Адаптивное размещение ориентиров при маршрутизации в нестационарных сетях

Валентина Быкова, Александр Солдатенко

Сибирский федеральный университет,
пр. Свободный, 79, г. Красноярск, 660041, Россия

Аннотация Рассмотрена задача Time-Dependent Shortest-Path, расширение задачи о кратчайшем пути в графе, когда вес дуги графа является функцией от времени отправления из начальной вершины этой дуги. Такой граф называют нестационарной сетью. Предложена модификация алгоритма ALT (A* with Landmarks & Triangle), осуществляющая целенаправленный поиск пути в нестационарной сети по ориентирам. Ориентиры устанавливаются с помощью адаптивной стратегии. Приведены итоги экспериментов.
Ключевые слова: маршрутизация, кратчайший путь, алгоритм ALT, ориентиры, эвристика, граф большой размерности.

В современных приложениях часто приходится иметь дело с расширением задачи о кратчайшем пути, когда время прибытия в конечную вершину y дуги $e = (x, y)$ исходного ориентированного графа $G = (V, E)$ зависит не только от веса дуги e, но и времени выхода из начальной вершины x этой дуги. Функции, заданные подобным образом на дугах графа G, называют функциями прибытия, а сам граф G – нестационарной сетью. Задача поиска кратчайшего пути в нестационарной сети носит название Time-Dependent Shortest-Path problem или коротко TDSP [1, 2]. Например, TDSP возникает при передвижении на автомобиле в час пик по городским дорожным сетям в условиях пробок. Подобные ситуации считаются предсказуемыми и описываются через функции прибытия. Известно, что TDSP для нестационарной сети общего вида без каких-либо ограничений на топологию сети и функции прибытия является NP-трудной задачей [1]. В случае, когда функции прибытия удовлетворяют условию FIFO (или, то же самое, условию монотонности), то задача TDSP полиномиально разрешима [2, 3].

Введем необходимые обозначения и сформулируем TDSP. Пусть в ориентированном графе $G = (V, E)$ для всякой дуги $e = (x, y) \in E$ весовая функция $w_{xy}(t) \geq 0$ определена формулой:

$$w_{xy}(t) = \frac{l_{xy}}{v_{xy}(t)}, \tag{1}$$

In: A. Kononov et al. (eds.): DOOR 2016, Vladivostok, Russia, published at http://ceur-ws.org

где l_{xy} – расстояние между вершинами $x, y \in V$, $v_{xy}(t)$ – скорость движения по дуге $(x, y) \in E$, при условии, что старт из вершины x выполнен в момент времени t. Считается, что $v_{xy}(t) > 0$ и расстояния между вершинами являются положительными постоянными величинами, удовлетворяющими неравенству треугольника. Нестационарную сеть $G = (V, E)$ с весовыми функциями (1), расстояние между вершинами которой удовлетворяют неравенству треугольника, называют нестационарной метрической сетью.

Уточним условие FIFO для нестационарной сети. Определим для дуги $(x, y) \in E$ функцию прибытия $F_{xy}(t)$ по формуле:

$$F_{xy}(t) = t + w_{xy}(t),$$

где t – время отправления из x при движении в y по дуге (x, y) с весом $w_{xy}(t)$. Если для каждой дуги $(x, y) \in E$ и любых $0 < t_1 \leq t_2$ верно неравенство $F_{xy}(t_1) \leq F_{xy}(t_2)$, то говорят, что функция прибытия удовлетворяет условию FIFO. Поскольку $t > 0$ и $w_{xy}(t) \geq 0$ для любой дуги $e = (x, y) \in E$, то

$$F_{xy}(t) \geq t \text{ для всех } e \text{ и } t. \tag{2}$$

Путь из вершины s в вершину d в графе $G = (V, E)$ определяется как последовательность вершин $P = \{x_0, x_1, x_2, \ldots, x_k\}$, в которой $s = x_0, d = x_k$,
$(x_{i-1}, x_i) \in E$ для $i = 1, 2, \ldots, k$. Если в G существует (s, d)-путь P, то вершина d достижима из s по пути P, что записывается как $s \underset{P}{\longmapsto} d$. Весом пути P называется сумма весов всех дуг, входящих в этот путь и обозначается через $w(P)$. Традиционно, когда веса дуг постоянные, под весом кратчайшего (s, d)-пути понимается величина

$$dist(s, d) = \min_P \{w(P) \colon s \underset{P}{\longmapsto} d\}.$$

При этом путь P, на котором достигается минимум, называется кратчайшим (s, d)-путем [4].

