь равны Q и на каждом шаге по многочлену $P_{\tilde{\alpha}}$ строятся многочлены $P_{\tilde{\alpha},0}$, $P_{\tilde{\alpha},1}$ такие, что

$$P_{\tilde{\alpha}} = P_{\tilde{\alpha},0} Q_{\tilde{\alpha},1} + P_{\tilde{\alpha},1}.$$

Если для каждого деления выбирать из двух соседних многочленов $Q_{\tilde{\alpha}}$ тот, который имеет большую степень, то сложность всех делений оценивается (с помощью задач 17.20, 17.25, 17.31, 17.32 из [21]) как

$$\begin{aligned}
&\sum_{k=1}^{l(s)} \sum_{\alpha_1=0,1;\dots;\alpha_k=0,1} \Big(3\,\mathrm{M}(n_{\tilde{\alpha},0}) + 3n_{\tilde{\alpha},0} + \mathrm{M}(n_{\tilde{\alpha},1}) + n_{\tilde{\alpha},1} + 2\,\mathrm{M}\Big(\frac{n_{\tilde{\alpha}}}{2}\Big) + \frac{n_{\tilde{\alpha}}}{2}\Big) \le \\
&\quad \le l(s)\Big(4\,\mathrm{M}\Big(\frac{n}{2}\Big) + \mathrm{M}(n) + \frac{5n}{2}\Big),
\end{aligned}$$

а сложность вычисления всех многочленов $Q_{\tilde{\alpha}}$, которые получаются путем последовательного возведения в квадрат или перемножения соседних многочленов, оценивается как $2\,\mathrm{M}(n)$, где $n = s\deg Q$, т.е. сложность всего подалгоритма не превосходит $l(s)(3\,\mathrm{M}(n) + 5n/2) + 2\,\mathrm{M}(n)$.

Как видим, и здесь фактически был применен метод Карацубы.

Окончательная оценка сложности алгоритма разложения на простейшие дроби имеет вид

$$\lceil \log_2 k \rceil \Big(17\,\mathrm{M}(n) + \frac{17n}{2}\Big) + 8\,\mathrm{M}(n) + 5n + \mathrm{GD}(n),$$

где k — длина разложения знаменателя на бесквадратные множители: $Q = q_1 q_2^2 \cdots q_k^k$, а $n = \deg Q$.

9. Оценка сложности собственно алгоритма интегрирования рациональных дробей. Оценим вначале сложность интегрирования дроби

$$\frac{P}{Q^s} = \sum_{i=1}^{s} \frac{P_i}{Q^i},$$

где $\deg P_i < \deg Q$, $s\deg Q = n$, $s > 1$. Для этого воспользуемся тождеством

$$\int \frac{P_i}{Q^i}\,dx = \frac{-P_i B/(i-1)}{Q^{i-1}} + \int \frac{P_i A + d(P_i B/(i-1))/dx}{Q^{i-1}}\,dx,$$

где $AQ + B\cdot(dQ/dx) = 1$, $\deg A, \deg B < \deg Q$, получаемым интегрированием по частям. Сложность вычисления многочленов A и B оцениваем как $\mathrm{GD}(\deg Q)$, потом находим числитель дроби, стоящей под вторым интегралом, со сложностью $2\,\mathrm{M}(\deg Q) + 6\deg Q$ и сразу делим его на Q с остатком, представляя в виде $CQ + D$, где $\deg C$, $\deg D < \deg Q$ со сложностью $6\,\mathrm{M}(\deg Q) + 5\deg Q$; далее складываем C с P_{i-2}, а D с P_{i-1} со сложностью $2\deg Q$, и задача сводится к такой же, но i уменьшается на 1. Значит, интегрирование дроби

$$\frac{P}{Q^s} = \sum_{i=1}^{s} \frac{P_i}{Q^i}$$

выполняется со сложностью

$$(s-1)\big(5\,\mathrm{M}(\deg Q) + 10\deg Q\big) + 3\,\mathrm{M}(\deg Q) + \deg Q + \mathrm{GD}(\deg Q),$$

так как вычислять A, B и "обращать" многочлен Q надо только один раз, а при $i = 2$ не надо делить на $i - 1$.

