

§ 3 БАЗЫ ЗНАНИЙ, ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ, ЭКСПЕРТНЫЕ СИСТЕМЫ, СИСТЕМЫ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ

Галочкин В.И.

ПЕРЕЧИСЛЕНИЕ РЕШАЮЩИХ ДЕРЕВЬЕВ НА И-ИЛИ ДЕРЕВЕ ПО МОНОТОННЫМ ОГРАНИЧЕНИЯМ

Аннотация: Рассматриваются широко применяемые в системах искусственного интеллекта И-ИЛИ деревья с заданными значениями показателей в терминальных дугах либо вершинах. Показатели рекурсивно определены на решающих деревьях с помощью непрерывных и монотонных по каждому аргументу функций свертки. Ставится задача перечисления решающих деревьев, удовлетворяющих системе ограничений на показатели. Для случая единственного аддитивного показателя ранее предложен линейный по сложности и памяти алгоритм, но уже для двух показателей не существует полиномиальных алгоритмов решения задачи. Предлагаются два алгоритма типа ветвей и границ. Первый алгоритм реализует последовательное выделение поддеревьев допустимых решений по ограничениям на отдельные показатели. Вторым алгоритмом позволяет эффективно отсекают недопустимые поддеревья. В основе обоих алгоритмов лежит введенное понятие минимальной оболочки И-ИЛИ дерева по монотонному ограничению. Первый алгоритм более применим при наличии большого числа решений системы неравенств, поскольку позволяет выбирать допустимые варианты не по отдельности, а блоками в виде поддеревьев И-ИЛИ дерева. Вторым алгоритмом ориентирован на случай, когда система неравенств имеет небольшое число решений.

Ключевые слова: Системы искусственного интеллекта, Алгоритм, Перечисление, И-ИЛИ граф, И-ИЛИ дерево, Решающее дерево, Система неравенств, Система ограничений, Монотонная функция, Сложность алгоритма

Abstract: The author reviews widely used in artificial intelligence systems AND-OR trees with specified parameters in the terminal arcs or vertices. The parameters are defined recursively on the decision trees with the use of continuous convolution functions monotone in each argument. The author solves a problem of enumerating convolution functions satisfying the system of restrictions on the parameters. Earlier a linear complexity and memory algorithm for the case of

a single additive index was presented. But even for two parameters there is no polynomial-time algorithm for solving the problem. The author suggests two "branch and bound" algorithms. The first algorithm implements a consistent selection of subtrees for allowable solutions using restrictions on separate parameters. The second algorithm can effectively cut off the unacceptable subtrees. The basis of the both algorithms is the concept of the minimum AND-OR tree subsect on monotonic restriction. The first algorithm is more applicable in the presence of a large number of solutions of the system of inequalities because it allows to choose the solutions as sets of AND-OR tree subtrees. The second algorithm is applicable in a case where the system of inequalities has a small number of solutions.

Keywords: *system of constraints, system of inequalities, decision tree, AND-OR tree, AND-OR graph, enumeration, algorithm, artificial intelligence systems, monotonic function, complexity of algorithm*

Введение

И-ИЛИ деревья широко применяются в системах искусственного интеллекта [1-4]. На основе И-ИЛИ деревьев строятся, например, методы сведения задач к подзадам [1, 3] и способы описания множества технических решений определенного класса [2].

И-ИЛИ дерево содержит два типа вершин: И-вершины и ИЛИ-вершины. Решающим деревом (деревом решения) [1] называется такое поддерево, которое строится сверху вниз и включает корень, всех сыновей для каждой своей И-вершины и единственного сына для каждой ИЛИ-вершины. Будем для краткости называть решающее дерево вариантом, как это сделано в [4].

Распространенной задачей для И-ИЛИ деревьев является перечисление вариантов с определенными свойствами [1-4]. В [1] рассматривается задача выбора варианта минимальной стоимости. Стоимость варианта определяется суммарной стоимостью его дуг.

В [2] количественные показатели приписываются вершинам И-ИЛИ дерева, а не дугам. Мы будем следовать подобному представлению. Фактически, оно получается простым переносом значения дуги на вершину, в которую она направлена. При обратном преобразовании значение показателя в вершине переносится на значение дуги, ведущей в данную вершину.

