

УДК 656.7

## МЕТОД РАСЧЕТА СЕТЕВЫХ ГРАФИКОВ ВЫПОЛНЕНИЯ РАБОТ, УЧИТЫВАЮЩИЙ НАЛИЧИЕ РАСПОЛАГАЕМЫХ РЕСУРСОВ

Г.В. ГОЛОВЧЕНКО<sup>1</sup>, М.И. РЕБЕЗОВА<sup>2</sup>

<sup>1</sup>Санкт-Петербургский университет гражданской авиации, г. Санкт-Петербург, Россия

<sup>2</sup>Транспортная Клиринговая Палата, г. Москва, Россия

Рассматривается сетевой график выполнения работ бригадами по комплексному обслуживанию рейсов в аэропорту. В известных алгоритмах нахождения критического пути предполагается, что длительности работ заданы, а начало выполнения работы определяется только временем наступления начального для него события. В действительности начало выполнения работы зависит также от наличия необходимых для ее выполнения ресурсов в нужный период времени.

Описывается формальная постановка задачи. Для решения задачи в работе вначале используется известный алгоритм построения дерева кратчайших путей сетевого графика в предположении, что длительности выполнения работ известны, а ресурсов достаточно, с тем чтобы начать каждую работу в нужное время. Далее этот алгоритм модифицируется с целью учета имеющихся в наличии ресурсов. Для описания ресурсов вводится специальная аналитическая функция, представленная в виде двумерной матрицы, в которой строки соответствуют различным единицам ресурса, а столбцы – моментам времени суток, равноудаленным на какое-то фиксированное время. Для решения оптимизационной задачи предлагается численный метод последовательного поиска с горизонтом просмотра, содержащий два цикла: внешний – по бригадам (им соответствуют строки матрицы) и внутренний – по времени суток (им соответствуют столбцы этой матрицы). Метод реализован с использованием вычислительной среды Matcad. Предложенный метод применим как для обслуживания отдельного объекта, так и для потока объектов. Метод валидирован на конкретных примерах.

В заключительной части статьи приводятся сведения об использовании разработанного алгоритма в пакете прикладных программ автоматизированной системы комплексного обслуживания рейсов в аэропорту (КОБРА) для планирования обслуживания задержанных рейсов в аэропорту в сбойных ситуациях.

**Ключевые слова:** сетевой график, критический путь, ограниченные ресурсы, обслуживание авиарейсов.

### ВВЕДЕНИЕ

Сетевой график выполнения работ является удобным и широко применяемым средством [1, 2, 3]. Он представляет собой ориентированный ациклический граф  $G = (V, E)$  с множеством вершин  $V$  и множеством дуг  $E$ . В графе имеется одна вершина (источник)  $v_0 \in V$  без входящих дуг и одна вершина (сток)  $v_k \in V$  без выходящих дуг. Дуги графа  $e \in E$  соответствуют работам, а вершины  $v \in V$  – событиям, заключающимся в том, что для всех входящих в вершину дуг соответствующие им работы завершены. В свою очередь каждая работа может быть начата, если произошло событие, которое является начальным для соответствующей дуги. Каждая дуга  $e$  имеет длину  $d(e)$ , равную длительности выполнения соответствующей работы. Длительность выполнения всех работ равна длине самого длинного (критического) пути из источника в сток.

Имеется большое количество красивых алгоритмов нахождения критического пути [4, 5, 6, 7]. В них всегда предполагается, что длины дуг (длительности работ) заданы, а начало выполнения работы определяется только временем наступления начального для него события. В действительности оно зависит также от наличия необходимых для выполнения работы ресурсов в нужный период времени. В статье излагаются алгоритмы нахождения критического пути для этого случая. Опишем формальную постановку задачи.

Итак, заданы граф  $G = (V, E)$  и длины дуг (длительности выполнения работ)  $\{d(e)\}$ . Кроме того, для каждой работы с номером  $m$  известны вид необходимого ресурса  $\rho(m)$  и потребляемый его объем  $r(m)$ . Известны также временные графики располагаемых ресурсов (бригад исполнителей, средств механизации и пр.) Аналитически график для  $\rho$ -го вида ресурса зада-

ется функцией времени  $g(t, \rho)$ . Значение  $g(t, \rho)$  равно объему  $\rho$ -го вида ресурса, имеющемуся в момент времени  $t$ .

