Е. В. Водолазский, Б. Флах, М. И. Шлезингер, Минимаксные задачи дискретной оптимизации, инвариантные относительно мажоритарных операторов, *Ж. вычисл. матем. и матем. физ.*, 2014, том 54, номер 8, 1368–1378

ЖУРНАЛ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ И МАТЕМАТИЧЕСКОЙ ФИЗИКИ, 2014, том 54, № 8, с. 1368–1378

УДК 519.218.43

МИНИМАКСНЫЕ ЗАДАЧИ ДИСКРЕТНОЙ ОПТИМИЗАЦИИ, ИНВАРИАНТНЫЕ ОТНОСИТЕЛЬНО МАЖОРИТАРНЫХ ОПЕРАТОРОВ[1)]

© 2014 г. Е. В. Водолазский*, Б. Флах, М. И. Шлезингер***

*(*Украина, 03680 ГСП Киев, пр-т Акад. Глушкова, 40, Междунар. научно-учебный центр ИТС НАН Украины;*
***Česká republika, 16636 Prague 6, Zikova4, Чешский технич. ун-т в Праге)*
e-mail: waterlaz@gmail.com, flachbor@cmp.felk.cvut.cz, schles@irtc.org.ua

Поступила в редакцию 21.10.2013 г.
Переработанный вариант 04.02.2014 г.

Исследован специальный класс задач дискретной оптимизации, который сформулирован как минимаксная модификация проблемы непротиворечивости ограничений, известной в английской терминологии как Constraint satisfaction problem. Минимаксная формулировка задачи обобщает классическую задачу на реалистические ситуации, когда ограничения определяют не дихотомию множества на допустимое и недопустимое подмножества, а упорядочивают элементы множества по степени их допустимости. Определено понятие инвариантности этого упорядочивания относительно того или иного оператора и доказана полиномиальная сложность дискретной минимизации функций, инвариантных относительно мажоритарных операторов. Приводится конкретный алгоритм этой минимизации. Библ. 6.

Ключевые слова: задача дискретной оптимизации, минимаксная модификация, алгоритм решения.

DOI: 10.7868/S004446691408016X

1. ВВЕДЕНИЕ

Исследуется специальный класс задач дискретной оптимизации, сформулированный в виде минимаксной модификации классической проблемы удовлетворения ограничений (см. [1]–[6]) (в английской терминологии – Constraint satisfaction problem). Минимаксная формулировка задачи обобщает классическую задачу на такие реалистичные ситуации, когда ограничения определяют не дихотомию множества решений на допустимые и недопустимые, а упорядочивают это множество по степени их допустимости.

Множество всех возможных задач удовлетворения ограничений образует NP-полный класс. Известен, однако, обширный полиномиально разрешимый подкласс этих задач, основанный на инвариантности ограничений относительно операторов определенного вида. В статье сформулирован алгоритм полиномиальной сложности для решения аналогичного подкласса минимаксных задач. Помимо этого своего прямого назначения алгоритм вносит дополнительную ясность в классические задачи удовлетворения ограничений, делает их более понятными, прозрачными. Для понимания основного результата статьи не требуются никакие априорные знания за пределами статьи, потому что все, что нужно для его доказательства, содержится в самой статье.

2. ОПРЕДЕЛЕНИЕ ОСНОВНЫХ ПОНЯТИЙ

Пусть T и K – конечные множества, называемые множеством объектов и множеством меток. Функцию вида $\bar{k} : T \to K$ назовем *разметкой множества T метками из множества K*, или просто разметкой. Для заданной разметки $\bar{k}$ обозначим через $k(t)$ ее значение для объекта $t \in T$, через $k(T') : T' \to K$ – сужение на подмножество $T' \subset T$. В случае, когда надо подчеркнуть, что T' есть

1) Работа выполнена по заданию Бюро отделения информатики НАН Украины (№ госрегистрации 0111U002091) и при поддержке Грантового Агентства Чешской Республики (грант P202/12/2071).

объединение непересекающихся подмножеств T'_1 и T'_2, сужение $k(T'_1 \cup T'_2)$ будем обозначать в виде пары $(k(T'_1), k(T'_2))$. Обозначим через K^T множество всех возможных функций формата $T \to K$ и через 2^T – множество всех возможных подмножеств $T' \subset T$.

Определение 1. Структура Γ множества T – это подмножество множества 2^T; порядок структуры Γ – это число $\max_{T' \in \Gamma}|T'|$.

Пусть W – полностью упорядоченное множество и для каждого $T' \in \Gamma$ определена функция $g_{T'} : K^{T'} \to W$.

Определение 2. Исходные данные минимаксной задачи разметки, или просто задача – это пятерка

$$z = \langle T, K, W, \Gamma \subset 2^T, (g_{T'} : K^{T'} \to W | T' \in \Gamma)\rangle; \tag{1}$$

порядок задачи z – это порядок структуры Γ.

Функции $g_{T'} : K^{T'} \to W$, $T' \in \Gamma$, в исходных данных задачи представлены в виде таблиц $\mathrm{Tab}(T') = \{(\bar{k}, g_{T'}(\bar{k}))|\bar{k} \in K^{T'}\}$. Величину $\Sigma_{T' \in \Gamma}|\mathrm{Tab}(T')|$ назовем *объемом исходных данных*.

Для каждой разметки $\bar{k} \in K^T$ исходные данные (1) определяют число $g(\bar{k}) = \max_{T' \in \Gamma} g_{T'}(k(T'))$, называемое *качеством разметки*. Указанную функцию $g : K^T \to W$ назовем *целевой функцией задачи*. Порядок функции $g : K^T \to W$ – это порядок задачи, для которой эта функция целевая.

Определение 3. Решением задачи z является величина $\min_{\bar{k} \in K^T} g(\bar{k})$ и разметка $\bar{k}^*$, обозначаемая $\arg\min_{\bar{k} \in K^T} g(\bar{k})$, такая, что $g(\bar{k}^*) = \min_{\bar{k} \in K^T} g(\bar{k})$.

