

УДК 004.3

А. С. Кобайло, кандидат технических наук, доцент (БГТУ)

МЕТОДИКА ФОРМИРОВАНИЯ МНОЖЕСТВА АПРИОРНЫХ ПРОЕКТНЫХ РЕШЕНИЙ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ РЕАЛЬНОГО ВРЕМЕНИ

В работе рассмотрена методика формирования множества альтернативных вариантов проектируемых вычислительных систем на одном из начальных этапов их синтеза, что позволит решать задачи анализа и оптимизации систем на ранней стадии их проектирования, тем самым существенно сократив трудоемкость алгоритма синтеза и потребность в памяти для хранения априорных вариантов проектируемой системы. Методика предусматривает учет всех возможных вариантов свертки вершин графа синтезируемой системы при формировании множества априорных проектных решений.

The present paper overviews the procedure for generating a multiplicity of alternative versions of computing systems being developed at one of the initial stages of their synthesis. The procedure enables the user to solve the tasks of system analysis and optimization at the early design stages thus attaining a significant reduction of algorithm synthesis routine complexity and memory volumes for the storage of a priori versions of a system being developed. The procedure takes a consideration of all possible convolutions of graph vertices of a system being synthesized when generating a multiplicity of a priori design solutions.

Введение. Большинство известных подходов к проектированию сложных технических объектов, в том числе и средств вычислительной техники, предполагает создание множества альтернативных вариантов системы с последующим их анализом с целью выбора оптимального (оптимальных) по заданным критериям качества. В рамках теории синтеза вычислительных систем реального времени (ТСВСРВ) множество альтернативных вариантов определяется множеством векторов назначения для графа вычислительного алгоритма (ГВА) реализации заданной математической модели и, соответственно, модели вычислительного процесса, подлежащего реализации проектируемой вычислительной системы (ВС) [1].

В ходе дальнейшего синтеза для каждого из векторов назначения формируются множества свертываемых вершин. Реализация операции свертки (простого гомоморфизма) для вершин этих множеств позволяет снизить объем оборудования создаваемой ВС по сравнению с базовым вариантом на базе ГВА, что, в свою очередь, приводит к понижению стоимости, потребляемой мощности, габаритов и т. п., повышению надежности.

Однако, наряду с очевидным положительным эффектом, выполнение операции свертки может повлечь и негативные последствия, связанные с потребностью в служебных функциональных устройствах (ФУ), в частности, мультиплексорах. Введение в операционную часть системы дополнительного служебного оборудования при усложнении управляющей части системы, связанном с необходимостью формирования кодов для управления мультиплексорами, может привести к существенному снижению эффекта от свертки

вершин, вплоть до полного его нивелирования. Указанные обстоятельства обуславливают целесообразность учета и анализа всех возможных вариантов системы, в том числе и экстремальных – с отсутствием сверток и с комбинацией всех возможных сверток из числа выявленных.

Основная часть. Рассмотрим основные положения теории синтеза вычислительных систем реального времени, на которых базируется методика решения выделенной проблемы.

Отметим, что требования реализации вычислительного процесса в реальном времени являются ужесточающими по отношению к методологии синтеза и синтезируемым системам, поэтому ТСВСРВ является универсальной теорией для синтеза специализированных ВС, при этом отсутствие требований реального времени приводит к упрощению ТСВСРВ, т. е. к ее сокращенному варианту.

Таким образом, рассматриваемые в данной работе положения являются отражением некоторых из этапов проектирования в рамках единого подхода к их синтезу.

Поиск множества альтернативных вариантов проводится на основе графа алгоритма (ГА) реализации математической модели вычислительного процесса $G = (V, E)$, множество V вершин которого соответствует множеству операций (фрагментов алгоритма), множество E ребер (для ориентированного графа – множество дуг) – информационным связям.

На основании ГВА, временных характеристик функциональных устройств $\{\tau_i^j\}$ из заданного набора и временных ограничений на формирование вычислительного процесса проектируемой системой, заданных в виде $\{t(\varphi_i) \leq \Delta t^{\text{зад}}\}$, можно сформировать множество $\{G^{bc}\}$. Здесь

τ_i^j – время реализации j -м функциональным устройством i -й операции алгоритма, $t(\varphi_i)$ – время выполнения i -й операции проектируемой ВС, $\Delta t^{\text{зад}}$ – заданный интервал времени, определяющий формирования вычислительного процесса в реальном времени, G^{bc} – граф базовой структуры.

Определение 1. Графом базовой структуры (ГБС) называется граф, описывающий процесс обработки данных вычислительной системой в соответствии с графом вычислительного алгоритма и изоморфный этому графу.

