

1. Массовая задача.

Формально массовая задача Π определяется:

- 1) общим списком всех параметров задачи (свободных параметров, значения которых не заданы),
- 2) формулировкой свойств, которым должен удовлетворять ответ (решение задачи).

2. Индивидуальная задача.

Индивидуальная задача $I \in \Pi$ получается из Π , если всем параметрам присвоить конкретные значения.

3. Алгоритм решения задачи.

Будем говорить, что алгоритм A решает массовую задачу Π , если для $\forall I \in \Pi$ алгоритм A применим к I (т. е. останавливается за конечное число шагов) и для $\forall I \in \Pi$ алгоритм A дает решение задачи I .

4. Полиномиальные алгоритмы решения массовой задачи.

Алгоритмы, решающие произвольную $I \in \Pi$ за время, ограниченное полиномом от "размера" I . Полиномиальные задачи - задачи, для которых существуют полиномиальные алгоритмы решения.

5. Задачи распознавания свойств.

Массовые задачи, предполагающие ответ "да" или "нет" в качестве решения.

6. Кодировка задачи.

Введем конечное множество - **алфавит** $\Sigma = \{\sigma_i\}$, а также множество Σ^* **слов над алфавитом** Σ - произвольных конечных последовательностей, составленных из символов алфавита, возможно повторяющихся, $\sigma = \sigma_{i_1}\sigma_{i_2}\dots\sigma_{i_n}$, $\sigma_{i_j} \in \Sigma \forall i_j$. Число n называется **длиной слова** σ и обозначается $|\sigma|$. **Кодировкой задачи** Π назовем такое отображение $e: \Pi \rightarrow \Sigma^*$, ставящее в соответствие любой индивидуальной задаче $I \in \Pi$ ее код $e(I) = \sigma \in \Sigma^*$ (слово из алфавита Σ^*), что

- 1) Возможно однозначное декодирование: $\forall I_1 \neq I_2 \ e(I_1) \neq e(I_2)$
- 2) e, e^{-1} полиномиально вычислимы: существует алгоритм, реализующий e, e^{-1} и полином $p()$, для которого $\forall I \in \Pi$ время определения $e(I), e^{-1}(e(I))$ не превосходит $p(|e(I)|)$
- 3) Кодировка избыточна: для любой другой кодировки e' , удовлетворяющей условиям 1) и 2), найдется полином p' , такой что $\forall I \in \Pi \ |e(I)| < p'(|e'(I)|)$.

7. Временная сложность алгоритма.

Алгоритм A решает массовую задачу Π , если $L(A) = L(\Pi, e)$ и $\forall \sigma \in \Sigma^*$ A останавливается. Обозначим $t_A(\sigma)$ – время работы над словом σ (число шагов). **Временной сложностью** алгоритма A решения массовой задачи Π назовем функцию

$$T_A(n) = \max_{\sigma \in \Sigma^*: |\sigma| < n} t_A(\sigma) \quad \forall n \in Z_+$$

8. Класс полиномиально разрешимых задач (P)

$P = \{L(\Pi, e) \mid \exists A, \text{ решающий } \Pi \text{ с кодировкой } e, \exists p() \text{ – полином: } T_A(n) < p(n) \forall n \in Z_+\}$
Если для задачи Π существует такая кодировка e , что $L(\Pi, e) \in P$, то будем называть Π полиномиальной.

9. Недетерминированная машина Тьюринга (НДМТ)

Определяется как набор обычных (детерминированных) машин Тьюринга $A(S)$ с алфавитом Σ , где S пробегает все множество слов из Σ^* : $\hat{A} = \{A(S)\}_{S \in \Sigma^*}$
НДМТ останавливается, когда останавливается первая из ДМТ $A(S)$, принимающая входное слово. Соответствующее конечное состояние – q_Y .
Язык НДМТ – множество слов, принимаемых хотя бы одной ДМТ.

