

РАЗРАБОТКА МЕТОДОВ ОПТИМИЗАЦИИ ДЛЯ ПОИСКА НАИЛУЧШЕЙ КОМБИНАЦИИ ПОКАЗАТЕЛЕЙ ПРОЧНОСТИ И ВЯЗКОСТИ СПЛАВОВ¹

Аннотация. Предложены математические методы для оптимизации химического состава сплавов и технологических параметров их изготовления в процессе разработки новых марок сталей в условиях значительной неопределенности. В оптимизационных задачах использованы статистические модели механических свойств новых сплавов. Для оценки правого хвоста распределения вероятностей показателя разрушения по методу Шарпи с V-образным надрезом (CVN) использован CVaR, который является выпуклой функцией. Предложены математические методы решения задач максимизации.

Ключевые слова: сплав, оптимизация, CVaR, квантиль, CVN.

ВВЕДЕНИЕ

Разработка новых сплавов — длительный и дорогостоящий процесс, который осуществляется в условиях значительной неопределенности. Его можно удешевить и ускорить, используя математические методы оптимизации и статистического моделирования, разработанные в [1–3]. В этих работах рассмотрены две механические характеристики сталей: предел текучести при растяжении (yield) и показатель разрушения по методу Шарпи с V-образным надрезом (CVN), а также показано, что экспериментальные данные по yield достаточно точно аппроксимируются обычной линейной регрессионной моделью. В то же время при моделировании CVN с помощью линейной регрессии достичь приемлемой точности не удастся, поскольку экспериментальные данные по CVN характеризуются существенным разбросом. Причины возникновения такой ситуации детально проанализированы в [4, 5]. Для преодоления описанного затруднения в [2] предложен более гибкий подход, с помощью которого можно оценивать любой квантиль распределения CVN. В качестве приемлемого инструментария выбран метод из [6], позволяющий строить линейные регрессии, прогнозирующие любые квантили функции распределения CVN. С использованием данных линейных моделей в [1] сформулированы задачи оптимизации механических характеристик сталей в виде задач линейного программирования.

В теории финансового риска квантиль, обычно называемый Value-at-Risk (VaR), ранее применялся для оценки хвоста распределений. В последнее десятилетие вместо показателя VaR чаще используется показатель Conditional Value-at-Risk (CVaR). Мера риска VaR оценивает только нижнюю границу правого хвоста распределения, а не его интегральную характеристику. В отличие от VaR мера риска CVaR создана именно для оценки последней, поскольку равна усредненному значению всех VaR, соответствующих правому хвосту распределения.

В работе [3] модифицирован подход к моделированию CVN, разработанный в [2]. В новом подходе для оценки правого хвоста распределения CVN используется CVaR (вместо VaR). Для этого построена стохастическая модель для логарифма

¹Работа выполнена при финансовой поддержке Европейского офиса аэрокосмических исследований (European Office of Aerospace Research and Development), грант P-590/EOARD 133063.

рифма показателя CVN в виде сценарной функции, представляющая собой линейную случайную функцию независимых переменных (химический состав сплава и параметры его технологической обработки) со случайными коэффициентами, которые имеют совместное конечномерное распределение [3]. Область значений этой функции для любой фиксированной точки в пространстве входных переменных состоит из набора сценариев значений логарифма показателя CVN, соответствующих этой точке.

Поскольку сценарная функция является линейной относительно независимых переменных, CVaR распределения значений логарифма показателя CVN — выпуклая (невогнутая) функция [7]. Как правило, любая выпуклая функция имеет много локальных максимумов на выпуклом допустимом множестве. Поэтому новую модель CVN нельзя использовать в задаче линейного программирования из [1].

В данной статье предложены методы поиска глобального максимума функции CVaR. Представлены две оптимизационные задачи: первая — максимизации невогнутой функции CVaR распределения значений логарифма показателя CVN при ограничениях, задающих верхнюю и нижнюю границы области определения переменных задачи, а вторая — максимизации CVaR распределения значений логарифма показателя CVN при ограничениях на yield и ограничениях, задающих верхнюю и нижнюю границы области определения переменных оптимизации. Для решения первой оптимизационной задачи предложен алгоритм выбора множества начальных точек в целях повышения эффективности работы метода мултистарта, представляющего собой семейство оптимизационных методов, которые сводят поиск глобального максимума многоэкстремальной функции к поиску нескольких локальных максимумов. Из полученных точек локальных максимумов выбирается та, у которой наибольшее значение целевой функции [13]. Эффективность этого метода в значительной степени зависит от выбора множества начальных точек, с которых осуществляется поиск локальных максимумов с использованием методов локальной оптимизации.

Для решения второй оптимизационной задачи предложен точный детерминированный метод поиска глобального максимума, основанный на утверждении, что глобальный максимум выпуклой функции на непустом замкнутом ограниченном выпуклом допустимом множестве достигается на некоторой экстремальной (крайней) точке этого множества [8]. С помощью этого метода задача решается в два этапа: на первом формируется конечное множество всех допустимых крайних точек допустимого множества, а на втором — находится глобальный максимум на сформированном множестве. На первом этапе с использованием только простых аналитических формул из множества всех базисных решений отсеивают недопустимые [10], при этом значения целевой функции не вычисляют. Предложенный метод можно применять для оптимизации химического состава сплавов и технологических параметров их изготовления, поскольку в этих задачах число параметров, как правило, не превышает 15. При большей размерности задачи данный метод можно легко адаптировать средствами параллельного программирования.

ПОСТАНОВКА ЗАДАЧИ

База данных, используемая для моделирования механических свойств новых марок сплавов, включает их химический состав и параметры термической обработки (входные параметры), а также результаты испытаний тестовых образцов сталей (выходные переменные). В рассматриваемом случае в базе данных содержится информация о $N = 34$ образцах стали: три значения показателя CVN и одно значение показателя yield для каждого образца. Количество входных параметров $I = 10$.

Пусть $\mathbf{x} = (x_1, x_2, \dots, x_I)$ — вектор входных параметров (независимых переменных), y — значение одного из выходных параметров: CVN или yield (зависимая переменная). Каждый образец нумеруется индексом n и характеризуется вектором входных параметров $\mathbf{x}^{(n)} = (x_1^{(n)}, x_2^{(n)}, \dots, x_I^{(n)})$, $n = 1, \dots, N$.

Экспериментальные данные использовались для построения статистической модели CVN и модели yield. При этом коллинеарные переменные не рассматривались, поскольку они приносят в модель одинаковую информацию и создают проблемы при оценке коэффициентов регрессии (оценки последних могут быть нестабильными).

