УДК 51.330.115+519.685

МОДЕЛЬ ПРОЦЕССА КОМПОЗИЦИИ ЛОКАЛЬНЫХ РЕШЕНИЙ С УЧЕТОМ КООПЕРАЦИИ

В.Л.Маршак, С.М.Анцыз, В.Г.Чупин

Задача построения оптимального плана развития производства экономической системы, реализуемая в процессе координации локальных планов подсистем, состоит в согласовании их общесистемных связей относительно достижения оптимального значения функционала.

При построении плана системы в условиях распределенных баз данных по объектам управления (подсистемам) в качестве предмета согласования выступают связи по общим ресурсам, общим продуктам и по продуктам, участвующим во внутрисистемной кооперации.

Ведущим моментом в процессе согласования перечисленных связей подсистем является процесс распределения капитальных вложений как общесистемного ресурса, который в конечном счете определяет динамику выпуска всех видов продукции подсистем, и уровни кооперации и специализации подсистем.

Процесс распределения общесистемных ресурсов в условиях, когда подсистемы характеризуются наличием индивидуальных и общих продуктов, подробно изложен и исследован в работах [1,2]. Там показано, что процесс приводит к достижению оптимального решения при нескольких общесистемных ресурсах. В настоящей работе предлагается дальнейшая модификация процесса построения общесистемного оптимального плана, направленного на получение совокупности оптимальных равноэффективных планов подсистем (см.[1]) и учитывающего их связи по кооперации.

Одним из наиболее сложных вопросов в формировании сбалансированных и оптимальных планов развития производственных систем является построение плана кооперированных поставок для входящих в нее подсистем, согласованного в рамках оптимального комплексного плана с другими его разделами: планом производства, капитального строительства и т.д.

В том случае, когда в качестве объекта планирования выступает отраслевая система типа машиностроительного министерства, можно утверждать, что характер кооперации подсистем, заданной в виде затрат изделий на изделие, может быть описан в виде ориентированного графа, в котором отсутствуют ориентированные циклы и, следовательно, по крайней мере одна из вершин не имеет входных дуг и одна вершина не имеет исходящих дуг. Также предполагается, что в каждой из подсистем каждому продукту соответствует одна технология его производства.

Ориентированная (последовательная) кооперация подсистем позволяет воспользоваться спецификой матрицы при моделировании процесса планирования. В случае последовательной кооперации матрица производства и распределения продукции подсистем перестановкой столбцов и строк может сводиться к блочно-треугольной, что позволяет для формирования общесистемного плана использовать достаточно простые вычислительные процедуры определения полной заявки на продукцию системы с учетом всех связей по внутренней кооперации. Достаточно подробно данный подход описан в работе В.В.Шафранского [3].

Расчет полного конечного выпуска системы (полной заявки) позволяет преобразовать исходную матрицу производства и распределения продукции подсистем в квазидиагональную, учитывая, что исходная матрица квадратная, так как в каждой из подсистем на каждый продукт имеется одна технология (способ) его производства. В данной постановке для координации планов подсистем может быть применен алгоритм построения совокупности оптимальных равноэффективных планов подсистем, описанный для случая распределения общесистемных ресурсов в [I]. Однако в этом случае задачи подсистемы будут достаточно большой размерности, поскольку в них необходимо учитывать всю номенклатуру изделий нижнего уровня кооперации.

I. Постановка задачи

Рассмотрим иной подход к отражению связей подсистем по кооперации. В математико-экономических моделях планирования развития производства экономических систем принята запись технологий (способов) производства продукции, в которых связи по кооперации отражаются как затраты на данный продукт продуктов, производимых другими способами, т.е. как

$$(-a_{1,j}, \dots, -a_{j-1,j}, a_{j,j}, -a_{j+1,j}, \dots, a_{n,j}). \qquad (I)$$

Здесь компонента a с номером $i = j$ указывает на количество производимой продукции при единичной интенсивности применения способа j, а компоненты с номерами $i \neq j$ – потребление продукции, производимой другими способами при единичной интенсивности способа j.

Матрица производства и распределения продукции в системах с ориентированной (последовательной) кооперацией, сформированная из столбцов вида (I), как уже отмечалось, может приводиться к блочно-треугольной.

Предлагаемый в работе подход к моделированию связей по кооперации основывается на иной форме представления технологий производства продукции. Рассматриваются только конечные продукты системы, на которые поступает задание на выпуск в виде заявки либо в другой форме, отражающей влияние внешней среды на систему.

