УДК. 621. 324

# РАСПРЕДЕЛЕНИЕ РАЗНОРОДНЫХ ИНФОРМАЦИОННЫХ ПОТОКОВ В ВЫЧИСЛИТЕЛЬНОЙ СЕТИ С ЦЕЛЬЮ МИНИМИЗАЦИИ ИХ СТОИМОСТНЫХ ХАРАКТЕРИСТИК

А.А. Пашнев
(представил д.т.н., проф.В.М. Илюшко)

Предложен алгоритм оптимального распределения разнородных информационных потоков в распределенных вычислительных сетях с целью минимизации стоимости аренды используемых каналов связи. Для решения оптимизационной задачи использован модифицированный симплекс - метод Данцига.

В настоящее время актуальной является задача оптимального распределения в сети нескольких разнородных потоков, связанных общим ограничением пропускных способностей, которая нашла свое применение в распределенных вычислительных сетях (РВС) с пакетной коммутацией, где информационных потоки, имеющие определенные пункты отправления и назначения, являются разнородными потоками с общими ограничениями на пропускные способности используемых каналов связи (КС) [1].

Особое место среди таких задач занимает задача оптимального распределения разнородных информационных потоков с целью минимизации стоимости их прохождения в неориентированной РВС.

Рассмотрим алгоритм ее решения. Задана РВС, которая характеризуется неориентированным, взвешенным по ребрам, графом $\mathbf{G} = <\mathbf{Z}, \mathbf{Y}, \mathbf{b}, \mathbf{c}>$ пропускных способностей КС и стоимостей передачи единицы информации по ним и графом $\mathbf{\Gamma} = <\mathbf{Z}, \mathbf{W}, \mathbf{r}>$ требований по передаче $\mathbf{m}$ информационных потоков между узлами РВС. Здесь $\mathbf{Z} = \{\mathbf{Z_1}, \ldots, \mathbf{Z}_l\}$ - перенумерованное множество вершин графов $\mathbf{G}$ и $\mathbf{\Gamma}$, находящихся в изоморфизме к узлам РВС, $l = |\mathbf{Z}|$ - число вершин графов $\mathbf{G}$ и $\mathbf{\Gamma}$; $\mathbf{Y} = \{\mathbf{y_1}, \ldots, \mathbf{y_n}\}$ - перенумерованное множество ребер графа $\mathbf{G}$, находящихся в изоморфизме к КС РВС, $\mathbf{n} = |\mathbf{Y}|$ - число ребер графа $\mathbf{G}$; $\mathbf{b} : \mathbf{Y} \to \mathbf{R}_+$ - весовая функция, определяющая каждому ребру $\mathbf{y_i} \in \mathbf{Y}$ пропускную способность $\mathbf{b_i}$, $\mathbf{1 \le i \le n}$; $\mathbf{c} : \mathbf{Y} \to \mathbf{R}_+$ - весовая функция, определяющая стоимость передачи единицы потока $\mathbf{C_i}$ по ребру $\mathbf{y_i} \in \mathbf{Y}$; $\mathbf{W} = \{\mathbf{W_1}, \ldots, \mathbf{W_m}\}$ - перенумерованное множество ребер графа $\mathbf{\Gamma}$. Ребро $\mathbf{W_j} \in \mathbf{W}$, $\mathbf{1 \le j \le m}$, соединяет две вершины графа $\mathbf{\Gamma}$, если между соответствующими узлами РВС производится обмен информацией. Величина потока информации, которая должна быть передана по ребру $\mathbf{W_j}$, задана весовой функцией $\mathbf{r: W \to R_+}$, определяющей каждому ребру $\mathbf{W_j \in W}$ графа $\mathbf{\Gamma}$ требуемый поток $\mathbf{r_j}$. Число информационных потоков, которые необходимо распределить в сети, равно $\mathbf{m = |W|}$.

Требуется при заданных графах $\mathbf{G}$ и $\mathbf{\Gamma}$ распределить потоки в сети таким образом, чтобы при удовлетворении требований графа $\mathbf{\Gamma}$ стоимость передачи потока в сети была минимальной.

Для изложения формальной постановки и алгоритма решения данной задачи введем ряд определений.

