

УДК 621.396.96

## **Алгоритмы решения задачи нумерации отождествленных целей в информационно-управляющих системах**

**Савченко Д. И.**

*Научно-производственная фирма «Меридиан», ул. Блохина, 9, Санкт-Петербург, 197198, Россия  
e-mail: cfdX@list.ru*

### **Аннотация**

Статья посвящена задаче нумерации отождествленных целей в информационно-управляющих системах различных носителей. Рассмотрены требования, которые должны удовлетворяться для получения устойчивой системы нумерации, облегчающей оценку обстановки оператором и обеспечивающей непрерывность решения задач в информационно-управляющих системах. Предложена совокупность алгоритмов решения задачи нумерации.

**Ключевые слова:** информационно-управляющая система, отождествление, источники информации, целевая обстановка, третичная обработка, нумерация целей

### **Введение**

Многие современные носители оснащаются целым рядом источников информации (ИИ) об обстановке (радиолокационные станции различного назначения, оптико-электронные средства и т.д.). Для повышения эффективности освещения обстановки, увеличения полноты и достоверности используемых данных о целях при решении задач в информационно-управляющих системах (ИУС) этих носителей обычно используется информация не от отдельных ИИ, а объединенные данные всех ИИ. Для этого в ИУС реализуется задача отождествления целевой информации (ЗО) от всех ИИ [1, 2]. Назначением ЗО является принятие решений о тождественности или нетождественности данных о целях (формуляров целей, ФЦ), поступающих от разных ИИ, то есть о принадлежности или непринадлежности этих данных одним и тем же реальным объектам. Решение ЗО, как правило, осуществляется регулярно с постоянным периодом, величина которого для современных ИУС составляет порядка  $10^{-1}..10^1$  секунд. Выходными данными каждого вызова ЗО является массив отождествленных ФЦ (ОФЦ). При этом в общем случае

количество и состав ОФЦ (то есть список ФЦ от ИИ, в результате отождествления которых был получен каждый ОФЦ), сформированных на очередном вызове ЗО, не совпадает со сформированными на предыдущем вызове. Это объясняется различием тактико-технических характеристик ИИ, случайным характером процесса сопровождения целей ИИ, ошибочными отождествлениями и неотождествлениями ФЦ от ИИ и т.д. Продолжительность решения задач (выполняемых оператором или в автоматическом режиме) в ИУС часто существенно превышает период решения ЗО. Поэтому возникает необходимость поддержания системы идентификаторов ОФЦ, называемых номерами целей в системе нумерации (СН) ИУС, которые бы, по возможности, оставались постоянными с течением времени, при разотождествлении/отождествлении ранее отождествляемых/неотождествляемых ФЦ от ИИ, и давали возможность непрерывного решения задач в ИУС в условиях изменяющего состава ОФЦ. Рассмотрению требований к СН ИУС и разработке алгоритмов решения задачи нумерации целей посвящена настоящая статья.

### **1 Постановка задачи. Основные требования к алгоритму нумерации**

Алгоритм нумерации является основным в совокупности алгоритмов решения задачи нумерации целей в СН ИУС. Частота его выполнения (вызовов) совпадает с частотой решения ЗО. Входными данными для алгоритма нумерации является массив ОФЦ  $trgs$ , формируемый ЗО и содержащий  $trgs\_cnt$  элементов, а выходными – тот же массив, в элементы которого (в ОФЦ) записаны присвоенные номера в СН ИУС.

Исходя из вышесказанного, данный алгоритм должен решать две задачи:

- а) восстановление номеров в СН ИУС, присвоенных ОФЦ на предыдущем вызове;
- б) присваивание новых номеров в СН ИУС для ОФЦ, которым не удалось восстановить присвоенные на предыдущем вызове номера. Это может иметь место, при взятии новой цели на сопровождение или если количество ОФЦ увеличилось в результате принятия решения о нетождественности ранее отождествляемых ФЦ от ИИ.

В контексте рассматриваемой задачи каждый ОФЦ может быть формально представлен в виде упорядоченной пары:

$$f_i = (N_{sys,i}, N_{src,i}), \quad (1)$$

где

$N_{sys,i} \in \{FN_{sys}, \dots, EN_{sys}\} \cup \{NN\}$  – номер в СН ИУС, который присваивается ОФЦ в результате выполнения алгоритма нумерации целей;

$N_{src,i} = (n_{src,i}^0, n_{src,i}^1, \dots, n_{src,i}^{SC-1})$  – упорядоченная по идентификаторам ИИ последовательность (массив) номеров в СН ИИ из ФЦ от ИИ, в результате отождествления которых получен  $i$ -ый ОФЦ;

$n_{src,i}^j \in \{FN_{src,j}, \dots, EN_{src,j}\} \cup \{NN\}$  – номер в СН  $j$ -ого ИИ;

$FN_{sys}, EN_{sys}$  – первый и последний номера в СН ИУС, задающие диапазон номеров, используемых для нумерации в СН ИУС;

$FN_{src,j}, EN_{src,j}$  – первый и последний номера в СН  $j$ -ого ИИ, задающие диапазон номеров, используемых для нумерации  $j$ -ым ИИ;

$NN \notin \{FN_{src,j}, \dots, EN_{src,j}\} \cup \bigcup_{j=0}^{SC-1} \{FN_{src,j}, \dots, EN_{src,j}\}$  – признак отсутствия номера в СН ИУС

или СН ИИ. Если для  $i$ -ого ОФЦ и  $j$ -ого ИИ  $n_{src,i}^j = NN$ , это означает, что соответствующая данному ОФЦ цель не сопровождается  $j$ -ым ИИ (ФЦ от данного ИИ не имел места среди ФЦ от ИИ, в результате отождествления которых был получен данный ОФЦ);

$SC$  – количество ИИ, данные от которых участвуют в отождествлении.

Для присвоенных ОФЦ номеров в СН ИУС должно выполняться условие:

$$\forall i \neq j, N_{sys,i} \neq NN : N_{sys,i} \neq N_{sys,j} . \quad (2)$$

То есть в массиве ОФЦ не может быть двух ОФЦ, которым присвоен один и тот же номер в СН ИУС (за исключением значения  $NN$ ).

Для номеров в СН ИИ из всех ОФЦ выполняются следующие условия:

$$\forall i \neq k, \forall j \in \{0, \dots, SC-1\}, n_{src,i}^j \neq NN : n_{src,i}^j \neq n_{src,k}^j , \quad (3)$$

$$\forall i, \exists j \in \{0, \dots, SC-1\} : n_{src,i}^j \neq NN . \quad (4)$$

Условие (3) означает, что один и тот же номер в СН некоторого ИИ не может иметь место сразу в двух ОФЦ. Условие (4) означает, что в каждом ОФЦ имеется хотя бы один номер в СН ИИ отличный от  $NN$ , то есть существует хотя бы один ФЦ от ИИ, на основе которого был сформирован данный ОФЦ.

