

Научная статья
УДК 004.832
doi:10.37614/2949-1215.2023.14.7.010

ДВА МЕТОДА ЛОКАЛЬНОГО ПОИСКА В ПЛАНИРОВАНИИ ОТКРЫТЫХ ГОРНЫХ РАБОТ

Алексей Владимирович Шестаков¹, Александр Анатольевич Зуенко²

^{1, 2}*Институт информатики и математического моделирования имени В. А. Путилова
Кольского научного центра Российской академии наук, Апатиты, Россия*

¹*shestakov@iimm.ru, <https://orcid.org/0000-0002-9052-2579>*

²*zuenko@iimm.ru, <https://orcid.org/0000-0002-7165-6651>*

Аннотация

Данная статья направлена на исследование способов планирования открытых горных работ с применением локального поиска (генетические алгоритмы в работе не рассматриваются). Открытые горные работы являются сложной и многогранной задачей, требующей эффективного планирования для оптимизации добычи полезных ископаемых. Методы локального поиска являются мощным инструментом оптимизации, позволяющим искать локальные оптимумы в пространствах высокой размерности. В статье рассмотрены два наиболее часто применяемых подхода к локальному поиску для планирования открытых горных работ.

Ключевые слова:

локальный поиск, планирование открытых горных работ, добыча полезных ископаемых, поиск с запретами, имитация отжига

Финансирование:

работа выполнена в рамках НИР «Разработка теоретических и организационно-технических основ информационной поддержки управления жизнеспособностью региональных критических инфраструктур Арктической зоны Российской Федерации» (регистрационный номер 122022800547-3).

Для цитирования:

Шестаков А. В., Зуенко А. А. Два метода локального поиска в планировании открытых горных работ // Труды Кольского научного центра РАН. Информационные технологии. 2023. Т. 14, № 7. С. 92–101. doi:10.37614/2949-1215.2023.14.7.010.

Original article

TWO METHODS OF LOCAL SEARCH IN OPEN-PIT MINING PLANNING

Aleksey V. Shestakov¹, Alexander A. Zuenko²

¹*Putilov Institute for Informatics and Mathematical Modeling of the Kola Science Centre of the Russian Academy of Sciences, Apatity, Russia*

¹*shestakov@iimm.ru, <https://orcid.org/0000-0002-9052-2579>*

²*zuenko@iimm.ru, <https://orcid.org/0000-0002-7165-6651>*

Abstract

This paper aims to investigate ways of planning open pit mining using local search (genetic algorithms are not discussed in the paper). Open pit mining is a complex and multifaceted problem that requires efficient planning to optimize mineral extraction. The Local search methods are a powerful optimization tool to search for local optima in the space of large size. This paper reviews two of the most commonly used local search approaches for open pit mining planning.

Keywords:

local search, open pit mine planning, mining, tabu search, simulated annealing

Funding:

the study was carried out within the framework of the Putilov Institute for Informatics and Mathematical Modeling of the Kola Science Centre of the Russian Academy of Sciences state assignment of the Ministry of Science and Higher Education of the Russian Federation, research topic “Development of theoretical and organizational and technical foundations of information support for managing the viability of regional critical infrastructures of the Arctic zone of the Russian Federation” (registration number of the research topic 122022800547-3).

For citation:

Shestakov A. V., Zuenko A. A. Two methods of local search in open-pit mining planning // Transactions of the Kola Science Centre of RAS. Series: Engineering Sciences. 2023. Vol. 14, № 7. P. 92–101. doi:10.37614/2949-1215.2023.14.7.010.

Введение

Одним из основных методов добычи полезных ископаемых является добыча открытым способом. При использовании такого подхода можно добывать большие объемы полезных ископаемых в сравнительно короткие сроки и при сравнительно низких затратах на добычу в сравнении с другими способами, открытая разработка часто экономически более эффективна. Достоинством также является возможность использовать мощные технические средства и оборудование, что снижает стоимость добычи на единицу продукции. Такая разработка предусматривает добычу полезных ископаемых даже на больших глубинах. Это является важным фактором для труднодоступных месторождений, где другие методы добычи неэффективны.

Планирование добычи открытым способом включает определение оптимального расположения месторождения, последовательности операций для оптимизации выемки и использования оборудования и ресурсов, а также управление безопасностью и рисками. Его цель состоит в обеспечении эффективности добычи, соблюдении экологических норм и обеспечении безопасности работников и населения.

