УДК 621.8:681.5

**М.Э. КУССУЛЬ, А.А. ГАЛИНСКАЯ**

# О СВОЙСТВАХ ЦИКЛОВ В МОДУЛЬНЫХ НЕЙРОННЫХ СЕТЯХ

***Abstract:*** *Modular neural networks are widely used to solve applied problems because of their flexibility and huge potential abilities. As a result, development of modeling aids for modular neural networks become very important. Networks that contain cycles are of particular interest. But for the networks with cycles there is the necessity to have tools for formal analysis, which allow to define the sequence of run of modules in the networks. The article is devoted to theoretical investigations of features of cycles in modular networks that are represented with directed graphs.*
***Key words:*** *modular neural networks, graphs of neural networks, cycles.*

***Анотація:*** *Модульні нейронні мережі широко використовуються при вирішенні прикладних завдань завдяки своїй гнучкості і великим потенційним можливостям. У зв'язку з цим великого значення набувають питання створення засобів моделювання модульних нейронних мереж. Особливий інтерес становлять мережі з циклами. Однак для мереж, які мають цикли, необхідно мати засоби формального аналізу, що дозволили б визначати послідовність роботи модулів у мережі. Ця стаття присвячена вивченню властивостей циклів у модульних мережах, які представлені у вигляді направлених графів.*
***Ключові слова:*** *модульні нейронні мережі, графи нейронних мереж, цикли.*

***Аннотация:*** *Модульные нейронные сети широко применяются при решении прикладных задач благодаря своей гибкости и большим потенциальным возможностям. В связи с этим все большее значение приобретают вопросы разработки средств моделирования модульных нейронных сетей. Особый интерес представляют сети с циклами. Однако для сетей, содержащих циклы, необходимо иметь средства формального анализа, позволяющие определять последовательность работы модулей в сети. Данная статья посвящена изучению свойств циклов в модульных сетях, представленных в виде направленных графов.*
***Ключевые слова:*** *модульные нейронные сети, графы нейронных сетей, циклы.*

## 1. Введение

Модульные нейронные сети представляют большой теоретический и практический интерес, поскольку позволяют решать комплексные задачи распознавания и управления за счет возможности одновременного анализа разнородных данных, а также благодаря возможности использования различных нейросетевых парадигм в единой архитектуре. Преимущества модульных нейронных сетей обусловили как широкое применение, так и разнообразие архитектур и способов их использования [1]. Особый интерес представляют модульные сети с циклами. Такие сети позволяют моделировать сложные системы с запаздыванием, используются для решения задач прогноза, а также в тех случаях, когда требуется композиция и декомпозиция информации, как, например, в ассоциативно-проективных сетях [2].

В статье [3] предложено использовать направленные графы для описания модульных нейронных сетей. Данный подход позволяет классифицировать циклы в модульных нейронных сетях и предоставляет средства для анализа свойств различных типов циклов. Свойства циклов определяют, в частности, возможности автоматического счета и обучения модульной сети без явного указания порядка работы модулей. Эти возможности особенно актуальны для систем автоматического проектирования, таких, как САПР МНС [4].

Примером цикла, для которого невозможно автоматически установить порядок работы модулей, может служить так называемая триггерная схема, представленная на рис. 1. Результат работы такой сети будет различным в зависимости от того, какой из алгоритмов обработки данных $b$ или $c$ был применен первым, либо они применялись параллельно.

В данной статье рассматриваются общие свойства циклов в модульных нейронных сетях и некоторые правила работы с циклами. Второй раздел содержит определения графов модульных нейронных сетей и основных типов циклов. В третьем разделе рассматриваются свойства циклов в модульных нейронных сетях.

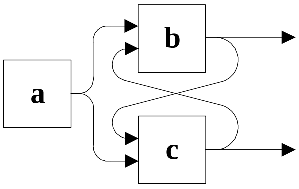

Рис. 1. Триггерная схема соединения модулей

**2. Графы модульных сетей**

Как показано в [3], модульная сеть может быть описана с помощью ориентированного графа $G(V,E)$, вершины $v \in V$ которого соответствуют модулям сети, а ребра $e \in E$ – связям между модулями. Путь и элементарный путь из вершины $v_1$ в вершину $v_2$ будем обозначать $w(v_1,v_2)$, при этом элементарный путь будем называть цепью.