Для нестационарного случая (s, d)-путь $P = \{x_0, x_1, x_2, \ldots, x_k\}$, старт которого осуществляется в момент времени t_s, обозначается через (s, d, t_s) или (P, t_s). Вес этого пути вычисляется по формуле:

$$w(P, t_s) = t_s + \sum_{i=0}^{k-1} w_{x_i x_{i+1}}(t_i), t_0 = t_s, t_{i+1} = F_{x_i x_{i+1}}(t_i), \tag{3}$$

а вес кратчайшего (s, d, t_s)-пути определяется следующим образом:

$$dist(s, d, t_s) = \min_P \{w(P, t_s) \colon s \underset{P}{\longmapsto} d\}. \tag{4}$$

Заметим, что значения величин $w(P, t_s)$ и $dist(s, d, t_s)$ всегда могут быть вычислены по формулам (3) и (4), если вершина d достижима из вершины s в графе G. Величину $w(P, t_s)$ можно трактовать как время прибытия в вершину d при прохождении (s, d, t_s)-пути, а величину $dist(s, d, t_s)$ – как самое раннее время прибытия в d при условии, что отправление из вершины s осуществлено в момент времени t_s. Тройку (s, d, t_s), где s и d – стартовая и целевая вершина соответственно, а t_s – стартовое время, будем называть запросом на поиск кратчайшего пути, или кратко (s, d, t_s)-запросом.

Задача TDSP формулируется следующим образом.

Time-Dependent Shortest-Path problem
Заданы: нестационарная метрическая сеть $G = (V, E)$ с условием FIFO, (s, d, t_s)-запрос.
Требуется: найти значение $dist(s, d, t_s)$ и последовательность вершин, образующих кратчайший (s, d, t_s)-путь.

Точное решение данной задачи может быть найдено с помощью алгоритма Дейкстры и алгоритма ALT. Алгоритм Дейкстры находит точное решение при выполнении неравенства (2) и условия FIFO [3]. Справедливость неравенства треугольника для расстояний между вершинами сети G гарантирует корректность алгоритма ALT [5].

Отметим особенности алгоритма ALT для стационарного случая. Алгоритм ALT осуществляет целенаправленный поиск по ориентирам. При этом под ориентиром понимается выделенная вершина входного графа. Алгоритм ALT состоит из двух фаз: на первой фазе выполняется расстановка ориентиров и вычисление потенциальных функций, на второй фазе осуществляется поиск кратчайшего (s, d)-пути с помощью алгоритма A* (эквивалента алгоритма Дейкстры, работающего с потенциальными функциями). Роль потенциальных функций — оценка снизу значения кратчайшего пути от текущей вершины v до целевой вершины d. Напомним, что традиционно в алгоритме Дейкстры поиск кратчайшего (s, d)-пути осуществляется лишь на основе верхнего оценивания значения (s, v)-пути. Алгоритм ALT базируется на алгоритме Дейкстры, поэтому время его выполнения на графе $G = (V, E)$ составляет $O(|V|^2)$ [5, 6].

Действенность потенциальных функций во многом зависит от расположения ориентиров относительно вершин s и d. Стремление оптимально расставить ориентиры в графе приводит к задаче выбора ориентиров, которая заключается в определении числа ориентиров и мест их расстановки в вершинах графа, минимизирующих пространство поиска алгоритма Дейкстры для последовательности (s, d)-запросов в заданном графе. К сожалению, эта задача труднорешаемая, так как доказано, что в частном случае, когда число ориентиров фиксировано, она NP-трудная [6]. Поэтому для еч решения на практике обычно прибегают к полиномиальным эвристикам. На сегодняшний день для расстановки ориентиров в алгоритме ALT используются различные эвристики, размещающие ориентиры в вершинах графа случайным образом [5, 6]. Основной их недостаток состоит в том, что они не учитывают особенности каждого отдельного (s, d)-запроса и не используют информацию об истории процесса обработки запросов.

Предлагается полиномиальная по времени эвристика, лишенная этих недостатков. Суть эвристики состоит в следующем. Перед выполнением первого (s, d)-запроса выполняется случайная расстановка заданного числа $K = |L|$ ориентиров в вершинах графа. Далее выбранное множество ориентиров L периодически частично обновляется через каждые Δ запросов на основе рейтинга ориентиров. Рейтинг ориентиров поддерживается следующим образом: всякий раз, когда ориентир предлагает наилучшую оценку пути среди всего набора ориентиров, он получает очко. Кроме того накапливается краткая история процесса обработки запросов. Как только завершается выполнение $(\Delta - 1)$-го запроса, начинается процедура обновления ориентиров. Ориентир с наименьшим рейтингом подлежит замене на вершину, которая наиболее удалена от текущего набора ориентиров. После обновления множества L статистика об эффективности ориентиров начинает формироваться заново.

