

С.В. Еремеев

## АЛГОРИТМ РАЗМЕЩЕНИЯ СЛОЕВ НА ЦИФРОВОЙ КАРТЕ В ГИС

### Введение

В последнее время в геоинформационных системах (ГИС) очень много внимания стало уделяться топологическим отношениям между объектами [1, 2]. Количество и виды отношений зависят от функций конкретной ГИС [3]. Будем рассматривать следующие топологические отношения: изолированность, близость, соседство и вложенность [4, 5].

Нанесение объектов на карту представляет собой трудоемкий и длительный процесс. Чтобы карта была топологически корректной необходимо добавлять информацию о связях между объектами. Чаще всего ввод пространственной и топологической информации представляет собой беспорядочный процесс, где нет четкой последовательности размещения объектов слоев. Это несет множество исправлений при создании карты. Поэтому сейчас актуальной является задача выбора последовательного размещения слоев карты. Например, сначала должны размещаться кварталы, а затем дома, которые в них вложены. Существующие ГИС не позволяют определять порядок размещения слоев, возлагая данную задачу на пользователя.

Наиболее сложно решение этой задачи в муниципальных ГИС (МГИС), где существует от нескольких десятков до нескольких сотен слоев. Причем большинство из них связаны междуслойными отношениями. Перед пользователем встает сложный вопрос: объекты какого слоя разместить первыми, а какого вторыми так, чтобы было проще установить взаимодействия между объектами.

В данной статье рассматривается алгоритм размещения слоев на цифровой карте, который позволяет:

- определить порядок следования слоев на основе междуслойных топологических отношений;
- выявить наиболее зависимые слои карты;
- формализовать ввод пространственной информации.

### Постановка задачи

Пусть  $S = \{s_i\}$  ( $i = 1, 2, \dots, n$ ) – множество слоев, которые состоят из пространственно-распределенных объектов.  $\Phi = [\varphi_{ij}^k]$  ( $i, j = 1, 2, \dots, n$ ) ( $k$  – вид топологического отношения) – матрица топологических отношений между слоями.

$$A = \begin{pmatrix} 1 & 2 & \dots & n \\ \alpha_1 & \alpha_2 & \dots & \alpha_n \end{pmatrix} \text{ – подстановка, в ко-}$$

торой перестановка  $(1\ 2\ \dots\ n)$  определяет исходный порядок индексов слоев  $\{s_i\}$ , перестановка  $(\alpha_1\ \alpha_2\ \dots\ \alpha_n)$  определяет новый порядок индексов слоев  $\{s_i\}$ .

$$\text{Причем } \sum_{i=1}^n i = \sum_{i=1}^n \alpha_i.$$

$$\text{Например, подстановка } A = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 4 & 3 & 1 & 2 \end{pmatrix} \text{ оз-}$$

начает, что слой  $s_1$  необходимо размещать на карте четвертым, т.к. его индекс 1 переходит в индекс 4,  $s_2$  – третьим,  $s_3$  – первым и  $s_4$  – вторым.

Необходимо найти перестановку  $(\alpha_1\ \alpha_2\ \dots\ \alpha_n)$ . Она позволит определить последовательность размещения слоев, т.е.  $S = \{s_{\alpha_1}, s_{\alpha_2}, \dots, s_{\alpha_n}\}$ .

Для решения данной задачи необходимо:

- присвоить приоритеты слоям на основе топологических отношений;
- найти зависимости между слоями;
- найти перестановку индексов слоев.

### Описание алгоритма

Пусть  $s_1 \in S$  и  $s_2 \in S$  – два слоя, между которыми существует топологическое отношение, т.е.  $\varphi_{12}^k \neq 0$ . Будем рассматривать приоритеты между слоями с учетом следующих видов топологических отношений между ними:

1. Изолированность ( $k = 1$ ).

Если слой  $s_1$  изолирован от слоя  $s_2$ , то  $s_1$  не зависит от  $s_2$ , тем самым их индексы, определяющие порядок размещения, могут быть больше или меньше по отношению друг к другу, т.е.  $(\alpha_1 > \alpha_2) \vee (\alpha_2 < \alpha_1)$ .

2. Близость ( $k = 2$ ).

Если слой  $s_1$  близок к слою  $s_2$ , то объекты слоя  $s_1$  должны располагаться после объектов слоя  $s_2$ , т.е.  $\alpha_1 > \alpha_2$ .

3. Соседство ( $k = 3$ ).