Учтем еще сложность преобразования рациональной части интеграла к виду

$$S_0 + \sum_{i=1}^{s-1} \frac{S_i}{Q^i},$$

где $\deg S_i < \deg Q$. Для этого заметим, что $-P_i B/(i-1)$ вычисляется со сложностью $\mathrm{M}(\deg Q) + 2\deg Q$ (при $i = 2$ последнее слагаемое исчезает), деление полученного результата на Q и тем самым представление его в виде $C_i Q + D_i$, где $\deg C_i$, $\deg D_i < \deg Q$, выполняется со сложностью $3\,\mathrm{M}(\deg Q) + 2\deg Q$ (так как "обращение" многочлена Q уже выполнено ранее), а сложение (при $i > 2$) многочленов C_i и D_{i-1} — со сложностью $\deg Q$, что в итоге дает оценку

$$(s-1)\big(4\,\mathrm{M}(\deg Q) + 5\deg Q\big) - 3\deg Q.$$

Складывая ее с полученной ранее оценкой, окончательно имеем оценку

$$(s-1)\bigl(9\,\mathrm{M}(\deg Q)+15\deg Q\bigr)+3\,\mathrm{M}(\deg Q)-2\deg Q+\mathrm{GD}(\deg Q).$$

Используя супераддитивность функции $\mathrm{M}(n)$, преобразуем эту оценку к виду

$$6\,\mathrm{M}(n-\deg Q)+15n+3\,\mathrm{M}(n)-17\deg Q+\mathrm{GD}(\deg Q).$$

Применяя указанную оценку к каждой из дробей, получающихся при разложении исходной дроби N/P, где $P=q_1q_2^2\cdots q_k^k$, а $n=\deg P$, и пользуясь супераддитивностью функций $\mathrm{M}(n)$ и $\mathrm{GD}(n)$, получаем оценку сложности собственно интегрирования в виде

$$3\,\mathrm{M}(n-\deg q_1)+6\,\mathrm{M}(p)+15p-2\deg Q+2\deg q_1+\mathrm{GD}(\deg Q)<3\,\mathrm{M}(n)+6\,\mathrm{M}(p)+\mathrm{GD}(n-p)+15p,$$

но на сей раз под Q понимаем многочлен $q_1q_2\cdots q_k$, а $p=\deg P_1$, где $P_1=P/Q$.

10. Оценка сложности сложения полученных рациональных дробей. Сам Остроградский представлял результат интегрирования в виде

$$\int\frac{N}{P}\,dx=\frac{X}{P}+\int\frac{Y}{Q}\,dx,$$

и, чтобы найти многочлены X и Y, надо сложить полученные дроби (и многочлены в рациональных частях интегралов). Применяя для этого метод деления пополам и замечая, что сложение двух дробей степени d требует не более $3\,\mathrm{M}(d)+2d$ операций, сумму правильных дробей

$$\sum_{i=1}^{s}\frac{P_i}{Q^i}$$

можно (решая обратную к рассмотренной в конце п. 8 задачу) вычислить за

$$l(s)\Bigl(\mathrm{M}(s\deg Q)+\frac{1}{2}s\deg Q\Bigr)$$

операций, многочлены степеней меньше $\deg q_i$ складываются со сложностью $\deg Q$, а потом их сумма прибавляется к ранее полученной правильной дроби со сложностью $2\deg Q+\mathrm{M}(\deg Q)$, получаем для сложности сложения дробей такую оценку:

$$\begin{aligned}&3\deg Q+\mathrm{M}(\deg Q)+\lceil\log_2 k\rceil\Bigl(3\,\mathrm{M}(p)+2p+\mathrm{M}(p)+\frac{p}{2}+3\,\mathrm{M}(\deg Q)+2\deg Q\Bigr)\le\\&\quad\le\lceil\log_k t\rceil\Bigl(3\,\mathrm{M}(n)+\mathrm{M}(p)+\frac{5p}{2}\Bigr)+\mathrm{M}(n-p)+3(n-p).\end{aligned}$$

