Взаимно однозначный англо-русский транслитератор

Пономарев А.С.

2023 год

1 Введение

Однажды автор этого текста задался вопросом, можно ли построить однозначную транслитерацию между русским и английским алфавитами. Иначе говоря, возможно ли взаимно однозначно сопоставить друг другу русские и английские слова, чтобы при этом получившееся соответствие было как можно ближе к привычному «транслиту», то есть к переписыванию побуквенно. Под словом, естественно, имеется в виду формальное слово, то есть любая конечная последовательность символов языка.

Ответ на поставленный вопрос – да, можно. Но насколько подобная транслитерация может соответствовать нашим желаниям и насколько она может быть простой – вопросы более сложные и к тому же совершенно расплывчатые. В этой работе предлагается определенная математическая конструкция и на ее основе строится одна из бесчисленного множества взаимно однозначных англо-русских транслитераций, которая, по мнению автора, и проста, и лингвистически удачна.

Зачем нужна такая транслитерация? В наш век высоких технологий большинство компьютерных систем оперируют английским языком. Подавляющее большинство языков программирования созданы на основе английского языка, базовый набор символов в компьютерах – латиница. Человек общается с машиной по-английски. При внедрении русского языка важно поддерживать совместимость имен: чтобы, например, программную функцию с английским названием можно было вызвать по-русски и наоборот. При транслитерации имен взаимная однозначность имеет ключевое значение.

Автор надеется, что работа послужит шагом на пути к созданию полноценного языка программирования на русском языке.

2 Алфавит. Язык

Для начала перечислим основные определения и условные обозначения.

Определение 1. *Алфавит* $\mathcal{X}$ – непустое конечное множество; элементы алфавита называются *символами*, или *буквами*. В дальнейшем будем работать с двумя конкретными алфавитами: русским $\mathcal{R} = \{\text{а}, \text{б}, \text{в}, ..., \text{я}\}$ и английским $\mathcal{A} = \{\text{a}, \text{b}, \text{c}, ..., \text{z}\}$. $|\mathcal{X}|$ – размер алфавита, то есть количество символов в нем. $|\mathcal{R}| = 33;\ |\mathcal{A}| = 26$.

Определение 2. *Словом* над алфавитом $\mathcal{X}$ называется конечная последовательность символов из $\mathcal{X}$: $\omega = \widetilde{x}_1\widetilde{x}_2...\widetilde{x}_n,\ n \in \mathbb{N}_0,\ \widetilde{x}_1, \widetilde{x}_2, ..., \widetilde{x}_n \in \mathcal{X}$. В частности, при $n = 0$ получается пустое слово ε. Слова будем обозначать строчными греческими буквами.

Определение 3. Пусть $\omega_1 = \widetilde{x}_1\widetilde{x}_2...\widetilde{x}_n$ и $\omega_2 = \widetilde{y}_1\widetilde{y}_2...\widetilde{y}_m$ – слова над алфавитом $\mathcal{X}$. *Конкатенацией* ω_1 и ω_2 называется слово $\omega =$
$= \widetilde{x}_1\widetilde{x}_2...\widetilde{x}_n\widetilde{y}_1\widetilde{y}_2...\widetilde{y}_m$, то есть полученное записью ω_1 и ω_2 «одного за другим». Пишут $\omega = \omega_1\omega_2$.

Определение 4. Пусть ω_1 и ω_2 – слова над алфавитом $\mathcal{X}$. Будем говорить, что ω_2 есть *подслово* ω_1, если существуют слова α, β над $\mathcal{X}$, такие что $\omega_1 = \alpha\omega_2\beta$.

Определение 5. *Языком* над алфавитом $\mathcal{X}$ называется произвольное множество слов над $\mathcal{X}$. Множество *всех* слов над алфавитом $\mathcal{X}$ обозначим $\mathcal{X}^*$. Язык $\mathcal{R}^*$ будем для краткости называть русским языком, $\mathcal{A}^*$ – английским.

Замечание 1. Любой язык $\mathcal{X}^*$ – счетно бесконечное множество.

Взаимно однозначная англо-русская транслитерация есть биективная функция $F : \mathcal{R}^* \longrightarrow \mathcal{A}^*$. Иначе говоря, нам нужно каждому русскому слову ρ поставить в соответствие некоторое английское слово $\alpha = F(\rho)$, так чтобы:
1) $\forall \rho_1, \rho_2 \in \mathcal{R}^*\ \ (\rho_1 \neq \rho_2 \Longrightarrow F(\rho_1) \neq F(\rho_2))$ – инъективность;
2) $\forall \alpha \in \mathcal{A}^*\ \ \exists \rho \in \mathcal{R}^* :\ \alpha = F(\rho)$ – сюръективность.

3 Разбиение слова. Система правил

Пусть $\mathcal{X}$ и $\mathcal{Y}$ – произвольные алфавиты. Хотелось бы иметь инструмент для удобного описания отображений $F : \mathcal{X}^* \longrightarrow \mathcal{Y}^*$.

Определение 6. *Правилом* из $\mathcal{X}^*$ в $\mathcal{Y}^*$ будем называть пару (ξ, η), где ξ и η – непустые слова, $\xi \in \mathcal{X}^*$, $\eta \in \mathcal{Y}^*$. Можно записать более наглядно: $\xi \longrightarrow \eta$. Правило есть «предписание к переводу».

Далее логично рассмотреть множество правил и сделать так, чтобы оно некоторым образом порождало функцию $F : \mathcal{X}^* \longrightarrow \mathcal{Y}^*$.

Определение 7. Пусть ω – слово над алфавитом $\mathcal{X}$. *Разбиением* слова ω на n частей назовем конечную последовательность $T = (t_0, t_2, ..., t_n)$, $n \in \in \mathbf{N}$, где $t_0 = 0,\ t_n = |\omega|,\ t_0 < t_1 < ... < t_n$. Применяя разбиение к слову, получаем набор из n соответствующих подслов: пусть $\omega = \widetilde{x}_1\widetilde{x}_2...\widetilde{x}_m,\ \widetilde{x}_i \in \in \mathcal{X},\ m \in \mathbf{N}$, тогда $T(\omega) = (\omega_1, \omega_2, ..., \omega_n)$, где $\omega_i = \widetilde{x}_{t_{i-1}+1}\widetilde{x}_{t_{i-1}+2}...\widetilde{x}_{t_i}$.

Пример 1. Пусть $\omega =$ абвг, $T = (0, 1, 3, 4)$. Тогда $T(\omega) =$ (а, бв, г).

Определение 8. Будем говорить, что из двух разбиений T_1 и T_2 слова ω T_1 *больше* T_2, и писать $T_1 > T_2$, если $T_1 \subsetneq T_2$, то есть T_1 строго вложено в T_2 как множество. Если $T_1 > T_2$, то T_2 можно получить из T_1, добавив один или несколько «разрезов» слова ω.

Замечание 2. Определенный выше порядок на множестве всех разбиений произвольного слова ω не является линейным: в общем случае существуют несравнимые разбиения. Например, $|\omega| = 3,\ T_1 = (0, 1, 3)$, $T_2 = (0, 2, 3)$.

Определение 9. Пусть $\mathcal{T}$ – некоторое множество разбиений слова ω, $T_0 \in \mathcal{T}$. T_0 называется *максимальным элементом* множества $\mathcal{T}$, если $\forall T \in \mathcal{T}\ (T \not> T_0)$, то есть не существует элемента больше T_0. T_0 называется *наибольшим элементом* множества $\mathcal{T}$, если $\forall T \in \mathcal{T}\ (T_0 > T)$, то есть если T_0 больше любого элемента.

Замечание 3. Из того, что элемент T_0 наибольший, следует, что он максимальный. Обратное в общем случае неверно, так как порядок на множестве $\mathcal{T}$ не линейный.

Определение 10. Пусть $\Gamma = \{(\xi_1, \eta_1), (\xi_2, \eta_2), ...\}$ – конечное или счетно бесконечное множество правил из $\mathcal{X}^*$ в $\mathcal{Y}^*$, $\omega \in \mathcal{X}^*$. Разбиение T слова ω называется *допустимым* относительно множества правил Γ, если $\exists i_1, i_2, ..., i_n \in \mathbb{N} :\ T(\omega) = (\xi_{i_1}, \xi_{i_2}, ..., \xi_{i_n}),\ (\xi_{i_1}, \eta_{i_1}), ..., (\xi_{i_n}, \eta_{i_n}) \in \Gamma$.

Определение 11. *Корректной системой правил*, или просто *системой правил*, из $\mathcal{X}^*$ в $\mathcal{Y}^*$ называется конечное или счетно бесконечное множество правил из $\mathcal{X}^*$ в $\mathcal{Y}^*$ $\Gamma = \{(\xi_1, \eta_1), (\xi_2, \eta_2), ...\}$, такое что:

1) все левые части правил ξ_i различны;

2) для любого слова $\omega \in \mathcal{X}^*$ в множестве его допустимых относительно Γ разбиений существует наибольший элемент.

Замечание 4. Существование наибольшего элемента в множестве допустимых разбиений в том числе означает, что это множество непустое.

Пример 2. Пусть $\mathcal{X} = \{\text{к}, \text{с}\} \subset \mathcal{R},\ \mathcal{Y} = \mathcal{A}$. Тогда $\Gamma = \{\text{к} \longrightarrow \text{k},\ \text{с} \longrightarrow \longrightarrow \text{s},\ \text{кс} \longrightarrow \text{x}\}$ – корректная система правил. Действительно, если в слово $\omega \in \mathcal{X}^*$ подслово кс входит m раз, то существует ровно 2^m допустимых разбиений (каждое кс можно оставить слитным либо разбить в (к, с)). Среди них, очевидно, есть наибольшее – то, где все подслова кс оставлены слитно.