Наиболее часто планирование добычи открытым способом происходит с помощью блочной модели [1], которая позволяет более точно и эффективно определить оптимальный план добычи, учитывая геологические, технические и экономические факторы. Это значительно повышает эффективность добычи и помогает достичь максимальной прибыли от разработки месторождения.

Задача добычи полезных ископаемых открытым способом (OPMPS — Open Pin Mine Production Shedulling) является оптимизационной проблемой, в которой нужно найти оптимальное распределение ресурсов, чтобы максимизировать прибыль при соблюдении различных ограничений таких, как геологические, экологические и экономические факторы. Для нахождения наилучшего плана добычи полезных ископаемых в горнодобывающей промышленности и оптимизации часто используются алгоритмы, основанные на локальном поиске.

Основная идея локального поиска заключается в том, что он начинает с некоторого начального решения, а затем производит итеративную оптимизацию этого решения, перемещаясь к соседним решениям, которые имеют более высокую оценку качества.

Локальный поиск может использоваться в системе планирования добычи полезных ископаемых для выполнения следующих шагов:

- 1) определение начального распределения ресурсов (начального плана);
- 2) определение функции оценки качества плана добычи полезных ископаемых;
- 3) непосредственно решение задачи;
- 4) оформление и анализ полученного плана добычи полезных ископаемых.

Локальный поиск в системе планирования добычи полезных ископаемых открытым способом позволяет эффективно оптимизировать план добычи, учитывая различные ограничения и требования. Таким образом, он может быть адаптирован к специфическим условиям каждой горнодобывающей компании и помогать повышать эффективность и экономическую эффективность добычи.

Имитация отжига для планирования добычи открытым способом

Постановка для случая задачи OPMPS с возможностью накопления запасов

В статье [2] представлена идея по использованию алгоритма имитации отжига (SA) для решения задачи OPMPS с учетом накопления запасов в качестве составляющей максимизации целевой функции. В работе также показана разработка и применение эвристики «жадного» поиска, направленной на улучшение вычислительной эффективности алгоритма SA.

Ниже описаны основные элементы, используемые для математической формулировки задачи:

- B — общее количество блоков, где $b = 1, 2, \dots, B$;
- T — общее количество производственных периодов или лет, где $t = 1, 2, \dots, T$;
- K — общее количество ячеек, где $k = 1, 2, \dots, K$;
- M — максимальная мощность добычи в тоннах материала в течение производственного года t ;

- C^d — максимальная мощность переработки в тоннах руды в пункте назначения d в течение производственного года t ;
- R — максимальная мощность переработки в тоннах, тройских унциях или граммах металла в течение производственного года t ;
- mr^d — извлечение металла в процентах в пункте назначения d ;
- q_b — доступное количество материала в блоке b ;
- $\bar{g}_b$ — содержание материала в процентах, тройской унции или граммах металла в блоке b ;
- sq_k — доступное количество материала в ячейке k ;
- G_{lk} — более низкий сорт в ячейке для сортировки k ;
- G_{uk} — высший сорт в сортировочной ячейке k ;
- $\bar{G}_k$ — средняя оценка в корзине оценок;
- p — цена металла в долларах за тонну, тройскую унцию или грамм металла;
- s — затраты на переработку, маркетинг или продажу в долларах за тонну, тройскую унцию или грамм металла;
- m — затраты на добычу в долларах США за тонну добытого материала;
- C^d — затраты на переработку в долларах США за тонну руды, переработанной в пункте назначения переработки d ;
- i — ставка дисконтирования в %;
- h — стоимость повторной обработки в долларах за тонну руды со склада.

Переменные:

- W_t — количество материала, добытого за период t , остается в пределах $0 \leq W_t \leq M$;
- CP_t^d — количество материала, обработанного в пункте назначения d за период t , остается в пределах $0 \leq CP_t^d \leq C^d$;
- RF_t — количество металла, очищенного за период t , остается в пределах $0 \leq RF_t \leq R$;
- $x_{bt} = \begin{cases} 1; & \text{если блок } b \text{ добыт в год } t; \\ 0; & \text{в противном случае} \end{cases}$;
- y_{btd} — количество руды, взятой из блока b в период t и отправленной в пункт назначения d ;
- s_{dt}^k — взято материалов k в период t из склада для хранения и отправлено к месту обработки d .

