УДК 004.03

Противоречия в свойствах двух основных типов сетевых моделей и пути их разрешения

Малинина Н.Л.

Аннотация:

Предлагаемая статья освещает два диаметрально противоположных подхода к построению сетевых моделей. В качестве математических абстракций предлагается графическое представление сетевых моделей в виде ориентированных графов. Рассматриваются преимущества и недостатки сетевых моделей, основанных на вершинных и реберных графах. Предлагается ставить задачу синтеза сетевых моделей как задачу преобразования вершинных графов в реберные графы.

Ключевые слова: сетевая модель; сложный процесс; система; ориентированный граф; вершинный граф; реберный граф.

Много лет в качестве структурных моделей сложных процессов и систем используется такой класс конфигураций, который имеет графическое представление в виде ориентированных графов. Часто такие конфигурации называют сетевыми моделями.

Любой сложный процесс включает в себя два подмножества элементов: элементарные операции (первичное множество) и моменты их начала и окончания (вторичное множество) [1]. Ориентированный граф также имеет два подмножества элементов: дуги и вершины. Когда реальный процесс представляется с помощью сетевой модели, то элементарным операциям (основному подмножеству) ставятся в соответствие либо дуги, либо вершины графа. Назовем сетевую модель, в которой элементарным операциям ставятся в соответствие дуги графа, обыкновенной сетевой моделью (ОСМ). Если же основному подмножеству ставятся в соответствие вершины графа, то такую сетевую модель назовем сопряженной сетевой моделью (ССМ).

Обыкновенные сетевые модели (реберные графы) применяются в таких методах планирования и управления, как метод ПЕРТ, метод критического пути, методы СПУ и т.д. Они получили широкое распространение благодаря большой наглядности, простоте анализа и возможности применения универсальных алгоритмов для расчета их характеристик.

Однако при этом обыкновенные сетевые модели отличаются чрезвычайной сложностью и трудоемкостью построения.

Сопряженные сетевые модели нашли свое применение во французском методе потенциалов, при построении практически любых моделей сложных процессов. Они (вершинные графы, блок-схемы) обладают простотой построения даже при большой размерности, однако при этом отличаются высокой сложностью и трудоемкостью их анализа. При разработке алгоритмов расчета их характеристик от исследователя требуются чудеса изобретательности. При этом обычно не удается разработать универсальные алгоритмы анализа [2, 3, 4, 5, 6].

Читателю может показаться странным, что большинство ссылок отправляет нас в 60-70 г.г. Дело в том, что в те времена ОСМ были надеждой на светлое будущее. Думалось, что сложности построения удастся преодолеть, если и не сегодня, то в самом ближайшем будущем. К сожалению это оказалось не так. Быстрорастущая сложность объектов моделирования вводила в искушение строить модели быстрее уже в виде сопряженных сетевых моделей. Создавалось подсознательное ощущение, что уж если удалось построить сетевую модель, то разобраться с анализом всегда получится. Однако уже 80-е г.г. характеризуются огромным ростом коллективов разработчиков, но, тем не менее, все большим количеством, мягко говоря, не совсем качественного программного продукта. Часто желаемое выдавалось за действительное [7]. С каждым днем мир все жестче и жестче ощущал на себе давление задач, связанных с полным перебором вариантов, однако эту проблему пока еще удавалось обходить. Компьютерные технологии в части быстродействия и увеличения объемов памяти развивались очень быстро. Однако уже к рубежу веков эти возможности оказались на пределе.

Итак, простота построения сопряженных сетевых моделей объясняется отсутствием в их составе фиктивных работ. Однако именно это обстоятельство является причиной огромной сложности их анализа. В то же время, необходимость введения фиктивных работ делает построение обыкновенных сетевых моделей весьма сложным делом, однако существенно упрощает их анализ.

Попробуем разобраться, почему это происходит. Перечень или список элементарных операций или действительных работ, которые составляют исследуемый или управляемый процесс и которые задаются в исходной информации, будем называть реальным исходным множеством $Q_0 = \{q_{0i}\}$. Реальному исходному множеству Q_0 в сетевой модели соответствует исходное множество $Q = \{q_i\}$ действительных дуг-работ (в обыкновенной сетевой модели) или действительных вершин-работ (в сопряженной сетевой модели).

