УДК 519.157

М.И. ШЛЕЗИНГЕР, Б. ФЛАХ, Е.В. ВОДОЛАЗСКИЙ

ПОИСК ЗАДАННОГО КОЛИЧЕСТВА РЕШЕНИЙ СИСТЕМЫ РАЗМЫТЫХ ОГРАНИЧЕНИЙ[1]

Аннотация. Исследована минимаксная модификация задачи распознавания совместимости системы ограничений, когда для каждого решения определена не бинарная допустимость, а ее количественная характеристика. Описанный в статье алгоритм находит за полиномиальное время требуемое количество наилучших решений системы размытых ограничений, если эти ограничения инвариантны относительно некоторого мажоритарного оператора. Существенно, что для реализации алгоритма не требуется знания этого оператора, более того, не требуется гарантировать его существование. Для любой системы размытых ограничений алгоритм либо находит заданное количество наиболее допустимых решений, либо выдает отказ от решения задачи. Последнее возможно, только если для решаемой системы ограничений такой оператор отсутствует.

Ключевые слова: дискретная оптимизация, минимаксные задачи, разметки, инварианты, полиморфизмы.

ВВЕДЕНИЕ

Исследуемая задача относится к проблеме дискретной оптимизации функций и распознавания совместимости ограничений [1]. В терминах дискретной оптимизации эта задача имеет следующий формат. Пусть K — конечный алфавит символов, W — упорядоченное множество, $\bar{x} = (x_1, x_2, \ldots, x_n) \in K^n$ — последовательность символов длины $n > 0$, $\varphi : K^n \to W$ — функция n переменных. Для заданного целого числа $0 < d < |K|^n$ задача d-оптимизации функции φ состоит в отыскании подмножества $Sol \subset K^n$, $|Sol| = d$, такого что $\varphi(\bar{x}) \le \varphi(\bar{y})$ для любых $\bar{x} \in Sol$, $\bar{y} \notin Sol$. Оптимизация функции в обычном смысле этого слова является 1-оптимизацией.

Аналогично d-оптимизации можно определить d-совместимость системы ограничений как естественное обобщение классической проблемы распознавания совместимости [1]. Пусть $R_j \subset K^n$, $j = 1, 2, \ldots, m$, — система ограничений на n-ки символов из K. Распознавание d-совместимости состоит в ответе на вопрос, существуют ли такие попарно различные n-ки $\bar{x}_i \in K^n$, $i = 1, 2, \ldots, d$, что $\bar{x}_i \in R_j$ для всех i и j. Классическая задача совместимости ограничений, т.е. проверка неравенства $\bigcap_{j=1}^{n} R_j \ne \varnothing$, является задачей 1-совместимости. Очевидно, распознавание d-совместимости сводится к d-оптимизации функции $\varphi : K^n \to \{0, 1\}$, принимающей значение $\varphi(\bar{x}) = 0$ при $\bar{x} \in \bigcap_{j=1}^{n} R_j$ и значение $\varphi(\bar{x}) = 1$ в противном случае.

Размытое ограничение понимается как функция $\varphi : K^n \to [0, 1]$, значение $\varphi(\bar{x})$ которой указывает степень допустимости решения $\bar{x}$ относительно ограничения φ, а степень допустимости решения $\bar{x}$ в системе $\varphi_j : K^n \to [0, 1]$, $j = 1, 2, \ldots, m$, размытых ограничений понимается как число $\min_j \varphi_j(\bar{x})$ [2, 3].

[1]Работа выполнена в рамках проекта № 7/2/267-39/17 «Создание интеллектуальных технологий на основе методов и средств образного мышления» (Отделение информатики НАН Украины) и при поддержке гранта № 16-05872S «Probabilistic graphical models and deep learning» (Грантовая агентура Чешской Республики).

Распознавание d-совместимости системы размытых ограничений понимается как отыскание d наиболее допустимых решений.

Необходимость d-оптимизации, $d > 1$, возникает в повседневных реальных ситуациях. Это происходит, например, когда искомый оптимум должен удовлетворять априори известному ограничению, а оптимизация φ с учетом этого ограничения сопряжена с серьезными вычислительными трудностями. Иная ситуация возникает, когда искомое решение x^* предназначено для последующей обработки с помощью алгоритма с ограниченной областью определения. В этом случае трудно априори формализовать ограничение на решение, а можно лишь апостериори констатировать, что текущее полученное решение x^* непригодно, и запросить решение, отличающееся от x^*, но с как можно меньшим значением функции φ. Наконец, формализовать ограничение на решение в принципе невозможно, когда оно адресуется человеку, который может отвергнуть предложенное решение x^* в силу только ему известных причин. Во всех этих ситуациях целесообразна d-оптимизация с последовательно возрастающим d и отыскание последовательности ухудшающихся решений, пока не будет найдено оптимальное решение, удовлетворяющее объективным, но заранее не известным ограничениям.

Основополагающая идея d-оптимизации и распознавания d-совместимости принадлежит Лоулеру (Eugene U. Lawler) [4] и состоит в следующем. Функцию $\psi : K^{l-1} \to W$ назовем сечением функции $\varphi : K^l \to W$, если существуют такие $1 \le i \le l$ и $a \in K$, что $\psi(x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n) = \varphi(x_1, \dots, x_{i-1}, a, x_{i+1}, \dots, x_n)$. Для функции $\varphi : K^n \to W$ обозначим $\Phi(\varphi)$ минимальное множество функций, содержащее φ и все возможные сечения каждой функции $\psi \in \Phi(\varphi)$. Если для функции φ существует алгоритм 1-оптимизации любой поданной на его вход функции $\psi \in \Phi(\varphi)$, то он называется консервативным алгоритмом оптимизации φ. Лоулер показал, что d-оптимизация функции φ сводится к $(d \times n \times |K|)$-кратному вызову консервативного алгоритма и 1-оптимизации определенной совокупности функций $\psi \in \Phi(\varphi)$, количество которых равно $d \times n \times |K|$.

Наряду с общим методом Лоулера, применимого для d-оптимизации широкого класса функций, известен ряд частных, но более быстродействующих алгоритмов d-оптимизации тех или иных подклассов функций [5–9]. Будем говорить, что подкласс функций допускает ускоренную d-оптимизацию, если вычислительная сложность d-оптимизации на этом подклассе на порядок меньше сложности алгоритма Лоулера. Известные подклассы ускоренно решаемых задач формулируются в терминах распознавания совместимости ограничений. Основные идеи решения этих подклассов задач описаны в разд. 2. Там же указано место предлагаемого метода среди известных.

В разд. 3–6 рассмотрен ускоренный алгоритм d-оптимизации функций, инвариантных относительно мажоритарных операторов (инвариантность функции относительно оператора определена в разд. 2). В работе [10] описан консервативный алгоритм 1-оптимизации таких функций. Следовательно, к этим функциям применима d-оптимизация по методу Лоулера, которая выполняется за время $O(C(n, |K|) \times d \times |K| \times n)$, где $C(n, |K|)$ — время однократного выполнения имеющегося консервативного алгоритма. Предлагаемый алгоритм выполняет d-оптимизацию за время $O(C(n, |K|) + d \times |K| \times n^2)$.

1. ФОРМУЛИРОВКА ОСНОВНЫХ ПОНЯТИЙ

Определение 1. Для конечного множества I, упорядоченного множества W, функции $\varphi : I \to W$ и целого числа d, $0 < d < |I|$, выражение $I^* = \arg\min_{i \in I}{}_{[d]}\, \varphi(i)$ обозначает, что

$$I^* \subset I,\ |I^*| = d \text{ и } \varphi(i) \le \varphi(j) \text{ для любых } i \in I^*,\ j \notin I^*; \tag{1}$$

для $d \ge |I|$ выражение $I^* = \arg\min_{i \in I}{}_{[d]}\, \varphi(i)$ обозначает, что $I^* = I$. ■

Подобно обозначению $\arg\min_{i\in I}\varphi(i)$, которое не обязательно определяет единственный элемент $i^*\in I$, обозначение $\arg\min_{i\in I}{}_{[d]}\,\varphi(i)$ не определяет однозначно единственное подмножество $I^*\subset I$. Выражение $I^*=\arg\min_{i\in I}{}_{[d]}\,\varphi(i)$ следует понимать как краткую запись утверждения, что подмножество I^* удовлетворяет условиям (1), возможно, наряду с какими-то другими подмножествами. Подмножества вида $\arg\min_{i\in I}{}_{[d]}$ имеют следующие свойства.

Лемма 1. Пусть I — конечное множество, $\varphi: I\to W$ — функция, $I^*=\arg\min_{i\in I}{}_{[d]}\,\varphi(i)$. Тогда

$$\max_{i\in I'}\varphi(i)\ge\max_{i\in I^*}\varphi(i)\ \text{ для любого }\ I'\subset I,\ |I'|=d. \tag{2}$$

Пусть $I_0\subset I$, $I_0^*=\arg\min_{i\in I_0}{}_{[d]}\,\varphi(i)$ и $\varphi(i)\ge\max_{j\in I_0^*}\varphi(j)$ для любого $i\notin I_0^*$. Тогда

$$I_0^*=\arg\min_{i\in I}{}_{[d]}\,\varphi(i).\ \blacksquare \tag{3}$$

Свойства, указанные в лемме 1, следуют непосредственно из определения 1.