Технологический способ производства каждого конечного продукта системы записывается в виде затрат мощностей (по видам) каждой из подсистем, участвующей в комплексации по данному продукту. В этом случае матрица производства продуктов системы будет единичной (при записи в способах затрат на единицу производимой продукции), а матрица кооперации фондов – прямоугольной. Перейдем к формулировке задачи определения оптимальных связей по внутрисистемной кооперации как к задаче кооперации мощностей.

Обозначения:

X^k – вектор интенсивностей применения технологических способов подсистемы k, $k = \overline{1, l}$;

Y^k – вектор интенсивностей ввода мощностей в подсистеме k;

z – уровень реализации общесистемной заявки;

F^{kk} – матрица коэффициентов затрат мощностей подсистемы k на конечные продукты, производимые в подсистеме k;

F^{kj} – матрица коэффициентов затрат мощностей подсистемы k на продукцию, производимую в подсистеме $j, j=\overline{1,l}, j \neq k$;

R^{k} – матрица коэффициентов затрат капвложений на единицу прироста мощностей в подсистеме k;

d^{k} – вектор заявки на продукцию подсистемы k;

N^{k} – вектор ограничений по базовым мощностям, имеющимся в подсистеме k на начало планового периода;

C^{k} – вектор ограничений по капвложениям для подсистемы k;

b – вектор общесистемных ограничений по капвложениям.

Отметим, что создание информации по коэффициентам затрат мощностей на производство единицы продукции не представляет труда для отраслевых АСУ, в которых решаются задачи номенклатурного планирования. Так, в действующей в промышленной эксплуатации системе "Броня" для того, чтобы построить эти массивы, решается задача определения полной заявки на продукцию системы с учетом внутренней кооперации и каждой подсистеме передаются объемы внешней заявки и выпуска продуктов подсистемы, необходимых для удовлетворения данной заявки.

В принятых обозначениях задача построения общесистемного плана с учетом кооперации подсистем может быть сформулирована как задача линейного программирования следующего вида:

определить векторы $X=(X^1,\ldots,X^l)$, $Y=(Y^1,\ldots,Y^l)$ и число z при условиях:

$$\begin{gathered}
X^k \geq 0, \quad Y^k \geq 0, \quad k=\overline{1,l}; \\
\sum_{j=\overline{1,l}} F^{kj} X^j - Y^k \leq N^k; \\
X^k - d^k z = 0; \\
\sum_{k=\overline{1,l}} R^k Y^k \leq b; \\
z \to max!
\end{gathered} \tag{2}$$

Общесистемная задача (2) приведена, с одной стороны, для иллюстрации задачи построения плана кооперации подсистем как кооперации мощностей и, с другой – как идеальная задача, решения которой требуется достичь в излагаемом ниже процессе композиции решений подсистем.

Полагаем, что план каждой подсистемы k, $k=\overline{1,l}$, определяется в результате решения следующей задачи линейного программирования:

определить векторы X^k, Y^k и число z^k при условиях:

$$
\begin{aligned}
&F^{kk}X^k + z^k \sum_{\substack{j=\overline{1,l}\\ j\neq k}} F^{kj}d^j - Y^k \le N^k;\\
&X^k - d^k z^k = 0;\\
&R^k Y^k \le c^k;\\
&z^k \to max
\end{aligned}
$$

или, учитывая, что $X^k = d^k z^k$, задачу подсистем можно записать в следующем виде:

$$
\begin{aligned}
&Y^k \ge 0;\\
&z^k \sum_{j=\overline{1,l}} F^{kj}d^j - Y^k \le N^k;\\
&R^k Y^k \le c^k;\\
&z^k \longrightarrow max!
\end{aligned}
\tag{3}
$$

При этом для системы в целом должны выполняться условия:

$$
\begin{aligned}
&\sum_{k=\overline{1,l}} c^k \le b;\\
&max \min_k z^k.
\end{aligned}
$$

Другими словами, требуется найти такое распределение общесистемных ресурсов (капвложений) между подсистемами, чтобы по системе в целом с учетом кооперации подсистем достигался бы максимально возможный общесистемный уровень реализации заявки. Отметим, что здесь не учитывается влияние локальных ограничений подсистем, не связанных с перераспределяемыми общесистемными ресурсами. Учет данных условий приводит к реализации лексикографического максимума относительно первоначально заданного вектора заявки. Подробно этот вопрос при построении равноэффективных планов подсистем рассмотрен в [I].