Если слой  $s_1$  соседствует со слоем  $s_2$ , то сначала нужно расположить на карте объекты слоя  $s_2$ , а затем соседствующие с ними из  $s_1$ , т.е.  $\alpha_1 > \alpha_2$ .

4. Вложенность ( $k = 4$ ).

Если слой  $s_1$  вложен в слой  $s_2$ , то в первую очередь нужно разместить объекты слоя  $s_2$ , которые вмещают объекты из  $s_1$ , т.е.  $\alpha_1 > \alpha_2$ .

На основании приоритетов (1-4) можно сделать следующие выводы:

- слой  $s_1$  не зависит от слоя  $s_2$ , т.е.  $(\alpha_1 > \alpha_2) \vee (\alpha_2 < \alpha_1)$ , если между двумя слоями установлено отношение изолированности или слои не взаимодействуют между собой, т.е.:

$$\varphi_{12}^k \neq 0 \quad (k = 1) \text{ или } \varphi_{12}^k \neq 0 \quad (k = 1 \vee k = 2 \vee k = 3 \vee k = 4);$$

- слой  $s_1$  зависит от слоя  $s_2$ , т.е.  $\alpha_1 > \alpha_2$ , если между ними существуют отношения близости, соседства или вложенности, т.е.  $\varphi_{12}^k \neq 0$  ( $k = 2 \vee k = 3 \vee k = 4$ );

Введем матрицу зависимостей  $\Omega = [\omega_{ij}]$  ( $i, j = 1, 2, \dots, n$ ) слоев  $\{s_i\}$ , которая вычисляется из матрицы топологических отношений  $\Phi = [\varphi_{ij}^k]$  между слоями следующим образом:

$$\omega_{ij} = \begin{cases} 1, & \text{если } \varphi_{ij}^k \neq 0 \quad (k = 2 \vee k = 3 \vee k = 4) \\ 0, & \text{если } \varphi_{ij}^k \neq 0 \quad (k = 1) \vee \varphi_{ij}^k = 0 \quad (k = 1 \vee k = 2 \vee k = 3 \vee k = 4) \end{cases}, \quad (1)$$

где  $[\omega_{ij}]$  ( $i, j = 1, 2, \dots, n$ ) – элементы матрицы зависимостей  $\Omega$ .

Т.е.  $\omega_{ij} = 1$ , если слой  $s_i$  зависит от слоя  $s_j$  и  $\omega_{ij} = 0$  в противном случае.

### Свойство матрицы $\Omega$

Симметричные элементы матрицы  $\Omega$  относительно главной диагонали не равны между собой, т.е.  $\omega_{ij} \neq \omega_{ji}$  ( $i, j = 1, 2, \dots, n$ ). Это означает, что нет обратной зависимости между слоями  $s_i$  и  $s_j$ .

Множество топологических отношений может быть расширено, но при каждом виде отношения необходимо определить, зависит слой  $s_i$  от слоя  $s_j$  или нет.

Для формирования матрицы  $\Omega$  нет ограничений на количество видов отношений.

### Анализ матрицы $\Omega$

Приведем пример матрицы зависимостей слоев:

$$\Omega = \begin{matrix} & \begin{matrix} s_1 & s_2 & s_3 & s_4 & s_5 \end{matrix} \\ \begin{matrix} s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \end{matrix} & \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{matrix}. \quad (2)$$

По строкам и столбцам матрицы  $\Omega$  расположены слои  $s_i$  ( $i = 1, 2, \dots, 5$ ).

Анализ матрицы  $\Omega$  по строкам показывает, что в строке  $i$  ( $i = 1, 2, \dots, 5$ ) можно найти такие слои  $s_j$  ( $j = 1, 2, \dots, 5$ ), от которых зависит слой  $s_i$ . Например, слой  $s_1$  зависит от слоя  $s_5$ ,  $s_2$  – от  $s_5$ ,  $s_3$  – от  $s_2$  и  $s_5$ ,  $s_4$  – от  $s_1$ .

При анализе матрицы  $\Omega$  по столбцам можно найти в столбце  $j$  ( $j = 1, 2, \dots, 5$ ) такие слои  $s_i$  ( $i = 1, 2, \dots, 5$ ), которые зависят от слоя  $s_j$ . Например, от слоя  $s_1$  зависит слой  $s_4$ , от  $s_2$  –  $s_3$ , от  $s_5$  –  $s_1$ ,  $s_2$  и  $s_3$ .