Для каждой работы с номером  $m$  необходимо определить время ее начала  $t^*(m)$  и окончания  $t^*(m)$  так, чтобы она могла быть выполнена в этот период и чтобы время выполнения всех работ сетевого графика (длина критического пути) было минимальным.

Обобщением описанной постановки является рассмотрение многих сетевых графиков (объектов проведения работ), которые должны быть выполнены в заданной последовательности.

Изложение материала настоящей статьи следующее. Во втором разделе приводится алгоритм построения дерева кратчайших путей, используемый далее. В третьем разделе описывается метод учета располагаемых ресурсов, применяемый при построении сетевых графиков выполнения работ. Основной алгоритм расчета сетевых графиков приведен в четвертом разделе. Статья завершается заключительными комментариями.

### АЛГОРИТМ НАХОЖДЕНИЯ ДЕРЕВА КРИТИЧЕСКИХ ПУТЕЙ

Ниже приводится известный алгоритм нахождения дерева кратчайших путей сетевого графика. В дальнейшем он будет модифицирован с учетом располагаемых ресурсов для выполнения работ. В данном алгоритме предполагается, что длительности выполнения работ  $\{d(e)\}$  известны и ресурсов достаточно, с тем чтобы начать каждую работу в нужное время.

*Вход.* Сетевой график  $G = (V, E)$  и матрица длин дуг  $Dwork = (d_{ij})$ , где  $i$  и  $j$  – номера вершин, а  $(i, j)$  – дуга из вершины  $v_i$  в вершину  $v_j$ . Предполагается, что нумерация вершин правильная:  $i < j$  для любой дуги  $(i, j)$ .

*Выход.* Кратчайшие пути от начальной вершины  $v_0$  до всех вершин, достижимых из  $v_0$ , и их длины.

*Метод.* Вершины  $v_j \in V$  просматриваются в порядке их нумерации. Для очередной вершины с номером  $j$  для всех предшествующих ей вершин уже известны наибольшие длины до них от начальной вершины. Это позволяет определить длину критического пути до вершины  $v_j$ . В алгоритме используются метки вершин  $\{L_j\}$ , равные длинам кратчайших путей до вершин. Используется также процедура  $Prec(j, G)$  (от английского слова *preceding* – предшествующий), выдающая множество вершин графа  $G$ , из которых ведут дуги в вершину  $v_j$ .

*Алгоритм.*

*Шаг 0.* Полагаем  $L_0 = 0, j = 1$ .

*Шаг j.* Организуем цикл по вершинам из множества  $Prec(j, G)$ . Для очередной вершины  $v_i$  определяем, имеется ли дуга из вершины  $v_i$  в вершину  $v_j$ . Если нет, то переходим к следующей вершине из множества  $Prec(j, G)$ . Если да и  $d_{i,j}$  есть длина этой дуги, то вычисляем длину пути до  $v_j$ , проходящего через вершину  $v_i$ :

$$\tilde{L}_j = L_i + d_{i,j}.$$

Наибольшее из этих значений для  $v_i \in Prec(j, G)$  и дает длину кратчайшего пути до вершины  $v_j$ . Дуга, соответствующая этой длине, вводится в дерево кратчайших путей.

Если рассмотрены не все вершины, то переход к шагу  $j = j + 1$ . Иначе – конец работы алгоритма.

### УЧЕТ РАСПОЛАГАЕМЫХ РЕСУРСОВ ДЛЯ ВЫПОЛНЕНИЯ РАБОТ

Располагаемые ресурсы для выполнения работ аналитически описываются функциями  $g(t, \rho)$ , где  $t$  – время суток,  $\rho$  – вид ресурса. В компьютерных программах для каждого фиксиро-

ванного вида ресурса  $\rho$ , функция  $g(t, \rho)$  представлена двумерной матрицей  $gg\rho = (gg\rho_{t,r})$ , в которой строки соответствуют различным единицам ресурса (бригадам, средствам механизации и т. п.), а столбцы – моментам времени суток, равноудаленным на время  $\Delta$  (например, 5 минут). Будем далее единицу ресурса называть бригадой. Сама матрица содержит располагаемые объемы ресурсов, дифференцированные по отдельным единицам ресурса (бригадам) и времени суток.