Для представления в виде графов все входы модульной сети считаем модулем входов, а все выходы – модулем выходов сети. Модули входов и выходов виртуальны, поскольку не реализуют алгоритмов обработки данных. Входом или истоком (source) графа $G$ будем называть вершину $I$, соответствующую виртуальному модулю входов сети. Вход сети определяется следующим условием: $s_{in}(I)=0$, где $s_{in}(I)$ – число входящих в вершину ребер. Выходом или стоком (sink) графа будем называть вершину $O$, соответствующую виртуальному модулю выходов. Для него справедливо утверждение $s_{out}(O)=0$, где $s_{out}(O)$ – число выходящих из вершины ребер.

Граф $G$ модульной сети удовлетворяет следующим условиям [3]:

1) граф $G$ – слабосвязный, то есть неориентированный граф, соответствующий $G$ – связный;

2) граф $G$ не имеет кратных ребер;

3) в графе $G$ существует единственная вершина – вход $I \in V$, и любая вершина достижима из входа, то есть $\forall v \in V : v \neq I \Rightarrow \exists\, w(I,v)$. Следовательно, $\forall v \in V : v \neq I \Rightarrow s_{in}(v) \geq 1$;

4) в графе $G$ существует единственная вершина – выход $O \in V$, и выход достижим из любой вершины, то есть $\forall v \in V : v \neq O \Rightarrow \exists\, w(v,O)$. Следовательно, $\forall v \in V : v \neq O \Rightarrow s_{out}(v) \geq 1$.

В отличие от принятого в теории графов определения, путь $w$ из вершины $a$ в вершину $b$ есть множество вершин и соединяющих их ребер, не включая конечную вершину. Аналогично цепь – это элементарный путь (без самопересечений), не включающий конечную вершину.

***Определение 1.*** Проекцией $P(b,a)$ (projection) вершины (модуля) $a \in V$ на вершину (модуль) $b \in V$ будем называть такой ориентированный подграф графа $G(V,E)$, который включает все возможные цепи $w(b,a)$ из $b$ в $a$:

$$P(b,a) = \bigcup w(b,a): \quad \forall w(b,a) = \{v_1, v_2, \ldots, v_n\} : v_1 = b, v_{n+1} = a; \quad v_i \neq v_j; \quad \forall i,j = \overline{1,n},$$

то есть в подграф проекции входят все вершины и соответствующие ребра всех цепей из $b$ в $a$, кроме вершины $a$.

Соответственно, проекция произвольного модуля $a$ на входы модульной сети $I$ записывается как $P(I,a)$.

***Определение 2.*** Если каждой вершине $v \in V$ поставить в соответствие параметр $t(v)$ такой, что $t(v)=1$ в том случае, если выходы соответствующего модуля сети на данном шаге еще не вычислены, и $t(v)=0$, если модуль уже посчитан, то неопределенностью (uncertainty) вершины $a$ называется сумма

$$U(a)=\sum_{v\in P(I,a)} t(v).$$

При этом неопределенность модуля входов $U(I)$ всегда равна нулю. Поскольку модуль входов виртуальный, то считаем, что его выходы всегда определены (вычислены).

***Аксиома 1.*** Из двух модулей сети, при прочих равных условиях и если порядок счета модулей не определен специальными требованиями, первым должен быть посчитан модуль с меньшей неопределенностью.

Необходимость аксиомы 1 очевидна, если рассматривать процесс обучения модульной нейронной сети. Если по пути от входов к некоторой нейронной сети (модулю) $b$ встречается необученный модуль (нейронная сеть) $a$, то обучение модуля $b$ не имеет смысла, пока не обучена нейронная сеть $a$.

Если исходные данные организованы в виде временных последовательностей, будем говорить о «рекуррентных» циклах. Если входящие в цикл модули реализуют ассоциативные поля или другие алгоритмы, для работы которых требуется несколько итераций с одним входным вектором данных, то будем говорить об итерационных циклах.

