УДК: 681.586

МОДИФИКАЦИЯ АЛГОРИТМА ПОИСКА МНОЖЕСТВА ВСЕХ ПРОСТЫХ ЦИКЛОВ В АВТОМАТИЗИРОВАННОЙ СИСТЕМЕ ПОЛУЧЕНИЯ АНАЛИТИЧЕСКОГО ВЫРАЖЕНИЯ ПЕРЕДАТОЧНОЙ ФУНКЦИИ

К. А. Шумак, Ю. А. Лежнина, Т. П. Кравченкова

Астраханский государственный архитектурно-строительный университет (Россия)

Для решения задачи получения аналитического выражения для передаточной функции схемы в заданной точке разработан приводимый ниже алгоритм, представляющий собой модификацию известных алгоритмов нахождения максимальных независимых множеств и алгоритмов поиска в графе в глубину с применением метода ветвей и границ.

Ключевые слова: алгоритм поиска на графе, метод ветвей и границ, передаточная функция, независимые множества.

To solve the problem of obtaining an analytical expression for the transfer function of the scheme at a given point, the following algorithm is developed, which is a modification of the known algorithms for finding the maximum independent sets and search algorithms in the graph in depth using the method of branches and boundaries.

Keywords: search algorithm on the graph, the method of branches and borders, transfer function, independent set.

Основной задачей разрабатываемой системы является получение аналитического выражения для передаточной функции схемы в заданной точке [1]. Для этого был выбран метод, основанный на получении отображения по Лапласу искомой передаточной функции схемы и перевода ее в область оригинала.

Каждый блок описывается линейным дифференциальным уравнением вида

$$a_n Y^{(n)}(t) + a_{n-1} Y^{(n-1)}(t) + \ldots + a_0 = b_m X^{(m)}(t) + b_{m-1} X^{(m-1)}(t) + \ldots + b_0$$

где

X(t) – закон изменения входного сигнала;

Y(t) – закон изменения выходного сигнала;

an, b_n – числовые коэффициенты.

Схема представляет собой совокупность блоков, соединенных определенным образом (параллельно, последовательно и с обратной связью). Ее можно описать системой дифференциальных уравнений, состоящей из уравнений блоков и уравнений потока сигналов в линиях передачи сигналов.

Найти при заданной функции YG(t), вырабатываемой генератором G(t), аналитического выражения одной, нескольких или всех функций X1(t),...,Xn(t),Y1(t),...,Ym(t) - значит, решить основную задачу системы.

Воспользоваться для этой цели обычными методами решения систем линейных дифференциальных уравнений не представляется возможным, т.к. в силу возможного наличия в схемах блоков и подсхем обратной связи уравнения, описывающие потоки сигналов в линиях передачи, оказываются рекурсивно связанными и, следовательно, не могут получить разрешения. Например, в приведенной выше схеме блок 4 реализует обратную связь, охватывающую подсхему, образованную блоками 1, 2 и 3. При попытке выразить из системы уравнений, составленной по схеме, одной из неизвестных величин возникает проблема, состоящая в том, что, в конечном итоге, любая неизвестная величина зависит от самой себя (т.е. определена рекурсивно). Отказаться от наличия в схеме обратных связей и свести таким образом решение задачи к детерминированному решению системы линейных дифференциальных уравнений не представляется целесообразным, так как обратная связь - это основной способ управления процессами в схемах подобного типа и отказ от обратных связей резко ограничил бы возможности системы по исследованию линейных непрерывных процессов в схемах различной природы.

Метод эквивалентных преобразований был отклонен, как более пригодный для ручного исследования схем, нежели для машинного, поскольку для его использования необходим человеческий опыт, которому, как известно, машина обучается с большим трудом. Для этого был предложен следующий метод получения передаточной функции системы без эквивалентных преобразований, основанный на применении формулы Мейсона [2]. Основная идея метода заключается в то, что система представляется в виде ориентированного графа, в котором дуги эквивалентны элементарным звеньям (блокам), а вершины - линиям передачи сигналов. Направление дуги соответствует направлению движения сигнала. Контрольной точкой, т.е. точкой, в которой будет вычисляться искомая функция, может быть только вершина. Так как каждая вершина может быть инцидентна, как входящим, так и исходящим дугам, сигнал в вершине представляет собой суперпозицию выходных сигналов блоков, соответствующих входящим в вершину дугам. Генератор представляется дугой, начальная вершина которой может быть только исходящей.