Пример 1. В формате (1) можно представить задачу удовлетворения ограничений, известную в следующей формулировке. Пусть T – конечное множество переменных, принимающих значения из конечного множества K, а $\bar{k}$ – отображение $T \to K$. На отображение $\bar{k}$ наложена система ограничений, каждое из которых, скажем, i-е ограничение, $i = 1, \dots, n$, задано в виде пары (T_i, K_i), $T_i \subset T$, $K_i \subset K^{T_i}$. Задача удовлетворения ограничений заключается в том, что для заданных множеств T, K и системы ограничений (T_i, K_i), $i = 1, \dots, n$, следует определить, существует ли отображение $\bar{k}^*$ такое, что $k^*(T_i) \in K_i$ для любого $i = 1, \dots, n$. Эта задача выражается в формате (1) минимаксной задачи разметки при $W = \{0, 1\}$, $\Gamma = \{T_i|i = 1, \dots, n\}$ и функциях $g_{T_i} : K^{T_i} \to W$, $T_i \in \Gamma$, принимающих значение $g_{T_i}(k(T_i)) = 0$ для $k(T_i) \in K_i$ и значение $g_{T_i}(k(T_i)) = 1$ для $k(T_i) \notin K_i$.

Пример 2. Задача удовлетворения ограничений, рассмотренная в примере 1, допускает естественное обобщение на случай так называемых размытых (нечетких) ограничений. Размытые ограничения представлены не в виде (T_i, K_i), где $K_i \subset K^{T_i}$, а в виде (T_i, q_i), где $q_i : K^{T_i} \to \{q|0 \le q \le 1\}$ – функция принадлежности определенному размытому множеству. Система (T_i, q_i), $i = 1, \dots, n$, размытых ограничений определяет не дихотомию множества отображений $\bar{k} : T \to K$ на допустимые и недопустимые, а степень допустимости $\min_i q_i(k^*(T_i))$ каждого отображения $\bar{k}^*$. Характеристикой системы (T_i, q_i), $i = 1, \dots, n$, размытых ограничений становится не ее непротиворечивость, а степень непротиворечивости, выражаемая числом $\max_{\bar{k} \in K^T} \min_i q_i(k(T_i))$. Определение степени непротиворечивости системы (T_i, q_i), $i = 1, \dots, n$, размытых ограничений сводится к решению задачи (1) при $W = \{q|0 \le q \le 1\}$, $\Gamma = \{T_i|i = 1, \dots, n\}$ и функциях $g_{T_i} : K^{T_i} \to W$, $T_i \in \Gamma$, со значениями $g_{T_i}(k(T_i)) = 1 - q_i(k(T_i))$.

Пример 3. Пусть для множества T задана функция $d : T \times T \to W$, определяющая различие $d(t, t')$ объектов $t \in T$ и $t' \in T$. Пусть пара (T_1, T_2), $T_1 \cup T_2 = T$, $T_1 \cap T_2 = \emptyset$, – разбиение множества T на два подмножества, а

$$F(T_1, T_2) = \max\{\max_{t \in T_1, t' \in T_1} d(t, t'), \max_{t \in T_2, t' \in T_2} d(t, t')\}$$

есть качество этого разбиения. Одна из возможных формулировок известной задачи кластеризации, или автоматической классификации множества, состоит в поиске наилучшего разбиения

$$(T_1^*, T_2^*) = \arg\min_{(T_1, T_2)} \max\{\max_{t \in T_1, t' \in T_1} d(t, t'), \max_{t \in T_2, t' \in T_2} d(t, t')\}. \tag{2}$$

Эту задачу можно представить в виде минимаксной задачи разметки, для которой $K = \{1, 2\}$,

$$\Gamma = \{\{t, t'\} | t \in T, t' \in T, t \neq t'\},$$

$$g_{\{t, t'\}}(k, k') = d(t, t'), \text{ если } k = k', \text{ и } g_{\{t, t'\}}(k, k') = \min W, \text{ если } k \neq k'; \ \{t, t'\} \in \Gamma.$$

Если найдено решение $\bar{k}^* = \arg\min_{\bar{k} \in K^T} \max_{\{t, t'\} \in \Gamma} g_{\{t, t'\}}(k(t), k(t'))$ этой задачи разметки, то решением задачи кластеризации (2) являются множества $T_k^* = \{t \in T | k^*(t) = k\}$, $k \in \{1, 2\}$.

Множество всех возможных минимаксных задач разметки образует NP-полный класс, так как оно содержит в себе множество всех возможных задач удовлетворения ограничений. Основным средством, с помощью которого формулируются полиномиально разрешимые подклассы задач удовлетворения ограничений, являются понятия полиморфизмов и инвариантов (раздел 8.3 в [4]). Мы сформулируем и исследуем эти понятия в более общем виде, необходимом для формулировки полиномиально разрешимого подкласса минимаксных задач.

3. ИНВАРИАНТЫ И ПОЛИМОРФИЗМЫ

Ключевым понятием данной статьи является соответствие между целевыми функциями вида $g : K^T \to W$ и функциями вида $p : K \times K \times K \to K$. Чтобы различать функции этих двух видов, функции вида $p : K \times K \times K \to K$ будем называть *операторами*.

Пусть $p : K \times K \times K \to K$ – трехместный оператор на множестве меток. Расширим его формат до вида $K^T \times K^T \times K^T \to K^T$ так, что результат действия оператора p разметки $\bar{k}_1 \in K^T$, $\bar{k}_2 \in K^T$, $\bar{k}_3 \in K^T$ – это разметка $\bar{k}^* = p(\bar{k}_1, \bar{k}_2, \bar{k}_3) \in K^T$ со значениями $k^*(t) = p(k_1(t), k_2(t), k_3(t))$, $t \in T$.

Определение 4. Функция g: $K^T \to W$ называется *инвариантом оператора* p: $K \times K \times K \to K$, а оператор p – *полиморфизмом функции* g, если для любых трех разметок $\bar{k}_1 \in K^T$, $\bar{k}_2 \in K^T$, $\bar{k}_3 \in K^T$ выполняется неравенство $g(p(\bar{k}_1, \bar{k}_2, \bar{k}_3)) \leq \max\{g(\bar{k}_1), g(\bar{k}_2), g(\bar{k}_3)\}$.

