

НОВАЯ ОПТИМИЗАЦИЯ АЛГОРИТМА БУХБЕРГЕРА

С. В. Агиевич

*НИИ прикладных проблем математики и информатики
Минск, Беларусь*

Пусть K – поле, $P = K[x_1, \dots, x_n]$ – кольцо многочленов от переменных $x_1, \dots, x_n$ над полем K , T – множество мономов на этих переменных, $<$ – допустимый мономиальный порядок на T (подробнее см. [1]). Для ненулевого $f \in P$ пишем: $LM(f)$ – старший (относительно порядка $<$) моном f , $LC(f)$ – коэффициент при старшем мономе, $LT(f) = LC(f) \cdot LM(f)$ – старший член.

Пусть $G = \{g_1, \dots, g_r\}$ – система многочленов P . Если f содержит член ct , $c \in K$, $t \in T$, который делится на некоторый $LM(g_i)$, то мы можем исключить данный член, заменив f на $f - cgt / LT(g_i)$. Последовательно исключая самые старшие подходящие члены f , мы в конце концов получим представление $f = h_1g_1 + \dots + h_rg_r + g$, где $h_i \in P$, $LM(h_ig_i) \leq LM(f)$ и в g нет членов, которые делятся на $LM(g_i)$. Многочлен g является остатком от деления f на G , что записывается как $f \xrightarrow{G} g$.

Остаток g определен, вообще говоря, неоднозначно, поскольку на шагах деления $f \leftarrow f - cgt / LT(g_i)$ могут встретиться сразу несколько делителей g_i , таких, что $LM(g_i) \mid t$. В таких случаях будем отдавать предпочтение многочлену g_i с минимальным индексом i , а соответствующий данной цепочке делений остаток обозначать через $\bar{f}^G$. Для пустой системы G полагаем $\bar{f}^G = f$.

Пусть I – идеал кольца P . Система G называется *базисом Гребнера* I , если для любого ненулевого $f \in I$ найдется $g_i \in G$, такой, что $LM(g_i) \mid LM(f)$. Эквивалентно, G – базис Гребнера, если остаток от деления f на G всегда нулевой, независимо от порядка делителей.

Базисы Гребнера обладают рядом замечательных свойств, что делает их нахождение важным при решении многих задач теории кодирования, теории инвариантов, криптографии и других математических дисциплин. В частности, решение системы полиномиальных уравнений сводится к построению базиса Гребнера идеала, порожденного многочленами системы.

Для нахождения базисов Гребнера используется алгоритм Бухбергера, который основан на следующем критерии: G – базис Гребнера тогда и только тогда, когда $S(g_i, g_j) \xrightarrow{G} 0$ для любых $(g_i, g_j) \in G$. Здесь $S(g_i, g_j)$ – так называемый S -многочлен пары (g_i, g_j) :

$$S(g_i, g_j) = \frac{[LM(g_i), LM(g_j)]}{LT(g_i)} g_j - \frac{[LM(g_i), LM(g_j)]}{LT(g_j)} g_i,$$

$[LM(g_i), LM(g_j)]$ – н. о. к. мономов $LM(g_i)$ и $LM(g_j)$.

Алгоритм Бухбергера

Вход: $F \subset P$, порядок $<$.

Выход: $G \subset P$ – базис Гребнера идеала $\langle F \rangle$ относительно порядка $<$.

Вспомогательные алгоритмы: Update.

Шаги:

1. $G \leftarrow \emptyset, B \leftarrow \emptyset, B^* \leftarrow \emptyset$.
2. Для всех $f \in F$:
 - (a) $g \leftarrow \bar{f}^G$;
 - (b) если $g \neq 0$, то $(G, B) \leftarrow \text{Update}(G, B, B^*, g)$.

3. Пока $B \neq \emptyset$:
 - a) выбрать $(g_i, g_j) \in B$;
 - b) $B \leftarrow B \setminus \{(g_i, g_j)\}$, $B^* \leftarrow B \cup \{(g_i, g_j)\}$;
 - c) $g \leftarrow \overline{S(g_i, g_j)}^G$;
 - d) если $g \neq 0$, то $(G, B) \leftarrow \text{Update}(G, B, B^*, g)$.
4. Возвратить G .

Алгоритм Update (обновление в алгоритме Бухбергера)

Вход: $G \subset P$, $B \subset P \times P$, $B^* \subset P \times P$, $g \in P \setminus \{0\}$.

Выход: обновленные G и B .