Распределение $\gamma$ потоков информации, передаваемой в РВС, определенной графами $\mathbf{G}$ и $\mathbf{\Gamma}$, математически представляется упорядоченными множествами $\mathbf{R'(\gamma)}$, $\mathbf{Q(\gamma)}$ и семейством $\mathbf{H(\gamma)}$. Здесь $\mathbf{R'(\gamma) = \{r'_1, \ldots, r'_m\}}$ - множество, устанавливающее объемы передачи информационных потоков в распределении $\gamma$ между каждой парой вершин, для которых в графе $\mathbf{\Gamma}$ имеет место ребро. Элемент $\mathbf{r'_j \in R'(\gamma)}$ определяет в распределении $\gamma$ реальный поток между $\mathbf{j}$ - й парой вершин графа $\mathbf{\Gamma}$. $\mathbf{Q(y) = \{q_1, \ldots, q_n\}}$ - множество, каждый элемент которого $\mathbf{q_i}$ есть суммарный поток по ребру $\mathbf{y_i \in Y}$ графа $\mathbf{G}$ в распределении $\gamma$. $\mathbf{H(\gamma) = \{H^1, \ldots, H^m\}}$ - семейство множеств $\mathbf{H^j = \{H^j_1, \ldots, H^j_{z(j)}\}}$, составленных из маршрутов, где передается $\mathbf{j}$ - й поток в распределении $\gamma$. Здесь $\mathbf{Z(j)}$ - число маршрутов для передачи $\mathbf{j}$ - го потока в распределении $\gamma$. Каждому маршруту $\mathbf{H^j_p, 1 \le p \le Z(j)}$, соответствует тройка $\mathbf{\langle A^j_p, d^j_p, x^j_p \rangle}$, где $\mathbf{A^j_p = \left\{ a^j_{1p}, \ldots, a^j_{d^j_p p} \right\}}$ - множество, определяющее состав маршрута (в перечне ребер) графа $\mathbf{G}$, каждый элемент которого $\mathbf{a^j_{sp}}$, $\mathbf{1 \le s \le d^j_p}$, есть ребро множества $\mathbf{Y}$ графа $\mathbf{G}$; $\mathbf{d^j_p}$ - длина маршрута $\mathbf{H^j_p}$; $\mathbf{x^j_p}$ - величина $\mathbf{j}$ - го потока по маршруту $\mathbf{H^j_p}$. Отметим, что в множестве $\mathbf{A^j_p}$ элементы $\mathbf{a^j_{sp}}$ расположены в соответствии с их порядком следования по маршруту.

Используя параметры маршрута $\mathbf{H^j_p}$ определим значение

$$\mathbf{q_i \in Q(\gamma): \forall y_i \in Y \exists\; q_i = \sum_{j=1}^{m} \sum_{p=1}^{z(j)} x^j_p h^j_p},$$

где
$$\mathbf{h^j_p} = \begin{cases} \mathbf{0}, & \text{если } \mathbf{y_i \notin A^j_p}; \\ \mathbf{1}, & \text{если } \mathbf{y_i \in A^j_p}, \end{cases}$$

и значение $\mathbf{r'_j \in R'(\gamma): \forall W_j \in W \exists r'_j = \sum_{p=1}^{z(j)} x_p^j}$ .

Дадим формальную постановку задачи. Даны графы **G = < Z, Y, b, c>** и **Γ = <Z, W, r>**. Требуется распределить потоки в сети, то есть сформировать множества **R′(γ)**, **Q(γ)** и семейство **H(γ)** таким образом, чтобы величина $\mathbf{F(\gamma) = \sum_{j=1}^{m} \sum_{p=1}^{z(j)} C_p^j X_p^j}$ принимала минимальное значение при выполнении следующих условий:

1) $\mathbf{\sum_{p=1}^{z(j)} X_p^j - R_j = r_j}$ ;

2) $\mathbf{\forall y_i \in Y \exists q_i \in Q(\gamma) \wedge b_i \in b(Y) | q_i + S_i = b_i}$ ;

3) $\mathbf{X_p^j,\ R_j, S_i \ge 0}$ ,

где $\mathbf{C_p^j}$ - стоимость передачи единицы **j** - го потока по $\mathbf{H_p^j}$ маршруту;

$\mathbf{R_j, S_i}$ - слабые переменные.

Величина $\mathbf{C_p^j}$ определяется выражением

$$\mathbf{C_p^j = \sum_{s=1}^{d_p^j} C_s} ,$$

где $\mathbf{C_s}$ - стоимость передачи единицы потока информации по ребру $\mathbf{a_{sp}^j}$ , входящему в состав маршрута $\mathbf{H_p^j}$ .

Для решения данной задачи с использованием модифицированного симплекс - метода Данцига [2] необходимо задать начальное допустимое распределение $\boldsymbol{\gamma_0}$, точнее семейство маршрутов $\mathbf{H(\gamma_0)}$. Распределение $\boldsymbol{\gamma_0}$ можно найти, решив задачу о максимальном, допустимом для заданной сети, разнородном информационном потоке. Формально задача о максимальном допустимом информационном потоке может быть поставлена следующим образом.