Полиморфизм – это инструмент, с помощью которого для любых трех точек x_1, x_2, x_3 из области определения некоторой функции $f : X \to Y$ можно получить четвертую точку, которая не хуже, чем худшая из предъявленных трех.

Пример 4. Усреднение $p(x_1, x_2, x_3) = (x_1 + x_2 + x_3)/3$ является полиморфизмом любой выпуклой функции. Усреднение является полиморфизмом и любой квазивыпуклой функции, т.е. такой, для которой множество $\{x \in X | f(x) \leq \theta\}$ выпукло при любом θ.

В данной статье мы рассматриваем функции на конечных областях определения, которые характеризуются полиморфизмом иного вида.

Определение 5. Оператор $p : K \times K \times K \to K$ называется *мажоритарным*, если $p(k_1, k, k) = p(k, k_2, k) = p(k, k, k_3) = k$ для любой четверки k, k_1, k_2, k_3 меток.

Пример 5. Пусть K – полностью упорядоченное множество и k_1, k_2, k_3 – некоторые три метки. Исключим метки $\max\{k_1, k_2, k_3\}$ и $\min\{k_1, k_2, k_3\}$ из этой тройки и назовем оставшуюся метку средней меткой тройки (k_1, k_2, k_3). Оператор, определяющий среднюю метку в тройке меток, является мажоритарным.

Пример 6. Пусть K – множество вершин неориентированного дерева, и пусть k_1, k_2, k_3 – некоторые три вершины. Вершину k^* назовем средней вершиной тройки (k_1, k_2, k_3), если в k^* пересекаются три пути: из k_1 в k_2, из k_2 в k_3 и из k_3 в k_1. Оператор, определяющий среднюю вершину в тройке вершин, является мажоритарным.

Пример 7. Для $|K| = 2$ любая функция $g: K \times K \to W$ двух переменных инвариантна относительно любого мажоритарного оператора.

Определение 6. Оператор $p: K \times K \times K \to K$ назовем *полиморфизмом задачи* $z = \langle T, K, W, \Gamma, (g_{T'} | T' \in \Gamma)\rangle$, а задачу z – *инвариантной относительно* p, если каждая функция $g_{T'}$, $T' \in \Gamma$, инвариантна относительно p.

Пример 8. Минимаксная задача, к которой сводится разбиение множества на два подмножества (см. пример 3), инвариантна относительно любого мажоритарного оператора, так как количество меток в этой задаче равно 2.

Мы покажем достаточно простой алгоритм решения любой минимаксной задачи, для которой существует мажоритарный полиморфизм, пусть и неизвестный. Простота этого алгоритма основана на следующих формальных свойствах полиморфизмов и инвариантов.

Пусть g' и g'' – две функции вида $K^T \to W$. Обозначим через $\max\{g', g''\}$ функцию $g^0: K^T \to W$ со значениями $g^0(k) = \max\{g'(k), g''(k)\}$, $\bar{k} \in K^T$.

Лемма 1. *Если функции g' и g'' инвариантны относительно одного и того же оператора p, то функция* $\max\{g', g''\}$ *инвариантна относительно этого оператора.*

Доказательство. Пусть $\bar{k}_1$, $\bar{k}_2$, $\bar{k}_3$ – три разметки и $\bar{k}_0 = p(\bar{k}_1, \bar{k}_2, \bar{k}_3)$. Инвариантность g' и g'' относительно p обозначает, что

$$\max\{g'(\bar{k}_1), g'(\bar{k}_2), g'(\bar{k}_3)\} \ge g'(\bar{k}_0) \quad \text{и} \quad \max\{g''(\bar{k}_1), g''(\bar{k}_2), g''(\bar{k}_2)\} \ge g''(\bar{k}_0).$$

Отсюда следует, что

$$\max\{\max\{g'(\bar{k}_1), g''(\bar{k}_1)\}, \max\{g'(\bar{k}_2), g''(\bar{k}_2)\}, \max\{g'(\bar{k}_3), g''(\bar{k}_3)\}\} \ge \max\{g'(\bar{k}_0), g''(\bar{k}_0)\},$$

т.е. что функция $\max\{g', g''\}$ инвариантна относительно p. Лемма доказана.

Пусть T – множество объектов, а T', T'' – два его подмножества, $T = T' \cup T''$, $T' \cap T'' = \emptyset$.

Определение 7. Проекция функции g: $K^T \to W$ на подмножество T' – это функция $g_{T'}: K^{T'} \to W$ со значениями $g_{T'}(k(T')) = \min\limits_{k(T'') \in K^{T''}} g(k(T'), k(T''))$.

Лемма 2. *Если функция $g: K^T \to W$ инвариантна относительно оператора p, то ее проекция $g_{T'}$ на любое подмножество $T' \subset T$ инвариантна относительно этого оператора.*

Доказательство. Положим, $T'' = T \backslash T'$. Пусть $k_1'(T')$, $k_2'(T')$, $k_3'(T')$ – некоторые три разметки множества T', а w_1, w_2, w_3 – значения функции $g_{T'}$ на этих разметках. Это значит, что существуют такие три разметки

$$\bar{k}_1 = (k_1'(T'), k_1''(T'')), \quad \bar{k}_2 = (k_2'(T'), k_2''(T'')), \quad \bar{k}_3 = (k_3'(T'), k_3''(T'')),$$

что

$$g(\bar{k}_1) = w_1, \quad g(\bar{k}_2) = w_2, \quad g(\bar{k}_3) = w_3.$$

Пусть

$$\bar{k}_0 = p(\bar{k}_1, \bar{k}_2, \bar{k}_3),$$

$$k_0'(T') = p(k_1'(T'), k_2'(T'), k_3'(T')),$$

$$k_0''(T'') = p(k_1''(T''), k_2''(T''), k_3''(T'')).$$

Поскольку g инвариантна относительно p, а $g_{T'}$ есть проекция g на T', то

$$g_{T'}(k_0'(T')) = \min_{k(T'') \in K^{T''}} g(k_0'(T'), k(T'')) \le g(k_0'(T'), k_0''(T'')) = g(\bar{k}_0) \le \max\{w_1, w_2, w_3\}.$$

Таким образом,

$$g_{T'}(p(k_1'(T'), k_2'(T'), k_3'(T'))) \le \max\{g_{T'}(k_1'(T')), g_{T'}(k_2'(T')), g_{T'}(k_3'(T'))\},$$

а это значит, что $g_{T'}$ инвариантна относительно p. Лемма доказана.

