



Решение данного дифференциального уравнения при начальных условиях

$$\int_0^t e(t)dt = - \int_1^{R_h(t)} \frac{1}{R_h(t)} dR_h(t) \quad (14)$$

имеет вид

$$R_h(t) = e^{\int_0^t e(t)dt}. \quad (15)$$

Полученное выражение (15) для определения вероятности безошибочной работы человека может быть использовано в качестве основы для формирования механизма предупреждения возникновения ошибок работы человека в непрерывной временной области.

Таким образом, предлагаемый подход для оценки и повышения надежности информационных сред промышленного предприятия, в основе которого лежат методы теории надежности технических средств и устройств, позволяет повысить надежность и эффективность проектирования сложных информационных систем, учитывая наличие человеческого фактора как источника основных отказов.

Литература

1. Черкесов Г.Н. Надежность аппаратно-программных комплексов: Учеб.пособ. – СПб.: Питер, 2005. – 479 с.
2. ГОСТ 27.205. Надежность в технике. Проектная оценка надежности сложных систем с учетом технического и программного обеспечения и оперативного персонала.

А.В. Докучаев

ВВЕДЕНИЕ ФИКТИВНЫХ ДУГ В ПРОЦЕССЕ ПОСТРОЕНИЯ СЕТЕВЫХ МОДЕЛЕЙ СЛОЖНЫХ ТЕХНИЧЕСКИХ СИСТЕМ

(Самарский государственный технический университет)

Задача построения сетевых моделей в виде ориентированных графов – пример одного из наиболее распространённых и успешных приложений дискретной математики в разнообразных областях: моделировании сложных технических систем [1-3, 6], управлении разработками программного обеспечения, планировании эксперимента и др.

Широкий спектр приложений диктует необходимость самых различных математических постановок задачи, значительно усложняющих решение. В частности, в настоящее время отсутствует экономный алгоритм решения практически важной задачи оптимального вложения дополнительного ограниченного ресурса, ускоряющего выполнение отдельных работ сложного проекта.



Существующие приёмы решения являются переборными и поэтому предъявляют завышенные требования к вычислительным и временным параметрам известных пакетов программ.

В процессе разработки программного обеспечения ставится задача разработчики программного обеспечения за минимальное время с привлечением минимального ресурса (например: меньшего числа разработчиков, денежных средств). Для оптимизации сроков выполнения разработки применяются сетевые модели «работа-дуга» [1-3].

Рассмотрим задачу сетевого планирования и управления (СПУ) связным проектом $P = \{a(i)\}_{i=1}^{k \geq 1}$ из проектных работ $a(i): i \neq j \Leftrightarrow a(i) \neq a(j)$ с предшественниками $s(a(i)) \subset P$, здесь $s(a(i))$ – собственное (возможно, пустое) подмножество работ проекта P , требующих завершения до начала выполнения работы $a(i)$. Схема построения сетевой модели проекта изображена на рис. 1.

