ОЦЕНКА УСТОЙЧИВОСТИ КРАТЧАЙШИХ ПУТЕЙ НА ДИНАМИЧЕСКИ ОПРЕДЕЛЯЕМЫХ ГРАФАХ

Ревотюк М. П., Бебех А. В., Хаджинова Н. В.
Кафедра информационных технологий автоматизированных систем,
Белорусский государственный университет информатики и радиоэлектороники
Минск, Республика Беларусь
E-mail: {rmp, kafitas}@bsuir.by

Предлагаются ускоренные процедуры оценки устойчивости решения задачи поиска кратчайших путей на динамических нагруженных орграфах, когда порядок порождаемых деревьев путей существенно меньше порядка графа. Под устойчивостью решения понимается его нечувствительность к независимым аддитивным возмущениям весов ребер графа. На основе метода бутстрэппинга показана возможность снижения сложности оценки устойчивости до линейной зависимости от объема сканируемого пространства.

I. Постановка задачи

Известно, что для поиска кратчайших путей на нагруженном ориентированном графе $G(N, A)$, где N – множество вершин, A – множество дуг с весовой функцией $W : A \to R^+$. лучшим является алгоритм Дейкстры [1]. Процесс построения дерева кратчайших путей имеет волновой характер до исчерпания возможности его развития из исходной вершины. В этом случае для каждой вершины x требуется представление множества непосредственно достижимых смежных вершин x', $x' = \{y|w(x, y) \geq 0\}$, где $w(x, y)$ – вес дуги $x \to y, x, y \in N$.

На практике возникает потребность поиска путей на графах с изменяемой структурой, когда описание графа задана легко модифицируемым списком дуг. Реляционная модель графа и пространства поиска решения – основа эффективной реализации поиска дерева путей методом бутстрэппинга[2]. Такой метод отличается ограничением потребности в памяти количеством фактически просматриваемых вершин без оценки предельных значений $|N|$, а также возможностью реализации открытых для расширения систем.

Анализ известных алгоритмов оценки устойчивости кратчайших путей [3,4], сложность которых $O(|A|log(|N|))$, показывает возможность их улучшения путем обработки фактически просматриваемых элементов графа.

Предмет обсуждения – развитие и обобщение приемов реализации метода бутстрэппинга [2] в задачах поиска и оценки устойчивости кратчайших путей на графах с переменной структурой.

II. Базовый алгоритм

Представим переменные состояния процесса построения дерева путей элементами отношения $T(x, d, p)$, где x – номер узла дерева (уникальный элемент), d – расстояние от корня s до узла x, p – номер предшествующего узла кратчайшего пути в узел x. По определению, $L(s)$ – проекция отношения $T(x, d, p)$ по атрибуту x.

Очевидно, что состояние процесса построения дерева на любом этапе k представлено тройкой $(x_k, d(x_k), p(x_k))$. Начальное состояние процесса построения дерева с корнем s соответствует тройке $(s, 0, s)$. Условие завершения процесса – $(L^* = \oslash)$ или $(x_k = t)$, если t – конечная вершина пути. В любом случае $k \leq |L(s)|$ после окончания поиска пути $s \to t$. Нумерация состояний в явном виде не потребуется, если в операции выборки очередного элемента x_k из L^*, обозначаемой далее как $L^k \ominus x_k$, контролировать условие завершения процесса.

Результат операции $L^k \ominus x_k$ – истинность условия $(L^k \neq \oslash)$, новое значение x_k в случае $(L^k \neq \oslash)$ и $k \leftarrow k + 1$. Так как по определению $d(x_k) \leq d(x_{k+1})$, то операция $'\ominus'$ требует упорядочения элементов $L(s)$ по ключу $d(x) \odot x$, где символ $'\odot'$ обозначает операцию конкатенации.

Операцию расширения дерева кратчайших путей обозначим символом $'\oplus'$. Результат операции $L^k \oplus x_k$ – истина, есди $x_k \notin L(s)$ и выполнено $L^k \leftarrow L^{k-1} \cup \{x_k\}$. Ее реализация требует упорядочения элементов $L(s)$ по ключу x.

Алгоритм Дейкстры поиска кратчайшего пути $s \to t$ в терминах определенных выше операций имеет вид:

```
function sp1(s,t) begin
    d(s) = 0; p(s) = s; L* ⊕ s;
    while (L* ⊖ x) ∧ (x ≠ t) do
        relax(x);
```

Здесь этап расширения дерева кратчайших путей из вершины x отражает функция

```
function relax(x) begin
    foreach (y ∈ x') do
        r = d(x) + w(x, y);
        if (L* ⊕ y) ∨ (d(y) > r) then
            d(y) = r; p(y) = x
```

Любая вершина x, к которой применялась операция $relax(x)$, становится узлом дерева кратчайших путей с постоянными значениями $p(x)$ и $d(x)$.

Дерево кратчайших путей – связный граф по определению. Если s – исходная вершина, а x – произвольный узел или лист дерева путей, $s, x \in N$, то после завершения поиска последовательность посещаемых вершин обратного пути $\{x, p(x), p(p(x)), \ldots, s\}$ представляет искомый кратчайший путь. Расширение дерева кратчайших путей происходит только из некоторого листа без пометки, а листья из множества $L(s) \setminus L^*$ представляют лишь исторический интерес и становятся пассивными. Обычно на практике $|L(s)| \ll |N|$, что объясняет привлекательность схемы бутстрэппинга.