Лемма 3. *Пусть для функции $g : K^T \to W$ существует мажоритарный полиморфизм*; T_1, T_2, T_3 — *попарно не пересекающиеся подмножества в T такие, что $T_1 \cup T_2 \cup T_3 = T$*; g_{12}, g_{13}, g_{23} — *проекции функции g на подмножества $T_1 \cup T_2$, $T_1 \cup T_3$, $T_2 \cup T_3$.*

В таком случае $g = \max\{g_{12}, g_{13}, g_{23}\}$.

Доказательство. Выберем произвольно некоторую разметку $\bar{k}^*$ и зафиксируем для ее дальнейшего рассмотрения. Справедливо неравенство

$$g(\bar{k}^*) \ge \max\{g_{12}(k^*(T_1), k^*(T_2)), g_{13}(k^*(T_1), k^*(T_3)), g_{23}(k^*(T_2), k^*(T_3))\}. \tag{3}$$

Действительно, по определению проекции функции справедливы неравенства

$$g(\bar{k}^*) \ge g_{12}(k^*(T_1), k^*(T_2)), \quad g(\bar{k}^*) \ge g_{13}(k^*(T_1), k^*(T_3)), \quad g(\bar{k}^*) \ge g_{23}(k^*(T_2), k^*(T_3)),$$

которые эквивалентны неравенству (3). Неравенство (3) будет использовано и далее, а не только для доказательства леммы, и поэтому отметим, что оно справедливо для любой функции g, а не только для тех, которые инвариантны относительно того или иного оператора.

Докажем, что

$$g(\bar{k}^*) \le \max\{g_{12}(k^*(T_1), k^*(T_2)), g_{13}(k^*(T_1), k^*(T_3)), g_{23}(k^*(T_2), k^*(T_3))\}. \tag{4}$$

Пусть

$$w_{12} = g_{12}(k^*(T_1), k^*(T_2)), \quad w_{13} = g_{13}(k^*(T_1), k^*(T_3)), \quad w_{23} = g_{23}(k^*(T_2), k^*(T_3)).$$

Это значит, что существуют три разметки $\bar{k}_{12}$, $\bar{k}_{13}$, $\bar{k}_{23}$ формата $K^T \to W$,

$$\bar{k}_{12} = (k^*(T_1), k^*(T_2), k_{12}(T_3)),$$

$$\bar{k}_{13} = (k^*(T_1), k_{13}(T_2), k^*(T_3)),$$

$$\bar{k}_{23} = (k_{23}(T_1), k^*(T_2), k^*(T_3)),$$

такие, что

$$g(\bar{k}_{12}) = g(k^*(T_1), k^*(T_2), k_{12}(T_3)) = w_{12}, \tag{5}$$

$$g(\bar{k}_{13}) = g(k^*(T_1), k_{13}(T_2), k^*(T_3)) = w_{13}, \tag{6}$$

$$g(\bar{k}_{23}) = g(k_{23}(T_1), k^*(T_2), k^*(T_3)) = w_{23}. \tag{7}$$

Результатом применения какого-либо мажоритарного оператора к разметкам $\bar{k}_{12}$, $\bar{k}_{13}$, $\bar{k}_{23}$ является именно разметка $\bar{k}^* = (k^*(T_1), k^*(T_2), k^*(T_3))$. По условию леммы функция g инвариантна относительно некоторого такого оператора, и поэтому из (5)–(7) вытекает, что $g(\bar{k}^*) \le \max\{w_{12}, w_{13}, w_{23}\}$, т.е. неравенство (4). Лемма доказана.

Из доказанной леммы немедленно следует, что любую функцию n переменных вида $K^n \to W$, для которой существует мажоритарный полиморфизм, можно представить с помощью $0.5n(n-1)$ функций двух переменных. В контексте данной статьи этот результат выражает следующая

Теорема 1. *Для минимаксной задачи любого порядка, имеющей мажоритарный полиморфизм, существует эквивалентная ей задача второго порядка.*

Исходные данные задачи второго порядка будем задавать не в виде пятерки (1), а в виде четверки

$$z = \langle T, K, W, (g_{tt'} : K \times K \to W | t \in T, t' \in T) \rangle. \tag{8}$$

При этом $g_{tt'}(k, k') = g_{t't}(k', k)$ для любых $t \in T$, $t' \in T$, $k \in K$, $k' \in K$ и к тому же $g_{tt}(k, k') = \min W$ для любого $t \in T$. На разметке $\bar{k} \in K^T$ целевая функция $g : K^T \to W$ задачи (8) принимает значение $\max\limits_{t \in T} \max\limits_{t' \in T} g_{tt'}(k(t), k(t'))$.

4. АЛГОРИТМЫ

4.1. Преобразование задачи произвольного порядка в задачу второго порядка

Трудоемкость преобразования задачи порядка n в задачу второго порядка, как и трудоемкость всех последующих алгоритмов, будет измеряться количеством операций $\max\{w, w'\}$ и $\min\{w, w'\}$ – количеством выборов большего или меньшего из двух чисел.

Как следует из леммы 3, для функции $g : K^T \to W$, инвариантной относительно некоторого мажоритарного оператора, существуют такие числа $g_{tt'}(k, k')$, $t \in T$, $t' \in T$, $k \in K$, $k' \in K$, что

$$g(\bar{k}) = \max_{t \in T} \max_{t' \in T} g_{tt'}(k(t), k(t')).$$

Преобразование функции g, представленной в виде таблицы $\mathrm{Tab} = \{(\bar{k}, g(\bar{k}))|\bar{k} \in K^T\}$, в массив чисел $g_{tt'}(k, k')$ выполняет следующий очевидный алгоритм.

