МЕРА ВАЖНОСТИ НАУЧНОЙ ПЕРИОДИКИ — „ЦЕНТРАЛЬНОСТЬ ПО ПОСРЕДНИЧЕСТВУ“

С. В. Бредихин, В. М. Ляпунов, Н. Г. Щербакова

Институт вычислительной математики и математической геофизики СО РАН,
630090, Новосибирск, Россия

УДК 001.12+303.2

Предметом изучения является сеть цитирования, отражающая структуру распределенной библиографической базы данных *RePEc*. Рассмотрены алгоритмы поиска кратчайших путей и вычисления меры „центральность по посредничеству“. Разработан алгоритм вычисления этой меры для взвешенных графов. Выполнено ранжирование коллекции периодических изданий БД на основе „центральности по посредничеству“.

Ключевые слова: сети цитирования, алгоритмы поиска кратчайших путей, мера „центральность по посредничеству“, алгоритм Брандеса.

Subject of studying is the citation network reflecting structure of the distributed bibliographic database *RePEc*. Shortest-path algorithms and betweenness centrality computing algorithms are reviewed. Betweenness centrality computing algorithm for weighted graphs is developed. *RePEc* journal ranking based on that measure is executed.

Keywords: citation networks, shortest path algorithms, betweenness centrality measure, Brandes algorithm.

Введение. В начале этого века в результате разработки и широкого внедрения языка разметки *XML* [1] удалось формализовать и „поставить на конвейер“ процесс массивной оцифровки документов и репродуцирования цитат. Появилась возможность „добывать“ библиометрическую информацию в „промышленных“ масштабах и хранить ее в удобных цифровых форматах. Современные библиографические базы данных (далее, ББД) являются основой для проведения исследований в области эволюции и динамики научной деятельности, структуры цитирования и последующего анализа „важности“ конкретных журналов и статей. Наборы библиографических данных впервые были проанализированы в работе [2] с целью количественной оценки индивидуальной производительности ученого. Эмпирическим путем было установлено, что число авторов, опубликовавших n статей, с точностью до константы, зависящей от научной дисциплины, равно $1/n^{\alpha}$ ($\alpha \sim 2$). После выхода в свет новаторской работы [3] стало ясно, что библиографические данные имеют естественное математическое представление в терминах графов. Прайс показал, что внутреннюю структуру связей между цитируемыми и цитирующими публикациями можно представить в виде ориентированного графа. Пусть рассматривается коллекция D, состоящая из n публикаций. Если документ d_j цитирует (имеет ссылку на) документ d_i, то акт цитирования можно изобразить в виде стрелки, идущей от вершины, представляющей d_j, к вершине, представляющей d_i. Таким образом, рассмотрев все взаимосвязи документов коллекции D, мы будем иметь граф, названный „графом цитирования“ или „сетью цитирования“ (далее СЦ). Матрица смежности графа размерности $n \times n$, такая что элемент этой матрицы $c_{ij} = 1$, если публикация d_j процитировала публикацию d_i названа „матрицей

цитирования“ (с точностью до транспонирования). Для преобразования ориентированной СЦ в неориентированную, как правило, рассматриваются отношения библиографического сочетания [4] или коцитирования [5, 6].

Определение СЦ можно распространить на случай, когда рассматриваются взаимные цитирования между научными журналами, авторами публикаций, исследовательскими институтами или другими сообществами. Согласно [7], будем называть объекты сети, взаимосвязи между которыми рассматриваются, „акторами“. Заметим, что сеть, акторами которой являются публикации, моделируется с помощью ориентированного ациклического графа без кратных ребер (пары вершин не могут соединяться более чем одним ребром). Если акторами являются журналы или авторы статей, граф имеет более сложную структуру.