В работе [3] разработана следующая линейная стохастическая модель для $\ln(\text{CVN}(\mathbf{x}))$:

$$\ln(\text{CVN}(\mathbf{x})) = \theta_0 + \sum_{i=1}^I \theta_i x_i, \quad (1)$$

где $\theta_0, \theta_1, \dots, \theta_I$ — случайные коэффициенты, принимающие J равновероятных значений (сценариев) $\theta_{j0}, \theta_{j1}, \dots, \theta_{jI}$, $j = 1, \dots, J$, причем некоторые из них могут быть нулевыми. Пусть $\boldsymbol{\theta} = (\theta_0, \theta_1, \dots, \theta_I)$ — случайный вектор коэффициентов линейной регрессии (1) с равновероятными сценариями $\boldsymbol{\theta}_j = (\theta_{j0}, \theta_{j1}, \dots, \theta_{jI})$, $j = 1, \dots, J$.

В работе [2] разработана следующая линейная регрессионная модель для yield:

$$\text{yield} = a_0^Y + \sum_{i=1}^I a_i^Y x_i, \quad (2)$$

где a_0^Y, a_i^Y , $i = 1, \dots, I$, — коэффициенты стандартной линейной регрессии.

Рассмотрим две постановки задачи оптимизации химического состава сплавов и технологических параметров их изготовления.

Задача 1 (простейшая постановка). Максимизировать CVaR распределения значений логарифма показателя CVN

$$\max_x \text{CVaR}_\alpha(\ln(\text{CVN}(\mathbf{x}))) \quad (3)$$

при ограничениях, задающих верхнюю и нижнюю границы области определения переменных оптимизации

$$l_i \leq x_i \leq u_i, \quad i = 1, 2, \dots, I. \quad (4)$$

В формуле (3) α — уровень доверия, $0 < \alpha < 1$.

В качестве нижней границы l_i для переменной x_i можно выбрать наименьшее значение i -го параметра в базе данных: $l_i = \min_{1 \leq n \leq N} x_i^{(n)}$, $i = 1, 2, \dots, I$.

Аналогично в качестве верхней границы u_i для переменной x_i можно выбрать наибольшее значение i -го параметра в базе данных: $u_i = \max_n x_i^{(n)}$, $i = 1, 2, \dots, I$.

Следующая оптимизационная задача позволяет отыскать наилучшую комбинацию показателей прочности и вязкости сплавов.

Задача 2. Максимизировать CVaR распределения значений логарифма показателя CVN

$$\max_x \text{CVaR}_\alpha(\ln(\text{CVN}(\mathbf{x}))) \quad (5)$$

при ограничениях на yield, определяемых по формуле (2),

$$a_0^Y + \sum_{i=1}^I a_i^Y x_i \geq b_Y, \quad (6)$$

а также ограничениях, задающих верхнюю и нижнюю границы области определения переменных оптимизации

$$l_i \leq x_i \leq u_i, \quad i=1, 2, \dots, I. \quad (7)$$

В формуле (6) b_Y — нижняя граница для yield.

Замечание 1. Поскольку $\ln(y)$ — возрастающая функция от y , максимизация $\text{CVaR}_\alpha(\ln(\text{CVN}(\mathbf{x})))$ в оптимизационных задачах 1 и 2 эквивалентна максимизации $\text{CVaR}_\alpha(\text{CVN}(\mathbf{x}))$.

Замечание 2. Ограничения в оптимизационных задачах 1 и 2 определяют выпуклые допустимые множества.

Далее приведен алгоритм выбора множества начальных точек в целях повышения эффективности работы метода мултистарта.

МАКСИМИЗАЦИЯ CVaR В ПОСТАНОВКЕ ЗАДАЧИ 1

В соответствии с (1) $\ln(\text{CVN}(\mathbf{x}))$ — линейная функция по \mathbf{x} . Следовательно, $\text{CVaR}_\alpha(\ln(\text{CVN}(\mathbf{x})))$ — выпуклая функция по \mathbf{x} [7].

Как правило, выпуклая функция имеет на выпуклом множестве много локальных максимумов. Одним из наиболее известных методов оптимизации многоэкстремальных функций является метод мултистарта — выбор некоторого множества начальных точек, с которых осуществляется поиск локальных экстремумов с использованием методов локальной оптимизации. Эффективность метода мултистарта существенно зависит от выбора начальных точек.

В настоящей работе предлагается выбирать начальные точки в подмножествах допустимого множества (4), в которых значение CVaR распределения значений логарифма показателя CVN существенно превышает значение CVaR на всем допустимом множестве. Данный подход основан на идеях, изложенных в [9].

Сгенерируем расширенную базу данных, используя формулу (1), для имитационного моделирования.

Для каждого образца n , характеризующегося вектором входных параметров $\mathbf{x}^{(n)} = (x_1^{(n)}, x_2^{(n)}, \dots, x_I^{(n)})$, $n=1, \dots, N$, и каждого сценария θ_j , $j=1, \dots, J$, случайного вектора $\boldsymbol{\theta}$ коэффициентов линейной регрессии (1) вычисляем соответствующий сценарий CVN_j случайного показателя CVN по формуле:

$$\text{CVN}_j(\mathbf{x}^{(n)}) = e^{\theta_{j0} + \sum_{i=1}^I \theta_{ji} x_i^{(n)}}, \quad j=1, \dots, J, \quad n=1, \dots, N. \quad (8)$$

В итоге генерируется база данных, содержащая точки наблюдения $\mathbf{x}^{(n)} = (x_1^{(n)}, x_2^{(n)}, \dots, x_I^{(n)})$, $n=1, \dots, N$, и результаты наблюдения (сценарии) $\text{CVN}_j(\mathbf{x}^{(n)})$ случайной величины CVN. Сгенерированные данные можно представить в виде следующей расширенной дизайн-матрицы (таблицы):

$$\begin{pmatrix}
 \text{CVN}_1(\mathbf{x}^{(1)}) & x_1^{(1)} & x_2^{(1)} & \dots & x_I^{(1)} \\
 \text{CVN}_2(\mathbf{x}^{(1)}) & x_1^{(1)} & x_2^{(1)} & \dots & x_I^{(1)} \\
 \dots & \dots & \dots & \dots & \dots \\
 \text{CVN}_J(\mathbf{x}^{(1)}) & x_1^{(1)} & x_2^{(1)} & \dots & x_I^{(1)} \\
 \dots & \dots & \dots & \dots & \dots \\
 \text{CVN}_1(\mathbf{x}^{(2)}) & x_1^{(2)} & x_2^{(2)} & \dots & x_I^{(2)} \\
 \dots & \dots & \dots & \dots & \dots \\
 \text{CVN}_J(\mathbf{x}^{(2)}) & x_1^{(2)} & x_2^{(2)} & \dots & x_I^{(2)} \\
 \dots & \dots & \dots & \dots & \dots \\
 \text{CVN}_1(\mathbf{x}^{(N)}) & x_1^{(N)} & x_2^{(N)} & \dots & x_I^{(N)} \\
 \dots & \dots & \dots & \dots & \dots \\
 \text{CVN}_J(\mathbf{x}^{(N)}) & x_1^{(N)} & x_2^{(N)} & \dots & x_I^{(N)}
 \end{pmatrix}. \quad (9)$$

В первом столбце (9) представлены сгенерированные случайные значения CVN, а в остальных — координаты исходных точек наблюдения, расположенных в начальном гиперпараллелепипеде в пространстве входных данных, который определяется ограничениями (4). При некоторых значениях индексов j и i сценарий θ_{ji} для случайного коэффициента θ_i в формуле (1) может быть нулевым, т.е. значение переменной $x_i^{(n)}$ в формуле (8) не влияет на значение соответствующего сценария $\text{CVN}_j(\mathbf{x}^{(n)})$. Для корректной работы алгоритма в строках матрицы (9), соответствующих нулевым значениям сценариев в линейной функции для $\text{CVN}_j(\mathbf{x}^{(n)})$, $n = 1, \dots, N$, фактическое значение компоненты $x_i^{(n)}$ заменяем специальным набором символов «п/а».