10. НДМТ решает массовую задачу Π с кодировкой e

НДМТ решает массовую задачу Π с кодировкой e , если $L(\Pi, e) = L(\hat{A})$
$$\left\{ \begin{array}{l} \forall \sigma \in L(\Pi, e) \exists S \in \Sigma^*: A(S) \text{ останавливается в } q_Y \\ \forall \sigma \in \Sigma^* \setminus L(\Pi, e) \forall S \in \Sigma^* A(S) \text{ не останавливается или останавливается в } q_N \end{array} \right.$$

11. Время работы НДМТ \hat{A} над словом σ

Минимальное из времен работы ДМТ $A(S)$ над словом σ с учетом времени прочтения слова S
$$\hat{t}_{\hat{A}}(\sigma) = \min_{\{S \mid \sigma \in L_{A(S)}\}} \{|S| + t_{A(S)}(\sigma)\}$$

12. Временная сложность НДМТ \hat{A} решения массовой задачи Π

Функция $T_{\hat{A}}(n) = \max_{\sigma \in \hat{L}(\hat{A}): |\sigma| < n} \hat{t}_{\hat{A}}(\sigma)$

13. Класс недетерминированно полиномиальных задач (NP)

$NP = \{L(\Pi, e) \mid \exists \hat{A} \text{ – НДМТ, решающая } \Pi \text{ с кодировкой } e, \exists p() \text{ – полином: } T_{\hat{A}}(n) < p(n) \forall n \in Z_+\}$

14. Теорема об экспоненциальной оценке временной сложности NP

$\forall \Pi \in NP \exists p() \text{ – полином, } \exists \text{ ДМТ } A: A \text{ решает } \Pi \text{ и } T_A(n) < 2^{p(n)} \forall n \in Z_+$

15. Дополнительная массовая задача

Дополнительная к Π массовая задача $\bar{\Pi}$ получается из Π распознавания свойств заменой альтернативного вопроса, определяющего ответ в задаче его отрицанием.
 $D(\bar{\Pi}) = D(\Pi)$; $Y(\bar{\Pi}) = D(\Pi) \setminus Y(\Pi)$

16. Классы co-P и co-NP

$co-P = \{\bar{\Pi} \mid \Pi \in P\}$
 $co-NP = \{\bar{\Pi} \mid \Pi \in NP\}$
Утверждение: co-P = P.

17. Задача, имеющая хорошую характеристику

Π имеет хорошую характеристику, если $\Pi \in NP \cap co-NP$, Π – распознавания свойств.

18. Полиномиальная сводимость

Π' распознавания свойств с кодировкой e' полиномиально сводится к Π с кодировкой e , если $\forall I' \in \Pi'$ может быть сведена за полиномиальное от ее длины время к некоторой $I \in \Pi$ с сохранением ответа.

19. Утверждения

Если $\Pi_1 \propto \Pi_2$ и $\Pi_2 \propto \Pi_3$, то $\Pi_1 \propto \Pi_3$

Если $\Pi_1 \propto \Pi$ и $\Pi \in P$, то $\Pi_1 \in P$

Если $\Pi' \propto \Pi$ и $\Pi \in NP$, то $\Pi' \in NP$

20. NP-полная задача

Массовая задача Π называется NP-полной (универсальной), если

$\Pi \in NP$ и $\forall \Pi' \in NP \Pi' \propto \Pi$

Класс NP-полных задач – NPC (NP-complete).

21. Задача о выполнимости (ВЫП)

Выяснить выполнимость КНФ (существует ли набор входных данных, на которых КНФ равна 1).

22. Теорема Кука

ВЫП \in NPC

23. Утверждения

Если $P \cap NPC \neq \emptyset$, то $P = NP$

Если $NPC \cap (NP \setminus P) \neq \emptyset$, то $NPC \subseteq NP \setminus P$

24. Критерий NP-полноты

Массовая задача Π NP-полна тогда и только тогда, когда она принадлежит классу NP, и к ней полиномиально сводится какая-либо NP-полная задача.