***Определение 3.*** Циклом с начальной вершиной $a$ будем называть такой ориентированный подграф $C_a(V_a,E_a)$ графа модульной сети $G(V,E)$, который включает все возможные пути $w(a,a)$ из вершины $a$ в саму себя:

$$C_a(V_a,E_a)=\bigcup w(a,a), \quad \forall w(a,a)=\{v_1,v_2,\dots,v_n\}: v_1=a,\ v_{n+1}=a.$$

В отличие от определения проекций, в определении циклов фигурируют пути, а не цепи. Введенное определение отличается от определения, принятого в теории графов, где циклом называют путь, для которого начальная вершина совпадает с конечной [5]. Циклы по определению 3 соответствуют сильносвязанным компонентам в орграфе и обладают очевидным свойством: $C_a \bigcap C_b = \varnothing$.

Необходимость определения цикла как набора всех путей иллюстрируется архитектурой, представленной на рис. 2. Очевидно, что при пересчете модулей, входящих в цикл $C_a: V_a=\{a,b,c,d\}$, прежде чем считать модуль $d$, требуется посчитать оба модуля $b$ и $c$. Заметим, что такой же порядок счета требует и аксиома 1.

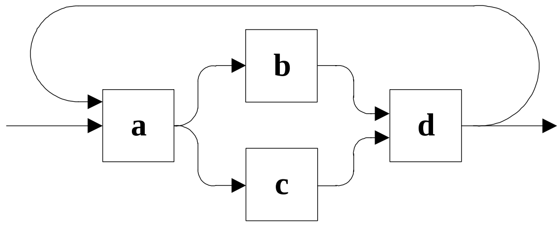


Рис. 2. Схема простого цикла

Из определения цикла непосредственно следует, что виртуальные модули (входы и выходы) никогда не входят ни в один цикл, поскольку модуль входов не имеет входящих ребер, а модуль выходов – выходящих. Кроме того, согласно условиям, накладываемым на граф, представляющий модульную сеть, любой цикл обязательно имеет как входящие, так и выходящие внешние ребра. То есть такие ребра, которые связывают вершины цикла с вершинами, не принадлежащими этому циклу.

Для любого цикла в модульной нейронной сети можно выделить три группы особых вершин. Вершины, которые имеют входящие ребра от вершин, не принадлежащих этому циклу. Вершины, имеющие выходящие ребра, которые входят в вершины за пределами цикла. И вершины, выходящие ребра которых являются входящими ребрами начальной вершины цикла.

Дадим формальное определение вершин, которые будут использоваться в дальнейшем. Для этого воспользуемся понятием длины пути $d(w)$, которое определяется как число ребер в этом пути.

***Определение 4.*** Начальной вершиной $f$ (first) цикла $C(V,E)$ будем называть такую вершину, для которой $d_f = \min\limits_{v \in V} \min\limits_{w = w(I,v)} d(w)$. Конечной вершиной $l$ (last) цикла будем называть такую вершину, для которой $d(w(l,f)) = 1$. Граничной вершиной $t$ (terminal) будем называть такую вершину, для которой выполняется условие

$$t : \left(\exists\ e(t,p),\ p \notin V\right) \& \left(\forall\ v \in V : P(f,t) \subset P(f,v) \Rightarrow \bar{\exists}\ e(v,g),\ g \notin V\right),$$

то есть $t$ – самая дальняя от начальной вершина цикла, имеющая связь (связи) с не принадлежащими циклу вершинами.

Данное определение не гарантирует единственность каждой из вершин, однако позволяет провести дальнейшую классификацию циклов. Введем определения для двух типов циклов, принципиально важных с точки зрения счета модульных сетей, и рассмотрим некоторые из свойств таких циклов.

***Определение 5.*** Обычным или обыкновенным циклом (ordinary cycle) с начальной вершиной $a$ будем называть такой цикл $Co_a(V_a,E_a)$, для которого не существует ни одной пары вершин, таких, что они принадлежат проекциям друг друга на вход, то есть $\forall\ v_1,v_2 \in V_a : v_1 \in P(I,v_2) \quad \Rightarrow \quad v_2 \notin P(I,v_1)$. Перекрестным циклом (crossed cycle) с начальной вершиной $b$ будем называть такой цикл $Cc_b(V_b,E_b)$, который содержит хотя бы одну пару вершин, которые входят в проекции друг друга на вход, то есть $\exists\ v_1,v_2 \in V_b : v_1 \in P(I,v_2) \quad \& \quad v_2 \in P(I,v_1)$.