В процессе работы алгоритмов матрица  $gg\rho$  корректируются путем уменьшения объемов располагаемых ресурсов в результате их использования при выполнении работ на объектах.

Опишем алгоритм определения времени начала  $t^*$  и окончания  $t^*$  конкретной работы  $e = (i, j)$ , требующей ресурса  $\rho$ . Предполагается, что уже известны времена окончания всех работ, предшествующих событию  $i$ . Максимальное из них дает минимально возможное время начала выполнения работы  $e = (i, j)$ . Обозначим его  $\tau$ .

Алгоритм содержит два цикла: внешний – по бригадам (им соответствуют строки матрицы  $gg$ ) и внутренний – по времени суток (им соответствуют столбцы этой матрицы).

*Цикл по времени суток для фиксированной бригады.* В специальном массиве указывается место нахождения бригады. На основании этого вычисляется время  $\Delta 1$ , необходимое для перемещения бригады к месту нахождения рассматриваемого объекта. Началом просмотра соответствующей строки матрица  $gg(t, \rho)$  является момент времени  $\tau - \Delta 1$ . Время  $\Delta 2$  выполнения работы  $(i, j)$  бригадой берется из типового сетевого графика  $G$ . Находится первый интервал длины  $\Delta = \Delta 1 + \Delta 2$ , в течение которого бригада свободна. Запоминается начало этого интервала  $t^*$  и номер бригады.

*Цикл по бригадам.* После просмотра всех бригад на работу назначается бригада, для которой время начала  $t^*$  наименьшее. Время окончания выполнения работы  $e = (i, j)$  определяется как  $t^* = t^* + \Delta$ . После этого корректируется строка массива  $gg$ , соответствующая выбранной бригаде. А именно, в соответствующих тактах времени будут записаны номер объекта и номер выполненной работы. Для тактов, соответствующих времени перемещения бригады, эта информация записывается со знаком минус. Кроме того, в специальном массиве записывается новое местонахождение бригады, в котором она оказывается после выполнения рассматриваемой работы.

*Вычисление срока наступления события.* После рассмотрения всех работ  $e_i = (i, j)$ , завершающихся событием  $j$  (т. е. всех  $i$  из множества  $Prec(j, G)$ ), срок наступления этого события  $T(j)$  вычисляется как наибольшее значение из времени окончания этих работ  $t^*$ .

## АЛГОРИТМ РАСЧЕТА СЕТЕВОГО ГРАФИКА

Вышеизложенное позволяет рассчитать весь сетевой график выполнения работ. Он состоит в последовательном рассмотрении работ  $(i, j)$  в порядке, при котором более часто меняется значение  $j$ :  $(0, 0)$ ,  $(0, 1)$ ,  $(0, 2)$ , ...,  $(1, 2)$ ,  $(1, 3)$ ,  $(1, 4)$ , ... Как и ранее,  $T(j)$  есть время наступления события  $j$  данного сетевого графика. Значение  $T(j)$  является заданным – оно было рассчитано в предыдущем разделе. Это есть самый ранний возможный срок начала работ, начинающихся с события  $j$ . Тогда алгоритм можно записать так.

*Алгоритм*

*Шаг 0.* Полагаем  $\tau = T(0)$ ,  $j = 1$ .

*Шаг j.* Организуем цикл по вершинам из множества  $Prec(j, G)$ . Для очередной вершины  $v_i$  определяем дугу  $(i, j)$  из графика  $G$ . Согласно предыдущему разделу, вычисляем время начала  $t^*$  и окончания  $t^*$  работы  $(i, j)$ . Из значений  $t^*$ , полученных для всех вершин из  $Prec(j, G)$ , выбирается наибольшее. Оно и принимается в качестве времени наступления  $j$ -го события  $T(j)$ . Если все вершины сетевого графика рассмотрены, то конец работы алгоритма. Иначе полагаем  $j = j + 1$  и переходим на шаг  $j$ .

Этот алгоритм может применяться для планирования работ не на одном, а на многих объектах. Конечно, каждый объект может иметь свой сетевой график обслуживания. Проведе-

ние работ каждого графика учитывается в содержании матриц  $\{gg(t, \rho)\}$ , в которые постоянно вводится информация об используемых ресурсах.