$\{\Pi \in NPC\} \Leftrightarrow \{\Pi \in NP \text{ и } \exists \Pi' \in NPC: \Pi' \propto \Pi\}$

25. Задача ЦЛН

Задача о существовании целочисленного решения системы линейных неравенств с целыми коэффициентами. Принадлежит классу NPC.

26. Задача БЛН

Задача о существовании булева решения системы линейных неравенств с целыми коэффициентами. Принадлежит классу NPC.

27. Задача 3-ВЫП

Частный случай ВЫП, когда функции от 3-х переменных.

28. Утверждение

ВЫП \propto 3 – ВЫП

29. Утверждение

Если для некоторой NP-полной задачи Π дополнительная к ней принадлежит классу NP, то $NP = co-NP$.

30. Класс NP-трудных задач

Класс NP-трудных задач объемлет NPC. Он содержит:

- 1) Π распознавания свойств, для которых доказано, что $\Pi' \propto \Pi$ для $\Pi' \in NPC$, но не доказано, что $\Pi \in NP$
- 2) Π оптимизации, для которых соответствующие Π распознавания свойств NP-полны
- 3) Остальные массовые задачи, к которым сводятся по Тьюрингу какие-либо NP-полные задачи.

31. NP-эквивалентные задачи

Π , для которых $\exists \Pi' \in NPC: \Pi' \propto_T \Pi$ и $\exists \Pi'' \in NP: \Pi \propto_T \Pi''$

32. Класс PSPACE

Класс задач, для решения которых существуют алгоритмы, требующие не более чем полиномиальной памяти.

33. Псевдополиномиальный алгоритм

Обозначим:

$\text{num}(I)$ – максимальное по модулю число (или 0), фигурирующее при задании числовых параметров индивидуальной задачи I

$|I| = |e(I)|$ – длину записи I .

Алгоритм A решения массовой задачи Π называется **псевдополиномиальным**, если для некоторого полинома $p()$ выполнено $t_A(e(I)) < p(|I|, \text{num}(I)) \quad \forall I \in \Pi$

34. Полиномиальное сужение массовой задачи

Множество индивидуальных задач, числовые параметры которых не превосходят полинома от длины входа.

35. Сильно NP-полная задача

Массовая задача Π распознавания свойств называется **сильно NP-полной**, если ее полиномиальное сужение NP-полно.

36. Теорема

Если $P \neq NP$, то ни для какой сильно NP-полной задачи не существует псевдополиномиального алгоритма.

37. Задачи дискретной (комбинаторной) оптимизации

Для оптимизационной постановки задачи Π решением каждой $I \in \Pi$ является

произвольная оптимизация $\text{Opt}_{\Pi}(I) = \max_{z \in S_{\Pi}(I)} f_{\Pi}(I, z)$

S_{Π} – область допустимых значений дискретной (целочисленной) переменной z .

$f_{\Pi}(I, z): S_{\Pi} \rightarrow Z$ – целевая функция задачи I .

38. Приближенный алгоритм решения

Алгоритм A называется **приближенным алгоритмом** решения массовой задачи Π оптимизации, если $\forall I \in \Pi$ он находит некоторую точку из допустимой области $z_A(I) \in S_\Pi(I)$, принимаемую за приближенное решение. Значение $f_\Pi(I, z_A(I))$ называется приближенным значением оптимума и обозначается $A(I)$.