Будем считать, что упорядоченное множество W содержит элемент ∞ такой, что $\infty\ge a$ для любого $a\in W$.

Определение 2. Для целого числа d, $0<d\le|I|$, выражение $\min_{i\in I}{}_{[d]}\,\varphi(i)$ означает совокупность значений $\varphi(i)$, $i\in\arg\min_{i\in I}{}_{[d]}\,\varphi(i)$, представленную монотонно неубывающей последовательностью; для $d>|I|$ выражение $\min_{i\in I}{}_{[d]}\,\varphi(i)$ означает последовательность длины d, которая начинается последовательностью значений $\varphi(i)$, $i\in I$, упорядоченной в неубывающем порядке, и заканчивается $d-|I|$ элементами, равными ∞. $\blacksquare$

В отличие от $\arg\min_{i\in I}{}_{[d]}\,\varphi(i)$ последовательность $\min_{i\in I}{}_{[d]}\,\varphi(i)$ определена однозначно для любой функции $\varphi: I\to W$.

В данной работе, как и в ряде работ, упомянутых во введении, вопросы ускоренной d-оптимизации исследованы в рамках следующей общей задачи разметки [1]. Пусть T и K — два конечных множества, элементы которых назовем объектами и метками. Структурой множества T назовем некоторое подмножество $\mathbb{S}\subset 2^T$. Для любого подмножества $S\subseteq T$ обозначим K^S множество всех возможных отображений вида $S\to K$. Пусть $\bar{x}: T\to K$ — отображение, называемое разметкой. Обозначим x_i значение разметки $\bar{x}: T\to K$ для объекта $i\in T$, а x_S — ее сужение на подмножество $S\subset T$. В случае, когда необходимо обратить внимание на то, что $S=P\cup R$, $P\cap R=\varnothing$, сужение $\bar{x}$ на S будем обозначать не в виде $x_{P\cup R}$, а в виде пары (x_P, x_R).

Пусть $(W,\oplus,\otimes)$ — коммутативное полукольцо, и для каждого $S\in\mathbb{S}$ определена функция $\varphi_S: K^S\to W$, представленная таблицей $\mathrm{Tab}(S)=\{(x_S,\varphi_S(x_S))\mid x\in K^S\}$. Поскольку операции $\oplus$ и $\otimes$ коммутативны и ассоциативны, далее используются выражения $\bigoplus_{i\in I}a_i$ и $\bigotimes_{i\in I}a_i$ для «суммы» конечного количества слагаемых и «произведения» конечного количества сомножителей.

Определение 3. Исходные данные $(\oplus, \otimes)$-задачи разметки — это пятерка

$$\Phi = \langle T, K, (W, \oplus, \otimes), \mathbb{S}, (\varphi_S \mid S \in \mathbb{S}) \rangle, \tag{4}$$

а ее решение — значение

$$w^* = \bigoplus_{\bar{x} \in K^T} \bigotimes_{S \in \mathbb{S}} \varphi_S(x_S) \in W \tag{5}$$

и разметка $\bar{x}^* \in K^T$, такая что

$$\bigotimes_{S \in \mathbb{S}} \varphi_S(x_S^*) = \bigoplus_{\bar{x} \in K^T} \bigotimes_{S \in \mathbb{S}} \varphi_S(x_S) = w^*, \tag{6}$$

если она существует. ■

Число $\mathrm{Ord}(\Phi) = \max_{S \in \mathbb{S}} |S|$ назовем порядком структуры $\mathbb{S}$, порядком задачи Φ и порядком функции $\bigotimes_{S \in \mathbb{S}} \varphi_S(x_S)$. Число $\mathrm{Vol}(\Phi) = \sum_{S \in \mathbb{S}} |\mathrm{Tab}(S)|$ назовем объемом исходных данных задачи. Классическая задача распознавания совместимости ограничений [11] есть задача (4)–(6), сформулированная для полукольца $\langle \{0, 1\}, \vee, \wedge \rangle$. Задача распознавания совместимости размытых ограничений [2, 3] есть задача (4)–(6), сформулированная на полукольце $\langle W, \min, \max \rangle$, где W — некоторое упорядоченное множество. Оптимизационная задача разметки [12] — это задача (4)–(6), сформулированная на полукольце $\langle W, \min, \otimes \rangle$, где $\otimes$ — любая операция, образующая с операцией min полукольцо на множестве W.

Пусть для каждого $i \in T$ определен трехместный оператор $p_i : K \times K \times K \to K$. Совокупность $P = (p_i \mid i \in T)$ этих операторов будем понимать как оператор $K^S \times K^S \times K^S \to K^S$, определенный для каждого $S \subseteq T$. В результате применения его к тройке разметок $x, y, z \in K^S$ получается разметка $P(x, y, z) = v \in K^S$, такая что $v_i = p_i(x_i, y_i, z_i)$, $i \in S$.

Определение 4. Оператор $P = (p_i \mid i \in T)$ назовем полиморфизмом функции $\varphi_S : K^S \to W$ на полукольце $(W, \min, \max)$, а функцию φ_S — инвариантом оператора P, если для любой тройки разметок $x, y, z \in K^S$ выполняется неравенство

$$\max\{\varphi_S(x), \varphi_S(y), \varphi_S(z)\} \geq \varphi_S(P(x, y, z)); \tag{7}$$

оператор P назовем полиморфизмом $(\min, \max)$-задачи (4)–(6), а задачу — инвариантной относительно P, если каждая функция $\varphi_S : K^S \to W$, $S \in \mathbb{S}$, инвариантна относительно P. ■

Определение 4 является естественным расширением на полукольцо $(\min, \max)$ известных понятий полиморфизмов и инвариантов, определенных на полукольце $(\{0, 1\}, \vee, \wedge)$ [11], где условию (7) соответствует условие $\varphi_S(x) \wedge \varphi_S(y) \wedge \varphi_S(z) \to \varphi_S(P(x, y, z))$. Определение 4 обобщает классические понятия полиморфизмов и инвариантов еще и в другом направлении. В теории $(\vee, \wedge)$-задач полагают, что оператор $P = (p_i \mid i \in T)$ однороден на множестве T в том смысле, что его компоненты p_i не зависят от $i \in T$. В соответствии с определением 4 оператор $P = (p_i \mid i \in T)$ может быть неоднородным в том смысле, что каждому $i \in T$ соответствует свой оператор p_i.

Известно [13], что множество функций, имеющих общий полиморфизм на полукольце $(\{0, 1\}, \vee, \wedge)$, замкнуто по определенным операциям над этими функциями. Следующие две леммы выражают эти свойства на полукольце $(W, \min, \max)$.

Определение 5. Проекция функции $\varphi : K^T \to W$ на подмножество $S \subset T$ — это функция $\varphi_S : K^S \to W$, принимающая значения $\varphi_S(x_S) = \min_{x_R \in K^R} \varphi(x_S, x_R)$, где $R = T \setminus S$. ■

Лемма 2. Если функция $\varphi : K^T \to W$ инвариантна на полукольце $(W, \min, \max)$ относительно оператора P, то ее проекция φ_S на любое подмножество $S \subset T$ инвариантна относительно этого же оператора. ■

Лемма 3. Если функции $\varphi : K^T \to W$ и $\psi : K^T \to W$ инвариантны на полукольце $(W, \min, \max)$ относительно оператора P, то функция $\omega : K^T \to W$ со значениями $\omega(\bar{x}) = \max\{\varphi(\bar{x}), \psi(\bar{x})\}$, $\bar{x} \in K^T$, инвариантна относительно этого же оператора. ■

Доказательства обеих лемм приведены в [10]. Они являются естественным обобщением лемм, доказанных для полукольца $(\{0, 1\}, \vee, \wedge)$ [13].

Определение 6. Оператор $P = (p_i \mid i \in T)$ называется мажоритарным, если для любого $i \in T$ и любых $x, y \in K$ выполняются равенства $p_i(y, x, x) = p_i(x, y, x) = p_i(x, x, y) = x$. ■

Функции, инвариантные на полукольце $(\min, \max)$ относительно мажоритарных операторов, имеют дополнительное важное свойство. Для функций с мажоритарным полиморфизмом в полукольце $(\vee, \wedge)$ это свойство доказано в [13] и обозначает, что такие функции представимы в виде конъюнкции своих проекций. Следующая лемма представляет естественное расширение этого утверждения на функции, для которых существует мажоритарный полиморфизм в полукольце $(\min, \max)$.

Лемма 4. Пусть $\varphi : K^T \to W$ — функция, а Q, R, S — попарно непересекающиеся подмножества в T, такие что $Q \cup R \cup S = T$. Пусть φ_{QR}, φ_{QS}, φ_{RS} — проекции функции φ на подмножества $Q \cup R$, $Q \cup S$, $R \cup S$ соответственно.

Если для φ существует мажоритарный полиморфизм, то равенство

$$\varphi(\bar{x}) = \max\{\varphi_{QR}(x_Q, x_R), \varphi_{QS}(x_Q, x_S), \varphi_{RS}(x_R, x_S)\}$$

выполняется для любой разметки $\bar{x} = (x_Q, x_R, x_S) \in K^T$. ■

Лемма доказана в работе [10].

