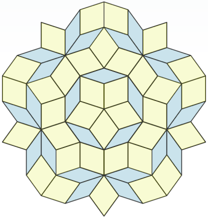


Структуры над словами: образцы и уравнения

Летняя практика, Переславль-Залесский

4–6 июля, 2022 г.

Проектирование структур с образцами

- Вопрос достижимости образца:

```
f (A : x) = Expr1
f [] = Expr2
f [A] = Expr3
```

- Вопрос накрытия образцами:

```
f ((x : y) : z) = Expr1
f [] = Expr2
```

- Вопрос перестановочности образцов:

```
f (x : (A : y)) = Expr1
f (A : (y : z)) = Expr2
```

...а с отказом от свободы и единственности вхождений переменных эти вопросы становятся намного сложнее.

Проектирование структур с образцами

- Вопрос достижимости образца:

```
f {x t t y = Expr1 }
f {x 'A' t z1 'A' y = Expr2 }
f {x 'A' y 'A' z = Expr3 }
```

- Вопрос накрытия образцами:

```
f {x1 (z) x2 t x3  = Expr1}
f {x (z)  = Expr2}
f {t x  = Expr3}
```

- Вопрос перестановочности образцов:

```
f {x, x 'A' : 'A' x = Expr1}
f (x, x 'AB' : 'BA' x = Expr2}
```

Необходимо определить выразительную силу образцов — языки, которые они описывают, и свойства этих языков.

Обозначения для подстановки

В курсе алгебры результат применения подстановки σ к терму t обозначают обычно $\sigma(t)$. У логиков (и в CS) принята постфиксная нотация: $t\sigma$. Мы будем использовать её.

Почему так?

Во многих классических работах по логике (Тарский, Карри) подстановка сразу записывалась в квадратных скобках $[t/A]$. В этом случае было более естественно приписывать её в конец выражения-аргумента: например, $F(t,t)[t/A]$.
В более современных работах приняты обозначения $[t := A]$ и $[t \mapsto A]$. В последнем случае форма стрелки существенна: $\to$ чаще используется для обозначения типов, а не отображений.

Базовые определения

$V_{\mathcal{T}}$ — множество переменных типа $\mathcal{T}$, $V = \bigcup^{\mathcal{T}} V_{\mathcal{T}}$. Рассматриваем e-переменные (типа строка/выражение) и t-переменные (типа терм). Т.е. $V = V_{\mathtt{e}} \cup V_{\mathtt{t}}$.

Кратность терма T в образце P обозначаем $|\mathsf{P}|_{\mathsf{T}}$.

Σ — по умолчанию неограниченный алфавит констант. $\mathcal{B}[S]$ — множество скобочных структур над строками из S.

Плоский образец P — строка в алфавите $V_{\mathtt{e}} \cup \mathcal{B}[\Sigma \cup V_{\mathtt{t}}]$. Образец P линеен, если $\forall x \in V_{\mathtt{e}}\ (|\mathsf{P}|_x = 1)$. Подстановка в образец — гомоморфизм, сохраняющий константы (т.е. для всех $\mathbf{A} \in \Sigma$ $\mathbf{A}\sigma = \mathbf{A}$).

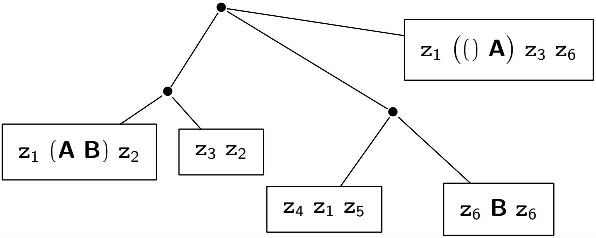
Образец допускает плоское разбиение, если он плоский, либо имеет вид $(\mathsf{P}_1)\ (\mathsf{P}_2) \dots\ (\mathsf{P}_n)\ \mathsf{P}_{n+1}$, где все P_i допускают плоское разбиение. Максимальные плоские подобразцы такого образца называем фрагментами плоского разбиения (ФПР).

Плоские разбиения и деревья

Рассмотрим следующий образец:

$$\Big(\big(z_1\ (\mathbf{A}\ \mathbf{B})\ z_2\big)\ z_3\ z_2\Big)\Big(\big(z_4\ z_1\ z_5\big)\ z_6\ \mathbf{B}\ z_6\Big)\ z_1\ \big(()\ \mathbf{A}\big)\ z_3\ z_6$$

Структура его ФПР приведена ниже.



Поскольку скобочные структуры могут возникнуть только сразу справа от открывающей скобки, то ФПР образуют древесные структуры, аналогичные АТД.

Пример образца, не разбиваемого на ФПР:

$$\big(x_1\ (\mathbf{A}\ x_2)\ x_1\ x_2\big)\ x_1$$

Языки, распознаваемые образцами

Определение

Языком $\mathscr{L}(\mathsf{P})$, ***распознаваемым образцом*** P, *назовем множество элементов* $\Phi \in \mathcal{B}[\Sigma]^*$, *для которых существует подстановка* σ: $\mathsf{P}\sigma = \Phi$. *Образец* P_1 *сводится к образцу* P_2, *если* $\mathscr{L}(\mathsf{P}_1) \subseteq \mathscr{L}(\mathsf{P}_2)$.

