

МОДЕЛЬ И АЛГОРИТМ КООРДИНАЦИИ РАБОТЫ КОММИВОЯЖЕРА

Ревотюк М. П., Кот О. В.

Кафедра информационных технологий автоматизированных систем, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: {rmp, kafitas}@bsuir.by

Рассматривается задача координации работы коммивояжера в случае изменения условий обслуживания, отражаемых изменениями элементов матрицы исходных данных. Предлагаемый быстрый алгоритм оценки устойчивости базируется на оценке устойчивости линейной задачи о назначении, соответствующей текущему решению задачи коммивояжера, что снижает вычислительную сложность координации.

I. ПОСТАНОВКА ЗАДАЧИ

В классической постановке формальная модель задачи коммивояжера имеет вид

$$Y = \min \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij},$$

$$u_i - v_j + n x_{ij} \leq n - 1, i, j = \overline{2, n}, i \neq j,$$

$$\sum_{i=1}^n x_{ij} = \sum_{j=1}^n x_{ij} = 1,$$

$$x_{ij} \geq 0, i, j = \overline{1, n}. \quad (1)$$

Ее решение обычно представлено вектором

$$R = (r_j = i \mid x_{ij} = 1, i, j = \overline{1, n}),$$

который отражает необходимые условия оптимальности

$$R = (r_j = i \mid c_{ij} = u_i + v_j, i, j = \overline{1, n}). \quad (2)$$

В реальных условиях с целью обеспечения нечувствительности к возмущениям часто необходимо найти интервалы (s_{ij}, f_{ij}) , $i, j = \overline{1, n}$, в которых изменение значений элементов $c_{ij} \in (s_{ij}, f_{ij})$, $i, j = \overline{1, n}$ не требует трудоемкого пересмотра плана работы коммивояжера. Известно, что задача оценки интервалов устойчивости задачи (1) в общем случае имеет экспоненциальную вычислительную сложность. Однако для частных случаев, когда изменяются лишь элементы матрицы с индексами $(i, j) = (r_j, j)$, $j = \overline{1, n}$, сложность оказывается полиномиальной.

II. СХЕМА РЕШЕНИЯ ЗАДАЧИ

Предлагаемая схема оценки интервалов устойчивости базируется на инвариантности выражения (2) от метода его формирования. Известно, что одним из точных методов решения (1) является метод ветвей и границ [1]. Схема алгоритма метода ветвей и границ может использовать разные способы порождения дерева вариантов. Наиболее успешный способ порождения

базируется на решении линейных задач о назначении (ЛЗН), анализе получающихся замкнутых циклов и, если таких циклов более одного, последующем переборе вариантов разрыва циклов. Рекурсия обхода дерева ЛЗН строится на матрице расстояний, где разрывы циклов задаются назначением бесконечных значений длин запрещаемых дуг [2]. В каждом узле дерева вариантов, включая и искомый оптимальный вариант, решается ЛЗН фиксированной размерности

$$Y = \min \sum_{i=1}^n \sum_{j=1}^n c_{ij}^* x_{ij},$$

$$\sum_{i=1}^n x_{ij} = \sum_{j=1}^n x_{ij} = 1,$$

$$x_{ij} \geq 0, i, j = \overline{1, n}, \quad (3)$$

где $(c_{ij}^*, i, j = \overline{1, n})$ – матрица текущей ЛЗН, в которой некоторые элементы исходной матрицы задачи (1) заменены бесконечными значениями. Очевидно, что элементы оптимального решения не меняются: $(c_{r_j j}^* = c_{r_j j}, j = \overline{1, n})$.

Отсюда следует, что задача оценки устойчивости задачи (1) может быть сведена к задаче оценки устойчивости решения задачи (3): для каждого элемента матрицы $(c_{ij}^*, i, j = \overline{1, n})$, используемой для формирования окончательного решения задачи (1), необходимо найти интервал (s_{ij}, f_{ij}) , $i, j = \overline{1, n}$, в котором изменение значения таких элементов не нарушает оптимального назначения.

Предлагаемая идея поиска интервалов устойчивости ЛЗН основана на том, что лучшие методы решения ЛЗН базируются на переходе от (3) к двойственной задаче линейного программирования

$$Y = \max \sum_{i=1}^n u_i + \sum_{j=1}^n v_j,$$

$$c_{ij} \geq u_i + v_j, i, j = \overline{1, n}. \quad (4)$$

Ключ для оценки интервалов устойчивости ЛЗН – векторы потенциалов $u_i + v_j, i, j = \overline{1, n}$.

III. АЛГОРИТМ ОЦЕНКИ УСТОЙЧИВОСТИ

Элементы решения (1) и (3) соответствуют ребрам графа совершенного паросочетания

$$E_m = \{(i, j) \mid x_{ij} = 1, i, j = \overline{1, n}\},$$

множество которых реально выделяется выражением (2):

$$E_m = \{(r_j, j), j = \overline{1, n}\}.$$

Оставшиеся элементы

$$E_u = \{(i, j), i, j = \overline{1, n}\} \setminus E_m$$

дополняют такой граф до полного. Интервал значений веса ребер, для которых назначение остается неизменным, может быть описан как

$$I_m(x, y) = \{(-\infty, c_{xy} + \epsilon_m(x, y)], (x, y) \in E_m\},$$

$$I_u(x, y) = \{[c_{xy} - \epsilon_u(x, y), +\infty), (x, y) \in E_u\},$$

где c_{xy} – вес, а $\epsilon_m(x, y)$ и $\epsilon_u(x, y)$ – допустимое изменение веса ребра $x \mapsto y$.