III. Алгоритм оценки устойчивости решения

Рассмотрим вершину y, для которой на кратчайшем пути $s \to y, s, y \in N$. постоянно фиксируется расстояние $d(y)$ и $p(y)$. Изменение структуры такого пути возможно после изменения весов любой из входных дуг вершины y – $\{(x, y), x \in' y\}$. Здесь $'y = \{x | w(x, y) \geq 0\}$ — множество входных вершин дуг в вершину y, где $x, y \in N$.

Выходные дуги вершин из множества $'y$ в вершину y следует разделить на три категории:

последняя дуга кратчайшего пути в вершину y (дуга $(p(y) \to y)$;

дуги вне дерева кратчайших путей из вершин, принадлежащих такому дереву;

дуги из вершин, не рассматриваемых до момента постоянной пометки вершины y.

Дуги вне дерева кратчайших путей из вершин, принадлежащих такому дереву, можно выделить на графе в момент ветвления альтернатив развития дерева путей из каждой вершины из множества $'y$. Множество входных дуг вне дерева кратчайших путей, появляющихся после построения дерева, остаются без пометки вхождения соответствующих дугам инцедентных вершин в список L^*.

Обозначим интервал допустимого изменения веса дуги как $[w_{min}(x, y), w_{max}(x, y)]$ для каждой дуги $x \to y, x, y \in L(s)$. Далее, для удобства различения принадлежности интервалов, пометим верхним индексом '-' границы весов дуг дерева кратчайших путей, а индексом '+' – дуг вне дерева кратчайших путей.

Интервал $[w^+_{min}(x, y), w^+_{max}(x, y))$ допустимого изменения весов дуг $x \to y$, не входящих в дерево кратчайших путей, определяется выражениями

$$
\begin{aligned}
w^+_{min}(x, y) &= d(y) - d(x), \\
w^+_{max}(x, y) &= \infty, (x \in' y) \wedge (x \neq p(y)).
\end{aligned}
$$

Интервал $(w^-_{min}(x, y), w^-_{max}(x, y)]$ допустимого изменения весов дуг $x \to y$, входящих в дерево кратчайших путей, определяется выражениями

$$
\begin{aligned}
w^-_{min}(x, y) &= 0; \\
w^-_{max}(x, y) &= w(x, y) + \delta(y) - d(y), \\
\delta(y) &= min\{d(v) + w(v, y) | v \in' y)\}.
\end{aligned}
$$

Последние выражения определены относительно момента постоянной пометки очередной вершины x_k растущего дерева кратчайших путей (момента вызова операции $relax(x_k)$).

Нетрудно заметить, что процессы вычисления границ интервалов допустимого изменения весов дуг определены лишь на входных дугах очередной вершины. Такие процессы могут быть совмещены с целью однократного сканирования списка входных дуг вершины y:

```
function spr(y) begin
    δ = ∞; q = p(y);
    foreach (x ∈' y) ∧ (x ∈ L(s) \ L*) do
        δ = min(δ, d(x) + w(x, y));
        w⁺_min(x, y) = d(y) − d(x);
        w⁺_max(x, y) = ∞;
    w⁻_min(q, y) = 0;
    w⁻_max(q, y) = w(q, y) + δ − d(y);
```

Важно отметить, что условие $(x \in L(s) \setminus L^*)$ позволяет выделить лишь действительно влияющие на устойчивость пути дуги в момент постоянной пометки вершины y.

Очевидно, что оценка рассмотренных интервалов устойчивости может быть отложена. Этим можно воспользоваться для исключения холостых вычислений невостребованных данных. Например, возможны следующие варианты выполнениz оценки интервалов устойчивости:

после построения дерева кратчайших путей для всех его дуг;

на каждой итерации построения дерева кратчайших путей;

после построения дерева кратчайших путей для смежных дуг конкретных кратчайших путей.

Заключение

Таким образом, алгоритм оценки устойчивости кратчайших путей на на неоднородных динамически определяемых графах представлен в форме, согласованной с возможностями максимального сокращения вычислительной сложности методом бутстрэппинга.

Список литературы

1. Ferone, D. Shortest paths on dynamic graphs: a survey/D. Ferone [et al.]//Pesquisa Operacional, 2017. – Vol. 37, iss. 3. – P. 487–508.
2. Ревотюк, М. П. Встречный поиск кратчайших путей на больших динамических графах методом бутстрэппинга/М.П. Ревотюк, Н.В. Хаджинова //BIG DATA and Advanced Analytics: сб. материалов VI Междунар. науч.-практ. конф., Минск, 20–21 мая 2020 г. В 2 ч. Ч. 1. – Минск: Бестпринт, 2020. – С. 324–331.
3. Shier, D. R., Witzgall C. Arc tolerances in shortest path and network flow problem// Networks. –1980. – Vol. 10. – P. 277–291.
4. Tarjan, R. E. Sensitivity analysis in minimal spanning trees and shortest path trees//Inf. Proc. Letters. – 1982. – Vol. 14, iss. 1. – P. 30–33.