Даны графы **G = < Z, Y, b, c>** и **Γ = <Z, W, r>**. Требуется максимизировать $\boldsymbol{\lambda = \sum_{j=1}^{m} \sum_{p=1}^{Z(j)} X_p^j}$ при условиях:

1) $\mathbf{\forall y_i \in Y \exists q_i \in Q(\gamma) \wedge b_i \in b(Y) | q_i + S_i = b_i}$ ;

2) $\mathbf{X_p^j, S_i \ge 0}$ .

Обозначим функцией $\mu$ долю суммарного требуемого потока, которая реализуется в распределении $\gamma$. В ходе дальнейшего изложения алгоритма воспользуемся векторной формой представления данных и результатов распределения. Сопоставим функциям $\mathbf{b}$ и $\mathbf{c}$ векторы $\tilde{\mathbf{b}}$ и $\tilde{\mathbf{c}}$, где $\tilde{\mathbf{b}} = \left(\tilde{\mathbf{b}}_1,...,\tilde{\mathbf{b}}_n, \tilde{\mathbf{b}}_{n+1}\right)$ - вектор пропускных способностей ребер графа **G**, $\tilde{\mathbf{b}}_1 = \mathbf{b}_i$, $\tilde{\mathbf{b}}_{n+1} = \mu$ (в начале выполнения $\tilde{\mathbf{b}}_{n+1} = 0$); $\tilde{\mathbf{C}} = \left(\tilde{\mathbf{C}}_1,...,\tilde{\mathbf{C}}_n\right)$ - вектор стоимости (длин) ребер графа **G**, $\tilde{\mathbf{C}}_i = \mathbf{C}_i$. Также имеем: единичную матрицу $\overline{\mathbf{B}}_0$ размера $(\mathbf{n+1})\times(\mathbf{n+1})$ (дополнительно вводится фиктивная переменная), в которой $\overline{\mathbf{b}}_{\alpha,\beta} = 1$, если $\alpha=\beta$, $1 \le \alpha \le (n+1)$, $1 \le \beta \le (n+1)$, и $\overline{\mathbf{b}}_{\alpha,\beta} = 0$ если $\alpha \ne \beta$, матрицу текущего базиса $\overline{\mathbf{B}}^{-1}$ (в начале выполнения алгоритма $\overline{\mathbf{B}}^{-1} = \overline{\mathbf{B}}_0$). Распределение $\gamma$ определяется текущей симплексной таблицей (матрицей $\overline{\mathbf{B}}^{-1}$ и вектором $\tilde{\mathbf{b}}$).

Алгоритм построения максимального допустимого потока имеет итерационный характер. На каждой **k**- й итерации производится модификация симплексной таблицы.

Шаг 1. Принимая за длину ребер значения вектора $\tilde{\mathbf{c}}$, находим в графе **G** для каждого требуемого потока графа **Γ** (ребра $\mathbf{W}_j \in \mathbf{W}$) кратчайший маршрут, например, с помощью метода Дейкстры [3] и не учитывая ограничения на пропускную способность ребер, реализуем по этому маршруту весь поток $\mathbf{r}_j \in \mathbf{r(W)}$. Параметры маршрута заносятся в семейство множеств $\mathbf{H}(\gamma_k)$. В результате имеем распределение суммарного потока по ребрам графа **G**, представленное вектором $\tilde{\mathbf{q}}^k = (\tilde{\mathbf{q}}_1^k,...,\tilde{\mathbf{q}}_n^k, \tilde{\mathbf{q}}_{n+1}^k)$,

где $$\tilde{\mathbf{q}}_i^k = \sum_{j=1}^{m} \mathbf{r}_j \mathbf{h}_j \left| \mathbf{h}_j = \begin{cases} 0, & \text{если} \quad \mathbf{y}_i \notin \mathbf{A}^j; \\ 1, & \text{если} \quad \mathbf{y}_i \in \mathbf{A}^j, \end{cases} \right.$$

$$\tilde{\mathbf{q}}_{n+1}^k = -1;$$

$\mathbf{A}^j$ - состав кратчайшего маршрута между **j** - й парой вершин графа **Γ**, $\mathbf{A}^j \in \mathbf{H}(\gamma_k)$.

Вектор $\tilde{\mathbf{q}}^k$ вводится в симплекс - таблицу.