Для формирования ГБС необходимо вершинам ГВА поставить в соответствие (назначить) ФУ, реализующие функции (операции), отождествленные с данными вершинами. Назначение ФУ вершинам ГВА производится в два этапа:

- построение матрицы соответствия;
- формирование множества векторов назначения.

Определение 2. Прямоугольная матрица D размером $I \times J$, где I – множество вершин графа базовой структуры, J – множество ФУ из заданного набора, для которой:

$$d_{ij} = \begin{cases} 1, & \text{если } j\text{-е ФУ реализует операцию,} \\ & \text{отождествленную с } i\text{-й вершиной графа;} \\ 0 & \text{– во всех остальных случаях,} \end{cases}$$

называется матрицей соответствия.

Множество альтернативных вариантов структуры проектируемого объекта может быть неявно задано в виде множества векторов назначения $\{R\}$.

Определение 3. Вектором назначения называется вектор $R = (r_1, r_2, \dots, r_I)$, i -й элемент которого равен номеру ФУ, назначенного i -й вершине графа.

Здесь $i = 1, \dots, I$, где I – количество вершин графа вычислительного алгоритма (ГВА) реализации математической модели проектируемого объекта, $r_i = j$ – номер функционального устройства из множества $Z = \{z_1, z_2, \dots, z_i, \dots, z_I\}$ экземпляров классов $Z_n | n = 1, \dots, N$ классов типовых элементов, реализующих в совокупности все операции некоторого класса задач, т. е. элемента $z_k(n)$ из подкласса $Z_n \leftrightarrow \varphi_n \leftrightarrow v_i$ – класса элементов, реализующих функцию φ_n , отождествленную с вершиной v_i ГВА, N – мощность множества Z_n .

Определение 4. Множество $R = \{R(1), R(2), R(3), \dots, R(W)\}$ векторов размером I , элементы которых $r(w) = j: d_{ij} = 1$, называется множеством векторов назначения.

Мощность этого множества определяется как NP -полный перебор всех комбинаций функциональных устройств, удовлетворяющих требованию реализации процесса в реальном времени

$$M = [R] = \prod_{i=1}^I \sum_{j=1}^J d_{ij},$$

где d_{ij} – элемент матрицы соответствия, $[R]$ – мощность множества векторов назначения.

Каждому вектору назначения соответствует свой вектор реализации, элементы которого $\{\tau_j\}$ формируются как параметры τ_i^j соответствующих ФУ.

Определение 5. Вектором реализации называется вектор-строка $\tau = (\tau_1, \tau_2, \dots, \tau_J)$, где $\tau_j, j = 1, \dots, J$ – время выполнения операции, отождествленной с вершиной, находящейся в начале j -й дуги.

Из определения 3 следует, что каждому из найденных векторов назначения соответствует свой уникальный вектор реализации. Тогда каждый из априорных альтернативных вариантов может быть описан кортежем вида

$$AP1_w = \{G_w^{BC}, \varphi_w^{BC}, R_w^{BC}, \tau_w^{BC}\},$$

где $AP1_w$ – априорное решение первого уровня для w -го вектора назначения; G_w^{BC} – граф базовой структуры (ГБС); φ_w^{BC} – спецификация вершин ГБС; R_w^{BC} и τ_w^{BC} – w -й вектор назначения и вектор реализации соответственно.

Множества AP 0-го уровня располагаются на 0-м ярусе графа поиска решений, представленного на рисунке.

Для каждого вектора $R_w, w=1, \dots, W$ может быть определено множество S^w подмножеств свертываемых вершин S^w , при этом формируется множество априорных решений 2-го уровня (1-й ярус ГПР). AP 2-го уровня описываются кортежем