Количество номеров в СН ИУС далее обозначается  $NC_{sys}$ , а количество номеров в СН  $j$ -ого ИИ –  $NC_{src,j}$ :

$$NC_{sys} = EN_{sys} - FN_{sys} + 1, \quad (5)$$

$$NC_{src,j} = EN_{src,j} - FN_{src,j} + 1. \quad (6)$$

Следует отметить, что в некоторых случаях (например, если количество ОФЦ превышает количество номеров, используемых для нумерации в СН ИУС) не всем ОФЦ могут быть восстановлены или присвоены новые номера в СН ИУС. В таких ситуациях алгоритм нумерации должен устанавливать признак отсутствия номера  $NN$  в соответствующие ОФЦ.

### 1.1 Восстановление присвоенных на предыдущем вызове номеров в СН ИУС

При первом вызове ЗО и алгоритма нумерации всем ОФЦ из массива ОФЦ могут быть присвоены произвольные неповторяющиеся номера в СН ИУС из диапазона  $\{FN_{sys}, \dots, EN_{sys}\}$  (см. п. 1.2).

Рассмотрим  $l$ -ый ( $l > 1$ ) вызов ЗО. На предыдущем  $(l-1)$ -ом вызове был сформирован массив ОФЦ, элементам которого алгоритмом нумерации были присвоены номера в СН ИУС. Каждый ОФЦ из этого массива содержал один или более отличных от  $NN$  номеров в СН ИИ (см. формулы (1) и (4)). Поэтому можно говорить о том, что номерам в СН ИИ (если эти номера использованы ИИ для нумерации сопровождаемых в данный момент целей) на  $(l-1)$ -ом вызове были поставлены в соответствие номера в СН ИУС (см. рисунок 1). Более строго, для каждого номера в СН каждого ИИ, задействованного данным ИИ для нумерации сопровождаемых целей, на  $(l-1)$ -ом вызове сформирована упорядоченная тройка:

$$k = (j, n_{src}, N_{sys}), \quad (7)$$

где

$j \in \{0, \dots, SC - 1\}$  – идентификатор ИИ;

$n_{src} \in \{FN_{src,j}, \dots, EN_{src,j}\}$  – номер в СН  $j$ -ого ИИ;

$N_{sys}$  – номер в СН ИУС, который был присвоен ОФЦ, полученному в результате отождествления ФЦ от  $j$ -ого ИИ, содержащего номер цели  $n_{src}$  в СН ИИ.

Все упорядоченные тройки  $k$  образуют множество  $K$ . Исходя из способа формирования этого множества и условия (3), для его элементов выполняется условие:

$$\forall (i, n, N) \in K, N \neq U : (i, n, U) \notin K. \quad (8)$$

То есть в множестве  $K$  не могут иметься два элемента (тройки  $(j, n_{src}, N_{sys})$ ) с одной и той же парой значений  $j$  и  $n_{src}$  (номеру в СН ИИ не может быть поставлено в соответствие два номера в СН ИУС).

Текущий ( $l$ -ый) вызов ЗО сформировал новый массив ОФЦ, каждый элемент которого (каждый ОФЦ) также содержит номера (один или более) в СН ИИ (см. формулы (1) и (4)), которым на  $(l-1)$ -ом вызове (если они использовались ИИ для нумерации сопровождаемых целей на предыдущем вызове) были поставлены в соответствие номера в СН ИУС. Информация об этом соответствии представлена множеством  $K$  (после исключения из него троек  $(j, n_{src}, N_{sys})$ , если  $j$ -ый ИИ снял цель с номером  $n_{src}$  в его СН с сопровождения). То есть для каждого  $i$ -ого ОФЦ существует некоторое множество  $X_i$  номеров в СН ИУС, которые были поставлены в соответствие содержащимся в данном ОФЦ номерам в СН ИИ на  $(l-1)$ -ом вызове. Для множества  $X_i$  с использованием множества  $K$  формально можно записать (см. рисунок 1):

$$X_i = \{N_{sys} \mid (j, n_{src}, N_{sys}) \in K \ \& \ n_{src,i}^j = n_{src}\}. \quad (9)$$

Под восстановлением номеров в СН ИУС понимается выбор для каждого  $i$ -ого ОФЦ одного из номеров, принадлежащих  $X_i$ , для присваивания этому ОФЦ на текущем вызове алгоритма нумерации. Для выполнения условия (2), по мере осуществления процесса восстановления номеров в СН ИУС из множеств  $X_i$ , должны исключаться номера, присвоенные (восстановленные) другим ОФЦ.

В [3] показано, что для реализации процесса восстановления номеров в СН ИУС алгоритм нумерации должен поочередно рассматривать элементы множества  $K$ , выбирая очередной элемент (тройку  $(j, n_{src}, N_{sys})$ ) исходя из следующей системы критериев:

1) наличие или отсутствие «жесткой связи» между номером  $N_{sys}$  в СН ИУС и номером  $n_{src}$  в СН  $j$ -ого ИИ;