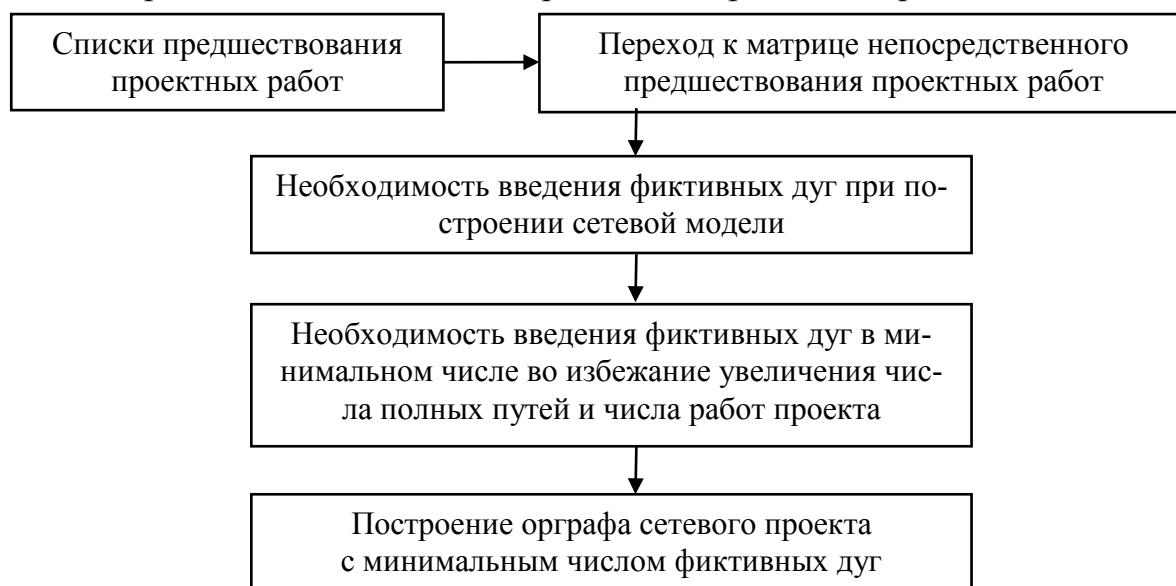


Рис. 1. Схема построения сетевой модели

После проведения процедур сокращения списка предшественников и упорядочивания работ [1] строится матрица непосредственного предшествования работ проекта.

В ходе построения орграфа проекта модели в ряде случаев возникает необходимость добавления фиктивных дуг. Фиктивная дуга соответствует работе с нулевым временем выполнения. Фиктивные дуги необходимо добавлять в минимальном числе, чтобы не увеличить общее число дуг проекта и как следствие число полных путей орграфа проекта. Задача оптимального построения орграфов так же актуальна при построении управляющих программ реального времени [5], изображении функциональных структур информационных систем [4].

Предложим универсальный алгоритм добавления необходимого числа фиктивных работ (соответствующих фиктивным дугам) $a(i) \notin P, i \geq k+1$.



Пусть D – неединичный блок матрицы непосредственного предшествования работ проекта и пусть $s \times r$ -подматрица D содержит единичную, то есть заполненную единицами, первую строку; остальные строки, а также все её столбцы ненулевые. Единичная строка может оказаться не единственной; обозначим их число через d . Очевидно, $1 \leq d \leq s-1$. Также в подматрице D может быть несколько единичных столбцов; обозначим их число через e . Очевидно, $0 \leq e \leq r-1$.

Добавим фиктивные работы по следующему алгоритму:

1) Выделим в $s \times r$ -подматрице D собственную 2×2 -блочную $s_1 \times r_1$ -подматрицу $D_1 < D$, $1 \leq s_1 \leq s-1$, $1 \leq r_1 \leq r-1$, состоящую из трёх единичных (заполненных единицами и обозначенных через

$$I[d_1 \times e_1], I[d_1 \times (r_1 - e_1)] \text{ и } I[(s_1 - d_1) \times e_1]$$

и одного нулевого, заполненного нулями и обозначенного

$$O[(s_1 - d_1) \times (r_1 - e_1)],$$

нижнего блока. Соответствующие размерности блоков удовлетворяют неравенствам

$$1 \leq d_1 \leq s_1 - 1 \text{ и } 1 \leq e_1 \leq r_1 - 1.$$

2) Для уменьшения числа добавляемых фиктивных работ необходимо так подобрать подматрицу $D_1 < D$, чтобы она не была собственной подматрицей другой аналогичной 2×2 -блочной подматрицы $D_2 < D$.