В [8] предложен алгоритм перечисления вариантов И-ИЛИ дерева, для которых стоимость не превышает заданной константы C . Описанный алгоритм позволяет получать очередной допустимый вариант с оценками по сложности и памяти $O(N)$, где N – количество вершин И-ИЛИ дерева.

В [7] рассматривается задача поиска допустимых вариантов по нескольким показателям. В листьях дерева определены значения показателей Q_i ($i = 1, 2, \dots, m$). Для каждой И-вершины z с сыновьями z_t ($t = 1, 2, \dots, k$) задана монотонная по любому аргументу непрерывная функция свертки $F_i(x_1, x_2, \dots, x_k)$, определяющая значение показателя Q_i в вершине z в зависимости от его значений в сыновьях. Тем самым показатели рекурсивно

определены на множестве вариантов. Требуется перечислить все варианты, на которых выполняется система ограничений

$$Q_i \leq C_i, \quad i = 1, 2, \dots, m.$$

Уже для двух показателей полиномиальные алгоритмы решения данной задачи отсутствуют. Действительно, рассмотрим трехуровневое И-ИЛИ дерево, корень которого является И-вершиной, а все сыновья корня – ИЛИ-вершинами. Каждая ИЛИ-вершина в свою очередь имеет двух сыновей – листьев И-ИЛИ-дерева. В одном из этих листьев задано некоторое положительное значение стоимости, а во втором стоимость нулевая. Пусть требуется найти такие варианты, на которых стоимость W удовлетворяет двум ограничениям $W \leq C$ и $-W \leq -C$, где C – некоторая положительная константа, меньшая суммы стоимостей листьев. Задача эквивалентна выбору подмножества листьев суммарной стоимости $W = C$. Это классическая NP -полная задача о сумме элементов подмножества [6].

В настоящей статье рассматриваются два подхода к решению поставленной задачи на основании обобщения введенных в [8] понятий минимальной оболочки и допустимого поддеревья решений. Будем сначала считать, что функции свертки всех показателей Q_i монотонно возрастают по каждому аргументу.

1. Минимальная оболочка по монотонному ограничению

В [8] вводятся определение и способ построения минимальной оболочки по ограничению на аддитивный показатель стоимости. Распространим данное определение на случай произвольного показателя с монотонно возрастающими функциями свертки.

Итак, имеется И-ИЛИ дерево D с N вершинами. Рассмотрим некоторый показатель Q , значения которого определены в листьях дерева. Путем обхода дерева варианта от листьев к корню показатель Q определен рекурсивно на множестве вершин варианта. Действительно, для вершины z с сыновьями z_i при $i = 1, 2, \dots, k$, соответствующими значениями показателя q_i и функцией свертки $F(x_1, x_2, \dots, x_k)$

$$Q(z) = F(q_1, q_2, \dots, q_k).$$

Величина показателя на всем варианте определяется его значением в корне.

Напомним некоторые определения из [8]. Поддеревом вариантов называется множество всех вариантов такого поддерева, которое включает корень, всех сыновей для каждой своей И-вершины и не менее одного сына для каждой ИЛИ-вершины. Пересечение поддеревьев вариантов представляет собой множество вариантов подобного И-ИЛИ поддерева, поэтому также является поддеревом вариантов.

Будем считать допустимыми варианты И-ИЛИ дерева, удовлетворяющие ограничению $Q \leq C$. Минимальной оболочкой И-ИЛИ дерева по ограничению $Q \leq C$ называется такое поддерево вариантов, которое

- включает все допустимые варианты исходного И-ИЛИ-дерева;
- не может быть усечено без потери допустимых вариантов.

Из этого определения следует, что каждая вершина минимальной оболочки входит в какой-либо допустимый вариант. Рассмотрим способ построения минимальной оболочки.

Для любой вершины z найдем минимальный отрезок $[a_z, b_z]$, в котором лежат все возможные значения монотонно возрастающей функции свертки F , определяющей показатель Q . Будем называть $[a_z, b_z]$ отрезком возможных значений показателя Q в вершине z .

Выполним обход дерева в направлении от листьев к корню r . Пусть z_i ($i = 1, 2, \dots, k$) - сыновья вершины z , а $[a_i, b_i]$ – рассчитанные отрезки возможных значений показателя Q в вершинах z_i (для листьев $a_i = b_i = z_i$).