Введение понятия обыкновенных циклов необходимо для установления порядка пересчета модулей в сети и, в первую очередь, чтобы отделить все возможные варианты триггерных схем (рис. 1). Нетрудно проверить, что цикл, изображенный на рис. 1, является перекрестным и содержит две начальные вершины, удовлетворяющие определению 4.

Если обозначить множество всех возможных циклов с начальными вершинами, удовлетворяющими определению 4, через $\Omega\{C_f\}$, то для множеств обыкновенных и перекрестных циклов очевидны соотношения

$$\Omega\{Cc\} \cap \Omega\{Co\} = \varnothing; \qquad \Omega\{Cc\} \cup \Omega\{Co\} = \Omega\{C_f\}.$$

Обыкновенный цикл может быть представлен только единственным образом, в отличие от циклов общего вида, которые тождественны с точностью до выбора начальной вершины. Перекрестные циклы этим свойством не обладают.

### 3. Свойства циклов

Разработка алгоритмов анализа архитектур модульных сетей автоматической системой существенно упрощается, если эти алгоритмы строятся на основе свойств, присущих графам модульных сетей. Данный раздел посвящен исследованию графов модульных сетей, содержащих циклы. В разделе приведены и доказаны утверждения, на основе которых строятся алгоритмы анализа модульных сетей. Кроме того, здесь же рассмотрены важные частные случаи циклов, имеющие самостоятельный интерес с точки зрения работы модульных сетей.

***Утверждение 1.*** Для любой вершины $v$, принадлежащей циклу $C_f$, обязательно найдется хотя бы один путь $w(v,v)$, содержащий начальную вершину $f$ этого цикла.

Это утверждение тривиально, но не всегда очевидно, поскольку, по определению, цикл может включать произвольное число путей.

**Доказательство.** Пусть $v$ принадлежит $C_f$. Тогда

$$\exists\, w(f,f) = w(f,v) \cup w(v,f) \quad \Rightarrow \quad \exists\, w(v,v) = w(v,f) \cup w(f,v).$$

***Теорема 1.*** Для любых двух вершин $a$ и $b$ цикла $C_f(V_f, E_f)$ всегда существует проекция $P(a,b)$, и эта проекция целиком принадлежит циклу

$$\forall a,b \in V_f : \exists\, P(a,b) \subseteq C_f.$$

**Доказательство.** Покажем, что поскольку вершины $a$ и $b$ принадлежат циклу, то существует хотя бы один путь вида $\tilde{w}(a,b)$. Из утверждения 1 следует, что

$$a \in C_f \quad \Rightarrow \quad \exists\, w(a,a) = w(a,f) \cup w(f,a) \quad \Rightarrow \exists\, w(a,f).$$

Тогда, если $b \in w(a,f) \Rightarrow w(a,f) = \tilde{w}(a,b) \cup \tilde{w}(b,f)$, то путь из $a$ в $b$ существует. Если вершина $b$ не принадлежит пути из $a$ в $f$, то поскольку $b \in C_f \quad \Rightarrow \quad \exists\, w(b,b) = w(b,f) \cup w(f,b)$, следовательно, существует путь $\tilde{w}(a,b) = w(a,f) \cup w(f,b)$.

Пусть путь $\tilde{w}(a,b)$ не элементарный, то есть существует вершина $\tilde{v}$, которая входит в этот путь хотя бы два раза: $\tilde{w}(a,b)=\{a,v_1...v_i,\tilde{v},...,\tilde{v},v_{i+k},...,v_n\},v_{n+1}=b$. Тогда можно построить путь из $a$ в $b$: $w(a,b)=\{a,v_1...v_i,\tilde{v},v_{i+k},...,v_n\},v_{n+1}=b$, который будет уже простым. То есть существует хотя бы одна цепь $w(a,b)$ и $P(a,b)\neq\varnothing$.