В нашем случае в качестве акторов СЦ выступают научные журналы. На этапе построения сети фиксируются временные интервалы: окно цитирования (период, к которому относятся цитирующие работы) и окно публикации (период, к которому относятся цитируемые работы). Журнал представляет собой единый контент, состоящий из множества статей. Будем рассматривать матрицу цитирования C, такую, что $c_{ij} = 1$, если публикация из журнала j за интервал цитирования процитировала работу/работы из журнала i, относящиеся к окну публикации, иначе $c_{ij} = 0$. В этом случае сеть моделируется невзвешенным ориентированным графом. Однако при этом теряется важная информация о „силе связи“ между акторами, поэтому рассмотрим „взвешенную матрицу цитирования“, в которой c_{ij} соответствует количеству цитирований, $cit(i, j)$, полученных публикациями журнала i от публикаций журнала j за соответствующие периоды, а сеть моделируется взвешенным ориентированным графом. Отметим, что эта матрица может также рассматриваться как матрица смежности мультиграфа, когда c_{ij} — это количество дуг, идущих из j в i. В работе [8] показано, что в ряде случаев взвешенные графы можно анализировать с помощью перехода к невзвешенным мультиграфам, однако нельзя утверждать, что этот прием годен во всех случаях.

На основе СЦ можно вычислять различные полезные параметры, которые отражают свойства сетевой топологии. Многие из них были разработаны для анализа социальных сетей и, тем не менее, используются вне этой области, образуя важную часть базовых сетевых понятий. Одним из средств сетевого анализа является определение „важных“ сетевых узлов, основанное на их взаимосвязанности. Формально мера, оценивающая взаимосвязанность узла в графе, так называемая „центральность“, это отображение вершин графа на множество неотрицательных вещественных чисел, такое, что большее значение означает „лучшую“ взаимосвязанность с другими вершинами. Существует несколько способов определять взаимосвязанность и, соответственно, индексы центральности, которые способны ранжировать акторов согласно их позиции в сети [9]. Простейший способ оценки центральности вершины неориентированного графа заключается в подсчете числа соединенных (*incident*) с ней ребер. Несмотря на простоту, эта мера, называемая „центральность по степени“ (*degree centrality*), является достаточно информативной. Например, в случае социальных сетей, актор, имеющий больше связей, явно имеет большее влияние в группе. Для ориентированного графа различают степень вершины по входу (*in-degree*) — количество входящих дуг — и степень по выходу (*out-degree*) — количество выходящих дуг. Количество цитирований, которое публикация получает от других публикаций, используется как грубая оценка важности публикации, а это степень вершины по входу для сети цитирования.

В данной работе рассматривается мера „центральность по посредничеству“ (*betweenness centrality*, далее C_B), говорящая о том, насколько часто рассматриваемая вершина графа лежит на путях между другими вершинами. Ее классическое определение, представленное в работе [10], основано на идее передачи сообщений между вершинами информационной сети по кратчайшему (геодезическому) пути, выбранному случайным образом. Вершины, через которые проходит большее количество геодезических путей, имеют больший индекс центральности, они могут обладать значительным влиянием, так как в этих вершинах можно, например, контролировать проходящую информацию. Удаление таких узлов может разрушить коммуникации между другими узлами. Центральность по посредничеству отличается от других определений центральности, поскольку интерес представляет не то, как вершина взаимосвязана с другими вершинами, а то, как часто вершина встречается на пути между другими.

Предметом настоящей работы является вычисление значения C_B для вершин СЦ, образованной распределенной базой библиографических данных (далее РББД) *RePEc* [11], с целью их ранжирования по этой мере. Под термином ранжирование будем понимать установление такого отношения между любыми двумя вершинами СЦ, при котором можно утверждать, что „одна вершина имеет ранг выше другой“ или „... ниже другой“ или „их ранги равны“; при этом само множество „рангов“ является упорядоченным. Отметим, что с помощью процедуры ранжирования нельзя однозначно выполнить упорядочивание вершин, поскольку СЦ может иметь вершины с одинаковым рангом.

1. Определения. Пусть $G = (V, E)$ — связный неориентированный граф и w — функция взвешивания ребер, такая что $w(e) > 0$, $e \in E$ для взвешенных графов и $w(e) = 1$ для невзвешенных графов. Определим путь от вершины $s \in V$ до вершины $t \in V$ как последовательность несовпадающих вершин и ребер $(v_1, e_1, v_2, e_2, \ldots, e_{n-1}, v_n)$, начинающуюся в s $(s = v_1)$ и заканчивающуюся в t $(t = v_n)$, такую что каждое ребро соединяет предшествующую вершину с последующей. Если в графе нет кратных ребер, то путь можно представлять в виде последовательности вершин. Длина пути p — это сумма весов его ребер:

$$w(p) = \sum_{i=1}^{n-1} w(e_i).$$

Расстояние между вершинами s и $t, d_G(s, t)$ — это минимум среди длин путей, соединяющих s и t. По определению, $d_G(s, s) = 0, d_G(s, t) = d_G(t, s)$. Путь между вершинами s и t, длина которого равна расстоянию между s и t, называется кратчайшим. Заметим, что таких путей может быть несколько. Если все ребра в графе имеют единичный вес, то кратчайшим является путь с наименьшим количеством ребер.