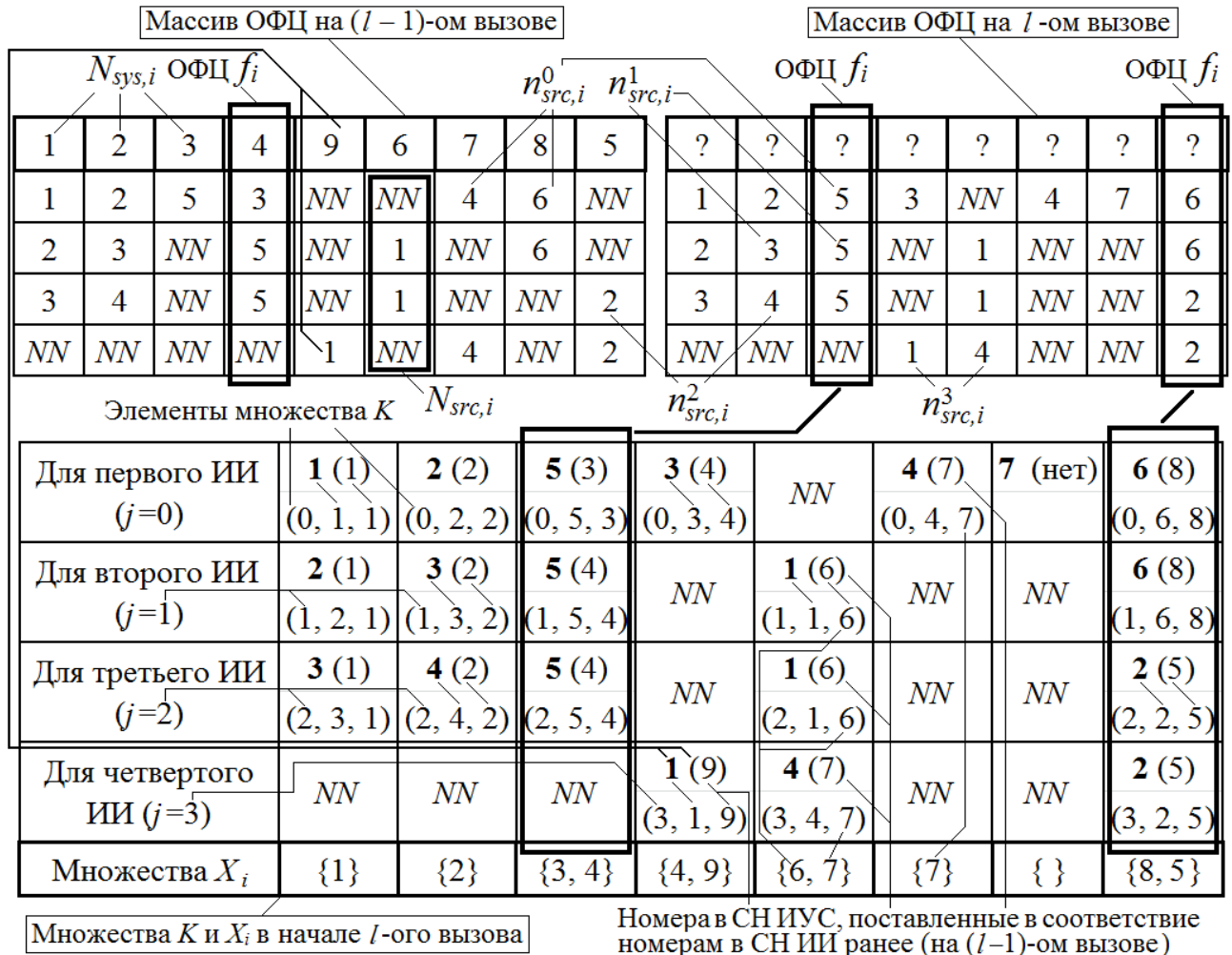


Рисунок 1 – Содержание множеств  $X_i$  и  $K$  ( $SC = 4$ )

- 2) приоритет  $p \in \{0, \dots, P_{\max}\}$  восстановления номера  $N_{sys}$  в СН ИУС;
- 3) максимальность числа  $T$  подряд идущих вызовов алгоритма нумерации, в течение которых в множестве  $K$  имела место тройка  $(j, n_{src}, N_{sys})$ ;
- 4) приоритет  $j$ -ого ИИ.

Обоснование использования указанных критериев приводится далее. Перечисленные критерии выбора очередной тройки  $(j, n_{src}, N_{sys})$  представлены в порядке их приоритетности. То есть сначала алгоритм должен пытаться выбрать тройку по первому критерию. Если его применение не дало возможности однозначного выбора очередной тройки, то применяется

второй критерий и так далее. Для каждой выбранной тройки, если номер  $N_{sys}$  из нее еще не был восстановлен какому-либо ОФЦ, должно определяться соответствующее множество  $X_i$  (см. формулу (9)) и, если  $i$ -ому ОФЦ еще не был восстановлен номер в СН ИУС, восстанавливается номер  $N_{sys}$   $i$ -ому ОФЦ. Необходимость выполнения указанных проверок (номер  $N_{sys}$  еще не был восстановлен какому-либо ОФЦ,  $i$ -ому ОФЦ не был восстановлен номер) объясняется тем, что множества  $X_i$  в общем случае содержат более одного номера в СН ИУС, что связано с возможностью ложных отождествлений и неотождествлений ФЦ от ИИ на предыдущем или текущем вызове ЗО. Данные проверки моделируют вычеркивание элементов из множеств  $X_i$  и самих уже рассмотренных множеств  $X_i$ .

### 1.1.1 Наличие «жесткой связи» между номером в СН ИУС и номером в СН ИИ

В качестве примера ИУС, где имеется необходимость установки «жесткой связи» между номером в СН ИУС и номером в СН некоторого ИИ, могут выступать боевые ИУС. В этих системах, в случае разотождествления ОФЦ (то есть если на очередном вызове имеется два ОФЦ, которые раньше признавались принадлежащими одной цели) с номером  $N$  в СН ИУС, по которому выдано целеуказание (ЦУ) некоторой системе оружия, необходимо, чтоб номер  $N$  в СН ИУС был присвоен (восстановлен) тому из полученных ОФЦ, в состав которого вошел ФЦ от ИИ, по данным которого выдается ЦУ (см. рисунок 2) [3].

Если в конкретной ИУС нет необходимости в установке такого рода «жестких связей», данный критерий может быть исключен из рассмотрения.

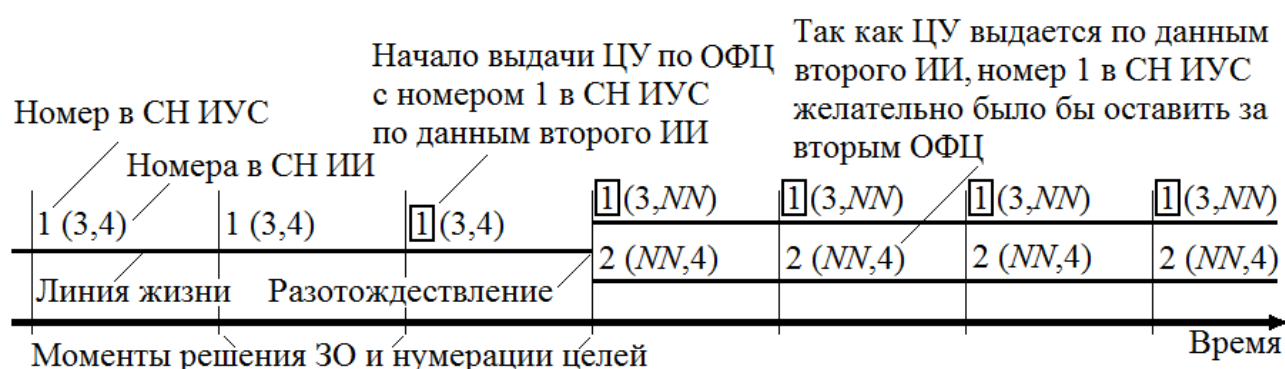


Рисунок 2 – Пример нежелательной ситуации 1 ( $SC = 2$ )

### 1.1.2 Приоритет восстановления номера в СН ИУС

Приоритет  $p \in \{0, \dots, P_{max}\}$  восстановления номера в СН ИУС (по умолчанию равный нулю для всех номеров в СН ИУС) должен устанавливаться номеру в СН ИУС в момент