Интервалы I_m и I_u можно найти применением экономной разностной схемы реоптимизации текущего решения для каждого элемента матрицы [1]. Очевидно, что ребро $x \mapsto y$, $(x, y) \in E_u$, не будет частью существующего решения (скрыто) после назначения веса из интервала $(u_x + v_y, +\infty)$, где u_x и v_y – потенциалы строк и столбцов.

Отсюда следует алгоритм построения интервала значений веса ребра: установим гарантированно скрывающее ребро значение $c_{xy} = +\infty$, а после реоптимизации решения получим $\epsilon_m(x, y) = Y^1 - Y^0$. Здесь Y^1 и Y^0 – оценки (1) до и после скрытия ребра.

Обработка последствий скрытия ребра эффективно реализуется предложенным в [3] алгоритмом реоптимизации решения ЛЗН после изменения элемента матрицы исходных данных. Анализ алгоритмов решения ЛЗН показывает, что процесс реоптимизации, начинающийся в вершине x , завершится в вершине y , потенциал которой тоже не изменится. Меняется только потенциал строки x . Из выражения

$$Y = \sum_{i=1}^n u_i + \sum_{j=1}^n v_j,$$

следует, что изменение оценок решения оценивается выражением $Y^1 - Y^0 = u_x^1 - u_x^0$. Здесь нулевой верхний индекс использован для пометки исходного, а единичный – нового решения. В результате получим

$$I_m(x, y) = \{(-\infty, c_{xy} + u_x^1 - u_x^0], (x, y) \in E_m\}. \quad (5)$$

Использование разности значений потенциалов исключает необходимость наивного прямолинейного вычисления оценок решений задачи (1), требующего n шагов.

Аналогично рассуждая, можно рассмотреть ребра, не принадлежащие оптимальному паросочетанию. В этом случае $c_{xy} \geq u_x + v_y$. Если такие ребра имеют вес в интервале $(u_x + v_y, +\infty)$, то структура решения (1) остается неизменной. Однако для определения $\epsilon_u(x, y)$, $(x, y) \in E_u$ придется строить вспомогательный граф, образуемый из графа оптимального паросочетания путем удаления всех дуг, инцидентных вершинам x и y [2]. В таком графе будет $n - 2$ ребра графа оптимального паросочетания. Реализация алгоритма построения интервала значений веса ребер не является эффективной. Для каждого из $n(n - 1)$ ребер придется решать ЛЗН, размерность которых $n - 2$. Учитывая дискретный характер процесса перемещения по вершинам симплекса при реализации алгоритма решения ЛЗН, предлагается воспользоваться выражением (5), инвертируя направление шагов процесса построения интервала [3]. Конечная граница интервала I_m после этого станет начальной границей интервала I_u . Нулевой шаг в (7) становится решением ЛЗН для гарантированно приводящего к открытию ребра значения $c_{xy} = -\infty$, а единичный шаг соответствует решению ЛЗН с исходной матрицей. В результате получаем

$$I_u(x, y) = \{(-\infty + u_x^0 - u_x^1, +\infty), (x, y) \in E_u\}. \quad (6)$$

Существенно здесь то, что оценка выражений (5) и (6) требует лишь одношаговой реоптимизации исходной задачи (2).

IV. ЗАКЛЮЧЕНИЕ

Таким образом, выявление в процессе координации работы коммивояжера необходимости пересмотра плана работы требует лишь наблюдения за изменением исходных данных. Определение интервалов устойчивости задачи коммивояжера легко проводится посредством реоптимизации ЛЗН текущего оптимального решения, если инвертировать принадлежность дуг графа задачи соответствующему совершенному паросочетанию и учесть эту принадлежность направлением нумерации состояний. Вычислительная сложность оценок устойчивости на основе разности потенциалов изменяемых строк ЛЗН – $O(n^4)$. Дополнительная память для хранения наследуемых значений потенциалов строк не превышает объема $O(n^2)$.

V. СПИСОК ЛИТЕРАТУРЫ

1. Ревотюк, М. П. Реоптимизация решения задач о назначении / М. П. Ревотюк, П. М. Батура, А. М. Полоневич // Доклады БГУИР. – 2011. – № 1(55). – С. 55–62.
2. Lantao, L. Assessing optimal assignment under uncertainty: An interval-based algorithm / L. Lantao, A. S. Dylan // The International Journal of Robotics Research. – 2011. – Vol. 30(7). – P. 936–953.
3. Ревотюк, М. П. Быстрая оценка интервалов устойчивости решения линейных задач о назначении / М. П. Ревотюк, М. К. Кароли, П. М. Батура // Доклады БГУИР. – 2013. – № 5(75). – С. 30–36.