Подстановка $\mathtt{x}\sigma = \varepsilon$ допустима! В терминологии pattern languages — рассматриваются E-pattern languages (EPL, сокращение от Erasing Pattern Languages, языки стирающих образцов).

- Язык, распознаваемый образцом-строкой $\mathsf{P} \in \Sigma^*$, есть $\{\mathsf{P}\}$.
- Язык, распознаваемый образцом $\mathsf{P} = \mathtt{x}_1\ \mathtt{x}_2\ \mathtt{x}_1$, есть всё множество $\mathcal{B}[\Sigma]^*$.

Языки, распознаваемые образцами

Определение

***Языком** $\mathscr{L}(\mathsf{P})$, **распознаваемым образцом** P, назовем множество элементов $\Phi \in \mathcal{B}[\Sigma]^*$, для которых существует подстановка σ: $\mathsf{P}\sigma = \Phi$. Образец P_1 сводится к образцу P_2, если $\mathscr{L}(\mathsf{P}_1) \subseteq \mathscr{L}(\mathsf{P}_2)$.*

- Язык, распознаваемый образцом-строкой $\mathsf{P} \in \Sigma^*$, есть $\{\mathsf{P}\}$.
- Язык, распознаваемый образцом $\mathsf{P} = \mathtt{x}_1\ \mathtt{x}_2\ \mathtt{x}_1$, есть всё множество $\mathcal{B}[\Sigma]^*$.

С точки зрения семантики сопоставления, образец $\mathtt{x}_1\ \mathtt{x}_2\ \mathtt{x}_1$ также неудачный: $\mathtt{x}_1$ всегда успешно сопоставляется с ε. Бывает и иначе: хотя $\mathscr{L}(\mathtt{z}_1\ \mathtt{z}_2\ \mathtt{z}_2) = \mathscr{L}(\mathtt{x}_1\ \mathtt{x}_2\ \mathtt{x}_1) = \mathcal{B}[\Sigma]^*$ из-за существования тривиальной подстановки $\mathtt{z}_2 := \varepsilon$, но ленивое сопоставление строки **ABB** с $\mathtt{z}_1\ \mathtt{z}_2\ \mathtt{z}_2$ построит подстановку $\mathtt{z}_2 :=$ **B**, а вовсе не $\mathtt{z}_2 := \varepsilon$.

Сводимость и эквивалентность

Если P_1, P_2 оба из $(V_e \cup \mathcal{B}[\Sigma])^*$, то:

- P_1 сводится к P_2 $\Leftrightarrow$ существует подстановка σ такая, что $P_2\sigma = P_1$;
- если P_2 линеен, тогда вычислительная сложность проверки сводимости образца P_1 к образцу P_2 линейна от суммы длин P_1 и P_2.

Из-за того, что образцы стирающие (определяют EPL), двухсторонняя сводимость не эквивалентна наличию переименовки: вспомним те же $\mathtt{x}_1\ \mathtt{x}_2\ \mathtt{x}_1$ и $\mathtt{z}_1\ \mathtt{z}_2\ \mathtt{z}_2$.

Сводимость и эквивалентность

Если P_1, P_2 оба из $(V_e \cup \mathcal{B}[\Sigma])^*$, то:

- P_1 сводится к P_2 $\Leftrightarrow$ существует подстановка σ такая, что $P_2\sigma = P_1$;
- если P_2 линеен, тогда вычислительная сложность проверки сводимости образца P_1 к образцу P_2 линейна от суммы длин P_1 и P_2.

Если рассматривать только образцы без идущих подряд переменных из V_e, тогда уже для образцов без переменных из V_t выполняется утверждение

$$\mathscr{L}(P_1) = \mathscr{L}(P_2) \Leftrightarrow \exists\sigma(P_1\sigma = P_2 \,\&\, \forall x \in V_e(x\sigma \in V_e))$$

Краткие и избыточные образцы

Определение

Образец P_1 *называется кратким, если любой образец* P_2 *такой, что* $\mathscr{L}(\mathsf{P}_1) = \mathscr{L}(\mathsf{P}_2)$*, имеет длину, не меньшую, чем* P_1*.*
Иначе P_1 *называется избыточным.*

Пример

Образец $\boxed{\mathtt{x}_1\ \mathtt{x}_2}$ **A** $\mathtt{x}_3$ **B** $\boxed{\mathtt{x}_1\ \mathtt{x}_2}$ избыточен.

Образец $\mathtt{x}_1\ \mathtt{x}_2\ \mathtt{x}_2\ \mathtt{x}_1$ является кратким.

Алгебраисты также говорят, что избыточные образцы определяются нетривиальными неподвижными точками морфизмов над образцами (т.е. существует нетривиальная подстановка, переводящая избыточный образец в себя).

Критерий избыточности образца (Reidenbach, 2004)

Образец P избыточен, если существует представление $P = Q_0 \, \boxed{R_1} \, Q_1 \ldots \boxed{R_n} \, Q_n$, $Q_i \in \{\mathcal{B}[\Sigma] \cup V_e\}^*$, $R_i \in {V_e}^+{V_e}^+$, такое, что:

- множества переменных образцов Q_i и R_j не пересекаются;
- в каждом слове R_i найдется имеющая единственное вхождение в R_i переменная x_i *(выделенная)* такая, что

$$\forall j(|\boxed{R_j}|_{x_i} > 0 \Rightarrow \boxed{R_j} = \boxed{R_i}).$$

Этот критерий является необходимым и достаточным условием при рассмотрении плоских образцов в $\{\mathcal{B}[\Sigma] \cup V_e\}^*$[a].