### ПРАКТИЧЕСКОЕ ПРИМЕНЕНИЕ РАЗРАБОТАННЫХ АЛГОРИТМОВ

Разработанные алгоритмы реализованы в пакете прикладных программ автоматизированной системы комплексного обслуживания рейсов в аэропорту (КОБРА) [4–6]. Этот пакет разработан Региональным информационно-вычислительным центром (РИВЦ) – Пулково для целей оперативного управления бригадами аэропорта в сбойных ситуациях.

Важнейшим исходным массивом является массив *Air*, содержащий информацию об очередности обслуживания вылетающих самолетов и их характеристиках. Структура массива представлена в табл. 1. Первая строка таблицы содержит порядковые номера самолетов, вторая – их типы, третья – вместимость самолетов (количество пассажиров на борту), четвертая – номера мест стоянки самолета на аэродроме. Например, для десятого по порядку самолета тип  $AirList_{2,10} = 3$ , вместимость –  $AirList_{3,10} = 120$  пассажиров, место стоянки  $AirList_{4,10} = 4$ .

Фрагмент типового сетевого графика обслуживания  $G = (V, E)$  и соответствующая матрица длин дуг  $Dwork = (d_{i,j})$  представлены массивом списка работ *List* (табл. 2). Столбцы соответствуют работам. Столбец последовательно содержит следующие показатели, записываемые в отдельных строках.

Таблица 1  
Table 1

Массив *AirList*  
Array of *AirList*

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	3	4	2	1	5	5	4	2	3	3	4	2	5	3
20	100	230	88	51	204	210	112	98	120	98	134	96	211	89
1	3	5	7	9	11	13	15	2	4	6	8	10	12	14

1. Порядковый номер работы.
2. Номер начального события *i*.
3. Номер заключительного события *j*.
4. Код потребляемого ресурса *ρ*.
5. Объем потребляемого ресурса  $h_{i,j}$ .
6. Длительность выполнения операции  $d_{i,j} = t_{i,j} - \bar{t}_{i,j}$ .

Первые четыре показателя являются идентификатором операции, а два последних – ее характеристиками.

Таблица 2  
Table 2

Массив списка работ *List*  
Array of *List* operations list

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	1	2	2	2	2	3	4	4	5	6	6	6	7
2	3	4	3	5	6	6	8	7	8	10	7	8	9	10
1	1	2	2	3	4	6	5	5	6	6	4	4	5	2
2	1	1	1	1	1	1	1	2	1	1	1	2	1	1
18	12	19	14	16	19	18	12	39	10	17	17	22	18	18

Двумерные матрицы  $gg\rho = (gg\rho_{t,r})$ , описывающие состояния располагаемых ресурсов для выполнения различных работ, описаны в табл. 3. Столбцы соответствуют интервалам времени суток  $t$ , а строки – конкретным единицам ресурсам  $r$  – бригадам по нашей терминологии. Значение содержимого одной клетки следующее. Ноль означает, что в это время ресурс отсутствует (бригада не работает), а  $-1$  означает, что он имеется в наличии, но не используется. Трехзначное  $ZYX$  или четырехзначное  $WZYX$  число означает, что на самолете с номером  $Z$  или  $WZ$  проводится работа с номером  $YX$ . Ранее в табл. 2 было приведено описание работы с этими номерами. Знак минус перед числами  $ZYX$  или  $WZYX$  означает, что ресурс перемещается к месту стоянки самолета. В каждой строке предпоследняя клетка содержит текущее местонахождение бригады, записанное с обратным знаком. Последняя клетка содержит момент времени  $TT$  окончания последней выполненной бригадой работы на самолете с номером  $Z$  или  $WZ$ . Формат записи следующий:  $ZTT$  или  $WZTT$ .

Таблица 3  
Table 3

Матрица состояния ресурса  $gg1 = (gg1_{t,r})$ ,  
Resource state matrix  $gg1 = (gg1_{t,r})$

$r/t$	1	2	3	4	5	6	7	8
1	-1	-101	101	101	101	-202	202	202
2	-1	-102	102	102	-201	201	201	201
3	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0

Далее будут представлены некоторые формы выходной информации. В следующей таблице указано время наступления событий сетевого графика  $List$  для конкретного самолета.