2. ЗАДАЧИ РАЗМЕТКИ И d-ОПТИМИЗАЦИЯ: ОБЗОР ИЗВЕСТНЫХ РЕШЕНИЙ

Известно [14], что, выбирая надлежащим образом полукольцо $(W, \oplus, \otimes)$, в формате $(\oplus, \otimes)$-задачи можно представить такие обширные классы задач, как проверка совместимости ограничений [11, 13], подсчет количества решений системы ограничений [5, 9], оценка совместимости размытых ограничений [2, 3] и разнообразные оптимизационные задачи вида $\min\limits_{\bar{x} \in K^T} \bigotimes\limits_{S \in \mathbb{S}} \varphi_S(x_S)$, где $\otimes$ — любая операция, образующая с операцией min полукольцо на упорядоченном множестве W [12]. В работе [15] показано, что в формате $(\oplus, \otimes)$-задачи можно представить и вычисление $\min\limits_{\bar{x} \in K^T}{}_{[d]} \bigotimes\limits_{S \in \mathbb{S}} \varphi_S(x_S)$ — качеств d наилучших разметок. В терминах и обозначениях настоящей статьи этот результат сформулирован в следующей лемме. Частный случай этой леммы для d-минимизации функций вида $\sum\limits_{S \in \mathbb{S}} \varphi_S(x_S)$ доказан в [16, 17].

Лемма 5. Для коммутативного полукольца $(W, \min, \otimes)$, функций $\varphi_S : K^S \to W$, $S \in \mathbb{S}$, и числа d существует коммутативное полукольцо $(W_{[d]}, \oplus_{[d]}, \otimes_{[d]})$ и функции $\varphi_{[d]S} : K^S \to W_{[d]}$, $S \in \mathbb{S}$, такие что $\bigoplus\limits_{\bar{x} \in K^T}{}_{[d]} \bigotimes\limits_{S \in \mathbb{S}}{}_{[d]} \varphi_{[d]S}(x_S) = \min\limits_{\bar{x} \in K^T}{}_{[d]} \bigotimes\limits_{S \in \mathbb{S}} \varphi_S(x_S)$. ■

Доказательство. Определим $W_{[d]}$ как множество всех монотонно неубывающих последовательностей $(a_1, a_2, \ldots, a_d)$, $a_i \in W$, и для заданного $a \in W$ обозначим $\langle a \rangle \in W_{[d]}$ последовательность, первый элемент которой равен a, а остальные равны ∞.

Пусть $a=(a_1, a_2, \dots, a_d)$ и $b=(b_1, b_2, \dots, b_d)$ — две последовательности. Представим совокупность величин $(a_1, a_2, \dots, a_d, b_1, b_2, \dots, b_d)$ в виде монотонно неубывающей последовательности и определим сумму $a \oplus_{[d]} b$ как последовательность первых d элементов в полученной упорядоченной последовательности.

Представим совокупность величин $a_i \otimes b_j$, $i, j = 1, \dots, d$, в виде монотонно неубывающей последовательности и определим произведение $a \otimes_{[d]} b$ как последовательность первых d элементов в полученной упорядоченной последовательности.

Из того факта, что тройка $(W, \min, \otimes)$ — коммутативное полукольцо с нулевым элементом ∞, следует, что операции $a \oplus_{[d]} b$ и $a \otimes_{[d]} b$ на множестве $W_{[d]}$ образуют коммутативное полукольцо с нулевым элементом $\langle \infty \rangle$.

Определим функции $\varphi_{[d]S} : K^S \to W_{[d]}$, $S \in \mathbb{S}$, так, что $\varphi_{[d]S}(x_S) = \langle\langle \varphi_S(x_S) \rangle\rangle$. Очевидно, что для любой функции $\varphi : I \to W$ справедливо, что

$$\bigotimes_{i \in I}{}_{[d]} \langle\langle \varphi(i) \rangle\rangle = \left\langle\!\!\left\langle \bigotimes_{i \in I} \varphi(i) \right\rangle\!\!\right\rangle, \text{ и поэтому } \bigotimes_{S \in \mathbb{S}}{}_{[d]} \langle\langle \varphi_S(x_S) \rangle\rangle = \left\langle\!\!\left\langle \bigotimes_{S \in \mathbb{S}} \varphi_S(x_S) \right\rangle\!\!\right\rangle.$$

Очевидно также, что

$$\bigoplus_{i \in I}{}_{[d]} \langle\langle \varphi(i) \rangle\rangle = \min_{i \in I}{}_{[d]} \varphi(i), \text{ и поэтому } \bigoplus_{\bar{x} \in K^T}{}_{[d]} \bigotimes_{S \in \mathbb{S}}{}_{[d]} \langle\langle \varphi_S(x_S) \rangle\rangle = \min_{\bar{x} \in K^T}{}_{[d]} \bigotimes_{S \in \mathbb{S}} \varphi_S(x_S). \blacksquare$$

Лемма 6. Пусть $Q_{\min}$ и $Q_{\otimes}$ — времена выполнения операций $\min\{a, b\}$ и $a \otimes b$. Вычислительная сложность операции $a \oplus_{[d]} b$ имеет порядок $d \cdot Q_{\min}$. Вычислительная сложность операции $a \otimes_{[d]} b$ имеет порядок $(d \log d) \cdot (Q_{\min} + Q_{\otimes})$. $\blacksquare$

Оба утверждения леммы достаточно легко доказываются, хотя второе утверждение несколько менее очевидное, чем первое.

Множество $(\oplus_{[d]}, \otimes_{[d]})$-задач d-оптимизации содержит множество $(\min, \otimes)$-задач 1-оптимизации, которое в свою очередь содержит NP-полный класс $(\vee, \wedge)$-задач совместимости ограничений. Поэтому универсальное решение всего класса $(\oplus_{[d]}, \otimes_{[d]})$-задач невозможно, тем более ускоренное решение этого класса. Ускоренная d-оптимизация возможна не более чем для таких подклассов функций, которые допускают эффективную 1-оптимизацию. Такие подклассы функций вида $\bigoplus_{S \in \mathbb{S}} \varphi_S(x_S)$ формируют двумя способами. При первом способе требуют, чтобы структура $\mathbb{S}$ задачи принадлежала определенному заданному классу структур, разрешая при этом произвольные функции $\varphi_S : K^S \to W$, $S \in \mathbb{S}$. При втором способе, наоборот, разрешают произвольные структуры $\mathbb{S}$ задачи, требуя при этом, чтобы функции $\varphi_S : K^S \to W$ принадлежали определенному заданному классу функций, называемому языком задач.

Ограничения на структуру $\mathbb{S}$ множества T формулируют в терминах гиперграфа $G(T, \mathbb{S})$ с множеством T вершин и множеством $\mathbb{S}$ гиперребер. Сложность структуры $\mathbb{S}$ характеризуют так называемой шириной $\mathrm{width}(\mathbb{S})$ ациклического покрытия гиперграфа $G(T, \mathbb{S})$ [18]. Решение $(\min, \otimes)$-задач 1-оптимизации с малой шириной $\mathrm{width}(\mathbb{S})$ можно получить с помощью непоследовательного динамического программирования [19], и на этом основан ряд алгоритмов d-оптимизации, в которых d-оптимизация сводится к той или иной разумной последовательности 1-оптимизаций. В работах [6, 8, 15] представлено элегантное обобщение этих частных методов на основе леммы 5 и того факта, что непоследовательное динамическое программирование применимо для решения $(\oplus, \otimes)$-задач на произвольном полукольце, а не только $(\min, \otimes)$-задач. При этом в силу леммы 6 d-оптимизация функции вида $\bigotimes_{S \in \mathbb{S}} \varphi_S(x_S)$ оказывается всего в $d \times \log d$ раз сложнее ее 1-оптимизации, а не в $d \times |T| \times |K|$ раз. Таким образом, в исследованиях d-оптимизации функций вида $\bigotimes_{S\in\mathbb{S}}\varphi_S(x_S)$ на структурах малой ширины достигнута определенная степень завершенности.

Что касается исследований d-оптимизации функций вида $\bigotimes_{S\in\mathbb{S}}\varphi_S(x_S)$ на произвольных структурах, то они еще далеки от какой-либо завершенности. В работе [12] достаточно полно описаны языки, т.е. подклассы функций $\varphi_S : K^S \to W$, порождающие полиномиально разрешимые подклассы (min, $\otimes$)-задач 1-оптимизации. Наиболее известными среди них являются задачи субмодулярной минимизации (см. [20]). Подклассы функций $\varphi_S : K^S \to \{0,1\}$, порождающие полиномиально разрешимые подклассы ($\vee$, $\wedge$)-задач совместимости, исследованы в [11]. Для этих подклассов функций (и для некоторых других) известны консервативные алгоритмы 1-оптимизации. Следовательно, для всех этих классов функций выполнима d-оптимизация по методу Лоулера или по методу, описанному в разд. 3. Однако авторам настоящей статьи не известны работы по ускоренной d-оптимизации для какого-либо из указанных классов. Описанная в разд. 4–6 ускоренная d-минимизация функций вида $\max_{S\in\mathbb{S}}\varphi_S(x_S)$, инвариантных относительно мажоритарного оператора, заполняет определенный фрагмент в этом еще не заполненном пробеле.