Алгоритм 1. Приведение функции $g : K^T \to W$ к функции второго порядка

0. Для всех $t \in T$, $t' \in T$, $k \in K$, $k' \in K$

$g_{tt'}(k, k') = \max W$;

1. для всех $(\bar{k}, w) \in \mathrm{Tab}$ и всех $t \in T$, $t' \in T$

$g_{tt'}(k(t), k(t')) = \min\{g_{tt'}(k(t), k(t')), w\}$;

2. для всех $(\bar{k}, w) \in \mathrm{Tab}$

если $w \neq \max_{t \in T} \max_{t' \in T} g_{tt'}(k(t), k(t'))$, то выдать сообщение “отказ”.

Если априори известно, что входная функция g инвариантна относительно некоторого мажоритарного оператора, пусть даже неизвестного, проверка условий по п. 2 избыточна: ни одно из них не выполнится. Однако проверка этих условий расширяет область действия алгоритма, позволяя подавать на его вход любые функции $|T|$ переменных, а не только те, для которых существует мажоритарный полиморфизм. При обработке любой такой функции либо происходит сообщение “отказ”, либо не происходит. Сообщение “отказ” возможно только в случае, если для входной функции отсутствует мажоритарный полиморфизм. Отсутствие сообщения “отказ” означает, что алгоритм представил входную функцию в виде функции второго порядка, и этот факт не зависит от того, существует для входной функции мажоритарный полиморфизм или нет. Трудоемкость алгоритма имеет порядок $(|K|^2 + |\mathrm{Tab}|)|T|^2$.

Преобразование исходной задачи $z = \langle T, K, W, \Gamma \subset 2^T, (g_{T'} : K^{T'} \to W|T' \in \Gamma)\rangle$ порядка n в эквивалентную задачу $z' = \langle T, K, W, (g_{tt'} : K \times K \to W|t \in T, t' \in T)\rangle$ второго порядка выполняет следующий алгоритм.

Алгоритм 2. Снижение порядка задачи до второго порядка

0. Для всех $t \in T$, $t' \in T$, $k \in K$, $k' \in K$

$g_{tt'}(k, k') = \min W$;

1. для всех $T' \in \Gamma$

1.1. с помощью Алгоритма 1

преобразовать $g_{T'} : K^{T'} \to W$ в массив чисел $g^*_{tt'}(k, k')$, $t \in T'$, $t' \in T'$, $k \in K$, $k' \in K$;

1.2. для всех $t \in T'$, $t' \in T'$, $k \in K$, $k' \in K$

$g_{tt'}(k, k') = \max\{g_{tt'}(k, k'), g^*_{tt'}(k, k')\}$.

Подобно алгоритму 1 на вход алгоритма 2 можно подавать любые задачи, а не только инвариантные относительно некоторого мажоритарного оператора. Работа алгоритма 2 либо сопровождается сообщением “отказ”, который генерирует алгоритм 1 при выполнении п. 1.1, либо завершается преобразованием исходной задачи в задачу второго порядка. Ситуация “отказ” возможна только в случае, когда для исходной задачи отсутствует мажоритарный полиморфизм.

Трудоемкость алгоритма зависит полиномиально от численных параметров задачи: количества $|T|$ объектов, количества $|K|$ меток, количества $|\Gamma|$ подмножеств в структуре, порядка n задачи и, наконец, объема $l = \Sigma_{T' \in \Gamma} |\mathrm{Tab}(T')|$ исходных данных задачи. Порядок трудоемкости п. 0 алгоритма равен $|T|^2 \times |K|^2$. Порядок трудоемкости п. 1.1 алгоритма равен $\Sigma_{T' \in \Gamma} (|K|^2 + |\mathrm{Tab}(T')|)|T'|^2$, что превышает $|\Gamma| \times |K|^2 \times n^2 + l \times n^2$. Порядок трудоемкости п. 1.2 алгоритма равен $|\Gamma| \times |K|^2 \times n^2$.

4.2. Преобразование звезды в симплекс

Решение минимаксной задачи второго порядка основано на многократном преобразовании задач специального вида, которое уместно назвать преобразованием звезды в симплекс.

Определение 8. Структуру $\{(t_0, t)|t \in T\}$ множества $T \cup \{t_0\}$, $t_0 \notin T$, назовем *звездой*; структуру $\{\{t, t'\}|t \in T, t' \in T\}$ множества T назовем *симплексом*.

В задаче, структура которой является звездой, качество $g(\bar{k})$ разметки $\bar{k} : T \cup (t_0) \to K$ определяется выражением

$$g(\bar{k}) = \max_{t \in T} g_t(k(t_0), k(t)),$$

где g_t – функция $K \times K \to W$ двух переменных, заданная для каждого $t \in T$. Пусть существует, хотя и не обязательно известен, мажоритарный оператор p, являющийся полиморфизмом любой функции g_t, $t \in T$, и пусть $g_T : K^T \to W$ – проекция $g : K^{T \cup \{t_0\}} \to W$ на T. Это значит, что $g_T(k(T)) = \min_{k_0} \max_{t \in T} g_t(k_0, k(t))$. Из лемм 1, 2 и 3 следуют три утверждения:

1. функция $g_T : K^T \to W$ инвариантна относительно p;

2. функция g_T представима в виде

$$g_T(k(T)) = \max_{t \in T} \max_{t' \in T} g_{tt'}(k(t), k(t')),$$

где $g_{tt'} : K \times K \to W$ – проекция $g : K^{T \cup \{t_0\} \to W}$ на подмножество $\{t, t'\}$;

3. функции $g_{tt'}$, $t \in T$, $t' \in T$, инвариантны относительно оператора p.