Целевая функция (1) в рассматриваемом методе описана в терминах, представляющих дисконтированный доход и затраты, связанные с добычей, переработкой, рафинированием и повторным обращением с минерализованным материалом:

$$\text{Maximize } z = \sum_{t=1}^T \left(\frac{1}{(1+i)^t} \right) [(s-r)RF_t - C^d * CP_t^d - m * W_t - \sum_{k=1}^K \sum_{d>1}^D h * s_{dt}^b]. \quad (1)$$

Ограничение (2) гарантирует, что блок добывается не более одного раза:

$$\sum_{t=1}^T x_{bt} \leq 1 \quad \forall b = 1, 2, \dots, B. \quad (2)$$

Приоритет между блоками обеспечивается с помощью ограничения (3):

$$x_{bt} - \sum_{\tau=1}^T x_{jt} \leq 0 \quad \forall b = 1, 2, \dots, B, \quad (3)$$

где j — набор предшественников блока b (набор блоков, извлекаемых до блока b).

Ограничения (4–6) обеспечивают использование мощностей по добыче, переработке и рафинированию в каждом периоде горизонта планирования:

$$\sum_{b=1}^B q_{bt} x_{bt} = W_t; \quad \forall t = 1, 2, \dots, T, \quad (4)$$

$$\sum_{b=1}^B y_{btd} + \sum_{k=1}^K s_{dt}^k = CP_t^d; \quad \forall t = 1, 2, \dots, T; d > 1, \quad (5)$$

$$\sum_{d>1}^D [\sum_{k=1}^B mr^d * \bar{g}_b * y_{btd} + \sum_{k=1}^K mr^d * \bar{G}_k * s_{dt}^k] = RF_t; \quad \forall t = 1, 2, \dots, T. \quad (6)$$

Ограничение (7) создает связь между бинарной x_{bt} и линейной переменной y_{btd} и гарантирует, что максимальное количество материала, добытого из блока, меньше или равно общему количеству, доступному в блоке.

$$\sum_{d=1}^D y_{btd} \leq q_b * x_{bt} \quad t = 1, 2, \dots, T. \quad (7)$$

Схема алгоритма имитации отжига

Имитация отжига (*SA*) — это алгоритм стохастического поиска для решения задач комбинаторной оптимизации высокой размерности [3]. Алгоритм использует механизм случайного поиска путем возмущения текущего решения, начиная с исходного, для генерации нового решения S' в окрестности текущего. Если значение целевой функции $f(S')$ улучшается, то шаг принимается, однако, если перемещение уменьшает целевое значение, то алгоритм *SA* переводит систему в это состояние с некоторой вероятностью p , известной как вероятность перехода, которая позволяет алгоритму отклониться от локального оптимума в поисках глобального оптимума. Вероятность p может быть получена с использованием критерия *metropolis*, приведенного ниже в уравнении (8):

$$p = \max \begin{cases} e^{-\frac{\Delta E}{T}} & \text{если } f(S') < f(S), \\ 1 & \text{если } f(S') \geq f(S) \end{cases} \quad (8)$$

где $\Delta E = f(S') - f(S)$ — изменение значения целевой функции; а T — температура.

Процесс повторяется итеративно путем поддержания постоянной температуры в течение достаточного числа модификаций таким образом, чтобы при этой температуре было достигнуто равновесное состояние, и затем она снижается с помощью схемы охлаждения. Производительность *SA* зависит от управляющих параметров алгоритма, т. е. температуры (T), схемы охлаждения, повторения возмущений при каждой температуре (N) и структуры окрестности.

Модификация метода имитация отжига

В статье рассматривается эвристический алгоритм ранжированного позиционного веса (*RPW* — *Ranked Positional Weight*), который был использован для генерации начального неоптимального входного решения для алгоритма *SA* [4].

Этот метод работает по логике, согласно которой, если есть какой-либо ценный блок, лежащий под блоком, то вышележащий блок должен получить больше очков, потому что его добыча приводит к получению ценных блоков, лежащих под ним. Первоначальное решение, сгенерированное *RPW*, затем передается в *SA* с возможностью накопления запасов для дальнейшей оптимизации.