Работы (элементарные операции) реального процесса связаны между собой попарно (не обязательно все) системой бинарных отношений, всегда имеющих характер причинно-следственных отношений. Система бинарных отношений, заданная на реальном исходном множестве Q_0, записывается как декартово произведение множества Q_0 на себя: $L_0 = Q_0 \times Q_0$.

Условия структурного подобия модели и процесса требуют, чтобы на исходном множестве дуг или вершин сетевой модели была задана такая же система бинарных отношений: $L_0 = L = Q \times Q$. Система бинарных отношений может быть представлена матрицей бинарных отношений или непосредственных путей [8]. Часто ее называют также матрицей смежности.

Если задано множество: $Q = \{q_i\}, где\, i = 1, 2, 3, \dots n_Q; (n_Q = n)$ и декартово произведение: $L = Q \times Q$, которое представляет собой множество упорядоченных пар (q_i, q_j), где $(q_i, q_j) \in Q$, тогда множество L может быть представлено матрицей $L = \left\| l_{ij} \right\|_1^n$, элементы которой:

$$l_{ij} = \begin{cases} 1, \text{ если } (q_i, q_j) \in L \subset Q \times Q \\ 0 - \text{ остальное} \end{cases}$$

Поскольку элементы q_i $(i = 1,2, \dots n)$ могут быть или вершинами или дугами сетевой модели, то матрица $L = \left\| l_{ij} \right\|_1^n$ может быть либо матрицей смежности вершин одной сетевой модели, либо матрицей смежности дуг другой сетевой модели, либо одновременно той и другой матрицей. В двух последних случаях матрица L должна удовлетворять ряду определенных требований [9].

Итак, сопряженной сетевой модели соответствует такая матрица L, в которой строки и столбцы соответствуют вершинам-работам, а элементы $l_{ij} = 1$ – дугам-связям. Такая матрица L называется матрицей смежности вершин некоторого графа $G(Q, \Gamma)$ и обозначается $E = \left\| e_{ij} \right\|_1^n$.

Определение: пусть матрица $L = E$ суть матрица смежности вершин ориентированного графа $G(Q, \Gamma)$, тогда:

$$e_{ij} = \begin{cases} 1, \text{если существует дуга } (q_i, q_j) \\ 0 - \text{остальное} \end{cases}$$

Графическое отображение сопряженной сетевой модели является вершинным графом.

Определение: если дано множество вершин $Q = \{q_i\}$ и отображение $(\Gamma: Q \to Q)$, определяемой матрицей $E = L$, то пара $G(Q, \Gamma)$ - ориентированный вершинный граф.

Исходная информация, которая необходима для построения сетевой модели содержит список работ, и для каждой работы, входящей в список, - подсписки непосредственно ей предшествующих работ (таблица 1). Для построения сопряженной сетевой модели подобная исходная информация содержит все необходимые сведения о ее вершинах (работах) и дугах (связях), то есть о первичном и вторичном множествах. Действительно, составим матрицу с числом рядов, равным числу работ, входящих в список и поставим единицу на месте e_{ij}, если работа q_i предшествует работе q_j, то получим матрицу E.

На практике принято описывать сопряженные сетевые модели с помощью таких ориентированных графов, в которых из вершины q_i в вершину q_j идет не более чем одна дуга. В этом случае все элементы e_{ij} матрицы E равны 0 или 1.

Таблица 1

№п/п	Код выполняемой работы	Содержание выполняемой работы	Коды работ, непосредственно предшествующих выполняемой работе
1	*B*	-“-	*A, C*
2	*F*	-“-	*B, D, H*
3	*A*	-“-	