Определяем CVaR распределения CVN на этом гиперпараллелепипеде, рассматривая значения CVN в сгенерированной базе данных как «объединенную выборку».

Цель предлагаемого подхода состоит в поиске подмножеств исходного допустимого множества (4), представленных простыми ограничениями

$$\tilde{l}_i \leq x_i \leq \tilde{u}_i, \quad i = 1, 2, \dots, I, \quad (10)$$

на которых значение CVaR распределения CVN существенно превышает значение CVaR распределения CVN на исходном допустимом множестве. В ограничениях (10) предполагается выполнение неравенств $l_i < \tilde{l}_i < \tilde{u}_i < u_i$, $i = 1, 2, \dots, I$. Такие подмножества будем называть оптимальными боксами.

Пусть \mathbf{Y} — вектор сгенерированных наблюдений показателя CVN (первый столбец матрицы (9)), Q — количество наблюдений (число строк в матрице (9)), $\mathbf{T}_i = \{x_i^1, \dots, x_i^Q\}$ — множество всех значений (включая «п/а»), расположенных в $i+1$ -м столбце матрицы (9), $i = 1, \dots, I$, $\mathbf{T} = \mathbf{T}_1 \times \mathbf{T}_2 \times \dots \times \mathbf{T}_I$ (далее таблица \mathbf{T}). Данные в таблице \mathbf{T} рассматриваются как категориальные. Обозначим y_q — сгенерированное наблюдение показателя CVN, которое соответствует q -й строке таблицы \mathbf{T} .

Пусть α , $0 < \alpha < 1$, — уровень доверия и β , $0 < \beta < 1$, — параметр алгоритма. Положим $\mathbf{T}^0 = \mathbf{T}$. Необходимо найти подтаблицу $G \subset \mathbf{T}^0$ такую, что

$$\text{CVaR}_\alpha(y(x)|x \in G) \gg \text{CVaR}_\alpha(y(x)|x \in \mathbf{T}^0). \quad (11)$$

Предлагается следующий эвристический алгоритм решения этой задачи.

Шаг 0. Положим $k = 0$ и $Q_0 = Q$. Отсортируем строки в Y и T^k в порядке убывания значений в Y , а затем вычислим $\text{CVaR}_\alpha(y(x)|x \in T^k)$.

Шаг 1. Положим $i = 1$.

Шаг 2. Формируем новую таблицу $T_{x_i}^k = T^k$, а также множество $Z_{T_{x_i}^k}$ всех категорий, которые встречаются в $[\beta \cdot Q_k]$ верхних строках в столбце x_i таблицы $T_{x_i}^k$. Затем исключаем из таблицы $T_{x_i}^k$ строки, которые не содержат категорий, включенных в множество $Z_{T_{x_i}^k}$. Вычисляем $\text{CVaR}_\alpha(y(x)|x \in T_{x_i}^k)$.

Шаг 3. Если $i < I$, то полагаем $i = i + 1$ и переходим к шагу 2.

Шаг 4. Находим индекс i^* такой, что

$$\text{CVaR}_\alpha(y(x)|x \in T_{x_{i^*}}^k) = \max_{1 \leq i \leq I} \text{CVaR}_\alpha(y(x)|x \in T_{x_i}^k). \quad (12)$$

Формируем таблицу T^{k+1} , включив в нее лишь строки из таблицы T^k , которые в столбце x_{i^*} содержат только категории из множества $Z_{T_{x_{i^*}}^k}$. Определяем количество строк Q_{k+1} в столбце T^{k+1} .

Шаг 5. Если $Q_{k+1} < Q_k$, то полагаем $k = k + 1$ и переходим к шагу 1.

Шаг 6. Для $i = 1, \dots, I$ определяем

$$l_i^* = \min_{1 \leq q \leq Q_k} x_i^q \text{ и } u_i^* = \max_{1 \leq q \leq Q_k} x_i^q, \quad (13)$$

где x_i^q — числовое значение переменной x_i , расположенное на пересечении q -й строки и i -го столбца в конечной таблице T^k , $i = 1, \dots, I$, $q = 1, \dots, Q_k$.

Определяем оптимальный бокс

$$G = \{x = (x_1, \dots, x_I) : l_i^* \leq x_i \leq u_i^*, i = 1, \dots, I\}. \quad (14)$$

Конец работы алгоритма.

Используя формулы (12), (13), можно обеспечить выполнение неравенства (11).

При различных значениях параметра β приведенный алгоритм генерирует разные оптимальные боксы (14). Они являются источниками перспективных начальных точек при запуске алгоритмов локальной максимизации.

В табл. 1 приведены результаты численных экспериментов при $\alpha = 0.2$, здесь строки соответствуют некоторым значениям параметра β , а столбцы —

Таблица 1. Динамика роста значения $\text{CVaR}_{0.2}$ при различных значениях параметра β

Значение параметра β	Значение CVAR на k -й итерации алгоритма											
	0	1	2	3	4	5	6	7	8	9	10	11
0.0001	65.75	78.76	90.17	100.86	105.64	111.50	111.50	–	–	–	–	–
0.0005	65.75	78.76	90.17	100.86	105.64	111.50	111.50	–	–	–	–	–
0.001	65.75	78.76	89.47	98.99	103.17	107.42	111.50	111.50	–	–	–	–
0.015	65.75	79.24	90.53	97.05	101.70	105.67	105.67	–	–	–	–	–
0.02	65.75	79.11	90.54	96.44	97.42	98.01	102.46	105.67	105.67	–	–	–
0.025	65.75	79.11	88.28	93.29	96.44	97.42	98.01	98.43	98.58	102.80	105.67	105.67
0.03	65.75	79.11	88.28	93.29	96.44	97.42	98.01	98.45	98.61	98.61	–	–

параметру k (номерам итераций алгоритма). На пересечении j -й строки и k -го столбца дано значение $\text{CVaR}_{0.2}(y(x)|x \in T^k)$, полученное на k -й итерации алгоритма при значении параметра β , указанного в первом столбце, в j -й строке (см. табл.1). Анализ численных экспериментов показывает, что алгоритм строит оптимальные боксы за небольшое количество итераций. При этом наилучшие результаты получены при малых значениях параметра β .