[a] У Рейденбаха он доказан для образцов в V_e^*. Для образцов над $\mathcal{B}[\Sigma]$ доказательство где-то в моих старых тетрадях — здесь существенно, что скобки порождают бесконечный «алфавит констант».

Критерий Рейденбаха под лупой

Пусть искомое разбиение образца P существует. Тогда по каждому блоку $\boxed{R_i}$ построим подстановку σ_{fix} так: $x_i\sigma_{\text{fix}} = \boxed{R_i}$, а образы прочих переменных из R_i равны ε (они могут встречаться и в сочетании с другой выделенной переменной x_k в прочих $\boxed{R}$-блоках). Очевидно, $P\sigma_{\text{fix}} = P$.

Образец $P \in V_e^*$ допускает неоднозначную нестирающую подстановку $\Leftrightarrow$ образец P избыточен по Рейденбаху.

Критерий Рейденбаха под лупой

Образец $P \in V_e^*$ допускает неоднозначную нестирающую подстановку $\Leftrightarrow$ образец P избыточен по Рейденбаху.

- Для образцов, содержащих константные фрагменты, это не верно: $\mathtt{x}_1$ **A** $\mathtt{x}_2$ допускает много подстановок в $\mathbf{A}^n$, хотя является кратким. Однако это верно для фрагментов таких образцов, не содержащих констант.
- Стирающих подстановок может быть и несколько: например, $\mathtt{x}_1\ \mathtt{x}_2\ \mathtt{x}_2\ \mathtt{x}_1$ допускает две подстановки в **A B A B**. Однако поиск возможных подстановок в такой формулировке может быть сделан экспоненциально быстрее, чем без знания о критерии Рейденбаха.
- Иногда структура слова слишком однородна, чтобы критерий Рейденбаха гарантировал единственность подстановки: см. $\mathbf{A}^n$ и любой краткий образец без констант, содержащий как минимум две различные переменные.

Добавление переменных типа терм

За увеличение выразительной силы образцов приходится платить усложнением теоретических конструкций.

- $\mathscr{L}(\mathsf{P}_1) \subseteq \mathscr{L}(\mathsf{P}_2)$ уже не определяется подстановкой.

  $\mathsf{P}_1 = \mathbf{A}\ \mathtt{x}_1$, $\mathsf{P}_2 = \mathtt{x}_2\ \mathtt{t}$. Язык P_1 вкладывается в язык P_2, а подстановки нет.

- Нет (пока ещё) исследованного понятия избыточного и краткого образца. Более того, образцы для одного и того же языка не образуют нижнюю полурешётку.

  Образы $\mathtt{x}\ \mathtt{t}$ и $\mathtt{t}\ \mathtt{x}$ оба краткие.

Плавающие t-переменные

Определение

*Назовем переменную t_i в плоском линейном образце P **якорной**, если*

- *t_i имеет кратность, не меньшую 2;*
- *или в P существует подслово α, не содержащее переменных из V_e, такое, что $\alpha = \alpha_1\ t_i\ \alpha_2$, причем α_1 и α_2 оба содержат хотя бы один символ или t-переменную, имеющую кратность не меньше 2.*

*В противном случае назовем t_i **плавающей**.*

Пример

Рассмотрим образец $t_1\ t_2\ x_1\ t_3\ t_4\ t_2\ x_2\ t_5$. Якорными переменными являются t_2 и t_1.

Плавающие переменные и языки образцов

Плавающая переменная в образце — указатель на то, что в соответствующий фрагмент образца нельзя подставить пустое слово. Аналог «нестираемых» фрагментов.

Плавающий сегмент линейного образца P — максимальное подслово P, содержащее только плавающие t-переменные и переменные из V_e.

Образец, в котором все e-переменные входят в плавающие сегменты — аналог нестирающего (non-erasing) образца. Хуже всего, если есть и стирающие, и нестирающие фрагменты.

Плавающие переменные и языки образцов

Плавающий сегмент линейного образца P — максимальное подслово P, содержащее только плавающие t-переменные и переменные из V_{e}.

Образец, в котором все e-переменные входят в плавающие сегменты — аналог нестирающего (non-erasing) образца. Хуже всего, если есть и стирающие, и нестирающие фрагменты.

Язык образца $P_1 = \mathbf{BBA}\,\mathtt{x}\,\mathbf{ABCDA}$ вкладывается в язык образца $P_2 = \mathtt{z}_0\,\mathtt{t}\,\mathtt{t}\,\mathtt{z}_1\,\mathtt{t}_1\,\mathtt{t}_2\,\mathtt{t}_3\,\mathtt{t}\,\mathtt{z}_2$. Чтобы это доказать, приходится перебирать два случая: пустоты и непустоты подставляемого в x значения.

