

Парная эволюция двух нагруженных деревьев

Горбунов К.Ю.¹, Кешабян Р.Р.²

¹ Институт проблем передачи информации им. А.А. Харкевича РАН, 127051, г.Москва, Б. Каретный пер., д.19, стр.1

² Российский государственный университет им. А.Н. Косыгина (Технологии. Дизайн. Искусство), 117997, г.Москва, ул. Садовническая, д.33, стр.1
gorbunov@iitp.ru, keshabyn@mail.ru

Аннотация. В статье [Горбунов, К.Ю., Любецкий, В.А.: Точный квадратичный алгоритм кратчайшего преобразования деревьев. Доклады Российской академии наук. Математика, информатика, процессы управления 519(1), 22–27 (2024)] решалась задача выравнивания двух данных нагруженных деревьев, где задавалась цена сопоставления каждой пары символов и требовалось преобразовать деревья заданными операциями (добавляющими новые вершины-делеции) к топологически изоморфным так, чтобы суммарная цена сопоставления символов в соответствующих вершинах была максимальной. Для этой задачи в упомянутой статье предложен новый точный квадратичный по сложности алгоритм. Там рассматривался набор из трёх операций: добавление вершин-делеций к ребру или корню дерева и подъём вершины. В данной заметке мы покажем, что тот же результат можно получить, построив два вложения: первого дерева во второе и второго в первое, отражающих их парную эволюцию друг относительно друга. Качества этих вложений определяет степень согласованности деревьев точнее, чем цена выравнивания из упомянутой статьи.

Ключевые слова: дискретная оптимизация, точный квадратичный алгоритм, эволюционное дерево, эволюционные события, согласование деревьев.

1 Постановка задачи

Задача согласования двух филогенетических деревьев имеет длинную историю. В [1] описан алгоритм построения вложения одного дерева в другое, минимизирующий суммарную цену событий дубликации, потери и горизонтального переноса. В нём деревья неравноправны: вложение отражает эволюцию первого дерева (обычно это дерево генов) во второе (обычно, дерево видов). С другой стороны, в [2] решалась задача выравнивания двух данных деревьев, где задавалась цена $c(x,y)$ сопоставления каждой пары символов x,y и требовалось преобразовать деревья заданными операциями (добавляющими новые вершины-делеции) к топологически изоморфным так, чтобы суммарная цена сопоставления символов в соответствующих вершинах была максимальной. Для этой зада-

чи в [2] предложен новый точный квадратичный по сложности алгоритм. Там рассматривался набор из трёх операций: добавление вершин-делеций к ребру или корню дерева и подъём вершины. Здесь деревья равноправны, но итоговое выравнивание не полностью определяет эволюционные события в их совместной эволюции. Сейчас мы определим вложение одного дерева в другое так, что эволюционные события при вложении дерева T_1 в T_2 и при вложении T_2 в T_1 двойственны друг другу, что обеспечивает равноправность T_1 и T_2 . Из каждого вложения можно получить выравнивание T_1 и T_2 , при этом качества этих двух вложений (или их среднее арифметическое) позволяют точнее определить степень согласованности деревьев, чем цена выравнивания из [2].

Дадим необходимые определения. *Нагруженным деревом* называется бинарное корневое филогенетическое дерево, каждая вершина которого помечена символом, например, клеточным типом. Все деревья считаем растущими «сверху вниз» и снабжёнными *корневым ребром*, т.е. ребром, идущим от корня вверх, его верхний конец называем *суперкорнем* и обозначаем s . Для дальнейшего введём обозначения: нижний конец ребра e обозначим как e_- , верхний – как e_+ . *Поддеревом* дерева называем дерево, содержащее всё, что ниже некоторого ребра e , обозначаем его T_e или $T(e)$. *Верхней* относительно ребра e вершиной в T назовём вершину, расположенную выше или несравнимо с e .

Модификацией T^* дерева T назовём дерево, полученное в результате применения к нему цепочки операций подъёма вершины (операция 3 на рис. 1 из [1]), помеченной не делецией. В отличие от [2] у нас лист с делецией не возникает, т.е. операция имеет следующий вид, см. Рис. 1.