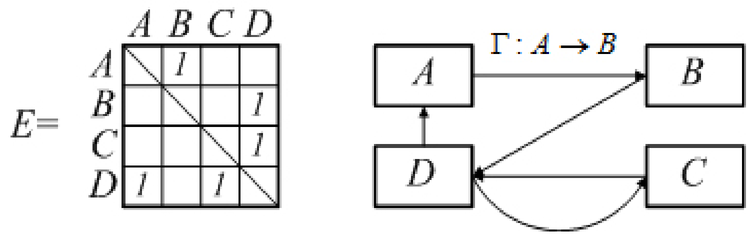


Рис.1

На рис.1 графу $G(Q,\Gamma)$ соответствует матрица E. Очевидно, что если $e_{ii} = 1$, то граф имеет петлю при вершине q_i. Транспонированная матрица E^T представляет собой матрицу смежности вершин такого ориентированного графа, который получается из исходного графа переменой ориентации всех дуг. Матрица E является симметричной в двух случаях: когда граф G является симметрическим и когда он – неориентированный. Для графического изображения сетевых моделей применяются только ориентированные графы, поскольку неориентированное ребро не может служить изображением причинно-следственной связи. В случае, когда $e_{ij} = e_{ji} = 1$, будем полагать, что такой паре всегда соответствуют две противоположно ориентированные дуги. Иногда говорят, что симметрический граф может быть получен из неориентированного графа путем операции удвоения [10].

Любая, произвольно заданная квадратная матрица, состоящая из нулей и единиц, позволяет непосредственно по ней построить вершинный граф.

В обыкновенной сетевой модели основным, или первичным, множеством является множество дуг-работ $Q = \{q_i\}$, а вторичным множеством – множество вершин-событий $V = \{v_h\}$. В этом случае исходной информации для построения обыкновенной сетевой модели оказывается недостаточно. Если принять множество Q за множество дуг, то оказывается, что подобная исходная информация не содержит сведений о вершинах V обыкновенной сетевой модели. Поэтому обыкновенную сетевую модель нельзя представить с помощью матрицы смежности вершин до ее построения. Обыкновенную сетевую модель можно представить с помощью матрицы смежности дуг R.

Определение: пусть R – матрица смежности дуг ориентированного графа $H(V, Q)$, тогда:

$$r_{ij} = \begin{cases} 1, \text{если дуги } q_i \text{ и } q_j \text{ имеют общую вершину } v_h \text{ и, кроме того, } \; q_i < q_j \\ 0 - \text{все остальные случаи} \end{cases}$$

Знак «<» означает предшествование дуги q_i дуге q_j, то есть: дуга q_i заходит в вершину v_h, а дуга q_j исходит из v_h вершины. Ориентированный граф $H(V, Q)$, которому соответствует матрица смежности дуг R, будем называть ориентированным реберным графом. В дальнейшем, поскольку мы будем рассматривать только ориентированные графы, то под термином реберный граф будем понимать ориентированный реберный граф, а под термином вершинный граф – ориентированный вершинный граф.

Определение: если дано $Q = \{q_i\}$, – множество дуг, и существует система бинарных отношений $L = Q \times Q$, заданная матрицей $L = R$, которая определяет такое множество вершин $V = \{v_h\}$, что $\Gamma_V: V \to V = Q$, то пара $H(V, Q)$ – реберный граф (рис. 2).

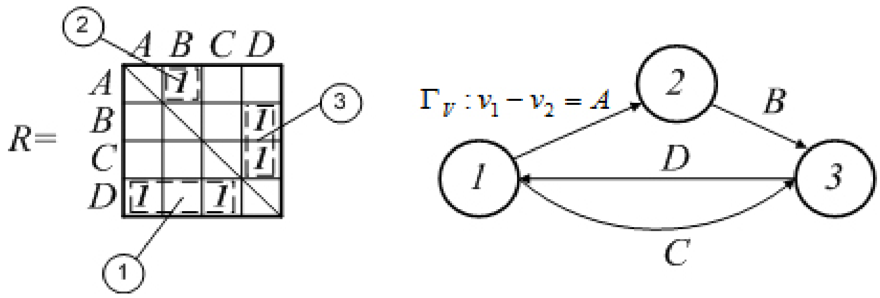


Рис. 2

Из определения реберного графа следует, что матрица смежности не может быть произвольной. Кроме того, что матрица смежности должна отражать систему бинарных отношений $L \subset Q \times Q$, она одновременно должна также определять множество вершин $V = \{v_h\}$. В этом и состоит дополнительное требование, которое предъявляется к матрице R.