### Матрица цепочек

Обозначим через  $\Lambda = [\lambda_{ij}]$  ( $i, j = 1, 2, \dots, n$ ) матрицу цепочек, где каждая цепочка соответствует строке матрицы. Элементы строки  $i$  ( $i = 1, 2, \dots, n$ ) содержат ненулевую цепочку индексов слоев  $s_j$  ( $j = 1, 2, \dots, n$ ), объекты которых последовательно необходимо разместить перед нанесением на карту объектов слоя  $s_i$ .

Матрица цепочек  $\Lambda$  формируется из матрицы зависимостей  $W$  следующим образом:

$$\begin{cases} \lambda_{is} = jnpui & \omega_{ij} \neq 0 \\ \lambda_{it} = 0 & npui \quad \omega_{ij} = 0 \end{cases} \quad (3)$$

( $i, j = 1, 2, \dots, n; s = 1, 2, \dots, q_i; t = q_i + 1, q_i + 2, \dots, n$ ), где  $q_i$  – количество ненулевых элементов цепочки в строке  $i$ .

Значения  $q_i$  определяются по следующей формуле:

$$q_i = \sum_{j=1}^n \omega_{ij} \quad (i, j = 1, 2, \dots, n). \quad (4)$$

Приведем пример матрицы цепочек, которая вычислена на основе матрицы (2):

$$\Lambda = \begin{matrix} & \begin{matrix} s_1 & s_2 & s_3 & s_4 & s_5 \end{matrix} \\ \begin{matrix} s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \end{matrix} & \begin{bmatrix} 5 & 0 & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 & 0 \\ 5 & 2 & 0 & 0 & 0 \\ 5 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{matrix}. \quad (5)$$

Из этой матрицы видно, что для размещения объектов слоя  $s_1$  нужно сначала разместить объекты слоя  $s_5$ ;  $s_2$  –  $s_5$ ;  $s_3$  –  $s_2$  и  $s_5$ ;  $s_4$  –  $s_1$  и  $s_5$ , а объект  $s_5$  является независимым.

Преобразуем матрицу  $\Lambda$  так, чтобы последний элемент цепочки строки  $i$  содержал индекс слоя  $s_i$ . В результате матрица цепочек (5) примет вид:

$$\Lambda = \begin{matrix} s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \end{matrix} \begin{bmatrix} 5 & 1 & 0 & 0 & 0 \\ 5 & 2 & 0 & 0 & 0 \\ 5 & 2 & 3 & 0 & 0 \\ 5 & 1 & 4 & 0 & 0 \\ 5 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (6)$$

**Поиск перестановки  $(\alpha_1, \alpha_2, \dots, \alpha_n)$**

**1 способ.** Сделаем фильтрацию матрицы  $\Lambda$  так, чтобы ее элементы не содержали повторяющихся значений. Все неповторяющиеся значения запишем в вектор  $\tau = (\tau_1, \tau_2, \dots, \tau_n)$ . В каждом столбце матрицы  $\Lambda$  нужно отбросить все повторяющиеся элементы. В результате получим матрицу  $\Lambda' = [\lambda'_{ij}]$  ( $i, j = 1, 2, \dots, n$ ), которая формируется следующим образом:  $\lambda'_{ij} = \lambda_{ij}$ , если не существует элемента  $\lambda_{ij}$  в  $j$ -ом столбце матрицы  $\Lambda'$ , и  $\lambda'_{ij} = 0$  в противном случае.

Покажем пример матрицы  $\Lambda'$ , которая сформирована из матрицы (6):

$$\Lambda' = \begin{matrix} s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \end{matrix} \begin{bmatrix} 5 & 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (7)$$

**Свойство матрицы  $\Lambda'$**

Сумма элементов матрицы  $\Lambda'$  равна сумме индексов слоев  $s_i$ , т.е.

$$\sum_{i=1}^n \sum_{j=1}^n \lambda'_{ij} = \sum_{i=1}^n i \quad (8)$$

Вектор  $\tau = (\tau_1, \tau_2, \dots, \tau_n)$  определяется из матрицы  $\Lambda'$ . Для заполнения вектора  $\tau$  необходимо сделать сканирование матрицы  $\Lambda'$  по столбцам и последовательно занести в него все ненулевые элементы этой матрицы.