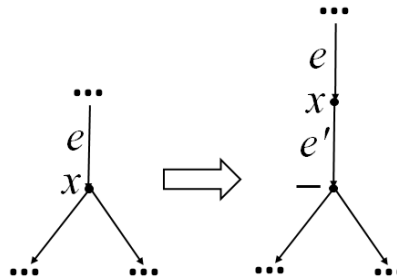


Рис. 1. Операция подъёма вершины с пометкой x .

Таким образом, операция разбивает ребро e на два ребра: верхнее ребро по-прежнему обозначаем e , нижнее – e' . Отметим: в результате операции возникает *нестрогое бинарное дерево*, т.е. в нём есть вершины (помеченные не делецией) ровно с одной дочерней вершиной. В частности, такова вершина x на Рис. 1 справа.

Частью дерева T называем дерево T' , полученное из T удалением, во-первых, дополнения до некоторого поддерева (*первая часть* преобразования), и, во-

вторых, некоторых рёбер и всего, что ниже них (*вторая часть*). Верхние концы этих рёбер остаются, таким образом, часть – нестрогое бинарное дерево.

Итак, даны два нагруженных строго бинарных дерева T_1 и T_2 (в разметках делеция отсутствует). Для наглядности рёбра в T_2 будем представлять в виде полых труб и называть *трубами*. Вложением T_1 в T_2 назовём отображение f вершин из непустой части $T_1^{*'}$ модификации дерева T_1 во множество вершин и рёбер некоторой модификации T_2^* дерева T_2 , сохраняющее (строгое) отношение предок-потомок. Дополнительно требуется, чтобы суперкорень s отображался в некоторую трубу. Делеционной вершиной в T_2^* назовём вершину ниже $f(s)$ такую, что она и все вершины ниже неё не имеют прообраза относительно f . Каждому вложению сопоставим множество следующих *событий*, сумма цен которых определяет *качество* вложения.

- 1) *Корневое удаление* – множество D вершин в T_1^* , удалённых при первой части преобразования T_1^* в $T_1^{*'}$. Цена события – некоторая функция h_1 от $|D|$.
- 2) *Рядовое удаление* – множество D вершин (максимального по включению) поддерева в T_1^* , удалённых при второй части преобразования T_1^* в $T_1^{*'}$. Цена события – некоторая функция h_2 от $|D|$.
- 3) *Корневой проросок* – множество D верхних относительно $f(s)$ вершин в T_2^* . Цена события – некоторая функция h_3 от $|D|$.
- 4) *Рядовой проросок* – множество D вершин (максимального по включению) поддерева в T_2^* , состоящего из делеционных вершин. Цена события – некоторая функция h_4 от $|D|$.
- 5) *Соответствие* «вершина-вершина», «вершина-труба» или «ребро-вершина». Цена события равна цене сопоставления соответствующих символов, при этом символом вершины считаем её пометку, символом трубы – «#», символом ребра – «~». Также полагаем: $c(-,-)=c(s,\#)=0$.

Цены событий 1–4 отрицательные или нулевые, цена события 5 может быть любой. Задача состоит в построении вложения $T_1 \rightarrow T_2$ максимального качества.

2 Описание алгоритма

Опишем алгоритм построения искомого вложения $T_1 \rightarrow T_2$. Перебираем пары: ребро e из T_1 , труба t из T_2 . Они перебираются от листьев к корню в лексикографическом порядке. Для каждой пары (e,t) вычисляем её качество $q(e,t)$, равное максимальной цене вложения $f_{e,t}$ поддерева T_e в поддерево T_t при условии, что $f_{e,t}(e)=t$. Кроме $q(e,t)$ вычисляем ещё три величины: $q(e',t')$, $q(e',t)$ и $q(e,t')$, которые вычисляются перед $q(e,t)$ в указанном порядке. Они имеют тот же смысл, что и $q(e,t)$, но с условием, что в вершинах, соответственно, e_- , t_- или в обеих, пометка заменена на делецию (им соответствуют вложения $f_{e',t'}$, $f_{e',t}$ и $f_{e,t'}$). Опишем шаг индукции при их вычислении (базисом является случай 5 ниже при листовой трубе t).