3. БАЗОВЫЙ АЛГОРИТМ *d*-ОПТИМИЗАЦИИ

Пусть $\varphi : K^T \to W$ — функция на множестве разметок, для которой следует построить подмножество $\arg\min_{\bar{x}\in K^T}{}_{[d]}\,\varphi(\bar{x})$. Для любого подмножества $S\subset T$ обозначим φ_S проекцию функции φ на подмножество S. Для $S\subseteq T$ введем для краткости обозначение $Sol_S = \arg\min_{x_S\in K^S}{}_{[d]}\,\varphi_S(x_S)$. Искомым решением задачи является Sol_T, и его построение основано на следующей взаимосвязи между Sol_S и $Sol_{S\setminus\{i\}}$.

Теорема 1. Пусть $S\subset T$, $i\in S$, $P=S\setminus\{i\}$, $Sol_P = \arg\min_{x_P\in K^P}{}_{[d]}\,\varphi_P(x_P)$. Пусть $Sol^*\subset K^S$ — некоторое подмножество разметок.

Если $Sol^* = \arg\min_{(x_P,x_i)\in Sol_P\times K}{}_{[d]}\,\varphi_S(x_P,x_i)$, то $Sol^* = \arg\min_{x_S\in K^S}{}_{[d]}\,\varphi_S(x_S)$. ■

Доказательство. Обозначим $y(x_P)=\arg\min_{x_i\in K}\varphi_S(x_P,x_i)$ и докажем, что

$$\max_{x_P\in Sol_P}\varphi_P(x_P)=\max_{x_P\in Sol_P}\varphi_S(x_P,\,y(x_P))\ge\max_{(x_P,x_i)\in Sol^*}\varphi_S(x_P,x_i). \qquad (8)$$

Равенство в этой цепочке справедливо, потому что функция φ_P, являясь проекцией функции φ на P, одновременно есть проекция функции φ_S на P. Неравенство в этой цепочке справедливо в силу общего свойства (2) множеств вида $\arg\min_{[d]}$. Действительно, $Sol^* = \arg\min_{(x_P,x_i)\in Sol_P\times K}{}_{[d]}\,\varphi_S(x_P,x_i)$, а множество $\{(x_P,y(x_P))\,|\,x_P\in Sol_P\}$ есть подмножество в $Sol_P\times K$, состоящее из d разметок.

Для любой разметки $(x_P^*,x_i)\notin Sol_P\times K$ справедлива цепочка

$$\varphi_S(x_P^*,x_i)\ge\min_{y\in K}\varphi_S(x_P^*,\,y)=$$

$$=\varphi_P(x_P^*)\ge\max_{x_P\in Sol_P}\varphi_P(x_P)\ge\max_{(x_P,x_i)\in Sol^*}\varphi_S(x_P,x_i). \qquad (9)$$

Первое неравенство в этой цепочке очевидно. Следующее за ним равенство справедливо, потому что φ_P есть проекция φ_S на P. Неравенство $\varphi_P(x_P^*) \geq \max\limits_{x_P \in Sol_P} \varphi_P(x_P)$ в цепочке справедливо, потому что $(x_P^*, x_i) \notin Sol_P \times K$ и, следовательно, $x_P^* \notin Sol_P$. Последнее неравенство в цепочке (9) следует из ранее доказанной цепочки (8).

Таким образом, неравенство $\varphi_S(x_P, x_i) \geq \max\limits_{(x_P, x_i) \in Sol^*} \varphi_S(x_P, x_i)$ выполняется для любой разметки $(x_P, x_i) \notin Sol_P \times K$, откуда в силу (3) следует, что $Sol^* = \arg\min\limits_{x_S \in K^S}{}_{[d]} \varphi_S(x_S)$. ■

Из доказанной теоремы следует алгоритм d-оптимизации φ. Алгоритм строит $Sol_{\{i_0\}}$ для подмножества, состоящего из единственного объекта i_0, а затем выполняет последовательность пересчетов $Sol_{S \setminus \{i\}}$ в Sol_S. Результатом работы алггоритма является $Sol_T = \arg\min\limits_{\bar{x} \in K^T}{}_{[d]} \varphi(\bar{x})$.

Сложность построения $Sol_{\{i_0\}}$ определяется сложностью следующих вычислений:

— для каждой метки $x_{i_0} \in K$ вычисляют величины $\varphi_{\{i_0\}}(x_{i_0}) = \min\limits_{x_R \in K^R} \varphi(x_{i_0}, x_R)$, где $R = T \setminus \{i_0\}$. Сложность этих вычислений равна сложности $|K|$-кратной 1-оптимизации;

— из полученных $|K|$ величин $\varphi_{\{i_0\}}(x_{i_0})$, $x_{i_0} \in K$, выбирают d наименьших. Известно [21], что сложность этого выбора линейно зависит от $|K|$, т.е. равна $O(|K|)$.

Таким образом, сложность построения $Sol_{\{i_0\}}$ равна $O(|K| \times C) + O(|K|) = O(|K| \times C)$, где C — сложность однократной 1-оптимизации.

Сложность пересчета $Sol_{S \setminus \{i\}}$ в Sol_S определяется сложностью следующих вычислений:

— для каждой разметки $x_{S \setminus \{i\}} \in Sol_{S \setminus \{i\}}$ и каждого значения $x_i \in K$ вычисляют величины

$$\varphi_S(x_{S \setminus \{i\}}, x_i) = \min_{x_R \in K^R} \varphi(x_{S \setminus \{i\}}, x_i, x_R), \tag{10}$$

где $R = T \setminus S$. Сложность этих вычислений равна сложности $d \times |K|$-кратной 1-оптимизации;

— из полученных $d \times |K|$ величин $\varphi_S(x_{S \setminus \{i\}}, x_i)$ выбирают d наименьших.

Таким образом, пересчет $Sol_{S \setminus \{i\}}$ в Sol_S имеет сложность $O(d \times |K| \times C)$, а сложность вычисления $Sol_T = \arg\min\limits_{\bar{x} \in K^T}{}_{[d]} \varphi(\bar{x})$ равна $O(d \times |K| \times |T| \times C)$.

Алгоритм d-оптимизации, котоый следует из теоремы 1, имеет такую же область применимости и такую же сложность, что и классический алгоритм Лоулера. Оба алгоритма применимы при наличии консервативного алгоритма 1-оптимизации и сводятся к $(d \times |K| \times |T|)$-кратной 1-оптимизации. Однако из теоремы 1 следует не только алгоритм d-оптимизации, по существу не уступающий классическому алгоритму Лоулера, но и возможность ускоренной d-оптимизации таких функций, которые допускают эффективный пересчет проекции φ_S в проекцию $\varphi_{S \setminus \{i\}}$ для любого подмножества $S \subset T$. В частности, как будет показано в разд. 5, такой эффективный пересчет возможен для функций $\varphi : K^T \to W$ вида $\varphi(\bar{x}) = \max\limits_{S \in \mathbb{S}} \varphi_S(x_S)$ на структурах второго порядка, инвариантных относительно мажоритарного оператора.

4. ПРЕОБРАЗОВАНИЕ (min, max)-ЗАДАЧ В ЗАДАЧИ ВТОРОГО ПОРЯДКА

Далее исследуются $(\oplus, \otimes)$-задачи не общего вида (4)–(6), а задачи в полукольце (min, max) при произвольно выбранных, но фиксированных множествах K и W. Поэтому задачу будем представлять не в виде пятерки (4), а более кратко в виде тройки $\Phi = \langle T, \mathbb{S}, (\varphi_S \mid S \in \mathbb{S})\rangle$. Целевая функция $\varphi : K^T \to W$ задачи Φ определяет качество $\varphi(\bar{x}) = \max\limits_{S \in \mathbb{S}} \varphi_S(x_S)$ разметки $\bar{x} \in K^T$. Решением задачи Φ является подмножество $Sol(\Phi) = \arg\min\limits_{\bar{x} \in K^T}{}_{[d]}\, \varphi(\bar{x})$, состоящее из d наилучших разметок.

Лемма 7. Если для задачи $\Phi = \langle T, \mathbb{S}, (\varphi_S \mid S \in \mathbb{S})\rangle$ существует мажоритарный полиморфизм, то существует задача $\Psi = \langle T, \mathbb{S}^*, (\psi_S \mid S \in \mathbb{S}^*)\rangle$ на структуре второго порядка, такая что для любой разметки $\bar{x} \in K^T$, выполняется равенство

$$\max_{S \in \mathbb{S}} \varphi_S(x_S) = \max_{S \in \mathbb{S}^*} \psi_S(x_S). \ \blacksquare \tag{11}$$

Лемма 7 является очевидным следствием леммы 4.

Задачу на структуре второго порядка будем представлять не тройкой $\langle T, \mathbb{S}, (\varphi_S \mid S \in \mathbb{S})\rangle$, а парой $\langle T, (\psi_{ij} \mid i, j \in T)\rangle$, где ψ_{ij}, $i, j \in T$, — функции вида $K \times K \to W$, такие что $\psi_{ij}(k, k') = \psi_{ji}(k', k)$ для любых $i, j \in T$, $k, k' \in K$ и $\psi_{ii}(k, k') = \min W$ для любого $i \in T$. Правая сторона равенства (11), т.е. значение целевой функции Ψ на разметке $\bar{x} \in K^T$, приобретает вид $\max\limits_{i, j \in T} \psi_{ij}(x_i, x_j)$.

