1 Задание

Дана следующая SRS:

$$\begin{cases} aabc \to bbaa \\ b \to ccaa \\ bc \to a \\ aac \to \varepsilon \end{cases}$$

По данной SRS определить:

- завершимость;
- конечность классов эквивалентности по НФ (для построения эквивалентностей считаем, что правила могут применяться в обе стороны). Если их конечное число, то построить минимальную систему переписывания, им соответствующую;
- локальную конфлюэнтность и пополняемость по Кнуту-Бендиксу.

Провести автоматическое тестирование предполагаемой эквивалентности двух указанных SRS:

Фазз-тестирование эквивалентности: строится случайное слово ω и случайная цепочка переписываний его в ω' по $\mathcal{T}$. Проверить, можно ли получить ω' из ω (или наоборот) в рамках правил $\mathcal{T}'$.

Метаморфное тестирование: выбрать инварианты, которые должны сохраняться (либо монотонно изменяться) при переписывании в рамках $\mathcal{T}$. Породить случайную цепочку переписываний над случайным словом в $\mathcal{T}'$ и проверить выполнимость инвариантов. Как минимум два разных инварианта.

2 Завершимость

Правило 2 — единственное, которое увеличивает длину строки (на 3 символа) и количество c (на 2). Оно потребляет один b. Правило 1 сохраняет длину, увеличивает количество b на 1, уменьшает количество c на 1. Правила 3 и 4 уменьшают длину и потребляют c без производства b или c.

Поскольку правила, уменьшающие длину, не могут применяться бесконечно (длина ограничена снизу нулем), то, чтобы иметь бесконечную последовательность должно быть бесконечно много применений правила 2 (единственного правила, увеличивающего длину) и правила 1 (для пополнения b, потребляемых правилом 2).

Анализ возможных выводов показывает, что увеличения количества b происходят только через правило 1, которое требует подстроки aabc. Эта подстрока в основном создается применением правила 2 к первому b в подстроке "bbc что создает "aabc"на стыке.

Кроме того, чистые увеличения количества b требуют специфических конфигураций, таких как 'bbcbc', чтобы позволить одно правило 2, за которым следуют два правила 1, с чистым приростом +1 b (т.е. мы получим строку ccbbbbaa). Однако после такой последовательности результирующая строка имеет дополнительные b, но без c после кластера b (b следуют за aa, c сдвигаются влево).

Чтобы воссоздать требуемый шаблон для другого увеличения ('bbcbc' или подобного), нужны дополнительные применения правила 2 для создания 'bc' или 'bbc' подстрок путем замены неведущих b в кластере. Например, для строки ccbbbbaa из прошлого абзаца мы можем заменить 3-ю b в кластере на ccaa. Получим ccbbccaabaa. Мы получили подстроку bbc, однако, чтобы применить повторно правило 1, мы должны применить правило 2 ещё один раз для первой b: ccccaabccaabaa → ccccbbaacaabaa → ccccbbaabaa. Создание полного шаблона для увеличения требует больше применений правила 2, чем прирост от правила 1, что приводит к уменьшению количества b или отсутствию дальнейшего увеличения, поскольку конфигурации разделяют b и c (c слева, b справа, a между ними), предотвращая новые образования "bbc"или "bbcbc"после ограниченного числа таких циклов.

Количество последовательностей с чистым увеличением ограничено начальным количеством b и c, поскольку каждая такая последовательность потребляет буквы c, которые стоят после кластеров b, а воссоздание этого стоит больше b, чем набирается.

В итоге больше не создается aabc, правило 1 останавливается, затем правила 2 и 3 уменьшают количество b до 0, а правило 4 уменьшают оставшуюся строку, пока не перестанут применяться правила.

Таким образом, все последовательности переписывания конечны, что доказывает завершаемость.

3 Конечность классов эквивалентности по НФ

Так как в данной SRS существует правило $aac \to \varepsilon$, а для построения эквивалентностей мы считаем, что правила могут применяться в обе стороны, то по этому правилу мы можем построить бесконечное количество различных строк, следовательно для данной SRS существует бесконечное количество классов эквивалентности.

4 Локальная конфлюэнтность

Рассмотрим неоднозначно сходящуюся критическую пару:

$u = aabc,$

$v = u \xrightarrow{R2} aaccaac \xrightarrow{R4} c,$

$w = u \xrightarrow{R1} bbaa \xrightarrow{R2} ccaaccaaaa \xrightarrow{R4} cccaaaa,$

$v \neq w.$

v и w являются нормальными формами, однако они не равны, следовательно локальная конфлюэнтность не выполняется

5 Пополняемость по Кнуту-Бендиксу

Рассмотрим строку $aa(bc)^n$. В ней можно использовать бесконечное количество раз правила R1 и R3 и получить расходящуюся пару:

$aa(bc)^n \xrightarrow{R1} bbaa(bc)^{n-1} \xrightarrow{R1} \dots \xrightarrow{R1} b^{2n}aa$

$aa(bc)^n \xrightarrow{R3} aaa(bc)^{n-1} \xrightarrow{R3} \dots \xrightarrow{R3} a^{n+2}$

Следовательно алгоритм Кнута-Бендикса будет добавлять бесконечное число правил и процедура не завершится.

6 Инварианты

1) $I_1(s) = (2|\omega|_a + |\omega|_b + |\omega|_c) \bmod 5$

2) $I_2(s) = (2|\omega|_a + 6|\omega|_b + 6|\omega|_c) \bmod 10$