Итак, ребро e (или e') начинается в трубе t (или t'). Вершины e_- (или e'_-) и t_- (или t'_-) помечены символами, соответственно, x и y , которые в случае e'_- или t'_- являются делециями. Если e и t не листовые, обозначим их дочерние рёбра,

соответственно, e_1 , e_2 и t_1 , t_2 . Выбираем случай с максимальным качеством из следующих.

1. Стандартный случай (Рис. 2.1): ребро e доходит до развилки в трубе t и разветвляется на ней (применяется, когда e и t не листовые). Имеем: $q(e,t)=q(e_1,t_1)+q(e_2,t_2)+c(x,y)$ (или $q(e,t)=q(e_1,t_2)+q(e_2,t_1)+c(x,y)$), в дальнейшем такие симметричные случаи не упоминаем). Отображение $f_{e,t}$ – объединение построенных по предположению индукции отображений f_{e_1,t_1} , f_{e_2,t_2} и соответствий $f_{e,t}(e_-)=t_-$, $f_{e,t}(e_+)=t$. Величины $q(e',t)$, $q(e,t')$ и $q(e',t')$ вычисляются по той же формуле, отображения $f_{e',t}$, $f_{e,t'}$ и $f_{e',t'}$ строятся аналогично (далее их не упоминаем). Событие: соответствие «вершина-вершина».

2. Удаление на развилке (Рис. 2.2): ребро e доходит до развилки в трубе t , ребро e_1 сворачивает в t_1 , e_2 -поддерево удаляется (применяется, когда t не листовое). Имеем: $q(e,t)=q(e_1,t_1)+c(x,y)+h_4(|T_2(t_2)|)$, $q(e',t)=q(e_1,t_1)+c(-,y)+h_4(|T_2(t_2)|)$, $q(e',t')=q(e_1,t_1)+h_4(|T_2(t_2)|)$, $q(e,t')=q(e_1,t_1)+c(x,-)+h_4(|T_2(t_2)|)$, где $|T_2(t_2)|$ – число вершин в дереве $T_2(t_2)$ с корневым ребром t_2 . Отображение $f_{e,t}$ – объединение отображения f_{e_1,t_1} и соответствий $f_{e,t}(e_+)=t$, $f_{e,t}(e_-)=t_-$. События: рядовой просок и соответствие «вершина-вершина».

3. Рядовой просок без подъёма (Рис. 2.3): ребро e доходит до развилки в трубе t и сворачивает в t_1 (применяется, когда t не листовое). Имеем: $q(e,t)=q(e,t_1)+c(-,y)+h_4(|T_2(t_2)|)$, $q(e,t')=q(e,t_1)+h_4(|T_2(t_2)|)$. Отображение $f_{e,t}$ – объединение отображения f_{e,t_1} и соответствия $f_{e,t}(e_+)=t$. Величины $q(e',t)$ и $q(e',t')$ вычисляются по тем же формулам с заменой всюду e на e' и (во втором случае) t на t' . События: рядовой просок и соответствие «ребро-вершина».

4. Рядовой просок с подъёмом (Рис. 2.4): ребро e доходит до развилки в трубе t , куда поднимается вершина x , и сворачивает в t_1 (применяется, когда t не листовое и только для $q(e,t)$ и $q(e,t')$). Имеем: $q(e,t)=q(e',t_1)+c(x,y)+h_4(|T_2(t_2)|)$, $q(e,t')=q(e',t_1)+c(x,-)+h_4(|T_2(t_2)|)$. Отображение $f_{e,t}$ – объединение отображения f_{e',t_1} и соответствий $f_{e,t}(e_+)=t$ и $f_{e,t}(e_-)=t_-$. События: рядовой просок и соответствие «вершина-вершина».

5. Листовая остановка (Рис. 2.5): листовое ребро e доходит до развилки или листа в трубе t и останавливается (применяется, когда e листовое). Имеем: $q(e,t)=c(x,y)+h_4(|T_2(t_1)|)+h_4(|T_2(t_2)|)$ (если t листовое, $q(e,t)=c(x,y)$, это базис индукции). Другие три величины вычисляются по тем же формулам, где x или y – деление. Отображение $f_{e,t}$ – соответствия $f_{e,t}(e_+)=t$, $f_{e,t}(e_-)=t_-$. События: два рядовых проскока (если t не листовое) и соответствие «вершина-вершина».