Шаг 2. Среди положительных компонент вектора $\tilde{\mathbf{q}}^k$ определяется $\tilde{\mathbf{q}}_i^k$ с минимальным значением отношения $\tilde{\mathbf{b}}_i^{k-1}/\tilde{\mathbf{q}}_i^k$ (здесь $\tilde{\mathbf{b}}_i^{k-1}$ - компонента **i** вектора $\tilde{\mathbf{b}}^{k-1}$ текущих пропускных способностей ребер графа **G**). Номер **i** выбранной компоненты определяет ведущую строку симплексной таблицы. Сформулируем новое значение вектора $\tilde{\mathbf{q}}^{\mathbf{k}*}$. Для этого разделим все, кроме $\tilde{\mathbf{q}}_{\mathbf{i}}^{\mathbf{k}}$ компоненты вектора $\tilde{\mathbf{q}}^{\mathbf{k}}$ на $\left(-\tilde{\mathbf{q}}_{\mathbf{i}}^{\mathbf{k}}\right)$ (отрицательное значение ведущего), а значение ведущего установим равным $\mathbf{1}/\tilde{\mathbf{q}}_{\mathbf{i}}^{\mathbf{k}}$.

<u>Шаг 3.</u> Заменим в исходной единичной матрице $\overline{\mathbf{B}}_{\mathbf{0}}$ **i** - й столбец вектором $\tilde{\mathbf{q}}^{\mathbf{k}*}$ и получим матрицу $\overline{\mathbf{B}}_{\mathbf{0}}^{\mathbf{k}}$. Можем сформировать текущую симплексную таблицу, т.е. формируем: матрицу текущего базиса $\overline{\mathbf{B}}_{\mathbf{k}}^{-1} = \overline{\mathbf{B}}_{\mathbf{0}}^{\mathbf{k}} \cdot \overline{\mathbf{B}}_{\mathbf{k-1}}^{-1}$ и новый вектор пропускных способностей $\tilde{\mathbf{b}}^{\mathbf{k}} = \overline{\mathbf{B}}_{\mathbf{0}}^{\mathbf{k}} \cdot \tilde{\mathbf{b}}^{\mathbf{k-1}}$, в которой вектор $\tilde{\mathbf{q}}^{\mathbf{k}}$ меняет текущую базисную переменную ведущей строки. Для построенного распределения потоков получим значение целевой функции $\mu_{\mathbf{k}} = \tilde{\mathbf{b}}_{\mathbf{n+1}}^{\mathbf{k}}$, т.е. долю от суммарного требуемого потока, которая реализуется в распределении $\gamma_k$, определяемом текущей симплексной таблицей и семейством $\mathbf{H}(\gamma_k)$.

Если $\mu_k \geq 1$, то на этом работа алгоритма построения максимального допустимого потока заканчивается. При $\mu_k < 1$ из множества ребер **Y** графа **G** удаляется ребро $y_i$, соответствующее **i** - й ведущей строке симплексной таблицы, и процесс построения максимального допустимого потока циклически продолжается до тех пор (выполняются шаги 1 - 3), пока либо $\mu_k \geq 1$, либо не будет возможно построить ни один маршрут, позволяющий распределить оставшиеся требуемые потоки $\mathbf{r_j} \in \mathbf{r}(\mathbf{W})$ графа **Г**. В последнем случае можно сказать, что заданная РВС, определенная графами **G** и **Г**, позволяет распределить требуемые потоки информации величиной не более $\mu_k$ от заданного значения, т.е.

$$\forall \mathbf{W_j} \in \mathbf{W} \exists \mathbf{r_j} \in \mathbf{r}(\mathbf{W}) \left| \sum_{\mathbf{p=1}}^{\mathbf{z(j)}} \mathbf{X_p^j} \leq \mu_{\mathbf{k}} \cdot \mathbf{r_j} \right. .$$

Полученное при этом распределение потока $\gamma_0 = \gamma_k$, определяемое семейством маршрутов $\mathbf{H}(\gamma_0)$ совместно с вектором пропускных способностей $\tilde{\mathbf{b}}$, используется для построения потоков минимальной стоимости.

Исходные данные для решения задачи построения потоков минимальной стоимости: число переменных симплекс – таблицы, равное **(n+m+1)** дополнительно вводится фиктивная переменная); вектор пропускных способностей ребер графа **G** $\tilde{\mathbf{f}} = \left(\tilde{\mathbf{f}}_{\mathbf{1}},\ldots,\tilde{\mathbf{f}}_{\mathbf{n}},\tilde{\mathbf{f}}_{\mathbf{n+1}},\ldots,\tilde{\mathbf{f}}_{\mathbf{n+m}},\tilde{\mathbf{f}}_{\mathbf{n+m+1}}\right)$, где $\tilde{\mathbf{f}}_{\mathbf{i}} = \tilde{\mathbf{b}}_{\mathbf{i}}$; $\tilde{\mathbf{f}}_{\mathbf{n+j}} = -\mathbf{r_j}$; $\tilde{\mathbf{f}}_{\mathbf{n+m+1}} = \mathbf{F}$ (в начале выполнения алгоритма $\tilde{\mathbf{f}}_{\mathbf{n+m+1}} = \mathbf{0}$); вектор стоимости (длин) ребер графы **G** $\tilde{\mathbf{C}} = \left(\tilde{\mathbf{C}}_{\mathbf{1}},\ldots\tilde{\mathbf{C}}_{\mathbf{n}}\right)$, где

$\widetilde{C}_i = C_i$; единичная матрица $\overline{B}_0$ размером $(n+m+1)\times(n+m+1)$; матрица текущего базиса $\overline{B}^{-1}$ (в начале выполнения $\overline{B}^{-1} = \overline{B}_0$); семейство маршрутов $H(\gamma_0)$.