МАКСИМИЗАЦИЯ CVAR В ПОСТАНОВКЕ ЗАДАЧИ 2

Рассмотрим задачу линейного программирования с ограниченными переменными

$$\max \sum_{j=1}^n c_j x_j \quad (15)$$

при ограничениях

$$\sum_{j=1}^n a_{ij} x_j = b_i, \quad i=1,2,\dots,m, \quad (16)$$

$$l_j \leq x_j \leq u_j, \quad j=1,\dots,n. \quad (17)$$

Этой задаче соответствуют расширенные базисные допустимые решения, в которых значения $n-m$ переменных фиксируются либо на их нижних, либо верхних границах (17), а базисной матрице \mathbf{B} , определяемой из ограничений (16), соответствуют базисные переменные, удовлетворяющие ограничениям (17). Другими словами, точка является допустимым базисным решением задачи (15)–(17) тогда и только тогда, когда она — допустимое базисное решение [10]. Множество допустимых базисных решений совпадает с множеством крайних (экстремальных) точек допустимого множества.

Преобразуем постановку задачи 2 (5)–(7), вводя искусственную переменную x_{I+1} и обозначения $b_1 = b_Y - a_0^Y$, $a_{1i} = a_i^Y$:

$$\max_x \text{CVaR}_\alpha(\ln(\text{CVN}(\mathbf{x}))) \quad (18)$$

при ограничениях

$$\sum_{i=1}^I a_{1i} x_i - x_{I+1} = b_1, \quad (19)$$

$$l_i \leq x_i \leq u_i, \quad i=1,2,\dots,I, \quad (20)$$

$$x_{I+1} \geq 0. \quad (21)$$

Допустимое множество (19)–(21) совпадает с допустимым множеством (6), (7). Целевая функция (18) является выпуклой и имеет много локальных максимумов, расположенных в крайних точках допустимого множества [8]. Для того чтобы отыскать глобальный максимум на множестве (19)–(21), необходимо найти все крайние точки этого допустимого множества.

Схема решения оптимизационной задачи (18)–(21). Данная задача содержит только одно линейное ограничение. Поэтому лишь одна переменная является базисной, а остальные I переменных должны быть небазисными.

Рассмотрим два случая.

Случай 1. Искусственная переменная x_{I+1} базисная. Тогда каждая небазисная переменная из x_1, \dots, x_I должна принимать значение своей нижней или верхней границы (20). Пусть x_i^* — значение небазисной переменной x_i , $i=1, \dots, I$. Следовательно, значение базисной переменной x_{I+1} имеет вид:

$$x_{I+1}^* = \sum_{i=1}^I a_{1i} x_i^* - b_1. \quad (22)$$

Если $x_{I+1}^* \geq 0$, то точка $x^* = (x_1^*, \dots, x_I^*, x_{I+1}^*)$ — базисное допустимое решение задачи (18)–(21). Значит, она является крайней точкой допустимого множества (19)–(21).

Случай 2. Искусственная переменная x_{I+1} небазисная. Тогда ее значение должно быть только нулевым. Следовательно, ее можно исключить из рассмотрения. Переменную x_i , которая не является искусственной, можно включить в базис, если соответствующий ей коэффициент a_{1i} в ограничении (19) ненулевой. Пусть x_{i_1} — базисная переменная. Положим значение каждой небазисной переменной, равным ее нижней или верхней границе (20). Пусть x_i^* , $i \in \{1, \dots, I\} \setminus i_1$, — значение i -й небазисной переменной и $x_{I+1}^* = 0$. Тогда значение базисной переменной x_{i_1} определяется по формуле:

$$x_{i_1}^* = \frac{1}{a_{1i_1}} \left[b_1 - \sum_{i \in \{1, \dots, I\} \setminus i_1} a_{1i} x_i^* \right]. \quad (23)$$

Если выполняется неравенство

$$l_{i_1} \leq x_{i_1}^* \leq u_{i_1}, \quad (24)$$

то точка $x^* = (x_1^*, \dots, x_I^*, x_{I+1}^*)$ — базисное допустимое решение и, следовательно, она является крайней точкой допустимого множества (19)–(21).

Используя формулы (23), (24) для всех возможных комбинаций значений небазисных переменных, соответствующих фиксированной базисной переменной x_{i_1} , можно аналогично сформировать множество $G_{x_{i_1}}$, состоящее из базисных допустимых решений. Выполнив эту процедуру для всех возможных базисных переменных (включая искусственную переменную x_{I+1}), сгенерируем множества $G_{x_1}, G_{x_2}, \dots, G_{x_{I+1}}$ базисных допустимых решений. Тогда глобальный максимум задачи (18)–(21) можно найти, максимизируя функцию $F(x_1^*, \dots, x_I^*) = \text{CVaR}_\beta(\log(\text{CVN}(x_1^*, \dots, x_I^*)))$ на множестве $G = \bigcup_{i=1}^{I+1} G_{x_i}$.

Опишем алгоритм, генерирующий множество допустимых базисных решений путем отсеивания недопустимых решений на ранних стадиях их построения. В качестве критерия для этой операции используются простые формулы, которые являются аналитическими решениями задач линейного программирования. Процедура отсеивания существенно ускоряет процесс формирования множества G , состоящего из всех базисных допустимых решений.

Алгоритм построения базисных допустимых решений. Если искусственная переменная x_{I+1} не включена в базис, для простоты полагаем, что x_I — базисная переменная, а x_1, \dots, x_{I-1} — небазисные переменные. В этом случае искусственную переменную x_{I+1} рассматривать не будем, поскольку она равна нулю. Если искусственная переменная x_{I+1} включена в базис, то x_1, \dots, x_{I-1}, x_I — небазисные переменные. Значения небазисной переменной x_i фиксируются либо на ее нижней l_i , либо верхней u_i границе, $i = 1, \dots, I-1$. Процедуру построения множества базисных допустимых решений графически можно представить в виде двоичного дерева (рис. 1).

Каждый узел дерева (см. рис. 1) ассоциируется с некоторой переменной, соответствующей определенному уровню дерева. Так корневой узел, расположенный на нулевом уровне, соответствует переменной x_1 , узлы на первом уровне — переменной x_2 , узлы на $k-1$ -м уровне — переменной x_k , $1 \leq k \leq I-1$. Количество уровней равно количеству небазисных переменных. Если искусственная переменная x_{I+1} включена в базис, то количество уровней равно I , в противном случае $I-1$.