Пример 3. Пусть $\mathcal{X} = \{\text{к}, \text{с}, \text{т}\} \subset \mathcal{R},\ \mathcal{Y} = \mathcal{A}$. Тогда $\Gamma_1 = \{\text{к} \longrightarrow \longrightarrow \text{k},\ \text{с} \longrightarrow \text{s}\}$ и $\Gamma_2 = \{\text{к} \longrightarrow \text{k},\ \text{с} \longrightarrow \text{s},\ \text{т} \longrightarrow \text{t},\ \text{кс} \longrightarrow \text{x},\ \text{ст} \longrightarrow \text{sht}\}$ не являются корректными системами правил. Множество допустимых разбиений слова т по Γ_1 пусто. Множество допустимых разбиений слова кст по Γ_2 равно $\{(0,1,2,3),\ (0,2,3),\ (0,1,3)\}$ – оно не имеет наибольшего элемента.

Определение 12. Пусть $\omega \in \mathcal{X}^*,\ \Gamma$ – система правил из $\mathcal{X}^*$ в $\mathcal{Y}^*$. Существует наибольшее допустимое разбиение T слова ω по Γ: $T(\omega) = = (\xi_{i_1}, \xi_{i_2}, ..., \xi_{i_n}),\ \ (\xi_{i_1}, \eta_{i_1}), ..., (\xi_{i_n}, \eta_{i_n}) \in \Gamma$. *Образом* ω по системе правил Γ называется слово $\Gamma(\omega) = \eta_{i_1}\eta_{i_2}...\eta_{i_n}$, то есть полученное конкатенацией правых частей правил.

Пример 4. Рассмотрим систему правил Γ из примера 2. $\Gamma(\varepsilon) = \varepsilon$ (единственное допустимое разбиение – (0)), $\Gamma(\text{к}) = \text{k},\ \Gamma(\text{ккссккс}) = \text{kxsxx}$.

Замечание 5. Система правил Γ из $\mathcal{X}^*$ в $\mathcal{Y}^*$ определяет функцию $F : \mathcal{X}^* \longrightarrow \mathcal{Y}^*,\ F(\omega) = \Gamma(\omega)$, притом $F(\varepsilon) = \varepsilon$.

Утверждение 1. *Пусть $F : \mathcal{X}^* \longrightarrow \mathcal{Y}^*$ – произвольная функция из $\mathcal{X}^*$ в $\mathcal{Y}^*$, такая что $F(\varepsilon) = \varepsilon$. Тогда существует система правил Γ, определяющая F.*

Доказательство. $\mathcal{X}^*$ – счетно бесконечное множество, рассмотрим его индексацию $\mathcal{X}^* = \{\omega_i\}_{i \in \mathbf{N}}$, все ω_i различны. Пусть $\Gamma = \{(\omega_i, F(\omega_i))\}_{i \in \mathbf{N}}$. Γ – корректная система правил (для любого непустого $\omega \in \mathcal{X}^*$ найдется наибольшее допустимое разбиение $(0, |\omega|)$); она задает F. □

Утверждение 2. *Пусть Γ – корректная система правил из $\mathcal{X}^*$ в $\mathcal{Y}^*$. Тогда $\forall \widetilde{x} \in \mathcal{X}\ \exists \eta \in \mathcal{Y}^* :\ (\widetilde{x}, \eta) \in \Gamma$.*

Доказательство. Рассмотрим слово $\omega = \widetilde{x}$ (из одной буквы). Существует наибольшее допустимое разбиение ω относительно Γ, но есть всего одно разбиение ω – $T = (0, 1)$. Значит, оно допустимо $\Longrightarrow \exists \eta \in \mathcal{Y}^* : (\widetilde{x}, \eta) \in \Gamma$. $\square$

Теорема 1. *Пусть* $\Gamma = \{(\xi_1, \eta_1), (\xi_2, \eta_2), ...\}$ *– конечное или счетно бесконечное множество правил из* $\mathcal{X}^*$ *в* $\mathcal{Y}^*$. *Тогда* Γ *– корректная система правил* $\Longleftrightarrow$ *выполнены условия:*

1) *все левые части правил* ξ_i *различны;*
2) $\forall \widetilde{x} \in \mathcal{X}\ \exists \eta \in \mathcal{Y}^* :\ (\widetilde{x}, \eta) \in \Gamma$*;*
3) $\forall i, j \in \mathbf{N}\ \forall \alpha, \beta, \gamma \in \mathcal{X}^*\ (\alpha, \beta, \gamma \neq \varepsilon,\ \xi_i = \alpha\beta,\ \xi_j = \beta\gamma \Longrightarrow \exists \eta \in \mathcal{Y}^* : (\alpha\beta\gamma, \eta) \in \Gamma)$.

Третий пункт означает, что если некоторые две левые части правил ξ_i *и* ξ_j *«накладываются» концом одной на начало другой, то существует правило с «объемлющей» левой частью.*

Доказательство. $(\Longrightarrow)$ Пункт 1 по определению; пункт 2 по утверждению 2. Установим пункт 3. Пусть $(\xi_i, \eta_i), (\xi_j, \eta_j) \in \Gamma,\ \alpha, \beta, \gamma \neq \varepsilon$ (непустые слова), $\xi_i = \alpha\beta,\ \xi_j = \beta\gamma$. Пусть $\omega = \alpha\beta\gamma$. $T_1 = (0,\ |\alpha| + |\beta|,\ |\alpha| + |\beta| + +1,\ ...,\ |\alpha|+|\beta|+|\gamma|)$ и $T_2 = (0,\ 1,\ ...,\ |\alpha|,\ |\alpha|+|\beta|+|\gamma|)$ есть допустимые разбиения слова $\omega = \alpha\beta\gamma$ относительно Γ, так как в Γ лежат правила $(\alpha\beta, \eta_i),\ (\beta\gamma, \eta_j)$ и, согласно утверждению 2, правила с левыми частями из одной буквы для любой буквы из $\mathcal{X}$. По определению системы правил для ω существует наибольшее допустимое разбиение $T, T > T_1,\ T > T_2$. В терминах множеств $T \subset T_1,\ T \subset T_2 \Longrightarrow T \subset T_1 \cap T_2 = (0, |\alpha|+|\beta|+|\gamma|) \Longrightarrow$ $\Longrightarrow T = (0, |\alpha|+|\beta|+|\gamma|)$. Значит, в Γ существует правило с левой частью $\alpha\beta\gamma$.

$(\Longleftarrow)$ От противного: пусть Γ не является корректной системой правил из $\mathcal{X}^*$ в $\mathcal{Y}^*$ – значит, $\exists \omega \in \mathcal{X}^*$, для которого не существует наибольшего допустимого разбиения. Пусть $\mathcal{T}$ – множество всех допустимых разбиений слова ω. Оно не пусто (из пункта 1 следует, что $(0, 1, ..., |\omega|) \in \mathcal{T}$), оно конечно, в нем нет наибольшего элемента $\Longrightarrow$, как несложно установить, $\exists T, U \in \mathcal{T} :\ T \not> U,\ U \not> T$ и $\forall V \in \mathcal{T}$ $(V \not> T$ и $V \not> U)$. Элементы T, U максимальны в $\mathcal{T}$ и не сравнимы между собой. Пусть $T = (t_0, t_1, ..., t_n),\ U = (u_0, u_1, ..., u_m)$. Пусть индекс k такой, что $\forall i \in$ $\in \overline{0, k}\ (t_i = u_i)$, но $t_{k+1} \neq u_{k+1}$. Без ограничения общности $t_{k+1} < u_{k+1}$. Пусть индекс p такой, что $\forall i \in \overline{0, p-1}\ (t_i < u_{k+1})$, но $t_p \geq u_{k+1}$. Очевидно, $p > k + 1$. Если $t_p = u_{k+1}$, то разбиение $(t_0, t_1, ...,\ t_k(= u_k),\ t_p$ $(= u_{k+1}),\ ..., t_n)$ допустимо и к тому же больше T – противоречие с выбором T. Значит, $t_p > u_{k+1}$. $\omega = \widetilde{x}_1\widetilde{x}_2...\widetilde{x}_{|\omega|},\ \widetilde{x}_j \in \mathcal{X}$. $\alpha = \widetilde{x}_{t_k+1}...\widetilde{x}_{t_{p-1}};\ \beta =$ $= \widetilde{x}_{t_{p-1}+1}...\widetilde{x}_{u_{k+1}};\ \gamma = \widetilde{x}_{u_{k+1}+1}...\widetilde{x}_{t_p}$. $\alpha\beta$ – часть номер $k+1$ при разбиении U; $\beta\gamma$ – часть номер p при разбиении T. По пункту 2 существует правило

в левой частью $\alpha\beta\gamma$, следовательно, разбиение $(t_0, t_1, ..., t_k(= u_k), t_p, ..., t_n)$ допустимо. Оно больше T – противоречие с выбором T. $\square$

Замечание 6. Теорема описывает системы правил куда более наглядно, чем исходное определение. С ее помощью по явно записанному множеству правил можно легко сказать, является оно корректной системой правил или нет: достаточно проверить три условия на слова ξ_i.

4 Алгоритм получения образа по системе правил

Замысловатое определение корректной системы правил имело целью дать свободу при переводе слова над одним алфавитом в слово над другим.

Определение 13. Пусть ω – слово над алфавитом $\mathcal{X}$. *Отрезком* слова ω назовем пару $(a, b),\ a, b \in \mathbb{N}_0,\ a \leq b \leq |\omega|$, которую для удобства будем записывать в виде $[a, b]$. Пусть $\omega = \widetilde{x}_1\widetilde{x}_2...\widetilde{x}_n,\ \widetilde{x}_i \in \mathcal{X}$. Подсловом отрезка $[a, b]$ назовем $\omega[a, b] = \widetilde{x}_{a+1}\widetilde{x}_{a+2}...\widetilde{x}_b$. В частности, $\omega[a, a] = \varepsilon$.