$$AP2_w = \{G_w^{BC}, \varphi_w^{BC}, R_w^{BC}, \tau_w^{BC}, S_w\},$$

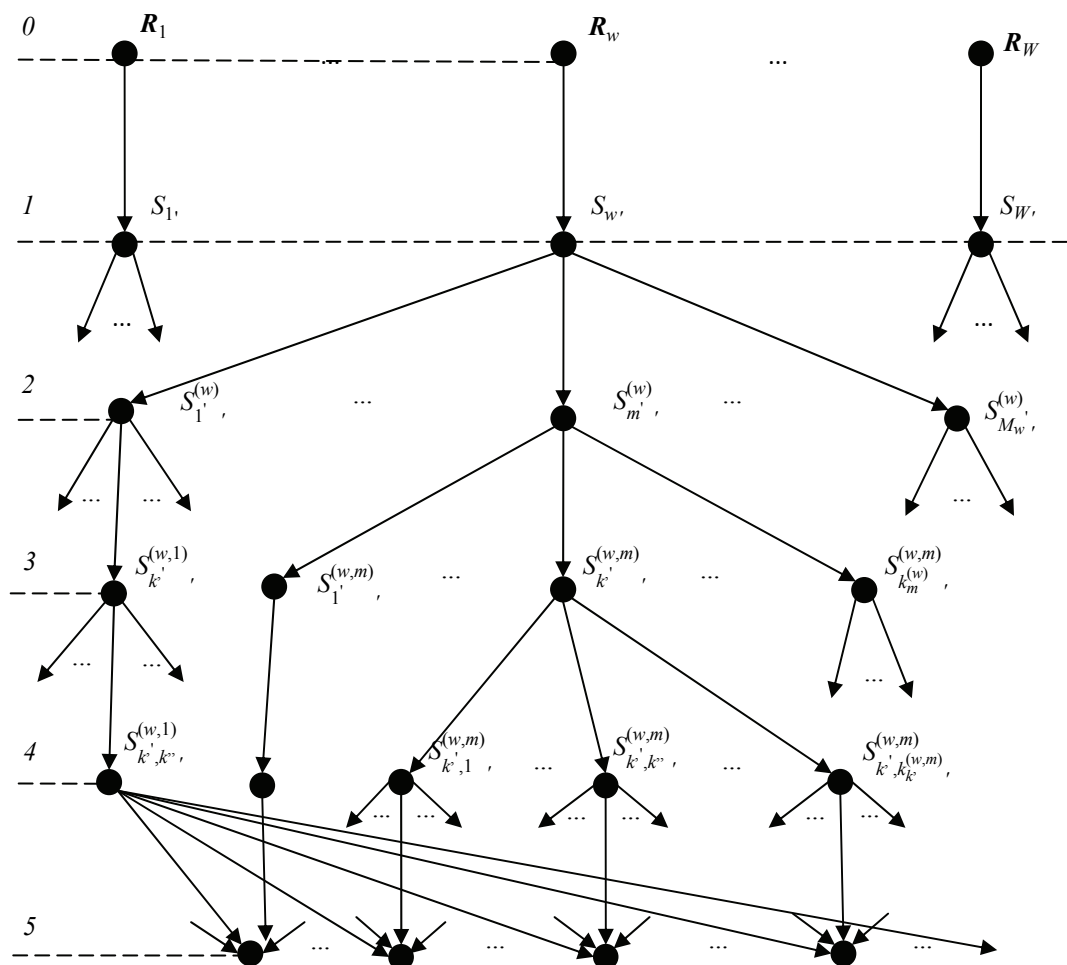
где $G_w^{BC}, \varphi_w^{BC}, R_w^{BC}$ и τ_w^{BC} имеют тот же смысл, что и в выражении для $AP1_w$; S_w – множество свертываемых вершин для вектора назначения R_w .

На ярусах 3–4 располагаются комбинации всех возможных вариантов сверток вершин, входящих во множества $\{S_w\}$ [2].

На пятом ярусе размещены априорные решения третьего уровня, формируемые в результате перебора всех вариантов сверток для каждого из векторов R_w как декартовы произведения M_w подмножеств четвертого яруса:

$$S_w = S_1^{(w)} \times S_2^{(w)} \times \dots \times S_{M_w}^{(w)} = \left\{ (S_1^{(w)}, \dots, S_2^{(w)}, \dots, S_{M_w}^{(w)}) : S_m^{(w)} \in S_m^{(w)}, \right\} \\ m = 1, 2, \dots, M_m$$

В этом случае общее число вариантов для всех векторов назначения будет вычисляться по выражению:



Граф поиска решений

$$N = \sum_{w=1}^W \prod_{m=1}^{M_w} \left(\sum_{k=2}^{K_m^{(w)}} K_m^{(w)}! / (k!(K_m^{(w)} - k)!) + 1 \right),$$

где M_w – мощность w -го подмножества S_w второго яруса; $K_m^{(w)}$ – мощность m -го варианта свертки для вектора R_w .

При этом количество анализируемых вариантов ВС увеличивается в $K = N/W$ раз, тем самым увеличивая пропорционально этому значению вероятность выбора оптимального варианта из всех возможных.

Заключение. Рассмотренная методика поиска множества априорных решений проектируемой системы позволит расширить дерево поиска альтернативных вариантов ВС, повысив

полноту анализа возможных работоспособных вариантов, обеспечивающих достоверность результатов проектирования ВС.

Литература

1. Кобайло, А. С. Теория синтеза вычислительных систем реального времени / А. С. Кобайло. – Минск: БГТУ, 2010. – 256 с.
2. Кобайло, А. С. Новые положения теории синтеза вычислительных систем реального времени в учебных курсах ВУЗов / А. С. Кобайло // Высшее техническое образование: проблемы и пути развития. Материалы VI Междунар. науч.-практ. конф. – Минск: БГУИР, 2012. – С. 222–223.

Поступила 05.03.2013