Алгоритм на основе SA

1. Генерация исходного неоптимального решения с использованием эвристического алгоритма ранжированного позиционного веса (*RPW*).
2. Выбор значения начальной температуры T_0 и общее количество итераций при каждой температуре (N).
3. Пока (текущее количество итераций меньше, чем максимальное количество итераций).
4. Найти блоки-кандидаты (т. е. изменение периода) для замены.
5. Создать новое решение, используя две возможности замены:
 - а. перенести выбранные блоки на ранний период;
 - б. перенести выбранные блоки на более поздний период.
6. Запустить эвристику накопления запасов для поиска нового решения.
7. Запустить жадную эвристику после заданного количества итераций.
8. Вычислить значение целевой функции для нового решения.

9. Если решение улучшилось, принять новое решение; в противном случае принять новое решение с вероятностью $e^{\frac{\Delta E}{T}}$.

10. Если критерии остановки выполнены: завершить, в противном случае перейти к шагу 2.

Производительность SA зависит от начальной температуры, состояния равновесия, функции охлаждения и критериев остановки. Если на старте оптимизации температура установлена слишком высокой, вероятность принятия нежелательных переходов будет выше, следовательно, SA будет выполнять более или менее случайный локальный поиск. Это увеличит время схождения SA и может не дать хороших результатов. Также, если температура выбрана слишком низкой, то SA , скорее всего, приведет в локальный оптимум в случае большой комбинаторной задачи. Следовательно, начальная температура должна быть рассчитана для каждой рассматриваемой задачи. В работе [2] температура рассчитывалась как в уравнении (9):

$$Initial\ temperature = \frac{\Delta E}{P}, \quad (9)$$

где P — вероятность принятия неблагоприятных решений, зависит от качества исходного решения, а ΔE — среднее значение целевой функции, рассчитанное путем запуска SA при очень высокой температуре и усреднения значения целевой функции только для принятых решений, не улучшающих ситуацию. Вероятность принятия выбирается с учетом качества исходного решения; если она установлена слишком высокой, преимущество использования исходного решения хорошего качества будет потеряно. Для расчета начальной температуры в статье было выбрано значение $P = 40\%$.

Как только начальная температура определена, она снижается после переходов для достижения равновесного. Если температура снижается быстро, то качество решения может быть низким, однако если ее снижать медленно, то будет получено решение ближе к оптимальному, но со значительно большими временными затратами. Температура обновляется по формуле (10):

$$T = \alpha T, \quad (10)$$

где α находится в диапазоне от 0 до 1.

Механизм перехода позволяет искать соседние решения путем перемещения некоторых блоков на предыдущий или следующий производственный период. Блоки, которые могут быть перемещены без нарушения ограничений приоритета и доступа, являются блоками-кандидатами. Для их определения проверяется период времени добычи t каждого блока, а также их предшественники и последователи. Таким образом:

- если период времени t добычи блока больше периода времени всех его предшественников $t - 1$, он может быть перемещен на предыдущий период;
- если период времени t добычи блока меньше времени всех его преемников $t + 1$, он может быть перемещен на более поздний период.

С помощью случайного числа $\{0,1\}$ определяется направление передачи блока, а также количество блоков, которые будут перемещены для формирования нового соседнего решения. При этом отклоняются переходы, нарушающие ограничения интеллектуального анализа.

Эвристика «жадного» поиска

Так как SA известен своей низкой скоростью сходимости при решении масштабной задачи комбинаторной оптимизации, авторы статьи [2] предлагают «жадный» эвристический подход, чтобы повысить вычислительную эффективность алгоритма. Предлагаемая «жадная» эвристика вводит в алгоритм следующие шаги:

- 1) замена низкосортных блоков раннего периода t на высокосортные блоки более позднего периода $t + 1$;
- 2) перемещение блоков руды на ранний период t , а блоков отходов — на более поздний период $t + 1$, при соблюдении ограничений по переработке, добыче полезных ископаемых и приоритету.

Поиск с запретами для решения проблемы планирования добычи открытым способом

Постановка задачи

В статье [5] представлен метаэвристический подход к решению, основанный на табу-поиске (поиск с запретами), для задачи планирования добычи открытым способом.