Шаг 1.

Маршруты семейства $H(\gamma_0)$ последовательно вводятся в симплекс - таблицу. Вводимый в таблицу маршрут, для передачи $j$ - го потока, представляется вектором $\widetilde{X}_p^j = \left(\widetilde{X}_1,\ldots,\widetilde{X}_n,\widetilde{X}_{n+1},\ldots,\widetilde{X}_{n+m},\widetilde{X}_{n+m+1}\right)$, где компонент $\widetilde{X}_i$ равен единице, если соответствующее индексу $i$ ребро входит в данный маршрут и равен нулю в остальных случаях; $\widetilde{X}_{n+j} = -1$ - коэффициент при $X_p^j$ в уравнении $R_j - \sum_{p=1}^{z(j)} X_p^j = -r_j$; $\widetilde{X}_{n+m+1} = C_p^j$ - стоимость передачи единицы $j$ - го потока по вводимому маршруту.

Перед вводом в симплекс - таблицу вектор $\widetilde{X}_p^j$ корректируется в соответствии с выражением $\widetilde{X}_p^{jk} = \overline{B}_0^{k-1} \cdot \widetilde{X}_p^j$. Среди положительных компонент $i$ вектора $\widetilde{X}_p^{jk}$ определяется такая $\widetilde{X}_i^k$, для которой отношение $\widetilde{f}_i^{k-1} \big/ \widetilde{x}_i^k$ имеет минимальное значение. Номер $i$ выбранной компоненты определяет ведущую строку симплекс - таблицы.

Шаг 2.

Осуществим ввод вектора $\widetilde{X}_p^{jk}$ в базис и сформируем текущую симплексную таблицу в соответствии со следующей процедурой.

Получим новое значение вектора $\widetilde{X}_p^{jk*}$. Для этого разделим все, кроме $\widetilde{X}_i^k$, компоненты вектора $\widetilde{X}_p^{jk}$ на $(-\widetilde{X}_i^k)$ (отрицательное значение ведущего), а значение ведущего установим равным $1 \big/ \widetilde{X}_i^k$. Заменим в исходной единичной матрице $\overline{B}_0$ $i$ - й столбец вектором $\widetilde{X}_p^{jk*}$ и получим матрицу $\overline{B}_0^k$. Сформируем текущую симплексную таблицу (формируем: матрицу текущего базиса $\overline{B}_k^{-1} = \overline{B}_0^k \cdot \overline{B}_{k-1}^{-1}$ и новый вектор пропускных способностей $\widetilde{f}^k = \overline{B}_0^k \cdot \widetilde{f}^{k-1}$), в которой вектор $\widetilde{X}_p^{jk}$ меняет текущую базисную переменную ведущей строки. Для построенного распределения потоков получим значение целевой функции $F\left(\gamma_k\right) = \widetilde{f}_{n+m+1}^k$, которое определяет стоимость передачи потоков информации в распределении $\gamma_k$. После окончания ввода маршрутов, входящих в семейство $H(\gamma_0)$, определяющих начальное базовое распределение потоков информации, производится перераспределение базовых потоков в целях минимизации функционала **F**.

Шаг 3.