Шаги:

1. $G \leftarrow G \cup \{g\}$.
2. $B \leftarrow B \cup \{(f, g) : f \in G\}$.
3. Возвратить (G, B) .

Элементы множества B в алгоритме Бухбергера принято называть *критическими парами*. Критические пары определяют S -многочлены $S(g_i, g_j)$, по которым, в свою очередь, вычисляются остатки g . Обработанные критические пары переносятся из B в B^* . Множество B^* , пока избыточное, будет использоваться в предлагаемой далее оптимизации алгоритма.

Нахождение остатков является самым трудоемким шагом алгоритма Бухбергера. С другой стороны, вычислительные эксперименты показывают, что большинство остатков оказываются нулевыми, т. е. бесполезными. Для того чтобы избежать бесполезных вычислений, оптимизированные редакции алгоритма Бухбергера снабжаются дополнительными механизмами исключения некоторых критических пар. Данные механизмы основаны на том, что только по старшим мономам многочленов g_i и g_j можно сделать вывод, что $S(g_i, g_j) \xrightarrow{H} 0$, где H – множество многочленов, которые содержатся в G либо будут включены в G на следующих шагах алгоритма Бухбергера. Указанный вывод позволяет исключить пару (g_i, g_j) из B , даже не выполняя деление.

Известны следующие механизмы исключения критических пар: исключение по критериям Бухбергера [1], исключение r -пар [2], исключение по критериям Гебауэра – Мюллера [4], построение минимальной системы генераторов модуля сизигий старших мономов G [3]. В качестве примера, первый критерий Бухбергера позволяет исключить пару (g_i, g_j) , у которой нет зацепления (старших мономов), т. е. старшие мономы взаимно просты: $[\text{LM}(g_i), \text{LM}(g_j)] = \text{LM}(g_i) \text{LM}(g_j)$. Для такой пары $S(g_i, g_j) \xrightarrow{\{g_i, g_j\}} 0$.

Нами предлагается новая оптимизация алгоритма Бухбергера, в которой явно или косвенно использованы все перечисленные механизмы исключения критических пар. Оптимизация состоит в замене алгоритма Update на алгоритм UpdateEx:

Алгоритм UpdateEx (оптимизированное обновление)

Вход: $G \subset P$, $B \subset P \times P$, $B^* \subset P \times P$, $g \in P \setminus \{0\}$.

Выход: обновленные G и B .

Шаги:

1. Исключить из B пары (f, h) , такие, что $\text{LM}(h)$ не делит $\text{LM}(f)$ и $\text{LM}(g) \mid [\text{LM}(f), \text{LM}(h)]$.
2. Для всех $f \in G$: если $\text{LM}(g) \mid \text{LM}(f)$, то $G \leftarrow G \setminus \{f\}$, $B \leftarrow B \cup \{(f, g)\}$.
3. $R \leftarrow \{(f, g) : f \in G\}$.
4. Для всех пар $(f, g) \in R$ без зацепления: если найдется другая пара $(h, g) \in R$, для которой
 - а) $[\text{LM}(h), \text{LM}(g)] \mid [\text{LM}(f), \text{LM}(g)]$,
 - б) $\text{LM}(g)$ не делит $[\text{LM}(f), \text{LM}(h)]$ или $\text{LM}(g)$ делит $[\text{LM}(f), \text{LM}(h)]$ и одна из пар (f, h) , (h, f) содержится в B^* ,то исключить (h, g) из B .
5. Исключить из B пары без зацепления.
6. $G \leftarrow G \cup \{g\}$.
7. $B \leftarrow B \cup R$.
8. Возвратить (G, B) .

В докладе объясняются принципы, заложенные в основу алгоритма UpdateEx, и обосновывается корректность замены Update на UpdateEx.

ЛИТЕРАТУРА

1. Кокс, Д. Идеалы, многообразия и алгоритмы/ Д. Кокс, Дж. Литтл, Д. О. Ши. М.: Мир, 2000.
2. Boulier, F. A new criterion to avoid useless critical pairs in Buchberger's algorithm // Technical report LIFL 2001-07, avail. at: <http://www.lifl.fr/boulier/PUBLICATIONS/LIFL2001-07.ps.gz>.
3. Caboara, M. Efficiently computing minimal sets of critical pairs/ M. Caboara, M. Kreuzer, L. Robbiano // Journal of Symbolic Computation. 2004. Vol. 38. 4. P. 1169–1190.
4. Gebauer, R. On an Installation of Buchberger's Algorithm / R. Gebauer, H. M. Möller // Journal of Symbolic Computation. 1987. 6. P. 257–286.