Определение 14. Пусть $[a, b]$ и $[c, d]$ – отрезки слова ω. Будем говорить, что $[a, b]$ *больше* $[c, d]$, и писать $[a, b] > [c, d]$, если $a \leq c,\ d \leq b$, причем $a < c$ или $d < b$ (то есть $[a, b] \neq [c, d]$).

Замечание 7. Как и в случае с отношением $>$ на множестве разбиений слова ω, определенное выше отношение порядка не является линейным, то есть в общем случае существуют несравнимые отрезки. Аналогично определяются *наименьший* и *наибольший*, *максимальный* и *минимальный* элементы произвольного множества отрезков слова ω.

Определение 15. Пусть $[a, b]$ – отрезок слова ω; $T = (t_0, t_1, ..., t_n)$ – разбиение ω. Будем говорить, что $[a, b]$ входит в T, если $\exists i \in \mathbb{N} :\ t_{i-1} = = a,\ t_i = b$.

Замечание 8. $T(\omega) = (\omega[t_0, t_1],\ \omega[t_1, t_2],\ ...,\ \omega[t_{n-1}, t_n])$.

Утверждение 3. *Пусть* $\Gamma = \{(\xi_1, \eta_1), (\xi_2, \eta_2), ...\}$ *– система правил из* $\mathcal{X}^*$ *в* $\mathcal{Y}^*$*;* ω *– слово над* $\mathcal{X}$*. Пусть* $\exists i, c, d \in \mathbb{N} :\ \omega[c, d] = \xi_i$*. Тогда в наибольшее разбиение* T *слова* ω *относительно* Γ *входит некоторый отрезок* $[a, b]$*, такой что* $[a, b] \geq [c, d]$*.*

Доказательство. По утверждению 2 $\forall \widetilde{x} \in \mathcal{X}\ \exists \eta \in \mathcal{Y}^* :\ (\widetilde{x}, \eta) \in \Gamma$. Следовательно, разбиение $U = (0, 1, 2, ...,\ c,\ d,\ d+1,\ ..., |\omega|)$ слова ω допустимо относительно Γ. $T > U$; в терминах множеств $T \subset U \Longrightarrow \forall t \in$ $\in \mathbb{N}\ (c < t < d \Longrightarrow t \notin T) \Longrightarrow \exists i \in \mathbb{N} :\ t_{i-1} \leq c,\ t_i \geq d$. Отрезок $[t_{i-1}, t_i]$ входит в T, $[t_{i-1}, t_i] \geq [c, d]$. □

Утверждение 4. *Пусть* $\Gamma = \{(\xi_1, \eta_1), (\xi_2, \eta_2), ...\}$ *– система правил из* $\mathcal{X}^*$ *в* $\mathcal{Y}^*$*;* ω *– слово над* $\mathcal{X}$*. Пусть* $\mathcal{S}$ *– множество всех отрезков* $[c, d]$ *слова* ω*, таких что* $\exists i \in \mathbb{N} :\ \omega[c, d] = \xi_i$*. Пусть* $[a, b]$ *– максимальный элемент* $\mathcal{S}$*. Тогда* $[a, b]$ *входит в наибольшее разбиение* T *слова* ω *относительно* Γ*.*

Доказательство. По утверждению 3 существует отрезок $[a', b']$ слова ω, входящий в T, такой что $[a', b'] \geq [a, b]$. Но $[a', b'] \in \mathcal{S}$, а в $\mathcal{S}$ не существует элемента, большего $[a, b], \Longrightarrow [a', b'] = [a, b]$, $[a, b]$ входит в T. □

Утверждение 5. *Пусть в условиях утверждения 4* $[a, b]$ *– максимальный элемент* $\mathcal{S}$*,* $(\xi_i, \eta_i) \in \Gamma,\ \omega[a, b] = \xi_i$*. Обозначим* $\alpha = \omega[0, a],\ \beta =$ $= \omega[b, |\omega|]$*;* $\omega = \alpha\, \omega[a, b]\, \beta$*. Тогда* $\Gamma(\omega) = \Gamma(\alpha)\, \eta_i\, \Gamma(\beta)$*.*

Доказательство. Пусть $T = (t_0, t_1, ..., t_n)$ – наибольшее допустимое разбиение ω относительно Γ, $T(\omega) = (\xi_{j_1}, \xi_{j_2}, ..., \xi_{j_n})$. По утверждению 4 $[a, b]$ входит в Γ, то есть $\exists k \in \mathbb{N} :\ a = t_{k-1},\ b = t_k,\ \xi_{j_k} = \xi_i,\ j_k = i$. Заметим, что $U = (t_0, t_1, ..., t_{k-1})$ и $V = (0,\ t_{k+1} - t_k,\ ...,\ t_n - t_k)$ – это наибольшие допустимые разбиения для α и β (иначе можно было бы построить разбиение для ω, большее T). Имеем $\Gamma(\omega) = \eta_{j_1} ... \eta_{j_{k-1}} \eta_i \eta_{j_{k+1}} ... \eta_{j_n},\ \Gamma(\alpha) =$ $= \eta_{j_1} ... \eta_{j_{k-1}},\ \Gamma(\beta) = \eta_{j_{k+1}} ... \eta_{j_n}$. □

Алгоритм 1 (перевода по системе правил)**.** Пусть Γ – система правил из языка $\mathcal{X}^*$ в язык $\mathcal{Y}^*$; $\omega \in \mathcal{X}^*$. Получим $\Gamma(\omega)$.

1. Если $\omega = \varepsilon$ (пустое), то результат найден: $\Gamma(\omega) = \varepsilon$.
2. $\mathcal{S}$ – множество отрезков $[c, d]$ слова ω, таких что $\exists i \in \mathbb{N} :\ \omega[c, d] =$ $= \xi_i,\ (\xi_i, \eta_i) \in \Gamma$. Возьмем $[a, b]$ – максимальный элемент $\mathcal{S}$.
3. Рекурсивно переведем по системе правил Γ слова $\alpha = \omega[0, a]$ и $\beta =$ $\omega[b, |\omega|]$.
4. Получим ответ конкатенацией: $\Gamma(\omega) = \Gamma(\alpha)\, \eta_i\, \Gamma(\beta)$.

В шаге 2 максимальный элемент $[a, b]$ найдется, так как $\mathcal{S}$ непустое и конечное. Алгоритм верен в силу утверждения 5.

Замечание 9. Алгоритм 1 приведен в общем виде: не определен способ выбора $[a, b]$; максимальных отрезков может быть несколько. Можно, например, рассматривать в множестве $\mathcal{S}$ только отрезки вида $[0, d]$, тогда $a = 0,\ \alpha = \varepsilon,\ b = \max\limits_{[0,d] \in \mathcal{S}} d$. Такой выбор однозначен; рекурсивный перевод шага 3 в этом случае соответствует движению по слову ω слева направо.

5 Системы правил и биективные отображения

Утверждение 6 (общеизвестное)**.** *Пусть A, B – произвольные множества. Функция $F : A \longrightarrow B$ биективна $\Longleftrightarrow \exists F^{-1} : B \longrightarrow A$, такая что $\forall a \in A\ (F^{-1}(F(a)) = a)$ и $\forall b \in B\ (F(F^{-1}(b)) = b)$. В этом случае F^{-1} называется обратной функцией.*

Определение 16. Пусть $\Gamma = \{(\xi_1, \eta_1), (\xi_2, \eta_2), ...\}$ – конечное или счетно бесконечное множество правил из $\mathcal{X}^*$ в $\mathcal{Y}^*$. *Обратным множеством правил* назовем $\Gamma^{-1} = \{(\eta_1, \xi_1), (\eta_2, \xi_2), ...\}$ правил из $\mathcal{Y}^*$ в $\mathcal{X}^*$.

Хотелось бы описать биекцию $F : \mathcal{X}^* \longrightarrow \mathcal{Y}^*$ системой правил Γ, так чтобы Γ^{-1} было корректной системой правил и задавало F^{-1}. На пути к этому есть проблемы.

Пример 5. Рассмотрим систему правил Γ из примера 2. Γ^{-1} не является системой правил из $\mathcal{Y}^* = \mathcal{A}^*$ в $\mathcal{X}^* = \{\text{к}, \text{с}\}^*$.

Пример 6. Рассмотрим Γ из примера 2, но заменим $\mathcal{Y}$ на $\{\text{k}, \text{s}, \text{x}\}$. Теперь $\Gamma^{-1} = \{\text{k} \longrightarrow \text{к},\ \text{s} \longrightarrow \text{с},\ \text{x} \longrightarrow \text{кс}\}$ – корректная система правил. Однако $F(\xi) = \Gamma(\xi),\ \xi \in \mathcal{X}^*$, не биекция (инъекция, но не сюръекция: например, нельзя получить слово ks). Значит, $G(\eta) = \Gamma^{-1}(\eta),\ \eta \in \mathcal{Y}^*$, не может быть обратной к F.

Пример 7. Рассмотрим случай, когда система правил Γ определяет биективную функцию $F : \mathcal{X}^* \longrightarrow \mathcal{Y}^*$, но Γ^{-1} не является корректной системой правил. Пусть $\mathcal{X} = \{0, 1\},\ \mathcal{Y} = \{\text{у}\}$. Зададим вспомогательную функцию G так: $\omega \in \mathcal{X}^*$, тогда 1ω – двоичная запись некоторого числа $n \in \mathbf{N}$; пусть $G(\omega)$ будет словом из $n - 1$ буквы у. $G : \mathcal{X}^* \longrightarrow \mathcal{Y}^*$ – биекция. Пусть $F(01) = G(00) = \text{ууу},\ F(00) = G(01) = \text{уууу}$, а на всех остальных аргументах F совпадает с G. Таким образом, F биекция. Ее определяет система правил $\Gamma = \{0 \longrightarrow \text{у},\ 1 \longrightarrow \text{уу},\ 00 \longrightarrow \text{уууу},\ 10 \longrightarrow \longrightarrow \text{ууууу},\ 11 \longrightarrow \text{ууууууу}\} \cup \{(\omega \longrightarrow F(\omega)) : |\omega| \geq 3\}$ (содержит все ω, кроме 01, образ которой есть конкатенация правых частей правил для 0 и 1). Но легко видеть, что Γ^{-1} при этом не является системой правил.