Если z – И-вершина, то $a_z = F(a_1, a_2, \dots, a_k)$; $b_z = F(b_1, b_2, \dots, b_k)$.

Если z – ИЛИ-вершина, то $a_z = \min(a_i)$; $b_z = \max(b_i)$ для $i = 1, 2, \dots, k$.

Таким образом находятся отрезки возможных значений для всех вершин И-ИЛИ дерева. Если окажется, что в корне $a_r > C$, то минимальная оболочка пуста. Это означает, что допустимые варианты отсутствуют. Если же $b_r \leq C$, то любой вариант И-ИЛИ дерева является допустимым, а минимальная оболочка совпадает со всем деревом. Будем далее считать, что $a_r \leq C \leq b_r$, что гарантирует наличие допустимых вариантов.

Следующий обход выполняется наоборот от корня дерева к листьям. При тех же обозначениях для любой вершины z_i найдем максимальное значение h_i показателя Q , при котором может существовать допустимый вариант, включающий вершину z_i .

Пусть для вершины z с сыновьями z_i ($i = 1, 2, \dots, k$) указанное значение равно h_z (для корня $h_r = C$).

Если z – И-вершина, рассмотрим для каждого i неравенство

$$F(a_1, a_2, \dots, a_{i-1}, x_i, a_{i+1}, \dots, a_k) \leq h_z.$$

Обозначим через y_i максимальное значение x_i , при котором выполняется это неравенство. Тогда

$$h_i = \min(y_i, b_i), i = 1, 2, \dots, k.$$

Ввиду монотонности и непрерывности функции F по каждому аргументу, значение y_i легко находится каким-либо численным методом, например, методом дихотомии.

Если z – ИЛИ-вершина, то отседем те вершины z_i , для которых $h_z < a_i$. Для остальных вершин $h_i = \min(h_z, b_i)$.

В результате описанной процедуры получается поддереву вариантов T , являющееся минимальной оболочкой дерева D по ограничению $Q \leq C$. Если считать, что сложность решения каждого неравенства по x_i ограничено некоторой общей константой, то сложность построения минимальной оболочки оценивается величиной $O(N)$.

2. Допустимые поддеревья вариантов

В [8] рассматривается получение допустимых поддеревьев вариантов по аддитивному показателю стоимости. Путем незначительной модификации можно находить допустимые поддеревья вариантов и по любому ограничению с монотонно возрастающими функциями

ями свертки, используя обход И-ИЛИ дерева в порядке сверху вниз.

Пусть для вершины z с сыновьями z_i ($i = 1, 2, \dots, k$) и функцией свертки $F(x_1, x_2, \dots, x_k)$ получено ограничение $Q(z) \leq d_z$ (для корня $d_r = C$). Наша задача состоит в том, чтобы получить ограничения вида $Q(z_i) \leq d_i$.

Если z – И-вершина, подберем произвольные значения d_i из отрезков $[a_i, b_i]$ такие, что

$$F(d_1, d_2, \dots, d_k) \leq d_z.$$

Можно, например, определять величины d_i по формулам $d_i = d_i(t) = a_i + t(b_i - a_i)$ и затем находить максимальное значение t на отрезке $[0, 1]$, при котором неравенство выполняется. Будем считать, что в i -й вершине получено ограничение $Q(z_i) \leq d_i$.

Если z – ИЛИ-вершина, отсечем вершины z_i , для которых $d_z < a_i$, а для остальных вершин положим $d_i = d_z$.

В итоге выделяется допустимое поддерево вариантов P . Будем говорить, что поддерево P получено в результате спуска ограничения $Q \leq C$.

В общем случае, минимальная оболочка не является допустимым поддеревом вариантов по ограничению $Q \leq C$, но гарантированно содержит вариант с минимальным значением Q . Снова если считать, что сложность решения каждого неравенства по t составляет $O(1)$, то сложность получения допустимого поддерева вариантов оценивается величиной $O(N)$.