Multi-pattern languages (Kari, Salomaa)

Определение

Языком $\mathscr{L}(\mathsf{P})$*, распознаваемым множеством образцов* P_i *(англ. — multi-pattern language, сокращенно MPL), назовем множество элементов* $\Phi \in \mathcal{B}[\Sigma]^*$*, для которых существует* $\mathtt{i} \in \mathbb{N}$ *и подстановка* σ*:* $\sigma(\mathsf{P}_\mathtt{i}) = \Phi$*.*

Множество MPL–объединений стирающих образцов совпадает с множеством MPL–объединений нестирающих образцов. Образец с плавающими $\mathtt{t}$-переменными тоже определяет мультиобразец, и здесь уже смешивание стирающих и нестирающих фрагментов может быть разрешено.

Однако переход от стирающих образцов к нестирающим порождает экспоненциальное разрастание описания MPL.

Пример «плавающего» MPL

Пусть $P_1 = x_1$ **A A C** x_2 **C A B** x_3 **B B C**,
$P_2 = z_1\ t_n\ t_n\ z_2\ t_{m1}\ z_3\ t_{m2}\ z_4\ t_{m3}\ z_5\ t_n\ z_6$.

Множество нестирающих образцов, порождающих P_2:

$$\begin{array}{l|l}
P_2^1 & t_n\ t_n\ z_1\ z_2\ z_3\ t_n \\
P_2^2 & t_n\ t_n\ z_1\ z_2\ z_3\ t_n\ z_4 \\
P_2^3 & z_0\ t_n\ t_n\ z_1\ z_2\ z_3\ t_n \\
P_2^4 & z_0\ t_n\ t_n\ z_1\ z_2\ z_3\ t_n\ z_4
\end{array}$$

Множество нестирающих образцов, порождающих P_1, и обобщающие их подобразцы из P_2:

$$\begin{array}{l|l}
\mathbf{A\ A\ C\ C\ A\ B\ B\ B\ C} & P_2^3\sigma_1,\ t_n\sigma_1 = \mathbf{C} \\
\mathbf{A\ A\ C\ C\ A\ B}\ x_3\ \mathbf{B\ B\ C} & P_2^3\sigma_2,\ t_n\sigma_2 = \mathbf{C} \\
\mathbf{A\ A\ C}\ x_2\ \mathbf{C\ A\ B\ B\ B\ C} & P_2^2\sigma_3,\ t_n\sigma_3 = \mathbf{A} \\
\mathbf{A\ A\ C}\ x_2\ \mathbf{C\ A\ B}\ x_3\ \mathbf{B\ B\ C} & P_2^2\sigma_4,\ t_n\sigma_4 = \mathbf{A} \\
x_1\ \mathbf{A\ A\ C\ C\ A\ B\ B\ B\ C} & P_2^3\sigma_5,\ t_n\sigma_5 = \mathbf{C} \\
x_1\ \mathbf{A\ A\ C\ C\ A\ B}\ x_3\ \mathbf{B\ B\ C} & P_2^3\sigma_6,\ t_n\sigma_6 = \mathbf{C} \\
x_1\ \mathbf{A\ A\ C}\ x_2\ \mathbf{C\ A\ B\ B\ B\ C} & P_2^4\sigma_7,\ t_n\sigma_7 = \mathbf{A} \\
x_1\mathbf{A\ A\ C}\ x_2\ \mathbf{C\ A\ B}\ x_3\ \mathbf{B\ B\ C} & P_2^4\sigma_8,\ t_n\sigma_8 = \mathbf{A}
\end{array}$$

Размер алфавита

Все хорошие свойства образцов, позволяющие работать с ними обычными методами (поиск подстановки, разбиение Рейденбаха) — следствие того, что мы подразумеваем $|\Sigma| = O(|\Sigma_{\mathtt{Prog}}|^2)$, где Σ — алфавит входных данных, $\Sigma_{\mathtt{Prog}}$ — множество символов, явно входящих в образцы. Допущение реалистичное, учитывая, что «буквами» выступают и константные деревья.

Языки образцов $\mathtt{x}$ **A B** $\mathtt{y}$ **A** $\mathtt{z}$ и $\mathtt{x}$ **A** $\mathtt{y}$ **B A** $\mathtt{z}$ в алфавите {**A**, **B**, **C**} очевидно не сравнимы: первый распознаёт слово **ABCA**, второй распознаёт **ACBA**. А в алфавите {**A**, **B**} эти образцы описывают один и тот же язык[a].

[a] И поэтому, если алфавит входных данных явно присутствует в образцах, нужны другие способы проверки подстановок на однозначность.

О распознаваемых словах

Предположим, мы рассматриваем краткий образец $\mathsf{P} = \mathrm{x}_1\ \mathrm{x}_2\ \mathrm{x}_2\ \mathrm{x}_1$. Если он сопоставляется со словом $\mathbf{A}^n$, то, как уже было видно, сопоставлений может быть много. С какими ещё словами происходит такая же ситуация?

Чтобы ответить на указанный вопрос, предположим, что слово сопоставилось с P двумя разными способами. То есть нашлись x_1, x_2, z_1, z_2 такие, что

$$x_1\ x_2\ x_2\ x_1 = z_1\ z_2\ z_2\ z_1$$

Что нам даёт такое равенство и как его упрощать? Ответы на этот вопрос потребуют краткое введение в теорию уравнений в словах.