Матрица R (рис. 2) отвечает такому требованию. Ее ненулевые элементы могут быть сгруппированы в ряд непересекающихся прямоугольных подматриц, каждая из которых определяет какую-либо вершину реберного графа.

Произвольная заданная квадратная матрица, состоящая из нулей и единиц, не всегда обладает таким свойством, и, таким образом, не всегда может быть принята в качестве матрицы смежности дуг ориентированного графа. Следовательно, в противоположность вершинному графу, реберный граф не всегда можно, а, как правило, нельзя построить по произвольно заданной матрице L. Далее будет рассмотрен вопрос о том, когда произвольная матрица L может быть принята в качестве матрицы смежности дуг ориентированного графа. Пока заметим лишь, что, если $R = \|r_{ij}\|_1^n$ – матрица смежности ориентированного графа $H(V,Q)$, то он может быть однозначно построен по заданной матрице R только в том случае, если он имеет не более одной минимальной и не более одной максимальной дуги. В силу этого обстоятельства вопросы синтеза сетевых моделей пока будут рассматриваться только применительно к таким основным множествам $Q = \{q_i\}$ и таким, вершинным и реберным графам, которые им соответствуют, и у которых каждая связная компонента имеет не более одного минимального и не более одного максимального элемента. Очевидно, что если множество Q не отвечает этому требованию, то оно всегда может быть дополнено единственным минимальным, либо единственным максимальным элементом, либо тем и другим одновременно. А поэтому вопросы синтеза могут быть распространены на произвольные ориентированные множества Q.

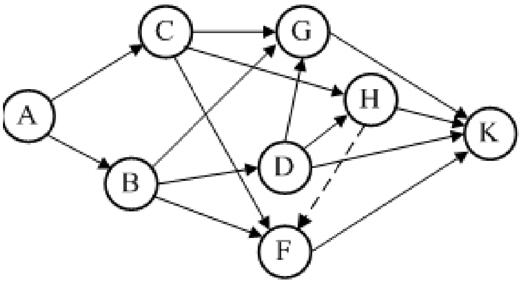
Введем еще одну матрицу – операторную матрицу смежности вершин реберного графа $H(V,Q)$. Пусть $F = \|f_{ng}\|_1^m$ – операторная матрица смежности вершин ориентированного реберного графа $H(V,Q)$. Тогда:

$$f_{ng} = \begin{cases} i, \text{если вершина } v_h \text{ соединена дугой } q_i \text{ с вершиной } v_g \\ 0 - \text{во всех остальных случаях} \end{cases}$$

Условимся в дальнейшем, если это не будет вызывать путаницы, пользоваться сокращениями без индексации, то есть: H, L, G, R, E, F и т.д.

Перейдем к причине преимуществ обыкновенных сетевых моделей. Матрица смежности дуг обладает одним фундаментальным свойством, существенно отличающим ее от произвольной матрицы смежности вершин. Рассмотрим это свойство на примерах. Пусть сетевая модель задана матрицей L. Если в матрице L в столбце q_j (строке q_i) имеется единственный ненулевой элемент $l_{ij} = 1$, то это означает, что от работы q_i к работе q_j возможен безусловный переход. Если же в столбце q_j кроме элемента $l_{ij} = 1$ имеется еще, по крайней мере, один элемент , то это означает, что работа по окончании работы может быть начата только при условии, что предварительно будет также выполнена работа . То есть: переход к работе после выполнения работы возможен только условный.

Пример 1. Пусть сетевая модель представлена вершинным графом и матрицей (рис. 3). В таблице 2 выписаны безусловные и условные переходы к последующим работам и условия перехода.



$$E=\begin{array}{c|cccccccc} & A & B & C & D & F & G & H & K \\ \hline A & & 1 & 1 & & & & & \\ B & & & & 1 & 1 & 1 & & \\ C & & & & & 1 & 1 & 1 & \\ D & & & & & & 1 & 1 & 1 \\ F & & & & & & & & 1 \\ G & & & & & & & & 1 \\ H & & & & & 1 & & & 1 \\ K & & & & & & & & \end{array}$$