Анализируются компоненты **(n + m + 1)** - й строки матрицы текущего базиса $\overline{\mathbf{B}}_{\mathbf{k}}^{-1}$. Если среди анализируемых компонентов $\overline{\mathbf{b}}_{\mathbf{n+m+1,t}}^{\mathbf{k}}$, $\mathbf{1 \le t \le (n+m+1)}$, существуют компоненты со значением меньше нуля, то из $\overline{\mathbf{b}}_{\mathbf{n+m+1,t}}^{\mathbf{k}} < \mathbf{0}$ выбираем компоненту $\overline{\mathbf{b}}_{\mathbf{n+m+1,r}}^{\mathbf{k}}$, $\mathbf{1 \le r \le (n+m+1)}$, удовлетворяющую условию $\overline{\mathbf{b}}_{\mathbf{n+m+1,r}}^{\mathbf{k}} = \min\limits_{\mathbf{t}} \overline{\mathbf{b}}_{\mathbf{n+m+1,t}}^{\mathbf{k}} < \mathbf{0}$. Столбец **r** - ведущий. Для ввода ведущего столбца в базис, с целью минимизации **F**, среди компонент $\overline{\mathbf{b}}_{\mathbf{t,r}}^{\mathbf{k}} > \mathbf{0}$ находим такую $\overline{\mathbf{b}}_{\mathbf{a,r}}^{\mathbf{k}}$, $\mathbf{1 \le a \le (n+m+1)}$, для которой отношение $\tilde{\mathbf{f}}_{\mathbf{t,r}}^{\mathbf{k}} \big/ \overline{\mathbf{b}}_{\mathbf{t,r}}^{\mathbf{k}}$ принимает минимальное значение, $\tilde{\mathbf{f}}_{\mathbf{a,r}}^{\mathbf{k}} \big/ \overline{\mathbf{b}}_{\mathbf{a,r}}^{\mathbf{t}} = \min\limits_{\mathbf{t}} \left( \tilde{\mathbf{f}}_{\mathbf{t,r}}^{\mathbf{k}} \big/ \overline{\mathbf{b}}_{\mathbf{t,r}}^{\mathbf{k}} \right)$. Строка **a** - ведущая. Сформируем новые значения компонент $\overline{\mathbf{b}}_{\mathbf{t,r}}^{\mathbf{k*}}$ ведущего столбца матрицы $\overline{\mathbf{B}}_{\mathbf{k}}^{-1*}$. Для этого разделим все, кроме $\overline{\mathbf{b}}_{\mathbf{a,r}}^{\mathbf{k}}$ компоненты $\overline{\mathbf{b}}_{\mathbf{t,r}}^{\mathbf{k}}$ на $(-\overline{\mathbf{b}}_{\mathbf{a,r}}^{\mathbf{k}})$ (отрицательное значение ведущего), а значение ведущего установим равным $\mathbf{1} \big/ \overline{\mathbf{b}}_{\mathbf{a,r}}^{\mathbf{k}}$.

Шаг 4.

Заменим в исходной единичной матрице $\overline{\mathbf{B}}_{\mathbf{0}}$ **a** - й столбец на вновь сформированный столбец **r** матрицы $\overline{\mathbf{B}}_{\mathbf{k}}^{-1*}$ и получим матрицу $\overline{\mathbf{B}}_{\mathbf{0}}^{\mathbf{k+1}}$. Теперь можем сформировать текущую симплексную таблицу, т.е. формируем матрицу текущего базиса $\overline{\mathbf{B}}_{\mathbf{k+1}}^{-1} = \overline{\mathbf{B}}_{\mathbf{0}}^{\mathbf{k+1}} \cdot \overline{\mathbf{B}}_{\mathbf{k}}^{-1}$ и новый вектор пропускных способностей $\tilde{\mathbf{f}}^{\mathbf{k+1}} = \overline{\mathbf{B}}_{\mathbf{0}}^{\mathbf{k+1}} \cdot \tilde{\mathbf{f}}^{\mathbf{k}}$, в котором базисная переменная ведущей строки **a** меняется переменной ведущего столбца **r**. Для построенного распределения потоков получим новое значение целевой функции $\mathbf{F}(\gamma_{\mathbf{k+1}}) = \tilde{\mathbf{f}}_{\mathbf{n+m+1}}^{\mathbf{k+1}}$.

Если в полученном распределении $\gamma_{\mathbf{k+1}}$ в **(n+m+1)** - й строке имеется хотя бы один компонент $\overline{\mathbf{b}}_{\mathbf{n+m+1,t}}^{\mathbf{k+1}}$ со значением меньше нуля, то работа алгоритма повторяется, начиная с шага 3. Если же компоненты $\overline{\mathbf{b}}_{\mathbf{n+m+1,t}}^{\mathbf{k+1}} \ge \mathbf{0}$, то формируется вектор $\tilde{\mathbf{C}}^* = \left| \tilde{\mathbf{C}}_{\mathbf{1}}^*, \ldots, \tilde{\mathbf{C}}_{\mathbf{n}}^* \right|$ текущей стоимости передачи единицы потока информации для каждого ребра множества **Y** графа **G**, $\tilde{\mathbf{C}}_{\mathbf{i}}^* = \tilde{\mathbf{C}}_{\mathbf{i}} + \overline{\mathbf{b}}_{\mathbf{n+m+1,i}}^{\mathbf{k+1}}$. Принимая за длину ребер значение вектора $\overline{\mathbf{C}}^*$, находим в графе **G** для каждого требуемого потока **j** графа **Г** кратчайший маршрут, например, с помощью метода Дейкстры. Найденный маршрут $\mathbf{H}^{\mathbf{j}}_{\mathbf{p+1}}$ представляется вектором

$$\widetilde{\mathbf{X}}^{\mathbf{j}}_{\mathbf{p+1}} = \left(\widetilde{\mathbf{x}}_1,\ldots,\widetilde{\mathbf{x}}_n, \widetilde{\mathbf{x}}_{n+1},\ldots,\widetilde{\mathbf{x}}_{n+m}, \widetilde{\mathbf{x}}_{n+m+1}\right).$$