Уравнения в словах. Определения

Даны алфавит констант Σ (здесь рассматриваем константные скобочные структуры типа `('A'('B'))` в роли констант), алфавит строковых и буквенных переменных V.

Уравнение в словах — равенство вида $\Psi = \Phi$, где $\Psi, \Phi \in \{\Sigma \cup V\}^*$.

Решить уравнение в словах — найти все такие подстановки σ переменных, входящих в Ψ Φ, что $\Psi\sigma = \Phi\sigma$.

Проблема существования корней уравнений в словах разрешима (Маканин). Множество решений уравнения в общем виде представляется графом сложной структуры.

Уравнение $\Psi = \Phi$ **квадратичное**, если ни одна переменная из V типа строка не входит в $\Psi = \Phi$ более, чем дважды.

Преобразование Нильсена

Лемма Леви

Если уравнение в словах имеет вид $x\,\Phi_1 = y\,\Phi_2$, тогда выполнено хотя бы одно из условий $x = y\,x' \vee y = x\,y'$ (возможно, для разных решений выполнены разные условия).

Рассмотрим произвольное решение уравнения, описываемое подстановкой σ. Если $|x\sigma| \geqslant |y\sigma|$, то $x = y\,x'$, в противном случае $|y\sigma| \geqslant |x\sigma|$ (даже строго больше), поэтому $y = x\,y'$.

Ветвление по подстановкам вида $x \mapsto y\,x'$ и $y \mapsto x\,y'$ называется преобразованием Нильсена. Оно может применяться как слева (по умолчанию), так и справа.

Системы уравнений

Пусть в Σ есть хотя бы две буквы. По каждой системе

$$\begin{aligned} \Phi_1 &= \Psi_1 \\ \Phi_2 &= \Psi_2 \end{aligned} \qquad (1)$$

можно построить уравнение, имеющее те же самые корни:

$$\Phi_1\, \mathbf{A}\, \Phi_2\, \Phi_1\, \mathbf{B}\, \Phi_2 = \Psi_1\, \mathbf{A}\, \Psi_2\, \Psi_1\, \mathbf{B}\, \Psi_2 \qquad (2)$$

Заметим, что $|\Phi_1\, \mathbf{A}\, \Phi_2| = |\Phi_1\, \mathbf{B}\, \Phi_2|$, и $|\Psi_1\, \mathbf{A}\, \Psi_2| = |\Psi_1\, \mathbf{B}\, \Psi_2|$.
Поэтому уравнение (2) эквивалентно системе:
$\Phi_1\, \mathbf{A}\, \Phi_2 = \Psi_1\, \mathbf{A}\, \Psi_2$ и $\Phi_1\, \mathbf{B}\, \Phi_2 = \Psi_1\, \mathbf{B}\, \Psi_2$.
Предположим, что есть решение этой системы, определяемое подстановкой σ. Поскольку σ сохраняет константы, получаем
$(\Phi_1\sigma)\, \mathbf{A}\, (\Phi_2\sigma) = (\Psi_1\sigma)\, \mathbf{A}\, (\Psi_2\sigma)$,
$(\Phi_1\sigma)\, \mathbf{B}\, (\Phi_2\sigma) = (\Psi_1\sigma)\, \mathbf{B}\, (\Psi_2\sigma)$.

Системы уравнений

Пусть в Σ есть хотя бы две буквы. По каждой системе

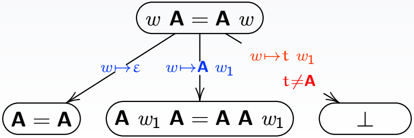
$$\begin{aligned} \Phi_1 &= \Psi_1 \\ \Phi_2 &= \Psi_2 \end{aligned} \tag{1}$$

можно построить уравнение, имеющее те же самые корни:

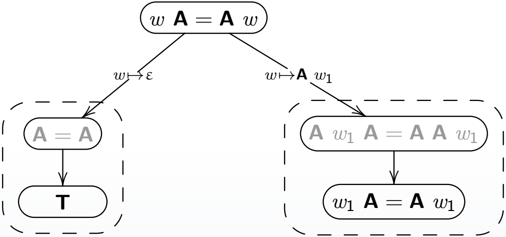
$$\Phi_1 \, \mathbf{A} \, \Phi_2 \, \Phi_1 \, \mathbf{B} \, \Phi_2 = \Psi_1 \, \mathbf{A} \, \Psi_2 \, \Psi_1 \, \mathbf{B} \, \Psi_2 \tag{2}$$

Уравнение (2) эквивалентно системе:
$\Phi_1 \, \mathbf{A} \, \Phi_2 = \Psi_1 \, \mathbf{A} \, \Psi_2$ и $\Phi_1 \, \mathbf{B} \, \Phi_2 = \Psi_1 \, \mathbf{B} \, \Psi_2$.
Предположим, что есть решение этой системы, определяемое подстановкой σ. Поскольку σ сохраняет константы, получаем
$(\Phi_1\sigma) \, \mathbf{A} \, (\Phi_2\sigma) = (\Psi_1\sigma) \, \mathbf{A} \, (\Psi_2\sigma)$,
$(\Phi_1\sigma) \, \mathbf{B} \, (\Phi_2\sigma) = (\Psi_1\sigma) \, \mathbf{B} \, (\Psi_2\sigma)$. Пусть $\Phi_1\sigma = (\Psi_1\sigma) \, \mathbf{A} \, \Theta$, тогда $\mathbf{A} \, \Theta \, \mathbf{B} \, (\Phi_2\sigma) = \mathbf{B} \, (\Psi_2\sigma)$, что невозможно. Аналогично, если $\Psi_1\sigma = (\Phi_1\sigma) \, \mathbf{A} \, \Theta$. Поэтому $\Phi_1\sigma = \Psi_1\sigma$ и $\Phi_2\sigma = \Psi_2\sigma$, значит, σ — решение исходной системы (1).