Рис. 3

Таблица 2

Выполняемые работы	Безусловные переходы к работам	Условные переходы к работам
A	*B, C*	-
B	*D*	*F*, если выполнены *C* и *H*; *G*, если *C* и *D*
C		*F*, если выполнены *B* и *H*; *G*, если *B* и *D*; *H*, если *D*
D		*G*, если выполнены *B* и *C*; *K*, если *F*, *G* и *H*; *H*, если *C*
F		*K*, если выполнены *D*, *G* и *H*
G		*K*, если выполнены *D*, *F* и *H*
H		*F*, если выполнены *B* и *C*; *K*, если *F*, *D* и *G*
K		-

Из таблицы 2 следует, что в сопряженной сетевой модели:

1. Безусловные и условные переходы к другим работам могут иметь место одновременно.
2. Условные переходы по окончании каких-либо работ к другим работам могут быть различны.

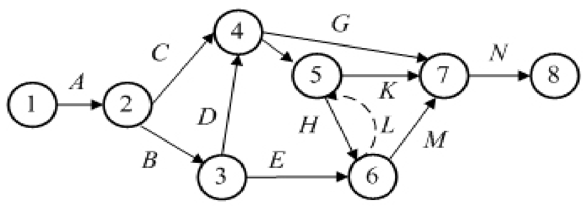
Условимся называть сопряженные сетевые модели, которые обладают перечисленными свойствами, произвольными сопряженными моделями. Очевидно, что в силу таких особенностей, анализ произвольных сопряженных моделей может быть очень сложным. Одновременное совмещение различных видов переходов и различных условий переходов приводит к тому, что анализ таких сетевых моделей требует, чтобы алгоритмы анализа содержали блоки перебора, поиска и сопоставления условий перехода, либо анализ может быть осуществлен только с помощью алгоритмов, разработанных применительно к конкретным моделям.

Пример 2. Пусть обыкновенная сетевая модель задана реберным графом , которому соответствует матрица смежности дуг (рис. 4). В таблице 3 выписаны виды переходов и условия переходов к другим работам. В обыкновенной сетевой модели, так же как и в сопряженной сетевой модели, возможны оба вида переходов. Однако, в отличие от произвольных сопряженных сетевых моделей, в обыкновенных сетевых моделях:

1. Безусловные и условные переходы к другим работам никогда не имеют место одновременно.

2. При условных переходах к другим работам условия перехода ко всем последующим работам всегда одинаковы.

Действительно, после выполнения работы возможен условный переход к работам и , но при одинаковом для обеих работ условии – условии предварительного выполнения работы .



$$R=\begin{array}{c|cccccccccccc} & A & B & C & D & E & F & G & H & K & L & M & N \\ \hline A & & 1 & 1 & & & & & & & & & \\ B & & & & 1 & 1 & & & & & & & \\ C & & & & & & 1 & 1 & & & & & \\ D & & & & & & 1 & 1 & & & & & \\ E & & & & & & & & & & 1 & 1 & \\ F & & & & & & & & 1 & 1 & & & \\ G & & & & & & & & & & & & 1 \\ H & & & & & & & & & & 1 & 1 & \\ K & & & & & & & & & & & & 1 \\ L & & & & & & & & 1 & 1 & & & \\ M & & & & & & & & & & & & 1 \\ N & & & & & & & & & & & & \end{array}$$

Рис. 4

Таблица 3

Выполняемые работы	Безусловные переходы к работам	Условные переходы к работам
A	*B, C*	–
B	*D, E*	–
C		*F* и *G*, если выполнена *D*
D		*F* и *G*, если выполнена *C*
E		*L* и *M*, если выполнена *H*
F		*H* и *K*, если выполнена *L*
G		*N*, если выполнены *M* и *K*
H		*L* и *M*, если выполнена *E*
K		*N*, если выполнены *M* и *G*
L		*N* и *K*, если выполнена *F*
M		*N*, если выполнены *K* и *G*
N		-

Условимся называть сетевые модели, обладающие перечисленными выше свойствами, квазиканоническими сетевыми моделями. Квазиканонической может быть не только обыкновенная сетевая модель, но и сопряженная модель тогда, когда соответствующий граф $G(Q,\Gamma)$ является графом смежности дуг графа $H(V,Q)$. Некоторые сетевые модели обладают кроме этих свойств еще одним, а именно: условный переход осуществляется всегда только к одной работе. Такие сетевые модели, и обыкновенные, и сопряженные, будем называть вполне каноническими или просто каноническими. Квазиканонические и канонические модели составляют класс канонических моделей.