Итоговая оценка сложности алгоритма интегрирования рациональной дроби имеет вид

$$\begin{aligned}&\lceil\log_2 k\rceil\Bigl(3\,\mathrm{M}(n)+\mathrm{M}(p)+\frac{5p}{2}\Bigr)+\mathrm{M}(n-p)+3(n-p)+3\,\mathrm{M}(n)+6\,\mathrm{M}(p)+\mathrm{GD}(n-p)+15p+\\&\quad+\lceil\log_2 k\rceil\Bigl(17\,\mathrm{M}(n)+\frac{17n}{2}\Bigr)+8\,\mathrm{M}(n)+5n+\mathrm{GD}(n)+2\,\mathrm{GD}(n)+7\,\mathrm{M}(n)+6n\le\\&\qquad\le\lceil\log_k\Bigl(20\,\mathrm{M}(n)+\mathrm{M}(p)+\frac{17n}{2}+\frac{5p}{2}\Bigr)+\\&\qquad\quad+3\,\mathrm{GD}(n)+\mathrm{GD}(n-p)+19\,\mathrm{M}(n)+5\,\mathrm{M}(p)+14n+12p\le\\&\qquad\le\lceil\log_2 n\rceil(21\,\mathrm{M}(n)+11n)+24\,\mathrm{M}(n)+26n+4\,\mathrm{GD}(n).\end{aligned}$$

11. Оценка сложности нахождения логарифмической части интеграла. Логарифмическую часть интеграла

$$\int \frac{N}{P}\,dx,$$

а именно

$$\int \frac{Y}{Q}\,dx,$$

нельзя найти чисто арифметическими средствами (обсуждение этого вопроса имеется, например, в [23]), но если известны корни многочлена Q, то для этого, очевидно, достаточно разложить дробь Y/Q на простейшие дроби. Действительно, если $Q = q_1 \cdots q_m$, где $q_i(x) = x - c_i$, $c_i \in \mathbb{C}$, $(q_i, q_j) = 1$, $i \neq j$, то положим $Q_i = Q/q_i$ и, как известно, получим разложение

$$\frac{Y}{Q} = \sum_{i=1}^{s} \frac{p_i}{q_i},$$

где

$$p_i = Y \cdot Q_i^{-1} \bmod q_i = Y \bmod q_i Q_i^{-1} \bmod q_i = \frac{Y(c_i)}{Q'(c_i)} = b_i \in \mathbb{C}.$$

Так как числа $Y(c_i)$ и $Q'(c_i)$ суть остатки от деления многочленов $Y(x)$ и $Q'(x) = dQ/dx$ на двучлены $q_i(x) = x - c_i$, то, как уже отмечалось выше в более общей ситуации, они могут быть вычислены со сложностью $l(m)(15\,\mathrm{M}(m) + 8m) + 2m - \mathrm{M}(m)$ (здесь учтено, что необходимые при вычислении как числителей, так и знаменателей дробей b_i произведения двучленов q_i достаточно вычислить только один раз). После этого выполняется попарное сложение комплексно-сопряженных дробей со сложностью не более $5m/2$ и все дроби интегрируются элементарным образом со сложностью не более $4m$.

Заметим, что если дано разложение знаменателя P интегрируемой дроби на множители над полем $\mathbb{R}$, то нет необходимости вычислять сначала многочлен Q методом Остроградского, так как задача сразу сводится к разложению дроби N/P на сумму дробей вида Y/Q^s, где Q — либо линейный двучлен, либо квадратный трехчлен. В первом случае разложение такой дроби на сумму простейших сводится к нахождению разложения многочлена P в ряд Тейлора относительно точки a, где $Q(x) = x - a$, а, как показано в [22], это может быть выполнено быстрее, чем указывалось выше, а именно со сложностью $O(s \log s)$. Второй случай, как отмечалось в [17], сводится к первому, если выделить в Q полный квадрат, сделать соответствующую линейную замену переменных в многочлене Y (а это то же самое, что и упомянутое только что разложение в ряд Тейлора) и после этого представить его в виде $Y_0(x^2) + xY_1(x^2)$, и после замены $z = x^2$ задача опять сводится к первому случаю, поэтому во втором случае сложность нахождения разложения рассматриваемой дроби на сумму простейших остается по порядку такой же.