Дерево решения уравнения

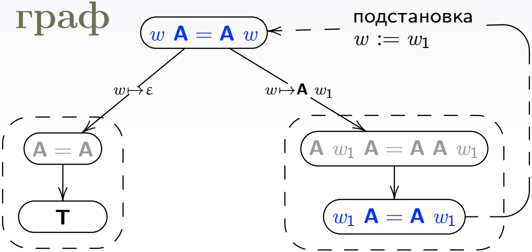


Если значение переменной w не пусто, то начинается с $\mathbf{A}$, либо уравнение не выполняется.



После подстановок $w \mapsto \varepsilon$, $w \mapsto \mathbf{A}\ w_1$ равные термы слева и справа сокращаются. Ветви дерева, приводящие к противоречию, отбрасываются.

Свёртка дерева решения уравнения в граф



Уравнение w_1 **A** $=$ **A** w_1 повторяет исходное с точностью до переименования w_1 в w. Его развёртка происходит точно так же, поэтому можно просто сослаться на узел с исходным уравнением.

Построен граф описания корней уравнения w **A** $=$ **A** w. Наличие в нем листьев, содержащих **T**, свидетельствует о существовании корней уравнения. Множество корней: $w \in \mathbf{A}^*$.

Метод Матиясевича

Дано уравнение в словах $\Phi_1^0 = \Phi_2^0$, $\Phi_i \in \{\Sigma \cup V\}^*$. Помещаем это уравнение в корень дерева.

Правила развертки узла с уравнением $t_1\ \Phi_1 = t_2\ \Phi_2$:

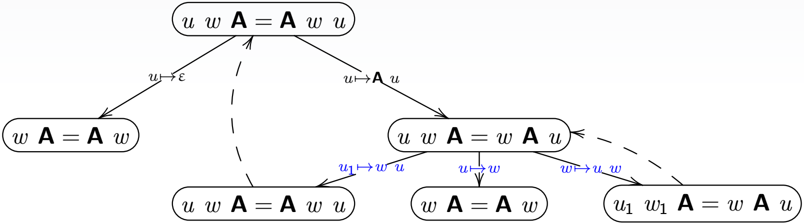
- Пусть $t_1 = t_2$. Заменяем уравнение в узле на $\Phi_1 = \Phi_2$.
- Уравнение $\Psi_1 = \Psi_2$, где $\Psi_i \in \{V_e\}^*$, $\Psi_j = \varepsilon$, объявляем листом и помечаем **T**.
- Пусть $t_i = \mathtt{t}$ (буквенная переменная), $t_j = \mathbf{A}$. Строим потомок $\Phi_1\sigma = \Phi_2\sigma$, где $\mathtt{t}\sigma = \mathbf{A}$, а для остальных $\nu \in V\ \nu\sigma = \nu$. Аналогично для $t_j = \mathtt{t}'$.
- Пусть $t_i = x$, $t_j = (\mathbf{A}\,|\,\mathtt{t})$ (буква или буквенная переменная). Строим два потомка, соответствующих подстановкам $x\sigma_1 = \varepsilon$, $x\sigma_2 = (\mathbf{A}\,|\,\mathtt{t})\ x$.
- Пусть $t_i = x$, $t_j = y$. Строим потомки по подстановкам $x\sigma_1 = y\ x$, $x\sigma_2 = x\ y$, $x\sigma_3 = y$.
- Если уравнение в текущем узле повторяет уравнение в узле-предке, строим обратное ребро в графе.

Соглашение об именах

При построении подстановок согласно преобразованию Нильсена (вида $x \mapsto yx'$) обычно за суффиксом x' закрепляется имя x. Например, в методе Матиясевича (см. ниже) таким образом удаётся избавиться от необходимости искать переименовки, чтобы факторизовать пути развёртки уравнений.

- Плюсы: дешёвое сравнение меток узлов, сохранение информации о том, суффикс какой исходной переменной рассматривается.
- Минус: иногда свёртка по переименовке (а не по точному совпадению) позволяет получить намного более короткое описание решений.

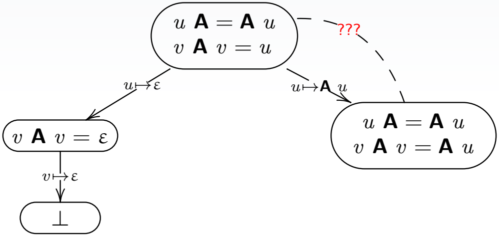
Алгоритм Матиясевича и граф развёртки



$u = \mathbf{A}$, $w = \mathbf{B}$ — решение уравнения $u\ w\ \mathbf{A} = \mathbf{A}\ w\ u$.

