Изоморфное вложение графов
Пересечение графов

Изоморфное вложение графов

- Скажем, что граф $G_1 = \langle V_1, E_1 \rangle$ вкладывается в граф $G_2 = \langle V_2, E_2 \rangle$, если существует инъективное отображение $\varphi: V_1 \to V_2$, такое, что:
$$\{v, u\} \in E_1 \Rightarrow \{\varphi(v), \varphi(u)\} \in E_2$$
- Будем обозначать факт вложимости графа G_1 в граф G_2 тем же символом, что используется для обозначения вложимости множеств: $G_1 \subseteq G_2$
- Если необходимо подчеркнуть, что вложение осуществляется отображением φ, будем писать
$$G_1 \underset{\varphi}{\subseteq} G_2$$

Изоморфное вложение графов

- Задача: для данных графов $G_1 = \langle V_1, E_1 \rangle$ и $G_2 = \langle V_2, E_2 \rangle$ определить, вкладывается ли граф G_1 в граф G_2
- Ясно, что, если $|V_1| > |V_2|$, вложение неосуществимо, поскольку в таком случае не существует инъективного отображения из $|V_1|$ в $|V_2|$
- Будем считать, что $V_1 = \{1,2, \dots, n\}$, $V_2 = \{1,2, \dots, m\}$
- Задача сводится к построению вектора $\langle x_1, x_2, \dots, x_n \rangle$, все элементы которого различны и принадлежат V_2, такого, что

$$\{i, j\} \in E_1 \Rightarrow \{x_i, x_j\} \in E_2$$

- Этот вектор может быть построен процедурой, абсолютно аналогичной процедуре генерации перестановок

Сокращение перебора

- Как и в случае с задачей определения изоморфизма графов, возможно сокращение перебора за счет использования специальных функций
- Например:
  - Если $\deg_{G_1}(u) > \deg_{G_2}(v)$, то вершина u не может быть отображена в вершину v, т.к. обязательно будет потеряно хотя бы одно ребро
  - Если количество путей заданной длины, проходящих через вершину u в графе G_1, больше количества путей той же длины, проходящих через вершину v в графе G_2, то вершина u также не может быть отображена в вершину v
  - …

Сокращение перебора

- Кроме того, необходимо осуществлять проверку основного свойства вложения: если $\langle x_1, x_2, \dots, x_{k-1} \rangle$ - текущий вектор, x_k - вершина-кандидат, то нужно проверить условия
$$\{i, k\} \in E_1 \Rightarrow \{x_i, x_k\} \in E_2, 1 \leq i < k$$
- Если хотя бы одно из них не выполнено, x_k не может стоять на k-й позиции

Пересечение графов

- Скажем, что граф $H = \langle V_H, E_H \rangle$ является общим фрагментом графов $G_1 = \langle V_1, E_1 \rangle$ и $G_2 = \langle V_2, E_2 \rangle$, если $H \subseteq G_1$ и $H \subseteq G_2$
- Нас будет интересовать поиск общих фрагментов, обладающих экстремальными свойствами
- Мы будем рассматривать задачу поиска общего фрагмента двух графов, максимальному по числу ребер, т.к. задача поиска общего фрагмента, максимального по числу вершин, решается тривиально

Пересечение графов

- Будем считать, что $|V_1| \leq |V_2|, V_1 = \{1,2,\dots,n\}$, $V_2 = \{1,2,\dots,m\}$
- Задача сводится к построению вектора $\langle x_1, x_2, \dots, x_n \rangle$, все элементы которого различны и принадлежат V_2. Если такой вектор построен, он соответствует общему фрагменту – графу
$$G = \langle V_1, E' \rangle, \text{где}$$
$$E' = \left\{ \{i,j\} \in E_1 \middle| \left\{x_i, x_j\right\} \in E_2 \right\}$$
- Этот вектор также можно строить на основе процедуры генерации перестановок

Сокращение перебора

- Для сокращения перебора в этой задаче используют различные способа вычисления оценки количества ребер, которые войдут в пересечение
- Предположим, уже построено пересечение, содержащее K ребер, $\langle x_1, x_2, \dots, x_k \rangle$ - текущий вектор
- Легко вычислить количество ребер, входящих в построенный на настоящий момент общий фрагмент:

$$K_1 = \left| \left\{ \{i, j\} \in E_1 \middle| \left\{x_i, x_j\right\} \in E_2 \right\}_{1 \le i,j \le k} \right|$$

Сокращение перебора

- Предположим, известна некоторая верхняя оценка на количество ребер, которые могут быть добавлены в уже построенный фрагмент в дальнейшем, и она равна K_2
- Если $K < K_1 + K_2$, то продолжение перебора для рассматриваемого вектора имеет смысл, иначе его необходимо остановить

Вычисление оценок

- Легко придумать простые способы вычислить оценку K_2, например:
$$K_2 = \{\{i, j\} \in E_1 | 1 \leq i \leq k, 1 \leq j \leq n\} -$$
- оценка, равная количеству ребер, не входящих в подграф графа G_1, индуцированный множеством вершин $\{1,2, \ldots, n\}$
- Мы рассмотрим более содержательный способ построения верхней оценки – метод Мак-Грегора

Метод Мак-Грегора

- Всякое ребро $\{v, u\} \in E_1$, которое потенциально возможно сохранить, расширяя текущий вектор отображения, можно отнести к одному из трех типов:
  - $v, u \in \{1,2, \dots, k\}$
  - $v, u \notin \{1,2, \dots, k\}$
  - Только одна из вершин, v или u, принадлежит множеству $\{1,2, \dots, k\}$
- Ясно, что

Метод Мак-Грегора

- Ясно, что ребро некоторого типа может быть отображено только в ребро соответствующего типа, т.е.:
  - Если $v, u \in \{1, 2, \ldots, k\}$, то его можно отобразить только в ребро, оба конца которого принадлежат множеству $\{x_1, x_2, \ldots, x_k\}$
  - Если $v, u \notin \{1, 2, \ldots, k\}$, то его можно отобразить только в ребро, оба конца которого не принадлежат множеству $\{x_1, x_2, \ldots, x_k\}$
  - Если только одна из вершин, v или u, принадлежит множеству $\{1, 2, \ldots, k\}$, то ребро можно отобразить только в ребро, один конец которого принадлежит множеству $\{x_1, x_2, \ldots, x_k\}$, а другой нет

Метод Мак-Грегора

- Метод заключается в подсчете в обоих графов ребер каждого из типов. Пусть в графе G_1 имеется l_1 ребро первого типа, l_2 ребер второго типа и l_3 ребер третьего типа, а в графе G_2 – имеется k_1 ребро первого типа, k_2 ребер второго типа и k_3 ребер третьего типа
- Тогда верхняя оценка на количество ребер в общем фрагменте будет равна $K_2 = \min(l_1, k_1) + \min(l_2, k_2) + \ldots + \min(l_3, k_3)$