39. Утверждение о погрешности

Если $P \neq NP$, то ни для какой константы $C > 0$ не существует полиномиального приближенного алгоритма A решения задачи о рюкзаке $3P$ с оценкой $|Opt_{3P}(I) - A(I)| \leq C \quad \forall I \in 3P$

40. ε -приближенный алгоритм

Приближенный алгоритм A решения массовой задачи Π называется ε -приближенным для некоторого $\varepsilon > 0$, если $\frac{|Opt_\Pi(I) - A(I)|}{|Opt_\Pi(I)|} < \varepsilon$

41. Теорема

Пусть для задачи Π оптимизации

1) Существует псевдополиномиальный алгоритм ее решения

2) $\forall I \in \Pi \quad |Opt_\Pi(I)| < p_1(|I|, num(I))$ и $num(I) < p_2(|I|, Opt_\Pi(I))$ для некоторых полиномов p_1 и p_2

3) $\forall \sigma = e(I), I \in \Pi$, параметры σ_S , задающие ограничения, и σ_f , задающие целевую функцию, не пересекающиеся и $\forall z \in S_\Pi(\sigma)$ функция цели $f_\Pi(\sigma, z)$ линейно зависит от параметров σ_f . Тогда

$\exists p()$ – полином: $\forall \varepsilon > 0 \exists \varepsilon$ -приближенный алгоритм A_ε решения Π с временной сложностью $T_{A_\varepsilon}(|I|) < p(|I|, 1/\varepsilon)$

42. Полностью полиномиальная приближенная схема (ПППС)

Набор алгоритмов из теоремы в пункте 41.

43. Теорема

Если для Π оптимизации соответствующая ей Π распознавания свойств является сильно NP -полной, и $\exists p'()$ – полином: $|Opt_\Pi(I)| < p'(num(I)) \quad \forall I \in \Pi$, то, при условии что $P \neq NP$, для Π не существует ПППС.

44. Утверждение

Если $P \neq NP$, то ни для какого $\varepsilon > 0$ не существует полиномиального ε -приближенного алгоритма решения оптимизационной постановки задачи коммивояжера.

45. Основная задача линейного программирования (озЛП)

Состоит в нахождении такого решения системы ЛН $Ax \leq b$, которое максимизирует целевую функцию.

46. Каноническая задача ЛП

$$\max_{Ax=b, x \geq \bar{0}} \langle c, x \rangle$$

47. Принцип граничных решений

Если задача $\max_{x \in R^n: Ax \leq b} \langle c, x \rangle$ имеет решение, то найдется такая подматрица A_1 матрицы A , что любое решение системы уравнений $A_1 x = b_1$ реализует максимум в $\max_{x \in R^n: Ax \leq b} \langle c, x \rangle$

48. Симплекс-метод

Метод направленного перебора смежных вершин в направлении возрастания целевой функции.

49. Функции алгебраической и битовой сложности

Функция алгебраической сложности: оценивает число арифметических операций в зависимости от размерности, не учитывает длину кода элементов.

Функция битовой сложности: оценивает число арифметических операций с битами (или конечными порциями – по размеру машинного регистра) цифровой записи параметров задачи в зависимости от длины входного слова.

50. Сильнополиномиальный метод

Метод, имеющий полиномиальную алгебраическую сложность.

51. Теорема о границах решений

Для любой целочисленной матрицы D введем параметр

$$\Delta D = \max_{\{D' - \text{квадратная подматрица } D\}} |\det D'|$$

$[A|b]$ – матрица, полученная приписыванием справа к A столбца b .

Теорема

Если озЛП размерности (n, m) с целыми коэффициентами разрешима, то у нее существует рациональное решение x^* в шаре $\|x\| \leq n^{1/2} \Delta([A|b])$ и значением озЛП $d^* = \langle c, x^* \rangle$ является рациональное число t/s со знаменателем, ограниченным величиной $\Delta(A)$.

52. ε -приближенное решение системы ЛН

Точка x^ε называется ε -приближенным решением системы неравенств, если $\langle a_i, x^\varepsilon \rangle \leq b_i + \varepsilon \quad \forall i = \overline{1, m}$, где a_i – i -ая строка матрицы A .