3. Перечисление вариантов путем выделения допустимых поддеревьев

В [8] представлен алгоритм выделения допустимых поддеревьев вариантов, удовлетворяющих единственному ограничению на аддитивный показатель. После нахождения очередного поддерева оставшееся множество вариантов разлагается на ряд поддеревьев, гарантированно содержащих решения. Описанные обобщения позволяют использовать данный алгоритм для любого ограничения с монотонно возрастающими функциями свертки. Если оценивать сложность решения приведенных выше элементарных неравенств по x_i и t величиной $O(1)$, то сложность получения очередного допустимого поддерева составляет $O(N)$.

Переходя к случаю системы ограничений, рассмотрим сначала наиболее естественный подход. Построим алгоритм типа ветвей и границ [5] для перечисления вариантов, удовлетворяющих неравенствам

$$Q_i \leq C_i, \quad i = 1, 2, \dots, m.$$

В соответствии с алгоритмом из [8] найдем первое допустимое поддерево вариантов по ограничению $Q_1 \leq C_1$. Будем в дальнейшем рассматривать его в качестве исходного И-ИЛИ дерева по остальным ограничениям.

Затем выделим допустимое поддерево вариантов по ограничению $Q_2 \leq C_2$ и т. д. Если после m шагов достигнут непустой результат, то полученное поддерево вариантов будет удовлетворять всем ограничениям системы. Если же на каком-то шаге минимальная

оболочка по очередному ограничению даст пустое дерево, перейдем в соответствии с алгоритмом из [8] к следующему допустимому поддереву вариантов по предыдущему ограничению.

Такой подход оправдан при наличии большого числа решений системы неравенств, поскольку алгоритм из [8] позволяет выбирать допустимые по каждому ограничению варианты не по отдельности, а блоками в виде поддеревьев И-ИЛИ дерева, в которых все варианты допустимы.

Можно рекомендовать этот подход для измененной постановки задачи, когда показатели в системе ограничений ранжированы по важности. В этом случае имеет смысл выдавать допустимые по наиболее важным ограничениям поддерева вариантов в качестве промежуточных результатов для дальнейшего исследования.

4. Перечисление вариантов путем отсекаания недопустимых поддеревьев

Опишем процедуры ветвления и отсекаания метода ветвей и границ, лежащие в основе данного алгоритма.

Для ветвления будем использовать стек из ИЛИ-вершин в порядке обхода дерева сверху вниз. У элемента из вершины стека выбирается очередной сын и временно запрещаются другие сыновья. В полученном И-ИЛИ дереве корректируются отрезки возможных значений всех показателей.

Для отсекаания будем строить последовательность минимальных оболочек по отдельным неравенствам системы, сокращая на каждом шаге исходное И-ИЛИ дерево. В [7] процедура отсекаания основана на простой оценке экстремумов показателей.

Наиболее полное отсекаание обеспечивается при циклическом рассмотрении неравенств, когда после последнего ограничения снова выбирается первое, затем второе и т. д. Действительно, после отсекаания И-ИЛИ дерева по ограничению на некоторый показатель в общем случае меняются отрезки возможных значений других показателей, поэтому повторное отсекаание по какому-либо из предыдущих ограничений может быть продуктивным. Процесс заканчивается, когда нет отсеканий для m ограничений подряд. Для каждой вершины полученного дерева и любого ограничения гарантируется существование варианта, включающего данную вершину и удовлетворяющего данному ограничению. Тем не менее наличие варианта, удовлетворяющего всей системе ограничений, очевидно, не обеспечивается.

Опишем алгоритм по шагам.

1. Объявление исходного И-ИЛИ дерева текущим.
2. Организация пустого стека ИЛИ-вершин.
3. Применение процедуры отсекаания к текущему И-ИЛИ дереву.
4. Если в текущем И-ИЛИ дереве остается единственный вариант, то вывод этого варианта в качестве очередного решения системы неравенств.
5. Если в текущем И-ИЛИ дереве остается более одного варианта, то включение в стек очередной ИЛИ-вершины в порядке обхода дерева сверху вниз.

6. Если стек не пуст и не все сыновья элемента из вершины стека выбраны, то применение процедуры ветвления и переход к шагу 3.
7. Если стек не пуст, то удаление элемента из вершины стека и переход к шагу 3.
8. Конец.

В случае полного отсечения дерева на шаге 3 выбирается следующий сын элемента из вершины стека, как и при получении и выдаче единственного допустимого варианта.