Тогда, если $\mathbf{H}^{\mathbf{j}}_{\mathbf{p+1}} \in \mathbf{H}(\gamma_0)$, то получено оптимальное решение задачи построения потоков минимальной стоимости и работа алгоритма заканчивается, иначе найденный маршрут $\mathbf{H}^{\mathbf{j}}_{\mathbf{p+1}}$ вводится в семейство маршрутов $\mathbf{H}(\gamma_0)$ и процесс построения потоков минимальной стоимости продолжается начиная с шага 1. Результирующее оптимальное распределение $\gamma = \gamma_{k+1}$ определяется маршрутами семейства $\mathbf{H}(\gamma)$, множествами $\mathbf{R}'(\gamma)$, $\mathbf{Q}(\gamma)$ и целевой функцией $\mathbf{F}(\gamma)$.

## ЛИТЕРАТУРА

1.Королев А.В., Кучук Г.А., Пашнев А.А. Распределение информационных потоков в вычислительных сетях // Інформаційно – керуючі системи на залізничному транспорті. – 1998 - № 6. – С. 47 – 50.

2. Ху. Т. Целочисленное программирование и потоки в сетях. – М.: Мир, 1974. – 519 с.

3. Майника Э. Алгоритмы оптимизации на сетях и графах. – М.: Мир, 1981. - 323 с.

**УДК 519.34**

# О ВОЗМОЖНОСТЯХ УЧЕТА АЛГОРИТМИЧЕСКИХ ОСОБЕННОСТЕЙ ЗАДАЧ

В.Н. Кондратьев, к.т.н. О.И. Богатов, Е.В. Кондратьева
(представил д.т.н., проф. Г.А. Поляков)

Рассматривается способ повышения производительности современных мультипроцессорных вычислительных систем, основанный на использовании в качестве процессорных элементов программируемых логических устройств.

Успехи микроэлектроники за последние десятилетия резко расширили границы алгоритмических возможностей в вычислительной технике. Это произошло за счет значительного увеличения быстродействия и степени интеграции элементов вычислительных устройств, объясняющегося рядом технологических достижений, а также распараллеливания и конвейеризации вычислительного процесса, которые стали возможными благодаря улучшению надежностных и экономических показателей процессорных элементов (ПЭ). Организацию вычислительного процесса можно интерпретировать как преобразование алгоритма, составленного в терминах операторов пользователя, в машинный алгоритм в терминах микроопераций. Традиционный путь состоит в программном преобразовании алгоритма пользователя в машинный алгоритм и непосредственной схемной реализации машинного алгоритма средствами комбинационной и регистровой логики. С появлением мультипроцессорных вычислительных систем (МВС) граница подобного деления стала в некоторой степени условной за счет возможностей реконфигурирования системы[1], обычно не затрагивающей структуру ПЭ. ПЭ, как правило, имеют традиционную систему команд, ориентированную на работу не более чем с двумя аргументами, фиксированную разрядность данных, традиционные схемы организации памяти, то есть никак не отражают в себе алгоритмические особенности решаемых задач.

В числе требований к основным техническим характеристикам специализированных МВС, входящих в состав ТТЗ, обычно доминирующую роль играют требования к производительности. Удовлетворение требований по производительности достигается за счет увеличения тактовой частоты и наращивания числа ПЭ в системе с дальнейшей их организацией в параллельную и/или конвейерную структуру, что обусловлено лишь продолжающимся техническим прогрессом в технологии электронных приборов. Подобный путь развития вычислительной техники можно назвать пассивным, так как он целиком зависит от успехов в области соответствующих технологий и не затрагивает собственных проблем информатики и смежных дисциплин. Стремление сохранить в мировой практике проектирования общие принципы, заложенные в существующих технических решениях, диктуется также и коммерческими соображениями. Так, ведущие производители электронной продукции организовали достаточно эффективную разработку и массовый выпуск сравнительно недорогих однокристальных компьютеров, которые находят широкий сбыт на мировом рынке.