Теорема 2. *Пусть Γ – система правил из $\mathcal{X}^*$ в $\mathcal{Y}^*$, причем Γ^{-1} – система правил из $\mathcal{Y}^*$ в $\mathcal{X}^*$. Пусть $F(\xi) = \Gamma(\xi),\ \xi \in \mathcal{X}^*$. Тогда F биекция $\Longleftrightarrow$ Γ^{-1} задает обратную функцию F^{-1}.*

Доказательство. $(\Longleftarrow)$ Очевидно по утверждению 6.
$(\Longrightarrow)$ От противного: пусть $\exists \eta \in \mathcal{Y}^* :\ F^{-1}(\eta) \neq \Gamma^{-1}(\eta)$. Обозначим

$\xi = F^{-1}(\eta),\ \omega = \Gamma^{-1}(\eta)$. $F(\xi) = \Gamma(\xi) = \eta$: по определению системы правил среди допустимых разбиений слова ξ относительно Γ существует наибольшее $T, \exists i_1, i_2, ..., i_n \in \mathbb{N} :\ T(\xi) = (\xi_{i_1}, \xi_{i_2}, ..., \xi_{i_n}),\ (\xi_{i_1}, \eta_{i_1}), ..., (\xi_{i_n}, \eta_{i_n}) \in \in \Gamma;\ \eta = \eta_{i_1}\eta_{i_n}...\eta_{i_n}$. Отсюда получаем допустимое разбиение U слова η относительно Γ^{-1}: $U = (0, |\eta_{i_1}|, |\eta_{i_1}| + |\eta_{i_2}|, ..., |\eta|);\ U(\eta) = (\eta_{i_1}, \eta_{i_2}, ..., \eta_{i_n})$. Существует наибольшее допустимое разбиение V слова η относительно Γ^{-1}. $V \neq U$, так как в таком случае было бы $\Gamma^{-1}(\eta) = \xi$. Значит, $V > U$. Пусть $U = (u_0, u_1, ..., u_n),\ V = (v_0, v_1, ..., v_m)$. В терминах множеств $V \subset U \Longrightarrow \exists j_1, j_2, k \in \mathbb{N}_0 :\ u_{j_1} = v_{k-1},\ u_{j_2} = v_k,\ j_2 > j_1 + 1$, то есть в V существуют два подряд идущих элемента v_{k-1}, v_k, равные двум элементам u_{j_1}, u_{j_2} из U, не идущим подряд. $\eta = \widetilde{y}_1\widetilde{y}_2...\widetilde{y}_{|\eta|},\ \widetilde{y}_j \in \mathcal{Y}$; рассмотрим $\beta = \widetilde{y}_{u_{j_1}+1}\widetilde{y}_{u_{j_1}+2}...\widetilde{y}_{u_{j_2}}$. $\xi = \widetilde{x}_1\widetilde{x}_2...\widetilde{x}_{|\xi|},\ \widetilde{x}_j \in \mathcal{X};\ T = (t_1, t_2, ..., t_n)$; рассмотрим $\alpha = \widetilde{x}_{t_{j_1}+1}\widetilde{x}_{t_{j_1}+2}...\widetilde{x}_{t_{j_2}}$. $T' = (0, t_{j_1+1} - t_{j_1}, t_{j_1+2} - t_{j_1}, ..., t_{j_2} - t_{j_1})$ есть разбиение слова α, допустимое относительно Γ. Более того, оно наибольшее среди допустимых для α, так как в противном случае можно было бы построить допустимое разбиение слова ξ, большее T. $T'(\alpha) = (\xi_{i_{t_1}}, \xi_{i_{t_1+1}}, ..., \xi_{i_{t_2}}) \Longrightarrow \Gamma(\alpha) = \eta_{i_{t_1}}\eta_{i_{t_1+1}}...\eta_{t_2} = \beta$. Притом β – это часть номер k в разбиении V слова $\eta \Longrightarrow \exists\gamma \in \mathcal{X}^* :\ \ (\beta, \gamma) \in \Gamma^{-1}$. Значит, $(\gamma, \beta) \in \Gamma,\ F(\gamma) = \Gamma(\gamma) = \beta$. $\alpha \neq \gamma$, так как в таком случае для α существовало бы допустимое разбиение $(0, |\alpha|)$, а по построению α в наибольшем допустимом разбиении T' хотя бы три числа $(j_2 > j_1 + 1)$. Получается, что $\alpha \neq \gamma,\ F(\alpha) = F(\gamma) = \beta$ – противоречие с биективностью F. □

6 Построение биективной англо-русской транслитерации

Необходимо построить биективную функцию $F : \mathcal{R}^* \longrightarrow \mathcal{A}^*$. Опишем ее системой правил Γ, такой что Γ^{-1} тоже система правил. Тогда по теореме 2 Γ^{-1} задаст обратную функцию F^{-1}.

Сократим рассматриваемые алфавиты: перейдем к $\mathcal{R}' \subset \mathcal{R},\ \mathcal{A}' \subset \mathcal{A}$. Будем шаг за шагом добавлять новые буквы и пополнять Γ, следя за тем, чтобы на рассматриваемых алфавитах Γ и Γ^{-1} были системами правил и чтобы функция, определяемая Γ, обладала инъективностью (никакие $\omega_1 \neq \omega_2 \in \mathcal{R}'^*$ не дают один и тот же образ) и сюръективнстью (любое $\eta \in \mathcal{A}'^*$ имеет прообраз). $\mathcal{R}'$ и $\mathcal{A}'$ по ходу рассуждений указывать явно не станем.

Выбор образов для тех или иных букв не имеет строгого логического обоснования. При их выборе автор руководствовался двумя принци-

пами. Во-первых, система правил должна быть простой, насколько это возможно, для удобства практического применения. Во-вторых, автор следовал собственной лингвистической интуиции, которую способен объяснить словами лишь отчасти.

Так как мы строим систему правил Г, такую что Γ^{-1} тоже система правил, то для наглядности в записи правил вместо $\longrightarrow$ будем использовать $\longleftrightarrow$.

Шаг 1 (очевидные правила буква-буква)**.**

а $\longleftrightarrow$ a	и $\longleftrightarrow$ i	о $\longleftrightarrow$ o	ф $\longleftrightarrow$ f
б $\longleftrightarrow$ b	к $\longleftrightarrow$ k	п $\longleftrightarrow$ p	х $\longleftrightarrow$ h
в $\longleftrightarrow$ v	л $\longleftrightarrow$ l	р $\longleftrightarrow$ r	
д $\longleftrightarrow$ d	м $\longleftrightarrow$ m	с $\longleftrightarrow$ s	
з $\longleftrightarrow$ z	н $\longleftrightarrow$ n	т $\longleftrightarrow$ t	

Шаг 2 (правила буква-буква, требующие внимания)**.**

Выбор следующих правил может быть не очевиден.

г $\longleftrightarrow$ g	ж $\longleftrightarrow$ j	й $\longleftrightarrow$ y	ц $\longleftrightarrow$ c

Шаг 3 (ч, ш, щ; бесконечная цепочка правил)**.**

Рассмотрим следующий частный случай. Хочется добавить правило ш $\longleftrightarrow$ sh. Но в этом случае Г перестанет быть биекцией: Γ(ш) = Γ(сх) = = sh. Мы только что «забрали» образ у любого русского слова, содержащего сх как подслово – придется добавить для сх новое правило. Пусть это будет сх $\longleftrightarrow$ skh. Возникает та же проблема: Γ(сх) = Γ(скх) = skh. Ее можно снова решить тем же образом и так далее до бесконечности.

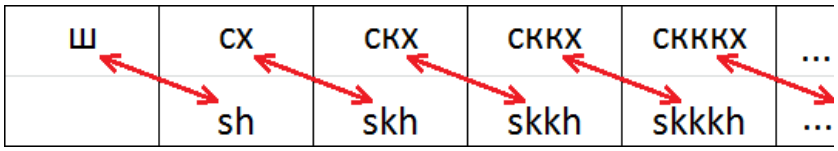

Рис. 1: цепочка исправлений для ш

Если проделать любое конечное число описанных исправлений, проблема, очевидно, останется. Но если переопределить образы для всей счетно бесконечной цепочки слов (рис. 1), она разрешится! Переопределение достигается добавлением счетно бесконечной цепочки правил: ш $\longleftrightarrow$ sh, сх $\longleftrightarrow$ skh, скх $\longleftrightarrow$ skkh, сккх $\longleftrightarrow$ skkkh, ...

Для лаконичного описания подобных цепочек введем следующий способ записи.