начала решения одной из задач (особенно решаемых в автоматическом режиме) по данным ОФЦ, которому присвоен этот номер. Значение приоритета должно быть пропорционально негативным последствиям, которые могут иметь место при прекращении решения данной задачи в связи с исчезновением ОФЦ с соответствующим номером в СН ИУС. В простейшем случае  $p \in \{0,1\}$ . Например, по команде оператора к началу решения задачи «Расчет вероятности столкновения с целью» по данным некоторого ОФЦ, номеру в СН ИУС этого ОФЦ может быть присвоен повышенный приоритет. Целесообразность этого обосновывается тем, что в случае отождествления двух и более ранее неотожествившихся ФЦ от ИИ, номерам в СН ИИ которых (формуляров) были поставлены в соответствие номера в СН ИУС, полученному ОФЦ должен быть присвоен (восстановлен) номер в СН ИУС, по которому решается задача. В противном случае, возможно исчезновение ОФЦ с этим номером в СН ИУС, что приведет к прекращению решения задачи, потере всех промежуточных результатов расчетов и необходимости перезапуска задачи. Это, в свою очередь, увеличивает время принятия решений и может крайне негативно сказываться на эффективности ИУС в целом. Пример такой ситуации представлен на рисунке 3.

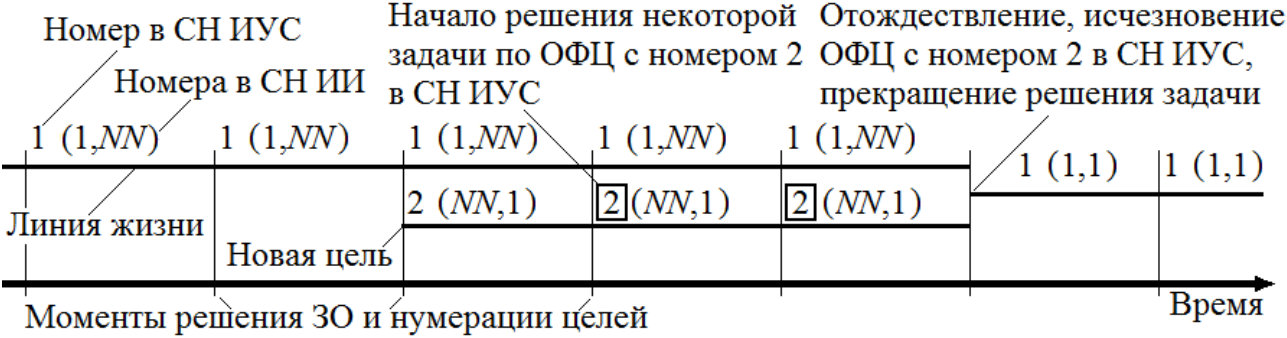


Рисунок 3 – Пример нежелательной ситуации 2 ( $SC = 2$ )

**1.1.3 Число подряд идущих вызовов алгоритма нумерации, в течение которых в множестве  $K$  имел место соответствующий элемент**

В качестве третьего по приоритетности критерия используется максимальность числа  $T$ , которое характеризует количество подряд идущих вызовов алгоритма нумерации, в течение которых номеру  $n_{src}$  в СН  $j$ -ого ИИ ставился в соответствие номер  $N_{sys}$  в СН ИУС из соответствующей тройки  $(j, n_{src}, N_{sys})$ . В случае отождествления двух и более ФЦ от ИИ, номерам в СН ИИ которых (формуляров) ранее были поставлены в соответствие разные номера в СН ИУС, это обеспечит восстановление полученному ОФЦ того номера в СН ИУС, который дольше всего соответствует одному из номеров в СН ИИ отождествленных ФЦ от



ИИ. В противном случае возможно возникновение ситуаций, аналогичных представленной на рисунке 4, которые затрудняют работу оператора.

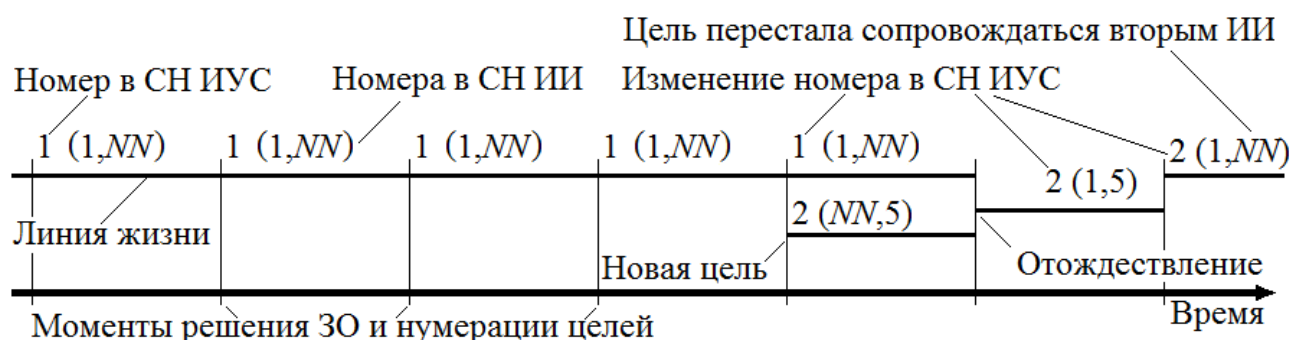


Рисунок 4 – Пример нежелательной ситуации 3 ( $SC = 2$ )

Для построения эффективного, в аспекте времени выполнения, алгоритма, максимальное значение  $T$  можно ограничить некоторой величиной  $T_{\max}$  и считать две тройки в множестве  $K$  (эквивалентные в смысле первых двух критериев), для которых значения  $T > T_{\max}$ , эквивалентными в смысле данного критерия.

#### 1.1.4 Приоритет ИИ

В случае отождествления двух и более ФЦ от ИИ, номерам в СН ИИ которых (формулярам) ранее были поставлены в соответствие номера в СН ИУС, полученному ОФЦ должен быть присвоен номер в СН ИУС, который был поставлен в соответствие номеру в СН наиболее приоритетного ИИ. Этот критерий является наименее приоритетным и служит для снятия неопределенности при наличии нескольких, эквивалентных в смысле первых трех критериев, элементов в множестве  $K$ .