В то же время технологические достижения в части повышения производительности имеют определенный предел. Это объясняется тем, что увеличение производительности связано с дальнейшей степенью микроминиатюризации, которая при достижении определенных размеров приводит к взаимному влиянию полупроводниковых приборов в кристалле, что при размерах соединительных линий 0,4 мкм [2] приводит к эффектам, которые не могут быть устранены при существующих методах архитектурного проектирования. Хотя ширина соединительных и изолирующих линий [3] равна от **$2\lambda$** до **$6\lambda$** при достигнутом в настоящее время значении технологического параметра **$\lambda = 0{,}25$** мкм, данное обстоятельство свидетельствует о близости к указанному пределу. Поэтому в течение последних десятилетий ведется интенсивный поиск новых решений, которые могут быть положены в основу перспективных МВС.

Одним из путей решения данной проблемы является создание автоматизированной системы проектирования МВС, в основу ПЭ которых положены программируемые логические интегральные схемы (ПЛИС), архитектура которых может быть изменена пользователем, исходя из алгоритмических особенностей решаемой задачи, что даст возможность в среднем на порядок повысить производительность МВС.

Теоретической основой подобного подхода должно стать создание методики разбиения алгоритма решаемой задачи на определенные фрагменты, удовлетворяющие следующим требованиям: каждый фрагмент должен реализовать в составе алгоритма определенную функцию, заданную в общепринятых математических терминах; фрагмент должен иметь возможность быть реализованным на уровне комбинационной логики и элементов памяти; разбиение на фрагменты должно преследовать цели обеспечения регулярности структуры компьютерной системы, реализующей заданный алгоритм. Указанные требования являются основой для автоматизированной разработки специализированных ПЭ, реализующих соответствующие фрагменты.

Исходя из вышеизложенного, процесс логического проектирования МВС на основе ПЛИС для конкретной задачи, будет состоять из трех последовательных этапов:

1) представление алгоритма задачи в терминах соответствующих фрагментов и операторов управления;

2) перевод алгоритма в терминах фрагментов в некоторую абстрактную пространственно - временную модель, однозначно определяющую структуру МВС со всеми информационными связями между ее элементами, а также временную последовательность их работы;

3) перевод абстрактной пространственно - временной модели в схему на конкретных микросхемах ПЛИС.

В такой постановке процесс проектирования представляется в виде ряда причинно связанных друг с другом этапов, имеющих четкую математическую подоплеку. Как показывает анализ, этап 1 является наиболее ответственным, требующим творческого подхода, в конечном итоге определяющим качество работы системы. На данном этапе требуется привлечение современных методов вычислительной математики, а также разработка новых с учетом указанных выше требований.

Решение задачи этапа 2 стало осуществимым благодаря построению некоторой абстрактной математической модели элемента вычислительного процесса и организации математического аппарата формальных преобразований, базирующихся на этой модели, которые позволили с единых позиций синтезировать множество всех пространственно - временных описаний схемных решений. Этот этап, согласно предварительной проработке, хорошо формализуется и не содержит принципиальных препятствий, не допускающих его полной автоматизации.

Этап 3 также может быть полностью автоматизирован. Однако следует иметь в виду, что он должен предусматривать большое количество различных вариантов, зависящих, с одной стороны, от типа ПЛИС, а с другой, от желания пользователя организовать одни и те же функции различным способом, например, выбрать адресную память или сдвиговые регистры, **D** – триггеры или **T** - триггеры и т.д. К тому же, при большом количестве ПЭ в МВС практически невозможно организовать непосредственные связи между каждой парой ПЭ. Поэтому становится актуальной задача разработки принципов организации регулярной сети обмена между ПЭ, оптимальной по времени его реализации.

Принципы организации указанных выше этапов проектирования были частично верифицированы на примерах построения проектов устройств для сортировки в рамках пакета PLDshell [4], основанных на параллельном алгоритме [5]. Создание теоретической базы логического проектирования МВС на основе ПЛИС позволит устранить парадокс, когда самая «интеллектуальная» научная дисциплина в своем перспективном развитии методично следует по пятам технологии в микроэлектронной промышленности.

## ЛИТЕРАТУРА


1. С. Кун. Матричные процессоры на СБИС. – М.: Мир, 1991.- 672 с.

2. Д.Ферри, Л.Эйкерс, Э.Гринич. Электроника ультрабольших интегральных схем. - М.: Мир, 1991. - 328 с.

3. Дж. Д. Ульман. Вычислительные аспекты СБИС. - М.: Радио и связь, 1990. - 480 с.

4. PLDshell Plus/Pldasm User's Guide. V3.0, Intel Corp., Order Number: 611943 - 001, 1992.

5. Кондратьев В.Н., Богатов О.И., Горовой С.И. Параллельный алгоритм сортировки // НТС ХВУ, № 14. – Харьков : ХВУ,1998. – С. 67 – 70.