Указанные свойства канонических моделей позволяют разрабатывать для их анализа универсальные алгоритмы, не содержащие блоков перебора и сопоставления условий перехода. Универсальные алгоритмы анализа и расчета характеристик канонических сетевых моделей, опираясь на эти фундаментальные свойства канонических сетевых моделей, работают по типу рекуррентных последовательностей. Можно сказать, что канонические модели обладают свойством рекуррентности.

Итак, мы рассмотрели причины, которые обуславливают большую наглядность и простоту анализа канонических сетевых моделей. Достоинства и недостатки обоих видов сетевых моделей, произвольных и канонических, представлены в таблице 4.

Простота анализа и универсальность обыкновенных сетевых моделей подтверждается и фактами из практики. Обыкновенные сетевые модели обладают еще одним важным преимуществом перед сопряженными сетевыми моделями. Это преимущество состоит в следующем. На множестве элементарных операций сложного процесса задается не только система бинарных отношений, но и система логических условий, которые определяют возможность реализации тех или иных элементарных операций. Эта система логических условий может быть выражена с помощью каких-либо логических функций, которые могут быть как детерминированными, так и вероятностными. Поскольку сложный процесс может иметь как детерминированную, так и вероятностную структуру, такую же структуру должна иметь и сетевая модель процесса. Представляется весьма удобным иметь возможность анализировать и строить логическую структуру процесса независимо от физического содержания его отдельных элементов.

В сопряженных сетевых моделях элементарным операциям соответствуют вершины графа, а связям между ними – дуги. В этом случае дуги графа соответствуют только системе бинарных отношений между элементами процесса. Система логических условий совмещается с множеством вершин графа. Это является причиной того, что анализ логической структуры процесса невозможно отделить от анализа его физического содержания.

Таблица 4

Основные этапы применения сетевых моделей / Виды сетевых моделей	Построение (синтез) сетевых моделей	Анализ сетевых моделей и расчет их характеристик
Произвольная сопряженная сетевая модель	Отличается простотой непосредственного построения сетевых моделей по исходной информации	Отличается сложностью и трудоемкостью
Каноническая (обыкновенная или сопряженная) сетевая модель	Отличается сложностью и трудоемкостью непосредственного построения сетевых моделей по исходной информации	Отличается простотой, меньшей трудоемкостью и возможностью применения универсальных алгоритмов

В обыкновенных сетевых моделях элементарным операциям соответствуют дуги графа, а связям и системе логических условий – вершины графа. Это обстоятельство позволяет отделить анализ логической структуры процесса от анализа его физического содержания. Такая особенность обыкновенных сетевых моделей существенно облегчает анализ моделей сложных процессов, особенно со стохастической структурой. Сопряженные сетевые модели такого преимущества не имеют. Описанные выше свойства обыкновенных и сопряженных сетевых моделей представлены в таблице 5.

Таблица 5

<table>
<tr><td colspan="2">Сетевые модели</td><td colspan="4">Содержание элементов сетевой модели</td></tr>
<tr><td rowspan="3">Класс</td><td rowspan="3">Вид</td><td colspan="2">Элементарные операции (работы)</td><td colspan="2">Структура</td></tr>
<tr><td>Действительные работы</td><td>Фиктивные работы</td><td>Логические операции</td><td>Связи (отношения)</td></tr>
<tr><td colspan="3">Вершины</td><td>Дуги</td></tr>
<tr><td rowspan="3">Сопряженные</td><td>Произвольные</td><td>$q_i \in Q$</td><td>-</td><td>$q_i \in Q^*$
$Q_1 \subset Q$</td><td>$(q_i; q_j) \in \Gamma$</td></tr>
<tr><td rowspan="4">Канонические</td><td colspan="3">Вершины</td><td>Дуги</td></tr>
<tr><td>$q_i \in Q_g$</td><td>$q_i \in \overline{(Q_g \cap Q_k)}$</td><td>$q_i \in Q^*$</td><td>$(q_i; q_j) \in \Gamma_k$</td></tr>
<tr><td rowspan="2">Обыкновенные</td><td colspan="2">Дуги</td><td colspan="2">Вершины</td></tr>
<tr><td>$q_i = (v_h; v_g)_k$
из $q_i \in Q$</td><td>$q_i = (v_h; v_g)_k$
из $q_i \in$
$\overline{(Q_g \cap Q_k)}$</td><td>$v_h \in V_{k1}$</td><td>$v_h \in V_k$</td></tr>
</table>

