

УДК 623.681.5

ДИСКРЕТНАЯ МНОГОМЕРНАЯ ЗАДАЧА СТАНДАРТИЗАЦИИ ПРИ ДЕТЕРМИНИРОВАННОМ СПРОСЕ И ЕЕ РЕШЕНИЕ МЕТОДОМ ВЕТВЕЙ И ГРАНИЦ

Канд. воен. наук СИНЯВСКИЙ В. К.

Научно-исследовательский институт Вооруженных Сил Республики Беларусь

Большой класс задач выбора оптимальных параметрических рядов сводится к многомерной задаче оптимизации [1]. Общая постановка таких задач может быть записана следующим образом. Задано множество $J = \{1, 2, \dots, N\}$ видов потребностей, которые необходимо выполнить. Каждый вид потребности определяется совокупностью характеристик (параметров). Имеется также множество $I = \{1, 2, \dots, M\}$ изделий, способных совместно обеспечить все виды потребностей. Каждому i -му изделию также соответствует набор значений параметров, характеризующих его эксплуатационные, технические, эргономические и другие свойства. Так как каждый набор значений параметров соответствует изделию с определенным номером i , будем считать, что тип изделия полностью определяется своим порядковым номером. Необходимо из множества I допустимых типов изделий выбрать такое подмножество $I^* \subset I$ (оптимальный ряд изделий), которое полностью обеспечит множество J видов потребностей с минимальными затратами. Требуется определить

$$C = \min \left(\sum_{i=1}^M c_i^0 y_i + \sum_{i=1}^M \sum_{j=1}^N c_{ij} x_{ij} \right) \quad (1)$$

при ограничениях:

$$\sum_{i=1}^M \alpha_{ij} x_{ij} = 1, \quad j = 1, 2, \dots, N; \quad (2)$$

$$x_{ij} \leq y_i, \quad i = 1, 2, \dots, M; \quad j = 1, 2, \dots, N; \quad (3)$$

$$x_{ij} \in \{0,1\}, \quad y_i \in \{0,1\}, \quad i = 1, 2, \dots, M; \quad j = 1, 2, \dots, N; \quad (4)$$

$$\sum_{i=1}^M y_i \leq M_0, \quad M_0 \leq M, \quad (5)$$

где c_i^0 – постоянные затраты на разработку и применение изделий i -го типа; $c_{ij} = \alpha_{ij} c_i b_j$ – затраты, связанные с обеспечением потребностей j -го вида изделиями i -го типа; α_{ij} – число изделий i -го типа, необходимых для обеспечения единицы потребностей j -го вида; c_{ij} – затраты на производство и эксплуатацию одного изделия i -го типа; b_j – величина потребностей j -го вида; M_0 – количество типов изделий, которые могут быть включены в оптимальный ряд I^* .

Взаимосвязь между элементами множеств I и J задается с помощью матрицы применений $\|\alpha_{ij}\|_{M \times N}$, в которой $\alpha_{ij} = 1$, если изделие i -го типа может обеспечивать j -й вид потребности, и $\alpha_{ij} = 0$ – в противном случае. Полагаем, что $c_{ij} = \infty$, если $\alpha_{ij} = 0$. Управляющие переменные:

$$y_i = \begin{cases} 1, & \text{если } i\text{-й тип изделия включают} \\ & \text{в выбираемый ряд изделий;} \\ 0 & \text{– в противном случае;} \end{cases}$$

$$x_{ij} = \begin{cases} 1, & \text{если } i\text{-й тип изделий назначен} \\ & \text{для удовлетворения потребности } j\text{-го вида;} \\ 0 & \text{– в противном случае.} \end{cases}$$

Ограничения (2) обозначают, что все потребности j -го вида должны быть обеспечены. Из (3) следует, что для обеспечения потребностей j -го вида можно назначать изделия только тех типов, для которых $y_i = 1$.

Рассмотрим решение задачи (1)–(4) методом ветвей и границ. Построение дерева возможных вариантов осуществляется в соответствии с односторонней схемой ветвления [2]. Непосредственное ветвление выбранной вершины производится в отличие от традиционных способов с помощью специальных показателей. Для оценки нижней границы решения в процессе ветвления используется задача линейного программирования, двойственная по отношению к (1)–(4), в которой условие целочисленности (4) заменяется условием

$$0 \leq y_i \leq 1, \quad i = 1, 2, \dots, M. \quad (6)$$

Проверка допустимого решения задачи (1)–(4) на оптимальность осуществляется решением оценочной двойственной задачи. Двойственная задача линейного программирования по отношению к релаксированной задаче (1)–(3), (6)

$$Z = \max \sum_{j \in J} V_j \quad (7)$$

при ограничениях

$$\sum_{j \in J} W_{ij} \leq c_i^0, \quad i = 1, 2, \dots, M; \quad (8)$$

$$V_j - W_{ij} \leq c_{ij} \quad j = 1, 2, \dots, N; \quad i = 1, 2, \dots, M; \quad (9)$$

$$W_{ij} \geq 0, \quad j = 1, 2, \dots, N; \quad i = 1, 2, \dots, M, \quad (10)$$

где V_j, W_{ij} – переменные двойственной задачи.