В случае нестационарной сети алгоритм ALT и предлагаемая эвристика работают аналогичным образом, однако вычисление потенциальных функций направлено на самые благоприятные условия, когда пробок нет, т. е. при весах дуг $w_{xy} = \min_t\{w_{xy}(t)\}, \forall(x, y) \in E$.

В предложенной модификации алгоритм ALT реализован в виде программы на C++. С помощью данной программы были выполнены вычислительные эксперименты. Эксперименты показали, что частоту обновления для больших графов разумно полагать $\Delta \approx 20$, а число ориентиров K целесообразно брать в диапазоне $9 \leq K \leq 16$. Превышение верхней границы диапазона резко увеличивает время предобработки и не обеспечивает должного ускорения.

Алгоритм был также проверен на реальных дорожных сетях, представленных в базе данных DIMACS [7]. Некоторые результаты сравнения показателей его работы с алгоритмом Дейкстры даны в табл. 1, где C_1 – коэффициент сокращения пространства поиска, C_2 – коэффициент ускорения, определяемый как отношение времени работы сравниваемых между собой алгоритмов, T – среднее время выполнения одного запроса. Тестирование осуществлялось на случайно сгенерированной последовательности из 1000 запросов. Эксперименты выполнялись на компьютере с процессором Intel® Core ™i7-720QM Processor (6M Cache, 1.60 GHz) и ОЗУ размером 4 ГБ.

Таблица 1. Результаты сравнения алгоритмов

Граф $G = (V, E)$ в базе DIMACS	$\lvert V\rvert$	$\lvert E\rvert$	C_1	C_2	T, *сек*
Rome99	3353	8870	8,39	3,77	0,00052
NY	264346	744846	14,48	3,27	0,41
COL	435666	1057066	19,7	1,45	0,62

Результаты экспериментов показали, что предложенная модификация алгоритма ALT на порядок сокращает размер пространства поиска кратчайшего пути в нестационарной сети и обеспечивает тем самым значительное ускорение алгоритма Дейкстры.

Список литературы

1. Sherali, H. D., Ozbay, K., Subramanian, S.: The time-dependent shortest pair of disjoint paths problem: Complexity, models, and algorithms. Networks. 31 (4), 259–272 (1998)

2. Kaufman, D. E., Smith, R. L.: Fastest paths in time-dependent networks for intelligent vehicle-highway systems application. J. of Intelligent Transportation Systems. 1 (1), 1–11 (1993)
3. Гимади, Э. Х., Глебов Н. И.: Математические модели и методы принятия решений. Новосибирск: Издательство НГУ (2008)
4. Емеличев, В. А., Мельников, О. И., Сарванов, В. И., Тышкевич, Р. И.: Лекции по теории графов. М.: Книжный дом «Либроком» (2012)
5. Goldberg, A., Kaplan, H., Werneck, R.: Reach for A*: Efficient point-to-point shortest path algorithms. Technical Report MSR-TR-2005-132, Microsoft Research. Microsoft Corporation (2005)
6. Goldberg, A., Kaplan, H., Werneck, R.: Better landmarks within reach. In Proc. International Workshop on Experimental Algorithms (WEA). 4525 of LNCS. Springer, 38–51 (2007)
7. DIMACS Implementation Challenge — Shortest Paths [Online]. Available: http://www.dis.uniroma1.it/challenge9/download.shtml [2016, 29 March]

Adaptive Landmark Selection when Routing in Time-dependent Network

Valentina Bykova, Alexander Soldatenko

Siberian Federal University,
Svobodny Ave. 79, Krasnoyarsk, 660041, Russian Federation

Аннотация The Time-Dependent Shortest-Path problem is extension of shortest path problem in the graph when the graph arc weight is a function of the departure time from the initial vertex of the arc. Such graph is called a time-dependent network. We propose a modification of the algorithm ALT (A* with Landmarks & Triangle), carrying out goal-directed search of way in the network using landmarks. Landmarks are set using adaptive strategy. Presents the results of experiments.

Keywords: routing, shortest path, ALT algorithm, landmarks, heuristic, big graphs.

In: A. Kononov et al. (eds.): DOOR 2016, Vladivostok, Russia, published at http://ceur-ws.org