Покажем, что все цепи из $a$ в $b$ принадлежат циклу. Так как обе вершины принадлежат циклу, то $a\in C_f \Rightarrow \exists\, w_1(f,a)$ и $b\in C_f \Rightarrow \exists\, w_2(b,f)$. Тогда существует путь $\exists\, w(f,f)=w_1(f,a)\cup w(a,b)\cup w_2(b,f)$. Таким образом, любая цепь $w(a,b)$ принадлежит циклу, так как он включает в себя все возможные пути из начальной вершины в нее же по определению.■

***Теорема 2*** (Признак существования цикла). Для того чтобы граф модульной сети содержал цикл (циклы), достаточно, чтобы граф содержал хотя бы одну такую вершину, что для нее существует хотя бы одно входящее ребро, не принадлежащее проекции этой вершины на входы.

**Доказательство.** Пусть в графе существует вершина $a$, для которой есть входящее ребро $e(b,a)$, не принадлежащее проекции этой вершины на входы. В соответствии с определением проекции это означает, что для вершины $b$, из которой выходит это ребро, нет пути от входов сети, не проходящего через вершину $a$. Так как по условиям, наложенным на граф, любая вершина достижима из входов, то для вершины $b$ обязательно существует хотя бы один путь $w(I,b)$. Следовательно, если существует $e(b,a)$, то любому пути $w(I,b)$ обязательно принадлежит вершина $a$ : $\exists\, e(b,a): e\notin P(I,a) \quad \Leftrightarrow \quad a\in w(I,b) \quad \forall\, w(I,b)$.

Тогда существует хотя бы один путь из вершины $a$ в саму себя $w(a,a)=w(a,b)\cup e(b,a)$ и, следовательно, существует цикл.■

Следует отметить, что признак существования цикла, предложенный в теореме 2, дает только достаточное условие существования цикла. Он позволяет находить все обыкновенные циклы в модульной сети, но только некоторые перекрестные циклы.

Рассмотрим несколько важных свойств обыкновенных циклов.

***Теорема 3*** (Теорема единственности обыкновенного цикла). Если обыкновенный цикл существует, то начальная вершина этого цикла может быть выбрана только единственным образом.

**Доказательство.** Проведем доказательство от противного. Пусть $C_{f1}(V_{f1},E_{f1})$ – обыкновенный цикл. Пусть в нем существует еще одна начальная вершина, то есть $\exists\, f2\in V_{f1},\, f2\neq f1:\quad d_{f1}=d_{f2}=\min\limits_{w(I,v)}(d_v)\quad \forall v\in V_{f1}$ (из определения начальной вершины). Так как минимальные расстояния от входа сети к начальным вершинам равны, то для вершины $f1$ существует путь от входа сети, не содержащий вершину $f2$. То есть $\exists\, w_1(I,f1): f2\notin w_1$. Так же существует путь $\exists\, w_2(I,f2): f1\notin w_2$, так как $f2$ – тоже начальная вершина. С другой стороны, поскольку вершины $f1$ и $f2$ обе принадлежат циклу, то, согласно утверждению 1, существуют пути $w_3(f1, f2)$ и $w_4(f2, f1)$. При этом, согласно определению пути, $f2 \notin w_3$ и $f1 \notin w_4$. Тогда существуют следующие пути:

$$(f1 \notin w_2 \ \& \ f1 \notin w_4) \Rightarrow \exists\, w(I, f1) = w_2(I, f2) \cup w_4(f2, f1);$$

$$(f2 \notin w_1 \ \& \ f2 \notin w_3) \Rightarrow \exists\, w(I, f2) = w_1(I, f1) \cup w_3(f1, f2).$$

Из определения проекции вершины на входы следует, что $f1 \in P(I, f2) \ \ \& \ \ f2 \in P(I, f1)$. То есть, по определению, цикл $C_{f1}(V_{f1}, E_{f1})$ является перекрестным, и, следовательно, в обыкновенном цикле существует единственная начальная вершина. ■

Непосредственно из проведенного доказательства вытекает:

***Следствие 3.1.*** Если обыкновенный цикл $Co_f$ существует $(V_f \neq 0)$, то на данном наборе вершин он определяется единственным образом.

Действительно, поскольку по определению цикла в него входят все возможные пути из начальной вершины в саму себя, то любой цикл определяется с точностью до начальной вершины. Поскольку в обыкновенном цикле начальная вершина только одна, то и цикл определяется однозначно.