6. Простой x -подъём (Рис. 2.6): в результате подъёма вершины e_- ребро e распадается на два ребра, и нижнее ребро e' продолжается в трубе t (применяется только для $q(e,t)$ и $q(e,t')$). Имеем: $q(e,t)=q(e',t)+c(x,\#)$, $q(e,t')=q(e',t')+c(x,\#)$. Отображение $f_{e,t}$ – объединение отображения $f_{e',t}$ и соответствий $f_{e,t}(e_+)=t$ и $f_{e,t}(e'_+)=t$. Событие: соответствие «вершина-труба».

7. Простой y -подъём (Рис. 2.7): в результате подъёма вершины t_- труба t распадается на две трубы, и ребро e продолжается в трубе t' (применяется только для $q(e,t)$ и $q(e',t)$). Имеем: $q(e,t)=q(e',t')+c(-,y)$, $q(e',t)=q(e',t')+c(-,y)$. Отображение $f_{e,t}$ – объединение отображения $f_{e',t'}$ и соответствия $f_{e,t}(e_+)=t$. Событие: соответствие «ребро-вершина».

8. Двойной подъём (Рис. 2.8): в результате подъёмов вершин e_- и t_- ребро e и труба t распадаются на две части, которые соответствуют друг другу (применяется только для $q(e,t)$). Имеем: $q(e,t)=q(e',t')+c(x,y)$. Отображение $f_{e,t}$ – объединение отображения $f_{e',t'}$ и соответствий $f_{e,t}(e_+)=t$ и $f_{e,t}(e'_+)=t'_+$. Событие: соответствие «вершина-вершина».

9. Удаление в трубе без подъёма (Рис. 2.9): ребро e теряет в трубе t дочернее e_2 -поддереву, ребро e_1 продолжается в t (применяется, когда e не листовое). Имеем: $q(e,t)=q(e_1,t)+c(x,\#)+h_2(|T_1(e_2)|)$, $q(e',t)=q(e_1,t)+c(-,\#)+h_2(|T_1(e_2)|)$ (для t' аналогично). Отображение $f_{e,t}$ – объединение отображения $f_{e_1,t}$ и соответствий $f_{e,t}(e_+)=t$, $f_{e,t}(e_-)=t$. События: рядовое удаление и соответствие «вершина-труба».

10. Удаление в трубе с подъёмом (Рис. 2.10): в результате подъёма вершины t_- труба t распадается на две трубы, и ребро e теряет в вершине t'_+ дочернее e_2 -поддереву, ребро e_1 продолжается в t' (применяется, когда e не листовое и только для $q(e,t)$ и $q(e',t)$). Имеем: $q(e,t)=q(e_1,t)+c(x,y)+h_2(|T_1(e_2)|)$, $q(e',t)=q(e_1,t)+c(-,y)+h_2(|T_1(e_2)|)$. Отображение $f_{e,t}$ – объединение отображения $f_{e_1,t'}$ и соответствий $f_{e,t}(e_+)=t$, $f_{e,t}(e_-)=t'_+$. События: рядовое удаление и соответствие «вершина-вершина».

11. Обрыв (Рис. 2.11): ребро e теряет в развилке или в листе трубы t оба дочерних поддерева (применяется, когда e не листовое). Имеем: $q(e,t)=c(x,y)+h_2(|T_1(e_1)|)+h_2(|T_1(e_2)|)+h_4(|T_1(t_1)|)+h_4(|T_2(t_2)|)$ (и аналогично для e' и t' с заменами x или y на делецию). Отображение $f_{e,t}$ – соответствия $f_{e,t}(e_+)=t$, $f_{e,t}(e_-)=t_-$. События: два рядовых удаления, два рядовых проскока (если t не листовое) и соответствие «вершина-вершина».