Преобразование звезды в симплекс – это преобразование совокупности функций g_t, $t \in T$, в совокупность функций $g_{tt'}$, $t \in T$, $t' \in T$. Это преобразование выполняется на основании следующих трех общих правил эквивалентного преобразования формул, составленных из операций max и min. Пусть X и Y – некоторые конечные множества, а $f : X \to W$ и $\varphi : Y \to W$ – две функции. В таком случае

$$\min_{x \in X, y \in Y} \max\{f(x), \varphi(y)\} = \max\{\min_{x \in X} f(x), \min_{y \in Y} \varphi(y)\}, \tag{9}$$

а для любого $x \in X$

$$\min_{y \in Y} \max\{f(x), \varphi(y)\} = \max\{f(x), \min_{y \in Y} \varphi(y)\}. \tag{10}$$

Пусть I – некоторое конечное множество, X_i, $i \in I \cup \{0\}$, – конечные множества, $f_i : X_i \to W$, $i \in I \cup \{0\}$, – функции. В таком случае для любого $x_0 \in X_0$

$$\max\{f_0(x_0), \max_{i \in I} \min_{x_i \in X_i} f_i(x_i)\} = \max\{f_0(x_0), \max_{i \in I \cup \{0\}} \min_{x_i \in X_i} f_i(x_i)\}. \tag{11}$$

Выберем два объекта $t_1 \in T$, $t_2 \in T$ и зафиксируем их для дальнейшего рассмотрения. Пусть $T' = T \backslash \{t_1, t_2\}$. Для искомого значения $g_{t_1 t_2}(k(t_1), k(t_2))$ справедлива цепочка равенств

$$g_{t_1 t_2}(k(t_1), k(t_2)) = \min_{k_0 \in K} \min_{k(T') \in K^{T'}} g(\bar{k}) = \min_{k_0 \in K} \min_{k(T') \in K^{T'}} \max_{t \in T} g_t(k_0, k(t)) =$$

$$
\begin{aligned}
&= \min_{k_0 \in K} \max\left\{ g_{t_1}(k_0, k(t_1)), g_{t_2}(k_0, k(t_2)), \min_{k(T') \in K^{T'}} \max_{t' \in T'} g_{t'}(k_0, k(t')) \right\} = \\
&= \min_{k_0 \in K} \max\{ g_{t_1}(k_0, k(t_1)), g_{t_2}(k_0, k(t_2)), \max_{t' \in T'} \min_{k(t') \in K} g_{t'}(k_0, k(t'))\} = \\
&= \min_{k_0 \in K} \max\{ g_{t_1}(k_0, k(t_1)), g_{t_2}(k_0, k(t_2)), \max_{t \in T} \min_{k(t) \in K} g_t(k_0, k(t))\}.
\end{aligned} \tag{12}
$$

Первые два звена в этой цепочке справедливы по определению, третье – по правилу (10), четвертое – по правилу (9), пятое – по правилу (11). Из цепочки (12) непосредственно следует алгоритм преобразования звезды в симплекс, т.е. вычисления значений $g_{tt'}(k, k')$, $t \in T$, $t' \in T$, $k \in K$, $k' \in K$, по заданным значениям $g_t(k_0, k)$, $t \in T$, $k_0 \in K$, $k \in K$.

Алгоритм 3. Преобразование звезды в симплекс

1. Для всех $k_0 \in K$ вычислить величину

$$q(k_0) = \max_{t \in T} \min_{k \in K} g_t(k_0, k);$$

2. для всех $t \in T$, $t' \in T$, $k \in K$, $k' \in K$ вычислить величину

$$g_{tt'}(k, k') = \min_{k_0 \in K} \max\{g_t(k_0, k), g_{t'}(k_0, k'), q(k_0)\}.$$

Трудоемкость п. 1 алгоритма имеет порядок $|K|^2 \times |T|$, а п. 2 – порядок $|K|^3 \times |T|^2$. Сложность преобразования звезды в симплекс имеет порядок $|K|^3 \times |T|^2$.

4.3. Поиск минимаксной разметки

Пусть T – множество, $t^* \in T$, $T^* = T \backslash \{t^*\}$, $g_T\colon K^T \to W$ – функция, и g_{T^*} – проекция g_T на T^*. Непосредственно из определения проекции следует, что

$$\min_{k(T) \in K^T} g_T(k(T)) = \min_{k(T^*) \in K^{T^*}} g_{T^*}(k(T^*)), \quad \operatorname*{argmin}_{k(T) \in K^T} g_T(k(T)) = (k^*(T^*), k^*(t^*)), \tag{13}$$

$$\text{где} \quad k^*(T^*) = \operatorname*{argmin}_{k(T^*) \in K^{T^*}} g_{T^*}(k(T^*)), \quad k^*(t^*) = \operatorname*{argmin}_{k(t^*) \in K} g_T(k^*(T^*), k^*(t^*)). \tag{14}$$

Если функция $g_T : K^T \to W$ представлена в виде $g_T(k(T)) = \max\limits_{t \in T, t' \in T} g_{tt'}(k(t), k(t'))$ и имеет мажоритарный полиморфизм, то ее проекция $g_{T^*}\colon K^{T^*} \to W$ на подмножество $T^* = T \backslash \{t^*\}$, $t^* \in T$, представима в виде $g_{T^*}(k(T^*)) = \max\limits_{t \in T^*, t' \in T^*} g^*_{tt'}(k(t), k(t'))$. Преобразование исходных величин $g_{tt'}(k, k')$ в искомые величины $g^*_{tt'}(k, k')$ сводится к преобразованию звезды в симплекс, как это следует из цепочки

$$
\begin{aligned}
g_{T^*}(k(T^*)) &= \min_{k(t^*) \in K} g_T(k(t^*), k(T^*)) = \min_{k(t^*) \in K} \max_{t \in T, t' \in T} g_{tt'}(k(t), k(t')) = \\
&= \min_{k(t^*) \in K} \max\{ \max_{t \in T^*, t' \in T^*} g_{tt'}(k(t), k(t')), \max_{t \in T^*} g_{tt'}(k(t), k(t^*))\} = \\
&= \max\{ \max_{t \in T^*, t' \in T^*} g_{tt'}(k(t), k(t')), \min_{k(t^*) \in K} \max_{t \in T^*} g_{tt^*}(k(t), k(t^*))\} = \\
&= \max\{ \max_{t \in T^*, t' \in T^*} g_{tt'}(k(t), k(t')), \max_{t \in T^*, t' \in T^*} g'_{tt'}(k(t), k(t'))\} = \\
&= \max_{t \in T^*, t' \in T^*} \max\{g_{tt'}(k(t), k(t')), g'_{tt'}(k(t), k(t'))\} = \max_{t \in T^*, t' \in T^*} g^*_{tt'}(k(t), k(t')).
\end{aligned} \tag{15}
$$