12. Оценка сложности оригинального алгоритма Остроградского интегрирования рациональных дробей. Изложенный выше алгоритм бесквадратной факторизации многочленов является лишь началом алгоритма интегрирования рациональных дробей Остроградского [1]. Этот алгоритм слишком громоздок для его изложения здесь (работа [1] занимает около 50 страниц). Применяя указанные выше приемы, можно оценить его сложность как

$$2\,\mathrm{GD}(n) + \mathrm{GD}(p) + O\left(\frac{nk\,\mathrm{M}(n_2)}{n_2}\right) + O\big(\log k \cdot \mathrm{M}(n)\big),$$

где $p = n - n_1$ и $n_i = \deg Q_i$, а последовательность многочленов Q_i та же самая, что и в п. 3. Вообще говоря, эта оценка хуже, чем оценка сложности алгоритма Эрмита, но при $p/n \to 0$ и $(k \log n_2) = o(\log^2 n)$ она все же лучше асимптотически в 2 раза.

13. Оценка сложности алгоритма китайской теоремы об остатках и задачи интерполирования многочленов. Китайская теорема об остатках и связанные с ней алгоритмы имеют многочисленные применения в разных разделах математики и во многих конкретных работах (например, алгоритм быстрого умножения [8] использует модулярную арифметику, которая собственно и основана на китайской теореме об остатках). Интересно отметить, что быстрый вариант китайского алгоритма тесно связан с быстрым алгоритмом разложения на простейшие дроби и оба они фактически основаны на идее, впервые примененной Карацубой в [4].

Пусть даны многочлены $q_1, \ldots, q_k$, $(q_i, q_j) = 1$, $i \neq j$, и многочлены $p_1, \ldots, p_k$, $\deg p_i < n_i = \deg q_i$, $n = n_1 + \cdots + n_k$. Китайская теорема об остатках утверждает существование и единственность по $\bmod Q$ многочлена P такого, что $P \bmod q_i = p_i$, $1 \leq i \leq k$. Положим

$$Q = q_1 \cdots q_k, \qquad Q_i = \frac{Q}{q_i}, \qquad R_i = Q_i^{-1} \bmod q_i;$$

тогда, как известно, справедливо равенство

$$P = \sum_{i=1}^{s} (p_i Q_i R_i) \bmod Q = Q \sum_{i=1}^{s} \frac{(p_i R_i) \bmod q_i}{q_i}.$$

Поэтому для вычисления многочлена P достаточно найти произведения $r_i = p_i R_i \bmod q_i$, что делается так же, как в п. 6, 7, со сложностью $\mathrm{GD}(n) + 7\,\mathrm{M}(n) + 5n + l(k)(10\,\mathrm{M}(n) + 5n) - \mathrm{M}(n)$, а потом вычислить сумму правильных дробей степеней n_i

$$\frac{P}{Q} = \sum_{i=1}^{k} \frac{r_i}{q_i}$$

так же, как в п. 10, со сложностью $l(k)(3\,\mathrm{M}(n) + n)$. Итоговая оценка имеет вид

$$\mathrm{GD}(n) + l(k)(13\,\mathrm{M}(n) + 6n) + 6\,\mathrm{M}(n) + 5n.$$

В случае, когда $q_i(x) = x - c_i$, а c_i и p_i принадлежат полю коэффициентов, как известно, многочлен P, построенный с помощью обратного китайского алгоритма, является решением интерполяционной задачи $P(c_i) = p_i$. В этом случае вычисление $R_i = Q_i^{-1} \bmod q_i$ по известным $Q_i \bmod q_i$ вместо применения расширенного алгоритма нахождения НОД многочленов требует просто n операций деления. Из сказанного выше вытекает, что сложность ее решения оценивается как

$$l(n)(13\,\mathrm{M}(n) + 6n) + 6\,\mathrm{M}(n) + 6n.$$

Если n является степенью двойки, то с учетом сделанного в п. 6 замечания эту оценку можно немного уменьшить.

Прямой китайский алгоритм вычисляет по многочлену P его остатки по заданным модулям q_i и, в частности, просто его значения в заданных точках c_i. Оценка его сложности фактически и была получена в п. 6.