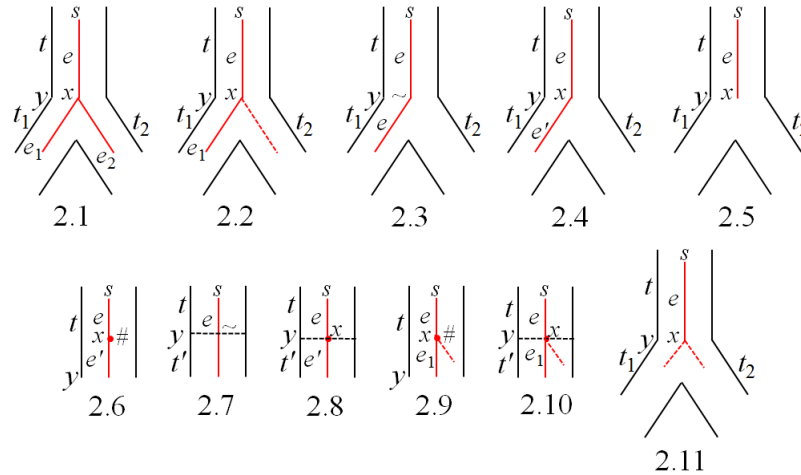


Рис. 2. Случаи, рассматриваемые алгоритмом. Чёрным пунктиром обозначен раздел между двумя частями трубы, красным пунктиром – удалённые из T_1 ветви.

Вычислим «поправленные» качества $c(e,t)$ вложений $f_{e,t}$, вычтя из каждого качества $q(e,t)$ штраф за корневое удаление (где D – множество верхних относи-

тельно e вершин) и за корневой проскок (где D – множество верхних относительно t вершин). Максимальное $c(e,t)$ определяет итоговое вложение.

Легко заметить связь между выравниванием деревьев T_1 и T_2 в смысле [2] и их вложениями друг в друга. Действительно, если к T_1 добавить части, соответствующие делеционным и верхним относительно $f(s)$ вершинам в T_2 (в местах, определяемых вложением $T_1 \rightarrow T_2$), а в T_2 добавить части, соответствующие вершинам, удалённым из T_1 , получим аналоги достроенных и выравненных друг с другом по [2] деревьев T_1 и T_2 . И наоборот, каждое их выравнивание индуцирует вложение, например, $T_1 \rightarrow T_2$. Действительно, операции 1–3 из [2], применённые к T_1 , добавляют в T_1 поддеревья, помеченные делециями, они соответствуют проскокам при вложении и создают (при операции 1) соответствие «ребро-вершина». Те же операции, применённые к T_2 , добавляют в T_2 поддеревья, помеченные делециями, они соответствуют удалениям, а также (при операции 1) создают соответствие «вершина-труба». При этом вложения $T_1 \rightarrow T_2$ и $T_2 \rightarrow T_1$, соответствующие одному и тому же выравниванию, двойственны в следующем смысле. События удаления в одном из них взаимно-однозначно соответствуют событиям проскока в другом. Также имеется взаимно-однозначное соответствие событий сопоставления ребра вершине и вершины трубе, а также сопоставлений «символ-делеция» и «делеция-символ». Соответствия равных и неравных символов одни и те же в обоих вложениях.

Можно было бы добавить в список событий предыдущего раздела дубликацию, т.е. развилку ребра дерева T_1 в трубе дерева T_2 . Алгоритм построения оптимального вложения остаётся прежним (с добавлением случая дубликации). Мы не стали его добавлять, поскольку двойственное к дубликации событие – раздвоение поддерева. не является естественным.

3 Точность и время алгоритма; пример его работы

Легко проверить, что при вложении $T_e \rightarrow T_t$ возможны ровно 11 вышеописанных случаев. В каждом из них точность алгоритма следует из предположения индукции. Время работы алгоритма квадратично, поскольку таково число пар (e,t) , а обработка каждой пары занимает константное время.

В качестве примера работы алгоритма рассмотрим тот же пример, на котором иллюстрировался алгоритм в [2]. Напомним рис. 3 из [2] (см. Рис. 3), где на рис. 3a показаны исходные деревья, на рис. 3b – их выравнивание при единичных сходствах и штрафах, а на рис. 3c – их выравнивание при цене 4 сопоставления равных символов, неравных – минус 3 и штрафа 2 за делецию.