Наличие после отсечения более одного варианта дает возможность дальнейшего ветвления.

После перебора всех сыновей элемента из вершины стека происходит возврат к предыдущей ИЛИ-вершине стека и выбор его очередного неисследованного сына.

Условие непустоты стека на шаге 6 обеспечивает правильное завершение в случае получения единственного варианта либо пустого дерева при начальном отсечении.

Алгоритм позволяет эффективно отсекал недопустимые варианты и в отличие от первого алгоритма ориентирован на случай, когда система неравенств имеет небольшое число решений.

Заметим в заключение, что приведенные алгоритмы легко переносятся на случай произвольной монотонности функций свертки по своим аргументам.

Библиография :

1. Нильсон Н. Искусственный интеллект. Методы поиска решений / Н. Нильсон. М.: Мир, 1973. 270 с.
2. Автоматизация поискового конструирования / А.И. Половинкин, Н.К. Бобков, Г.Я. Буш и др. Под ред. А.И. Половинкина. М.: Радио и связь, 1981. 344 с.
3. Братко И. Программирование на языке Пролог для искусственного интеллекта / И. Братко. М.: Мир, 1990. 560 с.
4. Кручинин В.В. Методы построения алгоритмов генерации и нумерации комбинаторных объектов на основе деревьев И/ИЛИ / В.В. Кручинин. Томск: В-Спектр, 2007. 200 с.
5. Рейнгольд Э. Н. Комбинаторные алгоритмы. Теория и практика / Э. Рейнгольд, Ю. Нивергельт, Н. Део. М.: Мир, 1980. 476 с.
6. Кормен Т. Х. Алгоритмы: построение и анализ, 2-е изд.: Пер. с англ. / Т. Кормен, Ч. Лайзерсон, Р. Ривест, К. Штайн. М.: Издательский дом "Вильямс", 2005. 1296 с.
7. Николаев С.А. Метод ветвей и границ для поиска технических решений. // Автоматизированные подсистемы поискового конструирования: Межвузовский сборник. Горький: ГГУ, 1981. С. 139-144.
8. Галочкин В. И. Перечисление решающих деревьев ограниченной стоимости на И-ИЛИ дереве // Программные системы и вычислительные методы. 2014. № 1. С. 191 – 196. DOI: 10.7256/2305-6061.2014.2.11925.

References:

1. Nil'son N. Iskusstvennyi intellekt. Metody poiska reshenii / N. Nil'son. M.: Mir, 1973. 270 s.

2. Avtomatizatsiya poiskovogo konstruirovaniya / A.I. Polovinkin, N.K. Bobkov, G.Ya. Bush i dr. Pod red. A.I. Polovinki-na. M.: Radio i svyaz', 1981. 344 s.
3. Bratko I. Programmirovaniye na yazyke Prolog dlya iskusstvennogo intellekta / I. Bratko. M.: Mir, 1990. 560 s.
4. Kruchinin V.V. Metody postroeniya algoritmov generatsii i numeratsii kombinatornykh ob'ektov na osnove derev'ev I/II / V.V. Kruchinin. Tomsk: V-Spektr, 2007. 200 s.
5. Reingol'd E. N. Kombinatornye algoritmy. Teoriya i praktika / E. Reingol'd, Yu. Nivergel't, N. Deo. M.: Mir, 1980. 476 s.
6. Kormen T. Kh. Algoritmy: postroenie i analiz, 2-e izd.: Per. s angl. / T. Kormen, Ch. Laizerson, R. Rivest, K. Shtain. M.: Izdatel'skii dom "Vil'yams", 2005. 1296 s.
7. Nikolaev S.A. Metod vetvei i granits dlya poiska tekhnicheskikh reshenii. // Avtomatizirovannye podsistemy poiskovogo konstruirovaniya: Mezhevuzovskii sbornik. Gor'kii: GGU, 1981. S. 139-144.
8. Galochkin V. I. Perechislenie reshayushchikh derev'ev ogranichennoi stoimosti na I-II dereve // Programmnye sistemy i vychislitel'nye metody. 2014. № 1. S. 191 – 196. DOI: 10.7256/2305-6061.2014.2.11925.