Для вычисления переходной характеристики системы по формуле Мейсона необходимо разработать алгоритм поиска множества всех “непересекающихся” контуров (простых циклов)

Множество простых циклов есть множество “непересекающихся” контуров. Представим каждый контур как вершину некоторого графа GM. Любые две вершины этого графа будем соединять ребром, если у контуров есть хотя бы одна общая вершина (т.е. контуры соприкасаются). Тогда задача сводится к поиску всех множеств внутренней устойчивости графа (независимых множеств).

Для решения этой задачи разработан приводимый ниже алгоритм, представляющий собой модификацию известных алгоритмов нахождения максимальных независимых множеств и алгоритмов поиска в графе в глубину с применением метода ветвей и границ.

Входные данные (Input data).

Список N простых контуров (List of Cycles) - S.

Выходные данные (Output data).

Список множеств несоприкасающихся контуров (List of “untouchment” Cycles) - L.

```
Algorithm Search_untouchment_cycles
L = ∅ ;
d = 1;  — количество контуров в текущем множестве
M = {1};      — текущее множество контуров
i = M[d]+1;

while (M[1] ≠{N}) do
        if (i = N+1)
        then
                ———————— STEP RETURN ————————;
                if (d = 1)
                then
                        M[d] = M[d]+1;
                        i = M[d]+1;
                else
                        i = M[d]+1;
                        REMOVE( M[d] );
                        d — — ;
                end_if
                ———————— STEP RETURN ————————;
        end_if
                while (not_untouchment(M, S[i])) and (i < N+1) do
                        i++
                end_while;
                        — нашли несоприкасающийся с текущим множеством контур
                if (i<N+1)
                then
                        M = M + {i};
                        d + + ;
                        L = L + {M}
                end_if
end_while
```

Формальное описание алгоритма

Входными данными алгоритма является список S всех простых контуров графа. Количество контуров N = || S ||. S[i] - i-тый контур, который представляет из себя список входящих в контур дуг и ребер.

Выходные данные - список L - содержит множества номеров “несоприкасающихся” контуров. Например, если контуры с номерами i и j не “соприкасаются”, то L содержит элемент {i, j}.

Идея алгоритма состоит в расширении и модификации текущего множества “несоприкасающихся” контуров. Перед началом работы основного цикла алгоритма **while** (M[1] ≠{N}) **do** текущее множество M содержит только номер первого контура: M = {1}, i - номер следующего контура.

Основной цикл алгоритма содержит 3 основных блока:

1) **if** (i = N+1)

Блок осуществляет шаг возврата. Выполняется, если в результате поиска в предыдущем проходе цикла не был найден “несоприкасающийся” с текущим множеством контур.

Возможны 2 ситуации:

1. в текущем множестве 1 номер: d = 1; В этом случае в текущее множество заносится следующая вершина

2. в текущем множестве больше 1 номера: d > 1; В этом случае последняя вершина из текущего множества удаляется (REMOVE(M[d]);), а просмотр контуров будет вестись со следующей после нее вершины (i = M[d]+1;), т.е. осуществляется поиск в графе GM(см. выше) в глубину.

2) **while** (not_untouchment(M, S[i])) **and** (i < N+1) **do**

Выполняет последовательный поиск в списке контуров S контура, который бы не “соприкасался” с текущим множеством M (т.е. не имел бы вершин и дуг, которые входили бы в контуры, включенные в M). Возможны 2 исхода завершения цикла:

1. найден такой контур:

(not_untouchment(M, S[i])) = false

(i < N+1) = true

Если контур найден то выполняется блок 3.

2. не найден такой контур:

(not_untouchment(M, S[i])) = true

(i < N+1) = false

При выполнении следующего прохода основного цикла выполняется блок - шаг возврата.

3) **if** (i<N+1)

Блок выполняется, если найден контур “не соприкасающийся” с текущим множеством M. Номер контура заносится в текущее множество: M = M + {i}; текущее множество заносится в список “несоприкасающихся” контуров: L = L + {M}.

Выход из основного цикла и окончание алгоритма произойдет по достижении последней вершины, рассматриваемой как основание текущего множества (M[1] = {N}).

Список литературы

1. Зарипова В. М., Петрова И. Ю., Петрова И. Ю., Шумак К. А., Лежнина Ю. А. Исследование динамических характеристик элементов автоматики умного дома по параметрическим структурным схемам // Вестник МГСУ. 2017. № 12. С. 1424–1434.

2. Михалевич С. С., Байдали С. А., Чурсин Ю. А. Расчет сложной передаточной функции обобщенного объекта. Промышленные АСУ и контроллеры. 2013. № 9. С. 46–51.