Введя дополнительные переменные γ_{ij}, δ_i ($i = \overline{1, M}, j = \overline{1, N}$), задачу (7)–(10) можно представить в виде основной задачи линейного программирования, исходное базисное решение которой имеет вид:

$$Z = \sum_{j=1}^N V_j;$$

$$\gamma_{ij} = c_{ij} + W_{ij} - V_j \quad i = 1, 2, \dots, M, j = 1, 2, \dots, N;$$

$$\delta_i = c_i^0 - \sum_{j=1}^N W_{ij} \quad i = 1, 2, \dots, M.$$

Полученную задачу (7)–(10) можно решить симплексным методом. Однако практика показала, что даже небольшие задачи унификации требуют решения задач линейного программирования большой размерности. Их опорные планы сильно вырождены, а установление оп-

тимальности в некоторых случаях связано с большим числом итераций. Действительно, количество переменных k_1 и число уравнений ограничений k_2 задачи линейного программирования связаны с размерностью исходной задачи соотношениями: $k_1 = M + 2MN + N$; $k_2 = M(N + 1)$. Для решения задачи линейного программирования такой размерности требуется обычно от $M(N + 1)$ до $3M(N + 1)$ итераций. В силу того, что оценка нижней границы производится многократно, такой способ решения является нерациональным. Нецелесообразно искать и точное решение задачи (7)–(10), так как можно получить точное решение не исходной целочисленной задачи (1)–(4), а лишь непрерывной релаксированной задачи (1)–(3), (6). Поэтому дальнейшее развитие рассматриваемого направления идет по пути построения приближенных методов решения двойственной задачи и упрощения способов оценки нижней границы исходной целочисленной задачи [3]. Основное отличие алгоритмов, использующих для оценки двойственную задачу, заключается в способе определения приближенного решения задачи (7)–(10).

Ее простая структура позволяет получить приближенное решение, достаточно близкое к оптимальному. Вычислительные на ЭВМ эксперименты показали, что особенностью задачи (1)–(4) является частое совпадение оптимального целочисленного решения с непрерывным. Поэтому обычно приближенно решают двойственную задачу, а от нее переходят к соответствующему решению непрерывной задачи (1)–(3), (6). При совпадении целочисленного и непрерывного решений полученный результат является оптимальным, в противном случае переходят к обычному алгоритму ветвей и границ. Однако переход от приближенного решения двойственной задачи к соответствующему приближенному решению задачи (1)–(4) сложный и не всегда однозначный. Более простым является способ, когда сначала определяется достаточно точное решение (1)–(4), а по нему – решение двойственной задачи. Совпадение решений свидетельствует об оптимальности целочисленного решения. Рассмотрим этот способ решения двойственной задачи (7)–(10).

Упростим ее структуру, для чего исключим переменные W_{ij} , входящие лишь в ограничения.

При любом допустимом наборе переменных V_j переменные W_{ij} можно положить равными максимальной из величин 0 и $V_j - C_{ij}$. С учетом этого двойственную задачу (7)–(10) запишем в виде

$$Z = \max \sum_{j=1}^N V_j \quad (11)$$

при ограничениях

$$\sum_{j=1}^N \max(V_j - c_{ij}; 0) \leq c_i^0, \quad i = 1, 2, \dots, M. \quad (12)$$

Сущность направленного решения двойственной задачи (11), (12) заключается в следующем. Учитывая ее простую структуру и возможность использования приближенного решения для оценки нижней границы на различных этапах ветвления, предлагается следующий способ формирования значений переменных V_j , доставляющих целевой функции (11) по возможности большее значение. Процесс начинается с допустимого решения $V_j = \min_{i \in I_1, I_3} c_{ij}$ задачи (11), (12) и состоит в многократном последовательном просмотре j -х столбцов ($j = 1, 2, \dots, N$) матрицы $\|c_{ij}\|$ с целью проверки возможности увеличения V_j до ближайшей большей величины c_{ij} . В каждом цикле просмотра проверяется возможность увеличения V_j для тех столбцов, в которых находится наименьшее количество элементов c_{is} , удовлетворяющих условию $c_{ij} \leq V_j$. Увеличение значений переменных V_j происходит с учетом (12) и ограничено величинами

$$V_j \leq \min[\min_{i \in I_1} c_{ij}; \min_{i \in I} (c_{ij} + c_i^0)]. \quad (13)$$

Оценка, получаемая с помощью данного способа, определяется элементами множеств I_1 , I_3 и, когда дальнейшее увеличение переменных V становится невозможным, вычисляется по формуле

$$Z(I_1) = z_x = \sum_{j=1}^N V_j + \sum_{i \in I_1} c_i^0. \quad (14)$$

Особенностью предлагаемого алгоритма является то, что решение двойственной задачи используется не только для оценки нижней границы решения задачи (1)–(4) в ходе ветвления, но главным образом для проверки оптимальности ее