Вспомогательные величины $g'_{tt'}(k(t), k(t'))$ в этой цепочке вычисляет алгоритм 3 на основании величин $g_{tt^*}(k(t), k(t^*))$. Этим обеспечивается равенство

$$\min_{k(t^*) \in K} \max_{t \in T^*} g_{tt^*}(k(t), k(t^*)) = \max_{t \in T^*, t' \in T^*} g'_{tt'}(k(t), k(t'))$$

для всех разметок $k(T^*) : T^* \to K$ и, как следствие, пятое равенство в цепочке (15). Остальные звенья цепочки очевидны. Справедливость цепочки приводит к следующему алгоритму преобразования величин $g_{tt'}(k, k')$, определяющих исходную функцию $g_T : K^T \to W$, в величины $g^*_{tt'}(k, k')$, определяющие ее проекцию на подмножество $T^* = T\backslash\{t^*\}$.

Алгоритм 4. Проекция функции $g_T : K^T \to W$ на подмножество $T^* = T\backslash\{t^*\}$

1. Для всех $k^* \in K$ вычислить

   $$g(k^*) = \max_{t \in T^*} \min_{k \in K} g_{t^*t}(k^*, k);$$

2. для всех $t \in T^*$, $t' \in T^*$, $k \in K$, $k' \in K$ вычислить

   $$g'_{tt'}(k, k') = \min_{k^* \in K} \max\{g_{tt^*}(k, k^*), g_{t't^*}(k', k^*), q(k^*)\};$$

3. для всех $t \in T^*$, $t' \in T^*$, $k \in K$, $k' \in K$ вычислить

   $$g^*_{tt'}(k, k') = \max\{g_{tt'}(k, k'), g'_{tt'}(k, k')\}.$$

Трудоемкость этого алгоритма имеет тот же порядок $|K|^3 \times |T|^2$, что и трудоемкость алгоритма 3 преобразования звезды в симплекс.

Из отношений (13) и (14) следует, что решение минимаксной задачи разметки сводится к последовательному проецированию ее целевой функции на совокупность вложенных друг в друга подмножеств. Пусть $T_0 = \{t_1, t_2, ..., t_m\}$, $T_i = T_{i-1}\backslash\{t_i\}$, $i = 1, 2, ..., m$. Поиск минимума функции $g_{T_0} : K^{T_0} \to W$ и разметки, в которой этот минимум достигается, выполняет следующий алгоритм.

Алгоритм 5. Решение минимаксной задачи разметки

1. Для $i = 1, 2, ..., m - 3$

   построить функцию g_{T_i} – проекцию $g_{T_{i-1}}$ на T_i с помощью алгоритма 4;

2. вычислить значение искомого минимума

   $$w^* = \min_{k(t_{m-2}) \in K} \min_{k(t_{m-1}) \in K} \min_{k(t_m) \in K} g_{T_{m-3}}(k(t_{m-2}), k(t_{m-1}), k(t_m))$$

   и разметку

   $$k^*(T_{m-3}) = \operatorname{argmin}_{k(t_{m-2}) \in K} \min_{k(t_{m-1}) \in K} \min_{k(t_m) \in K} g_{T_{m-3}}(k(t_{m-2}), k(t_{m-1}), k(t_m));$$

3. для $i = m - 4, m - 5, ..., 1, 0$

   {построить разметку $k^*(T_i) = (k^*(T_{i+1}), \operatorname{argmin}_{k(t_{i+1})} g_{T_i}(k(t_{i+1}), k^*(T_{i+1})))$;

   если $\min_{k(t_{i+1})} g_{T_i}(k(t_{i+1}), k^*(T_{i+1})) > w^*$, то выдать сообщение “отказ”}.

Если априори известно, что для исходной функции g_{T_0} существует мажоритарный полиморфизм, то проверка условия в п. 3 алгоритма является лишней. Это условие никогда не выполняется, и разметка $k^*(T_0)$, построенная алгоритмом, оптимальна. Однако, как и для алгоритма 1, наличие этой проверки расширяет сферу действия алгоритма 5. Множество задач, которые можно подавать на его вход, – это NP-полный класс всех возможных минимаксных задач второго порядка. Из неравенства (3), доказанного при доказательстве леммы 3, следует, что число w^*, вычисленное на п. 2 алгоритма, не превосходит искомый минимум и это справедливо для любой задачи, а не только инвариантной относительно того или иного оператора. Если алгоритм не сообщил “отказ”, то качество построенной разметки $k^*(T_0)$ равно w^*. Это значит, что $k^*(T_0)$ и есть искомая разметка, и этот факт не зависит от того, существует для входной задачи тот или иной полиморфизм или нет. Если алгоритм сообщил “отказ”, то это означает, что для решаемой задачи отсутствует мажоритарный полиморфизм. Таким образом, алгоритм завершает свою работу либо построением разметки, и в этом случае эта разметка гарантированно оптимальна, либо сообщением, что для решаемой задачи отсутствует мажоритарный полиморфизм, и это утверждение гарантированно верное.

Трудоемкость алгоритма 5 определяется $(|T| - 3)$-кратным выполнением алгоритма 4 и таким образом доказана теорема.