53. Теорема о мере несоместности

Если ЛН имеет ε_1 -приближенное решение для $\varepsilon_1 = 1/[(n+2)\Delta(A)]$, то эта система разрешима, то есть имеет точное решение.

54. ε -приближенное решение озЛП

Точка x_ε^* называется ε -приближенным решением озЛП, если она является ε -приближенным решением системы ЛН и реализует максимум с ε -точностью: $\langle a_i, x_\varepsilon^* \rangle \leq b_i + \varepsilon; \langle c, x_\varepsilon^* \rangle \geq d_i - \varepsilon$

55. Теорема о мере несоместности*

Если озЛП имеет ε_2 -приближенное решение для $\varepsilon_2 = 1/(2n^2 \Delta^3(A))$, то эта задача имеет точное решение.

56. Следствие разрешимой системы ЛН

Линейное неравенство $\langle c, x \rangle \leq d$ является следствием разрешимой системы ЛН, если для $\forall x$, удовлетворяющего системе, выполнено $\langle c, x \rangle \leq d$.

57. Лемма Фаркаша (аффинная)

Линейное неравенство $\langle c, x \rangle \leq d$ является следствием разрешимой в вещественных переменных системы ЛН тогда и только тогда, когда существует вектор $\lambda \in R^m$:

$$c = \sum_{i \in M} \lambda_i a_i; \quad d \geq \sum_{i \in M} \lambda_i b_i; \quad \lambda_i \geq 0 \quad \forall i \in M$$

58. Лемма Фаркаша о неразрешимости

Система ЛН $Ax \leq b$ неразрешима тогда и только тогда, когда разрешима система

$$\sum_{i \in M} \lambda_i a_i = \bar{0}; \quad \sum_{i \in M} \lambda_i b_i \leq -1; \quad \lambda_i \geq 0 \quad \forall i \in M$$

59. Двойственная задача

Двойственной к задаче ЛП на максимум с ограничениями неравенствами в форме оЗЛП называется следующая задача ЛП с ограничениями в канонической форме:

$$\min \{ \sum_{i \in M} \lambda_i b_i \mid \sum_{i \in M} \lambda_i a_i = c; \quad \lambda_i \geq 0 \quad \forall i \in M \}$$
 или кратко $\min_{\lambda A=c, \lambda \geq \bar{0}} \langle \lambda, b \rangle$

60. Теорема о двойственности ЛП

Задача ЛП разрешима тогда и только тогда, когда разрешима двойственная к ней. В случае разрешимости оптимальные значения целевых функций совпадают.

61. Утверждение

Задача ЛП оптимизации эквивалентна решению системы ЛН.

62. Утверждение

Задача ЛП эквивалентна решению системы линейных уравнений в неотрицательных переменных.

63. Утверждение

Задача ЛП эквивалентна поиску неотрицательного ненулевого решения однородной системы линейных уравнений.

64. Метод Крамакара

65. Классификация задач МП

ЗМП – задачи с вещественными переменными.

1) Условные задачи $X \subset R^n$

2) Безусловные $X = R^n$

По свойствам целевой функции ЗМП делятся на

1) Задачи ЛП

2) Задачи выпуклого программирования

3) Гладкие и негладкие и др.

С точки зрения численных методов

Локальная и глобальная оптимизация

66. Точка локального минимума в ЗМП

$x^0 \in X$ называется точкой локального минимума в ЗМП, если
 $\exists \varepsilon > 0: f(x^0) \leq f(x) \quad \forall x \in X \cap O_\varepsilon(x^0)$

67. Утверждение

ЦЛН \propto ЗМП

68. Выпуклая функция

Функция $f(x)$ называется выпуклой на X , если ее надграфик $epigraph_x f = \{(x, y) | y \geq f(x), x \in X\}$ – выпуклое множество. Функция называется выпуклой, если она выпукла на всей области определения. Множество называется выпуклым, если для любых двух своих точек оно содержит отрезок, их соединяющий.