Пусть σ_{st} — количество кратчайших путей от вершины $s \in V$ до вершины $t \in V$, а $\sigma_{st}(v)$ — количество кратчайших путей от s до t, проходящих через $v \in V$. Тогда индекс $C_B(v)$ для вершины v определяется следующим образом:

$$C_B(v) = \sum_{s \neq t \neq v \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}.$$

В случае ориентированного графа путь строится с учетом направления ребер, так что $d_G(s, t)$ не обязательно равно $d_G(t, s)$. Для адаптации определения для случая, когда не

существует пути из s в t, принято соглашение: $\dfrac{\sigma_{st}(v)}{\sigma_{st}} = 0$, если числитель $\sigma_{st}(v)$ и знаменатель σ_{st} равны нулю.

Значения индекса C_B иногда удобно нормализовать. Одним из естественных способов нормализации является деление значения центральности на количество пар вершин $|V|^2$, т. е. определять „центральность по посредничеству“ следующим образом:

$$C_B(v) = \frac{1}{|V|^2} \sum_{s \neq t \neq v \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}.$$

Тогда [9] значения $C_B(v)$ будут находиться в интервале $[0, 1]$.

2. Базисные алгоритмы нахождения кратчайших путей. Для определения центральности вершины необходимо найти все кратчайшие пути между всеми связанными парами вершин. Предполагаем, что веса ребер неотрицательны.

2.1. Одним из базисных алгоритмов, составляющим основу многих других, является „алгоритм поиска в ширину“ (*breadth-first search, BFS*) [12]. Он применяется к невзвешенным ориентированным и неориентированным графам и относится к числу алгоритмов обхода графа (*traversal algorithms*). Пусть имеется граф $G = (V, E)$ и зафиксирована начальная вершина s. Алгоритм перечисляет все достижимые из s вершины в порядке возрастания расстояния от s. При этом сначала рассматриваются все соседи, потом соседи соседей и т. д. В процессе поиска из графа выделяется часть, называемая „деревом поиска в ширину“, с корнем в s. Для каждой вершины путь из корня в дереве поиска будет одним из кратчайших путей (из начальной вершины) в графе. Поскольку граф невзвешенный, то расстояние от s совпадает с числом ребер кратчайшего пути. Время работы алгоритма оценивается как $O(|V| + |E|)$, т. е. пропорционально размеру представления графа в виде списков смежных вершин.

2.2. Алгоритм Дейкстры (*Dijkstra algorithm*) [13] решает задачу о кратчайших путях из одной вершины для взвешенного ориентированного графа $G = (V, E)$, в котором веса всех ребер неотрицательны, т. е. $w(u, v) \geq 0$ для всех ребер $(u, v) \in E$. Алгоритм начинает работу с некоторой вершины $s \in V$ и на каждом шаге добавляет ближайшую вершину к множеству уже обнаруженных вершин, лежащих на кратчайших путях, таким образом добиваясь выявления всех кратчайших путей из источника до всех остальных. Алгоритм основан на ряде свойств кратчайших путей; первое: отрезки кратчайших путей сами являются кратчайшими путями; второе: для всякой вершины $s \in V$ и для всякого ребра $(u, v) \in E$ выполняется неравенство $d_G(s, v) \leq d_G(s, u) + w(u, v)$. При работе алгоритма используется прием, называемый „релаксацией“ (*relaxation*). На начальный момент для всех $v \in V$ верхняя оценка $d[v]$ для расстояния от исходной вершины s устанавливается равной ∞. Релаксация ребра состоит в следующем: $d[v]$ уменьшается до $d[u] + w(u, v)$, если $d[v] > d[u] + w(u, v)$.