Таблица 4  
Table 4

Время наступления событий сетевого графика  $List$  для самолета  $Reis = 5$   
The start time of events for  $List$  network diagram for the aircraft  $Reis = 5$

$No$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$t_k$	14	22	38	51	38	44	66	53	66	73	71	74	75	77	80

Табл. 5 содержит время начала (вторая строка) и окончания (третья строка) обслуживания самолетов списка  $Airlist$ .

Таблица 5  
Table 5

Время начала  $t_0$  и окончания  $t_k$  обслуживания самолетов списка  $Airlist$   
The start time  $t_0$  and the termination time  $t_k$  of aircraft maintenance for the array of  $Airlist$

$No$	1	2	3	4	5	6	7	8
$t_0$	3	6	10	13	14	21	26	27
$t_k$	42	50	70	75	80	100	102	140

Табл. 6 содержит время начала  $t_0$  и окончания  $t_k$  работ (с номерами  $No$ ) на конкретном самолете.

Таблица 6  
Table 6

Время начала  $t_0$  и окончания  $t_k$  выполнения работ на самолете  $Reis = 5$   
The start time  $t_0$  and the termination time  $t_k$  of the aircraft  $Reis = 5$  operations

$No$	1	2	3	4	5	6	7	8	9	10
$t_0$	20	14	49	37	30	42	40	40	60	53
$t_k$	22	15	51	38	31	44	42	41	66	53
$No$	11	12	13	14	15	16	17	18	19	20
$t_0$	40	46	46	46	71	65	66	55	62	50
$t_k$	42	48	48	48	74	66	67	56	64	51
$No$	21	22	23	24	25	26	27	28	29	30
$t_0$	50	69	69	75	73	74	73	73	77	79
$t_k$	51	71	71	77	74	75	74	74	79	80

### ЗАКЛЮЧЕНИЕ

В статье изложен метод расчета сетевых графиков выполнения работ, учитывающий наличие располагаемых ресурсов. Этот метод применим как для обслуживания отдельного объекта, так и для потока объектов. Разработано программное обеспечение представленных алгоритмов, реализованное в контуре оперативного управления АС «Кобра» и предназначенное для планирования обслуживания задержанных рейсов в аэропорту в сбойных ситуациях.

### СПИСОК ЛИТЕРАТУРЫ

1. Hoare H.R. Project Management Using Network Analysis. McGraw-Hill. London, 1973, 120 p.
2. Mukherjee S., Basu K. Solution of interval PERT/CPM network problems by a simplified tabular method. OPSEARCH. Vol. 48, issue 4, 2011, pp. 355–370.
3. Vanhoucke M. Project Management with Dynamic Scheduling. Springe. Berlin, 2012, 310 p.
4. Кофман А., Дебазей Г. Сетевые методы планирования и их применение. М.: Прогресс, 1969.
5. Ore O. Theory of Graphs. American Mathematical Society. Providence, 1962, 270 p.
6. Christofides N. Graph Theory. An Algorithmic Approach. Academic Press Inc., 1975, 415 p.
7. Andronov A.M., Rebezova M.I. Polynomial approximation of the activity time distribution in network diagrams. Automatic control and computer sciences, 2013, vol. 47, № 4, pp. 192–201.
8. Автоматизация производственной и финансово-экономической деятельности предприятий гражданской авиации: учебное пособие / Г.В. Головченко, А.В. Губенко, Э.И. Махарев, М.Ю. Смуров. М.: Спутник, 2016. 348 с.
9. Головченко Г.В. К задаче оптимизации оперативного управления самолетыльетами в сбойных ситуациях в аэропорту // Вестник Санкт-Петербургского государственного университета гражданской авиации. 2016. № 1 (10). С. 98–106.
10. Головченко Г.В. Метод оптимизации очередности подготовки самолетов к вылету при сбойных ситуациях // Вестник Санкт-Петербургского государственного университета гражданской авиации. 2016. № 2 (11).

### СВЕДЕНИЯ ОБ АВТОРАХ

Головченко Глеб Валентинович, директор по развитию автоматизированных технологий авиапредприятий РИВЦ-Пулково, g.golovchenko@rivc-pulkovo.ru.