Функции φ_S, $S \in \mathbb{S}$, преобразуются в функции ψ_{ij}, $i, j \in T$, следующим образом. Вначале вычисляются проекции функций φ_S, $S \in \mathbb{S}$, на пары $\{i, j\} \subset S$, т.е. вычисляются величины

$$\psi_{ij}^S(x_i, x_j) = \min_{x_R \in K^R} \varphi_S(x_R, x_i, x_j), \tag{12}$$

где $R = S \setminus \{i, j\}$. Сложность этого пересчета равна $O(\mathrm{Vol}(\Phi) \times (\mathrm{Ord}(\Phi))^2)$, где $\mathrm{Vol}(\Phi)$ и $\mathrm{Ord}(\Phi)$ — объем исходных данных и порядок задачи Φ соответственно.

Затем вычисляются величины

$$\psi_{ij}(x, y) = \max_{S \in \mathbb{S}} \psi_{ij}^S(x, y) \tag{13}$$

для каждой пары $i, j \in T$ и для каждой пары $x, y \in K$. Сложность этих вычислений равна $O(|T|^2 \times |K|^2 \times |\mathbb{S}|)$.

Вычисления по формулам (12) и (13) не зависят от вида мажоритарного оператора, существование которого является условием леммы 7. Более того, эти вычисления применимы к любым функциям φ_S, $S \in \mathbb{S}$, а не только к тем, для которых такой оператор существует. Однако, если для Φ существует мажоритарный полиморфизм, то гарантируется эквивалентность функций Φ и Ψ, а в противном случае такая эквивалентность не исключена, но и не гарантируется. В силу этого, казалось бы, на вход алгоритма (12), (13) следует подавать только такие функции Φ, для которых существует мажоритарный полиморфизм. Проверка инвариантности входной функции Φ относительно некоторого априори неизвестного мажоритарного оператора может оказаться слишком сложной, особенно, если учесть, что речь идет о неоднородных по множеству T полиморфизмах. Следуя [18], можно преодолеть эти трудности, включив в вычисления (12) и (13) процедуру самоконтроля, которая предотвратит неэквивалентное преобразование Φ в Ψ без проведения, возможно, трудной проверки инвариантности задачи Φ относительно некоторого мажоритарного полиморфизма. В рассматриваемый алгоритм (12), (13) сравнительно легко ввести такой самоконтроль. Для этого необходимо проверить, выполняются ли равенства

$$\varphi_S(x_S) = \max_{i,j \in \mathbb{S}} \psi^S_{i,j}(x_i, x_j) \tag{14}$$

для всех $S \in \mathbb{S}$ и для всех $x_S \in K^S$.

Сложность $O(\mathrm{Vol}(\Phi) \times (\mathrm{Ord}(\Phi))^2)$ этой проверки такая же, как и вычисления (12), и поэтому включение самоконтроля (14) в алгоритм (12), (13) не увеличивает его сложности. Самоконтролирующийся алгоритм эквивалентного преобразования состоит из вычислений (12), самоконтроля (14) и вычислений (13), если выполняются все равенства (14). В этом случае результатом работы алгоритма является задача Ψ второго порядка, эквивалентная исходной задаче Φ. Если хотя бы одно из равенств (14) не выполняется, алгоритм выдает сообщение о невозможности желаемого эквивалентного преобразования. Однако это происходит, только если для исходной задачи Φ отсутствует мажоритарный полиморфизм.

Следует иметь в виду, что алгоритм может завершиться успешным получением задачи Ψ для некоторых входных задач Φ, для которых мажоритарный полиморфизм отсутствует. Это, вообще-то, положительное свойство неизбежно приводит к риску при последующем решении задачи Ψ, для которой тоже может отсутствовать мажоритарный полиморфизм. Поэтому последующие алгоритмы решения полученной задачи Ψ второго порядка также должны быть самокотролирующимися.

5. ФУНКЦИИ ВТОРОГО ПОРЯДКА И ИХ *d*-ОПТИМИЗАЦИЯ

Примем для удобства, что $T = \{1, 2, \dots, n\}$, и определим семейство подмножеств $S_r \subset T$, $r \in \{2, 3, \dots, n\}$, $S_n = T$, $S_{r-1} = S_r \setminus \{r\}$.

Определение 7. Задачу $\Phi = \langle T, (\varphi_{ij} \mid i, j \in T) \rangle$ назовем согласованной по семейству $S_r \subset T$, $r = 2, \dots, n$, если для любого r проекция функции $\max\limits_{i,j \in T} \varphi_{ij}(x_i, x_j)$ на S_r есть функция $\max\limits_{i,j \in S_r} \varphi_{ij}(x_i, x_j)$. ■

Ускоренная d-оптимизации функций вида $\max\limits_{i,j \in T} \varphi_{ij}(x_i, x_j)$ основана на двух идеях. Первая идея состоит в том, что если для задачи $\Phi = \langle T, (\varphi_{ij} \mid i, j \in T) \rangle$ существует мажоритарный полиморфизм, то для нее существует согласованная задача $\Phi^* = \langle T, (\varphi^*_{ij} \mid i, j \in T) \rangle$, эквивалентная задаче Φ в том смысле, что $\max\limits_{i,j \in T} \varphi^*_{ij}(x_i, x_j) = \max\limits_{i,j \in T} \varphi_{ij}(x_i, x_j)$ для всех $\bar{x} \in K^T$. Вторая идея состоит в эффективном вычислении $\arg \min\limits_{\bar{x} \in K^T}{}_{[d]} \max\limits_{i,j \in T} \varphi^*_{i,j}(x_i, x_j)$ с помощью базового алгоритма d-оптимизации, основанного на теореме 1. Эквивалентное преобразование $(\min, \max)$-задачи к согласованному виду основано на следующей лемме.

Лемма 8. Пусть $\Phi = \langle T, (\varphi_{ij} \mid i, j \in T) \rangle$ — $(\min, \max)$-задача, $t \in T$, $S = T \setminus \{t\}$. Если для Φ существует мажоритарный полиморфизм, то существует задача $\Psi = \langle T, (\psi_{ij} \mid i, j \in T) \rangle$, такая что

$$\max_{i,j \in T} \varphi_{ij}(x_i, x_j) = \max_{i,j \in T} \psi_{ij}(x_i, x_j), \ \bar{x} \in K^T, \tag{15}$$

$$\max_{i,j \in S} \psi_{ij}(x_i, x_j) = \min_{x_t \in K} \left[\max_{i,j \in T} \varphi_{ij}(x_i, x_j) \right], \ x \in K^S. \ ■ \tag{16}$$

Доказательство. Введем в рассмотрение вспомогательную функцию $\varphi^*: K^T \to W$ со значениями $\varphi^*(\bar{x}) = \max\limits_{i \in S} \varphi_{it}(x_i, x_t)$ и функцию $\psi^*: K^S \to W$ — проекцию φ^* на S. Поскольку для задачи Φ существует мажоритарный полиморфизм, то в силу леммы 2 функция ψ^* инвариантна относительно этого же полиморфизма. В свою очередь, из леммы 4 следует, что функция ψ^* представима в виде $\psi^*(\bar{x}) = \max\limits_{i,j\in S} \psi^*_{ij}(x_i, x_j)$, где ψ^*_{ij} — проекция φ^* на $\{i, j\}$. Определим величины $\psi_{ij}(x_i, x_j)$, $i, j \in T$, $x_i, x_j \in K$, так что

$$\psi_{ij}(x_i, x_j) = \max\{\varphi_{ij}(x_i, x_j), \psi^*_{ij}(x_i, x_j)\} \text{ для } i, j \in S, \tag{17}$$

$$\psi_{it}(x_i, x_t) = \varphi_{it}(x_i, x_t) \text{ для } i \in S, \tag{18}$$

и покажем, что для них справедливы равенства (15) и (16).