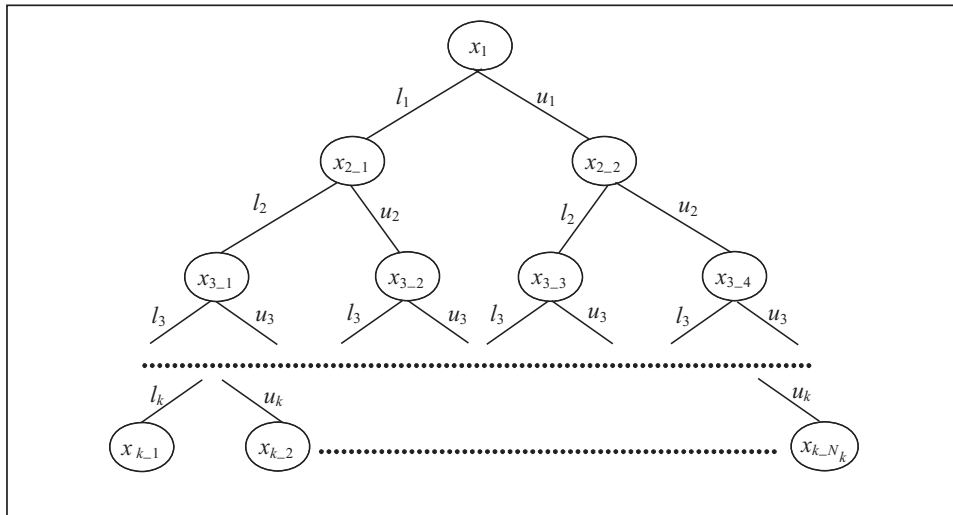


Рис. 1. Двоичное дерево построения множества допустимых базисных решений

Узлы обозначаются следующим образом: вначале записан номер переменной, с которой данный узел ассоциирован, а после символа « » указан номер пути, связывающий данный узел с корневым. Путь состоит из ребер, соединяющих два узла, расположенных на соседних уровнях. Ребру, исходящему из узла, ассоциированного с переменной x_i , соответствуют метки l_i (значение переменной x_i установлено на ее нижней границе) или u_i (значение этой переменной установлено на ее верхней границе). Пусть F_k — множество всех путей, соединяющих корневой узел с узлами, расположенными на k -м уровне. Каждый путь из множества F_k можно представить в виде набора значений переменных (x'_1, \dots, x'_k) , где x'_i принимает одно из значений: l_i или u_i , $i=1, \dots, k$.

Алгоритм начинает построение базисных допустимых решений с множества F_1 , которое может состоять не более чем из двух наборов значений переменных: (l_1) и (u_1) .

Для случая 1, когда искусственная переменная x_{I+1} является базисной, решаем два раза (при $\hat{x} = l_1$ и $\hat{x} = u_1$) следующую задачу линейного программирования: найти максимальное значение искусственной переменной x_{I+1}

$$\max_{x_2, \dots, x_I} \left(a_{11}\hat{x} + \sum_{i=2}^I a_{1i}x_i - b_1 \right) \quad (25)$$

при ограничениях

$$l_i \leq x_i \leq u_i, \quad i=2, \dots, I. \quad (26)$$

Решение задачи (25), (26) можно записать в аналитическом виде

$$x_i^{\max} = \begin{cases} u_i, & \text{если } a_{1i} \geq 0, \\ l_i, & \text{если } a_{1i} < 0, \end{cases} \quad i=2, \dots, I. \quad (27)$$

Следовательно, верхняя граница для базисной переменной x_{I+1} вычисляется по формуле

$$h_{I+1}^{\max} = \begin{cases} a_{11}l_1 + \sum_{i=2}^I a_{1i}x_i^{\max} - b_1, & \text{если } \hat{x} = l_1, \\ a_{11}u_1 + \sum_{i=2}^I a_{1i}x_i^{\max} - b_1, & \text{если } \hat{x} = u_1. \end{cases} \quad (28)$$

Из (27), (28) следует, что если $a_{11}l_1 + \sum_{i=2}^I a_{1i}x_i^{\max} - b < 0$, то набор значений переменных (l_1) необходимо исключить из множества F_1 , поскольку любое его допустимое продолжение приведет к недопустимому решению. Аналогично, если $a_{11}u_1 + \sum_{i=2}^I a_{1i}x_i^{\max} - b < 0$, то набор значений переменных (u_1) следует исключить из множества F_1 .

Если после отсеивания недопустимых решений множество F_1 содержит по крайней мере один набор значений переменных, то приступаем к построению множества F_2 . Полагаем N_1 — количество наборов значений переменных в множестве F_1 .

Если множество F_1 пустое, то алгоритм заканчивает построение множества допустимых базисных решений, соответствующих базисной переменной x_{I+1} , и приступает к построению допустимых базисных решений, соответствующих следующей базисной переменной.

Для случая 2, когда искусственная переменная x_{I+1} не является базисной, для простоты будем предполагать, что x_I — базисная переменная, а x_1, \dots, x_{I-1} — небазисные.

Решаем два раза (при $\hat{x} = l_1$ и $\hat{x} = u_1$) следующие задачи линейного программирования: найти минимальное значение для переменной x_I

$$\min_{x_2, \dots, x_{I-1}} \left(\frac{1}{a_{1I}} \left[b_1 - a_{11}\hat{x} - \sum_{i \in \{2, \dots, I-1\}} a_{1i}x_i \right] \right) \quad (29)$$

при ограничениях

$$l_i \leq x_i \leq u_i, \quad i \in \{2, \dots, I-1\} \quad (30)$$

и найти максимальное значение для переменной x_I

$$\max_{x_2, \dots, x_{I-1}} \left(\frac{1}{a_{1I}} \left[b_1 - a_{11}\hat{x} - \sum_{i \in \{2, \dots, I-1\}} a_{1i}x_i \right] \right) \quad (31)$$

при ограничениях

$$l_i \leq x_i \leq u_i, \quad i \in \{2, \dots, I-1\}. \quad (32)$$

Решение задач (29), (30) и (31), (32) можно записать в аналитическом виде соответственно

$$x_i^{\min} = \begin{cases} l_i, & \text{если } -\frac{a_{1i}}{a_{1I}} \geq 0, \\ u_i, & \text{если } -\frac{a_{1i}}{a_{1I}} < 0, \end{cases} \quad i \in \{2, \dots, I-1\}, \quad (33)$$

$$x_i^{\max} = \begin{cases} u_i, & \text{если } -\frac{a_{1i}}{a_{1I}} \geq 0, \\ l_i, & \text{если } -\frac{a_{1i}}{a_{1I}} < 0, \end{cases} \quad i \in \{2, \dots, I-1\}. \quad (34)$$

Обозначим

$$h_I^{\min}(\hat{x}) = \frac{1}{a_{1I}} \left[b_1 - a_{11}\hat{x} - \sum_{i \in \{2, \dots, I-1\}} a_{1i} x_i^{\min} \right], \quad (35)$$

$$h_I^{\max}(\hat{x}) = \frac{1}{a_{1I}} \left[b_1 - a_{11}\hat{x} - \sum_{i \in \{2, \dots, I-1\}} a_{1i} x_i^{\max} \right]. \quad (36)$$

Если не выполняется неравенство

$$\max \{h_I^{\min}(l_1), l_I\} \leq \min \{h_I^{\max}(l_1), u_I\}, \quad (37)$$

то исключаем набор значений переменных (l_1) из множества F_1 .