По вектору  $\tau = (\tau_1, \tau_2, \dots, \tau_n)$  найдем искомую перестановку  $(\alpha_1, \alpha_2, \dots, \alpha_n)$ :

$$\alpha_i = j, \text{ если } i = \tau_j, (i, j = 1, 2, \dots, n). \quad (9)$$

Например, в результате сканирования матрицы (7) по столбцам найдем вектор  $\tau = (5, 1, 2, 3, 4)$ . Тогда подстановка будет равна:

$$A = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 3 & 4 & 5 & 1 \end{pmatrix}$$

**2 способ. Формирование перестановки  $(\alpha_1, \alpha_2, \dots, \alpha_n)$  методами теории графов.** Недостаток первого способа состоит в том, что он не

позволяет построить перестановку  $(\alpha_1, \alpha_2, \dots, \alpha_n)$  по цепочке связей. То есть хотелось бы получить следующую последовательность размещения объектов на карте:  $s_5, s_1, s_4, s_2$  и  $s_3$ .

Для решения данной задачи из матрицы  $\Lambda$  сформируем граф:

$$G = (V, E), \quad (10)$$

где  $V = \{v_i\}$  ( $i = 1, 2, \dots, n$ ) – множество вершин, которые соответствуют слоям  $s_i$ ,  $E$  – множество дуг между вершинами.

Приведем граф зависимостей слоев по матрице (6), который показан на рис. 1.

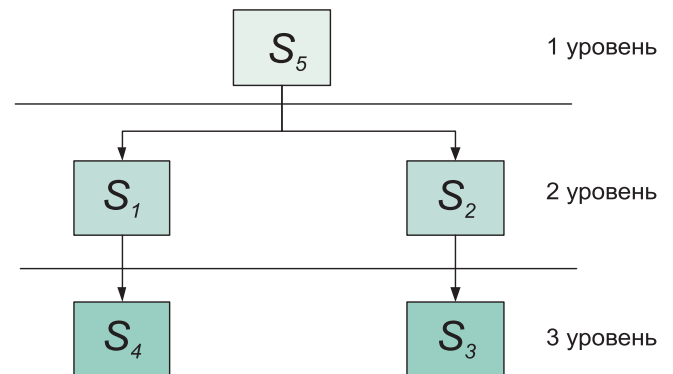


Рис. 1. Граф зависимостей слоев

Первый уровень составляют слои из первого столбца матрицы  $\Lambda$ , второй – из второго столбца, третий – из третьего. Количество уровней равно количеству ненулевых столбцов матрицы  $\Lambda$ .

Требуется найти вектор  $\tau = (\tau_1, \tau_2, \dots, \tau_n)$ , который определяет порядок следования слоев. Пусть списки смежности вершин имеют вид:

$$S_5 \rightarrow (S_1, S_2), S_1 \rightarrow (S_4), S_2 \rightarrow (S_3).$$

В результате поиска в глубину имеем следующий вектор  $\tau = (S_5, S_1, S_4, S_2, S_3)$ , а при использовании алгоритма поиска в ширину  $\tau = (S_5, S_1, S_2, S_4, S_3)$ .

Аналогично первому способу по вектору  $\tau = (\tau_1, \tau_2, \dots, \tau_n)$  найдем искомую перестановку  $(\alpha_1, \alpha_2, \dots, \alpha_n)$  и вместе с ней подстановку  $A$ :

$$1. A = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 4 & 5 & 3 & 1 \end{pmatrix} \text{ – поиск в глубину;}$$

$$2. A = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 3 & 5 & 4 & 1 \end{pmatrix} \text{ – поиск в ширину.}$$

**Практическое использование алгоритма**

В качестве примера рассмотрим карту некоторого города. Для задания топологических отношений между слоями создадим пространственную модель данных, изображенную на рис. 2.

Исходный порядок слоев представим в виде:  $s_1$  – площади;  $s_2$  – фонтаны;  $s_3$  – памятники;  $s_4$  – реки;  $s_5$  – берега;  $s_6$  – пляжи;  $s_7$  – церкви;  $s_8$  – авто-