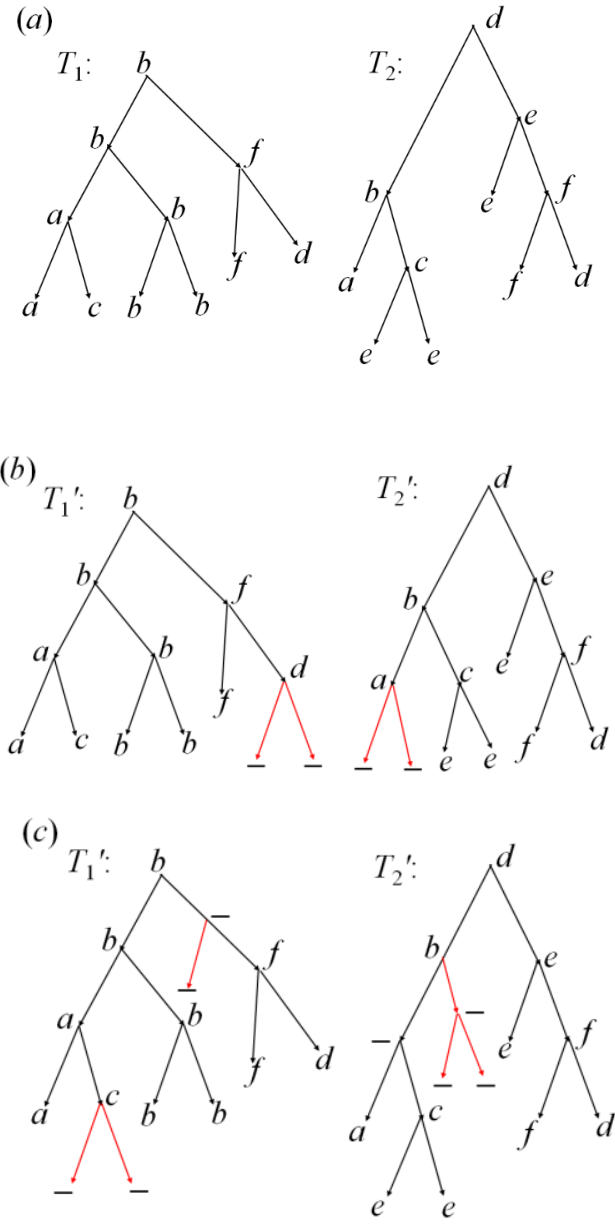


Рис. 3. Рисунок 3 из [2].

Покажем результат нашего алгоритма – вложения $T_1 \rightarrow T_2$ и $T_2 \rightarrow T_1$. Если задать единичную цену соответствия всех пар символов и значения всех функций h_1-h_4 равными минус мощности аргумента, получаем пару вложений, показанных на Рис. 4.

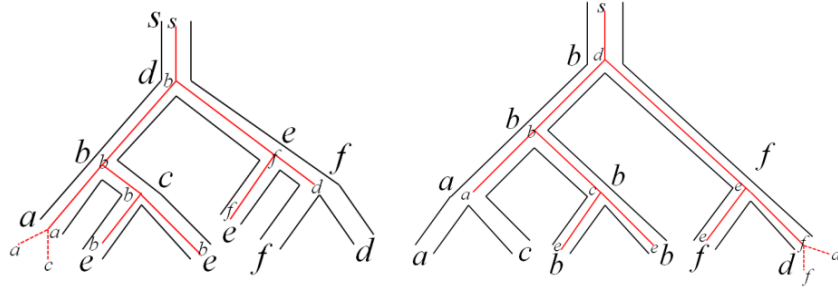


Рис. 4. Вложение $T_1 \rightarrow T_2$ (слева) и $T_2 \rightarrow T_1$ (справа) при единичных ценах соответствия всех пар символов и значениях всех функций h_1-h_4 , равных минус мощности аргумента.

Как видим, оба вложения содержат 9 соответствий пар символов, два рядовых проскока и два рядовых удаления; во всех четырёх случаях $|D|=1$. Качество каждого вложения равно 5. Можно заметить, что любое из двух вложений соответствует выравниванию, показанному на Рис. 3б.

Теперь зададим цену соответствия двух равных символов (не делеций) равной 4, двух различных символов – минус 3, цену соответствий «символ-делеция», «делеция-символ», «ребро-вершина» и «вершина-труба» – минус 2 и значения всех функций h_1-h_4 равными минус мощности аргумента, умноженной на 2. Получаем пару вложений, показанных на Рис. 5.