#### 1.2 Присваивание новых номеров в СН ИУС

После выполнения восстановления присвоенных на предыдущем вызове номеров в СН ИУС могут остаться ОФЦ, которым не присвоены (не восстановлены) номера в СН ИУС. Это имеет место при первом вызове ЗО, при взятии новой цели на сопровождение ИИ или при принятии решения о нетождественности ранее отождествлявшихся ФЦ от ИИ. Таким ОФЦ должны быть присвоены новые номера в СН ИУС. Каждый новый присваиваемый номер в СН ИУС должен выбираться из множества неиспользованных (свободных) номеров в СН ИУС (принадлежащих диапазону  $\{FN_{sys}, \dots, EN_{sys}\}$ ) по критерию наибольшего времени его пребывания в этом множестве. Это исключит ситуацию, когда недавно использовавшийся номер в СН ИУС будет присвоен ОФЦ новой цели, что, например, могло бы ввести в заблуждение оператора. Включение номера в СН ИУС в множество

неиспользованных номеров (освобождение номера в СН ИУС) должно осуществляться в случае, если он еще не содержится в этом множестве и на очередном вызове алгоритма нумерации он не присвоен (или восстановлен) ни одному ОФЦ.

## 2 Алгоритмы решения задачи нумерации целей

### 2.1 Алгоритм нумерации

Для представления множества  $K$  в алгоритме нумерации вводится массив  $NumRes$ , содержащий  $NRL$  элементов, где  $NRL$  – суммарное количество номеров в СН всех ИИ:

$$NRL = \sum_{i=0}^{SC-1} NC_{src,i}. \quad (10)$$

Каждый элемент этого массива соответствует одному номеру в СН одного из ИИ, что проиллюстрировано рисунком 5. В общем случае расположение элементов в массиве  $NumRes$  должно быть таким, чтобы номерам в СН более приоритетных ИИ отвечали меньшие индексы в массиве  $NumRes$ . Для простоты изложения, далее считается, что приоритет ИИ равен его идентификатору.

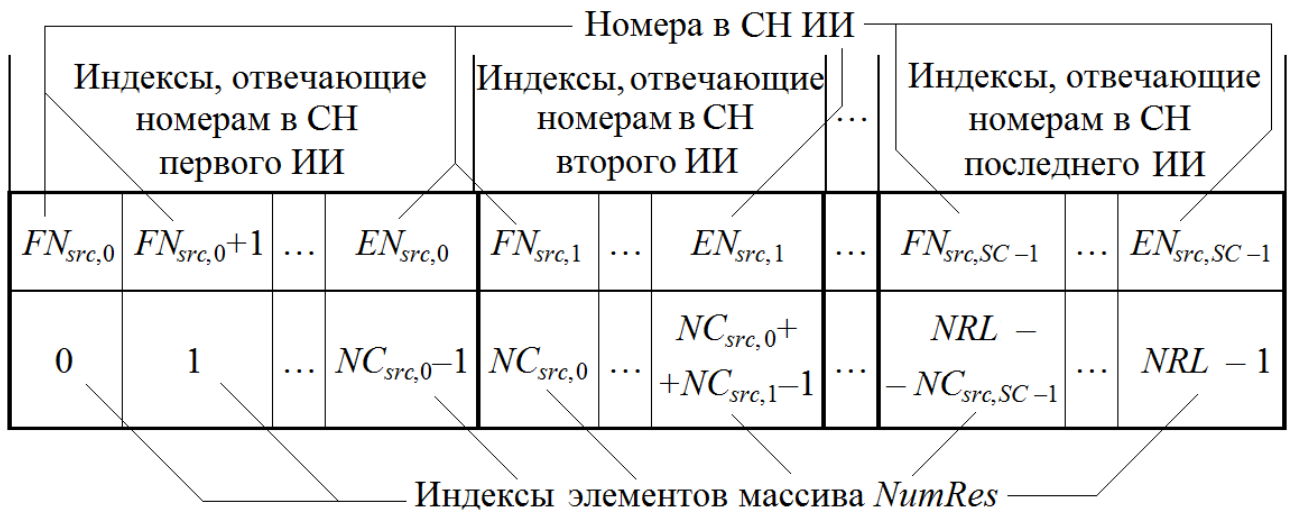


Рисунок 5 – К пояснению соответствия индексов в массиве  $NumRes$  и номеров в СН ИИ

Каждый элемент массива  $NumRes$  имеет структуру, представленную в таблице 1.

Исходя из рисунка 5, индекс  $i = NRI(j, n)$  элемента массива  $NumRes$ , отвечающего номеру  $n$  в СН  $j$ -ого ИИ, может быть получен по следующей формуле:

$$NRI(j, n) = \sum_{i=0}^{j-1} NC_{src,i} + n - FN_{src,j}. \quad (11)$$

Содержание элементов массива *NumRes*

Таблица 1

Содержание поля элемента	Название	Значения
Номер в СН ИУС, поставленный в соответствие номеру в СН ИИ, отвечающему данному элементу, на предыдущем вызове	$N_{sys}$	$\{FN_{sys}, \dots, EN_{sys}\}$ $\cup \{NN\}$
Признак наличия «жесткой связи» между номером $N_{sys}$ в СН ИУС и номером в СН ИИ, отвечающим данному элементу	$RC$	0 – нет, 1 – есть
Число подряд идущих вызовов алгоритма нумерации, в течение которых номеру в СН ИИ, отвечающему данному элементу, ставился в соответствие номер $N_{sys}$ в СН ИУС	$times$	$\{0, \dots, T_{max}\}$
Приоритет восстановления номера $N_{sys}$ в СН ИУС	$pr$	$\{0, \dots, P_{max}\}$

При использовании формулы (11) может быть задействован вспомогательный массив, содержащий  $SC$  элементов,  $j$ -ый элемент которого хранит значение суммы  $\sum_{i=0}^{j-1} NC_{src,i}$ . Это обеспечит константное время расчета по формуле (11) в алгоритмах.

Для любой возможной тройки  $(j, n_{src}, N_{sys})$  в множестве  $K$  имеется элемент массива *NumRes* (с индексом  $NRI(j, n_{src})$ ), отвечающий паре значений  $j$  и  $n_{src}$ . Причем в множестве  $K$  не может быть двух элементов с одной и той же парой  $j$  и  $n_{src}$  (условие (8)). Это обосновывает возможность использования массива *NumRes* для представления множества  $K$  и корректность предлагаемых далее алгоритмов. Если элемент с индексом  $i = NRI(j, n_{src})$  этого массива содержит в поле  $N_{sys}$  значение  $N \neq NN$ , то тройка  $(j, n_{src}, N)$  принадлежит множеству  $K$ . В противном случае (если в поле  $N_{sys}$  содержится  $NN$ ), в множестве  $K$  отсутствует тройка, первые два элемента которой равны  $j$  и  $n_{src}$ .

Дополнительные поля элементов массива *NumRes* ( $RC$ ,  $times$  и  $pr$ ) содержат информацию, необходимую для выбора очередной тройки из множества  $K$  исходя из системы критериев, описанной в п. 1.1.

Массив *NumRes* от вызова к вызову хранится в памяти алгоритма решения задачи нумерации. Изначально (перед первым вызовом рассматриваемого алгоритма), полям  $N_{sys}$

всех элементов массива  $NumRes$  должно быть присвоено значение  $NN$ , а полям  $RC$ ,  $times$  и  $pr$  – нулевые значения.