При планировании блоков в работе рассматриваются следующие ограничения:

- по запасам: блок может быть добыт не более одного раза в течение планирования;
- по углу наклона: блок не может быть добыт до предшествующих ему блоков.

Действительно, чтобы получить доступ к данному блоку, все вышележащие над ним, называемые его предшественниками, должны быть извлечены;

- по добыче полезных ископаемых: общий вес блоков (отходов и руды), добытых в течение каждого периода, должен быть, по крайней мере, равен минимальному значению, чтобы избежать несбалансированного потока добычи в течение периодов;

- по переработке: общий вес блоков руды, добытых в течение каждого периода, должен быть, по крайней мере, равен минимальному количеству, необходимому для подачи на обоганительную фабрику, но он не должен превышать мощность обоганительной фабрики.

Пусть существует конечный набор возможных сценариев, каждый из которых определяет содержание металла в каждом блоке, и что каждый сценарий имеет равную вероятность возникновения. Тогда данную задачу можно сформулировать как двухэтапную модель стохастического целочисленного программирования [6].

Первый этап. Для каждого периода определяется набор блоков, подлежащих добыче, с соблюдением минимальных и максимальных пределов добычи таким образом, чтобы каждый блок в каждом наборе планировался к выемке ровно один раз после всех своих предшественников. Содержание металла в блоках неизвестно.

Второй этап. Выявляются противоречия в каждом из сценариев. В некоторые периоды, имеющиеся в наличии рудные блоки и требующие переработки, могут иметь общий вес, превышающий мощность обоганительной фабрики, в то время как в другие периоды они могут не соответствовать минимальным требованиям.

В статье используются следующие обозначения для формулировки задачи:

- N — количество блоков, учитываемых при планировании;
- i — индекс блока, $i = 1, \dots, N$;
- T — горизонт планирования;
- t — индекс периода, $t = 1, \dots, T$;
- P_i — набор предшественников блока i ; т. е. блоков, которые должны быть удалены до того, как блок i сможет быть добыт. Если блок p является предшественником блока i , то i называется преемником p ;
- S_i — множество последователей блока i ;
- w_i — вес блока i ;
- $\underline{W}_t$ — минимальный вес, который должен быть добыт в течение периода t (с учетом как блоков руды, так и отходов);
- $\overline{W}_t$ — максимальный вес, который может быть добыт в течение периода t (мощность добывающего оборудования);
- S — количество сценариев, используемых для моделирования содержания по металлу;
- s — индекс сценария, $s = 1, \dots, S$;
- o_{is} — параметр, указывающий группу блока i в соответствии со сценарием s :

$$o_{is} = \begin{cases} 1: & \text{если блок } i \text{ является рудным блоком по сценарию } s; \\ 0: & \text{в противном случае} \end{cases};$$

- m_{is} — содержание металла в блоке i по сценарию s ;

• v_{its} — дисконтированная экономическая стоимость блока i , если он будет добыт в течение периода t и если произойдет сценарий s . Если обозначить через d_1 ставку дисконтирования, а через p_{is} — экономическую ценность блока i по сценарию s , то v_{its} определяется по следующей формуле (11):

$$v_{its} = \frac{p_{is}}{(1+d_1)^t}; \quad (11)$$

• $\underline{Q}_t$ — минимальный вес руды, необходимый для обогатительной фабрики в течение периода t ;
• $\bar{O}_t$ — максимальный вес руды, который может быть переработан на заводе в течение периода t (мощность перерабатывающего завода);
• $c_t^{0-} = \frac{c^{0-}}{(1+d_2)^t}$ — стоимость недостачи единицы продукции, связанная с невыполнением $\underline{Q}_t$ в течение периода t (c^{0-} — недисконтированная стоимость недостачи единицы продукции, а d_2 представляет собой ставку дисконтирования риска);
• $c_t^{0+} = \frac{c^{0+}}{(1+d_2)^t}$ — дополнительные затраты на единицу продукции, понесенные, если общий вес рудных блоков, добытых в течение периода t , превышает $\bar{O}_t$.