2. Описание процесса

2.I. Начальный шаг процесса ($\gamma = 0$). Определяется объем ресурсов, необходимый для всех подсистем, для полного выполнения заявки при $z = z^1 = \ldots = z^\ell = 1{,}0$.

Для этого решаются ℓ задач линейного программирования следующего вида:

определить вектор $Y^k(0)$ при

$$\begin{gathered} Y^k(0) \geq 0; \\ \sum_{j=\overline{1,\ell}} F^{kj} d^j - Y^k(0) \leq N^k; \\ \sum_i [R^k Y^k(0)] \longrightarrow \min! \end{gathered} \tag{4}$$

По результатам решения задач (4) получаем потребность общесистемных ресурсов, необходимых системе в целом для полного удовлетворения заявки:

$$\bar{b} = \sum_{k=\overline{1,\ell}} R^k \bar{Y}^k(0),$$

где $\bar{Y}^k(0)$ определено из решения задачи (4).

2.2. $\gamma = 1$. Определяются базовые возможности подсистем по удовлетворению заявки, т.е. решаются задачи подсистем (3) при $c^k = 0$, $k = \overline{1,\ell}$, которые можно сформулировать следующим образом:

определить число $z^k(1)$ при условии

$$\begin{gathered} z^k(1) \sum_{j=\overline{1,\ell}} F^{kj} d^j \leq N^k; \\ z^k(1) \longrightarrow \max. \end{gathered} \tag{5}$$

При решении задач (5) получаются значения $z^k(1)$ (уровень удовлетворения заявки), достигаемые на базовых мощностях, т.е. когда лимитирующим фактором являются локальные условия подсистем.

2.3. $\gamma = 2,3,\dots$

2.3.I. Определяется $\min_k z^k(\gamma-1) = z^m(\gamma-1)$ – общесистемный допустимый уровень реализации заявки, достигнутый на шаге $\gamma - 1$.

2.3.2. Вычисляется объем общесистемных ресурсов, необходимый для всех подсистем для выполнения заявки на уровне $z^m(\gamma-1)$:

$$\bar{c}^k(\gamma-1) = R^k Y^k(\gamma-1)\,\frac{z^m(\gamma-1)-z^m(\gamma-2)}{z^k(\gamma-1)-z^k(\gamma-2)},\ k=\overline{1,l}. \quad (6)$$

2.3.3. Определяются объемы избыточных ресурсов, которые подлежат перераспределению на шаге γ :

$$\Delta b(\gamma) = b - \sum_{k=\overline{1,l}} \bar{c}^k(\gamma-1). \quad (7)$$

2.3.4. Определяются "оценки" $\rho^k(\gamma)$ изменения $z^k(\gamma)$ от $c^k(\gamma)$:

$$\rho^k(\gamma) = \frac{z^k(\gamma-1)-z^k(\gamma-2)}{c^k(\gamma-1)-c^k(\gamma-2)},\ k=\overline{1,l}. \quad (8)$$

Данные усредненные оценки изменения уровней реализации заявки от размера дополнительно выделенного ресурса тем меньше отличаются от двойственных оценок соответствующей задачи линейного программирования, чем меньше разброс значений $z^k(\gamma)$ и $z^k(\gamma-1)$ для всех k по решению задач подсистем. В случае реализации частично-целочисленных задач для определения планов подсистем данные усредненные оценки, естественно, также могут быть определены и использованы в процессе.

2.3.5. Определяется распределение "избыточных" ресурсов для всех подсистем для достижения на следующем шаге процесса максимально возможного и пропорционального роста подсистем относительно общего уровня реализации заявки:

$$\Delta c^k(\gamma) = \frac{\Delta b(\gamma)}{\rho^k(\gamma)\sum_{k=\overline{1,l}}(1/\rho^k(\gamma))},\ k=\overline{1,l}. \quad (9)$$

2.3.6. Определяется размер ресурсов, выделяемых подсистеме на следующем шаге:

$$c^k(\gamma)=\bar{c}^k(\gamma-1)+\Delta c^k(\gamma),\ \ k=\overline{1,\ell}. \qquad (10)$$

2.3.7. Решаются ℓ задач типа (3) при ограничениях $c^k = c^k(\gamma)$.

2.3.8. Проверяется процесс на завершенность:
процесс завершен, если

$$\max_k z^k(\gamma) - \min_k z^k(\gamma) \le \varepsilon \quad , \text{ где } \varepsilon \ge 0;$$

в противном случае процесс следует продолжить, проведя перераспределение ресурсов по формулам (6)-(10).