Кроме рассмотренного массива в алгоритме нумерации используются следующие вспомогательные массивы:

1) массив  $used$ , содержащий  $NC_{sys}$  элементов. Каждый  $i$ -ый элемент этого массива соответствует номеру  $(FN_{sys} + i)$  в СН ИУС и содержит единицу, если этот номер был использован для нумерации ОФЦ на текущем вызове рассматриваемого алгоритма (то есть присвоен или восстановлен какому-либо ОФЦ). Применение данного массива моделирует вычеркивание номеров в СН ИУС из множеств  $X_j$  по мере их присваивания (восстановления) ОФЦ. Вначале работы алгоритма всем элементам данного массива присваиваются нулевые значения;

2) массив  $numbered$ , содержащий  $trgs\_cnt$  элементов, где  $trgs\_cnt$  – количество ОФЦ в массиве ОФЦ, поступившем на вход алгоритма нумерации. Каждый  $i$ -ый элемент этого массива отвечает  $i$ -ому ОФЦ ( $i$ -ому элементу массива ОФЦ) и соответственно множеству  $X_i$  и содержит единицу, если данному ОФЦ уже был восстановлен или присвоен новый номер в СН ИУС на текущем вызове алгоритма нумерации. Применение данного массива моделирует вычеркивание рассмотренных множеств  $X_i$ . Вначале работы алгоритма нумерации всем элементам данного массива присваиваются нулевые значения;

3) массив  $trgs\_ind$ , содержащий  $NRL$  элементов. Каждый  $i$ -ый элемент данного массива соответствует  $i$ -ому элементу массива  $NumRes$  и содержит индекс ОФЦ (в массиве  $trgs$ ), в котором имеется номер  $n_{src}$  в СН  $j$ -ого ИИ такой, что  $NRI(j, n_{src}) = i$ . В случае отсутствия такого ОФЦ (если  $j$ -ый ИИ не использует номер  $n_{src}$  в его СН для нумерации сопровождаемых им в данный момент целей), элемент массива  $trgs\_ind$  содержит значение минус 1. Использование данного массива позволяет за константное время найти ОФЦ, которому должен быть восстановлен номер в СН ИУС на основе очередной выбранной тройки  $(j, n_{src}, N_{sys}) \in K$ . Заполнение данного массива осуществляется вначале каждого вызова алгоритма нумерации;

4) массив  $indexes$ , содержащий не более  $NRL$  элементов. В данный массив записываются индексы элементов массива  $NumRes$ , значения полей  $N_{sys}$  которых различно от  $NN$ . Необходимость применения данного массива обосновывается следующим. Для организации поочередного выбора элементов множества  $K$  исходя из системы критериев,

описанной в п. 1.1, в алгоритме нумерации производится сортировка элементов массива *NumRes* (представляющего множество  $K$ ). Однако расположение элементов этого массива не должно меняться (см. формулу (11), рисунок 5), поэтому вводится вспомогательный массив ссылок на элементы *NumRes* (индексов), к которому можно применять сортировку.

Алгоритм нумерации (см. рисунок 6 а)) включает в себя следующие основные шаги:

1) обнуление всех элементов массивов *used* и *numbered* (оператор 4). Сложность данного шага составляет  $O(trgs\_cnt + NC_{sys})$ ;

2) заполнение массива *trgs\_ind* (оператор 5). Для этого в цикле ( $i = \overline{0, trgs\_cnt - 1}$ ) осуществляется просмотр всех элементов массива ОФЦ *trgs*. Для каждого ОФЦ в цикле ( $j = \overline{0, SC - 1}$ ) осуществляется просмотр всех номеров в СН ИИ и, если очередной номер  $n_{src,i}^j$  в СН ИИ не равен  $NN$ , то в элемент массива *trgs\_ind* с индексом  $NRI(j, n_{src,i}^j)$  заносится значение  $i$ . Перед выполнением данного шага всем элементам массива *trgs\_ind* присваиваются значения минус 1. Сложность шага составляет  $O(trgs\_cnt \cdot SC + NRL)$ ;

3) обнуление элементов массива *NumRes*, отвечающих неиспользуемым номерам в СН ИИ (оператор 6). Для этого в цикле ( $i = \overline{0, NRL - 1}$ ) осуществляется просмотр всех элементов массива *trgs\_ind*. Если очередной элемент равен минус 1, то в поле  $N_{sys}$   $i$ -ого элемента массива *NumRes* заносится значение  $NN$  (исключение тройки из множества  $K$ ), а поля *RC*, *times* и *pr* обнуляются. Сложность данного шага составляет  $O(NRL)$ ;

4) заполнение массива *indexes* (оператор 7). Для этого в цикле ( $i = \overline{0, NRL - 1}$ ) осуществляется просмотр всех элементов массива *NumRes*. Если поле  $N_{sys}$  очередного элемента содержит значение, отличное от  $NN$ , то индекс  $i$  этого элемента записывается в  $I$ -ый элемент массива *indexes*, а значение  $I$  (перед началом выполнения данного шага равное нулю) увеличивается на единицу. Сложность шага составляет  $O(NRL)$ ;

5) Сортировка (трехкратная) использованных, то есть лежащих в диапазоне индексов  $[0, I - 1]$ , элементов массива *indexes* (оператор 8). Применяемый алгоритм сортировки должен быть устойчивым (см. далее). Вначале сортировка осуществляется в порядке убывания значений полей *times*, соответствующих элементов массива *NumRes* (то есть по критерию  $NumRes[indexes[i]].times > NumRes[indexes[j]].times$ ), затем в порядке убывания значений полей *pr*, затем в порядке убывания значений полей *RC*. Пояснения к данному шагу приводятся далее;