Определение 17. Введем символы $+$ и $*$. Пусть $\mathcal{X}$ и $\mathcal{Y}$ – алфавиты; $n \in$ $\in \mathbb{N}$; $\alpha_0, \alpha_1, ...\alpha_n \in \mathcal{X}^*$, $\beta_0, \beta_1, ..., \beta_n \in \mathcal{Y}^*$ – некоторые слова; $\widetilde{x}_1, \widetilde{x}_2, ..., \widetilde{x}_n \in$ $\in \mathcal{X}$; $\widetilde{y}_1, \widetilde{y}_2, ..., \widetilde{y}_n \in \mathcal{Y}$ – некоторые буквы; $a_1, a_2, ..., a_n, b_1, b_2, ..., b_n \in \{+, *\}$. Выражение $\alpha_0\widetilde{x}_1^{a_1}\alpha_1\widetilde{x}_2^{a_2}...\widetilde{x}_n^{a_n}\alpha_n \longleftrightarrow \beta_0\widetilde{y}_1^{b_1}\beta_1\widetilde{y}_2^{b_2}...\widetilde{y}_n^{b_n}\beta_n$ обозначает множество тех и только тех правил, которые имеют вид

$$\alpha_0\widetilde{x}_1...\widetilde{x}_1(i_1 \text{ раз})\alpha_1\widetilde{x}_2...\widetilde{x}_2(i_2 \text{ раз})...\widetilde{x}_n...\widetilde{x}_n(i_n \text{ раз})\alpha_n \longleftrightarrow$$
$$\longleftrightarrow \beta_0\widetilde{y}_1...\widetilde{y}_1(j_1 \text{ раз})\beta_1\widetilde{y}_2...\widetilde{y}_2(j_2 \text{ раз})...\widetilde{y}_n...\widetilde{y}_n(j_n \text{ раз})\beta_n,$$

где числа i_k, j_k соотносятся внутри каждой пары следующим образом:

1) если $a_k = *$ и $b_k = *$, то $0 \leq i_k = j_k$;
2) если $a_k = *$ и $b_k = +$, то $0 \leq i_k = j_k - 1$;
3) если $a_k = +$ и $b_k = *$, то $0 \leq i_k - 1 = j_k$;
4) если $a_k = +$ и $b_k = +$, то $0 \leq i_k - 1 = j_k - 1$.

Описанный способ записи будем называть *(*,+)-нотацией.*

Используя (*,+)-нотацию, можно записать цепочку правил для ш как ш $\longleftrightarrow$ sh, ск*х $\longleftrightarrow$ sk$^+$h. Средние символы (из обоих алфавитов) в таких цепочках будем в дальнейшем называть *разделителями.*

Аналогичную цепочку правил построим для ч: ч $\longleftrightarrow$ ch, цк*х $\longleftrightarrow$ $\longleftrightarrow$ ck$^+$h.

Для щ хочется добавить правило щ $\longleftrightarrow$ shch. Тогда, аналогично рассмотренному ранее случаю, образ «потеряют» все русские слова, содержащие шч как подслово (обратим внимание, что именно шч, а не, например, схцх, ведь мы уже ввели правила для ш и ч). Проблему решим по известному образцу, используя в качестве разделителя букву т: шт*ч $\longleftrightarrow$ sht$^+$ch.

Итак, были добавлены следующие правила:

ч $\longleftrightarrow$ ch	ш $\longleftrightarrow$ sh	щ $\longleftrightarrow$ shch
цк*х $\longleftrightarrow$ ck$^+$h	ск*х $\longleftrightarrow$ sk$^+$h	шт*ч $\longleftrightarrow$ sht$^+$ch

Шаг 4 (th и ph)**.**

Пока что Γ^{-1}(th) = тх, что плохо отражает суть этого английского буквосочетания. Автору кажется хорошим решением ввести правило тф $\longleftrightarrow$ th.

Заметим, что теперь Γ(тх) = Γ(тф) = th, а в tf не переходит никакое русское слово: образ тх «передали» тф. Самым простым решением будет «отдать» тх старый образ тф, а именно добавить тх $\longleftrightarrow$ tf.

Неудачен также перевод Γ^{-1}(ph) = пх. Гораздо лучше Γ^{-1}(ph) = пф. Поступим аналогично!

Итак, добавлены следующие правила:

тф $\longleftrightarrow$ th тх $\longleftrightarrow$ tf пф $\longleftrightarrow$ ph пх $\longleftrightarrow$ pf

Факт того, что решения двух, казалось бы, совершенно не связанных проблем образуют стройную систему, несомненно, свидетельствует в пользу избранной автором транслитерации.

Шаг 5 (у, э и йотированные гласные)**.**

Добавим правила у $\longleftrightarrow$ ou, ю $\longleftrightarrow$ u. Такое сопоставление автору подсказывает лингвистическая интуиция. Перевод u в ю во многих английских словах смотрится гармонично, а английская буква u даже называется «ю». Записывать у как «оу» вообще в духе и русского, и английского языка: в древнерусском у записывалась диграфом оу, в английском буквосочетание ou произносится как «у».

Введем правила э $\longleftrightarrow$ oe, е $\longleftrightarrow$ e. Соответствие е $\longleftrightarrow$ e крайне удобно, а правило для э сконструируем аналогично правилу для у.

Для ё и я добавим правила ё $\longleftrightarrow$ yo, я $\longleftrightarrow$ ya.

Как и на шаге 3, чтобы сохранить биективность Г, нужно добавить бесконечные цепочки правил с символами-разделителями. Для у и э разделителем пусть будет й, для ё и я – и. В результате имеем:

ю $\longleftrightarrow$ u е $\longleftrightarrow$ e

у $\longleftrightarrow$ ou	э $\longleftrightarrow$ oe	ё $\longleftrightarrow$ yo	я $\longleftrightarrow$ ya
ой*ю $\longleftrightarrow$ оу$^+$u	ой*е $\longleftrightarrow$ оу$^+$e	йи*о $\longleftrightarrow$ yi$^+$o	йи*а $\longleftrightarrow$ yi$^+$a

Замечание 10. На текущий момент Г не является системой правил: например, есть правила с левыми частями йо и ою, но нет – с йою (не выполнено условие критерия 1). Будем говорить, что цепочки для ё и у *конфликтуют*. Эту ситуацию разрешим позже.

Шаг 6 (ъ, ы, ь)**.**

Собрав в кулак имеющуюся лингвистическую интуицию и представления о практичности и эстетике, автор сочинил следующие правила: ъ $\longleftrightarrow$ oa, ы $\longleftrightarrow$ ea, ь $\longleftrightarrow$ ie.

Добавив цепочки с символами-разделителями й и х (перед а), получаем:

ъ $\longleftrightarrow$ oa	ы $\longleftrightarrow$ ea	ь $\longleftrightarrow$ ie
ох*а $\longleftrightarrow$ oh$^+$a	ех*а $\longleftrightarrow$ eh$^+$a	ий*е $\longleftrightarrow$ iy$^+$e

Шаг 7 (разрешение конфликтов гласных: ё и у)**.**

Рассмотрим бесконечные цепочки правил ё $\longleftrightarrow$ yo, йи*о $\longleftrightarrow$ yi$^+$o и у $\longleftrightarrow$ ou, ой*ю $\longleftrightarrow$ оу$^+$u.

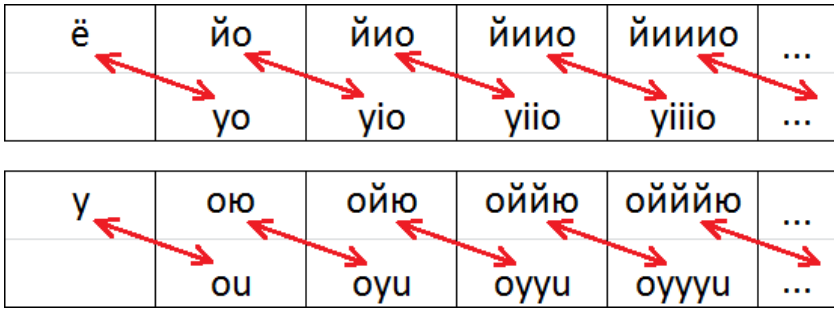

Рис. 2: цепочки правил для ё и у

Как видно на рис. 2, цепочки правил получаются при «сдвиге» соответствий слов на один относительно порожденных побуквенно. В начало добавляется правило для нового, ранее не встречавшегося, символа. Этот символ будем называть *головой* цепочки.

Если взять любое (кроме головы) слово из первой цепочки и любое (кроме головы) слово на том же языке из второй цепочки, то их можно «наложить» друг на друга одним символом (например, йио и ою, уо и ou). Значит, ни Г, ни Γ^{-1} не станут системами правил, пока не будут добавлены правила для всех «объемлющих» слов (таких как, например, йиою и you). Это и есть *конфликт*. В рассматриваемом случае обе головы русские, принадлежат к одному языку – возникающий конфликт назовем *односторонним*.

Итак, для всех слов, в которые слова из цепочек входят как подслова «без наложений», образы уже корректно определены. Выпишем все возможные вхождения «с наложениями» в бесконечную таблицу (рис. 3). Номер клетки по горизонтали – количество первого по порядку символа-разделителя, по вертикали – второго.

	йу	йиу	...
ёю	йою	йиою	...
	you	yiou	
ёйю	йойю	йиойю	...
	yoyu	yioyu	
⋮	⋮	⋮	

Рис. 3: таблица конфликтующих «наложений»

Рис. 4: разрешение конфликта

Русские слова заполняют всю таблицу, кроме верхней левой клетки, английские – кроме первого ряда и первого столбца. Нам нужно сгруппировать все слова по парам. Поступим следующим образом: сдвинем верхний ряд на одну клетку влево и каждому русскому слову поставим в пару английское, стоящее от него снизу справа (рис. 4). Все полученные правила добавим в Г. Таким образом, вся счетная бесконечность конфликтующих наложений разрешается.

Естественная структура множества добавленных правил не одномерная (цепочка), а двумерная. Его можно представить в (*,+)-нотации, однако такая запись мало помогает пониманию:

$$\text{йи}^*\text{у} \longleftrightarrow \text{yi}^*\text{ou}$$
$$\text{ёй}^*\text{ю} \longleftrightarrow \text{yoy}^+\text{u}$$
$$\text{йи}^*\text{ой}^*\text{ю} \longleftrightarrow \text{yi}^+\text{oy}^+\text{u}$$

Гораздо легче осмыслить его в виде следующей закономерности: в описанной ситуации перед использованием правила с головным символом разделитель не требуется. Примеры:

1) Г(йу) = you (что не может не радовать), Г(йиу) = yiou, Г(йииуу) = = yiiiou: использовано правило с головой у, перед ним разделитель не требуется, а именно, букв и столько же, сколько i;
2) Г(ёю) = yoyu, Г(ёйю) = yoyyu, Г(ёййю) = yoyyyyu: использовано правило с головой ё, но после разделитель требуется, а именно, букв у на одну больше, чем й;
3) Г(йою) = yioyu, Г(йиоййю) = yiioyyyu – правил с головным символом не используется, разделители требуются.