Таким образом,

$$D_1 = \begin{pmatrix} I[d_1 \times e_1] & I[d_1 \times (r_1 - e_1)] \\ I[(s_1 - d_1) \times e_1] & O[(s_1 - d_1) \times (r_1 - e_1)] \end{pmatrix}$$

или, с очевидными переобозначениями размерностей блоков,

$$D_1 = \begin{pmatrix} I[d_1 \times e_1] & I[d_1 \times (r_1 - e_1)] \\ O[(s_1 - d_1) \times e_1] & I[(s_1 - d_1) \times (r_1 - e_1)] \end{pmatrix}.$$

3) Дальнейшие построения проводятся в обоих случаях аналогично, поэтому рассмотрим лишь первый вариант блочного представления D_1 . Добавим фиктивную работу $a(k+1)$ с условиями непосредственного предшествования $a(i_t) \in s(a(k+1))$ и $a(k+1) \in s(a(j_t))$, где правильные номера i_t непосредственно предшествующих фиктивной работе $a(k+1) \notin P$ проектных работ $a(i_t) \in P$ соответствуют попавшим в блок $I[d_1 \times e_1]$ подматрицы D_1 строкам, а правильные номера j_t непосредственно последующих за фиктивной работой $a(k+1) \notin P$ проектных работ $a(j_t) \in P$ соответствуют попавшим в блок $O[(s_1 - d_1) \times (r_1 - e_1)]$ подматрицы D_1 столбцам.

Применение описанного выше алгоритма позволяет автоматизировать построение сетевых моделей с добавлением минимального числа фиктивных дуг.

Литература

1. Докучаев А. В., Котенко А. П. «Свойства графов задач сетевого планирования и управления». *Вестн. Сам. гос. техн. ун-та. Сер. Физ.-мат. науки*, 5(21), 2010. – с. 204–211.



2. Докучаев А.В., Котенко А.П. Построение графа задачи оптимизации сетевого планирования и управления // Информационные, измерительные и управляющие системы (ИИУС-2010): Материалы международной научно-технической конференции. 17-21 мая 2010 г. Самар. гос. техн. ун-т. – Самара, 2010. – с. 291-294.

3. Докучаев А.В., Котенко А.П. Построение графа задачи оптимизации сетевого планирования // Сб. Математическое моделирование и краевые задачи: Тр. седьмой Всероссийской научной конференции с международным участием. Ч.2: Моделирование и оптимизация динамических систем и систем с распределёнными параметрами. – Самара: Самар. гос. тех. ун-т, 2010. – с. 86-90.

4. Дидрих И.В. О надёжности программно-технических комплексов // Перспективные информационные технологии (ПИТ 2014): труды Международной научно-технической конференции / под ред. С.А. Прохорова. – Самара: Издательство Самарского научного центра РАН, 2014. – с. 442-444.

5. Мясникова Е.А., Тюгашев А.А. Параметрический генератор управляющих программ реального времени // Перспективные информационные технологии (ПИТ 2013): труды Международной научно-технической конференции / под ред. С.А. Прохорова. – Самара: Издательство Самарского научного центра РАН, 2013. – с. 354-357.

6. Постовалова И. П. Эффективный синтез сетевой модели «работы-дуги» с минимальным числом фиктивных работ. Управление большими системами, 2014, выпуск 52, – с. 118–132.

Н.В. Ефимушкина, Д.А. Панюшкин

ИМИТАЦИОННЫЕ МОДЕЛИ ДЛЯ ИССЛЕДОВАНИЯ ПАМЯТИ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

(Самарский государственный технический университет)

Введение

Общеизвестно, что современные вычислительные системы характеризуются сложными структурами и режимами функционирования. Для изучения особенностей их работы используются методы теории вычислительных систем (ВС) [1, 3]. Наиболее достоверные результаты позволяют получить эксперименты непосредственно над объектом в реальных или специально созданных условиях. Высокая сложность вычислительных систем ограничивает применение этих методов для обучения студентов.

Формулировка проблемы

Наиболее перспективными представляются методы имитационного моделирования, которые предполагают описание системы в виде алгоритма, называемого имитационной моделью. Соответствующая программа содержит процедуры, воспроизводящие структуру системы и протекающие в ней процессы. Важнейшее свойство имитационного моделирования — универсальность. Ме-