Поскольку ψ^* — это проекция φ^* на S, то неравенство $\max\limits_{i,j\in S} \psi^*_{ij}(x_i, x_j) \le$ $\le \max\limits_{i\in S} \varphi_{it}(x_i, x_t)$ выполняется для любого $\bar{x} \in K^T$ и поэтому

$$\max_{i,j\in T} \varphi_{ij}(x_i, x_j) = \max\left\{ \max_{i,j\in S} \varphi_{ij}(x_i, x_j), \max_{i\in S} \varphi_{it}(x_i, x_t) \right\} =$$

$$= \max\left\{ \max_{i,j\in S} \varphi_{ij}(x_i, x_j), \max_{i,j\in S} \psi^*_{ij}(x_i, x_j), \max_{i\in S} \varphi_{it}(x_i, x_t) \right\} =$$

$$= \max\left\{ \max_{i,j\in S} \psi_{ij}(x_i, x_j), \max_{i\in S} \psi_{it}(x_i, x_t) \right\} = \max_{i,j\in T} \psi_{ij}(x_i, x_j),$$

что доказывает равенство (15). Равенство (16) доказывает цепочка

$$\min_{x_t\in K} \max_{i,j\in T} \varphi_{ij}(x_i, x_j) = \min_{x_t\in K} \max\left\{ \max_{i,j\in S} \varphi_{ij}(x_i, x_j), \max_{i\in S} \varphi_{it}(x_i, x_t) \right\} =$$

$$= \max\left\{ \max_{i,j\in S} \varphi_{ij}(x_i, x_j), \min_{x_t\in K} \max_{i\in S} \varphi_{it}(x_i, x_t) \right\} =$$

$$= \max\left\{ \max_{i,j\in S} \varphi_{ij}(x_i, x_j), \max_{i,j\in S} \psi^*_{ij}(x_i, x_j) \right\} = \max_{i,j\in S} \psi_{ij}(x_i, x_j). \ \blacksquare$$

Сложность преобразования Φ в Ψ определяется сложностью вычисления проекции функции $\varphi^*: K^T \to W$ на $S = T \setminus \{t\}$, т.е. пересчета величин $\varphi_{it}(x_i, x_t)$, $i \in S$, в величины $\psi^*_{ij}(x_i, x_j)$, $i, j \in S$. Выполним этот пересчет подобно преобразованию звезды в симплекс, описанному в [22] для иной ситуации. В рассматриваемом случае этот пересчет основан на следующих рассуждениях.

Выберем объекты $m, n \in S$, обозначим $R = S \setminus \{m, n\}$ и для функции ψ^*_{mn}, которая является проекцией φ^* на $\{m, n\}$, запишем цепочку равенств

$$\psi^*_{mn}(x_m, x_n) = \min_{x_t\in K} \min_{x_R\in K^R} \varphi^*(\bar{x}) = \min_{x_t\in K} \min_{x_R\in K^R} \max_{i\in S} \varphi_{it}(x_i, x_t) =$$

$$= \min_{x_t\in K} \min_{x_R\in K^R} \max\left\{ \varphi_{mt}(x_m, x_t), \varphi_{nt}(x_n, x_t), \max_{i\in R} \varphi_{it}(x_i, x_t) \right\} =$$

$$= \min_{x_t\in K} \max\left\{ \varphi_{mt}(x_m, x_t), \varphi_{nt}(x_n, x_t), \min_{x_R\in K^R} \max_{i\in R} \varphi_{it}(x_i, x_t) \right\} =$$

$$= \min_{x_t\in K} \max\left\{ \varphi_{mt}(x_m, x_t), \varphi_{nt}(x_n, x_t), \max_{i\in R} \min_{x_i\in K} \varphi_{it}(x_i, x_t) \right\} =$$

$$= \min_{x_t \in K} \max \left\{ \varphi_{mt}(x_m, x_t), \varphi_{nt}(x_n, x_t), \max_{i \in S} \min_{x_i \in K} \varphi_{it}(x_i, x_t) \right\}. \qquad (19)$$

Из цепочки (19) следует, что вычисление значений $\psi^*_{ij}(x, y)$, $i, j \in S$, $x, y \in K$, сводится к вычислению вспомогательных величин

$$q_t(k) = \max_{i \in S} \min_{x_i \in K} \varphi_{it}(x_i, k), \; k \in K, \qquad (20)$$

а затем — величин

$$\psi^*_{ij}(x, y) = \min_{k \in K} \max \{\varphi_{it}(x, k), \varphi_{jt}(y, k), q_t(k)\}. \qquad (21)$$

Вычисления (20) имеют сложность порядка $(|K|^2 \times |T|)$, а вычисления (21) — порядка $(|K|^3 \times |T|^2)$. Таким образом, преобразование задачи Φ в задачу Ψ имеет сложность $O(|K|^3 \times |T|^2)$, а эквивалентное преобразование задачи в согласованную — сложность $O(|K|^3 \times |T|^3)$, потому что сводится к T-кратному выполнению операций (20) и (21).

Пусть $S_r \subset T$, $r = 2, 3, \ldots, n$, — семейство подмножеств, таких что $S_n = T$, $S_{r-1} = S_r \setminus \{r\}$, а $\Phi = \langle T, (\varphi_{ij} \mid i, j \in T) \rangle$ — согласованная по этому семейству задача. Обозначим $Sol_r = \arg\min_{x \in K^{S_r}}{}_{[d]} \max_{i, j \in S_r} \varphi_{ij}(x_i, x_j)$. Поскольку $\max_{i, j \in S_r} \varphi_{ij}(x_i, x_j)$ есть проекция функции $\max_{i, j \in T} \varphi_{ij}(x_i, x_j)$ на S_r, то в силу теоремы 1 для множеств Sol_{r-1} и Sol_r справедлива рекурсия

$$Sol_r = \arg\min_{(x^*, k) \in Sol_{r-1} \times K}{}_{[d]} \max \left\{ \max_{i, j \in S_{r-1}} \varphi_{ij}(x^*_i, x^*_j), \max_{i \in S_{r-1}} \varphi_{ir}(x^*_i, k) \right\}. \qquad (22)$$

Алгоритм построения решения $Sol_n = \arg\min_{x \in K^T}{}_{[d]} \max_{i, j \in T} \varphi_{ij}(x_i, x_j)$ начинает работу с построения Sol_2 для $S_2 = \{1, 2\}$, а затем в результате последовательности пересчетов Sol_{r-1} в Sol_r в соответствии с (22) получает решение Sol_n. Построение Sol_2 состоит в выборе d наименьших величин из $|K|^2$ величин $\varphi_{12}(x, y)$, $x, y \in K$:

$$Sol_2 = \arg\min_{x, y \in K}{}_{[d]} \varphi_{12}(x, y), \qquad (23)$$

и имеет сложность $O(|K|^2)$.

Пересчет Sol_{r-1} в Sol_r в соответствии с (22) состоит из следующих вычислений. Для каждой разметки $x^* \in Sol_{r-1}$ и каждой метки $k \in K$ вычисляется величина

$$f(x^*, k) = \max \left\{ \max_{i, j \in S_{r-1}} \varphi_{ij}(x^*_i, x^*_j), \max_{i \in S_{r-1}} \varphi_{ir}(x^*_i, k) \right\}. \qquad (24)$$

С учетом того, что величины $\max_{i, j \in S_{r-1}} \varphi_{ij}(x^*_i, x^*_j)$ уже были получены при формировании Sol_{r-1}, вычисление значения $f(x^*, k)$ для одной пары $x^* \in Sol_{r-1}$, $k \in K$, имеет оценку сложности $O(|T|)$, а не $O(|T|^2)$, а сложность вычисления $f(x^*, k)$ для $d \times |K|$ таких пар равна $O(d \times |K| \times |T|)$.

Затем находят d наименьших чисел в полученной совокупности $d \times |K|$ чисел и строят

$$Sol_r = \arg\min_{(x^*, k) \in Sol_{r-1} \times K}{}_{[d]} f(x^*, k). \qquad (25)$$

Сложность этого поиска равна $O(d \times |K|)$. Суммарная сложность пересчета Sol_{r-1} в Sol_r равна $O(d \times |K| \times |T|)$, потому что состоит из сложности $d \times |K| \times |T|$ вычислений (24) и сложности $O(d \times |K|)$ вычислений (25). В результате $|T|$-кратного пересчета Sol_{r-1} в Sol_r, $r = 3, 4, \ldots, n$, получается решение $Sol_n = \arg\min_{x \in K^T}{}_{[d]} \max_{i,j \in T} \varphi_{ij}(x_i, x_j)$. Сложность этого пересчета имеет порядок $(d \times |K| \times |T|^2)$.

Таким образом, представленный алгоритм d-оптимизации состоит из двух частей: преобразования решаемой задачи в согласованную по формулам (20), (21) и $|T|$-кратного пересчета Sol_{r-1} в Sol_r по формулам (24), (25). Вычислительная сложность алгоритма в целом определяется сложностью первой части, которая не зависит от d и имеет тот же порядок $(|K|^3 \times |T|^3)$, что и известный консервативный алгоритм 1-оптимизации [10]. Вторая составляющая алгоритма имеет сложность $O(d \times |K| \times |T|^2)$ и линейно зависит от d с коэффициентом пропорциональности, который на порядок меньше, чем $|K|^3 \times |T|^3$. Это значит, что вопреки ожиданиям сложность d-оптимизации рассматриваемого класса функций несущественно зависит от d. Если для какой-то функции выполнена 1-оптимизация посредством преобразования этой функции в согласованную, то полученную согласованную функцию можно без существенных затрат использовать и для 2-оптимизации, 3-оптимизации и т.д.

6. САМОКОНТРОЛИРУЮЩИЙСЯ АЛГОРИТМ d-ОПТИМИЗАЦИИ

Применение алгоритма (20)–(25) d-оптимизации имеет те же особенности, что и алгоритма (12), (13) эквивалентного преобразования задач. Для его реализации не требуется знания того мажоритарного оператора, относительно которого инвариантна решаемая задача. Более того, эти вычисления можно выполнить для любой (min, max)-задачи, а не только для тех, которые имеют мажоритарный полиморфизм. При работе с любой входной задачей алгоритм за конечное время выходит на останов, и в этом смысле областью определения алгоритма является NP-трудный класс всех возможных (min, max)-задач второго порядка. Однако, если входная задача $\Phi = \langle T, (\varphi_{ij} \mid i, j \in T) \rangle$ имеет мажоритарный полиморфизм, то алгоритм гарантирует получение правильного результата $Sol_n = \arg\min_{\bar{x} \in K^T}{}_{[d]} \max_{i,j \in T} \varphi_{ij}(x_i, x_j)$, а в противном случае это не гарантируется. Поэтому алгоритм следует дополнить средствами самоконтроля, которые для данного алгоритма менее очевидны, чем для алгоритма (12), (13), и основаны на следующих свойствах d-оптимизации.