Выводы

Исследование свойств двойственных между собой ориентированных графов: реберного и сопряженного ему вершинного и свойств соответствующих им двух видов сетевых моделей: обыкновенных и сопряженных позволил выявить их следующие основные достоинства и недостатки:

1. Сопряженные сетевые модели обладают возможностью и простотой непосредственного построения модели с детерминированной структурой по произвольной матрице бинарных отношений.

2. В сопряженных сетевых моделях отсутствует свойство рекуррентности.

3. В сопряженных сетевых моделях происходит совмещение физического содержания элементарных операций реального сложного процесса и логических условий их реализаций в одних и тех же элементах сетевой модели – в вершинах.

4. Обыкновенные сетевые модели обладают сложностью (или невозможностью) построения модели по произвольно заданной матрице бинарных отношений.

5. Обыкновенные сетевые модели обладают свойством рекуррентности.

6. В обыкновенных сетевых моделях физическое содержание элементарных операций реального процесса и логические условия реализации разделены друг от друга в различных элементах сетевой модели.

7. Свойство рекуррентности обыкновенных сетевых моделей обеспечивает возможность построения стандартизированных универсальных алгоритмов анализа и расчета их характеристик с детерминированной структурой.

8. Анализ вопросов двойственности ориентированных вершинных и реберных графов [9] позволяет сформулировать задачу синтеза обыкновенных сетевых моделей как задачу построения ориентированного реберного графа по заданному ориентированному вершинному графу.

Литература

1. Малинин Л.И., Малинина Н.Л. Топологические свойства процесса проектирования. Эл. журнал Труды МАИ, № 30, 2008, 17 с.
2. Eisner Howard. Essentials of Project and Systems Engineering Management, John Wiley & Sons, New York, 2002, 425 p.
3. Элмаграби Салах Е. Алгебра для анализа обобщенных планов. Приложение к книге Miller R. Shedule, Cost and Profit Control with PERT, McGraw-hill, New York, 277 p.
4. Miller R. Shedule, Cost and Profit Control with PERT, McGraw-hill, New York, 277 p.
5. Авондо-Бодино Дж. Применение в экономике теории графов. М.: Прогресс, 1966, 160 с.
6. Форд Л.Р., Фалкерсон Д.Р. Потоки в сетях. М.: Мир, 1966, 276 с.

7. Брукс Ф. Как проектируются и создаются программные комплексы (мифический человеко-месяц), М.: Символ-Плюс, 2006, 304 с.

8. Оре О. Теория графов. М.: Наука,1968, 352 с.

9. Малинин Л.И., Малинина Н.Л., Изоморфизм графов в теоремах и алгоритмах, М.: URSS, 2009, 249 с.

10. Берж К. Теория графов и ее применения, М.: ИЛ, 1962, 320 с.

11. Малинина Н.Л. Техногенные катастрофы через призму математики, М.: Авиапанорама, 2009, № 4, стр. 34-35.

12. Малинина Н.Л. Техногенные катастрофы через призму математики, М.: Авиапанорама, 2009, № 5, стр. 97-99.

13. Фор Р., Кофман А., Дени-Паллен М. Современная математика, М.: Мир, 1966, 271 с.

Данные об авторе:

Малинина Наталия Леонидовна, Московский Авиационный институт (Технический университет), кафедра Вычислительной математики и программирования, доцент, кандидат физико-математических наук.

Почтовый адрес: Москва, Грайвороновская ул., 17-354

Электронный адрес: malinina806@gmail.com

/Малинина Наталия Леонидовна/

«____»________________2010г.