Отметим, что все указанные выше результаты переносятся и на числовой вариант китайской теоремы об остатках, и на числовой вариант задачи о разложении дроби на сумму простейших. Отметим еще, что в [13] вопрос о быстром китайском алгоритме (в варианте без предварительной обработки информации) изложен несколько смутно, а задача о разложении дроби на простейшие вообще не рассматривается (статьи [17, 18] появились позже).

14. Оценка сложности умножения многочленов и алгоритм китайской теоремы об остатках. Для вычисления многочлена r, являющегося произведением многочленов p и q степени, меньшей $n/2$, естественно выполнить их умножение по модулю x^n-1, для чего вначале надо вычислить остатки от деления этих многочленов на линейные попарно взаимно простые двучлены $x-\varepsilon^k$, получающиеся при разложении многочлена x^n-1 на множители (например, над полем $\mathbb{C}$), т.е. попросту вычислить значения $p(\varepsilon^k)$ и $q(\varepsilon^k)$, $k=0,\dots,n-1$, потом, попарно перемножив их, получить значения $r(\varepsilon^k)=p(\varepsilon^k)q(\varepsilon^k)$ и, применяя "обратный китайский алгоритм", восстановить многочлен r. Удобно при этом предполагать n равным степени двойки (добавляя, если надо, нулевые младшие коэффициенты к многочленам p и q).

Применяя для вычисления значений $p(\varepsilon^k)$ и $q(\varepsilon^k)$, $k=0,\dots,n-1$, прием, изложенный в п. 6, находим вначале

$$p \bmod x^{n/2}-1, \quad p \bmod x^{n/2}+1, \qquad q \bmod x^{n/2}-1, \quad q \bmod x^{n/2}+1,$$

потом вычисляем остатки по модулям $x^{n/4}+1$, $x^{n/4}-1$, $x^{n/4}+i$, $x^{n/4}-i$ и т.д., пока не найдем остатки по модулям $x^2-\varepsilon^{2k}$, $k=0,\dots,n/2$, и, наконец, по модулям $x-\varepsilon^k$, $k=0,\dots,n$. Так как деление многочлена степени, меньшей m, на двучлен степени $m/2$ осуществляется "школьным" алгоритмом со сложностью m, то сложность всего алгоритма вычисления значений $p(\varepsilon^k)$ и $q(\varepsilon^k)$, $k=0,\dots,n-1$, оценивается как $2n\log_2 n$ (умножений и сложений, выполненных в поле $\mathbb{C}$).

Для восстановления (интерполяции) многочлена r по его известным значениям $r(\varepsilon^k)$, $k=0,\dots,n-1$, естественно применить формулу Лагранжа

$$r=f(x)\sum_{k=0}^{n-1}\frac{r(\varepsilon^k)/f'(\varepsilon^k)}{x-\varepsilon^k}=f(x)\sum_{k=0}^{n-1}\frac{r(\varepsilon^k)\varepsilon^k}{n(x-\varepsilon^k)},$$

где $f(x)=x^n-1$, и суммирование дробей выполнить так же, как в п. 10. Так как при надлежащем выборе порядка суммирования все получающиеся дроби будут иметь двучленные знаменатели, а умножение многочлена степени, меньшей m, на двучлен степени m "школьным" методом имеет сложность m, то сложение двух таких дробей имеет сложность $4m$, а весь алгоритм интерполяции — сложность $2n+2n\log_2 n$. В итоге получаем известное равенство $\mathrm{M}(n)=O(n\log n)$.

Как мы видим, быстрое умножение многочленов осуществляется фактически с помощью применения китайской теоремы об остатках и метода "деления пополам", впервые примененного к этой задаче А. А. Карацубой [4]. О возможности применения китайской теоремы об остатках для умножения чисел указывал А. Н. Колмогоров (как рассказано в [5], ему принадлежит инициатива исследований в этой области).