Пусть $\Phi = \langle T, (\varphi_{ij} \mid i, j \in T) \rangle$ — (min, max)-задача (не обязательно согласованная). Как и прежде, будем считать, что $T = \{1, 2, \ldots, n\}$. Для $r = 2, 3, \ldots, n$ обозначим S_r подмножество $\{1, 2, \ldots, r\} \subset T$, обозначим x_r метку объекта $r \in T$, а $\bar{x}_r$ — разметку $S_r \to K$ множества $S_r \subset T$. Обозначим также $\varphi_r : K^{S_r} \to W$ функцию r переменных, которая на разметке $\bar{x}_r$ принимает значение $\varphi_r(\bar{x}_r) = \varphi_r(x_1, x_2, \ldots, x_r) = \max_{i,j \in S_r} \varphi_{ij}(x_i, x_j)$. Обозначим $Sol_r = \arg\min_{\bar{x} \in K^{S_r}}{}_{[d]} \varphi_r(\bar{x}_r)$. Для множеств Sol_{r-1} и Sol_r, $r = 3, 4, \ldots, n$, справедливо рекуррентное отношение, подобное сформулированному в теореме 1.

Лемма 9. Пусть r — любое число из множества $r \in \{3, 4, \ldots, n\}$ и

$$Sol_r^* = \arg\min_{(\bar{x}_{r-1}, x_r) \in Sol_{r-1} \times K}{}_{[d]} \varphi_r(\bar{x}_{r-1}, x_r). \tag{26}$$

Если равенство

$$\varphi_{r-1}(\bar{x}_{r-1}) = \min_{x_r \in K} \varphi_r(\bar{x}_{r-1}, x_r) \tag{27}$$

выполняется для любой разметки $\bar{x}_{r-1} \in Sol_{r-1}$, то

$$Sol_r^* = \underset{(\bar{x}_{r-1}, x_r) \in K^{S_r}}{\arg\min}_{[d]} \varphi_r(\bar{x}_{r-1}, x_r). \ \blacksquare$$

Доказательство. Обозначим $y(\bar{x}_{r-1}) = \underset{x_r \in K}{\arg\min} \varphi_r(\bar{x}_{r-1}, x_r)$ и запишем цепочку

$$\max_{\bar{x}_{r-1} \in Sol_{r-1}} \varphi_{r-1}(\bar{x}_{r-1}) = \max_{\bar{x}_{r-1} \in Sol_{r-1}} \varphi_r(\bar{x}_{r-1}, y(\bar{x}_{r-1})) \geq \max_{(\bar{x}_{r-1}, x_r) \in Sol_r^*} \varphi_r(\bar{x}_{r-1}, x_r). \tag{28}$$

Равенство в этой цепочке справедливо в силу условия (27). Следующее за ним неравенство справедливо в силу общего свойства (2) множеств вида $\arg\min_{[d]}$. Действительно, равенство $Sol_r^* = \underset{(\bar{x}_{r-1}, x_r) \in Sol_{r-1} \times K}{\arg\min}_{[d]} \varphi_r(\bar{x}_{r-1}, x_r)$ есть условие (26) леммы, а множество $\{(\bar{x}_{r-1}, y(\bar{x}_{r-1})) \mid \bar{x}_{r-1} \in Sol_{r-1}\}$ есть подмножество в $Sol_{r-1} \times K$, состоящее из d разметок.

Для любой разметки $(\bar{x}_{r-1}^*, x_r) \notin Sol_{r-1} \times K$ справедлива цепочка

$$\varphi_r(\bar{x}_{r-1}^*, x_r) \geq \varphi_{r-1}(\bar{x}_{r-1}^*) \geq \max_{\bar{x}_{r-1} \in Sol_{r-1}} \varphi_{r-1}(\bar{x}_{r-1}) \geq \max_{(\bar{x}_{r-1}, x_r) \in Sol_r^*} \varphi_r(\bar{x}_{r-1}, x_r). \tag{29}$$

Первое неравенство в этой цепочке справедливо, потому что

$$\varphi_r(\bar{x}_{r-1}^*, x_r) = \max\left\{ \max_{i,j \in S_{r-1}} \varphi_{ij}(x_i^*, x_j^*), \max_{i \in S_{r-1}} \varphi_{ir}(x_i^*, x_r) \right\} \geq$$

$$\geq \max_{i,j \in S_{r-1}} \varphi_{ij}(x_i^*, x_j^*) = \varphi_{r-1}(\bar{x}_{r-1}^*). \tag{30}$$

Следующее за ним неравенство $\varphi_{r-1}(\bar{x}_{r-1}^*) \geq \max_{\bar{x}_{r-1} \in Sol_{r-1}} \varphi_{r-1}(\bar{x}_{r-1})$ справедливо по определению 1, поскольку $(\bar{x}_{r-1}^*, x_r) \notin Sol_{r-1} \times K$, то $\bar{x}_{r-1}^* \notin Sol_{r-1}$. Последнее неравенство в цепочке (29) следует из ранее доказанной цепочки (28). Таким образом,

$$\varphi_r(\bar{x}_{r-1}^*, x_r) \geq \max_{(\bar{x}_{r-1}, x_r) \in Sol_r^*} \varphi_r(\bar{x}_{r-1}, x_r)$$

для любой разметки $(\bar{x}_{r-1}^*, x_r) \notin Sol_{r-1} \times K$, откуда в силу (3) следует, что

$$Sol_r^* = \underset{(\bar{x}_{r-1}, x_r) \in K^{S_r}}{\arg\min}_{[d]} \varphi_r(\bar{x}_{r-1}, x_r). \ \blacksquare$$

Из леммы 9 следует алгоритм, который проверяет условие (27) при каждом пересчете Sol_{r-1} в Sol_r по формулам (24) и (25). Если это условие выполняется для каждого $r = 3, 4, \ldots, n$, то результат Sol_n, полученный в конце работы алгоритма, гарантированно правильный, т.е. $Sol_n = \underset{\bar{x} \in K^T}{\arg\min}_{[d]} \varphi_n(\bar{x})$ вне зависимости от существования или отсутствия мажоритарного полиморфизма, а также вне зависимости от согласованности или несогласованности решаемой задачи. Иными словами, множество задач, для которых условие (27) выполняется, образует подобласть компетентности этого алгоритма. Все согласованные задачи входят в эту подобласть, потому что в таких задачах равенство (27) выполняется для любой разметки $\bar{x} \in K^{S_{r-1}}$, а не только для $\bar{x} \in Sol_{r-1}$. В свою очередь, любую задачу, для которой существует мажоритарный полиморфизм, пусть и неизвестный, можно свести к согласованной.

Сложность проверки условия (27) при r-м пересчете Sol_{r-1} в Sol_r равна $O(d \times |K| \times |T|^2)$. Однако если учесть, что величины $\varphi_r(\bar{x}, y)$, $\bar{x} \in Sol_{r-1}$, $y \in K$, уже вычислены в процессе самого пересчета Sol_{r-1} в Sol_r, то сложность дополнительных вычислений, необходимых для контроля (27), равна $O(d \times |K|)$. Контроль условия (27) на всех $|T|$ пересчетах Sol_{r-1} в Sol_r увеличивает сложность алгоритма на величину порядка $O(d \times |K| \times |T|)$, что пренебрежительно мало по сравнению со сложностью $O(|K^3| \times |T|^3 + d \times |K| \times |T|^2)$ алгоритма в целом, т.е. вычислений по формулам (20)–(25).

Представим последовательность вычислений по формулам (20), (21), (23)–(25), (27) в виде следующего алгоритма и соответствующей теоремы. Как и прежде, в алгоритме используется обозначение $S_r = \{1 \ldots r\}$ для $r \in \{1 \ldots n\}$.

Алгоритм 1

Приведение задачи к согласованному виду:

for $t := n$ **downto** 3 **do**

for all $k \in K$ **do** $q_t(k) := \max\limits_{i \in S_{t-1}} \min\limits_{x_i \in K} \varphi_{it}(x_i, k)$ **end for** // согласно (20)

for all $(i, j, x, y) \in S_{t-1} \times S_{t-1} \times K \times K$ **do**

$$\varphi_{ij}(x, y) := \max\left\{ \varphi_{ij}(x, y), \min_{k \in K} \max\{\varphi_{it}(x, k), \varphi_{jt}(y, k), q_t(k)\} \right\} \text{ // согласно (21)}$$

end for

end for

Построение решения:

$Sol_2 := \arg\min\limits_{x, y \in K}{}_{[d]}\, \varphi_{12}(x, y)$ // согласно (23)

for $r := 3$ **to** n **do**

for all $x^* \in Sol_{r-1}$ **do**

for all $k \in K$ **do**

$$f(x^*, k) := \max\left\{ \max_{i, j \in S_{r-1}} \varphi_{ij}(x_i^*, x_j^*), \max_{i \in S_{r-1}} \varphi_{ir}(x_i^*, k) \right\} \text{ // согласно (24)}$$

end for

if $\min\limits_{k \in K} f(x^*, k) \neq \max\limits_{i, j \in S_{r-1}} \varphi_{ij}(x_i^*, x_j^*)$ **then ОТКАЗ! end if** // согласно (27)

end for

$Sol_r := \arg\min\limits_{(x^*, k) \in Sol_{r-1} \times K}{}_{[d]}\, f(x^*, k)$ // согласно (25)

end for

Обоснование алгоритма, приведенное в разд. 5 и 6, является доказательством следующей теоремы.

