

Задача поиска графа-паттерна на помеченном графе

Мурадян Илья Валерьевич

Прикладная математика и информатика
Кафедра алгебры и дискретной математики

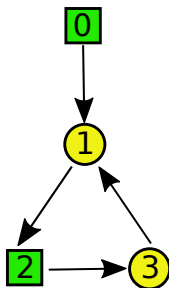
Научный руководитель: доцент, к.ф.-м.н Скороходов Владимир Александрович

Июнь 2018

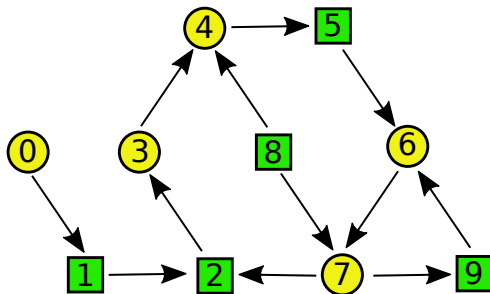
- 1 Проанализировать алгоритмы поиска паттернов на помеченных графах из статьи «Towards Practical and Robust Labeled Pattern Matching in Trillion-Edge Graphs».
- 2 Модифицировать алгоритмы для решения более общей задачи.
- 3 Реализовать разработанные алгоритмы.

Задача поиска графа-паттерна на помеченном графе

Задача: найти частичный подграф архивного графа G , изоморфный графу-паттерну G' .



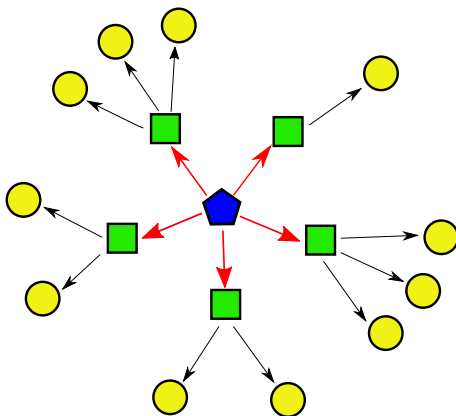
Граф-паттерн G'



Архивный граф G

Почему модификация необходима?

- 1 Поиск симметричных графов-паттернов при обработке изображений.
- 2 Поиск иерархических структур в графах социальных взаимодействий.



Общий алгоритм исключения кандидатов

Отображение кандидатов $f : V' \rightarrow 2^V$ сопоставляет вершинам графа-паттерна возможно соответствующие им вершины архивного графа.

Вход: графы $G(V, E)$, $G'(V', E')$

начало Exclusion

Инициализировать отображение кандидатов f

до тех пор, пока вершины продолжают исключаться **выполнять**

$f := LocalConstraintsChecking(G, G', f, N)$

$f := CycleChecking(G, G', f)$

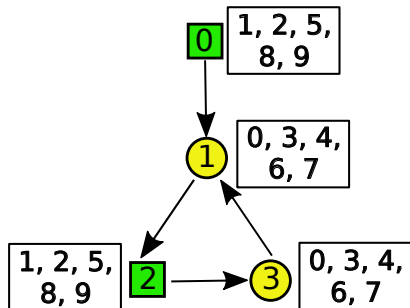
вернуть f

Алгоритм заключается в исключении как можно большего числа несоответствующих вершин из множеств $f(v')$.

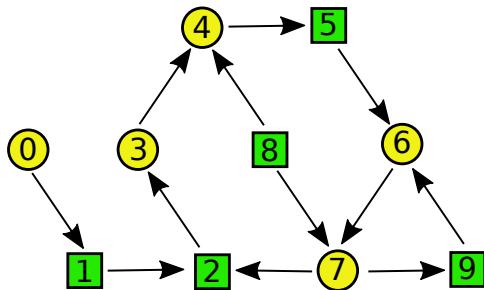
При инициализации каждое из множеств $f(v')$ инициализируется теми вершинами v , которые имеют такие же метки, как и вершина v' :

$$f(v') = \{v \in V \mid l(v) = l(v')\}$$

Пример: инициализация



Граф-паттерн G'



Архивный граф G

Алгоритм исключения по локальным условиям

Вход: $G(V, E)$, $G'(V', E')$, f , число итераций N

начало LocalConstraintsChecking

для $i = 1, 2, \dots, N$:

для $(q_0, q) \in E'$:

для $v_0 \in f(q_0)$:

$flag := False$

для $v \in \Gamma(v_0)$:

если $v \in f_K(q)$ и $\rho(\chi(v_0, v), \chi'(q_0, q)) = 1$ **тогда**

$flag := True$

если $flag = False$ **тогда**

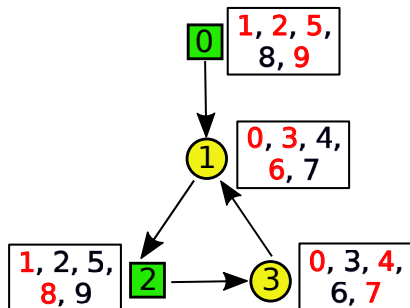
 Исключить q_0 из $f(v_0)$

 Аналогично – для обратных дуг

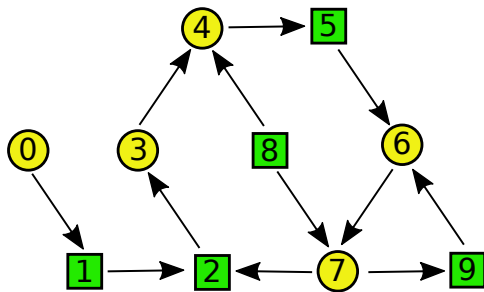
вернуть f

Сложность: $O(|E'| \cdot |V|^2)$.