Целевая функция (12) включает в себя два элемента для максимизации ожидаемой чистой приведенной стоимости операции по добыче полезных ископаемых и для минимизации ожидаемых затрат. В рассматриваемой задаче авторы принимают, что все сценарии имеют равную вероятность возникновения, и, следовательно, коэффициент $\frac{1}{S}$ представляет вероятность того, что сценарий s произойдет:

$$\max \frac{1}{S} \left\{ \sum_{s=1}^S \sum_{t=1}^T \sum_{i=1}^N v_{its} x_{it} - \sum_{s=1}^S \sum_{t=1}^T (c_t^{0-} d_{ts}^{0-} + c_t^{0+} d_{ts}^{0+} + c_t^{m-} d_{ts}^{m-}) \right\}. \quad (12)$$

Ограничения (13) гарантируют, что каждый блок i добывается не более одного раза в течение горизонта планирования:

$$\sum_{t=1}^T x_{it} \leq 1 \quad i = 1, \dots, N. \quad (13)$$

Соблюдение угла наклона обеспечивается ограничениями (14):

$$x_{it} - \sum_{\tau=1}^t x_{p\tau} \leq 0 \quad i = 1, \dots, N, p \in P_i \quad (14)$$

$t = 1, \dots, T$.

Ограничения (15 и 16) обеспечивают соблюдение требований к уровням добычи в течение каждого периода:

$$\sum_{i=1}^N w_i x_{it} \leq \bar{W}_t \quad t = 1, \dots, T, \quad (15)$$

$$\sum_{i=1}^N w_i x_{it} \geq \underline{W}_t \quad t = 1, \dots, T. \quad (16)$$

Ограничения (17 и 18) связаны с требованиями к уровням обработки (ограничения обработки). Для каждого сценария s цель состоит в том, чтобы общий вес рудных блоков, добытых в течение любого периода t , находился в интервале $[\underline{Q}_t, \bar{O}_t]$. Если оно равно значению, меньшему, чем $\underline{Q}_t$ (соответственно, большему, чем $\bar{O}_t$), то стоимость штрафа за недостачу равна $c_t^{0-} d_{ts}^{0-}$ (соответственно, стоимость штрафа за излишек равна $c_t^{0+} d_{ts}^{0+}$):

$$\sum_{i=1}^N o_{is} w_i x_{it} + d_{ts}^{0-} \geq \underline{Q}_t \quad t = 1, \dots, T \quad (17)$$

$$s = 1, \dots, S,$$

$$\sum_{i=1}^N o_{is} w_i x_{it} - d_{ts}^{0+} \leq \bar{O}_t \quad t = 1, \dots, T \quad (18)$$

$$s = 1, \dots, S.$$

Решение x кодируется как массив длины N , где i -я переменная связана с блоком i , представляет период, на который запланирована выемка блока i . Это означает, что $x_i = t$, если $x_{it} = 0$. Таким образом, если $x_{it} = 0$ для всех $t = 1, \dots, T$ (т. е. блок i не добывается в течение периода), тогда x_i устанавливается равным $(T + 1)$. В этом случае блок i не несет затрат и не приносит дохода.

Для других периодов $t \neq (T + 1)$ допускается нарушения ограничений на добычу за счет добавления к целевой функции соответствующих штрафных издержек.

Первоначальное решение

Метод, используемый для генерации начального решения, представляет собой последовательную эвристическую процедуру, состоящую из $(T + 1)$ основных итераций.

ε — текущий набор блоков во время процедуры, которые имеют право на добычу, поскольку они еще не запланированы, а все их предшественники уже запланированы к выемке.

На каждой основной итерации t ($t = 1, \dots, T$) определенное количество блоков выбирается из ε и планируется на период t .

B_t — набор блоков, запланированных на момент t . Формируется путем последовательного добавления блоков из ε следующим образом:

- 1) случайным образом выбирается блок i из числа блоков в ε и добавляется в B_t ;
- 2) удаление блока i из ε ;
- 3) процесс повторяется до тех пор, пока общий вес блоков в B_t не станет больше или равен

$W_t = \frac{W_t + W_t}{2}$, или ε пусто;

- 4) выполнение новой итерации, переход к следующему периоду $(t + 1)$. Блоки, оставшиеся невыбранными в конце T -й итерации, включаются в B_{T+1} .

Выбор блоков из ε гарантирует, что решение, сгенерированное эвристикой, удовлетворяет ограничениям запаса и ограничениям угла наклона. С другой стороны, в то время как некоторое внимание уделяется удовлетворению ограничений по добыче полезных ископаемых, другие цели и ограничения игнорируются. Цель на данном этапе состоит в том, чтобы быстро найти решение, удовлетворяющее ограничениям.