Теорема 2. Алгоритм 1 завершает работу за время $O(|K|^3 \times |T|^3 + d \times |K| \times |T|^2)$. Результатом является либо d наилучших разметок Sol_n, либо отказ от решения задачи. Последнее возможно только, если для входной задачи отсутствует мажоритарный полиморфизм. ■

ЗАКЛЮЧЕНИЕ

Исследована задача поиска d наилучших разметок вида $\bar{x} : T \to K$, где T и K — конечные множества, а качество $\varphi : K^T \to W$ разметок представлено в формате, принятом в теории удовлетворения ограничений. Показано, что если функция $\varphi: K^T \to W$ инвариантна относительно неоднородного мажоритарного оператора, пусть и неизвестного, то поиск d наименьших чисел из $|K^T|$ чисел сводится за полиномиальное время к $(|T|-2)$-кратному поиску d наименьших чисел из $|K| \times d$ чисел. В частности, при $d=1$ минимизация функции $|T|$ переменных сводится к $(|T|-2)$-кратной минимизации функции одной переменной, принимающей $|K|$ значений.

Это достоинство было бы существенно обесценено, если бы не было известно, как себя поведет разработанный алгоритм при подаче на его вход задачи, для которой не существует мажоритарного полиморфизма. Для этого случая его следовало бы дополнить алгоритмом входного контроля задачи, который проверяет существование или отсутствие для нее подходящего полиморфизма. Авторам не известен такой алгоритм, и, возможно, он довольно трудный. Достоинство разработанного алгоритма в том, что для него такой входной контроль не нужен. Множество задач, которые можно подавать на вход алгоритма, — это NP-трудный класс всех возможных минимаксных задач, представленных в определенном формате. Работа с любой задачей завершается либо ее решением, либо сообщением «отказ», последнее возможно только в том случае, если для решаемой задачи отсутствует мажоритарный полиморфизм.

СПИСОК ЛИТЕРАТУРЫ

1. Rossi F., van Beek P., Walsh T. Handbook of constraint programming (foundations of artificial intelligence). New York: Elsevier Science Inc., 2006. 975 p.
2. Cooper M.C. Reduction operations in fuzzy or valued constraint satisfaction. *Fuzzy Sets and Systems*. 2003. Vol. 134, N 3. P. 311–342.
3. Ruttkay Z. Fuzzy constraint satisfaction. IEEE World Congress on Computational Intelligence. *Proc. of he Third IEEE Conference. Fuzzy Systems*. 1994. Vol. 2. P. 1263–1268.
4. Lawler E.L. A procedure for computing the k best solutions to discrete optimization problems and its application to the shortest path problem. *Management Science*. 1972. Vol. 18, N 7. P. 401–405.
5. Bulatov A.A., Dalmau V., Grohe M., Marx D. Enumerating homomorphisms. *J. Comput. Syst. Sci.* 2012. Vol. 78, N 2. P. 638–650.
6. Dechter R., Flerova N., Marinescu R. Search algorithms for m best solutions for graphical models. *Proc. of the Twenty-Sixth AAAI Conference on Artificial Intelligence. AAAI'12*. AAAI Press, 2012. P. 1895–1901.
7. Eppstein D. Finding the k shortest paths. *SIAM J. Comput*. 1999. Vol. 28, N 2. P. 652–673.
8. Flerova N., Rollon E., Dechter R. Bucket and mini-bucket schemes for *M* best solutions over graphical models. Berlin; Heidelberg: Springer, 2012. P. 91–118.
9. Greco G. and Scarcello F. Structural tractability of enumerating csp solutions. *Constraints*. 2013. Vol. 18, N 1. P. 38–74.
10. Водолазский Е., Флах Б., Шлезингер М. Минимаксные задачи дискретной оптимизации, инвариантные относительно мажоритарных операторов. *ЖВММФ*. 2014. Т. 54, № 8. С. 1368–1378.
11. Cohen D., Jeavons P. The complexity of constraint languages. F. Rossi, P. van Beek, T. Walsh (Eds). *Handbook of Constraint Programming*. Ch. 6. Amsterdam: Elsevier, 2006. P. 169–204.
12. Jeavons P., Krokhin A., Zivny S. The complexity of valued constraint satisfaction. *Bulletin of EATCS*. 2014. Vol. 113. 35 p.
13. Jeavons P., Cohen D., Cooper M.C. Constraints, consistency and closure. *Artif. Intell*. 1998. Vol. 101, N 1–2. P. 251–265.
14. Bistarelli S., Montanari U., Rossi F., Schiex T., Verfaillie G., Fargier H. Semiring-based csps and valued CSPS: Frameworks, properties, and comparison. *Constraints*. 1999. Vol. 4, N 3. P. 199–240.
15. Rollon E., Flerova N., Dechter R. Inference schemes for m best solutions for soft CSPS. *Proc. of Workshop on Preferences and Soft Constraints*, 2011. P. 124–138.
16. Шлезингер М. Математические средства обработки изображений. Киев: Наук. думка, 1989. 197 с.

17. Шлезингер М., Главач В. Десять лекций по статистическому и структурному распознаванию. Киев: Наук. думка, 2004. 545 с.

18. Hubie Chen, Dalmau V. Beyond hypertree width: Decomposition methods without decompositions. Berlin; Heidelberg: Springer, 2005. P. 167–181.

19. Bertele U., Brioschi F. Nonserial dynamic programming. Orlando, FL: Academic Press, Inc., 1972. 235 p.

20. Kolmogorov V. Minimizing a sum of submodular functions. *Discrete Appl. Math*., 2012. Vol. 160, N 15. P. 2246–2258.

21. Blum M., Floyd R.W., Pratt V., Rivest R.L., Tarjan R.E. Time bounds for selection. *Journal of Computer and System Sciences*. 1973. Vol. 7, N 4. P. 448–461.

22. Schlesinger M., Flach B. Some solvable subclasses of structural recognition problems. *Czech Pattern Recognition Workshop*. 2000. P. 55–62.

Надійшла до редакції 06.10.2016

М.І. Шлезінгер, Б. Флах, Є.В. Водолазський

ПОШУК ЗАДАНОЇ КІЛЬКОСТІ РОЗВ'ЯЗКІВ СИСТЕМИ РОЗМИТИХ ОБМЕЖЕНЬ

Анотація. Досліджено мінімаксну модифікацію задачі розпізнавання несуперечності системи обмежень, коли для кожного розв'язку визначено не бінарну допустимість, а її кількісну характеристику. Описаний в статті алгоритм знаходить за поліноміальний час задану кількість найкращих розв'язків системи розмитих обмежень, якщо ці обмеження інваріантні відносно деякого мажоритарного оператора. Важливо, що для реалізації алгоритму не потрібно знання цього оператора, більш того, не потрібно гарантувати його існування. Для довільної системи розмитих обмежень алгоритм або знаходить задану кількість найбільш допустимих розв'язків, або дає відмову від розв'язку задачі. Останнє можливо, тільки якщо для розв'язаної системи обмежень такого оператора не існує.

Ключові слова: дискретна оптимізація, мінімаксні задачі, розмітки, інваріанти, поліморфізми.

M.I. Schlesinger, B. Flach, E. Vodolazskiy

FINDING A GIVEN NUMBER OF SOLUTIONS TO A SYSTEM OF FUZZY CONSTRAINTS

Abstract. A minimax modification of a fuzzy constraint satisfaction problem is considered, where constraints determine not whether a given solution is feasible but the numerical value of satisfiability. The algorithm is proposed that finds a given number of solutions with the highest value of satisfiability in polynomial time for a subclass of problems with constraints invariant to some majority operator. It is essential that knowing the operator itself is not required. Moreover, it is not necessary to guarantee its existence. For any system of fuzzy constraints, the algorithm either finds a given number of best solutions or declines the problem. The latter is only possible when no such operator exists.

Keywords: discrete optimization, minimax problems, labeling, invariants, polymorphisms.

Шлезингер Михаил Иванович,
доктор физ.-мат. наук, профессор, главный научный сотрудник Международного научно-учебного центра информационных технологий и систем НАН Украины и МОН Украины, Киев,
e-mail: schles@irtc.org.ua.

Флах Борис,
доктор наук, доцент, Центр машинного зрения, Чешский технический университет, Прага,
e-mail: flachbor@cmp.felk.cvut.cz.

Водолазский Евгений Валериевич,
научный сотрудник Международного научно-учебного центра информационных технологий и систем НАН Украины и МОН Украины, Киев, e-mail: waterlaz@gmail.com.