В приведенном доказательстве нет упоминания о дискретном преобразовании Фурье. На самом деле, проведенное выше вычисление значений многочлена в корнях n-й степени из единицы совпадает с дискретным преобразованием Фурье, а указанный для этого прием совпадает с одним из множества известных сейчас алгоритмов быстрого преобразования Фурье (этим алгоритмам посвящена монография [24]). Такой же алгоритм для быстрого преобразования Фурье описан в [13]. Интерполяция многочлена по значениям в корнях n-й степени из единицы совпадает с обратным преобразованием Фурье, и поэтому для ее проведения указанный выше алгоритм можно было бы не применять, так как оценка сложности обратного преобразования Фурье, как известно, фактически совпадает с оценкой сложности прямого преобразования Фурье. Применение китайской теоремы в указанной ситуации эквивалентно применению так называемой теоремы о циклической свертке.

СПИСОК ЛИТЕРАТУРЫ

1. *Остроградский М. В.* Об интегрировании рациональных дробей // Избранные труды. М.: Изд-во АН СССР, 1958. С. 56–105.
2. *Остроградский М. В.* Заметка о равных множителях целых полиномов // Избранные труды. М.: Изд-во АН СССР, 1958. С. 106–107.
3. *Hermite C.* Quevres de Charles Hermite. Paris: Gauthier-Villars, Imprimeur-Librae, 1912. V. 3.
4. *Карацуба А. А., Офман Ю. П.* Умножение многозначных чисел на автоматах // ДАН СССР. 1962. Т. 145, № 2. С. 293–294.
5. *Карацуба А. А.* Сложность вычислений // Тр. МИАН. 1995. Т. 211. С. 186–202.
6. *Тоом А. Л.* О сложности схемы из функциональных элементов, реализующей умножение целых чисел // ДАН СССР. 1963. Т. 150, № 2. С. 496–498.
7. *Шёнхаге А., Штрассен В.* Быстрое умножение больших чисел // Кибернетический сборник. М.: Мир, 1973. Вып. 10. С. 87–98.
8. *Schönhage A.* Schnelle Multiplikation grossen Zahlen // Computing. 1966. V. 1. P. 182–196.
9. *Cook S. A.* On the minimum computation time of functions: Thes. Harvard Univ., 1966. P. 51–77.
10. *Бендерский Ю. В.* Быстрые вычисления // ДАН СССР. 1975. Т. 223, № 5. С. 1041–1043.
11. *Kung H. T.* Fast evaluation and interpolation. Pittsburgh: Dept. Comput. Sci. Carnegie-Mellon Univ., 1973.
12. *Штрассен В.* Алгоритм Гаусса не оптимален // Кибернетический сборник. М.: Мир, 1970. Вып. 7. С. 67–70.
13. *Ахо А., Хопкрофт Дж., Ульман Дж.* Построение и анализ вычислительных алгоритмов. М.: Мир, 1979.
14. *Кнут Д.* Искусство программирования на ЭВМ. М.: Мир, 1976. Т. 2.
15. *Schönhage A., Grotefeld A. F. W., Vetter E.* Fast algorithms. Manheim; Leipzig; Wien; Zürich: Wissenschaftsverlag, 1994. Bd. 1.
16. *Yun D. Y. Y.* Fast algorithm for rational function integration // Proc. IFIP Congr. Amsterdam: North-Holland, 1977. V. 7. P. 493–498.
17. *Kung H. T., Tong D. M.* Fast algorithms for fraction decompositon // SIAM J. Comput. 1977. V. 6. P. 582–593.
18. *Chin P. I.* Complexity of partial fraction expansion // SIAM J. Comput. 1977. V. 6. P. 554–562.
19. *Moenk R.* Fast algorithm of GCD's // Proc. 5th Ann. ACM Sympos. on theory of computing. 1973. P. 142–151.
20. *Schönhage A.* Schnelle Berechnung von Kettenbruchentwicklungen // Acta inform. 1971. V. 1. P. 139–144.
21. *Гашков С. Б., Чубариков В. Н.* Арифметика. Алгоритмы. Сложность вычислений. М.: Наука, 1996.
22. *Aho A. V., Steiglitz K., Ullman J. D.* Evaluation of polynomials at fixed set of points // SIAM J. Comput. 1975. V. 6. P. 533–539.
23. *Дэвенпорт Дж.* Интегрирование алгебраических функций. М.: Наука, 1985.
24. *Блейхут Р.* Быстрые алгоритмы цифровой обработки сигналов. М.: Мир, 1989.