К этой закономерности мы еще вернемся.

Шаг 8 (разрешение конфликтов гласных: двойные конфликты)**.**

Только что был разрешен односторонний конфликт цепочек для букв ё и у. После добавления правил для гласных, ъ и ь возникло пять подобых конфликтов: ё-у, ё-ъ, ё-э, э-ы, ь-ы. Наглядно их можно представить на графе (рис. 5), где вершины – английские гласные, а направленные ребра – двухбуквенные образы русских гласных, ъ и ь.

Конфликту двух букв соответствует путь длиной в два ребра, их в графе как раз пять штук. Легко видеть, что в графе к тому же есть один путь ё-э-ы длиной в три ребра: он соответствует тройному одностороннему конфликту. Никаких более путей (как и циклов) в графе гласных нет.

Оставшиеся четыре двойных односторонних конфликта разрешим полностью аналогично конфликту ё-у, рассмотренному на предыдущем

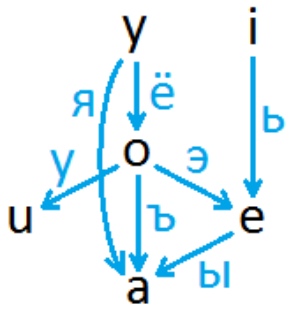

Рис. 5: граф гласных

шаге. Будет верен тот же принцип: перед использованием правила с головным символом разделитель не требуется. Результат можно описать и в (*,+)-нотации:

$$\text{йи}^*\text{ъ} \longleftrightarrow \text{yi}^*\text{oa} \qquad \text{ой}^*\text{ы} \longleftrightarrow \text{oy}^*\text{ea}$$
$$\text{ёх}^*\text{а} \longleftrightarrow \text{yoh}^+\text{a} \qquad \text{эх}^*\text{а} \longleftrightarrow \text{oeh}^+\text{a}$$
$$\text{йи}^*\text{ох}^*\text{а} \longleftrightarrow \text{yi}^+\text{oh}^+\text{a} \qquad \text{ой}^*\text{ех}^*\text{а} \longleftrightarrow \text{oy}^+\text{eh}^+\text{a}$$

$$\text{йи}^*\text{э} \longleftrightarrow \text{yi}^*\text{oe} \qquad \text{ий}^*\text{ы} \longleftrightarrow \text{iy}^*\text{ea}$$
$$\text{ёй}^*\text{е} \longleftrightarrow \text{yoy}^+\text{e} \qquad \text{ьх}^*\text{а} \longleftrightarrow \text{ieh}^+\text{a}$$
$$\text{йи}^*\text{ой}^*\text{е} \longleftrightarrow \text{yi}^+\text{oy}^+\text{e} \qquad \text{ий}^*\text{ех}^*\text{а} \longleftrightarrow \text{iy}^+\text{eh}^+\text{a}$$

Шаг 9 (разрешение конфликтов гласных: тройной конфликт)**.**

Осталось разрешить односторонний конфликт ё-э-ы. Как и раньше, построим бесконечную таблицу конфликтующих «наложений», но она будет уже не двумерной, а трехмерной (приведены первые три слоя – рис. 6, 7, 8). Номер клетки по горизонтали – количество первого символа-разделителя, по вертикали – второго, в глубину – третьего.

Вид слоев, начиная со второго, повторяет вид двумерной таблицы для двойного конфликта. На первом слое английских слов нет вовсе, русские занимают все клетки, кроме первого ряда. Подобная сложная форма множества рассматриваемых русских слов объясняется тем, что правила с первой и второй, а также со второй и третьей головами не могут быть применены одновременно, а с первой и третьей – могут (первый столбец первого слоя).

Чтобы разрешить конфликт, поступим следующим образом. Весь первый слой сдвинем на клетку вверх; во всех последующих слоях сдвинем верхний ряд на клетку влево. Теперь объединим каждое русское слово в пару с английским, стоящим от него снизу справа на следующем слое (рис. 9, 10, 11). Все полученные правила добавим в Г.

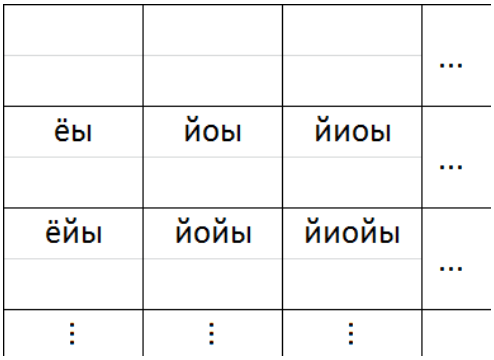

Рис. 6: конфликтующие «наложения», слой 1

Рис. 7: конфликтующие «наложения», слой 2

Рис. 8: конфликтующие «наложения», слой 3

Рис. 9: разрешение конфликта, слой 1

Рис. 10: разрешение конфликта, слой 2

Рис. 11: разрешение конфликта, слой 3

Естественная структура множества добавленных правил трехмерная. Его можно представить в (*,+)-нотации:

$$\text{ёй}^*\text{ы} \longleftrightarrow \text{уоу}^*\text{еа}$$
$$\text{йи}^*\text{ой}^*\text{ы} \longleftrightarrow \text{уі}^+\text{оу}^*\text{еа}$$
$$\text{йи}^*\text{эх}^*\text{а} \longleftrightarrow \text{уі}^*\text{оеh}^+\text{а}$$
$$\text{ёй}^*\text{ех}^*\text{а} \longleftrightarrow \text{уоу}^+\text{еh}^+\text{а}$$
$$\text{йи}^*\text{ой}^*\text{ех}^*\text{а} \longleftrightarrow \text{уі}^+\text{оу}^+\text{еh}^+\text{а}$$

Заметим, что добавленное множество правил описывается практически той же закономерностью, что и разрешение двойного одностороннего конфликта: непосредственно перед использованием правила с головным символом разделитель не требуется. В качестве уточнения добавлено слово непосредственно. Примеры:

1) Г(ёы) = уоеа, Г(ёйы) = уоуеа: одновременно использованы правила с первой (ё) и третьей (ы) головами, разделитель й/у перед третьей не требуется;
2) Г(йоы) = уіоеа, Г(йиоы) = уііоеа, Г(йойы) = уіоуеа, Г(йиойы) = = уііоуеа: использовано правило с головой ы, второй разделитель (й/у, непосредственно перед) не нужен, но первый (и/і) нужен;
3) Г(йэа) = уоеhа, Г(йиэа) = уіоеhа, Г(йэха) = уоеhhа, Г(йиэха) = = уіоеhhа: использовано правило с головой э, первый разделитель не нужен, третий нужен;
4) Г(ёеа) = уоуеhа, Г(ёйеа) = уоууеhа, Г(ёеха) = уоуеhhа, Г(ёйеха) = = уоууеhhа: использовано правило с головой ё, второй и третий разделители нужны;
5) Г(йоеа) = уіоуеhа, Г(йиойеха) = уііоууеhhа: правила с головами не встречаются, все разделители нужны.

Итак, верен следующий закон.

Закон 1. Чтобы выполнить перевод при одностороннем конфликте (ё-у, ё-ъ, ё-э, э-ы, ь-ы либо ё-э-ы), нужно применить обычные правила из конфликтующих цепочек с тем лишь отличием, что непосредственно перед использованием правила с головным символом разделитель не требуется.

Шаг 10 (q)**.**

Для всех русских букв были добавлены правила, содержащие их в качестве левой части. Однако осталось три английских буквы, для которых еще нет правил: q, w и x.

Добавим правило къ $\longleftrightarrow$ q. Как и раньше, для биективности Г необходимо добавить бесконечную цепочку правил. Ситуация симметричная привычной нам: символов-разделителей на один больше не в правых частях, а в левых. Пусть разделителем будет г/g.

$$\text{къ} \longleftrightarrow \text{q}$$
$$\text{кг}^{+}\text{ъ} \longleftrightarrow \text{kg}^{*}\text{oa}$$

Шаг 11 (х, разрешение конфликтов)**.**

Для буквы х добавим следующее правило, вместе с ним – бесконечную цепочку:

$$\text{кс} \longleftrightarrow \text{x}$$
$$\text{кг}^{+}\text{с} \longleftrightarrow \text{kg}^{*}\text{s}$$

Обратим внимание, что между цепочками для х и ш и для х и щ возникли конфликты. Рассмотрим для начала конфликт х-ш (рис. 12).

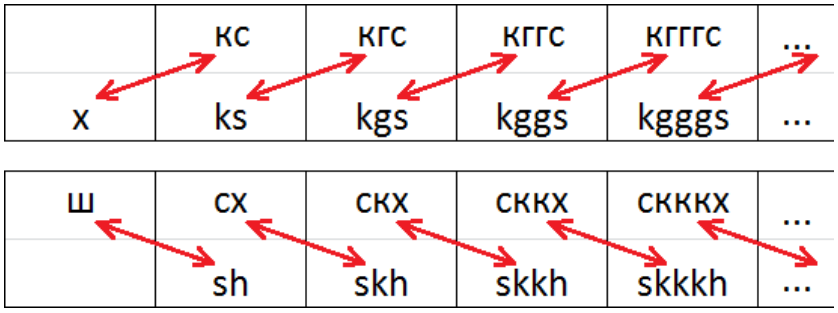

Рис. 12: цепочки правил для х и ш

Ситуация похожа на уже известную нам (например, конфликт ё-у), но теперь один головной символ английский, другой русский. Такой конфликт назовем *двусторонним*.

Как и раньше, выпишем все конфликтующие «наложения» слов из цепочек в бесконечную таблицу (рис. 13); номер клетки по горизонтали – количество первых символов-разделителей, по вертикали – количество вторых. Русские слова заполняют всю таблицу кроме первого столбца, английские – кроме первой строки. Чтобы разрешить конфликт, каждому русскому слову поставим в пару английское, стоящее от него снизу слева (рис. 14). Полученные правила добавим в Г.