В процессе работы алгоритм строит множество $S \subseteq V$, состоящее из вершин, для которых $d_G(s, v)$ уже найдено. Алгоритм выбирает вершину $u \in V \backslash S$ с наименьшим $d[u]$ и добавляет ее в множество S, производя релаксацию ребер, выходящих из u, после чего цикл повторяется. При работе алгоритма поддерживается очередь Q с приоритетами, определяемыми значениями функции d. Если эта очередь реализуется в виде массива, то время работы алгоритма оценивается как $O(|V|^2)$. Для разреженных графов используется модифицированный алгоритм Дейкстры, в котором время оценивается как $O(|E| \log |V|)$. Если же реализовать очередь Q в виде „фибоначчиевой кучи“ (*Fibonacci heap*), то можно

добиться оценки $O(|V|\log|V|+|E|)$. Для получения кратчайших путей для всех вершин нужно применить алгоритм $|V|$ раз [12].

2.3. Алгоритм Флойда — Уоршолла (*Floyd — Warshall algorithm*) [14], [15], использующий технику динамического программирования, решает задачу о нахождении кратчайших путей для всех пар вершин взвешенного ориентированного графа (без циклов отрицательного веса). Время работы алгоритма оценивается как $O(|V|^3)$.

Пусть дана матрица $W=(w_{ij})$, где:

$$w_{ij}=\begin{cases}0, & \text{если } i=j,\\ \text{вес ребра}(i,j), & \text{если } i\neq j,\ (i,j)\in E,\\ \infty, & \text{если } i\neq j,\ (i,j)\notin E.\end{cases}$$

Обозначим через d_{ij}^k вес кратчайшего пути из вершины i в вершину j с промежуточными вершинами из множества $\{1,2,\dots k\}$. При $k=0$ промежуточных вершин нет, поэтому $d_{ij}^{(0)}=w_{ij}$. Алгоритм основан на рекуррентной формуле для длин кратчайших путей:

$$d_{ij}^k=\begin{cases}w_{ij}, & \text{если } \ k=0,\\ \min\left(d_{ij}^{(k-1)}, d_{ik}^{(k-1)}+d_{kj}^{(k-1)}\right), & \text{если } \ k\geq 0.\end{cases}$$

Он производит преобразование матрицы W в матрицу $D^{(n)}=\left(d_{ij}^n\right)$, содержащую искомые решения, т. е. $d_{ij}^{(n)}=d_G(i,j)$ для всех $i,j\in V$. При этом можно не только вычислять длины и количество кратчайших путей для всех пар вершин, но и строить сами пути. Обобщением подхода Флойда — Уоршолла является способ решения задач о путях в ориентированных графах, основанный на понятии идемпотентного замкнутого полукольца [16].

3. Алгоритмы вычисления индекса C_B и их сложность. Пусть парная зависимость (*pair dependency*) вершин s и t от промежуточной вершины v определяется как

$$\delta_{st}(v)=\frac{\sigma_{st}(v)}{\sigma_{st}}.$$

С использованием определения парной зависимости „центральность по посредничеству“ для вершины v можно представить следующим образом:

$$C_B(v)=\sum_{s\neq t\neq v\in V}\delta_{st}(v).$$

Процедура вычисления центральности традиционно состоит из двух шагов:

1) вычислить длины и количество всех кратчайших путей между всеми парами вершин;

2) просуммировать все парные зависимости.

Заметим, что если известно количество кратчайших путей от вершины u до вершины v, то для того, чтобы вычислить количество тех путей, которые проходят через промежуточную вершину t, используется свойство отрезка кратчайшего пути:

$$\sigma_{uv}(t) = \begin{cases} \sigma_{ut} \times \sigma_{tv}, & \text{если } d_G(u,t) + d_G(t,v) = d_G(u,v), \\ 0, & \text{в противном случае.} \end{cases}$$

Для каждой вершины все парные зависимости можно вычислить за время $O(|V|^2)$, так что общее время вычисления зависит от времени нахождения кратчайших путей.