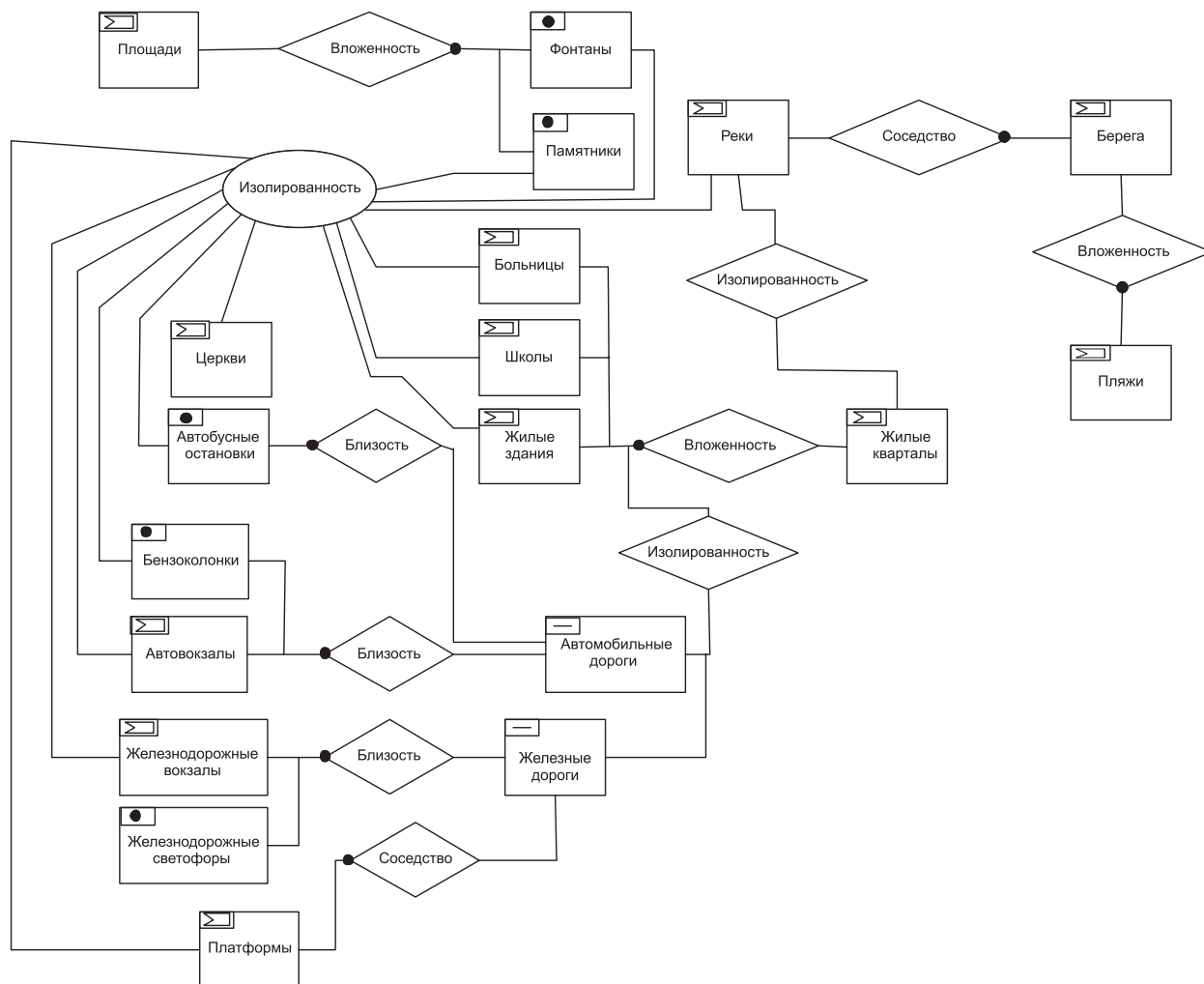


Рис. 2. Пространственная модель данных карты некоторого города

бусные остановки;  $s_9$  – бензоколонки;  $s_{10}$  – автовокзалы;  $s_{11}$  – железнодорожные вокзалы;  $s_{12}$  – железнодорожные светофоры;  $s_{13}$  – платформы;  $s_{14}$  – больницы;  $s_{15}$  – школы;  $s_{16}$  – жилые здания;  $s_{17}$  – жилые кварталы;  $s_{18}$  – автомобильные дороги;  $s_{19}$  – железные дороги.

На этой модели отражены слои и топологические отношения между ними. При отображении слоев учитывается тип пространственных объектов (точечный, линейный и площадной). Например, точечные объекты «Фонтаны» и «Памятники» вложены в площадной объект «Площади». В тоже время «Фонтаны» и «Памятники» изолированы от следующих объектов: «Платформы», «Железнодорожные вокзалы», «Автовокзалы», «Бензоколонки», «Автобусные остановки», «Церкви», «Жилые здания», «Школы», «Больницы», «Реки». В свою очередь, с реками соседствуют берега, в которые вложены пляжи. К линейным объектам в данной модели относятся автомобильные и железные дороги.

Отношения между слоями задаются через диалог с пользователем и хранятся в базе данных.