Теорема 2. *Пусть T и K – конечные множества, W – упорядоченное множество, K^T – множество разметок вида $\bar{k} : T \to K$, $g_{tt'} : K \times K \to W$ – функция, заданная для каждой пары $t \in T$, $t' \in T$, $g(\bar{k}) = \max_{t \in T, t' \in T} g_{tt'}(k(t), k(t'))$ – качество разметки $\bar{k}$.*

Если существует мажоритарный оператор, относительно которого инвариантны все функции $g_{tt'}$, $t \in T$, $t' \in T$, то величина $\min_{\bar{k} \in K^T} g(\bar{k})$ и разметка $\bar{k}^* = \operatorname{argmin}_{\bar{k} \in K^T} g(\bar{k})$ вычисляются за время порядка $|T|^3 \times |K|^3$.

5. ОБСУЖДЕНИЕ РЕЗУЛЬТАТА

Основным результатом статьи является алгоритм дискретной оптимизации функций, инвариантных относительно мажоритарного оператора. Исходные данные задачи представлены в виде $\langle T, K, W, \Gamma \subset 2^T, (g_{T'} | T' \in \Gamma)\rangle$, где T и K – конечные множества, W – упорядоченное множество, $g_{T'}$, $T' \in \Gamma$, – функции вида $K^{T'} \to W$, представленные в виде таблиц $\mathrm{Tab}(T') = \{(\bar{k}, g_{T'}(\bar{k})) | \bar{k} \in K^{T'}\}$. По указанным исходным данным следует вычислить величину $\min_{\bar{k} \in K^T} \max_{T' \in \Gamma} g_{T'}(k(T'))$ и разметку $\operatorname{argmin}_{\bar{k} \in K^T} \max_{T' \in \Gamma} g_{T'}(k(T'))$.

Трудоемкость разработанного алгоритма зависит от $|T|$, $|K|$, $|\Gamma|$, порядка $n = \max_{T' \in \Gamma} |T'|$ задачи и $l = \Sigma_{T' \in \Gamma} |\mathrm{Tab}(T')|$ – суммарного объема таблиц, представляющих функции $g_{T'}$, $T' \in \Gamma$. Задачи второго порядка решаются алгоритмом 5 за время порядка $|T|^3 \times |K|^3$. Задачи более высокого порядка преобразуются в задачи второго порядка с помощью алгоритма 2 за время порядка $|T|^2 \times |K|^2 + |\Gamma| \times |K|^2 \times n^2 + l \times n^2$, а затем решаются с помощью алгоритма 5.

Этот результат нетрудно обобщить на следующий более широкий класс минимаксных задач разметки. Пусть для каждого $t \in T$ определен мажоритарный оператор $p_t : K \times K \times K \to K$. Совокупность $P = (p_t | t \in T)$ понимается как оператор вида $K^T \times K^T \times K^T \to K^T$, который для любой тройки $(k_1(T), k_2(T), k_3(T))$ разметок указывает разметку $P(k_1(T), k_2(T), k_3(T)) = k_0(T)$ такую, что $k_0(t) = p_t(k_1(t), k_2(t), k_3(t))$, $t \in T$.

Минимаксную задачу $z = \langle T, K, W, \Gamma \subset 2^T, (g_{T'} | T' \in \Gamma)\rangle$ назовем *инвариантной относительно совокупности $P = (p_t : K \times K \times K \to K | t \in T)$ операторов*, если для любого $T' \in \Gamma$ и любых трех разметок $k_1(T') \in K^{T'}$, $k_2(T') \in K^{T'}$, $k_3(T') = K^{T'}$ справедливо неравенство

$$\max\{g_{T'}(k_1(T')), g_{T'}(k_2(T')), g_{T'}(k_3(T'))\} \geq g_{T'}(P(k_1(T'), k_2(T'), k_3(T'))).$$

Совокупность P операторов в этом случае назовем *коллективным мажоритарным полиморфизмом задачи z*.

Все доказанные в статье утверждения без особого труда обобщаются на класс задач, для которых существует коллективный мажоритарный полиморфизм, а мажоритарный полиморфизм может и отсутствовать. Алгоритм 2 преобразует любую задачу из этого расширенного класса в задачу второго порядка, а алгоритм 5 решает любую задачу второго порядка из этого расширенного класса. Это достоинство было бы существенно обесценено, если бы не было известно, как себя поведут алгоритмы 2 и 5 при подаче на их вход задачи вне этого класса. Для этого случая их следовало бы дополнить алгоритмом входного контроля задачи, который проверяет существование для нее того или иного коллективного мажоритарного полиморфизма. Нам неизвестен такой алгоритм, и, возможно, он довольно трудный. Достоинство алгоритмов 2 и 5 в том, что для них такой входной контроль не нужен. Множество задач, которые можно подавать на вход алгоритма 2 – это NP-полный класс всех возможных минимаксных задач, а на вход алгоритма 5 можно подавать любую минимаксную задачу второго порядка, множество которых также образует NP-полный класс. Работа с любой задачей завершается либо ее решением, либо сообщением об отказе, которое возможно только в том случае, если для задачи, предложенной для решения, отсутствует коллективный мажоритарный полиморфизм.

СПИСОК ЛИТЕРАТУРЫ

1. *Bulatov A., Jeavons P.* Tractable constraints closed under a binary operation. Technical Report PGR-TR-12-00, Oxford University Computing Laboratory, 2000.
2. *Bulatov A.* Tractable conservative constraint satisfaction problems. In Proc. 18th Annual IEEE Symposium on Logic in Computer Science, LICS'03, Washington, DC, USA, 2003.
3. *Bulatov A.* Complexity of conservative constraint satisfaction problems // ACM Trans. Comput. Logic. 12(4), 24:1–24:66, July 2011.
4. *Rossi F., van Beek P., Walsh T.* Handbook of Constraint Programming. Foundation of Artificial Intelligence. Elsevier Science, 2006.
5. *Schlesinger M.I., Flach B.* Some solvable subclasses of structural recognition problems. Czech Pattern Recognition Workshop, 2000.
6. *Щербина О.А.* Удовлетворение ограничений и программирование в ограничениях // Интеллектуальные системы. 2011. Т. 15. № 1–4. С. 53–170.