3.1. Работа [10] ссылается на матричный метод подсчета и нахождения кратчайших путей для всех пар вершин, который может служить основой для подсчета центральности вершин. Пусть $d_{ij}^{(m)}$ обозначает минимальный вес пути из вершины i в вершину j с не более чем m ребрами. Построение основано на формуле, утверждающей, что путь из i в j длины $m \geq 1$ можно разбить на отрезок из $(m-1)$ ребер, ведущий из i в некоторую вершину k и на последнее ребро (k,j), так что

$$d_{ij}^{(m)} = \min\left(d_{ij}^{(m-1)}, \min_{1 \leq k \leq n}\left\{d_{ik}^{(m-1)} + w_{kj}\right\}\right).$$

Вычисление путей идет „снизу вверх“, преобразование аналогично произведению матриц. На основании матрицы $W = (w_{ij})$ строится последовательность матриц $D^{(1)} = W, D^{(2)}, ..., D^{(n-1)}$, где $D^{(m)} = \left(d_{ij}^{(m)}\right)$. Последняя матрица будет содержать веса кратчайших путей. Одновременно строятся матрицы предшествования $\Pi^{(m)}$, такие что $\pi_{ij}^{(m)}$ — это вершина, предшествующая j на каком-либо кратчайшем пути из i в j, состоящем не более чем из m ребер. В худшем случае время работы оценивается как $O(|V|^4)$. Возможно улучшить время до $O(|V|^3 \log|V|)$ [12].

3.2. В работе [17] задача вычисления „центральности по посредничеству“ решается с помощью построения „геодезического“ полукольца, то есть рассматривается обобщенная схема Флойда — Уоршолла. Предложен алгоритм вычисления центральности по посредничеству с оценкой времени $O(|V|^3)$.

3.3. Алгоритм с оценкой $O(|V||E|)$ для невзвешенных графов и оценкой $O(|V||E| + |V|^2 \log|V|)$ для взвешенных предложен в работе [18]. Это наиболее эффективный из известных алгоритмов вычисления „центральности по посредничеству“. В нем техника аккумулирования значений центральности интегрирована с поиском кратчайших путей.

Зависимость (*dependency*) вершины $s \in V$ от единичной вершины $v \in V$ определяется как

$$\delta_{s.}(v) = \sum_{t \in V} \delta_{st}(v). \tag{1}$$

Тогда „центральность по посредничеству“ для вершины v можно представить в виде:

$$C_B(v) = \sum_{s \in V} \delta_{s.}(v). \tag{2}$$

Обозначим через $P_s(v)$ множество предшественников (*predecessors*) вершины $v \in V$ на кратчайшем пути из $s \in V$:

$$P_s(v) = \{u \in V : (u,v) \in E, d_G(s,v) = d_G(s,u) + w(u,v)\}.$$

Связь количества кратчайших путей от s до v ($s \neq v \in V$) с количеством путей до предшественников:

$$\sigma_{sv} = \sum_{u \in P_{s(v)}} \sigma_{su}. \tag{3}$$

В работе [18] показано, что для вычисления значения зависимости вершины $s \in V$ от любой вершины $v \in V$ можно применять следующую ключевую формулу:

$$\delta_{s.}(v) = \sum_{w: v \in P_s(w)} \frac{\sigma_{sv}}{\sigma_{sw}} (1 + \delta_{s.}(w)). \tag{4}$$

Выявление кратчайших путей от начальной вершины s до всех других производится с использованием алгоритмов обхода графа в порядке неубывания расстояния от s: BFS для невзвешенных графов и Дейкстры для взвешенных. В конце каждой итерации, соответствующей вершине $s \in V$, взятой в качестве начальной, рассматриваются все пройденные вершины v в порядке невозрастания расстояния от s, пересчитываются зависимости от предшественников, а зависимость s от $v, \delta_{s.}(v)$ добавляется к значению индекса $C_B(v)$. Таким образом, для любой вершины $s \in V$ выполняются два шага:

1) производится подсчет длин и количества кратчайших путей от $s \in V$ до всех остальных вершин;

2) вычисляются зависимости s от всех вершин на путях с использованием (1), (3), (4) и суммируются со значениями центральности согласно (2).

В результате будут рассмотрены все пары вершин и просуммированы парные зависимости. В случае неориентированных графов значение нужно делить на два, так как все кратчайшие пути рассматриваются дважды.

4. Вычислительный эксперимент. Целью является вычисление индекса C_B для данных, извлеченных из РББД *RePEc* (*Research Papers in Economics* — „Исследовательские статьи по экономике“). В состав этой базы входят периодические издания из области экономики, финансов, менеджмента и маркетинга. В работе [19] мы уже обращались к этой базе с целью вычисления метрик *Eigenfactor* и *ArticleInfluence*, опирающихся на понятие „центральности собственного вектора“ (*eigenvector centrality*, [20]).