После задания топологических отношений между слоями на основе пространственной модели

данных формируется граф зависимостей слоев, изображенный на рис. 3.

На основании данного графа автоматически определяется порядок размещения слоев на карте, который покажем в табл. 1.

Из таблицы 1 видно, что сначала в списке в ширину идут топологически независимые слои (если не учитывать изолированность): «Площади», «Реки», «Церкви», «Жилые кварталы», «Автомобильные дороги», «Железные дороги», затем все остальные слои, которые топологически зависят от этих слоев или друг от друга. В списке в глубину сначала идет независимый слой, например, «Площади», а затем слои, которые от него зависят: «Фонтаны» и «Памятники».

### Заключение

Предложен алгоритм автоматического определения последовательности размещения слоев на карте, позволяющий переопределять исходный порядок следования слоев на основании междуслойных топологических отношений. Разработаны две модификации алгоритма с использованием методов теории графов. Поиск в глубину дает лучший результат по срав-

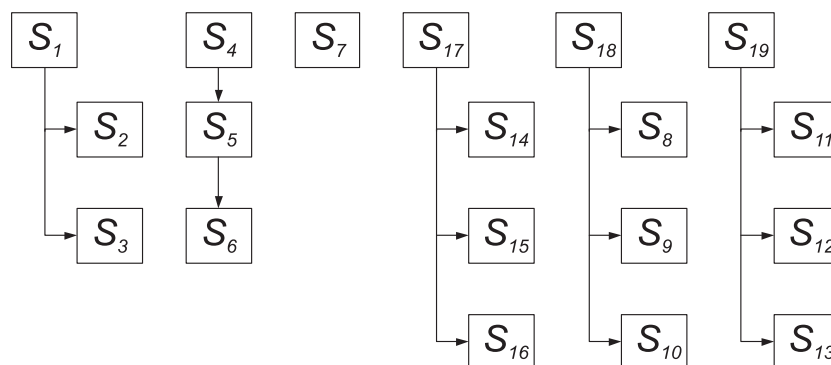


Рис. 3. Граф зависимостей слоев

Таблица 1

## Порядок размещения слоев на карте города

№	Исходный список слоев	Список в ширину	Список в глубину
1	площади	площади	площади
2	фонтаны	реки	фонтаны
3	памятники	церкви	памятники
4	реки	жилые кварталы	реки
5	берега	автомобильные дороги	берега
6	пляжи	железные дороги	пляжи
7	церкви	фонтаны	церкви
8	автобусные остановки	памятники	жилые кварталы
9	бензоколонки	берега	больницы
10	автовокзалы	больницы	школы
11	железнодорожные вокзалы	школы	жилые здания
12	железнодорожные светофоры	жилые здания	автомобильные дороги
13	платформы	автобусные остановки	автобусные остановки
14	больницы	бензоколонки	бензоколонки
15	школы	автовокзалы	автовокзалы
16	жилые здания	железнодорожные вокзалы	железные дороги
17	жилые кварталы	железнодорожные светофоры	железнодорожные вокзалы
18	автомобильные дороги	платформы	железнодорожные светофоры
19	железные дороги	пляжи	платформы

нению с поиском в ширину. Рассмотренный алгоритм позволяет более формализовано подойти к вводу пространственной и топологической информации, значительно уменьшает время на установление топологических отношений между объектами, что существенно снижает нагрузку пользователю. Алгоритм также можно использовать при проектировании информационных систем и разработки программного обеспечения, где требуется определить зависимости между объектами и модулями программы.

## ЛИТЕРАТУРА

1. Шаши Шехар, Санжей Чаула. Основы пространственных баз данных./ Пер. с англ. – М.: КУДИЦ-ОБРАЗ, 2004. – 336 с.  
2. [www.integro.ru](http://www.integro.ru).

3. Основы геоинформатики: В 2 кн. Кн.2: Учебное пособие для студ. вузов / Е.Г. Капралов, А.В. Кошкарев, В.С. Тикунов и др.; Под. ред. В.С. Тикунова. – М.: Издательский центр «Академия», 2004. 480 с.

4. Садыков С.С., Андрианов Д.Е., Еремеев С.В. Формальное определение топологических отношений между картографическими объектами в ГИС // Обработка информации: методы и системы: Сборник научных статей – М.: Горячая линия – Телеком, 2003, С. 49-58.

5. Садыков С.С., Еремеев С.В. Анализ информации в ГИС, распределенной по разным слоям // Современные управляющие и информационные системы: Сборник научных статей, г. Ташкент, 2003, С. 120-125.