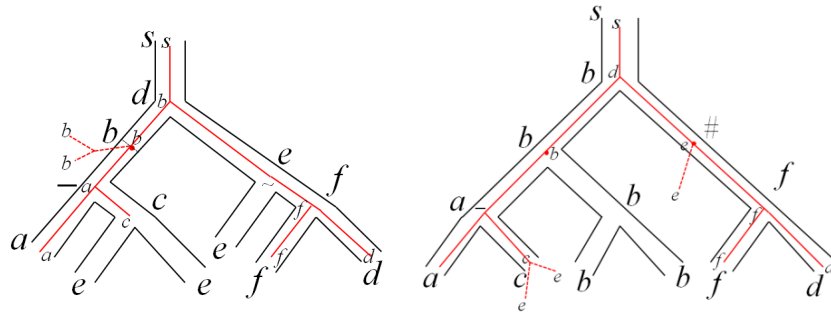


Рис. 5. Вложение $T_1 \rightarrow T_2$ (слева) и $T_2 \rightarrow T_1$ (справа) при цене соответствия двух равных символов (не делеций) равной 4, двух различных символов – минус 3, цене соответствий «символ-делеция», «делеция-символ», «ребро-вершина» и «вершина-труба» – минус 2 и значениях всех функций h_1-h_4 равных минус мощности аргумента, умноженной на 2.

Как видим, оба вложения содержат 6 соответствий пар равных символов и одну пару неравных. Первое вложение содержит одно соответствие «символ-

делеция» и одно «ребро-вершина», а второе – одно «делеция-символ» и одно «вершина-труба». Также первое вложение имеет три одноэлементных рядовых проскока и одно трёхэлементное рядовое удаление. В свою очередь, второе вложение имеет три одноэлементных рядовых удаления и один трёхэлементный рядовой проскок. Качество каждого вложения равно 5. Снова заметим, что любое из двух вложений соответствует выравниванию, показанному на Рис. 3с.

Если цены двойственных событий различаются, вложения не обязательно будут порождать одно и то же выравнивание. Таким образом, пара вложений может дать больше информации по согласованию деревьев, чем их любое выравнивание. Ещё одной причиной этого является возможность задавать аффинные и нелинейные функции h_1-h_4 , что не предусмотрено в алгоритме выравнивания из [2].

4 Заключение

Мы описали эффективный алгоритм, минимизирующий суммарную цену событий в эволюции двух деревьев друг относительно друга. В качестве показателя степени согласованности деревьев T_1 и T_2 можно взять среднее арифметическое качеств вложений $T_1 \rightarrow T_2$ и $T_2 \rightarrow T_1$. Если поменять знаки цен событий на противоположные и прибавить к ним константу так, чтобы они стали неотрицательными, получим вместо качества вложения (которое требуется максимизировать) цену вложения (которую требуется минимизировать). Тогда среднее минимальных цен вложений $T_1 \rightarrow T_2$ и $T_2 \rightarrow T_1$ естественно рассматривать как расстояние между T_1 и T_2 . Определение другого, по-видимому, более точного расстояния дано в [3]. Задача его вычисления NP-трудна, для неё в [3] описан приближённый алгоритм квадратичного времени, у которого мультипликативная ошибка не превышает 2. Представляет интерес задача уменьшения ошибки алгоритма или определения других эффективно вычисляемых расстояний.

Финансирование. Работа выполнена в рамках государственного задания ИППИ РАН, утверждённого Минобрнауки России.

Список литературы

1. Горбунов, К.Ю., Любецкий, В.А.: Об одном алгоритме согласования деревьев генов и видов с учетом дубликаций, потерь и горизонтальных переносов генов. Информационные процессы 10(2), 140–144 (2010).
2. Горбунов, К.Ю., Любецкий, В.А.: Точный квадратичный алгоритм кратчайшего преобразования деревьев. Доклады Российской академии наук. Математика, информатика, процессы управления 519(1), 22–27 (2024).
3. Olver, N., Schalekamp, F., van der Ster, S., Stougie, L., van Zuylen, A.: A duality based 2-approximation algorithm for maximum agreement forest. Mathematical Programming 198, 811–853 (2023).