Пусть граф $G = (V, E)$ представляет сеть цитирования. Дуга $(u, v) \in E$ тогда и только тогда, когда публикация из издания u за рассматриваемый год (окно цитирования, в данном случае 2013 год), цитирует публикацию из издания v, относящегося к пяти предыдущим годам (окно публикации). В качестве веса дуги $w(u, v)$ рассматриваем величину, обратную количеству цитирований от u к v $(1/cit(v, u))$. Для вычисления индекса C_B для вершин невзвешенного и взвешенного графов используется подход Брандеса. Псевдокод алгоритма для невзвешенных графов предложен в работе [18], псевдокод алгоритма для взвешенных графов предложен авторами данной статьи и приведен в Приложении.

На время извлечения данных о цитировании в РББД насчитывалось 5811 коллекций документов, из них 1708 журналов. Из этого набора журналов отобраны 317, каждый из которых получил более 5 цитирований, опубликовал более 20 статей за „окно цитирования“, и его процент самоцитирований меньше или равен 50 %. Обозначим это множество через L.

Полной коллекции соответствует граф G_1, имеющий 3011 изолированных вершин, 2 связанных только друг с другом и 2798 вершин, образующих слабо связанную компоненту. Максимальное расстояние между связанными вершинами G_1 равно 14. Взвешенный вариант графа G_1 обозначим через G_2. Получены значения C_B для вершин графов G_1 и

Таблица 1

Ранг журналов множества L (взвешенный вариант)

Ранг	Индекс C_B	Название журнала / Издательство	In-degree	Out-degree
1	190236.67	American Economic Review / American Economic Association	629	27
2	102555.33	Research Policy / Elsevier	155	100
3	83201.83	Journal of Economic Dynamics and Control / Elsevier	219	171
4	78716.17	Journal of International Economics / Elsevier	276	142
5	69437.00	Journal of Development Economics / Elsevier	288	58
6	56479.33	Journal of Econometrics / Elsevier	285	74
7	55815.67	Economics Letters / Elsevier	316	157
8	47609.00	Games and Economic Behavior / Elsevier	165	66
9	47265.33	Journal of Health Economics / Elsevier	147	74
10	44213.67	Journal of Financial Economics / Elsevier	246	57
11	42868.17	World Development / Elsevier	177	50
12	39539.50	Labour Economics / Elsevier	203	43

Таблица 2

Ранг журналов множества L (невзвешенный вариант)

Ранг	Индекс C_B	Название журнала/Издательство	In-degree	Out-degree
1	80793.32	Economics Letters / Elsevier	316	157
2	65131.82	American Economic Review / American Economic Association	629	27
3	58862.80	Journal of Economic Dynamics and Control / Elsevier	219	171
4	48286.68	Research Policy / Elsevier	155	100
5	46557.76	Journal of International Economics / Elsevier	276	142
6	38765.05	Journal of Econometrics / Elsevier	285	74
7	25770.87	World Development / Elsevier	177	50
8	25218.45	Review of Economic Dynamics / Elsevier for the Society for Economic Dynamics	160	124
9	24026.93	Applied Economics / Taylor & Francis Journals	224	34
10	23265.53	Journal of Development Economics / Elsevier	288	58
11	23160.63	Journal of International Money and Finance / Elsevier	182	90
12	21225.93	Journal of Economic Literature / American Economic Association	359	29

G_2, и произведено ранжирование по невозрастанию журналов полной коллекции на основе значений индексов соответствующих вершин.

Для представления результатов из ранжированной коллекции выбирается упорядоченное множество журналов L. Двенадцать журналов, имеющих наивысший ранг для взвешенного варианта, представлены в табл. 1 (для невзвешенного варианта в табл. 2). Там же приведены значения центральности соответствующих вершин по степени в графе G_1. Для графического представления результатов взаимного цитирования журналов множества L (рис. 1) использован программный продукт *Pajek* [21]. Рис. 1 иллюстрирует связность журналов с наивысшим рангом между собой, при этом величина „кружка“ пропорциональна значению степени по входу (см. значение *In-degree*). Вершины расположены по часовой стрелке, начиная с вершины с наивысшим рангом. Номера вершин соответствуют рангам журналов, приведенных в табл. 1.