Ребезова Марина Ильинична, начальник отдела по работе с нерезидентами РФ ПАО «Транспортная Клиринговая Палата», rebezova@tch.ru.

## NET GRAF CALCULATION METHOD FOR JOB COMPLETION CONSIDERING OF RESOURCES AVAILABALE

Gleb V. Golovchenko<sup>1</sup>, Marina I. Rebezova<sup>2</sup>

<sup>1</sup> St. Petersburg State University of Civil Aviation, Saint Petersburg, Russia

<sup>2</sup>Transport Clearing House, Moscow, Russia

### ABSTRACT

The research considers the activity based network of routines performed by the integrated servicing crews at the airport. The known algorithms for critical path claiming suppose that the activity duration is given and the beginning of any activity is determined. However, the beginning of the activity also depends on the presence of the necessary resources for the given period of time.

The formal statement of the problem is described. For the solution of the problem the well-known algorithm for constructing the tree of the shortest paths of the network graph is used. Herewith we suppose, that activity durations are known and resources are sufficient in order to start each activity at the right time. Further, this algorithm is modified in order to take into account available resources. The special two-dimensional matrix with rows showing resource units and columns denoting the equally separated time intervals is used to demonstrate the resources state. The numerical method is proposed for solving the optimization problem. The method is implemented using the Mathcad computing environment and validated by specific examples.

The method is implemented as a complex automated solution for operational management of airport services in the critical situations.

**Key words:** network graph, critical path, scarce resources, aero navigation service.

### REFERENCES

1. **Hoare H.R.** Project Management Using Network Analysis. McGraw-Hill. London, 1973, 120 p.
2. **Mukherjee S., Basu K.** Solution of interval PERT/CPM network problems by a simplified tabular method. OPSEARCH, 2011, vol. 48, issue 4, pp. 355–370.
3. **Vanhoucke M.** Project Management with Dynamic Scheduling. Springe. Berlin, 2012, 310 p.
4. **Kofman A., Debazej G.** *Setevye metody planirovaniya i ih primenenie* [Network planning methods and their application]. Moskva, Progress, 1969. (in Russian)
5. **Ore O.** Theory of Graphs. American Mathematical Society. Providence, 1962, 270 p.
6. **Christofides N.** Graph Theory. An Algorithmic Approach. Academic Press Inc., 1975, 415 p.
7. **Andronov A.M., Rebezova M.I.** Polynomial approximation of the activity time distribution in network diagrams. Automatic control and computer sciences, 2013, vol. 47, № 4, pp. 192–201.
8. **Golovchenko G. V., Gubenko A. V., Maharev Je. I., Smurov M. Ju.** *Avtomatizacija proizvodstvennoj i finansovo-jekonomicheskoy dejatel'nosti predpriyatij grazhdanskoj aviatsii* [Automation of Industrial, Financial and Economic activities of Civil Aviation enterprises]. M., Sputnik, 2016. 348 p. (in Russian)
9. **Golovchenko G.V.** *K zadache optimizacii operativnogo upravlenija samoletovyletami v sbojnyh situacijah v ajeroportu* [To the task of optimizing the operational management of aircraft departures in bad situations at an airport]. *Vestnik Sankt-Peterburgskogo gosudarstvennogo universiteta grazhdanskoj aviatsyii*, 2016, no. 1 (10), p. 98–106. (in Russian)
10. **Golovchenko G.V.** *Metod optimizatsyii ocherjodnosti podgotovki samolyotov k vyiletu pri sbojnyh situatsiyah* [Method for optimizing the order of preparation of aircraft for take-off in case of bad situations]. *Vestnik Sankt-Peterburgskogo gosudarstvennogo universiteta grazhdanskoj aviatsyii*, 2016, no. 2 (11). (in Russian)

### INFORMATION ABOUT THE AUTHORS

**Gleb V. Golovchenko**, Director for the Development of Automated Technologies for Airlines, RIVC-Pulkovo, g.golovchenko@rivc-pulkovo.ru.

**Marina I. Rebezova**, Head of Non-Residents Department, PJSC "Transport Clearing House", rebezova@tch.ru.

Поступила в редакцию 15.03.2017  
Принята в печать 27.04.2017

Received 15.03.2017  
Accepted for publication 27.04.2017