***Теорема 4.*** В обыкновенном цикле существует только одна вершина, связанная входящими ребрами с вершинами вне цикла. Это вершина начальная:

$$v \in Co_f : \ \ \exists\, e(p,v),\, p \notin Co_f \Rightarrow v = f.$$

**Доказательство.** Проведем доказательство от противного. Пусть $C_f(V_f, E_f)$ – обыкновенный цикл и, кроме начальной, в нем существует еще одна вершина $v$, имеющая внешнее входящее ребро $e(p,v)$. То есть $\exists\, v \in V_f, v \neq f : \ \ \exists\, e(p,v),\, p \notin V_f$.

Так как любая вершина достижима из входов, то существует путь $w(I, p)$. Покажем, что если вершина $p$ циклу не принадлежит, то для любого пути $\forall\, w(I, p),\, p \notin V_f \Rightarrow f \notin w$.

Пусть существует путь, содержащий начальную вершину цикла $\exists\, w(I, p) : f \in w \ \Rightarrow \ w(I, p) = w(I, f) \cup w(f, p)$. Следовательно, существует путь $w(f, p)$. Согласно исходному предположению, $v \in V_f \Rightarrow \exists\, w(v, f)$ и существует ребро $e(p,v)$. А значит, существует путь из начальной вершины в саму себя через вершину $p$: $w(f, f) = w(f, p) \cup e(p,v) \cup w(v, f)$, следовательно, $p \in V_f$, что противоречит условию.

Поскольку начальная вершина цикла не принадлежит пути от входа к вершине $p$, то для вершины $v$ существует путь от входа сети, не содержащий начальную вершину $f$: $w_2(I,v) = w(I, p) \cup e(p,v)$. Так же существует путь $\exists\, w_1(I, f) : v \notin w_1$, так как $f$ – начальная вершина цикла. С другой стороны, обе вершины $f$ и $v$ принадлежат циклу и, согласно утверждению 1, существуют пути $w_3(f,v)$ и $w_4(v, f)$. При этом, согласно определению пути, $v \notin w_3$ и $f \notin w_4$. Тогда существуют следующие пути:

$$( f \notin w_2 \ \& \ f \notin w_4 ) \Rightarrow \exists\, w( I, f ) = w_2( I,v ) \cup w_4( v, f );$$

$$( v \notin w_1 \ \& \ v \notin w_3 ) \Rightarrow \exists\, w( I,v ) = w_1( I, f ) \cup w_3( f ,v ).$$

Из существования таких путей и определения проекции вершины на входы следует, что $f \in P( I,v ) \ \ \& \ \ v \in P( I, f )$. То есть, по определению, цикл $C_f (V_f ,E_f )$ является перекрестным, что противоречит условию, и, следовательно, в обыкновенном цикле существует только одна вершина, связанная входящими ребрами с вершинами вне цикла, и эта вершина начальная.■

***Следствие 4.1.*** В обыкновенном цикле проекция начальной вершины на входы не содержит ни одной вершины цикла.

**Доказательство.** Пусть в обыкновенном цикле $C_f (V_f ,E_f )$ найдется вершина, попадающая в проекцию начальной вершины этого цикла на входы $\exists v \in V_f : v \in P( I, f )$. Из определения проекции вершины на входы получим, что существует путь $w( I, f ) : v \in w \ \& \ f \notin w$. Следовательно, этот путь может быть разложен на два $w(I,f)=w(I,v) \cup w(v,f) \ \Leftrightarrow \ \exists w(I,v): f \notin w$. Так же существует кратчайший путь $\exists\, w( I, f ) : v \notin w$, так как $f$ – начальная вершина. По аналогии с доказательством теоремы 4 получим, что цикл является перекрестным.■

***Следствие 4.2.*** Для любой вершины $v \neq f$ обыкновенного цикла $C_f$ все цепи от входов содержат начальную вершину, то есть

$$P( I,v ) = P( I, f ) \cup P( f ,v ) \quad \forall v \in C_f .$$

**Доказательство.** Поскольку любая вершина достижима из входов, а, согласно теореме 4, ни одна вершина цикла (кроме начальной) не имеет внешних связей, то все цепи (пути) от входов к вершинам обыкновенного цикла содержат начальную вершину.■

В статье [3] приведено несколько утверждений, касающихся специальных видов циклов. Для доказательства утверждения 2 из [3] используются результаты теоремы 4. Поэтому доказательство утверждения приведено в данной статье.