Очевидно, что данный процесс согласования локальных планов подсистем с учетом внутрисистемной кооперации при описании ее в виде кооперации мощностей сводится к процессу построения совокупности оптимальных равноэффективных планов подсистем при распределении общесистемных ресурсов, который, как показано в работах [1,2] , приводит к оптимальному решению общесистемной задачи (2).

3. Алгоритм координации локальных решений для задачи, учитывающей один общесистемный ресурс

3.1. Пусть матрицы R^k являются вектор-строками, т.е. в задаче (2) рассматривается только один общесистемный ресурс. В этом случае для задачи можно предложить простой алгоритм решения, учитывающий динамику в развитии мощностей и локальные ограничения на них. Заметим, что алгоритм перераспределения, описанный выше, также может учитывать эти факторы.

Для того чтобы уточнить формулировку задачи (2), введем дополнительные обозначения:

$N^k(0)$ – вектор ограничений базовых мощностей, имеющихся в подсистеме k на начало первого года планового периода;

S_t^k – вектор ограничений на прирост мощностей в год для подсистемы k .

Всем введенным выше векторам и матрицам приписывается дополнительный индекс t – номер года, в котором они определены, $t=\overline{1,T}$, где t – число лет планируемого периода. Тогда задачу (2) можно формулировать следующим образом:

определить векторы Y_t^k и величину z такую, что:

$$Y_t^k \geqslant 0; \quad k=\overline{1,l};$$

$$z\sum_{j=\overline{1,l}} F^{kj} d_t^j - \sum_{\tau=\overline{1,t}} Y_t^k \leqslant N_0^k; \qquad \text{(II)}$$

$$\sum_{t=\overline{1,T}} \sum_{k=\overline{1,k}} R_t^k Y_t^k \leqslant b;$$

$$Y_t^k \leqslant S_t^k$$

$$z \longrightarrow max!$$

3.2. Начальным (нулевым) шагом алгоритма назовем решение для каждой подсистемы следующей задачи линейного программирования: определить z^k такое, что:

$$z^k \sum_{j=\overline{1,l}} F_t^{kj} d_t^j \leqslant N_0^k, \quad t=\overline{1,T};$$

$$z^k \longrightarrow max!$$

3.3. Обозначим через $z^k(p)$ значение функционала k-й подсистемы на p-м шаге алгоритма, K - множество номеров подсистем.

Определим $z_{min} = \min\limits_k z^k(0)$, $z_{max} = \max\limits_k z^k(0)$, $K = \emptyset$.

3.4. Очередной p-й шаг работы алгоритма начинается сравнением величин z_{min} и z_{max}.

3.4.I. Если $z_{max} - z_{min} \leqslant \varepsilon$, то для завершения алгоритма переходим к п.3.5, иначе - к п.3.4.2.

3.4.2. Полагаем

$$\xi = \frac{z_{min} + z_{max}}{2}$$

и

$$K = \{k : z^k(0) \leqslant \xi\}.$$

3.4.3. Для всех $k \in K$ решаем задачи линейного программирования: определить векторы Y_t^k такие, что

$$\xi \sum_{j=\overline{1,l}} F_t^{kj} d_t^j - \sum_{\tau=\overline{1,t}} Y_\tau^k \leq N_0^k;$$

$$Y_t^k \leq S_t^k, \quad Y_t^k \geq 0; \qquad (12)$$

$$f^k(Y^k) = \sum_{t=\overline{1,T}} R_t^k Y_t^k \longrightarrow min.$$

Пусть $\overset{*}{Y}{}^k$ - решение задачи (I2).

3.4.4. Если

$$\sum_{k\in K} f^k(\overset{*}{Y}{}^k) \leq b, \qquad (13)$$

то полагаем $z_{min} = \xi$, иначе полагаем $z_{max} = \xi$ и переходим к шагу 3.4.I с новыми z_{min} и z_{max}.

3.5. Полагаем общесистемный уровень удовлетворения заявки $\bar{z}$ равным z_{min}, $\bar{X}_t^k = z_{min} \cdot d_t^k$, для подсистем, номера которых принадлежат K, $\bar{Y}^k = \overset{*}{Y}{}_t^k$, а для остальных подсистем $\bar{Y}_t^k = 0$.

УТВЕРЖДЕНИЕ. План $\{\bar{z}, \bar{Y}_t^k\}$ является оптимальным решением задачи (II).

Действительно, в силу (I2) и (I3) $\{\bar{z}, \bar{Y}_t^k\}$ удовлетворяют ограничениям задачи (II).