Сравнение табл. 1 и 2 показывает, что по составу есть различие в пяти журналах,

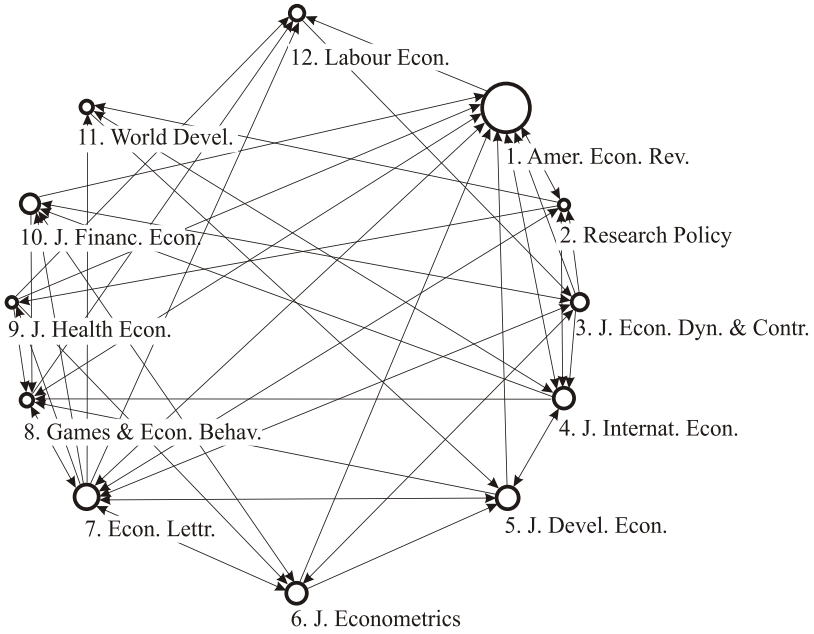

Рис. 1. Взаимное цитирование журналов множества L

имеющих наивысший ранг. Для сравнения результатов для всей коллекции изданий были рассчитаны коэффициенты ранговой корреляции Спирмена (0,850) и коэффициент корреляции Пирсона (0,823). То есть, для этой коллекции различия между взвешенным и невзвешенным вариантом не столь существенны.

Приложение. Псевдокод алгоритма вычисления индекса C_B для вершин ориентированного взвешенного графа.

$C_B(v) \leftarrow 0, v \in V$;
for $s \in V$ **do**
    $S \leftarrow$ empty stack;
    $P[w] \leftarrow$ empty list, $w \in V$;
    $\sigma[t] \leftarrow 0, t \in V$; $\sigma[s] \leftarrow 1$;
    $d[t] \leftarrow \infty, t \in V$; $d[s] \leftarrow 0$;
    $Q \leftarrow$ empty queue ;
    enqueue $v \rightarrow Q, v \in V$;
    // *найти расстояние от s до всех вершин*
    while Q *not empty* **do**
        // *взять вершину с минимальным значением d*
        extract_min $v \leftarrow Q$;
        enqueue $v \rightarrow Q_1$;
        // *рассмотреть всех соседей v*
        foreach *neighbor w of v* **do**
            // *путь до w через v является кратчайшим?*
            if $d[w] > d[v] + w(v, w)$ **then**
                $d[w] = d[v] + w(v, w)$

```
            end
        end
    end
        // вычислить количество кратчайших путей до всех вершин и их состав
        // в Q_1 вершины находятся в порядке не убывания расстояния от s
        while Q_1 not empty do
            dequeue v ← Q_1;
            push v → S;
            foreach neighbor w of v do
                // путь до w через v является кратчайшим?
                if d[w] = d[v] + w(v, w) then
                    σ[w] ← σ[w] + σ[v];
                    append v → P[w];
                end
            end
        end
        δ[v] ← 0, v ∈ V ;
        // Пересчитать зависимости от предшественников и переопределить
        центральности; S возвращает вершины в порядке невозрастания
        расстояния от s
        while S not empty do
            pop w ← S;
            for v ∈ P[w] do δ[v] ← δ[v] + (σ[v]/σ[w])(1 + δ[w]);
            if w ≠ s then C_B[w] ← C_B[w] + δ[w];
        end
end
```

Здесь $d[t]$ — верхняя оценка длины кратчайшего пути от начальной вершины до вершины t; Q — очередь вершин с приоритетами, определяемыми значениями функции d; $P[w]$ — список предшественников w на кратчайших путях от начальной вершины до w; $\sigma(w)$ — оценка количества кратчайших путей от начальной вершины до w; $\delta(v)$ — оценка зависимости начальной вершины от v; Q_1 — очередь, в которой вершины хранятся в порядке неубывания расстояния от начальной вершины; S — стек, в котором вершины хранятся в порядке невозрастания расстояния от начальной вершины.