Множество добавленных правил можно представить в (*,+)-нотации:

$$\text{кш} \longleftrightarrow \text{xh}$$
$$\text{кг}^{+}\text{ш} \longleftrightarrow \text{kg}^{*}\text{sh}$$
$$\text{кск}^{*}\text{х} \longleftrightarrow \text{xk}^{+}\text{h}$$
$$\text{кг}^{+}\text{ск}^{*}\text{х} \longleftrightarrow \text{kg}^{*}\text{sk}^{+}\text{h}$$

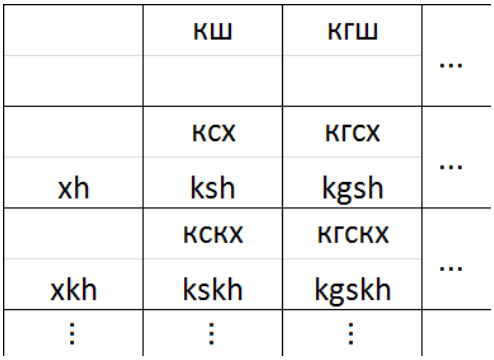

Рис. 13: таблица конфликтующих «наложений»

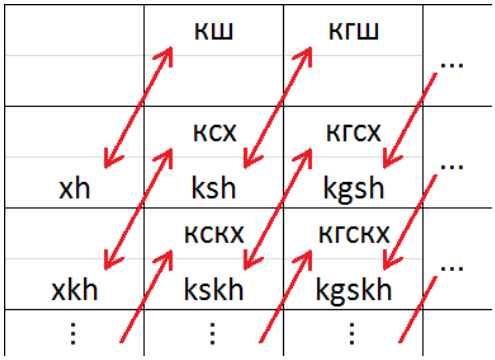

Рис. 14: разрешение конфликта

Проще осмыслить его через следующую закономерность. Заметим, что для перевода любого слова из таблицы нужно сначала провести замену на одно слово вперед по цепочке с головой на исходном языке, затем перевод побуквенно, и, наконец, замену на одно слово назад по цепочке с головой на целевом языке. Примеры:

1) Г(кш) = xh: кш $\longrightarrow$ (замена вперед по цепочке для ш) ксх $\longrightarrow$ $\longrightarrow$ ksh $\longrightarrow$ (замена назад по цепочке для x) xh; обратный перевод: xh $\longrightarrow$ (замена вперед по цепочке для x) ksh $\longrightarrow$ ксх $\longrightarrow$ (замена назад по цепочке для ш) кш;
2) Г(кгш) = ksh: кгш $\longrightarrow$ кгсх $\longrightarrow$ kgsh $\longrightarrow$ ksh; обратно: ksh $\longrightarrow$ $\longrightarrow$ kgsh $\longrightarrow$ кгсх $\longrightarrow$ кгш;
3) Г(ксх) = xkh: ксх $\longrightarrow$ кскх $\longrightarrow$ kskh $\longrightarrow$ xkh; обратно: xkh $\longrightarrow$ $\longrightarrow$ kskh $\longrightarrow$ кскх $\longrightarrow$ ксх;
4) Г(кхсх) = kskh: кхсх $\longrightarrow$ кхскх $\longrightarrow$ khskh $\longrightarrow$ kskh; обратно: kskh $\longrightarrow$ khskh $\longrightarrow$ кхскх $\longrightarrow$ кхсх.

Конфликт цепочек x и щ имеет аналогичный вид (рис. 15). Разрешается он тем же способом, верна та же закономерность.

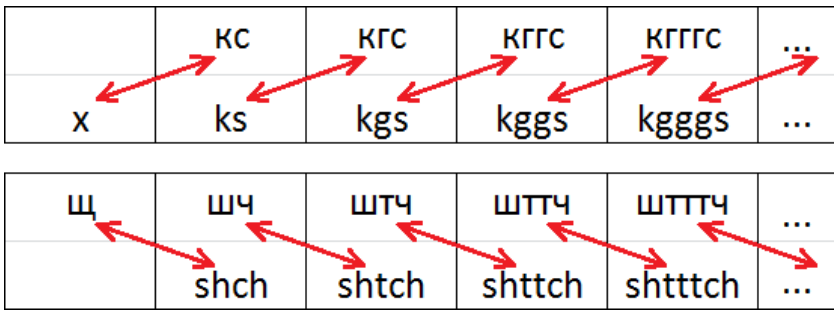

Рис. 15: цепочки правил для x и ш

Добавленные правила можно описать в (*,+)-нотации:

$$\text{кщ} \longleftrightarrow \text{xhsh}$$
$$\text{кг}^{+}\text{щ} \longleftrightarrow \text{kg}^{*}\text{shch}$$
$$\text{кшт}^{*}\text{ч} \longleftrightarrow \text{xht}^{+}\text{ch}$$
$$\text{кг}^{+}\text{шт}^{*}\text{ч} \longleftrightarrow \text{kg}^{*}\text{sht}^{+}\text{ch}$$

Обратим внимание на следующую особенность. При «наложении» русских слов из цепочек «на стыке» стоит ш (в него «сворачивается» sh). Так, например, кс и шч могут образовать конфликтующее «наложение» кшч, хотя на первый взгляд это может показаться противоестественным. Также нужно использовать уже введенные правила для ш и ч при побуквенном переводе. Примеры:

1) Г(кщ) = xhch: кщ $\longrightarrow$ (замена вперед по цепочке для щ) кшч $\longrightarrow$ $\longrightarrow$ kshch $\longrightarrow$ (замена назад по цепочке для х) xhch; обратный перевод: xhch $\longrightarrow$ (замена вперед по цепочке для х) kshch $\longrightarrow$ кшч $\longrightarrow$ $\longrightarrow$ (замена назад по цепочке для ш) кщ;
2) Г(кгщ) = kshch: кгщ $\longrightarrow$ кгшч $\longrightarrow$ kgshch $\longrightarrow$ kshch; обратно: kshch $\longrightarrow$ kgshch $\longrightarrow$ кгшч $\longrightarrow$ кгщ;
3) Г(кшч) = xhtch: кшч $\longrightarrow$ куштч $\longrightarrow$ kshtch $\longrightarrow$ xhtch; обратно: xhtch $\longrightarrow$ kshtch $\longrightarrow$ куштч $\longrightarrow$ кшч;
4) Г(кгшч) = kshtch: кгшч $\longrightarrow$ кгштч $\longrightarrow$ kgshtch $\longrightarrow$ kshtch; обратно: kshtch $\longrightarrow$ kgshtch $\longrightarrow$ кгштч $\longrightarrow$ кгшч.

Итак, верен следующий закон.

Закон 2. Чтобы выполнить перевод при двустороннем конфликте (х-ш либо х-щ), нужно провести замену на одно слово вперед по цепочке с головой на исходном языке, затем перевод побуквенно, и, наконец, замену на одно слово назад по цепочке с головой на целевом языке.

Шаг 12 (w, разрешение конфликта)**.**

Добавим следующее правило, вместе с ним – бесконечную цепочку:

$$\text{ув} \longleftrightarrow \text{w}$$
$$\text{уф}^{+}\text{в} \longleftrightarrow \text{ouf}^{*}\text{v}$$

Возникает конфликт (рис. 16) между этой цепочкой и йи*у $\longleftrightarrow$ yi*ou (правилами верхней строки таблицы на рис. 4).

В цепочке йи*у $\longleftrightarrow$ yi*ou слова в некотором смысле соответствуют побуквенно. Она не является привычной нам цепочкой правил, получающейся сдвигом сопоставлений слов на один относительно побуквенных и прибавлением головы. У конфликта только одна голова – обозначим его йу-w.

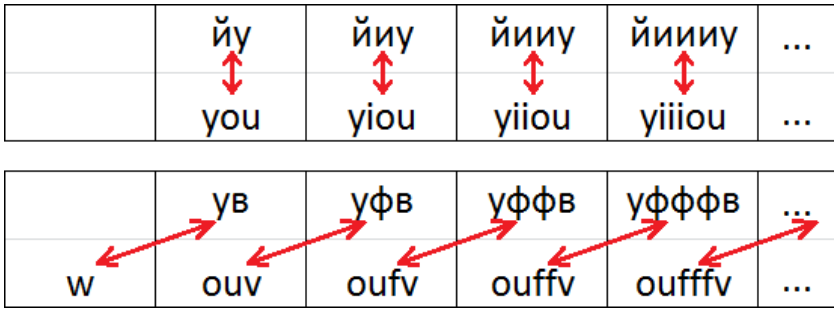

Рис. 16: цепочки правил йу и w

Построим бесконечную таблицу конфликтующих «наложений» слов из цепочек (рис. 17). Русские слова заполняют всю таблицу кроме первого ряда, английские – всю целиком. Чтобы разрешить конфликт, сдвинем все русские слова на строку вверх и поставим в пару слова, стоящие в одной клетке (рис. 18)

yw	yiw	yiiw	...
йув youv	йиув yiouv	йииув yiiouv	...
йуфв youfv	йиуфв yioufv	йииуфв yiioufv	...
⋮	⋮	⋮	

Рис. 17: таблица конфликтующих «наложений»

йув yw	йиув yiw	йииув yiiw	...
йуфв youv	йиуфв yiouv	йииуфв yiiouv	...
йуффв youfv	йиуффв yioufv	йииуффв yiioufv	...
⋮	⋮	⋮	

Рис. 18: разрешение конфликта

В (*,+)-нотации множество добавленных правил описывается так:

$$\text{йи}^*\text{ув} \longleftrightarrow \text{yi}^*\text{w}$$
$$\text{йи}^*\text{уф}^+\text{в} \longleftrightarrow \text{yi}^*\text{ouf}^*\text{v}$$

Как видно, выполнена несложная закономерность: перевод происходит так, будто правила из цепочки йу игнорируются.