Если не выполняется неравенство

$$\max \{h_I^{\min}(u_1), l_I\} \leq \min \{h_I^{\max}(u_1), u_I\}, \quad (38)$$

то исключаем набор значений переменных (u_1) из множества F_1 .

Формулы (33)–(38) определяют процедуру отсеивания недопустимых решений. Если после выполнения этой процедуры множество F_1 содержит по крайней мере один набор значений переменных, то приступаем к построению множества F_2 . Полагаем N_1 — количество наборов значений переменных в множестве F_1 .

Если множество F_1 пустое, то алгоритм заканчивает построение множества допустимых базисных решений, соответствующих текущей базисной переменной, и приступает к построению допустимых базисных решений, соответствующих следующей базисной переменной.

Предположим, что для фиксированной базисной переменной полностью построено множество F_{k-1} , $1 < k \leq I-1$, и после процедуры отсеивания в этом множестве содержатся N_{k-1} наборов значений переменных, состоящих из $k-1$ компонент $\{(\tilde{x}_1^j, \dots, \tilde{x}_{k-1}^j)\}_{j=1}^{N_{k-1}}$, где \tilde{x}_i^j — фиксированное значение переменной x_i , $1 \leq i \leq k-1$, которое было установлено на ее нижней l_i или верхней u_i границе. Тогда выполняется следующая пошаговая процедура для случая 1.

Шаг 0. Полагаем $j=1$, $F_k = \emptyset$, $N_k = 0$.

Шаг 1. Решаем два раза (при $\hat{x} = l_k$ и $\hat{x} = u_k$) задачу линейного программирования: найти максимальное значение искусственной переменной x_{I+1}

$$\max_{x_{k+1}, \dots, x_I} \left(\sum_{i=1}^{k-1} a_{1i} \tilde{x}_i^j + a_{1k} \hat{x} + \sum_{i=k+1}^I a_{1i} x_i - b_1 \right) \quad (39)$$

при ограничениях

$$l_i \leq x_i \leq u_i, \quad i = k+1, \dots, I. \quad (40)$$

Шаг 2. Решение задачи (39), (40) можно записать в аналитическом виде:

$$x_i^{\max} = \begin{cases} u_i, & \text{если } a_{1i} \geq 0, \\ l_i, & \text{если } a_{1i} < 0, \end{cases} \quad i = k+1, \dots, I. \quad (41)$$

Следовательно, верхняя граница для базисной переменной x_{I+1} вычисляется по формуле:

$$h_{I+1}^{\max} = \begin{cases} \sum_{i=1}^{k-1} a_{1i} \tilde{x}_i^j + a_{1k} l_k + \sum_{i=k+1}^I a_{1i} x_i^{\max} - b_1, & \text{если } \hat{x} = l_k, \\ \sum_{i=1}^{k-1} a_{1i} \tilde{x}_i^j + a_{1k} u_k + \sum_{i=k+1}^I a_{1i} x_i^{\max} - b_1, & \text{если } \hat{x} = u_k. \end{cases} \quad (42)$$

Шаг 3. Используя формулы (41), (42) выполняем следующую процедуру. Если $\sum_{i=1}^{k-1} a_{1i} \tilde{x}_i^j + a_{1k} l_k + \sum_{i=k+1}^I a_{1i} x_i^{\max} - b_1 \geq 0$, то прибавляем набор значений переменных $(\tilde{x}_1^j, \dots, \tilde{x}_{k-1}^j, l_k)$ к множеству F_k . Полагаем $N_k = N_k + 1$.

Если $\sum_{i=1}^{k-1} a_{1i} \tilde{x}_i^j + a_{1k} u_k + \sum_{i=k+1}^I a_{1i} x_i^{\max} - b_1 \geq 0$, то прибавляем набор значений переменных $(\tilde{x}_1^j, \dots, \tilde{x}_{k-1}^j, u_k)$ к множеству F_k . Полагаем $N_k = N_k + 1$.

Шаг 4. Если $j < N_{k-1}$, то полагаем $j = j + 1$. Переходим к шагу 1.

Шаг 5. Если $F_k = \emptyset$, то полагаем $G_{x_{I+1}} = \emptyset$. Переходим к построению допустимых базисных решений, соответствующих следующей базисной переменной.

Шаг 6. Если $F_k \neq \emptyset$ и $k < I$, то приступаем к построению F_{k+1} . Полагаем $k = k + 1$. Переходим к шагу 0.

Шаг 7. Если $F_k \neq \emptyset$ и $k = I$, то множество F_I содержит N_I набор значений переменных, состоящих из I компонент $\{(x_1^{*j}, \dots, x_I^{*j})\}_{j=1}^{N_I}$. Для каждого набора значений переменных $(x_1^{*j}, \dots, x_I^{*j})$ из множества F_I вычисляем значение x_{I+1}^{*j} базисной переменной x_{I+1} по формуле (22), $j = 1, \dots, N_I$. В соответствии с описываемой процедурой переменная x_{I+1}^{*j} неотрицательная и, следовательно, допустимая. Таким образом, $G_{x_{I+1}} = \{(x_1^{*j}, \dots, x_I^{*j}, x_{I+1}^{*j})\}_{j=1}^{N_I}$ является множеством всех допустимых базисных переменных, соответствующих базисной (искусственной) переменной x_{I+1} .

Переходим к построению допустимых базисных решений, соответствующих базисной переменной для случая 2.

Шаг 0. Полагаем $j = 1$, $F_k = \emptyset$, $N_k = 0$.

Шаг 1. Решаем два раза (при $\hat{x} = l_k$ и $\hat{x} = u_k$) следующую задачу линейного программирования: найти минимальное значение переменной x_I

$$\min_{x_{k+1}, \dots, x_{I-1}} \left(\frac{1}{a_{1I}} \left[b_1 - \sum_{i \in \{1, \dots, k-1\}} a_{1i} \tilde{x}_i^j - a_{1k} \hat{x} - \sum_{i \in \{k+1, \dots, I-1\}} a_{1i} x_i \right] \right) \quad (43)$$

при ограничениях

$$l_i \leq x_i \leq u_i, \quad i \in \{k+1, \dots, I-1\} \quad (44)$$

и найти максимальное значение для переменной x_I

$$\max_{x_{k+1}, \dots, x_{I-1}} \left(\frac{1}{a_{1I}} \left[b_1 - \sum_{i \in \{1, \dots, k-1\}} a_{1i} \tilde{x}_i^j - a_{1k} \hat{x} - \sum_{i \in \{k+1, \dots, I-1\}} a_{1i} x_i \right] \right) \quad (45)$$

при ограничениях

$$l_i \leq x_i \leq u_i, \quad i \in \{k+1, \dots, I-1\}. \quad (46)$$

Шаг 2. Решение задач (43), (44) и (45), (46) можно записать в аналитическом виде соответственно

$$x_i^{\min} = \begin{cases} l_i, & \text{если } -\frac{a_{1i}}{a_{1I}} \geq 0, \\ u_i, & \text{если } -\frac{a_{1i}}{a_{1I}} < 0, \end{cases} \quad i \in \{k+1, \dots, I-1\},$$

$$x_i^{\max} = \begin{cases} u_i, & \text{если } -\frac{a_{1i}}{a_{1I}} \geq 0, \\ l_i, & \text{если } -\frac{a_{1i}}{a_{1I}} < 0, \end{cases} \quad i \in \{k+1, \dots, I-1\}.$$