Этому решению не соответствует никакого пути в графе развёртки, оканчивающегося узлом $\mathbf{T}$. Причина — в нулевом индуктивном переходе $w \mapsto \varepsilon w$ (после отцепления от u буквы $\mathbf{A}$). Для того, чтобы получить корректное описание решений, в метод Матиясевича нужно добавить выходы по $u \mapsto \varepsilon$, $w \mapsto \varepsilon$.

Нетривиальность совместной развёртки уравнений



Как только в системе появляется переменная, встречающаяся больше двух раз по совокупности, алгоритм построения графа развертки перестаёт работать, потому что появляется возможность бесконечного разрастания уравнений.

Для квадратичных уравнений алгоритм Матиясевича является разрешающим (всегда завершается и даёт точный ответ о существовании корней).

Сложность задачи

Длина минимального решения уравнения:

$$x_n\,\mathbf{A}\,x_n\,\mathbf{B}\,x_{n-1}\,\mathbf{B}\,x_{n-2}\ \ldots\ \mathbf{B}\,x_1 = \mathbf{A}\,x_n\,x_{n-1}^2\,\mathbf{B}\,x_{n-2}^2\,\mathbf{B}\ \ldots\ \mathbf{B}\,x_1^2\,\mathbf{B}\,\mathbf{A}^2$$

экспоненциальна по n.

Специальные уравнения

- Уравнения сопряжения со словом: $\Phi_1\Phi_2\, x = x\, \Phi_2\Phi_1$, где $\Phi_i \in \Sigma^*$. Они имеют решения вида $(\Phi_1\Phi_2)^n\Phi_1$ (причём если $\Phi_1\Phi_2$ — это степень, то берётся её корень).
  Например, рассмотрим $\mathbf{A\,B\,A\,B}\, x = x\, \mathbf{B\,A\,B\,A}$.
  Формально решения должны иметь вид $(\mathbf{A\,B\,A\,B})^n\, \mathbf{A}$, однако очевидно, что при таком описании теряется, например, корень $\mathbf{A\,B\,A}$. Правильный ответ здесь: $(\mathbf{A\,B})^n\, \mathbf{A}$.

- Уравнения сопряжения с переменными: $x\, w = w\, y$. Их решения описываются в параметрах: $x = z_1\, z_2$, $y = z_2\, z_1$, $w = (z_1\, z_2)^n z_1$. Видно, что это просто обобщение предыдущего класса уравнений.

Хорошие классы уравнений

- Уравнения от одной переменной. Разрешимы за линейное время.
- Квадратичные уравнения (каждая переменная имеет не больше чем 2 вхождения в обе части). Сложность решения: NP-трудна.
- Straight-line уравнения. Кратность (относительно всего уравнения) всех переменных хотя бы в одной части уравнения равна 1.
- Уравнения, сводящиеся к системе, где одно уравнение принадлежит классу выше, а остальные — уравнения вида $x_i \Phi_i = \Psi_i x_i$, где Φ_i, Ψ_i — константы.

Задача на однозначность

Какие слова сопоставляются с образцом $x_1\ x_2\ x_2\ x_1$ неоднозначно?

Решим уравнение неоднозначности: $x_1\,x_2\,x_2\,x_1 = z_1\,z_2\,z_2\,z_1$.
Заметим, что это уравнение эквивалентно системе

$$\begin{cases} x_1\,x_2 = z_1\,z_2 \\ x_2\,x_1 = z_2\,z_1 \end{cases} \tag{3}$$

Допустив, что $|x_1| > |z_1|$, применим двухстороннюю лемму Леви:

$$\begin{cases} x_1 = z_1\,x_{1,s} \\ x_1 = x_{1,p}\,z_1 \end{cases} \tag{4}$$

Значит,
$x_{1,s} = u\,v$, $x_{1,p} = v\,u$, $z_1 = (v\,u)^n\,v$. Тогда $x_1 = (v\,u)^{n+1}\,v$. При этом полагаем $(v\,u)$ простым словом.

Задача на однозначность

Какие слова сопоставляются с образцом $x_1\, x_2\, x_2\, x_1$ неоднозначно?

Решим уравнение неоднозначности: $x_1\, x_2\, x_2\, x_1 = z_1\, z_2\, z_2\, z_1$. Заметим, что это уравнение эквивалентно системе

$$\begin{cases} x_1\, x_2 = z_1\, z_2 \\ x_2\, x_1 = z_2\, z_1 \end{cases} \tag{3}$$

$x_{1,s} = u\, v$, $x_{1,p} = v\, u$, $z_1 = (v\, u)^n\, v$. Тогда $x_1 = (v\, u)^{n+1}\, v$. При этом полагаем $(v\, u)$ простым словом. Подставляя полученные значения в систему (3), имеем $x_2\, v\, u = u\, v\, x_2$. Это опять уравнение сопряжения. Если сопряжение такое же, как для z_1 (т.е. x_2 имеет вид $(v\, u)^m\, u$), тогда мы получаем, что $x_1\, x_2\, x_2\, x_1 = (v\, u)^{n+2}\, (u\, v)^{n+2}$. Частный случай этого решения — если $v = \varepsilon \vee u = \varepsilon$.