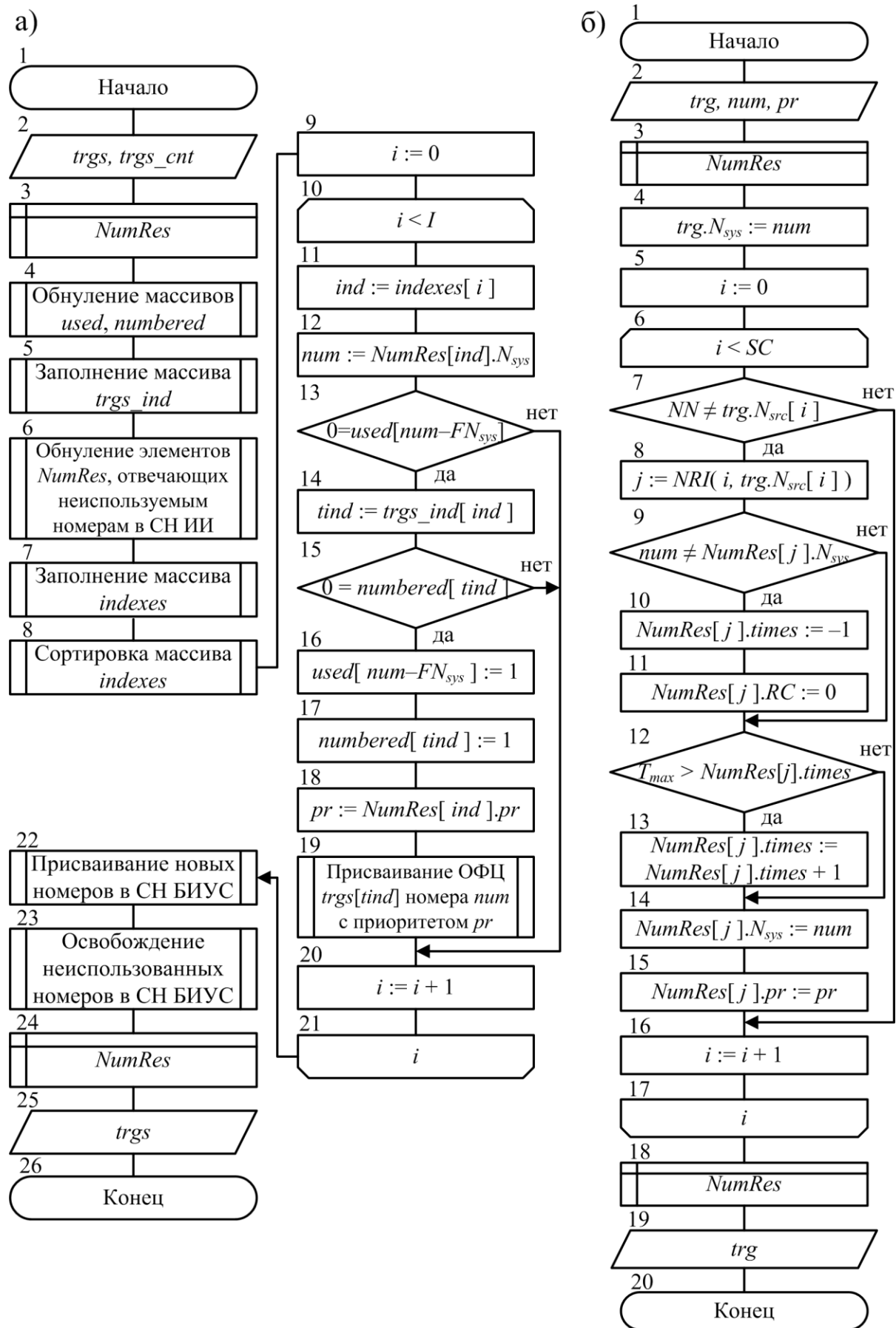


Рисунок 6 – Блок-схемы алгоритма нумерации (а) и процедуры присваивания номера (б)

б) восстановление номеров в СН ИУС для ОФЦ (операторы 9-21). Для этого перебираются элементы массива  $NumRes$  (элементы множества  $K$ ) в порядке следования их индексов в массиве  $indexes$ . Для очередного элемента, если номер в СН ИУС из поля  $N_{sys}$  еще не был восстановлен какому-либо ОФЦ, то определяется индекс ОФЦ в массиве  $trgs$  (с использованием массива  $trgs\_ind$ ), в котором содержится номер в СН ИИ, отвечающий текущему элементу массива  $NumRes$ . Если такой ОФЦ существует и ему не восстановлен номер в СН ИУС, то ему восстанавливается номер в СН ИУС (процедура присваивания номера в СН ИУС ОФЦ представлена на рисунке 6 б)). Так как  $I \leq NRL$ , то сложность данного шага (с учетом сложности вспомогательной процедуры) составляет  $O(NRL \cdot SC)$ ;

7) присваивание новых номеров в СН ИУС ОФЦ, которым не были восстановлены номера в СН ИУС (оператор 22). Для этого в цикле ( $i = \overline{0, trgs\_cnt - 1}$ ) осуществляется просмотр всех элементов массива  $numbered$ . Если  $i$ -ый элемент равен нулю (то есть соответствующему ОФЦ не был восстановлен номер в СН ИУС на шаге б), то выбирается новый свободной номер в СН ИУС (см. п. 2.3), соответствующий этому номеру элемент массива  $used$  устанавливается в единицу и вызывается процедура присваивания номера в СН ИУС  $i$ -ому ОФЦ (с приоритетом восстановления равным нулю  $pr = 0$ ). Сложность данного шага (с учетом сложности процедуры присваивания номера) составляет  $O(trgs\_cnt \cdot SC)$ ;

8) освобождение незадействованных для нумерации ОФЦ на текущем вызове номеров в СН ИУС (оператор 23). Для этого в цикле ( $i = \overline{0, NC_{sys} - 1}$ ) осуществляется просмотр всех элементов массива  $used$ . Если  $i$ -ый элемент равен нулю, то номер ( $FN_{sys} + i$ ) в СН ИУС освобождается (см. п. 2.3). Сложность данного шага составляет  $O(NC_{sys})$ .

Процедура присваивания номера в СН ИУС (см. рисунок 6 б)), входными данными для которой являются ОФЦ  $trg$ , присваиваемый номер в СН ИУС  $num$  и приоритет восстановления этого номера  $pr$ , записывает информацию о постановке в соответствие номера в СН ИУС (и о приоритете восстановления этого номера) в элементы массива  $NumRes$ , отвечающие номерам в СН ИИ из ОФЦ. При этом если номер в СН ИУС совпадает с присвоенным на предыдущем вызове, то увеличивается значение поля  $times$ , в противном случае, значение этого поля обнуляется. Сложность процедуры составляет  $O(SC)$ .

Выполняемая на шаге б сортировка элементов массива  $indexes$ , содержащего индексы элементов массива  $NumRes$ , обеспечивает описанный выше (см. п. 1.1) порядок

рассмотрения элементов множества  $K$ . Сортировка по каждому из четырех критериев производится в порядке обратном приоритету этих критериев. Вкупе с использованием устойчивого алгоритма сортировки, это обеспечивает упорядоченность элементов массива  $indexes$  в порядке наличия/отсутствия признака «жесткой связи» в соответствующих элементах массива  $NumRes$  (критерий 1), для элементов с одинаковым значением этого признака – упорядоченность в порядке убывания приоритета восстановления номеров (критерий 2) и т.д. Явная сортировка по последнему критерию (приоритету ИИ) не производится, так как массив  $indexes$  на шаге 4 заполнялся в соответствии с расположением элементов в массиве  $NumRes$ , упорядоченных в порядке приоритета ИИ (рисунок 5). Так как диапазоны значений признаков для сортировки относительно малы ( $RC \in \{0,1\}$ ,  $pr \in \{0, \dots, P_{\max}\}$ ,  $times \in \{0, \dots, T_{\max}\}$ ), точнее они много меньше количества сортируемых элементов (которое в худшем случае равно  $NRL$ ), то целесообразно использование алгоритма сортировки подсчетом [4]. Данный алгоритм является устойчивым, обеспечивает сложность  $O(n+k)$ , где  $n$  – количество сортируемых элементов,  $k$  – количество значений признака для сортировки, и требует столько же дополнительной памяти. Таким образом, сложность шага 6 составляет  $O(NRL + P_{\max} + T_{\max})$ , а сложность всего алгоритма нумерации целей в СН ИУС составляет  $O(NRL \cdot SC + NC_{sys} + P_{\max} + T_{\max})$  (так как  $trgs\_cnt \leq NRL$ ).