Обозначим

$$h_I^{\min}(\hat{x}) = \frac{1}{a_{1I}} \left[b_1 - \sum_{i \in \{1, \dots, k-1\}} a_{1i} \tilde{x}_i^j - a_{1k} \hat{x} - \sum_{i \in \{k+1, \dots, I-1\}} a_{1i} x_i^{\min} \right],$$

$$h_I^{\max}(\hat{x}) = \frac{1}{a_{1I}} \left[b_1 - \sum_{i \in \{1, \dots, k-1\}} a_{1i} \tilde{x}_i^j - a_{1k} \hat{x} - \sum_{i \in \{k+1, \dots, I-1\}} a_{1i} x_i^{\max} \right].$$

Шаг 3. Если

$$\max \{h_I^{\min}(l_k), l_I\} \leq \min \{h_I^{\max}(l_k), u_I\},$$

то прибавляем набор значений переменных $(\tilde{x}_1^j, \dots, \tilde{x}_{k-1}^j, l_k)$ к множеству F_k .
Полагаем $N_k = N_k + 1$.

Если

$$\max \{h_I^{\min}(u_k), l_I\} \leq \min \{h_I^{\max}(u_k), u_I\},$$

то прибавляем набор значений переменных $(\tilde{x}_1^j, \dots, \tilde{x}_{k-1}^j, u_k)$ к множеству F_k .
Полагаем $N_k = N_k + 1$.

Шаг 4. Если $j < N_{k-1}$, то полагаем $j = j + 1$. Переходим к шагу 1.

Шаг 5. Если $F_k = \emptyset$, то полагаем $G_{x_I} = \emptyset$. Переходим к построению допустимых базисных решений, соответствующих следующей базисной переменной.

Шаг 6. Если $F_k \neq \emptyset$ и $k < I$, то приступаем к построению F_{k+1} . Полагаем $k = k + 1$. Переходим к шагу 0.

Шаг 7. Если $F_k \neq \emptyset$ и $k = I - 1$, то множество F_{I-1} содержит N_{I-1} набор значений переменных, состоящих из $I - 1$ компонент $\{(x_1^{*j}, \dots, x_{I-1}^{*j})\}_{j=1}^{N_{I-1}}$. Для каждого набора значений переменных $(x_1^{*j}, \dots, x_{I-1}^{*j})$ из множества F_{I-1} вычисляем значение x_I^{*j} базисной переменной x_I по формуле (23), $j = 1, \dots, N_{I-1}$. В соответствии с описываемой процедурой переменная x_I^{*j} является допустимой. Следовательно, $G_{x_I} = \{(x_1^{*j}, \dots, x_{I-1}^{*j}, x_I^{*j})\}_{j=1}^{N_{I-1}}$ является множеством всех допустимых базисных переменных, соответствующих базисной переменной x_I . Переходим к построению допустимых базисных решений, соответствующих следующей базисной переменной.

Результаты вычислений. С помощью предложенного алгоритма решалась задача 2 (5)–(7) при фиксированном значении параметра b_Y , который представля-

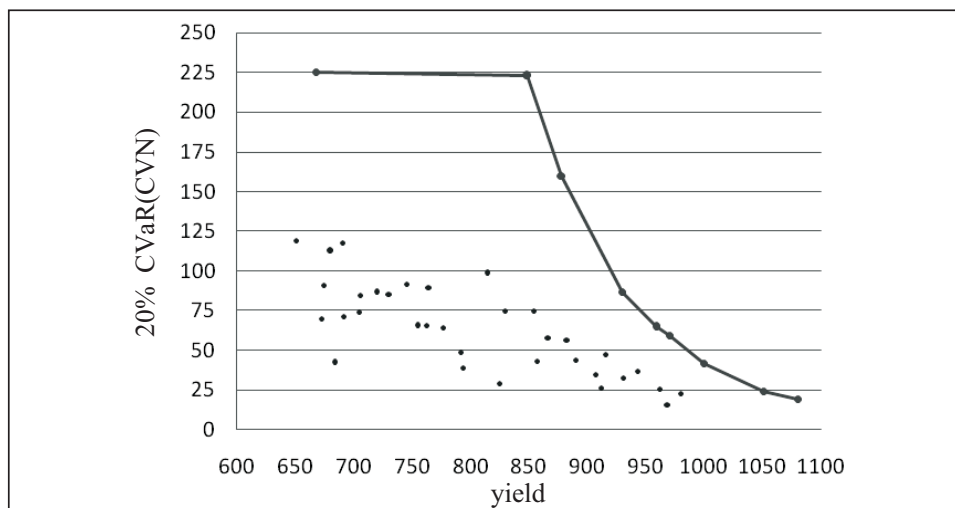


Рис 2. Граница эффективности 20% CVaR(CVN)–yield

ет собой нижнюю допустимую границу для показателя yield. В результате решения задачи (5)–(7) получена точка $x^{b_Y} = (x_1^{b_Y}, \dots, x_I^{b_Y})$, характеризующая оптимальную комбинацию параметров химического состава сплавов и технологических параметров их изготовления. Этой оптимальной точке соответствует некоторый гипотетический образец стали с наилучшими прогнозируемыми значениями показателей 20% CVaR(CVN) и yield. Многократно решая задачу (5)–(7) при различных значениях параметра b_Y , получаем множество таких гипотетических образцов. Отобразим это множество в виде кривой (рис. 2) на диаграмме 20% CVaR(CVN)–yield.

В теории портфельной оптимизации кривую (см. рис. 2) называют границей эффективности Марковица (efficient frontier) [11]. При фиксированном значении показателя yield граница эффективности определяет границу оптимальности, выше которой значений показателя 20% CVaR(CVN) не существует. Все точки, расположенные на этой границе, соответствуют образцам сталей с наилучшими соотношениями показателей 20% CVaR(CVN)–yield.

Реальные экспериментальные образцы сталей изображены в виде отдельных точек, расположенных ниже границы эффективности (см. рис. 2). Для любого образца стали можно указать оптимальную комбинацию параметров химического состава и технологических параметров, соответствующую точке, лежащей на границе эффективности. Эта точка характеризуется более высоким значением показателя 20% CVaR(CVN) при том же уровне yield.