Задача на однозначность

Какие слова сопоставляются с образцом $x_1\ x_2\ x_2\ x_1$ неоднозначно?

Решим уравнение неоднозначности: $x_1\,x_2\,x_2\,x_1 = z_1\,z_2\,z_2\,z_1$. Заметим, что это уравнение эквивалентно системе

$$\begin{cases} x_1\,x_2 = z_1\,z_2 \\ x_2\,x_1 = z_2\,z_1 \end{cases} \tag{3}$$

Пусть $x_1 = (v\,u)^{n+1}v$, $x_2\,v\,u = u\,v\,x_2$, причём $v\,u$ — простое и $v \neq \varepsilon$, $u \neq \varepsilon$. Предположим теперь, что существуют такие v_1, u_1, что $v\,u = v_1\,u_1$, $u\,v = u_1\,v_1$, но $|v| \neq |v_1|$. Если окажется, что u_1 или v_1 — пустое слово, тогда $v\,u = u\,v$, поэтому $v\,u$ — не простое. Иначе можно заметить, что система уравнений на u, v, u_1, v_1 в точности повторяет (3), но $|v| < |x_1|$. По индукции, этот процесс рано или поздно приведёт к $|v_i| = \varepsilon$ или $|u_i| = \varepsilon$ (и будет получено противоречие с начальными условиями).

Задача на однозначность

Какие слова сопоставляются с образцом $x_1\ x_2\ x_2\ x_1$ неоднозначно?

Решим уравнение неоднозначности: $x_1\,x_2\,x_2\,x_1 = z_1\,z_2\,z_2\,z_1$.
Заметим, что это уравнение эквивалентно системе

$$\begin{cases} x_1\,x_2 = z_1\,z_2 \\ x_2\,x_1 = z_2\,z_1 \end{cases} \tag{3}$$

Таким образом, удалось показать, что решениями системы (3) будут лишь слова вида $(v\,u)^{n+2}\,(u\,v)^{n+2}$, а значит, при сопоставлении только с такими словами образца $x_1\ x_2\ x_2\ x_1$ может возникнуть неоднозначность (причём перебор при сопоставлении в этом случае достаточно делать лишь по степеням $v\,u$).

Уравнения как способ описания однозначности образца

Скажем, что образец $P(x_1, \ldots, x_n)$ однозначный, если уравнение $P(x_1, ..., x_n) = P(z_1, ..., z_n)$ имеет только решение $x_i = z_i$.

- Любой образец от одной переменной однозначен;
- Образец $(x_1\,x_2)(x_2\,x_2\,x_1)$ однозначен, поскольку $|x_2|$ определяется как разность длин строк, сопоставляемых с его ФПР $x_1\,x_2$ и $x_2\,x_2\,x_1$;
- Образец $(x_1\,x_2)(x_2\,x_3)(x_3\,x_1)$ однозначен, поскольку решение на длины переменных, входящих в него, всегда единственно, если существует;
- Образец $(x_1\,x_2)(x_2\,x_1)$ неоднозначен (см. выше).

Однозначность длин

Пусть образец P содержит константные строки только вида $(\mathcal{P})^i$ для некоторого простого слова $\mathcal{P}$. Тогда вопрос об его однозначности можно свести к вопросу об единственности решения уравнения на длины входящих в него переменных.

Действительно, рассмотрим слова вида $(\mathcal{P})^{m_j}$, сопоставляемые с ФПР такого образца. По предположению о простоте $\mathcal{P}$, переменные образца получат значения вида $(\mathcal{P})^{k_i}$. В силу уравнения коммутативности для любых двух таких значений, получаем, что существует отображение μ сопоставлений вида $P_j \ : \ (\mathcal{P})^{m_j}$ в диофантовы уравнения $\mu(P_j) = m_j$ над $\mathbb{N}$. Такое отображение переводит переменные в себя, конкатенацию — в сложение, а константные фрагменты $(\mathcal{P})^i$ — в натуральные числа i.

Матрица кратностей

Имея ФПР $P_1, P_2, \ldots, P_n$ образца $P(x_1, \ldots, x_m)$, можно построить матрицу $\mathcal{M}(P)$ кратностей переменных образца: $a_{i,j} = |P_i|_{x_j}$.

Если ранг матрицы $\mathcal{M}(P)$ равен m, то образец P всегда сопоставляется однозначно.

Если ранг матрицы $\mathcal{M}(P)$ меньше m, то имеет смысл искать такие подмножества ФПР P_i, ранг матрицы кратности которых равен числу входящих в них различных переменных. Тогда все переменные, входящие в такие подмножества, будут сопоставлены однозначно. То есть задача поиска однозначных сопоставлений может быть аппроксимирована сверху задачей поиска ранга $\mathcal{M}(P)$.
(Для поиска целочисленных решений аналогично: ссылка на быстрый алгоритм)

Некоммутативный случай

Коммутативный случай является «худшим» с точки зрения сложности сопоставлений. Но худший случай не всегда достигается, если в образцах встречаются разнородные константные фрагменты.

Образец x_1 **A** $x_2\, x_1$ **B** x_2 однозначен (см. выше), хотя порождает уравнение на длины $2x_1 + 2x_2 + 2 = M$, имеющее много решений.