Из описания алгоритма пп.3.2-3.5 следует, что не существует величины $\bar{\xi} > \bar{z}_{min} + \varepsilon$ такой, что $\sum_{k\in K} f^k(\bar{Y}_t^k) \leq b$, где $\bar{Y}_t^k$ - решение задачи (I2) при $\xi = \bar{\xi}$.

Пусть $\{z, Y_t^k\}$ - план, допустимый для ограничений задачи (II) и пусть $z > \bar{z} + \varepsilon$. Тогда план $\{Y_t^k\}$ является допустимым для задачи (I2) при $\xi = z$, $\sum_{k\in K} f^k(Y_t^k) \leq$ $\leq b$, что противоречит предыдущему утверждению. Следовательно, план $\{\bar{z}, Y_t^k\}$ - оптимальный для задачи (II).

В построении алгоритма, описанного в п.3, используется схема отыскания экстремума функции методом дихотомии. Число итераций этого алгоритма не превосходит $\log_2((z_{max}(0) - z_{min}(0))/\varepsilon)$.

В заключение сделаем несколько замечаний о практической реализации предлагаемых алгоритмов.

4. Практическая реализация алгоритмов.

4.I. Основным преимуществом предлагаемой в работе модели кооперации мощностей является сокращение числа переменных в задаче согласования планов подсистем по сравнению с моделями, в которых кооперация учитывается ввиде затрат изделия на изделие [I,3]. В задачах (2) и (I2) размерность векторов совпадает с числом изделий, конечных для экономической системы в целом. В моделях, учитывающих затраты изделия на изделие, необходимо рассматривать все продукты, конечные для подсистем, т.е. в 2-4 раза большую номенклатуру изделий.

4.2. Задачи подсистем (2) и (I2) являются задачами линейного программирования малой размерности и простой структуры. Число переменных Y_t^k в них совпадает с числом учитываемых видов мощностей, т.е. несколько десятков.

4.3. Задачи подсистем (3) совместны на всех итерациях процесса композиции локальных решений. Задачи подсистем (I2) могут оказаться несовместными из-за наличия локальных ограничений. В этом случае предлагается следующая модификация алгоритма, описанного в п.3.

Начальный шаг (3.2) дополняется решением для каждой подсистемы пары задач линейного программирования следующего вида:

определить величину ξ^k и вектор Y_t^k такие, что

$$\xi^k \sum_{j=\overline{1,l}} F_t^{kj} d^j - \sum_{\tau=\overline{1,t}} Y_\tau^k \le N_0^k;$$

$$Y_t^k \le S_t^k, \quad Y_t^k \ge 0; \qquad (\text{T4})$$

$$\xi^k \to max.$$

и задачу (I2) при $\xi = \overline{\xi}^k$, где $\overline{\xi}^k$ берется из решения (I4). Пусть z^k - значение функционала последней задачи. Очевидно, что последняя пара задач совместна для всех $k=\overline{1,l}$.

Очередной p-й шаг алгоритма начинается сравнением величин z_{min} и z_{max} (см.3.4.I); п.3.4.2 изменяется следующим образом.

Полагаем

$$\xi = \frac{z_{min} + z_{max}}{2};$$

$$K = \{k : z^k(p-1) \le \xi, \quad \xi^k > \xi\};$$

$$K_1 = \{k : \bar{\xi}^k < \xi\}.$$

Далее выполняем п.3.4.3, а п.3.4.4 модифицируем. Если $\sum_{k \in K} f^k(\overset{*}{Y}{}_t^k) + \sum_{k \in K_1} z^k \le b$, то полагаем $z_{min} = \xi$, иначе $z_{max} = \xi$, и переходим к шагу 3.4.I.Задачи вида (I2) в этой модификации алгоритма всегда совместны.

Авторы считают, что с помощью предлагаемых алгоритмов можно проводить процесс построения плана подсистем, состоящих из нескольких сотен подсистем (т.е. для систем - министерства, подсистем - предприятия), в том числе на современных микрокомпьютерах.

ЛИТЕРАТУРА

I. Макаров В.Л., Маршак В.Д. Модели оптимального функционирования отраслевых систем. - М.: Экономика, 1979.
2. Булавский В.А., Маршак В.Д. Алгоритм построения системы оптимальных равноэффективных планов при распределении нескольких ресурсов // Оптимизация. - 1983. - Вып.3I. - С.156-165.
3. Шафранский В.В. Математические модели и методы планирования развития отраслей промышленности. - М.: Наука, 1984.

Поступила в ред.-изд. отдел
09.06.1987 г.