Используемый вариант процедуры поиска с запретами

Окрестность допустимого решения x генерируется путем сдвига блока i , запланированного в данный момент на период x_i , на другой период $t \neq x_i$. Эта структура позволяет вставлять новый блок в расписание (если $x_i = T + 1$) или удалять существующий блок из него (если $t = T + 1$), или перемещать блок на другой период (если $x_i, t \neq T + 1$).

Сгенерированное новое решение обозначается $x \oplus (i, t)$. Решение принадлежит окрестности x , и сдвиг осуществим тогда и только тогда, когда выполняются ограничения угла наклона, т. е. тогда и только тогда, когда $x_p \in [1, t]$ для всех $p \in P_i$ и $x_s \in [t, T + 1]$ для всех $s \in S_i$ (где P_i и S_i — набор предшественников и преемников блока i соответственно). Следовательно, переход будет возможным, если $t \in [e(x_i), l(x_i)]$, где $e(x_i) = \max_{p \in P_i} \{x_p\}$ (ближайшие предшественники i запланированы в $e(x_i)$) и $l(x_i) = \min_{s \in S_i} \{x_s\}$ (ближайшие преемники i запланированы в $l(x_i)$).

Чтобы легко было отслеживать допустимые сдвиги для каждого блока i , необходимо определить матрицу C из $2 \times N$, где элементы в столбце i указаны как $C_{1i} = e(x_i)$ и $C_{2i} = l(x_i)$. Матрица C обновляется на каждой итерации, поскольку сдвиг, включающий блок i , вызывает изменение информации в столбце i и в столбцах, связанных с ее ближайшими предшественниками и ближайшими преемниками.

Чтобы избежать цикличности, используется краткосрочный список табу (*TL* — *Tabu List*), который запрещает изменять последние смены. Если выполнен переход от x к $x \oplus (i, t)$, то алгоритм добавит блок i на x_i (сдвиг (i, x_i)) в список табу, и в течение следующих θ итераций данный блок при планировании рассматриваться не будет, θ — случайное целое число, выбранное за $[\theta_{\min}, \theta_{\max}]$. Также может быть применен сдвиг табу, если это приводит к решению, лучшему, чем те, что найдены на данный момент, и обозначаемому x_{best} .

Изменение целевой функции $\Delta_x(i, t) = f(x \oplus (i, t)) - f(x)$, связанное с соседним решением $x \oplus (i, t)$, легко оценить, поскольку сдвиг (i, t) влияет только на периоды x_i и t .

Может существовать несколько сдвигов с одинаковым значением наилучшей модификации. Чтобы осуществить из них выбор, используются вторичный критерий выбора, основанный на частотной памяти. В работе вводят $F = [F_{it}]$ матрицу частот размерности $N \times (T + 1)$. Каждая запись F_{it} , связанная с парой блоков i и периодом t , представляет количество раз, когда i было запланировано на t с начала процесса решения. Эта частотная матрица обновляется всякий раз, когда применяется сдвиг (i, t) путем увеличения значения элемента F_{it} на 1. Чтобы разорвать связи, выбирается наилучшее возможное решение $x \oplus (i, t)$ с наименьшим значением частоты F_{it} . Этот вторичный критерий отбора можно рассматривать как стратегию диверсификации, используемую для продвижения поиска в менее исследованные регионы пространства поиска.

Наконец, процедура поиска с запретами завершается, когда число последовательных итераций без улучшения достигает заданного значения. Пусть x^* — текущее наилучшее решение, сгенерированное с помощью табу-поиска. Если x^* лучше, чем решение x_{best} , сгенерированное в предыдущих приложениях поиска (лучшее решение, найденное на данный момент), то x_{best} заменяется на x^* .

В статье было проведено численное сравнение двух различных стратегий диверсификации — *Long-term memory (LTM)* и *Variable neighborhood (VN)*. Первая стратегия использует долговременную память истории поиска, в то время как вторая основана на методе поиска по переменной окрестности. Если прошедшее время меньше, чем $timemax$, эти стратегии могут быть использованы для создания нового начального решения x^0 для повторной инициализации процедуры табу-поиска.