Пример: шаг 1, алгоритм исключения по локальным условиям



Граф-паттерн G'



Архивный граф G

Алгоритм проверки контуров

\mathcal{K}_0 – множество проверяемых контуров.

Вход: $G(V, E)$, $G'(V', E')$, f

начало CycleChecking

для каждого $C_0 \in \mathcal{K}_0$ **выполнять**

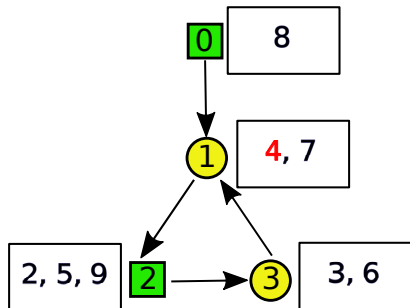
 Пусть (v'_0, v'_1) – первая дуга контура C_0 .

 Исключить из множества $f(v'_0)$ те вершины, в которых не
 начинается контур, соответствующий C_0 .

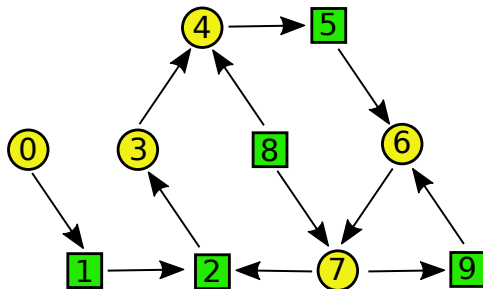
вернуть f

Сложность: $O(|\mathcal{K}_0| \cdot |V|^3 \cdot |E'|)$.

Пример: шаг 1, алгоритм проверки контуров

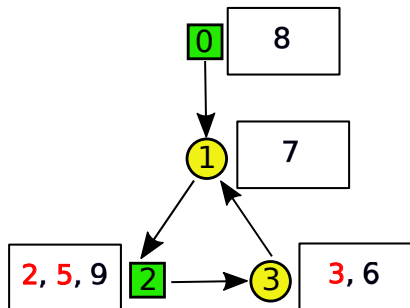


Граф-паттерн G'

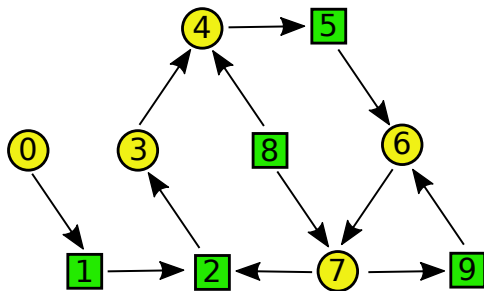


Архивный граф G

Пример: шаг 2, алгоритм исключения по локальным условиям

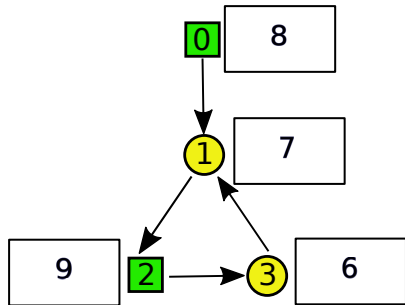


Граф-паттерн G'

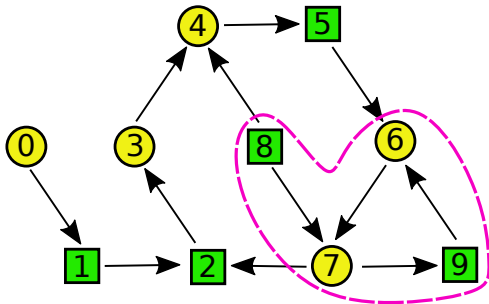


Архивный граф G

Пример: окончательный результат



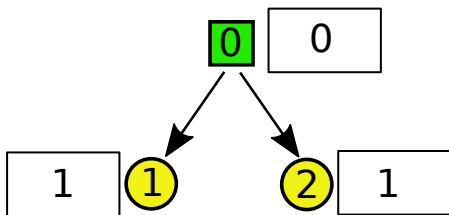
Граф-паттерн G'



Архивный граф G

Возможные проблемы

Применение общего алгоритма не гарантирует исключения всех кандидатов. В частности, в нижеприведённом случае не происходит исключения несоответствующих вершин.



Граф-паттерн G'



Архивный граф G

Решения: проверить степени вершин при инициализации, применить алгоритм выделения взвешенного совпадения в конце.

Алгоритм выделения взвешенного совпадения

- ❶ Преобразовать граф-паттерн G' к особому виду, снабдив его дополнительной информацией, получив таким образом граф G_0 .
- ❷ Положить $i := 0$.
- ❸ Пока граф G_i имеет более одной вершины:
 - ❶ Выполнить на графе G_i обход в ширину с окраской.
 - ❷ Выделить подграфы графа G_i с вершинами одного цвета.
 - ❸ На каждом выделенном подграфе $G_i^j(V_i^j, E_i^j)$ построить все возможные соответствия между его вершинами и вершинами архивного графа.
 - ❹ Сформировать новый граф G_{i+1} , взяв в качестве вершин подграфы G_i^j , а в качестве дуг – дуги, вершины которых принадлежат сразу нескольким подграфам.
 - ❺ Положить $i := i + 1$.

- 1 Модифицированы и описаны алгоритмы исключения по локальным условиям и проверки контуров.
- 2 Разработан алгоритм выделения взвешенного совпадения.
- 3 Разработанные алгоритмы реализованы, репозиторий:
https://github.com/ileasile/diploma/tree/master/python_graph_pattern_project