допустимых решений, получаемых в процессе ветвления. Способ, предлагаемый для проверки оптимальности, основан на том, что при решении двойственной непрерывной задачи используется найденное целочисленное решение задачи (1)–(4). Эффективность применения способа и точность полученного с его помощью результата в значительной степени определяются точностью используемого целочисленного решения. Если для оценки нижней границы целевой функции путем направленного решения двойственной задачи (11), (12) используются ограничения (13), то при проверке допустимого решения на оптимальность вместо (13) требуется выполнение неравенств:

$$c_{rj} \leq V_j \leq \min[c_{ij}; \min_{i \in I} (c_{ij} + c_i^0)]; \quad (15)$$

$$j = 1, 2, \dots, N,$$

где

$$c_{ri} = \min_{i \in I_D} c_{ij}; \quad c_{ij} = \min_{\substack{i \in I \\ i \neq j}} c_{ij}.$$

Процесс проверки допустимого решения задачи (1)–(4) на оптимальность начинается с допустимого решения $V_i = \min_{i \in I} c_{ij}$ задачи (11), (12).

Необходимость неравенств (15) следует из соотношений:

$$y_i^* [c_i^0 - \sum_{j=1}^N \max(V_j - c_{ij}; 0)] = 0;$$

$$(y_i^* - x_j^*) \max(V_j - c_{ij}; 0) = 0,$$

определяющих оптимальность полученных решений для непрерывной и целочисленной задач. Знак «*» указывает на оптимальность решений соответствующих задач. При выполнении ограничения (15) совпадение величины

$$Z(I_D) = \sum_{j=1}^N V_j \quad (16)$$

полученной в результате решения двойственной задачи (11), (12), со значением целевой функции задачи (1)–(4), соответствующей проверяемому целочисленному решению

$$C_D = \sum_{i \in I_D} c_i^0 + \sum_{j=1}^N \min_{i \in I_D} c_{ij}, \quad I_D = I, \quad (17)$$

указывает на оптимальность найденного це-

лочисленного решения, т. е. $I^* = I_d$; $C^* = C_d$. В случае несовпадения значений целевых функций поиск оптимального решения задачи (1)–(4) продолжается методом ветвей и границ. Поиск первого допустимого решения и дальнейшее ветвление осуществляется с помощью показателей вида

$$\rho_i = \sum_{j=1}^N \max(V_j - c_{ij}; 0) - c_i^0, \quad i \in I_3,$$

где

$$V_j = \min \left[\min_{i \in I_1} c_{ij}; \min_{i \in I} (c_i^0 + c_{ij}) \right]. \quad (18)$$

Таким образом, общая схема решения задачи (1)–(4) включает:

1) определение допустимого решения I_d , C_d исходной задачи;

2) проверку допустимого решения на оптимальность путем решения двойственной задачи;

3) уточнение допустимого решения методом ветвей и границ и проверку его на оптимальность.

Вычислительный процесс заканчивается при оптимальности допустимого решения с помощью решения двойственной задачи или в случае анализа всех перспективных ветвей дерева вариантов.

ВЫВОД

Большинство многомерных задач стандартизации, как правило, не удовлетворяют свойству связности, и следовательно, метод динамического программирования не может быть использован для их решения. Наиболее перспективным методом решения таких задач является метод ветвей и границ, достоинства которого заключаются в возможности учета их особенностей в ходе вычисления оценок в процессе ветвления, в том числе и наличия целочисленных переменных. Сформулированная задача многомерной оптимизации может найти широкое применение в области стандартизации и унификации промышленной продукции различного назначения при ее разработке и проектировании.

ЛИТЕРАТУРА

1. Берсенеv, В. Л. Экстремальные задачи стандартизации / В. Л. Берсенеv, Э. X. Гимади, В. Г. Дементьев. – Новосибирск: Наука, 1978. – 248 с.
2. Анисимов, В. Г. Метод решения одного класса задач целочисленного программирования / В. Г. Анисимов, Е. Г. Анисимов // Журнал вычислительной математики и математической физики. – 1984. – № 10. – С. 1565–1570.
3. Анисимов, В. Г. Формальная структура задач стандартизации и унификации при управлении развитием сложных технических систем / В. Г. Анисимов, Е. Г. Анисимов // Защита и безопасность. – 2004. – № 4. – С. 26–31.

Поступила 10.02.2005

УДК 351.746.1+623.4

ПЕРИОДЫ РАЗВИТИЯ, ВНЕДРЕНИЯ И ПРИМЕНЕНИЯ ТЕХНИЧЕСКИХ СРЕДСТВ И СРЕДСТВ АВТОМАТИЗАЦИИ ПОГРАНИЧНОГО КОНТРОЛЯ В ПОГРАНИЧНЫХ ВОЙСКАХ РЕСПУБЛИКИ БЕЛАРУСЬ

КАМЗАЛОВ С. Н.

Брестская пограничная группа

Применение технических средств пограничного контроля (ТСПК) в служебной деятельности подразделений пограничного контроля осуществляется более 50 лет. При классификации периодов развития ТСПК необходимо учитывать исторические эпохи, цели и зада-

чи, стоящие перед пограничными войсками, а также содержание руководящих документов, отражающих развитие научной мысли и достижения практики на момент их создания, объективные и субъективные факторы, которые влияли на происходившие изменения.