Напомним, что простым циклом мы называем такой обыкновенный цикл, для всех вершин которого любой замкнутый путь проходит через начальную вершину этого цикла (рис. 2). То есть цикл $Co_f (V_f ,E_f )$ простой, если $\forall\, w( v,v ) : v \in V_f \quad \Rightarrow \quad f \in w( v,v )$ (определение 6 в [3]).

Сцепленными циклами называем такие простые циклы $Co_1$ и $Co_2$, которые имеют одну или более общих вершин. Причем, если для двух общих вершин $a$ и $b$ вершина $a$ принадлежит проекции вершины $b$ на начальную вершину любого из этих циклов, то проекция $P( a,b )$ целиком принадлежит каждому из циклов $Co_1$ и $Co_2$, то есть

$$\forall a,b : a,b \in Co_1 ,Co_2 \quad a \in P( f ,b ) \quad \Rightarrow \quad P( a,b ) \subset Co_1 \ \& \ P( a,b ) \subset Co_2 ,$$

где $f$ – начальная вершина любого из циклов $Co_1$ или $Co_2$ (определение 8 в [3]).

***Утверждение.*** Если два простых цикла сцеплены и ни одна из начальных вершин не принадлежит пересечению этих циклов, то такие сцепленные циклы образуют перекрестный цикл.

**Доказательство.** Пусть простые циклы $Co_1(V_{f1}, E_{f1})$ и $Co_2(V_{f2}, E_{f2})$ сцеплены и $f2 \notin V_{f1}$, $f1 \notin V_{f2}$. Это означает, что $\exists\, w_1(I, f1): f2 \notin w_1$ и $\exists\, w_2(I, f2): f1 \notin w_2$. Согласно теореме 4, сцепленный цикл $C = Co_1 \bigcup Co_2$ будет перекрестным, так как в нем две вершины имеют внешние входящие ребра.■

На основе рассмотренных свойств циклов можно предложить формальные определения разрешенных и запрещенных архитектур, а также способы разрешения противоречий в случае триггерных циклов (рис. 1).

## 4. Выводы

Общие свойства циклов, рассматриваемые в данной статье, позволяют формально анализировать модульные нейронные сети. Использование свойств циклов позволяет построить алгоритмы анализа архитектур сетей автоматической системой. Более того, автоматический поиск всех модулей, входящих в цикл, позволяет формально приписать каждому из циклов сети некоторые свойства. Это оказывается очень полезным в системах автоматического проектирования, где пользователь может, во-первых, уточнить порядок счета, если цикл не является простым, и, во-вторых, определить каждый цикл как рекуррентный или итерационный.

Обработка данных с помощью модульной нейронной сети подразумевает, что последовательность счета модулей известна. Приведенные выше определения и формулировки предназначены, в первую очередь, для вывода правил автоматического построения последовательностей счета модулей, а также для контроля архитектур, созданных пользователем. Свойства циклов позволяют определить, какие архитектуры модульных сетей являются разрешенными без введения дополнительных условий.

Формальные определения разрешенных и запрещенных архитектур модульных сетей, а также пути разрешения противоречий в «запрещенных» архитектурах будут даны в последующих публикациях. Также в последующих публикациях будут приведены алгоритмы анализа модульных сетей, основанные на свойствах предложенных графов.

## СПИСОК ЛИТЕРАТУРЫ

1. Галинская А.А. Модульные нейронные сети: обзор современного состояния разработок // Математические машины и системы. – 2003. – № 3, 4. – С. 87 – 102.
2. Куссуль Э.М. Ассоциативные нейроподобные структуры. – Киев: Наукова думка, 1992. – 143 с.
3. Куссуль М.Э. Графы модульных нейронных сетей // Математические машины и системы. – 2005. – № 1. –С. 26 – 38.
4. Резник А.М., Куссуль М.Э., Сычев А.С., Садовая Е.Г., Калина Е.А. Система проектирования модульных нейронных сетей САПР МНС // Математические машины и системы. – 2002. – № 3. – С. 28 – 37.
5. Дистель Р. Теория графов. – Новосибирск: Изд-во Ин-та математики, 2002. – 336 с.