## 2.2 Дополнительные алгоритмы

Дополнительными алгоритмами, входящими в совокупность алгоритмов решения задачи нумерации ОФЦ, являются:

- 1) алгоритм установки приоритета восстановления номера в СН ИУС;
- 2) алгоритм установки «жесткой связи» между номером в СН ИУС и номером в СН некоторого ИИ;
- 3) алгоритм снятия «жесткой связи» между номером в СН ИУС и номерами в СН ИИ.

Каждый из перечисленных алгоритмов заносит соответствующую его названию информацию в элементы массива  $NumRes$ .

Входными данными для алгоритма установки приоритета восстановления номера в СН ИУС являются значение приоритета и ОФЦ, которому алгоритмом нумерации был присвоен соответствующий номер в СН ИУС. Алгоритм в цикле ( $j = \overline{0, SC-1}$ ) осуществляет просмотр всех номеров в СН ИИ из ОФЦ, и если очередной номер  $n_{src,i}^j$  в СН ИИ не равен



$NN$ , то в поле  $pr$  элемента массива  $NumRes$  с индексом  $NRI(j, n_{src,i}^j)$  заносится переданное значение приоритета. Сложность алгоритма составляет  $O(SC)$ .

Входными данными для алгоритма установки «жесткой связи» между номером в СН ИУС и номером в СН некоторого ИИ являются ОФЦ, которому алгоритмом нумерации был присвоен соответствующий номер в СН ИУС, и идентификатор ИИ  $j$ , «жесткая связь» с номером в СН которого устанавливается. Алгоритм выбирает из ОФЦ номер  $n_{src,i}^j$  в СН  $j$ -ого ИИ и (если  $n_{src,i}^j \neq NN$ ) заносит в поле  $RC$  элемента массива  $NumRes$  с индексом  $NRI(j, n_{src,i}^j)$  единичное значение. Сложность алгоритма составляет  $O(1)$ .

Входным параметром для алгоритма снятия «жесткой связи» между номером в СН ИУС и номерами в СН ИИ является ОФЦ, которому алгоритмом нумерации был присвоен соответствующий номер в СН ИУС. Алгоритм в цикле ( $j = \overline{0, SC - 1}$ ) осуществляет просмотр всех номеров в СН ИИ из ОФЦ, и если очередной номер  $n_{src,i}^j$  в СН ИИ не равен  $NN$ , то в поле  $RC$  элемента массива  $NumRes$  с индексом  $NRI(j, n_{src,i}^j)$  заносится нулевое значение.

### **2.3 Поддержание множества неиспользованных (свободных) номеров в СН ИУС**

Множество неиспользованных (свободных) номеров в СН ИУС может быть смоделировано посредством очереди  $que$  и массива  $free$ , содержащего  $NC_{sys}$  элементов. Каждый  $i$ -ый элемент массива  $free$  соответствует номеру ( $FN_{sys} + i$ ) в СН ИУС и содержит единицу, если этот номер является свободным (принадлежит множеству неиспользованных номеров), и ноль в противном случае. Очередь  $que$  содержит номера в СН ИУС, принадлежащие множеству неиспользованных, в порядке их освобождения.

С использованием этих структур данных процедура освобождения номеров в СН ИУС (используемая на шаге 8 алгоритма нумерации) сводится к следующему. Если элемент массива  $free$ , отвечающий освобождаемому номеру, содержит ноль, то записать в него единицу и добавить номер в очередь  $que$ .

Процедура получения нового свободного номера в СН ИУС (используемая на шаге 7 алгоритма нумерации) сводится к следующему. Если очередь  $que$  не пуста, то извлечь из нее очередной номер в СН ИУС, записать ноль в соответствующий этому номеру элемент массива  $free$  и вернуть значение номера. В противном случае – вернуть значение  $NN$ .

Сложность обоих алгоритмов является константной. Изначально все элементы массива *free* должны содержать единицы, а в очереди *que* должны содержаться все номера в СН ИУС из диапазона  $\{FN_{sys}, \dots, EN_{sys}\}$ .

### **Заключение**

Настоящая работа посвящена решению задачи нумерации ОФЦ, формируемых ЗО, в ИУС различного назначения. Сформулированы требования, которым должна удовлетворять СН ИУС, обеспечивающие достаточно устойчивую систему нумерации ОФЦ, облегчающую оценку целевой обстановки оператором и обеспечивающую непрерывность решения задач ИУС в условиях меняющегося состава ОФЦ. Предложенный алгоритм нумерации, удовлетворяющий сформулированным требованиям, обладает линейной относительно максимального количества ОФЦ сложностью. Дополнительные алгоритмы, входящие в совокупность алгоритмов решения рассмотренной задачи, обладают линейной относительно количества ИИ или константной сложностью. Эти обстоятельства обуславливают возможность применения предложенных подходов для решения задачи нумерации ОФЦ в реальных ИУС.

### **Библиографический список**

1. Савченко Д.И. Оптимальные решающие правила и алгоритм отождествления целевой информации // Электронный журнал «Труды МАИ». – 2012. – № 58, URL: <http://www.mai.ru/science/trudy/published.php?ID=33244> (дата обращения: 28.09.2012).
2. Савченко Д.И. Отождествление целевых данных в информационно-управляющих системах надводных кораблей и судов // Современные проблемы науки и образования. – 2012. – № 4 – URL: [www.science-education.ru/104-6773](http://www.science-education.ru/104-6773) (дата обращения: 27.07.2012).
3. Савченко Д.И. Способ нумерации целей в боевых информационно-управляющих системах // Сборник докладов науч.-техн. конф. "Состояние, проблемы и перспективы создания корабельных информационно-управляющих комплексов". – М.: ОАО «Концерн «Моринформсистема-Агат», 2013.
4. Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы: построение и анализ, 2-е издание: Пер. с англ. – М.: Издательский дом «Вильямс», 2005. – 1296 с.