Закон 3. При конфликте йу-w перевод происходит без учета правил из цепочки йу.

7 Итоговые формулировки

Итак, было построено следующее множество правил Γ из $\mathcal{R}^*$ в $\mathcal{A}^*$. Для удобства представим его в виде $\Gamma = \Gamma_1 \cup \Gamma_2$: в Γ_1 лежат ключевые правила, в Γ_2 – добавленные для разрешения конфликтов.

Γ_1:

а $\longleftrightarrow$ a
б $\longleftrightarrow$ b
в $\longleftrightarrow$ v
г $\longleftrightarrow$ g
д $\longleftrightarrow$ d
е $\longleftrightarrow$ e

ж $\longleftrightarrow$ j
з $\longleftrightarrow$ z
и $\longleftrightarrow$ i
й $\longleftrightarrow$ y
к $\longleftrightarrow$ k
л $\longleftrightarrow$ l

м $\longleftrightarrow$ m
н $\longleftrightarrow$ n
о $\longleftrightarrow$ o
п $\longleftrightarrow$ p
р $\longleftrightarrow$ r
с $\longleftrightarrow$ s

т $\longleftrightarrow$ t
ф $\longleftrightarrow$ f
х $\longleftrightarrow$ h
ц $\longleftrightarrow$ c
ю $\longleftrightarrow$ u

тф $\longleftrightarrow$ th
тх $\longleftrightarrow$ tf
пф $\longleftrightarrow$ ph
пх $\longleftrightarrow$ pf

ч $\longleftrightarrow$ ch
цк*х $\longleftrightarrow$ ck$^+$h
у $\longleftrightarrow$ ou
ой*ю $\longleftrightarrow$ oy$^+$u
ъ $\longleftrightarrow$ oa
ох*а $\longleftrightarrow$ oh$^+$a
къ $\longleftrightarrow$ q
кг$^+$ъ $\longleftrightarrow$ kg*oa

ш $\longleftrightarrow$ sh
ск*х $\longleftrightarrow$ sk$^+$h
э $\longleftrightarrow$ oe
ой*е $\longleftrightarrow$ oy$^+$e
ы $\longleftrightarrow$ ea
ех*а $\longleftrightarrow$ eh$^+$a
кс $\longleftrightarrow$ x
кг$^+$с $\longleftrightarrow$ kg*s

щ $\longleftrightarrow$ shch
шт*ч $\longleftrightarrow$ sht$^+$ch
ё $\longleftrightarrow$ yo
йи*о $\longleftrightarrow$ yi$^+$o
ь $\longleftrightarrow$ ie
ий*е $\longleftrightarrow$ iy$^+$e
ув $\longleftrightarrow$ w
уф$^+$в $\longleftrightarrow$ ouf*v

я $\longleftrightarrow$ ya
йи*а $\longleftrightarrow$ yi$^+$a

Γ_2:

йи*у $\longleftrightarrow$ yi*ou
ёй*ю $\longleftrightarrow$ yoy$^+$u
йи*ой*ю $\longleftrightarrow$ yi$^+$oy$^+$u

йи*ъ $\longleftrightarrow$ yi*oa
ёх*а $\longleftrightarrow$ yoh$^+$a
йи*ох*а $\longleftrightarrow$ yi$^+$oh$^+$a

йи*э $\longleftrightarrow$ yi*oe
ёй*е $\longleftrightarrow$ yoy$^+$e
йи*ой*е $\longleftrightarrow$ yi$^+$oy$^+$e

ой*ы $\longleftrightarrow$ oy*ea
эх*а $\longleftrightarrow$ oeh$^+$a
ой*ех*а $\longleftrightarrow$ oy$^+$eh$^+$a

ий*ы $\longleftrightarrow$ iy*ea
ьх*а $\longleftrightarrow$ ieh$^+$a
ий*ех*а $\longleftrightarrow$ iy$^+$eh$^+$a

ёй*ы $\longleftrightarrow$ yoy*ea
йи*ой*ы $\longleftrightarrow$ yi$^+$oy*ea
йи*эх*а $\longleftrightarrow$ yi*oeh$^+$a
ёй*ех*а $\longleftrightarrow$ yoy$^+$eh$^+$a
йи*ой*ех*а $\longleftrightarrow$ yi$^+$oy$^+$eh$^+$a

$$\text{кш} \longleftrightarrow \text{xh} \qquad \text{кщ} \longleftrightarrow \text{xhsh}$$
$$\text{кг}^{+}\text{ш} \longleftrightarrow \text{kg}^{*}\text{sh} \qquad \text{кг}^{+}\text{щ} \longleftrightarrow \text{kg}^{*}\text{shch}$$
$$\text{кск}^{*}\text{х} \longleftrightarrow \text{xk}^{+}\text{h} \qquad \text{кшт}^{*}\text{ч} \longleftrightarrow \text{xht}^{+}\text{ch}$$
$$\text{кг}^{+}\text{ск}^{*}\text{х} \longleftrightarrow \text{kg}^{*}\text{sk}^{+}\text{h} \qquad \text{кг}^{+}\text{шт}^{*}\text{ч} \longleftrightarrow \text{kg}^{*}\text{sht}^{+}\text{ch}$$

$$\text{йи}^{*}\text{ув} \longleftrightarrow \text{yi}^{*}\text{w}$$
$$\text{йи}^{*}\text{уф}^{+}\text{в} \longleftrightarrow \text{yi}^{*}\text{ouf}^{*}\text{v}$$

Теорема 3. *Описанное выше множество* Γ*, а также* Γ^{-1} *– корректные системы правил,* Γ *задает англо-русскую биекцию* $F : \mathcal{R}^{*} \longrightarrow \mathcal{A}^{*}$*,* Γ^{-1} *задает обратную функцию* F^{-1}*.*

Доказательство. По теореме 1 Γ и Γ^{-1} – корректные системы правил. $F(\omega) = \Gamma(\omega)$ есть биективная функция из $\mathcal{R}^{*}$ в $\mathcal{A}^{*}$ (построена биективной в предыдущем разделе). По теореме 2 система правил Γ^{-1} задает обратную функцию F^{-1}. □

Для получения $\Gamma(\xi)$ по данному $\xi \in \mathcal{R}^{*}$ и $\Gamma^{-1}(\eta)$ по данному $\eta \in \mathcal{A}^{*}$ можно использовать алгоритм 1. При переводе на компьютере программа-транслитератор может оперировать правилами из Γ, представленными в (*,+)-нотации либо как-то иначе.

Для транслитерации вручную гораздо проще использовать множество ключевых правил Γ_1 и сформулированные в предыдущем разделе законы 1, 2 и 3. В этом случае в алгоритме 1 выбирается максимальный отрезок с учетом конфликтующих «наложений». В предыдущем разделе показано, что использование трех законов равносильно применению правил из Γ_1.

8 Обобщенная транслитерация текста

Кратко рассмотрим прикладную задачу транслитерации. Пусть $\mathcal{R}^{c}$ – алфавит заглавных русских букв, $\mathcal{A}^{c}$ – заглавных английских; $\mathcal{Z}$ – алфавит посторонних символов.

Определение 18. Алфавит $\mathcal{R} \cup \mathcal{R}^{c} \cup \mathcal{Z}$ назовем *расширенным русским* алфавитом, язык $(\mathcal{R} \cup \mathcal{R}^{c} \cup \mathcal{Z})^{*}$ всех слов над ним – *расширенным русским* языком. Аналогично $\mathcal{A} \cup \mathcal{A}^{c} \cup \mathcal{Z}$ – *расширенный английский* алфавит, $(\mathcal{A} \cup \mathcal{A}^{c} \cup \mathcal{Z})^{*}$ – *расширенный английский* язык. Слово расширенного языка будем называть *текстом*.

Требуется на основе англо-русской биекции $F : \mathcal{R}^{*} \longrightarrow \mathcal{A}^{*}$ построить биекцию $G : (\mathcal{R} \cup \mathcal{R}^{c} \cup \mathcal{Z})^{*} \longrightarrow (\mathcal{A} \cup \mathcal{A}^{c} \cup \mathcal{Z})^{*}$. Это можно сделать просто и естественно следующим образом.

Определение 19. *Словом в узком смысле* над расширенным алфавитом $\mathcal{R} \cup \mathcal{R}^c \cup \mathcal{Z}$ (или $\mathcal{A} \cup \mathcal{A}^c \cup \mathcal{Z}$) назовем конечную последовательность букв из $\mathcal{R} \cup \mathcal{R}^c$ (соответственно, из $\mathcal{A} \cup \mathcal{A}^c$), в которой все буквы строчные, кроме, быть может, первой, которая допускается заглавной.

Утверждение 7. *Любой текст над $\mathcal{R} \cup \mathcal{R}^c \cup \mathcal{Z}$ (или $\mathcal{A} \cup \mathcal{A}^c \cup \mathcal{Z}$) можно представить в виде конкатенации наименьшего возможного числа слов в узком смысле и последовательностей символов из $\mathcal{Z}$ единственным образом.*

Алгоритм 2 (транслитерации текста)**.** Пусть дано $\tau \in (\mathcal{R} \cup \mathcal{R}^c \cup \mathcal{Z})^*$, получим $G(\tau)$.

1. Проделаем разбиение τ, описанное в утверждении 7.
2. Пусть ω – слово в узком смысле, содержащееся в разбиении. Тогда заменим первую букву на строчную, получим перевод $F(\omega)$, а затем заменим первую букву на заглавную, если у исходного слова она была заглавной.
3. Сконкатенируем транслитерированные слова в узком смысле и оставленные без изменения последовательности символов над $\mathcal{Z}$. Получен образ $G(\tau)$.

Полностью аналогично можно получить $G^{-1}(\upsilon)$ по данному $\upsilon \in (\mathcal{A} \cup \cup \mathcal{A}^c \cup \mathcal{Z})^*$. Несложно установить, что построенная функция G биективна.