Заключение

В статье рассмотрены два наиболее часто применяемых подхода к локальному поиску для планирования открытых горных работ: подход, основанный на имитации отжига, и подход на основе поиска с запретами.

В работе [2] авторы предложили подход, основанный на имитации отжига с эвристикой «жадного» поиска. Применение данной эвристики в алгоритме имитации отжига, как показано в работе, позволило значительно повысить его эффективность.

В статье [5] предложен метаэвристический метод, основанный на процедуре поиска с запретами. Описаны два варианта метода решения *TS-LTM* и *TS-VN*, полученные путем объединения процедуры поиска по табу с двумя стратегиями диверсификации *LTM* и *VN*, далее было проведено сравнение методов, которое продемонстрировало, что для задач относительно небольших размеров *TS-VN* выдает решения не хуже, чем *TS-LTM*. Однако он не конкурирует с *TS-LTM* в решении более масштабных задач.

Анализ работ показал, что предложенные методы имеют потенциал для улучшения планирования открытых горных работ. Сравнение данных методов на различных задачах позволяет судить о том, что размерность решаемых с их помощью задач выше, чем размерность задач, которые способны решить методы смешанного целочисленного программирования. Таким образом, для решения реальных задач, возникающих на производстве, больше подходят методы, развиваемые в рамках искусственного интеллекта. Хотя они и не гарантируют получение оптимального решения, но позволяют получить достаточно неплохую его аппроксимацию.

Список источников

1. Espinoza D., Goycoolea M., Moreno E., Newman A. MineLib: a library of open pit mining problems // Ann. Oper. Res. 2013. Vol. 206(1). P. 93–114.
2. Danish A., Khan A., Khan M., Waqas A., Salman S. A simulated annealing based approach for open pit mine production scheduling with stockpiling option // Resources Policy. 2021. Vol. 71. p. 102016.

3. Kirkpatrick S., Gelatt C. D., Vecchi M. P. Optimization by simulated annealing // Science. 1983. Vol. 220(4598). P. 671–680.
4. Gershon M. Heuristic approaches for mine planning and production scheduling // Int. J. Min. Geol. Eng. 1987. Vol. 5(1). P. 1–13.
5. Lamghari A., Dimitrakopoulos R. A diversified Tabu search approach for the open-pit mine production scheduling problem with metal uncertainty // European Journal of Operational Research. 2012. Vol. 222(3). P. 642–652.
6. Birge J. R., Louveaux F. Introduction to Stochastic Programming. New York: Springer-Verlag, 1997.

References

1. Espinoza D., Goycoolea M., Moreno E., Newman A. *MineLib: a library of open pit mining problems*, 2013. Ann. Oper. Res., vol. 206, no. 1, pp. 93–114.
2. Danish A., Khan A., Khan M., Waqas A., Salman S. *A simulated annealing based approach for open pit mine production scheduling with stockpiling option*, 2021 Resources Policy, vol. 71, p. 102016.
3. Kirkpatrick S., Gelatt C. D., Vecchi M. P. *Optimization by simulated annealing*, 1983. Science, vol. 220, no. 4598, pp. 671–680.
4. Gershon M. *Heuristic approaches for mine planning and production scheduling*, 1987. Int. J. Min. Geol. Eng., vol. 5(1), pp. 1–13.
5. Lamghari A., Dimitrakopoulos R. *A diversified Tabu search approach for the open-pit mine production scheduling problem with metal uncertainty*. European Journal of Operational Research, 2012, vol. 222, no. 3, pp. 642–652.
6. Birge J. R. *Louveaux F. Introduction to Stochastic Programming*. Springer-Verlag, New York, 1997, 512 p.

Информация об авторах

А. В. Шестаков — стажер-исследователь;

А. А. Зуенко — кандидат технических наук, ведущий научный сотрудник.

Information about the authors

A. V. Shestakov — Intern Researcher;

A. A. Zuenko — Candidate of Science (Tech.), Leading Researcher of the Putilov Institute for Informatics and Mathematical Modeling Kola Science Centre of the Russian Academy of Sciences.

Статья поступила в редакцию 09.10.2023; одобрена после рецензирования 01.11.2023; принята к публикации 08.11.2023.
The article was submitted 09.10.2023; approved after reviewing 01.11.2023; accepted for publication 08.11.2023.