ЗАКЛЮЧЕНИЕ

В статье предложены математические методы для оптимизации химического состава сплавов и технологических параметров их изготовления в процессе разработки новых марок сталей. Этот процесс осуществляется в условиях значительной неопределенности. Рассмотрены методы оптимизации двух важных механических характеристик сталей: предел текучести при растяжении (yield) и показатель разрушения по методу Шарпи с V-образным надрезом (CVN). Если yield достаточно точно аппроксимируется обычной линейной регрессионной моделью, то при моделировании CVN с помощью линейной регрессии достичь приемлемой точности не удалось, поскольку экспериментальные данные по CVN характеризуются существенным разбросом. Поэтому CVN рас-

считается как случайная величина, а оптимизации (максимизации) подлежит характеристика ее функции распределения CVaR, являющейся выпуклой (невогнутой) и имеющей много локальных максимумов на выпуклом множестве, генерируемом линейным ограничением на yield.

В статье предложены два метода поиска глобального максимума функции CVaR: первый состоит в генерировании множества перспективных точек, являющихся хорошим начальным приближением, в целях повышения эффективности работы метода мультистарта, а второй основан на утверждении, что глобальный максимум выпуклой функции на непустом замкнутом ограниченном выпуклом допустимом множестве достигается на некоторой экстремальной (крайней) точке этого множества. Он позволяет найти точное решение рассматриваемой задачи при количестве переменных, не превышающем 15. Рассматриваемая задача оптимизации химического состава сплавов и технологических параметров их изготовления в процессе разработки новых марок сталей удовлетворяет этому ограничению. При большей размерности задачи предлагаемый метод можно легко адаптировать с использованием средств параллельного программирования.

Разработанный инструментарий использовался для решения задачи максимизации 20% CVaR(CVN) при ограничении на yield. Полученные решения представлены в виде границы эффективности, т.е. границы оптимальности, выше которой значений показателя 20% CVaR(CVN) при фиксированном значении показателя yield не существует.

СПИСОК ЛИТЕРАТУРЫ

1. Golodnikov A., Macheret Y., Trindade A., Uryasev S., Zrazhevsky G. Optimization of composition and processing parameters for the development of steel alloys: A statistical model-based approach // *Journal of Industrial and Management Optimization*. — 2007. — 3(3). — P. 489–501.
2. Golodnikov A., Macheret Y., Trindade A., Uryasev S., Zrazhevsky G. Statistical modeling of composition and processing parameters for alloy development // *Modeling and Simulation in Material Science and Engineering*. — 2005. — 13. — P. 633–644.
3. Zrazhevsky G., Golodnikov A., Uryasev S., Zrazhevsky A. Advanced statistical tools for modelling of composition and processing parameters for alloy development / A. Migdalas, A. Karakitsiou (eds.) // *Optimization, Control, and Applications in the Information Age, Springer Proceedings in Mathematics & Statistics*, 2015. — P. 393–413.
4. Corowin W.R., Houghl A.M. Effect of specimen size and material condition on the Charpy impact properties of 9Cr-1Mo-V-Nb steel // *The Use of Small-Scale Specimens for Testing Irradiated Material*, ASTM STP 888 (Philadelphia, PA), 1986. — P. 325–338.
5. Lucon E., Chaouadi R., Fabry A., Puzzolante J.-L., Valle E. Characterizing material properties by the use of full-size and sub-size Charpy tests / T. Siewert, M.P. Manahan (eds.) // *Pendulum Impact Testing: A Century of Progress*, ASTM STP 1380, American Society for Testing and Materials (West Conshohocken, PA), 1999. — P. 146–163. — 399 p.
6. Koenker R., Bassett G. Regression quantiles // *Econometrica*, 1978. — 46, N 1. — P. 33–50.
7. Rockafellar R.T., Uryasev S. Conditional Value-at-Risk for general loss distributions // *Journal of Banking and Finance*. — 26, N 7. — 2002, — P. 1443–1471.
8. Rockafellar R.T. *Convex analysis*. — Princeton (NJ): Princeton University Press, 1970. — 453 p.
9. Friedman J.H. and Fisher N.I. Bump-hunting in high-dimensional data // *Statistics and Computing*. — 1999. — 9. — P. 123–143.

10. Gass S.I. Linear programming. Methods and applications. 5th ed. — New York: McGraw-Hill, 1985.
11. Markowitz H.M. Portfolio selection // Journal of Finance. — 1952. — 7. — P. 77–91.
12. Goldren A.P., Cox T.B. AMAX Report, CPR-2, AMAX Materials Research Center, Ann Arbor, MI (1986).
13. Жиглявский А.А., Жилинская А.Г. Методы поиска глобального экстремума. — М.: Наука, 1991. — 248 с.
14. Hansen E. Global optimization using interval analysis. — New York: Dekker, 1992. — 532 p.

Надійшла до редакції 04.01.2016

Г.М. Зражевський, О.М. Голодніков, С.П. Урясьєв, О.Г. Зражевський
РОЗРОБКА МЕТОДІВ ОПТИМІЗАЦІЇ ДЛЯ ПОШУКУ НАЙКРАЩОЇ КОМБІНАЦІЇ
МІЦНОСТІ ТА В'ЯЗКОСТІ СПЛАВІВ

Анотація. Запропоновано математичні методи для оптимізації хімічного складу сплавів та технологічних параметрів їхнього виготовлення в процесі розроблення нових марок сталей в умовах значної невизначеності. В оптимізаційних задачах використовуються статистичні моделі механічних властивостей нових сплавів. Для оцінки правого хвоста ймовірнісного розподілу показника руйнування за методом Шарпі з V-подібним надрізом (CVN) використано CVaR, який є опуклою функцією. Запропоновано математичні методи розв'язання таких задач максимізації.

Ключові слова: сплав, оптимізація, CVaR, квантиль, CVN.

G.M. Zrazhevsky, A.N. Golodnikov, S.P. Uryasev, A.G. Zrazhevsky
OPTIMIZATION TECHNIQUES TO OBTAIN THE BEST COMBINATION
OF ALLOY STRENGTH AND TOUGHNESS

Abstract. The paper proposes mathematical tools for optimization of combinations of processing parameters and alloy compositions for the development of new and improved steels. Development process of new steels involves a lot of uncertainties. For this reason statistical models for mechanical properties of new alloys are used in optimization problems. Particularly, we use CVaR for estimating the right tail of CVN (Charpy V-Notch impact on toughness) distribution. This results in non-concave (convex) objective functions in maximization problem. The paper suggests methods for solving such a problem.

Keywords: alloy, optimization, CVaR, quantile, CVN.

Зражевский Григорий Михайлович,

кандидат физ.-мат. наук, доцент Киевского национального университета имени Тараса Шевченко, e-mail: zgrig@univ.kiev.ua.

Голодников Александр Николаевич,

кандидат физ.-мат. наук, ведущий научный сотрудник Института кибернетики им. В.М. Глушкова НАН Украины, Киев.

Урясьєв Станислав Павлович,

кандидат физ.-мат. наук, профессор Флоридского университета, Гейнсвилл, Флорида, США, e-mail: uryasev@ufl.edu.

Зражевский Алексей Григорьевич,

кандидат техн. наук, аналитик-статистик компании American Optimal Decisions, Gainesville, FL, USA, e-mail: alex.zrazhevsky@aorda.com.