69. Утверждение

Любая точка локального минимума выпуклой функции является точкой ее глобального минимума.

70. Преимущества выпуклого случая

- 1) Задача поиска ε -приближенного решения задачи выпуклого программирования полиномиально разрешима
- 2) Для острых задач – когда функция цели убывает в окрестности минимума не медленнее некоторой линейной функции, можно найти точное решение.

71. Направление убывания функции

Вектор $h \in R^n$ называется вектором убывания f в точке x , если $f(x + ah) < f(x)$ для \forall достаточно малых $a > 0$

72. Утверждение

Пусть f дифференцируема в X . Тогда, если $\langle \text{grad} f(x), h \rangle < 0$, то h – направление убывания f в точке x . Если h – направление убывания f в точке x , то $\langle \text{grad} f(x), h \rangle \leq 0$

73. Градиентный метод оптимизации

$$x^{t+1} = x^t - \alpha_t \text{grad} f(x^t) \quad \forall x^1 \in R^n$$

Выбор α_t :

- 1) Пассивные способы – выбираются заранее
- 2) Адаптивные способы – зависит от реализующейся итерации (метод скорейшего спуска, метод дробления шага, правило Армихо)

74. Метод проекции градиента

$$x^{t+1} = Pr_X \{x^t - \alpha_t \text{grad} f(x^t)\} \quad \forall x^1 \in R^n$$

75. Дифференцируемая по Адамару функция

Функция называется дифференцируемой по Адамару в точке $x \in R^n$, если существует вектор $\nabla f(x) \in R^n$, такой что $\forall y \in R^n$ выполнено

$$\lim_{(\tau, y') \rightarrow (+0, y)} \frac{f(x + \tau y') - f(x)}{\tau} = \langle \nabla f(x), y \rangle$$

76. Контингентный конус

Контингентным конусом ко множеству X в точке x называется множество векторов $K(X, x) = \{y \mid \exists \{(\tau_t, y^t)\}_{t=1}^{\infty}: (\tau_t, y^t) \rightarrow (+0, y); x + \tau_t y^t \in X \forall t\}$

77. Общий вид необходимых условий локального минимума

Пусть f дифференцируема по Адамару, $X \subset R^n, X \neq \emptyset, x^0$ – точка локального минимума f . Тогда $\forall y \in K(X, x^0) \langle \nabla f(x^0), y \rangle \geq 0$

78. Регулярное множество

$J(x) = \{i \in M \mid g_i(x) = 0\}$ – множество активных ограничений.

$G(x) = \{y \in R^n \mid \langle \nabla g_j(x), y \rangle \leq 0 \quad \forall j \in J(x)\}$

Множество X для ограничений неравенств называется **регулярным** в точке $x \in X$, если $G(x) \subseteq K(X, x)$

79. Необходимые условия локального минимума с ограничениями-неравенствами

Пусть $f, g_i \quad \forall i \in M$ дифференцируемы по Адамару, $X \neq \emptyset, x^0$ – точка локального минимума f и X регулярно в x^0 . Тогда $\exists \lambda_j \geq 0: \nabla \{f(x^0) + \sum_{j \in J(x^0)} \lambda_j g_j(x^0)\} = 0$

80. Принцип оптимальности Лагранжа

В предположениях теоремы из пункта 79 существует неотрицательный вектор множителей Лагранжа $\lambda \geq 0$, такой что для x^0 выполнены условия оптимальности $\nabla_x L(x^0, \lambda) = \bar{0}$

81. Теорема Куна-Таккера

Если функции $f, g_i \in C^1(R^n)$ выпуклы и множество X регулярно в любой точке, то x^* – точка оптимума в этой задаче тогда и только тогда, когда в ней выполнены условия оптимальности для $\lambda \geq \bar{0}$

82. Вполне унимодулярная матрица

Матрица называется унимодулярной, если определитель любой ее невырожденной квадратной подматрицы равен по модулю 1.