Замечание. При реализации вместо использования стека S (приведенного для совместимости с изложением алгоритма Брандеса в части расчета центральности) Q_1 формируется как массив, который в первом цикле (**while** Q_1) просматривается с начала, во втором случае (**while** S) — просматривается с конца.

Список литературы

1. XML Media Types, IETF RFC 3023. Jan. 2001.

2. Lotka A. The frequency distribution of scientific productivity // J. Washington Acad. Sci. 1926. V. 16, N 12. P. 317–324.

3. Price D. A general theory of bibliometric and other cumulative advantage process // J. Amer. Soc. Inform. Sci. 1976. N 27. P. 292–306.

4. Kessler M. M. Bibliographic coupling between scientific papers // Amer. Documentation. 1963. V. 14, iss. 1. P. 10–25.

5. Маршакова И. В. Система связей между документами, построенная на основе ссылок: по данным Science Citation Index // НТИ. Сер. 2. 1973. № 6. С. 3–8.

6. Small H. Co-citation in the scientific literature: A new measure of the relationship between two documents // J. Amer. Soc. Inform. Sci. 1973. V. 24, iss. 4. P. 265–269.

7. Латур, Б. Пересборка социального: введение в акторно-сетевую теорию. М.: Высш. шк. экономики. 2014. 384 с.

8. Newman M. E. J. Analysis of weighted networks // Phys. Rev. 2004. E 70, 056131.

9. Newman M. E. J. Networks. An Introduction. NY: Oxford University Press. 2010. 772 P.

10. Freeman L. C. A set of measures of centrality based upon betweenness // Sociometry. 1977. V. 40. P. 35–41.

11. RePEc. General principles. [Electron. resource]. http://repec.org/.

12. Кормен Т., Лейзерсон Ч., Ривест Т. Алгоритмы: построение и анализ. М.: МЦНМО. 2002. 960 с.

13. Dijkstra E. W. A note on two problems in connexion with graphs. // Numerische Mathematik. 1959. V. 1. P. 269–271.

14. Floyd R.W. Algorithm 97: Shortest path // Communications of the ACM. 1962. V. 5, iss. 6. P. 345.

15. Warshall S. A theorem on Boolean matrices // J. Assoc. Comp. Math. 1962. V. 9, N. 1. P. 11–12.

16. Ахо А., Хопкрофт Дж., Ульман Дж. Построение и анализ вычислительных алгоритмов. М.: Мир, 1979. 536 с.

17. Batagelj V. Semirings for social network analysis // J. of Math. Sociology. 1992. V. 19, iss. 1. P. 53–68.

18. Brandes U. A faster algorithm for betweenness centrality // J. of Mathematical Sociology. 2001. V. 25, iss. 2. P. 163–177.

19. Бредихин С. В., Ляпунов В. М., Щербакова Н. Г. Ранжирование коллекции периодических изданий базы данных RePec на основе метрик Eigenfactors // Проблемы информатики. 2014. № 1. С. 36–42.

20. Bonacich P. F. Power and centrality: A family of measures // Am. J. Sociol. 1987. V. 92. P. 1170–1182.

21. Batagelj V., Mrvar A. Pajek. Program for Large Network Analysis [Electron. resource]. http://vlado.fmf.uni-lj.si/pub/networks/doc/pajek.pdf.

Бредихин Сергей Всеволодович — канд. техн. наук,
зав. лабораторией Института вычислительной
математики и математической геофизики СО РАН;
e-mail: bred@nsc.ru;
Щербакова Наталья Григорьевна — ст. науч. сотр.
Института вычислительной математики
и математической геофизики СО РАН;
e-mail: nata@nsc.ru

Дата поступления — 19.08.2014